Univerza v Ljubljani

Fakulteta za računalništvo in informatiko

Borut Batagelj

Prepoznavanje človeških obrazov s pomočjo hibridnega sistema

DOKTORSKA DISERTACIJA

Ljubljana, 2007

Mentor: prof. dr. Franc Solina

Prepoznavanje človeških obrazov s pomočjo hibridnega sistema

Borut Batagelj

Mentor: Franc Solina

Povzetek

Prepoznavanje človeških obrazov je postalo v zadnjih letih eno najpomembnejših, pa tudi najuspešnejših področij analize in razumevanja slik. Sedanji sistemi so v glavnem omejeni le na iskanje in prepoznavo in v primeru, da ne prepoznajo obraza, javijo, da je oseba neznana. Poleg tega pa je bilo predlaganih že veliko metod za analizo človeških obrazov. V doktorski disertaciji na podlagi celovite analize vseh obstoječih metod sestavimo celovit modularni sistem za prepoznavanje, klasifikacijo in analizo človeških obrazov. Uporabnost tako razširjenega sistema predstavimo na področju pametnega oglaševanja.

Predlaganih je bilo že veliko metod za prepoznavanje obrazov in različne raziskovalne skupine po svetu poročajo o zelo različnih in pogosto tudi nasprotujočih si rezultatih uspešnosti le-teh. Raziskovalci namreč preizkušajo metode na različnih podatkovnih bazah obrazov. V primeru, če uporabijo iste baze, pa največkrat uporabijo drugačne učne množice, postopke normalizacije ali tehnike primerjave izločenih značilk, kar tudi vpliva na natančnost razpoznave. Zaradi tega izvedemo neodvisno primerjavo najpomembnejših metod za prepoznavanje obrazov (PCA, LDA, ICA in Gaborjevi valčki). Preizkuse izvedemo s pomočjo enotnega postopka normalizacije in implementacije algoritmov na dveh javno dostopnih podatkovnih bazah FERET in AR. Sistematično preizkusimo različne postopke normalizacije in različne načine merjenja razdalj pri postopku primerjave značilk. Različne kombinacije metod za prepoznavanje obrazov in merjenje razdalj preizkusimo na standardnih testnih množicah za nalogo identifikacije na zaprti množici in sicer za najboljše ujemanje (rang 1) in pri višjih rangih. Na podlagi preizkusov ugotovimo, da delujejo metode različno pri različnih rangih ter, da nobena kombinacije ne deluje najboljše za celotno podatkovno bazo. Ugotovimo pa lahko, katere kombinacije so najboljše za določeno testno množico.

Ker je tudi sam postopek normalizacije največkrat različen, smo želeli raziskati, kako normalizacija dejansko vpliva na uspešnost prepoznavanja obrazov. Poleg različnih postopkov normalizacije smo preizkusili tudi prepoznavo obrazov samo na posameznih delih obraz in poskušali ugotoviti, kateri del obraza je za prepoznavanje najpomembnejši.

Na podlagi preizkusov določimo najboljše kombinacije metod in razdalj pri različnih rangih in jih združimo v hibridni sistem. Hibridni sistemi so v zadnjem času postali zelo razširjeni tudi na področju biometrije, še posebej pri uporabi več mo-

dalnosti. Tako lahko s pravo kombinacijo različni biometričnih podatkov izboljšamo učinkovitost biometričnih sistemov za identifikacijo. Predlagamo združitev različnih metod v vzporedne in zaporedne sisteme, ki jih tudi preizkusimo na enotni podatkovni bazi. S pravilnim združevanjem pravih metod zgradimo hibridni sistem, ki je pri prepoznavanju človeških obrazov učinkovitejši in predvsem časovno zmogljivejši.

Ključne besede: biometrija, iskanje obrazov, prepoznavanje obrazov, analiza obrazov, kategorizacija obrazov, normalizacija obraznih slik, mere razdalj, metode osnovane na videzu, PCA, LDA, ICA, Gaborjevi valčki, hibridne metode, hibridni sistemi, pametno oglaševanje.

Human Face Recognition using a Hybrid System

Borut Batagelj

Supervisor: Franc Solina

Abstract

Face recognition is one of the most successful applications of image analysis and understanding and has gained much attention in recent years. Existing systems for face recognition generally consist of a detection and a recognition module. In addition there are systems for categorization and analysis of human faces. On the basis of a complete survey of existing methods, we present in this dissertation a modular system for recognition, classification and analyses of faces. We show the usage of this extended system in the field of smart advertising.

Various methods for face recognition were proposed and research groups report different and often contradictory results when comparing them. Researchers often test their methods on different databases, even their own private databases; they use even different training sets, normalization and comparison techniques, which also has an influence on the accuracy of the recognition. One of the aims of this dissertation is to present an independent, comparative study of the four most popular methods for face recognition (PCA, LDA, ICA and Gabor wavelets) under the same working conditions regarding normalization and algorithm implementation. Tests are done on two standard face databases: FERET and AR. We systematically test different normalization processes and all possible distance metrics for features comparison. For the case of a closed-set identification, we test the face recognition methods in combination with all possible distance metrics for the best match (rank 1) and for higher ranks on the standard test sets. We conclude that the metric showing the best results at rank one did not always yield the best results at higher ranks and that no particular method-metric combination is the best across all standard test sets, however we can choose an appropriate combination for a specific task.

Because the normalization preprocessing is often different and not clearly defined among different reported research tests, we test how different normalization methods influence the performance of different face recognition systems. We also test how masking different parts of the face influences the recognition to find out which part of the face is the most significant for recognition.

We use the optimal combinations of methods and metrics to build the hybrid systems in a serial and a parallel manner. Hybrid systems are becoming very important also in biometrics. Most popular systems are multi-modal biometric systems which use fusion to combine more biometrics samples. In this case, we design the hybrid system to work with different recognition methods on the same biometric

data. The results on the same standard databases show that with the right combination of methods we can get a better performance and lower time complexity of the face recognition system.

Key words: biometrics, face detection, face recognition, face analysis, face categorization, face image normalization, metrics, subspace methods, PCA, LDA, ICA, Gabor wavelets, hybrid methods, hybrid systems, smart advertising.

Kazalo

Povzetek iii
Abstract v

1 Uvod **1**
1.1 Sistem za prepoznavanje obrazov 4
1.2 Razširjen sistem za prepoznavanje in analizo obrazov 6
1.3 Motivacija in cilji 7
1.4 Pričakovani prispevki k znanosti 8
1.5 Struktura disertacije 9

2 Pregled metod za iskanje in prepoznavo obrazov **11**
2.1 Uvod 11
2.2 Pregled metod za iskanje obrazov 11
2.2.1 Na porazdelitvi temelječa metoda – Sung in Poggio, 1997 . . . 13
2.2.2 Na nevronskih mrežah temelječa metoda – Rowley et al., 1998 14
2.2.3 Metoda podpornih vektorjev – Osuna 1997 16
2.2.4 Metoda naivnega Bayesovega klasifikatorja – Schneiderman in Kanade, 1998 16
2.2.5 Na algoritmu AdaBoost temelječa metoda – Viola in Jones, 2001 17
2.2.6 Bayesova metoda diskriminantnih značilk – Liu 2003 19
2.2.7 Rezultati preizkusa metod za iskanje obrazov 22
2.3 Pregled metod za prepoznavanje obrazov 23
2.3.1 Metode na osnovi celotnega videza obraza 24
2.3.2 Metode na osnovi značilk in geometrije 25
2.3.3 Hibridne metode 26
2.3.4 Hibridni sistemi 29
2.4 Povzetek poglavja 33

3 Podatkovne baze in metodologije preizkusov **35**
3.1 Uvod 35
3.2 Podatkovne baze obrazov 35
3.2.1 Podatkovna baza FERET in njena metodologija 38
3.2.2 Podatkovna baza AR in njena metodologija 42

3.3 Povzetek poglavja 43

4 Pregled standardnih preizkusov FERET in FRVT 45
4.1 Uvod 45
4.2 Standardni kazalci uspešnosti 45
4.2.1 Identifikacija na odprti množici 46
4.2.2 Verifikacija 49
4.2.3 Identifikacija na zaprti množici 51
4.3 Preizkusna protokola FERET in FRVT 2002 51
4.4 Preizkusi FERET in FRVT 53
4.4.1 Preizkusi FERET 54
4.4.2 Preizkus FRVT 2000 55
4.4.3 Preizkus FRVT 2002 56
4.5 Povzetek poglavja 57

5 Normalizacija slik obrazov 59
5.1 Uvod 59
5.2 Slike obrazov 60
5.3 Rotacija slike obraza 61
5.4 Sprememba velikosti slike obraza 62
5.5 Rezanje slike obraza 62
5.6 Maskiranje slike obraza 64
5.7 Svetlobna normalizacija 65
5.8 Različna normalizacija in različni deli obraza 66
5.9 Povzetek poglavja 68

6 Metode osnovane na videzu 69
6.1 Tehnični izzivi in obrazni podprostori 69
6.1.1 Obrazni podprostori 71
6.2 Metoda glavnih komponent 73
6.2.1 Matematična definicija 73
6.2.2 Lastnosti metode PCA 77
6.2.3 Enostaven 2D primer 81
6.2.4 Uporaba metode PCA za prepoznavanje obrazov 83
6.3 Linearna diskriminantna analiza 91
6.3.1 Matematična definicija 91
6.3.2 Enostaven 2D primer 96
6.3.3 Uporaba metode LDA za prepoznavanje obrazov 98
6.4 Metoda neodvisnih komponent 105
6.4.1 Prva arhitektura metode ICA – ICA I 107
6.4.2 Druga arhitektura metode ICA – ICA II 112
6.5 Povzetek poglavja 117

7 Metoda Gaborjevih valčkov 119
7.1 Uvod . . . 119
7.2 Gaborjevi valčki . . . 119
7.2.1 Matematična definicija Gaborjevih valčkov . . . 120
7.3 Uporaba Gaborjevih filtrov za prepoznavanje obrazov . . . 127
7.4 Povzetek poglavja . . . 130

8 Različne mere razdalj metode najbližjega soseda 133
8.1 Uvod . . . 133
8.1.1 Različne mere razdalj . . . 134
8.2 Povzetek poglavja . . . 138

9 Preizkusi metod in hibridni sistemi 139
9.1 Uvod . . . 139
9.1.1 Metodologija preizkusov . . . 139
9.1.2 Postopek učenja . . . 140
9.1.3 Postopek prepoznave . . . 142
9.2 Rezultati nad podatkovno bazo FERET . . . 143
9.2.1 Potek preizkusov posamezne metode . . . 144
9.2.2 Rezultati preizkusov vseh metod na bazi FERET . . . 148
9.3 Rezultati nad podatkovno bazo AR . . . 154
9.3.1 Določitev optimalnega števila komponent . . . 154
9.3.2 Rezultati preizkusov vseh metod na bazi AR . . . 155
9.4 Primerjava rezultatov različnih metod in baz . . . 159
9.4.1 Rezultati za rang 1 . . . 159
9.4.2 Rezultati za višje range . . . 162
9.4.3 Zaključek in ugotovitve . . . 167
9.5 Rezultati preizkusov pri različnih normalizacijah . . . 169
9.5.1 Maskiranje in svetlobna normalizacija . . . 169
9.5.2 Različne vrednosti parametra $d0$ in $e0$ . . . 171
9.5.3 Različni deli obraza . . . 172
9.5.4 Zaključek in ugotovitve . . . 176
9.6 Vzporedni in zaporedni hibridni sistemi . . . 176
9.6.1 Vzporedni hibridni sistem . . . 177
9.6.2 Preizkus in rezultati vzporednega hibridnega sistema . . . 179
9.6.3 Zaporedni hibridni sistem . . . 181
9.6.4 Preizkus in rezultati zaporednega hibridnega sistema . . . 184
9.6.5 Zaključek in ugotovitve . . . 185
9.7 Povzetek poglavja . . . 186

10 Uporaba sistemov za prepoznavanje obrazov 187
10.1 Uvod . . . 187
10.2 Področja uporabe in tipične aplikacije . . . 188

10.2.1 Identifikacija obraza 188
10.2.2 Kontrola dostopa 188
10.2.3 Varnost 190
10.2.4 Nadzor 191
10.2.5 Pametne kartice 191
10.2.6 Organi pregona 192
10.2.7 Zbirke digitalnih fotografij 192
10.2.8 Upravljanje z večpredstavnostnimi vsebinami 194
10.2.9 Interakcija med človekom in računalnikom 195
10.2.10 Druga področja uporabe 195
10.2.11 Omejitve trenutnih sistemov 196
10.3 Primer uporabe: pametni oglasi 197
10.4 Povzetek poglavja 198

11 Zaključek in nadaljnje delo 199
11.1 Zaključek 199
11.2 Prispevki k znanosti 201
11.3 Nadaljnje delo 202

Zahvala 203

Izjava 205

Tabele

2.1 Rezultati preizkusa različnih metod za iskanje obrazov. 23

3.1 Seznam podatkovnih baz obrazov za preizkus uspešnosti prepoznavanja obrazov. 37
3.2 Seznam podatkovnih baz za preizkus uspešnosti prepoznavanja različnih obraznih izrazov. 37
3.3 Pregled podatkovnih baz za preizkus uspešnosti iskanja obrazov. . . . 38
3.4 Seznam različnih testnih množic podatkovne baze FERET. 39
3.5 24 kategorij slik znotraj podatkovne baze FERET. 40
3.6 Število slik galerije in posamezne testne množice v podatkovni bazi FERET. 41

4.1 Rezultati uspešnosti treh najboljših algoritmov na preizkusu FERET. 55

9.1 Rezultati preizkusa metode PCA nad podatkovno bazo FERET. . . . 146
9.2 Rezultati uspešnosti metode PCA na bazi FERET. 148
9.3 Rezultati uspešnosti metode PCA na bazi FERET. 149
9.4 Rezultati uspešnosti metode PCA brez prvih treh komponent na bazi FERET. 149
9.5 Rezultati uspešnosti metode LDA+PCA na bazi FERET. 151
9.6 Rezultati uspešnosti direktne metode LDA na bazi FERET. 151
9.7 Rezultati uspešnosti metode ICA1 na bazi FERET. 152
9.8 Rezultati uspešnosti metode PCA1 na bazi FERET. 152
9.9 Rezultati uspešnosti metode ICA2 na bazi FERET. 153
9.10 Rezultati uspešnosti metode PCA2 na bazi FERET. 153
9.11 Rezultati uspešnosti metode Gaborjevih valčkov na bazi FERET. . . 153
9.12 Rezultati uspešnosti metode PCA na bazi AR. 155
9.13 Rezultati uspešnosti metode PCA brez prvih treh komponent na bazi AR. 155
9.14 Rezultati metode LDA+PCA na bazi AR. 156
9.15 Rezultati uspešnosti direktne metode LDA na bazi AR. 156
9.16 Rezultati uspešnosti metode ICA1 na bazi AR. 156
9.17 Rezultati uspešnosti metode PCA1 na bazi AR. 157
9.18 Rezultati uspešnosti metode ICA2 na bazi AR. 157

9.19 Rezultati uspešnosti metode PCA2 na bazi AR. 157
9.20 Rezultati uspešnosti metode Gaborjevih valčkov na bazi AR. 158
9.21 Rezultati najboljših kombinacij metod in razdalj za posamezno testno množico podatkovnih baz FERET in AR. 159
9.22 Najboljši rezultati uspešnosti prepoznave obrazov za posamezno testno množico. 167
9.23 Najboljše kombinacije metode in razdalje za posamezne testne množice pri višjih rangih. 168
9.24 Rezultati uspešnosti različnih vzporednih hibridnih sistemov nad posameznimi testnimi množicami. 180
9.25 Uspešnost metode PCA pri različnih rangih za testne množice baze FERET. 183
9.26 Uspešnost metode PCA pri različnih rangih za testne množice baze AR. 183
9.27 Časi prepoznave 100-tih obrazov s pomočjo metode PCA, metode Gaborjevih valčkov in zaporednega hibridnega sistema. 184
9.28 Rezultati uspešnosti zaporednega hibridnega sistema v primerjavi z rezultati metode PCA in Gaborjeve metode. 185

Slike

1.1 Primer iluzije Thatcher avtorja Thompsona. 2
1.2 Primerjava različnih biometričnih podatkov. 4
1.3 Tri naloge sistema za prepoznavanje obrazov. 5
1.4 Sistem za avtomatsko prepoznavanje obrazov. 5
1.5 Razširjen sistem za prepoznavanje, kategorizacijo in analizo obraza. . 6

2.1 Porazdelitvena modela za razred obrazov in neobrazov. 13
2.2 Diagram nevronske mreže za iskanje obrazov na sliki. 15
2.3 Štirje tipi pravokotnih značilk za iskanje obrazov na slikah. 18
2.4 Izračun vsote slikovnih elementov znotraj območja za integralno sliko. 19
2.5 Kaskada n-tih močnih klasifikatorjev. 21
2.6 Posamezne komponente vektorja značilk za sliko povprečnega obraza. 21
2.7 Rezultati ujemanja slik metode na podlagi videza značilk. 27
2.8 Jedra metode LFA za različne lokacije na sliki obraza. 28
2.9 Shema sistema za prepoznavanje obrazov, ki uporablja prilagodljiv model na osnovi videza. 29
2.10 Vzorci na frontalnem obrazu in obrazu s profila vzporednega hibridnega sistema. 30
2.11 Shema hibridnega sistema za prepoznavanje obrazov, ki združuje tri statistične metode: PCA, ICA in LDA in dve odločitveni strategiji: pravilo vsote in mrežo RBF. 31
2.12 Shemi različnih arhitektur primerkov RBF mrež zaporednega hibridnega sistema. 32
2.13 Zaporedni hibridni sistem, ki kombinira nevronsko mrežo in skrite Markove modele. 33

3.1 Slike iste osebe iz različnih testnih množic podatkovne baze FERET. 40
3.2 Primeri slik različno obrnjene glave iste osebe iz podatkovne baze FERET 41
3.3 Primeri slik iste osebe s polnega, polovičnega in četrtinskega profila iz podatkovne baze FERET. 41
3.4 Primeri slik iste osebe podatkovne baze AR. 42

4.1 Primer krivulje ROC za primer identifikacije na odprti množici. . . . 48

5.1 Trije primeri nenormaliziranih slik obrazov iz podatkovne baze FERET. 60
5.2 Trije primeri nenormaliziranih slik obrazov iz podatkovne baze AR. . 60
5.3 Primer rotacije slike obraza na podlagi koordinat oči. 61
5.4 Primer zmanjšanja slike obraza na podlagi razdalje med očmi. 63
5.5 Primer različno rezanega obraza iz slike. 63
5.6 Primer odrezanega obraza iz slike glede na pozicijo oči. 64
5.7 Primer maskiranja obrezane slike obraza. 64
5.8 Primer normaliziranega obraza s pomočjo postopka izravnave histograma. 65
5.9 Primer normaliziranega obraza na osnovi povprečne vrednosti in standardne deviacije. 65
5.10 Primeri normaliziranega obraza z različnimi razdaljami med očmi. . . 66
5.11 Primeri normaliziranega obraza z različnimi pozicijami oči. 66
5.12 Primeri delno zakritega normaliziranega obraza. 67
5.13 Primeri posameznih delov normaliziranega obraza. 67

6.1 Mnogoterost obraza in neobraza ter mnogoterosti različnih oseb. . . . 69
6.2 Variacije istega obraza zaradi različnih pogojev. 70
6.3 Izziva pri prepoznavanju obrazov: slabo naučen klasifikator in evklidska razdalja kot neučinkovita ločnica med različnima osebama. 71
6.4 Vektorska predstavitev slike. 72
6.5 Tipični lastni spekter in energija. 79
6.6 Delovanje metode PCA za primer 2D podatkov. 82
6.7 Primeri slik obrazov iz učne množice. 85
6.8 Slika povprečnega obraza. 85
6.9 Primeri prvih petih in zadnjih petih lastnih obrazov. 87
6.10 Lastni spekter za primer obraznih slik in razdelitev na dva ortogonalna podprostora. 88
6.11 Originalna slika obraza in njena projekcija v obrazni prostor. 89
6.12 Predstavitev slike obraza s posameznimi prispevki lastnih obrazov, ki jih določajo glavne komponente. 89
6.13 Delovanje metode LDA za primer 2D podatkov. 98
6.14 Primeri prvih petih lastnih obrazov metode LDA in kombinacije metod PCA in LDA. 100
6.15 Model slepega ločevanja izvorov. 105
6.16 Postopek iskanja statistično neodvisnih baznih slik s pomočjo metode ICA. 107
6.17 Predstavitev prve arhitekture metode ICA na slikah obrazov. 108
6.18 Predstavitev slike obraza s pomočjo prve arhitekture metode ICA. . . 111
6.19 Primeri prvih petih lastnih obrazov metode ICA I. 111
6.20 Primeri prvih petih lastnih obrazov metode PCA I. 112
6.21 Postopek iskanja statistično neodvisnih koeficientov s pomočjo metode ICA. 113

6.22 Predstavitev druge arhitekture metode ICA na slikah obrazov. 113
6.23 Predstavitev slike obraza s pomočjo druge arhitekture metode ICA. . 115
6.24 Primeri prvih petih lastnih obrazov metode ICA II. 115
6.25 Primeri prvih petih lastnih obrazov metode PCA II. 116

7.1 Primer predstavitve Gaborjevega valčka v 3D prostoru. 120
7.2 Primer konvolucije z Gaborjevim valčkom. 123
7.3 Primer konvolucije slike z dvodimenzionalnim Gaborjevim valčkom. . 125
7.4 Primeri valčkov različnih orientacij. 126
7.5 Primeri valčkov različnih valovnih dolžin. 126
7.6 Primeri valčkov z različnimi fazami. 126
7.7 Primeri valčkov z različnimi velikostmi Gaussa. 127
7.8 Primeri valčkov z različnimi razmerji Gaussa. 127
7.9 Realni del množice Gaborjevih valčkov. 128
7.10 Predstavitev slike obraza s pomočjo Gaborjevih valčkov. 129
7.11 Primer Gaborjevega vektorja značilk za sliko obraza. 130

9.1 Primeri normaliziranih slik obrazov podatkovnih baz FERET in AR. 141
9.2 Postopek učenja in preslikava znanih obrazov. 142
9.3 Postopek prepoznavanja testne slike s pomočjo metod osnovanih na videzu. 143
9.4 Postopek prepoznavanja testne slike s pomočjo metode Gaborjevih valčkov. 143
9.5 Določitev števila komponent na podlagi metode PCA in baze FERET. 145
9.6 Rezultati uspešnosti prepoznave pri različnih merah razdalj za metodo PCA na bazi FERET. 145
9.7 Rezultati uspešnosti metode PCA pri višjih rangih na testnih množicah baze FERET. 147
9.8 Razlika uspešnosti prepoznave za metodo PCA in spremenjeno PCA brez prvih treh komponent. 150
9.9 Določitev števila komponent za metodo PCA na bazi AR. 154
9.10 Rezultati najboljših kombinacij različnih metod za posamezno testno bazo FERET in AR. 161
9.11 Krivulja CMS za najboljše kombinacije metod in razdalj testne množice $fafb$. 162
9.12 Krivulja CMS za najboljše kombinacije metod in razdalj testne množice $fafc$. 163
9.13 Krivulja CMS za najboljše kombinacije metod in razdalj testne množice $dup1$. 164
9.14 Krivulja CMS za najboljše kombinacije metod in razdalj testne množice $dup2$. 164
9.15 Krivulja CMS za najboljše kombinacije metod in razdalj testne množice fb_{AR}. 165

9.16 Krivulja CMS za najboljše kombinacije metod in razdalj testne množice fc_{AR}. 166
9.17 Primer maskirane slike obraza in različno svetlobno normaliziranih slik. 169
9.18 Primerjava rezultatov uspešnosti na maskiranih slikah obrazov. 170
9.19 Primerjava uspešnosti različno svetlobno normaliziranih slik obrazov. 170
9.20 Primeri različno normalizirane slike obraza z različnimi razdaljami med očmi. 171
9.21 Primerjava uspešnosti prepoznave obrazov z različnimi razdaljami med očmi. 172
9.22 Primeri različno normalizirane slike obraza z različno zamaknjenimi obrazi. 172
9.23 Primerjava uspešnosti različno zamaknjenih obrazov. 173
9.24 Primeri različnih delov obraza: desna, leva stran, zgornji, srednji in spodnji del. 173
9.25 Primerjava uspešnosti za desno in levo stran obrazov. 174
9.26 Primerjava uspešnosti za zgornji, srednji in spodnji del obrazov. . . . 174
9.27 Primeri posameznih delov obraza: čelo+pričeska, oči, nos in usta. . . 175
9.28 Primerjava uspešnosti za posamezen del obrazov. 175
9.29 Tri različne metodologije združevanja vzporednega hibridnega sistema. 178
9.30 Rezultati uspešnosti vzporednega hibridnega sistema. 181
9.31 Shema zaporednega hibridnega sistema. 182
9.32 Rezultati uspešnosti zaporednega hibridnega sistema. 185

Algoritmi

1 Algoritem učenja AdaBoost 20
2 Metoda glavnih komponent – PCA 76
3 Postopek učenja z metodo PCA 87
4 Postopek prepoznave z metodo PCA 90
5 Linearna diskriminantna metoda – LDA 99
6 Metoda Fisherjevih obrazov – LDA+PCA 101
7 Direktna metoda LDA – DLDA 103

Poglavje 1

Uvod

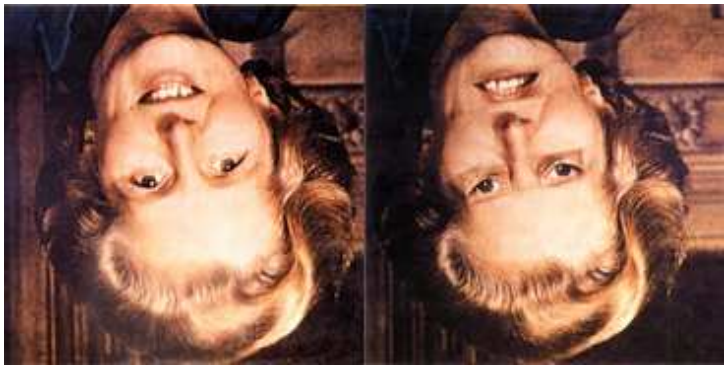
V zadnjih letih je prepoznavanje človeških obrazov pritegnilo veliko zanimanja raziskovalcev z najrazličnejših področij, kot so biometrija, razpoznavanje vzorcev in računalniški vid [13, 115]. V raziskave prepoznavanja človeških obrazov so se aktivno vključili tudi raziskovalci s področja računalniške grafike in strojnega učenja. Skupni interes raziskovalcev s tako različnih računalniških področij motivira izredna sposobnost človeka za prepoznavanje oseb in dejstvo, da je razumevanje človeške aktivnosti prvotnega pomena tako v vsakdanjem življenju kot tudi v navideznem svetu. Poleg tega obstaja na področju komercialnih, varnostnih in forenzičnih sistemov velika potreba po tehnologiji prepoznave obrazov. Področje uporabe takšnih sistemov je zelo široko in vključuje sisteme za avtomatsko video kontrolo, kontrolo vstopa, identifikacijo, multimedijsko komunikacijo in upravljanje z večpredstavnimi vsebinami. Zaradi vedno zmogljivejše in cenejše opreme za procesiranje in zajemanje slike pa vstopajo sistemi za prepoznavanje obrazov vedno bolj tudi v naše vsakdanje življenje. Med te sisteme vključujemo napredne uporabniške vmesnike, ki omogočajo kontrolo dostopa in interakcijo med računalnikom in človekom s pomočjo obraza. Ti sistemi vključujejo poleg same prepoznave obrazov tudi nalogi iskanja in sledenja obrazov.

S področjem prepoznavanja obrazov se ukvarjajo tudi druge discipline, kot so psihologija, plastična kirurgija, preiskovalna policija ter pravni izvedenci. Znanstveniki s teh področij gledajo na problem prepoznavanja s svojega zornega kota. Kakorkoli, izgleda, da je problem prepoznavanja obrazov v človeškem zaznavnem sistemu rešen. Prav ta sposobnost človeka za razlikovanje različnih oseb je inženirje povezala z znanstveniki s področja psihologije zaznavanja in nevrologije. Ravno ugotovitve znanstvenikov s teh področij so lahko v veliko pomoč pri gradnji in razumevanju računalniških sistemov za prepoznavanje človeških obrazov. V nadaljevanju bomo predstavili kratke izsledke raziskav človeškega zaznavnega sistema na področju prepoznavanja obrazov [88].

Prvo vprašanje, ki se nam pojavi je: ali se mora človek prepoznavanj obrazov naučiti ali je ta sposobnost predhodno zapisana v človeških možganih? V tem kontekstu sta Johnston in Ellis [45] pokazala, da dojenček star komaj 10 minut pokaže

večji interes za obrazu podoben vzorec, kot za kakšen drugi vzorec. Po 48 urah življenja ima dojenček že sposobnost prepoznati obraz mame in 10 tednov starega dojenčka najbolj stimulira prav obrazni vzorec. Sposobnost prepoznavanja obrazov se nato drastično izboljšuje do približno 12 leta starosti. Poizkusi, pri katerih so spreminjali obrazne značilke s pomočjo gibanja in različnih izrazov, so pokazali, da so na otroke stare pod 11 let močneje vplivale posamezne obrazne značilke, kot na odrasle, ki so večjo pozornost namenili strukturi in celotnemu videzu obraza.

Dokazano je bilo tudi, da človek lažje prepozna osebe, ki pripadajo njegovi rasni skupini, torej učenje obrazov pri človeku bazira na določeni rasi ljudi, ki jih oseba srečuje v svojem življenju. Dokazano je bilo tudi, da je prepoznavanje nagnjenih in obrazov iz negativa slike težje (prepoznava se poslabša za 10%). Thompson [96] je predstavil zanimivo iluzijo (angl. *Thatcher illusion*), da bi pokazal, da nimamo enake sposobnosti zaznavanja posameznih značilk, če gledamo na glavo obrnjeno sliko obraza. Slika 1.1 prikazuje primer takšne iluzije. Če pogledamo obe sliki obraza obrnjenega na glavo se nam zdita oba obraza običajna. Vendar, če pogledamo obraza pravilno obrnjena vidimo, da je levi obraz prav zastrašujoč. To nas vodi do spoznanja, da klasične strategije, ki se uporabljajo pri razpoznavanju objektov, morda niso ravno najprimernejše za prepoznavanje tako specifičnih objektov, kot je to obraz.



Slika 1.1: Primer iluzije Thatcher avtorja Thompsona. Predstavitev obraza z obrnjenimi značilkami obraza se pri pogledu pravilno obrnjenega obraza spremeni iz običajnega v zastrašujoč obraz.

Dodatno je bilo pokazano, da je lahko pri prepoznavanju obrazov zelo uporabno premikanje, ker deluje kot dodatna pomoč vizualnemu sistemu pri konstrukciji 3D strukture obraza. Študije bolnikov, ki so imeli poškodovane možgane, so pokazale, da skrbi v naših možganih za prepoznavanje obrazov poseben center. Pacienti s tako imenovano prosopagnozo (angl. *prosopagnosia*) namreč niso zmožni prepoznati obraza, medtem ko lahko druge objekte normalno prepoznajo.

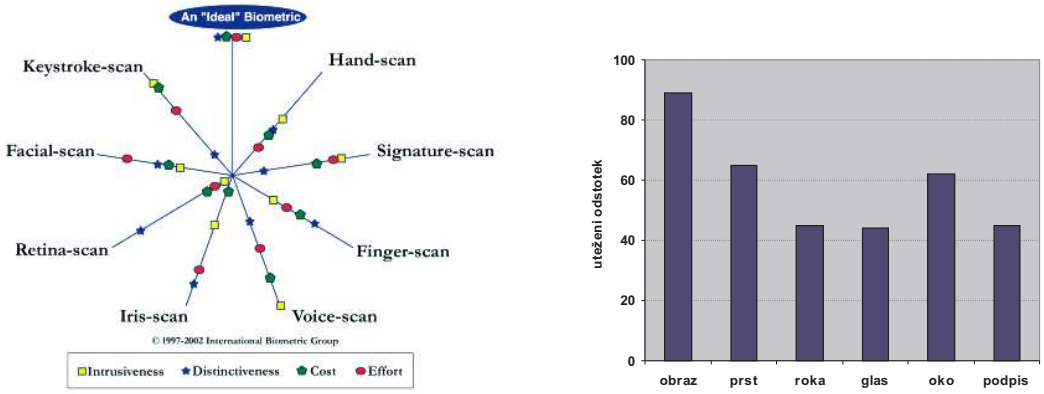
Druge študije psihologov so pokazale izredno zanimivo dojemljivost človeka, ki hitreje prepozna človeka iz njegove karikature kot iz originalne slike obraza. Študije

so pokazale tudi izredno sposobnost prepoznave spola z uspešnostjo večjo od 95%. Primerjava s prepoznavanjem obraza je pokazala, da sta sposobnost prepoznavanja obraza in spola časovno neodvisna, kar dokazuje, da gre za ločeni nalogi v vizualnemu sistemu človeka.

Obraz je zelo pomemben biometrični podatek človeka, zaradi česar se veliko uporablja tudi v sistemih za kontrolo in nadzor. Za vstop v nekatere države so že uvedli kontrolo prstnega odtisa in obraza. Tudi pri nas so že uvedli potne liste, ki vsebujejo na posebnem čipu sliko obraza. Biometrija omogoča avtomatsko identifikacijo osebe s pomočjo njegovih fizioloških karakteristik ali karakteristik obnašanja. Fiziološki biometrični podatki so biološke ali kemične karakteristike osebe, ki so lahko prirojene ali naravno prirasle, medtem ko določajo biometrični podatki obnašanja vedenje, ki se ga je oseba naučila ali ga je pridobila. V prvo skupino uvrščamo telesni vonj, DNA, obraz, prstni odtis, geometrijo roke, šarenico, utrip, mrežnico, teksturo dlani, odtis nohta, odtis jezika, struktura ušesa, žile na roki, itd. Med biometrične podatke obnašanja pa uvršamo pisavo, podpis, glas, hojo, frekvenco tipkanja, itd. V to skupino lahko uvrstimo tudi obraz, če nanj gledamo kot na dinamično strukturo. Tako se lahko za ločevanje med različnimi osebami upošteva tudi nasmeh, premikanje ustnic itd.

Biometrična tehnologija je postala temelj na področju reševanja problemov visoke varnosti identifikacije in osebne verifikacije. V primerjavi z običajnimi metodami za identifikacijo in verifikacijo, ki bazirajo na osebnih geslih in PIN kodah, ponuja biometrična tehnologija določene prednosti. Kot prvo so biometrični podatki osebnostne značilnosti, medtem ko lahko gesla uporablja ali ukrade druga, nepooblaščena oseba. Poleg tega je biometrični podatek zelo priročen, ker ne rabimo nič prenašati ali si zapomniti. Biometrični podatki postajajo tudi vedno bolj natančni in cenejši.

Pred ostalimi biometričnimi podatki ima obraz določene prednosti: je naravno prirojen, sistem za prepoznavo je nevsiljiv in zelo preprost za uporabo. Različni biometrični podatki so zaradi svojih lastnosti primerni za različne naloge. Med seboj se razlikujejo po različni stopnji vsiljivosti, natančnosti, ceni in naporu pri zajemanju. Na sliki 1.2 (a) vidimo karakteristike nekaterih najbolj razširjenih biometričnih podatkov. Vidimo, da idealen biometrični podatek ne obstaja, s pravo kombinacijo biometričnih podatkov pa se mu lahko zelo približamo. Med šestimi biometričnimi podatki: obrazom, prstnim odtisom, roko, glasom, očmi in pisavo, se je obraz uvrstil na prvo mesto za namene potovalnih dokumentov (angl. *Machine Readable Travel Documents*, MRTD) [32] (slika 1.2 (b)). Pri primerjavi so upoštevali šest kriterijev: (1) zahtevnost registracije in (2) ponovne obnovitve podatkov, (3) strojna podpora, (4) redundantnost sistema, (5) dojemljivost uporabnikov in (6) zahteve po pomnilnem prostoru in učinkovitost. Med pomembnejšimi lastnostmi biometričnega podatka obraza pa je nedvomno prav zmožnost zbiranja podatkov brez sodelovanja osebe.



(a) (b)

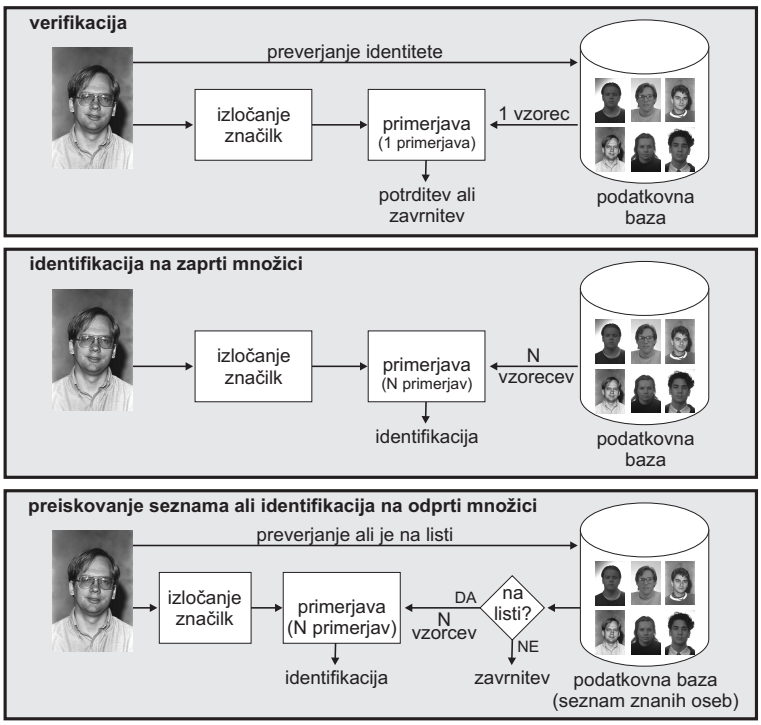
Slika 1.2: Primerjava različnih biometričnih podatkov: (a) na podlagi analize Zephyr, (b) za namene potovalnih dokumentov.

1.1 Sistem za prepoznavanje obrazov

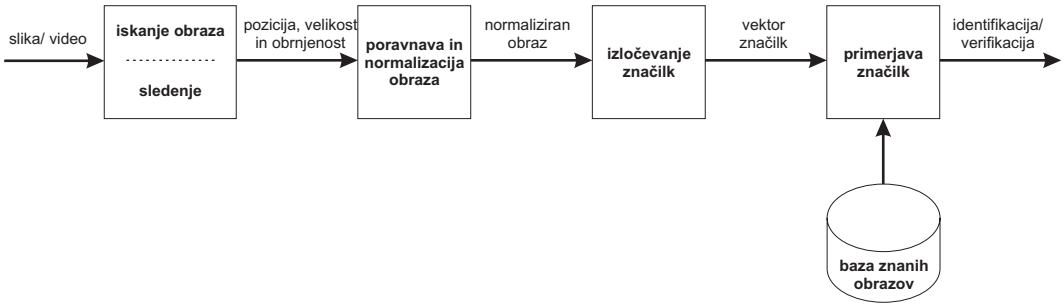
Od sistema za prepoznavanje obrazov pričakujemo, da identificira obraze, ki so prisotni na sliki ali video posnetku. Sistem lahko opravlja tri različne naloge: (1) verifikacijo, kjer sistem preveri istovetnost osebe, (2) identifikacijo, kjer mora sistem s pomočjo podatkovne baze znanih oseb ugotoviti komu pripada iskani obraz in (3) preiskovanje seznama (angl. *watch list*) ali tako imenovano identifikacijo na odprti množici, kjer mora sistem najprej preveriti, če je oseba v podatkovni bazi in če je, jo prepoznati. Slika 1.3 prikazuje vse tri naloge, kjer je naloga preiskovanja seznama najtežja naloga sistema. V disertaciji se bomo omejili na nalogo identifikacije na zaprti množici. V nadaljevanju bomo predstavili osnovni postopek avtomatske prepoznave obrazov. Na koncu pa bomo predstavili razširjen sistem za prepoznavo in analizo obrazov.

Sistem za avtomatsko prepoznavanje sestavljajo štirje moduli (slika 1.4): za iskanje, normalizacijo, izločanje značilk in njihovo primerjavo. Nalogi iskanja in normalizacije sta predkoraka obrazne prepoznave. V samo prepoznavo pa uvrščamo nalogi izločanja značilk in njihovo primerjavo.

Naloga **iskanja obraza** je segmentacija ozadja od regije, kjer se nahaja obraz. Če gre za video posnetek moramo običajno tako najdeni regiji še slediti s pomočjo metode za **sledenje**. Modul za poravnavo skrbi za natančnejšo pozicioniranje in s tem **normalizacijo** obraza, medtem ko modul iskanja najde samo grobo pozicijo in velikost obraza. Glede na pozicijo obraznih značilk, kot so oči, nos in usta, se obraz geometrijsko normalizira. Nato se običajno izvede še normalizacija osvetlitve. Po postopku normalizacije sledi **izločanje značilk**. Izločijo se tiste značilke, ki najbolj razlikujejo različne osebe med seboj in so hkrati tudi neodvisne od geometrijskih in svetlobnih sprememb. Za prepoznavo se izločene značilke obraza **primerjajo** z



Slika 1.3: Tri naloge sistema za prepoznavanje obrazov: verifikacija, identifikacija na zaprti množici in pregled seznama ali identifikacija na odprti množici.



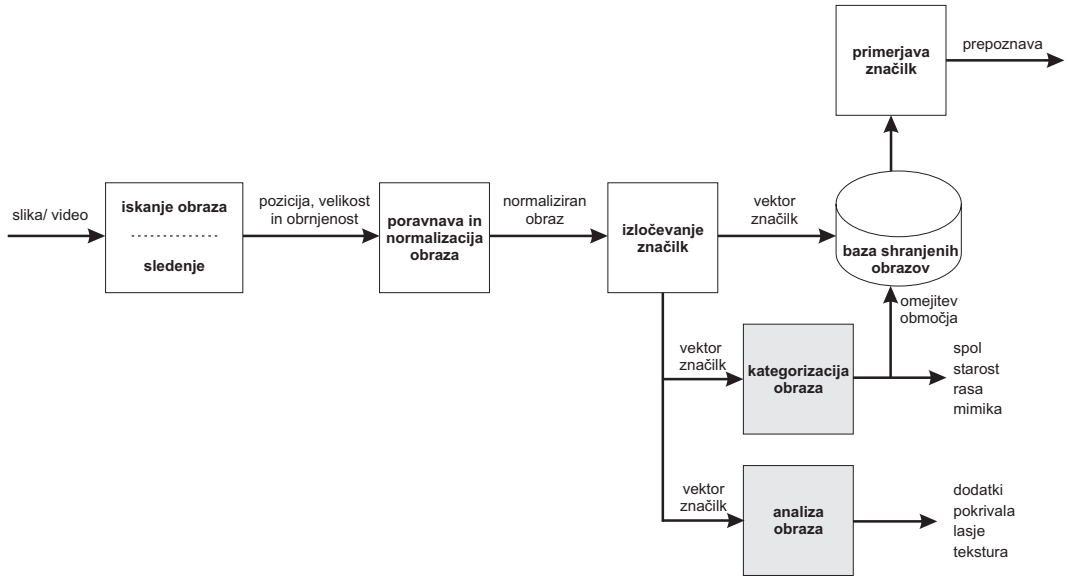
Slika 1.4: Sistem za avtomatsko prepoznavanje obrazov.

ostalimi značilkami znanih oseb iz podatkovne baze. Če sistem najde ujemanje, je rezultat identiteta osebe, drugače sistem javi, da je oseba neznana.

Rezultat uspešnega prepoznavanja je zelo odvisen od izločenih značilk, s katerimi obraz predstavimo ter od klasifikacijske metode, ki med seboj loči posamezne osebe. Seveda pa sta za uspešno določanje in izločanje značilk zelo pomembna tudi predkoraka normalizacije in iskanja obraza.

1.2 Razširjen sistem za prepoznavanje in analizo obrazov

Naloga identifikacije na odprti množici je med vsemi tremi nalogami najtežja. Sistem mora najprej preveriti ali je oseba, katere identiteto iščemo, sploh v podatkovni bazi. Če je, poišče njeno identiteto, v primeru če je ni, pa vrne odgovor da je oseba za sistem neznana. Takšen sistem za identifikacijo na odprti množici lahko dodatno nadgradimo z novimi moduli za analizo, kategorizacijo in prepoznavo obrazne dinamike. Slika 1.5 prikazuje tako razširjen sistem za prepoznavanje, kategorizacijo in analizo obrazov.



Slika 1.5: Razširjen sistem za prepoznavanje, kategorizacijo in analizo obraza.

Glavna prednost tako razširjenega sistema je v tem, da tudi v primeru neuspešne prepoznave vrne informacijo o osebi. Obraz osebe ustrezno kategorizira in analizira. S pomočjo modula **kategorizacije** lahko sistem določi spol osebe (moški, ženska), njeno starostno skupino (mladoletna, polnoletna), rasno skupino in mimiko obraza (vesel, žalosten, presenečen, jezen, prestrašen). S pomočjo **analize** pa lahko izvemo ali oseba nosi kakšne obrazne dodatke (uhani, očala, brki, brada, ličila), kakšno pričesko ima (barva, dolžinam, oblika), ali ima kakšna pokrivala (kapa, klobuk, šal). Natančneje lahko določimo tudi teksturo kože (nagubana, gladka, mozoljasta).

Na takšen način se uporabnost sistema močno poveča. Področje uporabe se tako razširi iz varnostnih sistemov in sistemov za kontrolo dostopa na področja, kot so inteligentni uporabniški vmesniki, oglaševanje, zabavništvo (računalniške igrice) in interaktivno umetnost. Posamezne tehnologije za kategorizacijo in analizo obrazov se že uporabljajo, ne obstaja pa še sistem, ki bi združeval vse te module v celovit sistem.

Naslednja velika prednost takšnega sistema je uporaba modula kategorizacije za povečanje točnosti samega modula prepoznavanja. S pomočjo kategorizacije lahko ugotovimo določene karakteristike osebe in na podlagi tega dodatno omejimo preiskovanje po podatkovni bazi znanih oseb. Če, na primer sistem neznano osebo kategorizira kot moškega bo nadalje preiskoval identiteto osebe samo po moškem delu podatkovne baze znanih oseb.

V poglavju 10, kjer bomo predstavili področja uporabe, bomo podrobneje predstavili sistem pametnega oglaševanja, ki uporablja omenjene prednosti tako razširjenega sistema.

1.3 Motivacija in cilji

Metode za prepoznavanje obrazov so postale v zadnjih letih ene najuspešnejših metod na področju analize in razumevanja slik. Temu je tako predvsem zaradi zelo velike uporabnosti na področju nadzora in varovanja, ki je v zadnjih letih še pridobilo na pomembnosti.

Da je vse več zanimanja za prepoznavanje obrazov pričajo tudi številne konference, ki so namenjene izključno temu področju: (1) International Conference on Audio- and Video-based Biometric Person Authentication (AVBPA) od leta 1997, (2) International Conference on Automatic Face and Gesture Recognition (AFGR) od leta 1995 in številni neodvisni preizkusi: (1) The Facial Recognition Technology (FERET), avgusta 1994, marca 1995 in septembra 1996, (2) Face Recognition Vendor Test (FRVT) leta 2000, 2002, 2006, (3) Face Recognition Grand Challenge (FRGC), leta 2005 in (4) preizkusni XM2VTS protokol.

Predlaganih je bilo že ogromno najrazličnejših metod za prepoznavanje obrazov. Glavna težava vseh metod pa je, da so večinoma preizkušene na različnih podatkovnih bazah. Zaradi tega lahko imamo zelo različne rezultate za isto metodo. Velikokrat so si rezultati preizkusov različnih metod celo nasprotni.

Težava je v tem, da avtorji, ki metodo predlagajo, le-to tudi preizkusijo v takšnem okolju, da se izkaže za boljšo nad drugimi. Tudi v primeru, če si avtorji izberejo isto podatkovno bazo, je lahko izbira učne in testnih množic različna. Na uspešnost prepoznave pa vpliva tudi normalizacijski postopek. V večini primerov normalizacijski postopek ni natančneje opisan, kar onemogoča, da bi lahko preizkus ponovili. Zelo pomembne so tudi različne tehnike izločanja značilk in njihovo število. V primeru metod osnovanih na videzu je to velikost podprostora oziroma število glavnih komponent, v primeru metode Gaborjevih valčkov pa število filtrov. Velik vpliv imajo tudi različne metode za primerjavo vektorjev značilk. Nekateri avtorji uporabljajo navadne razdalje, medtem ko nekateri uporabljajo utežene različice razdalj.

Glavni cilj doktorske disertacije je izvesti neodvisen preizkus najpopularnejših metod za prepoznavanje obrazov. Vse metode bomo preizkusili na skupni standardni podatkovni bazi FERET in podatkovni bazi AR. Za preizkus bomo uporabili skupen

postopek, ki bo vključeval enoten normalizacijski postopek, enako število značilk in s tem tudi enako časovno kompleksnost, enake učne in testne množice ter enake metode primerjanja. Za metodo primerjave bomo izbrali metodo najbližjih sosedov z različnimi merami razdalj.

Cilj preizkusa je najti najboljšo metodo za prepoznavanje obrazov, predvsem pa najboljšo kombinacijo metode in mere razdalje. Ker bomo metode preizkusili na različnih testnih množicah, bomo na podlagi tega poskušali ugotoviti katera metoda je nad katero množico najuspešnejša.

Vsaka metoda pa ima svoje slabe in dobre lastnosti. S pomočjo preizkusa bomo poskušali tudi ugotoviti takšne lastnosti posamezne metode. Vsako metodo lahko s pomočjo skrbno nastavljenih parametrov pripeljemo do njenih najboljših rezultatov. Zaradi tega ni rešitev za izboljšanje sistema za prepoznavanje v prilagajanju metod, ampak v združitvi posameznih metod v tako imenovani hibridni sistem. Na podlagi preizkusov bomo poizkušali ugotoviti, katera metoda nam daje najboljše rezultate tudi pri višjih rangih in na podlagi tega predlagali najboljše metode za posamezne komponente hibridnega sistema. Tako predlagani hibridni sistem bomo na koncu tudi preizkusili na obeh podatkovnih bazah obrazov.

1.4 Pričakovani prispevki k znanosti

1. **Predstavitev sistema za prepoznavanje, klasifikacijo in analizo obrazov.** Splošni sistem za prepoznavanje razširimo z metodami za klasifikacijo in analizo človeškega obraza. Predstavljen sistem je zasnovan modularno in omogoča iskanje, prepoznavanje, kategorizacijo in analizo človeških obrazov. Tako zasnovan sistem odpira nova področja pri interakciji sistemov s človekom.

2. **Implementacija in primerjava različnih metod za prepoznavanje.** Izbrali bomo osem najbolj znanih metod za prepoznavanje obrazov in jih preizkusili na enotni podatkovni bazi z enotnim postopkom normalizacije in primerjave. S pomočjo takšnega preizkusa bomo poskušali ugotovili, katere kombinacije tehnik izločanja značilk in tehnik primerjave so najuspešnejše, ter katera metoda za prepoznavanje obrazov deluje najbolje nad obrazi z določenimi lastnostmi.

3. **Preizkus različnih razdalj za merjenje oddaljenosti med vektorji značilk.** Za primerjavo vektorjev značilk bomo uporabili metodo najbližjega soseda pri kateri lahko uporabimo različne razdalje za merjenje podobnosti med vzorci. Vse predlagane metode bomo preizkusili v kombinaciji s 14 različnimi merami za razdalje. Preizkusili bomo uspešnost najboljšega ujemanja in uspešnost pri višjih rangih.

4. **Ugotavljanje vpliva postopka normalizacije na uspešnost prepoznave.** Na področju prepoznavanja obrazov ni enotnega postopka normalizacije, ki bi

najbolje deloval v različnih situacijah, niti ni znano kako normalizacijski postopek vpliva na uspešnost prepoznavanja. Preizkusili bomo različne normalizacijske postopke in predlagali najboljši postopek za različne metode in naloge. Pri preizkusih ne bomo upoštevali samo celotnih obrazov ampak tudi različne dele obraza. Na podlagi tega bomo poskušali ugotovili, kateri del obraza je za prepoznavanje najpomembnejši.

5. **Izdelava hibridnega sistema za prepoznavanje obrazov.** Sistemi, ki dandanes obetajo najboljšo uspešnost prepoznavanja obrazov so ravno hibridni sistemi, ki združujejo več različnih metod in s tem lahko odpravijo pomanjkljivosti posameznih metod. Na podlagi rezultatov uspešnosti za posamezne metode bomo predlagali najustreznejše metode za hibridni sistem. Tako izdelan hibridni sistem bomo tudi preizkusili na podatkovni bazi obrazov in ugotovili njegove značilnosti.

6. **Inovativna uporaba sistema za prepoznavanje, kategorizacijo in analizo na področju oglaševanja.** Predstavili bomo inovativno idejo uporabe sistemov za prepoznavanje obrazov na področju oglaševanja. Oglaševalski panoji opremljeni s to tehnologijo se bodo lahko prilagajali uporabniku in mu vedno poskušali ponuditi to, kar najbolj potrebuje.

1.5 Struktura disertacije

V uvodnem poglavju 1 predstavimo psihološki pogled na prepoznavanje obrazov. Opišemo tudi postopek standardnega sistema za prepoznavanje obrazov. Na osnovi obstoječega sistema predlagamo razširjen sistem za prepoznavanje, kategorizacijo in analizo obrazov.

Sledi pregled metod za iskanje in prepoznavo obrazov. V poglavju 2 naredimo celoten pregled najpomembnejših metod za iskanje obrazov na slikah. Podrobneje opišemo metodi na podlagi algoritma AdaBoost avtorjev Viole in Jonesa in Bayesovo metodo diskriminantnih značilk avtorja Liu-a. V sklopu pregleda metod za prepoznavanje obrazov predstavimo metode, ki delujejo na osnovi celotnega videza obraza in metode na osnovi značilk in geometrije. Predstavimo pa tudi obstoječe hibridne metode in sisteme.

V poglavju 3 predstavimo celotni seznam vseh najpomembnejših podatkovnih baz obrazov, ki se uporabljajo tako za metode iskanja, kot tudi za metode prepoznavanja obrazov. Podrobneje opišemo podatkovni bazi FERET in AR, ki smo ju uporabili v naših preizkusih.

Izvedenih je bilo že kar nekaj neodvisnih preizkusov različnih metod za prepoznavanje obrazov. Med pomembnejše preizkuse vsekakor spadajo preizkusi FERET in FRVT. Podrobnejše opise preizkusov in ugotovitve bomo navedli v poglavju 4. Na začetku poglavja bomo predstavili tudi standardne kazalce uspešnosti, ki se uporabljajo pri različnih nalogah prepoznavanja.

Ker je normalizacija zelo pomemben predkorak samega postopka prepoznave jo podrobneje predstavimo v poglavju 5. Opišemo postopek geometrijske in svetlobne normalizacije slik obrazov.

V poglavjih 6 in 7 predstavimo posamezne metode za prepoznavanje obrazov. Na začetku vsake metode predstavimo teoretično ozadje, nato predstavimo metodo na enostavnem primeru in na koncu predstavimo metodo na primeru prepoznavanja obrazov. V prvem delu predstavimo metode osnovane na videzu obraza: PCA, LDA in metodo ICA. V drugem delu pa metodo Gaborjevih valčkov, ki deluje na posameznih značilkah obraza.

V poglavju 8 predstavimo vse mere razdalj, ki jih lahko uporabljamo pri postopku primerjave dveh vektorjev. Na podlagi razdalje lahko izračunamo koliko so si vektorji značilk med seboj podobni, kar določa tudi podobnost slik obrazov.

V poglavju 9 opišemo preizkuse, ki smo jih izvedli s posameznimi metodami z različnimi merami razdalj. Preizkusili smo najboljše ujemanje pri rangu 1 in uspešnost metod pri višjih rangih. Poglavje zaključimo z ugotovitvami in predlaganim vzporednim in zaporednim hibridnim sistemom, ki smo ju tudi preizkusili na testnih množicah podatkovnih baz FERET in AR.

V zadnjem poglavju 10 predstavimo različna področja uporabe sistemov za prepoznavanje obrazov. Poseben poudarek namenimo inovativni ideji uporabe sistema za prepoznavanje, kategorizacijo in analizo na področju pametnega oglaševanja.

Disertacijo sklenemo z zaključkom in predlogi za nadaljnje delo.

Poglavje 2

Pregled metod za iskanje in prepoznavo obrazov

2.1 Uvod

Celoten sistem prepoznavanja obrazov je sestavljen iz iskanja obrazov, izločanja značilk, normalizacije in samega prepoznavanja. Glavna dela sistema sta vsekakor iskanje in prepoznavanje obrazov. Za iskanje obrazov na slikah je bilo predlaganih že veliko metod. V tem poglavju bomo predstavili trenutno najboljše metode za iskanje obrazov. Podrobneje si bomo ogledali dve zelo učinkoviti metodi, ki ju zaradi hitrosti lahko uporabljamo tudi v realnem času. Naredili bomo tudi pregled nad metodami za prepoznavanje obrazov, ki jih razdelimo v dve glavni skupini: metode na osnovi videza obraza in metode, ki temeljijo na posameznih značilkah in geometriji. Na koncu bomo pregledali nekaj obstoječih hibridnih metod in hibridnih sistemov za prepoznavanje obrazov.

2.2 Pregled metod za iskanje obrazov

Prvi korak pri sistemih za prepoznavanje obrazov je najti obraze na sliki. Točnost metod za iskanje obrazov ima zelo velik vpliv na normalizacijo in posredno na samo točnost prepoznave. Cilj metode za iskanje obrazov je na dani sliki ali video posnetku najti in locirati vse obraze na sliki, ne glede na njihov položaj, velikost, nagib, starost ali mimiko. Metode morajo biti tudi neobčutljive na različne osvetlitve in spremenljive okoliščine, ki se pojavljajo na sliki ali videu.

Prve metode za iskanje obrazov so se osredotočile na en ali nekaj obrazov na enakomernem ozadju, medtem ko omogočajo trenutno najboljše metode določanje točne pozicije in obrnjenosti vseh obrazov na neenakomernem ozadju [92, 82, 75, 46, 85, 58].

Za iskanje človeških obrazov na slikah je bilo predlaganih že veliko različnih tehnik. Čeprav uporabljajo nekatere metode različne pristope, jih lahko razdelimo v

štiri glavne skupine: (1) metode na osnovi znanja (angl. *knowledge-based methods*), (2) metode na osnovi nespremenljivih značilk (angl. *feature invariant approaches*), (3) metode ujemanja z vzorcem (angl. *template matching methods*) in (4) metode na osnovi videza (angl. *appearance-based methods*) [33, 110, 3].

Med vsemi pristopi za iskanje obrazov na slikah so se za najuspešneje izkazale metode na osnovi videza ali slike obraza (angl. *appearance- or image-based methods*). Za razliko od pristopa, ki temelji na prej definiranih vzorcih, pri tem pristopu določimo vzorec na podlagi učnih slik. S pomočjo strojnega učenjem in statistične analize določimo glavne karakteristike slik obrazov in neobrazov[1]. Tako naučene karakteristike lahko predstavimo s pomočjo porazdelitvenih modelov ali ločitvenih funkcij (angl. *discriminant functions*). Število karakteristik pogosto zmanjšamo z namenom večje učinkovitosti učenja in delovanja same metode.

Veliko metod, ki delujejo na osnovi videza, bi lahko uvrstili med verjetnostne metode. Na vektor značilk (angl. *feature vector*), ki ga dobimo iz slike, lahko namreč gledamo kot na naključno spremenljivko $\mathbf{x}$. To naključno spremenljivko $\mathbf{x}$ lahko označimo kot obraz ali kot neobraz na podlagi verjetnostne porazdelitvene funkcije $p(\mathbf{x}|obraz)$ in $p(\mathbf{x}|neobraz)$. Za klasifikacijo izbrane slike na podlagi izračunane verjetnosti lahko uporabimo Bayesov klasifikator ali metodo maksimalne verjetnosti (angl. *maximum likelihood*). Bayesovega klasifikatorja ne moremo neposredno uporabiti iz dveh razlogov: (1) dimenzija spremenljivke $\mathbf{x}$ je zelo velika in (2) verjetnostni funkciji $p(\mathbf{x}|obraz)$ in $p(\mathbf{x}|neobraz)$ predstavljata kompleksen model.

Ostale metode, ki se uporabljajo za iskanje obrazov na osnovi videza, pa poskušajo najti ustrezno ločitveno funkcijo (t.i. odločitveno ravnino, razdelitveno hiperravnino ali mejno funkcijo), ki bo razlikovala med obrazi in neobrazi. Običajno se ločitveno funkcijo poišče tako, da se vzorce slik preslika v prostor z manjšo dimenzijo, nato pa se na podlagi tako zgrajenega prostora izoblikuje mejno funkcijo. Drugi način je s pomočjo nevronskih mrež poiskati ustrezno nelinearno odločitveno ravnino. Pri iskanju ustrezne ločnice si lahko pomagamo tudi s pomočjo metode podpornih vektorjev (angl. *Support Vector Machines*, SVM). Ta metoda deluje tako, da vzorce slik predstavi v višje dimenzijskem prostoru in na podlagi tega določi ravnino, ki ločuje med obrazi in neobrazi.

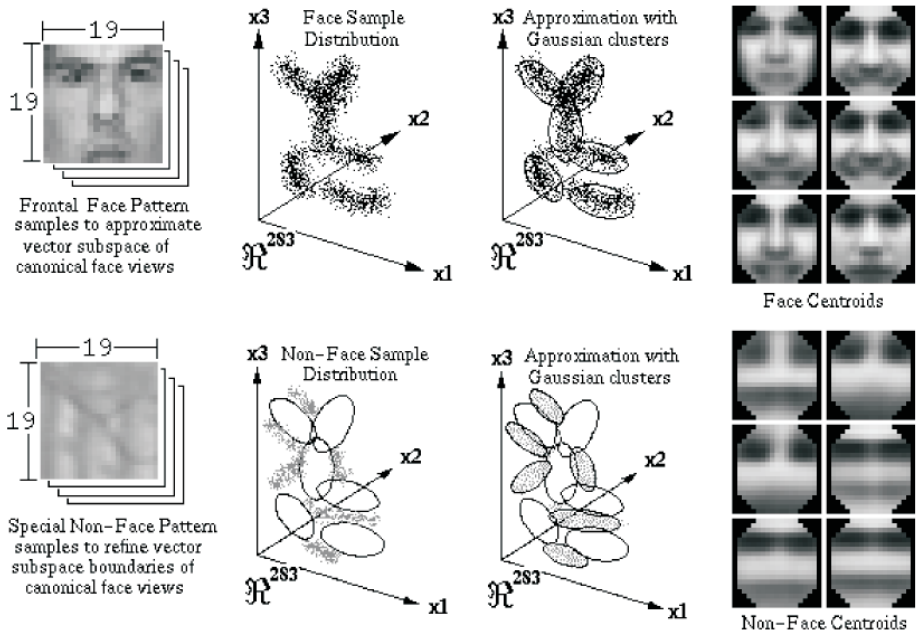
Pri iskanju obrazov sta pomembni dve statistiki: pravilno najdeni obrazi (angl. *True Positives*, TP), ki označujejo stopnjo detekcije (angl. *detection rate*), ter napačno najdeni obrazi (angl. *False Positivies*, FP), kjer metoda najde obraze tudi tam, kjer jih ni. Idealni sistem za detekcijo obrazov mora imeti čim večjo stopnjo detekcije in čim manjše število napačno najdenih obrazov.

V nadaljevanju bomo predstavili šest najuspešnejših metod za iskanje obrazov na slikah. Podrobneje bomo predstavili metodi na osnovi algoritma AdaBoost avtorjev Viole in Jones [100, 101, 102] in Bayesovo metodo diskriminantnih značilk (angl. *Bayesian Discriminating Features method*, BDF) avtorja Liu [58]. Na koncu bomo predstavili rezultate preizkusa posameznih metod na standardizirani bazi obrazov.

[1]Izraz "neobraz" označuje vse vzorce razen vzorca, ki predstavlja obraz.

2.2.1 Na porazdelitvi temelječa metoda – Sung in Poggio, 1997

Sung in Poggio [92, 91] sta leta 1997 zasnovala metodo, ki temelji na sistemu porazdelitvenih modelov. Pokazala sta, kako lahko določimo porazdelitev slikovnih vzorcev enega razreda s pomočjo negativnih in pozitivnih primerov tega razreda. Njun sistem sestavljajo porazdelitveni modeli za razred obrazov in razred neobrazov (slika 2.1) ter klasifikator na osnovi več-nivojskega perceptrona.



Slika 2.1: Porazdelitvena modela za razred obrazov in neobrazov, ki sta ju uporabila Sung in Poggio v svojem delu [92]. Posamezen model sestavlja mešanica Gaussovih funkcij. Na desni strani so prikazane slike, ki predstavljajo centre Gaussovih krivulj.

Avtorja sta posamezno učno sliko obraza in neobraza najprej normalizirala in zmanjšala na velikost 19×19 slikovnih elementov. Sliko sta nato obravnavala kot vektor dolžine 361 elementov. Tako zgrajene vektorje, oziroma vzorce obrazov in neobrazov, sta na podlagi *k-means* algoritma razdelila v šest skupin. S pomočjo več dimenzijske Gaussove funkcije sta vsako skupino predstavila s srednjo vrednostjo in kovariančno matriko. Med testnim vzorcem in tako določenimi modeli sta izračunala dva tipa razdalj. Prva razdalja je bila normalizirana Mahalanobis razdalja, merjena med testnim vzorcem in središči posamezne porazdelitvene skupine vzorcev. Razdalje so bile merjene v dimenzijsko zmanjšanem podprostoru, ki ga je določalo 75 največjih lastnih vektorjev modela. Druga razdalja je bila Evklidska razdalja med testnim vzorcem in njegovo projekcijo v omenjeni 75 dimenzijski podprostor. Ta razdalja je določila razliko med vzorci, ki jo prva razdalja ni merila. 12 tako določenih

parov razdalj sta nato pripeljala na vhod nevronske mreže, ki je ločevala obraze od neobrazov. Klasifikator sta učila s standardnim postopkom vzvratnega razširjanja (angl. *backpropagation*) iz 47.316 učnih primerov. Od tega je bilo pozitivnih primerov, ki so predstavljali obraz 4.150, ostali primeri pa niso vsebovali obraza. Poudariti je potrebno, da pozitivnih učnih primerov, ki predstavljajo obraze ni težko izbrati. Težje je izbrati reprezentativne primere, ki ne vsebujejo obrazov. Za rešitev tega problema je bila predlagana metoda razmnoževanja učnih primerov (angl. *bootstrap*), ki dopolnjuje učno množico neobrazov med samim učenjem. S pomočjo nevronske mreže, naučene na podlagi majhne učne množice neobrazov, izvedemo iskanje nad poljubnim zaporedjem slik in izberemo napačno klasificirane primere. Te primere nato dodamo v učno množico kot nove primere neobrazov. Takšna tehnika razmnoževanja učnih primerov rešuje problem slabo določenih učnih primerov neobrazov in je bila kasneje še mnogokrat uporabljena [75, 82, 100].

2.2.2 Na nevronskih mrežah temelječa metoda – Rowley et al., 1998

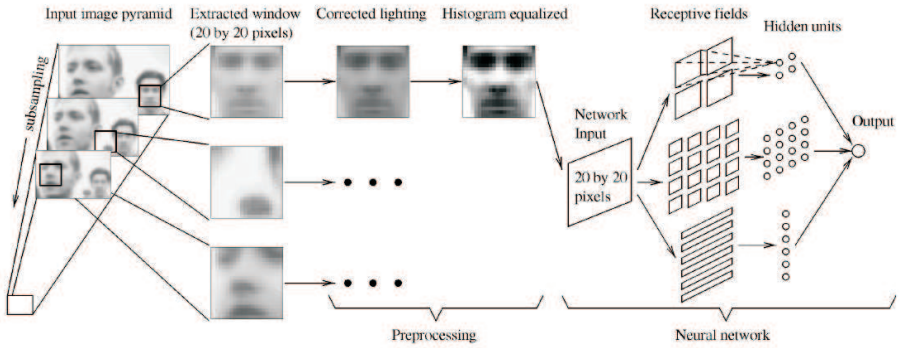
Nevronske mreže se pogosto uporabljajo pri problemih razpoznavanja vzorcev, kot so na primer optično prepoznavanje znakov (angl. *Optical Character Recognition*, OCR), prepoznavanje objektov in samodejna vožnja vozil. Na problem iskanja obrazov lahko gledamo kot na nalogo razpoznavanja vzorcev z dvema razredoma: razredom obrazov in razredom neobrazov (primeri slik na katerih ni obrazov). Za tako specializiran problem je bilo predlaganih že veliko različnih arhitektur nevronskih mrež. Prednost uporabe nevronskih mrež za iskanje obrazov je vsekakor njena velika sposobnost učenja kompleksnosti vzorca. Za doseganje sprejemljive stopnje učenja pa moramo znati nevronsko mrežo ustrezno naravnati: določiti moramo ustrezno število nivojev, število nevronov na vsakem nivoju in pravo stopnjo učenja.

Med vsemi metodami za iskanje obrazov, ki temeljijo na nevronskih mrežah velja izpostaviti delo avtorja Rowleya et al. [82, 83, 84]. Za učenje vzorca obrazov in vzorca neobrazov s pomočjo učnih primerov slik obrazov in neobrazov je uporabil večnivojsko nevronsko mrežo. Sung je na primer uporabil nevronsko mrežo samo za določitev ločitvene funkcije, ki je ločevala obraze od neobrazov [91].

Za razliko od drugih avtorjev, ki so za učenje uporabili eno ali dve nevronski mreži, je Rowley učinkovitost povečal z uporabo več nevronskih mrež in z različnimi metodami za izbiro med izhodi teh nevronskih mrež. Tako sestavljata njegovo metodo dve glavni komponenti: nevronske mreže za iskanje obrazov in odločitveni model, ki določa končno odločitev med vsemi izhodi nevronskih mrež.

Na sliki 2.2 lahko vidimo primer nevronske mreže, ki na vhod sprejme sliko velikosti 20×20, na izhodu pa vrne rezultat v intervalu od -1 do 1. Tako nam nevronska mreža vrne vrednost blizu -1, če na vhodni sliki ni obraza, ali pa vrednost blizu 1, če je na vhodu slika obraza.

S pomočjo tako definiranega klasifikatorja lahko poiščemo vse obraze na sliki tako, da preiščemo vse lokacije celotne slike. Če hočemo poiskati tudi obraze, ki so



Slika 2.2: Diagram nevronske mreže, ki jo je uporabil Rowley [82] v svoji metodi za iskanje obrazov na sliki. Sliko pred vstopom v nevronsko mrežo ustrezno normalizira. Za izbiro med različnimi izhodi nevronskih mrež pa uporabi različne metode odločanja.

večji od 20×20 slikovnih elementov, moramo vhodno sliko postopno zmanjševati in za vsako velikost iskanje ponoviti.

Za učenje nevronske mreže je Rowley uporabil okrog 1.050 učnih slik obrazov različnih velikosti, nagibov, položajev in osvetlitev. Na podlagi ročno označenih značilk obrazov (oči, vrh nosu, robovi in središče ustnic) je slike ustrezno normaliziral na enotno velikost, nagib[2] in obrnjenost[3]. V drugem delu svojega sistema je združil rezultate večkrat odkritih obrazov in se odločil med posameznimi izhodi nevronskih mrež. Za odločanje je uporabil preprosto odločitveno shemo, ki je temeljila na logičnih operacijah IN in ALI in tehniki glasovanja (angl. *voting*). Predlagal je veliko različic sistemov za iskanje obrazov, ki so vsebovali različne metode za odločanje med izhodi. Vsi sistemi so bili računsko manj zahtevni od sistema, ki sta ga razvila Sung in Poggio [92], in so dosegali boljše rezultate na testni množici 24 slik s 144 obrazi.

Oba predstavljena sistema pa imata to pomanjkljivost, da lahko najdeta samo obraze, ki so frontalni in pokončni. Rowley [82] je kasneje svojo metodo nadgradil tako, da poleg pokončnih obrazov najde tudi različno nagnjene obraze. To je dosegel s pomočjo nevronske mreže, ki ugotovi, za koliko je obraz na vhodni sliki nagnjen. Pravilno obrnjeno sliko nato pripelje na vhod prej predstavljene nevronske mreže, ki razlikuje med obrazi in neobrazi. Takšen sistem se je pri pokončnih obrazih izkazal za nekoliko slabšega od prejšnjega, vendar je še vedno na dveh testnih bazah obrazov našel kar 76.9% vseh različno nagnjenih obrazov z zelo majhnim številom napačno najdenih obrazov.

[2]Nagib označuje odklon od navpičnega položaja glave.
[3]Obrnjenost označuje zasuk od frontalnega položaja glave.

2.2.3 Metoda podpornih vektorjev – Osuna 1997

Metodo podpornih vektorjev (angl. *Suport Vector Machines*, SVM) je za iskanje obrazov prvič uporabil Osuna et al. [75]. Metoda podpornih vektorjev omogoča nov način učenja drugačen, kot smo ga bili do sedaj vajeni pri klasifikatorjih, kot so Bayesov ali nevronske mreže. Večina postopkov učenja temelji namreč na minimizaciji klasifikacijske napake (t.i. empiričnega tveganja), metoda podpornih vektorjev pa deluje na drugačen način. Njen cilj je minimizirati zgornjo mejo pričakovane napake posplošitve, kar imenujemo minimizacija strukturnega tveganja. Klasifikator na podlagi SVM-ja je linearni klasifikator, ki poskuša najti takšno hiperravnino, da je pričakovana klasifikacijska napaka, za nek neznan primer, najmanjša. Takšno optimalno hiperravnino določimo s pomočjo utežne kombinacije majhne domene učnih vektorjev, ki jih imenujemo tudi podporni vektorji. Na iskanje optimalne hiperravnine lahko gledamo tudi kot na kvadratni optimizacijski problem z omejitvami.

Metoda podpornih vektorjev je tako časovno kot spominsko zelo zahtevna, kljub temu pa je Osuna [75] na njeni osnovi izdelal učinkovito metodo za iskanje obrazov. S pomočjo metode, naučene s 10 milijoni učnih primerov velikosti 19×19 slikovnih elementov, je pri iskanju dobil zelo majhno število napak. Njegova metoda je 30 krat hitrejša od metode avtorjev Sung in Poggio [92]. Kasneje je bila metoda podpornih vektorjev v kombinaciji z metodo valčkov (angl. *wavelets*) še večkrat uporabljena za iskanje obrazov in pešcev na slikah.

2.2.4 Metoda naivnega Bayesovega klasifikatorja – Schneiderman in Kanade, 1998

V nasprotju s prej opisanimi metodami, ki modelirajo globalno predstavitev celotnega obraza, sta Schneiderman in Kanade [86] s pomočjo naivnega Bayesovega klasifikatorja ocenila skupno verjetnost lokalnega videza in položaja obraznih vzorcev (t.i. podregij obraza) pri različnih velikostih.

Upoštevala sta lokalni videz objekta, saj so nekateri lokalni vzorci bolj pomembni, kot ostali. Pri obrazu je na primer vzorec okrog oči veliko bolj značilen od vzorca na licih.

Naivni Bayesov klasifikator, ki predpostavlja statistično neodvisnost vhodnih vzorcev, sta uporabila iz dveh razlogov. Prvič, ker dobro določi samo funkcijo porazdelitve za posamezne podregije. Drugič pa zato, ker lahko s pomočjo naivnega Bayesovega klasifikatorja določimo posteriorno verjetnost (angl. *posterior probability*). Pri različnih velikostih sliko najprej razdelita na štiri podregije. Te podregije nato s pomočjo metode PCA preslikata v nižje-dimenzijski prostor in jih združita v končno množico vzorcev. Vsaki tako preslikani podregiji določita verjetnost na podlagi preslikanih vzorcev. Metoda določi, da je na sliki obraz, če je razmerje verjetnosti večje od razmerja apriornih verjetnosti. S 93% točnostjo se metoda lahko primerja z metodo na osnovi nevronskih mrež, s tem, da ta metoda najde tudi delno nagnjene obraze in obraze s profila.

2.2.5 Na algoritmu AdaBoost temelječa metoda – Viola in Jones, 2001

S pomočjo metode učenja AdaBoost lahko zgradimo kompleksen nelinearen klasifikator $H_M(\mathbf{x})$, imenovan tudi močan klasifikator (angl. *strong classifier*). Takšen klasifikator sestavlja linearna kombinacija M preprostih, lažje zgrajenih klasifikatorjev, imenovanih tudi šibki klasifikatorji (angl. *weak classifier*). Močan klasifikator $H_M(\mathbf{x})$ lahko s pomočjo M šibkih klasifikatorjev $h_m(\mathbf{x}) \in \{0, 1\}$ izračunamo s pomočjo enačbe:

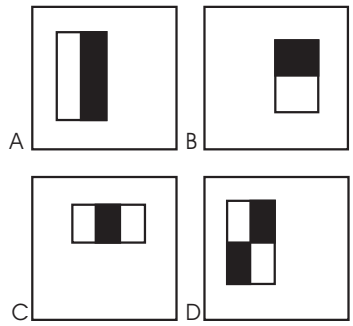
$$H_M = \frac{\sum_{m=1}^{M} \alpha_m h_m(\mathbf{x})}{\sum_{m=1}^{M} \alpha_m} \quad , \tag{2.1}$$

kjer $\mathbf{x}$ določa vzorec, ki ga želimo klasificirati, $\alpha_m \geq 0$ so pripadajoči koeficienti v $\mathbb{R}$ in vsota $\sum_{m=1}^{M} \alpha_m$ določa normalizacijski faktor. V diskretni različici lahko šibki klasifikator $h(\mathbf{x})$ zavzame diskretno vrednost 0 ali 1, medtem ko je pri realni različici izhod iz klasifikatorja $h_m(\mathbf{x})$ realna vrednost v $\mathbb{R}$. Vrednost klasifikatorja $H_M(\mathbf{x})$ je realna vrednost, določitev razreda pa se izračuna s pomočjo $\hat{y}(x) = \text{sign}[H_M(\mathbf{x})]$, normalizirano stopnjo zaupanja pa kot $|H_M(\mathbf{x})|$.

Namen učnega postopka AdaBoost je sistem naučiti zaporedje najboljših šibkih klasifikatorjev $h_m(\mathbf{x})$ in pripadajočih uteži α_m. Podano imamo množico N označenih učnih primerov $\{(x_1, y_1), \ldots, (x_N, y_N)\}$, kjer je $y_i \in \{0, 1\}$ oznaka razreda kateremu pripada primer $x_i \in \mathbb{R}^n$. Porazdeljenost učnih primerov določajo uteži $[w_1, \ldots, w_N]$, kjer posamezna utež w_i pripada učnemu primeru (x_i, y_i). Uteži se izračunajo med postopkom učenja in se pri vsaki iteraciji popravijo tako, da vedno določajo porazdelitev učnih primerov. Po m-ti iteraciji pridobijo tisti učni primeri (x_i, y_i), ki jih je težje klasificirati večjo utež $w_i^{(m)}$, tako da se jim pri naslednji iteraciji $m + 1$ bolj posvetimo. Algoritem AdaBoost mora vsebovati tudi postopek učenja šibkega klasifikatorja $h_m(\mathbf{x})$ iz učne množice z dano porazdelitvijo $[w_i^{(m)}]$. V delu avtorjev Viole in Jonesa se šibki klasifikator določi na podlagi praga izbranih značilk iz ogromne množice značilk, ki so podobne Haarovim valčkom.

Avtorja sta predlagala štiri vrste značilk, ki jih prikazuje slika 2.3. Takšna značilka se nahaja kjerkoli v podoknu slike in se lahko razlikuje po obliki in velikosti. Za podokno velikosti 20×20 slikovnih elementov imamo tako na deset tisoče značilk različnih oblik, velikosti in na različnih lokacijah. Posamezni značilki k lahko na vzorcu $\mathbf{x}$ priredimo vrednost skalarja $z_k(\mathbf{x}) \in \mathbb{R}$ in tako dobimo ogromno množico značilk za posamezen nižje dimenzijski vzorec $\mathbf{x}$. Tako definirane značilke so zanimive iz dveh razlogov: (1) iz tako določenih značilk lahko zgradimo učinkovit klasifikator, ki ločuje med obrazi in neobrazi in (2) izračun posamezne značilke lahko poenostavimo s pomočjo integralne slike (angl. *integral image*).

Integralno sliko $II(x, y)$ v točki (x, y) izračunamo kot vsoto vrednosti slikovnih elementov pod in levo od točke (x, y):



Slika 2.3: Štirje tipi pravokotnih značilk, ki sta jih uporabila avtorja Viola in Jones [100], za iskanje obrazov na slikah. Vsaka značilka določa skalarno vrednost, izračunano kot vsoto vrednosti slikovnih elementov pod belo regijo, zmanjšano za vsoto vrednosti elementov pod črno regijo.

$$II(x,y) = \sum_{x' \leq x,\ y' \leq y} I(x,y) \tag{2.2}$$

Integralno slika lahko učinkovito izračunamo z enim samim prehodom čez originalno sliko z uporabo sledečih enačb:

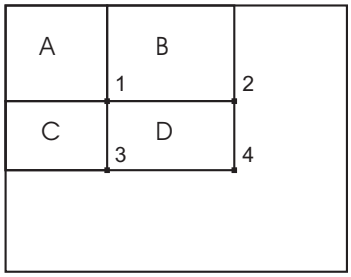
$$S(x,y) = S(x,y-1) + I(x,y) \tag{2.3}$$

$$II(x,y) = II(x-1,y) + S(x,y) \ , \tag{2.4}$$

kjer $S(x,y)$ določa kumulativno vsoto vrstice. Velja tudi $S(x,-1) = 0$ in $II(-1,y) = 0$. Z uporabo integralne slike lahko izračunamo vsoto kateregakoli območja samo iz štirih referenčnih točk, kot to prikazuje slika 2.4. Tako predstavljene slike pripomorejo k ogromnemu prihranku časa za izračun značilk na različnih lokacijah in različnih oblik.

Prednost integralne slike je tudi v tem, da omogoča zelo učinkovito računanje variance znotraj posameznega območja D. Tako lahko vsako podsliko svetlobno normaliziramo tako, da vse slikovne elemente območja D delimo z variacijo.

Pri učenju šibkega klasifikatorja moramo, glede na trenutno določene uteži, iz velike množice značilk izbrati najuspešnejšo značilko, ki najbolje ločuje negativne primere (slike neobrazov) od pozitivnih primerov (slike obrazov). Tako izbran najmočnejši šibki klasifikator se nato kombinira z obstoječim močnim klasifikatorjem. Pri diskretni različici AdaBoost se za določitev najprimernejšega šibkega klasifikatorja uporablja odločitveno drevo z enim vozliščem. Za vsako značilko moramo med učenjem šibkega klasifikatorja določiti optimalno pragovno funkcijo, tako da je število napačno klasificiranih primerov najmanjše. Šibki klasifikator $h(\mathbf{x}, f, p, \theta)$, ki vsebuje značilko (f), prag (θ) in predznak (p), je določen kot:



Slika 2.4: Vsoto slikovnih elementov znotraj območja D lahko izračunamo s pomočjo štirih referenčnih točk. Vrednost v točki 1 je vsota slikovnih elementov območja A. Vrednost v točki 2 je $A + B$, v točki 3 $A + C$ in v točki 4 $A + B + C + D$. Vsoto znotraj območja D lahko tako izračunamo kot (4+1)-(2+3).

$$h(x, f, p, \theta) = \begin{cases} 1 & \text{če je } pf(x) < p\theta \\ 0 & \text{sicer} \end{cases} , \tag{2.5}$$

kjer $\mathbf{x}$ določa podokno velikosti 24×24 slikovnih elementov znotraj slike. V praksi posamezna značilka ne more klasificirati učnih primerov brez napake. Značilke izbrane na začetku učenja napačno klasificirajo od 10% do 30% učnih primerov, medtem ko imajo kasnejše značilke, ki imajo opraviti s težjo nalogo, vrednost klasifikacijske napake od 40% do 50%. Celoten postopek učenja močnega klasifikatorja H_M iz šibkih h_m s pomočjo algoritma AdaBoost prikazuje Algoritem 1.

Tako določen močan klasifikator izloči veliko število podslik, ki ne vsebujejo obraza, medtem ko ohrani visoko stopnjo detekcije. Kljub temu posamezen močan klasifikator ne more zagotoviti dovolj nizke stopnje napačne detekcije (reda 10^{-6} ali manj). Rešitev je v združitvi več močnih klasifikatorjev v tako imenovano kaskado klasifikatorjev (slika 2.5).

Avtorja Viola in Jones sta idejo o združitvi močnih klasifikatorjev v kaskado nadgradila z učnim postopkom. Učenje prvega klasifikatorja poteka po opisanem postopku (algoritem 1), vsak nadaljnji klasifikator pa učita s pomočjo negativnih primerov prejšnjega klasifikatorja. Običajno je v kaskado združenih od 10 do 20 močnih klasifikatorjev. Pri iskanju obrazov na sliki se podslike, ki so bile izločene s pomočjo enega klasifikatorja, v naslednjih stopnjah ne obravnavajo več. S takim postopkom hitrost metode močno povečamo in hkrati zmanjšamo število napačnih detekcij pri malenkostnem poslabšanju uspešnosti.

2.2.6 Bayesova metoda diskriminantnih značilk – Liu 2003

Bayesova metoda diskriminantnih značilk (angl. *Bayesian Discriminating Features method*, BDF) [58] s pomočjo dveh Gaussovih modelov obrazov in neobrazov

Algoritem 1 : postopek učenja z AdaBoost

Vhod: Učne slike $(x_1, y_1), \ldots, (x_n, y_n)$ kjer $y_i = 0, 1$ označuje negativne in pozitivne primere.

Izhod: Končni močan klasifikator H_M in oznaka razreda $\hat{y}(x)$.

1: Določi začetne uteži $w_1, i = \frac{1}{2r}$ za $y_i = 0$ in $w_1, i = \frac{1}{2l}$ za $y_i = 1$, kjer r in l določata število negativnih in pozitivnih primerov.

2: **za vse** $m \in \{1 \ldots M\}$ **delaj**

3: Normaliziraj uteži: $w_{m,i} \leftarrow \frac{w_{m,i}}{\sum_{j=1}^{n} w_{m,j}}$.

4: Izberi najboljšega šibkega klasifikatorja glede na uteženo napako:

$$\epsilon_m = \min_{f,p,\theta} \sum_i w_i |h(x_i, f, p, \theta) - y_i|.$$

5: Definiraj $h_m(x) = h(x, f_m, p_m, \theta_m)$, kjer f_m, p_m in θ_m pripadajo minimalni vrednosti ϵ_m.

6: Spremeni uteži:

$$w_{m+1,i} = w_{m,i} \beta_t^{1-e_i} \quad ,$$

kjer je $e_i = 0$, če je primer pravilno klasificiran, drugače pa je $e_i = 1$ in $\beta_m = \frac{\epsilon_m}{1-\epsilon_m}$.

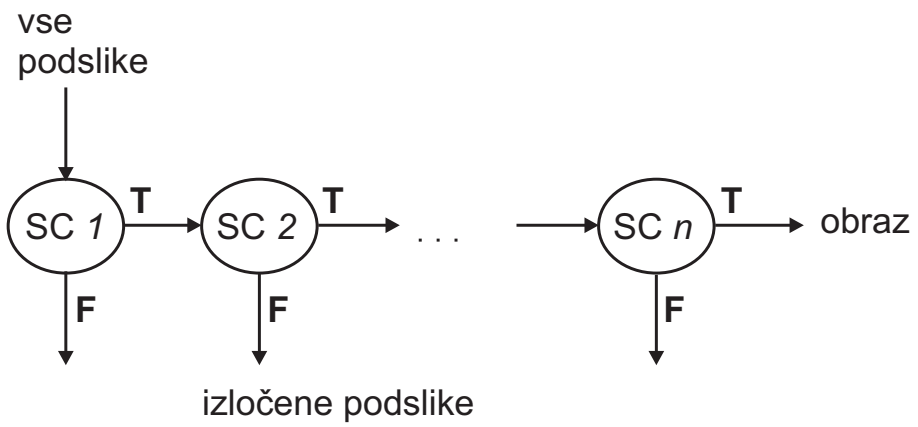
7: **konec za**

8: Končni močan klasifikator:

$$H_M = \frac{\sum_{m=1}^{M} \alpha_m h_m(x)}{\sum_{m=1}^{M} \alpha_m} \quad ,$$

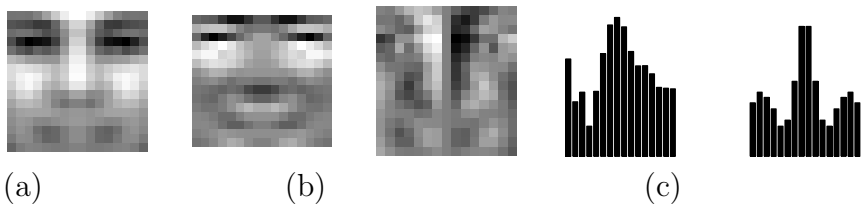
kjer $\alpha_m = \log \frac{1}{\beta_m}$

9: Oznaka razreda: $\hat{y}(x) = \text{sign}[H_M(x) - T_M]$,
kjer T_M določa dodaten prag za razmerje med stopnjo pravilne in napačne detekcije.



Slika 2.5: Kaskada n-tih močnih klasifikatorjev (SC). Na vhod prvega klasifikatorja pripeljemo vse podslike. Vsak nadaljnji klasifikator pregleda samo tiste podslike, ki jih prejšnji klasifikatorji niso izločili (F). Podslika, ki vsebuje obraz (T), mora tako ustrezati vsem klasifikatorjem.

uspešno najde frontalne obraze na sivinskih slikah. Diskriminatno značilko sestavljajo slikovni elementi vhodne slike, njena predstavitev s Haarovimi valčki in njena amplitudna preslikava. Slika 2.6 prikazuje posamezne komponente vektorja značilk: povprečno sliko obraza, njeno vertikalno in horizontalno sliko razlik (Haarova predstavitev) in horizontalno in vertikalno amplitudno predstavitev.



Slika 2.6: Posamezne komponente vektorja značilk za sliko povprečnega obraza: (a) vhodna slika, (b) njena Haarova predstavitev in (c) amplitudna predstavitev.

Na takšen način vsako učno sliko predstavimo s 768 dimenzijskim vektorjem značilk, ki ga ustrezno normaliziramo tako, da mu odštejemo njegovo povprečno vrednost in ga delimo s standardno deviacijo.

S predpostavko, da tako dobljeni vektorji značilk ustrezajo večrazsežni normalni porazdelitvi, jim lahko s pomočjo analize glavnih komponent (angl. *Principal Component Analysis*, PCA) zmanjšamo dimenzijo. Iz glavnih komponent metode PCA in njenega ostanka zgradimo funkcijo pogojne verjetnosti za razred obrazov. Na enak način zgradimo tudi model za razred neobrazov, pri čemer izberemo slike neobrazov s pomočjo modela obrazov, tako da ležijo neobrazi zelo blizu samega modela obrazov. S pomočjo tako zgrajenega modela obrazov (ω_f) in neobrazov (ω_n) ter

Bayesovega odločitvenega pravila določimo ali vhodni vzorec ($\mathbf{Y}$) vsebuje obraz ali ne (enačba 2.6).

$$\mathbf{Y} \in \begin{cases} \omega_f & \text{če je } P(\omega_f|\mathbf{Y}) > P(\omega_n|\mathbf{Y}) \\ \omega_n & \text{sicer} \end{cases}, \tag{2.6}$$

kjer $P(\omega_f|Y)$ določa aposteriorno verjetnost za razred obrazov in $P(\omega_n|Y)$ za razred neobrazov in ju lahko določimo s pomočjo Bayesovega teorema.

Modela obrazov in neobrazov sta bila zgrajena na podlagi učne množice 600 frontalnih slik obrazov iz podatkovne zbirke FERET [77] in 9 slik narave, ki niso vsebovale obrazov.

Metoda se je izkazala za zelo uspešno na testni množici, ki je vsebovale različne frontalne obraze baze FERET in slike iz standardne testne baze MIT/CMU. Na 807 testnih slikah iz baze FERET je metoda delovala z uspešnostjo 98.8%, brez napačno najdenih obrazov. Na množici 227 slik iz baze MIT/CMU, ki je vsebovala različno velike in nagnjene obraze ter obraze različne kvalitete, pa z uspešnostjo 97.4%, z enim napačno najdenim obrazom. Skupna uspešnost na vseh slikah je bila tako 98.5%, z enim napačno najdenim obrazom.

2.2.7 Rezultati preizkusa metod za iskanje obrazov

Primerjava učinkovitosti različnih metod ni enostavna, saj veliko avtorjev svoje metode preizkuša na svojih nestandardnih testnih množicah. Pri tistih, ki uporabljajo sicer iste testne množice slik, pa se pojavita druga dva problema: (1) avtorji različno interpretirajo, kaj je uspešno najden obraz in (2) za iste metode uporabljajo različne učne množice in tehnike učenja.

Kljub temu pa nekateri avtorji preizkusijo metode tudi na standardnih bazah kot sta to bazi MIT in CMU. Podatkovna baza MIT avtorjev Sung in Poggio [92] vsebuje na 23 slikah 155 frontalnih obrazov. Slike iz te baze so slabše kvalitete, poleg slik človeških obrazov vsebuje baza tudi narisane obraze ter en delno zakrit obraz. Podatkovna baza CMU avtorja Rowley et al. [84] pa vsebuje na 107 slikah 352 frontalnih obrazov. Poleg človeških obrazov vsebuje tudi 24 narisanih obrazov na 5 slikah. Podatkovna baza MIT/CMU vsebuje 130 omenjenih slik iz baze MIT in CMU, ki vsebujejo 507 frontalnih obrazov ter 50 dodatnih slik na katerih je 223 nagnjenih obrazov od katerih je 210 nagnjenih za več kot 10° (tabela 3.3).

Za preizkus metod za iskanje frontalnih obrazov se največkrat uporablja podatkovna baza MIT in podmnožica frontalnih obrazov podatkovne baze MIT/CMU. Tabela 2.1 prikazuje rezultate najuspešnejših metod za iskanje obrazov na slikah, ki smo jih predstavili. Nekatere metode so preizkušene na obeh omenjenih testnih množicah, nekatere pa samo na eni izmed njih. Pri nekaterih metodah pa so uporabljane samo podmnožice slik iz posameznih testnih baz. Tako so bile iz preizkusa metode naivnega Bayesovega klasifikatorja izvzete narisane slike obrazov (19 obrazov iz tesne množice MIT in 24 obrazov iz testne množice MIT/CMU). Metoda BDF

pa je bila preizkušena samo na 80 slikah testne množici MIT/CMU, ki so vsebovale 227 človeških frontalnih obrazov. Izvzeti so bili živalski obrazi, narisani obrazi, delno zakriti ali zamaskirani obrazi, obrazi risanih junakov in obrazi, ki niso bili frontalni.

Metoda	MIT		MIT/CMU	
	TP	FP	TP	FP
na osnovi porazdelitev	81.9%	13	-	-
nevronske mreže	90.3%	42	92.5%	862
podporni vektorji	74.2%	20	-	-
naivni Bayes	91.2%	12	93.0%	88
AdaBoost metoda	-	-	92.0%	65
metoda BDF	-	-	97.4%	1

Tabela 2.1: Rezultati preizkusa različnih metod za iskanje obrazov na testni bazi MIT in podmnožici testne baze MIT/CMU. Oznaka TP označuje delež pravilno najdenih obrazov, FP pa število napačno najdenih obrazov.

Pri pregledu metod za iskanje obrazov smo se omejili zgolj na frontalne obraze, v nekaterih domenah uporabe pa je pomembno poiskati tudi obraze zajete iz različnih zornih kotov. Pri preiskovanju digitalne zbirke fotografij je približno 75% vseh obrazov na slikah takšnih, ki niso frontalni. Pristopov za iskanje različno obrnjenih obrazov je veliko. Najpogosteje se uporabljata dva pristopa: (1) pristop na osnovi različnih pogledov (angl. *mult-view detection*) [29, 85] in (2) pristop na osnovi nespremenljivih značilk [107]. Pri obrazih, ki so obrnjeni za več kot 35° se je prvi pristop izkazal za boljšega.

2.3 Pregled metod za prepoznavanje obrazov

Področje prepoznavanja obrazov postaja vedno bolj pomembno in zanimivo. Področje je zanimivo za širok krog znanstvenikov in je pritegnilo veliko raziskovalcev z najrazličnejših področji, kot so psihologija, razpoznavanje vzorcev, računalniški vid, strojno učenje in računalniška grafika. Zaradi tega je literatura, ki se ukvarja s tem področjem, zelo obsežna in raznolika.

Za prepoznavanje obrazov je bilo predlaganih že veliko različnih metod. Pogosto pa posamezna metoda vsebuje različne pristope za izločanje značilk in za klasifikacijo obrazov. Zaradi različnih pristopov, ki jih metode uporabljajo, jih je zelo težko razdeliti glede na to, kakšne tehnike uporabljajo. Za razdelitev zato raje uporabimo predlog psihologov, ki razlikujejo kako si človek zapomni obraz. V prvo skupino uvrščamo tiste metode, ki za prepoznavanje uporabljajo celotno sliko obraza, v drugo pa tiste, ki uporabljajo samo značilke obraza. V zadnjem času pa so se kot zelo uspešne metode izkazale hibridne metode oziroma hibridni sistemi, ki združujejo tako posamezne značilke obraza kot tudi celoten videz. V nadaljevanju bomo predstavili metode iz posamezne skupine.

2.3.1 Metode na osnovi celotnega videza obraza

Pri metodah na osnovi celotnega videza (angl. *holistic matching methods*) se kot vhod v sistem za prepoznavanje uporablja slika celega obraza. Najbolj razširjena predstavitev slik obrazov je s pomočjo lastnih vektorjev [50]. Lastne vektorje slike lahko dobimo s pomočjo metod, ki temeljijo na eni izmed analiz, kot so analiza glavnih komponent (angl. *Principal Component Analysis*, PCA), linearna diskriminantna analiza (angl. *Linear Discriminant Analysis*, LDA) ali analiza neodvisnih komponent (angl. *Independent Component Analysis*, ICA). Najbolj razširjena in osnovna je metoda PCA.

Za predstavitev obrazov ter njihovo iskanje in prepoznavo sta lastne vektorje kot prva uporabila avtorja Turk in Pentland leta 1991 [98]. Lastne vektorje obrazov, ki sta jih dobila s pomočjo metode PCA, sta poimenovala lastni obrazi. S pomočjo lastnih obrazov sta vse slike obrazov predstavila s pomočjo veliko manjših vektorjev uteži. Testno sliko na vhodu sta najprej predstavila s pomočjo vektorja uteži in nato njeno identiteto pripisala tistemu vektorju, katerega uteži so bile najbližje testnemu vektorju (Evklidska razdalja). Opisala sta tudi postopek iskanja obrazov na slikah z upoštevanjem dejstva, da se slike obrazov in neobrazov med seboj razlikujejo, če jih predstavimo z lastnimi vektorji.

Metodo sta Moghaddam in Pentland [69] kasneje razširila tako, da sta namesto Evklidske razdalje za merjenje podobnosti uporabila verjetnost, ki sta jo izračunala s pomočjo Bayesove metode. Da določimo porazdelitev po tej metodi potrebujemo veliko število učnih primerov znotraj posameznega razreda. To težavo sta rešila tako, da sta problem več razredov (vsak razred določa posamezna oseba) predstavila kot problem dveh razredov. Razreda sta določila z merjenjem podobnosti med razredi s pomočjo Bayesove analize razlik med slikami. Definirala sta razred Ω_I, ki predstavlja spremembe med slikami istih oseb in razred Ω_E, ki predstavlja razlike med slikami različnih oseb. S predpostavko, da imata oba razreda Gaussovo porazdelitev, lahko za dano razliko slik $\Delta = I_1 - I_2$ določimo verjetnostni funkciji $P(\Delta|\Omega_I)$ in $P(\Delta|\Omega_E)$. S pomočjo danih verjetnostnih funkcij in pravila največje aposteriorne verjetnosti (angl. *Maximum A Posteriori Probability*, MAP) lahko tako določimo ali sliki pripadata isti osebi - če izpolnjujeta pogoj $P(\Delta|\Omega_I) > P(\Delta|\Omega_E)$. Avtorja poročata o precejšnjem izboljšanju uspešnosti glede na prejšnjo metodo, ki je uporabljala Evklidsko merjenje razdalje.

Za zelo uspešne sisteme za prepoznavanje so se izkazali tudi sistemi, ki temeljijo na linearni diskriminantni analizi – LDA. Za razliko od metode PCA, ki poskuša najti takšne vektorje, ki posamezne slike med seboj kar najbolje ločujejo, poišče metoda LDA takšne vektorje, ki poskušajo med seboj najbolje ločevati posamezne razrede. Tako moramo poiskati matriko $\mathbf{S}_B$, ki določa porazdelitev podatkov med razredi in matriko $\mathbf{S}_W$, ki določa porazdelitev podatkov znotraj posameznega razreda.

Avtorja Swets in Weng [93] sta metodo LDA uporabila za kategorizacijo posameznih razredov (slike obrazov in slike, kjer ni obrazov) ter kot prva tudi za kategoriza-

cijo posameznih oseb znotraj razreda obrazov. S pomočjo drevesne strukture sta na osnovi metod PCA in LDA sistem rekurzivno učila na vedno manjši učni množici. S takšnim rekurzivnim deljenjem učne množice v vsakem vozlišču sta nadaljevala, dokler niso učni primeri pripadali samo enemu razredu.

Primerjavo različnih metod je naredil Belhumeur [4]. Med seboj je primerjal štiri različne metode: korelacijsko metodo, različico linearne metode, metodo PCA in metodo LDA. Preden je uporabil metodo LDA, je podobno kot avtorja Swets in Weng, uporabil metodo PCA, s čemer je odpravil singularnost matrike $\mathbf{S}_W$. Rezultati preizkusa nad manjšo bazo so pokazali, da je od uporabljenih najboljša metoda LDA.

Zhao [112, 113] je v svojih prispevkih preizkusil metodo LDA ter metodo LDA v kombinaciji z metodo PCA (LDA+PCA). V svojih preizkusih je v primeru metode LDA+PCA pokazal na veliko zmožnost generalizacije brez ponovnega učenja. Dimenzijo podprostora je avtor izbral na podlagi podobnosti lastnih vektorjev z obrazi namesto na podlagi lastnih vrednosti, kot je to običajno. Namesto navadne Evklidske razdalje je za merjenje podobnosti uporabil uteženo Evklidsko razdaljo. Da je odpravil singularnost matrike $\mathbf{S}_W$ je uporabil regulacijski postopek, s katerim je spreminjal matriko, dokler ni le-ta postala nesingularna. Za odpravo singularnosti so bile kasneje predlagane tako imenovane neposredne metode LDA (angl. *direct LDA*) [111, 42], ki omogočajo izvedbo metode LDA neposredno na slikah brez predhodne preslikave le-teh v podprostor s pomočjo metode PCA. Avtorji neposrednih metod LDA trdijo, da podatki v podprostoru matrike $\mathbf{S}_W$, ki jih zavržemo zaradi lastnih vrednosti blizu ničle, nosijo več informacij za razlikovanje kot podatki, ki jih obdržimo.

Za prepoznavanje obrazov je bila predlagana tudi metoda, ki temelji na analizi neodvisnih komponent [2]. Zagovorniki te metode (za razliko od metode PCA, ki upošteva le statistiko drugega reda) trdijo, da se glavne informacije za ločevanje med posameznimi obrazi skrivajo v statistikah višjega reda. Naloga metode ICA je preslikati podatke tako, da so med seboj čimbolj neodvisni. Za prepoznavanje obrazov sta bili predlagani dve arhitekturi: prva določa statistično neodvisne bazne slike, ki določajo podprostor vseh slik [5], druga pa takšno preslikavo, ki omogoča statistično neodvisne parametre s katerimi so predstavljene posamezne slike [6].

2.3.2 Metode na osnovi značilk in geometrije

Običajno se pri metodah na osnovi značilk in geometrije najprej poišče lokalne značilke obraza kot so oči, nos in usta. Nato se za prepoznavo uporabi njihova lokacija ter njihove geometrijske in vizualne lastnosti. Prav geometrijske lastnosti posameznih značilk ter samega obraza so bile uporabljene tudi v prvih metodah za prepoznavanje obrazov. Za primerjavo so se uporabljale značilke kot so dimenzija obraza, razdalja med očmi in razdalja med očmi in usti [48], pa tudi razdalje in koti med robovi oči, mejami ustnic, nosnicama in vrhom brade [47].

Med najbolj uspešne metode te skupine pa vsekakor spada metoda na osnovi

grafov [74, 108], ki uporablja kot osnovo dinamično povezovalno arhitekturo (angl. *Dynamic Link Arhitecture*, DLA) [10, 55]. Glavno vlogo za predstavitev posameznih značilk igrajo valčki (angl. *wavelets*), še posebej Gaborjevi valčki. Tipično predstavitev značilke tako predstavlja množica valčkov različnih velikosti in nagibov. Tako definirani valčki različnih koeficientov so se izkazali za zelo robustne na različne osvetlitve, premike, popačenja, nagibe in velikosti. Primerjavo med slikami izvedemo tako, da posamezno sliko predstavimo z Gaborjevimi valčki v prej definiranih točkah na sliki. Te točke lahko določajo specifične obrazne značilke, lahko pa so enakomerno porazdeljene po sliki. Čeprav dobimo pri izračunu posameznega valčka vrednost faze in amplitude, za primerjavo uporabimo samo vrednost amplitude. Lastnost faze je namreč ta, da se z lokacijo na sliki zelo hitro spreminja, čeprav določa isto značilko. Vrednost faze je zato uporabna pri razlikovanju v primeru enake amplitude ali pa za določitev natančnejše lokacije značilk. Kasneje je bila metoda DLA razširjena z metodo ujemanja elastičnih grafov množic (angl. *Elastic Bunch Graph Matching*, EBGM) [108]. Za razliko od prejšnje predstavitve, vsebuje v metodi EBGM vsako vozlišče grafa množico značilk različnih posameznikov. S pomočjo tako definiranega grafa množic lahko predstavimo splošni model obraza.

2.3.3 Hibridne metode

Ugotovljeno je bilo, da uporablja človeški sistem zaznavanja za prepoznavanje tako posamezne značilke obraza kot tudi celoten videz obraza. Hibridne metode poskušajo to idejo uporabiti in zato za prepoznavanje obrazov uporabljajo tako sliko celotnega obraza kot tudi posamezne značilke. S pravo kombinacijo obeh pristopov poskušajo odpraviti slabosti enega in drugega. Tako zasnovane metode so lahko zaradi tega uspešnejše od metod iz drugih dveh skupin.

V skupini hibridnih metod bomo predstavili modularno metodo na osnovi lastnih prostorov [68], hibridno metodo, ki združuje metodo PCA in analizo lokalnih značilk (angl. *Local Feature Analysis*, LFA) [76], prilagodljiv model na osnovi videza [56] in sistem na osnovi komponent [36].

Enega prvih sistemov, ki je kombiniral sliko celotnega obraza s posameznimi značilkami je predlagal Pentland [68]. Koncept lastnih obrazov je uporabil za iskanje lastnih značilk kot so oči, usta, nos, itd. Preizkus na majhni množici obrazov je pokazal, da se največja razlika med metodama pojavi pri predstavitvi z majhnim številom podpornih vektorjev. Predstavitev z lastnimi značilkami se je izkazala za veliko boljšo od predstavitve z lastnimi obrazi, medtem ko je bila kombinirana metoda, ki je združevala obraz in značilke najuspešnejša. Izkazalo se je tudi, da je prepoznava s pomočjo značilk uspešnejša, če imamo opraviti s slikami obrazov, med katerimi so velike razlike (slika 2.7).

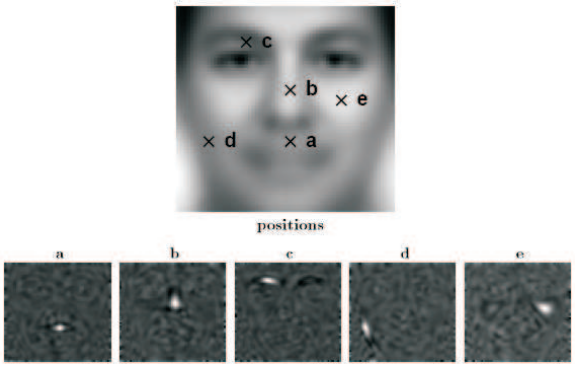
Avtorja Penev in Atick [76] sta predlagala hibridno metodo, ki združuje metodo PCA in metodo na osnovi lokalnih značilk – LFA. Metoda, ki sta jo izdelala je za globalno predstavitev obrazov uporabljala lastne slike obrazov in s tem uspešno zmanjšala šum na slikah, za predstavitev višjenivojskih značilk pa sta uporabila

Slika 2.7: Primerjava rezultatov ujemanja med: (a) testmi slikami in (b) slikami, ki jih dobimo s pomočjo metode na podlagi celotnega videza (angl. *eigenfaces*) ter (c) slikami, ki jih dobimo z metodo na podlagi videza posameznih značilk (angl. *eigenfeatures*).

metodo LFA.

Pri metodi LFA, kot biološko predlagani metodi, izhajamo iz dejstva, da se pri zaznavanju objektov v očesni mrežnici, ki vsebuje na milijone živčnih celic aktivira samo majhen del le-teh. Signal iz tako izbranega števila aktivnih živčnih celic potuje naprej v možgane, ki morajo ugotoviti kje in kakšen objekt se nahaja v vidnem polju ter kakšne so njegove lastnosti. Zaradi tega lahko tudi sliko objekta predstavimo v nižje dimenzijskem prostoru tako, da poiščemo ustrezno parametrizacijo. Za uspešno metodo za predstavitev normaliziranih frontalnih obrazov se je izkazala metoda PCA, ki uspešno zmanjša redundanco med posameznimi slikovnimi elementi. Celoten obraz namreč stimulira celotno polje živčnih celic, med katerimi vsaka določa določeno lokacijo na obrazu. Med temi celicami pa so nekatere tudi neaktivne. Da ugotovimo redundanco med njimi lahko s pomočjo metode LFA iz globalne predstavitve PCA izločimo topografske lokalne značilke. Za razliko od jeder metode PCA (angl. *PCA kernels*), ki ne vsebujejo topografskih informacij, pa jedra LFA v izbrani točki nosijo tudi informacijo o lokaciji (slika 2.8).

Izkazalo se je, da je glede na rekonstrukcijsko napako predstavitev s pomočjo metode LFA boljša od predstavitve s PCA. Rekonstrukcijska napaka pa se še dodatno zmanjša, če poleg LFA predstavitve ohranimo tudi PCA predstavitev. Tako



Slika 2.8: Jedra metode LFA za različne lokacije na sliki obraza.

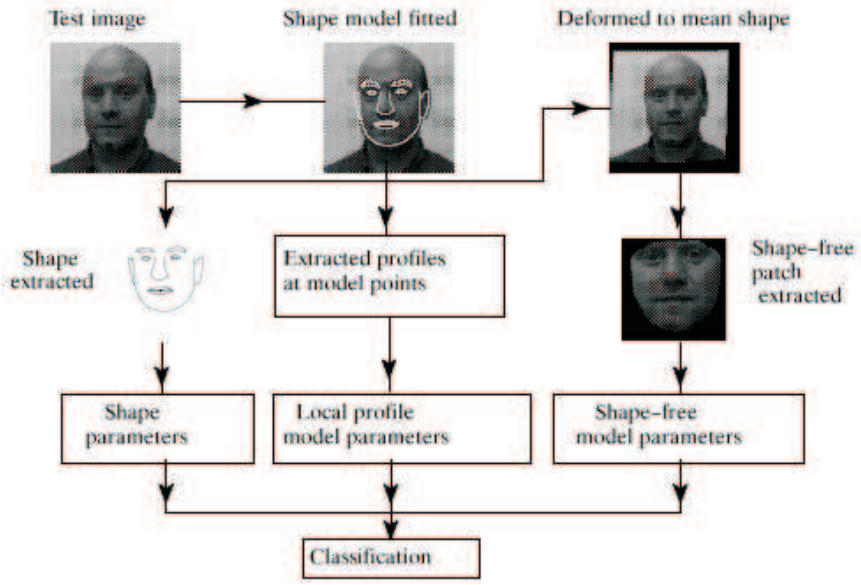
združeni globalna predstavitev s pomočjo PCA in lokalna predstavitev s pomočjo LFA potrjujeta, da so lahko hibridni sistemi učinkovitejši od metod iz posameznih skupin.

Avtor Lanitis je za avtomatsko prepoznavanje obrazov predstavil prilagodljiv model na osnovi videza [56]. Za predstavitev obraza je uporabil tako obliko kot tudi posamezne lastnosti značilk. Obliko je predstavil s pomočjo aktivnega modela oblik (angl. *Active Shape Model*, ASM). ASM je statistični model oblik različnih objektov, ki se iterativno prilagaja na obliko novega obraza. Model je naučen s pomočjo metode PCA, kjer parametri metode določajo koordinate na modelu oblike. Za namene klasifikacije pa se uporablja diskriminantna analiza, ki ločuje spremembe v obliki med posameznimi razredi.

Na podlagi povprečnega modela oblike je avtor izdelal globalno sivinsko sliko obraza, ki je neodvisna od oblike. Da je povečal robustnost modela glede na različen videz značilk pod različnimi pogoji, je za posamezne točke modela oblike izdelal tudi modele posameznih značilk. Za model značilk je uporabil kar pravokotni profil glede na značilko. Vse tako dobljene informacije (parametre modela oblike, parametre slike neodvisne od oblike in profile lokalnih značilk) je nato uporabil za izračun Mahalanobis razdalje, na podlagi katere je klasificiral slike neznanih osebe na vhodu (slika 2.9).

Pred kratkim je bila za prepoznavo obrazov predlagana metoda na osnovi komponent [36], ki razčleni obraz v posamezne komponente, kot so usta, oči in jih poveže v prilagodljiv model. Model je zelo podoben predstavitvi s pomočjo grafov množic – EBGM, vendar s to razliko, da so značilke namesto z Gaborjevimi valčki predstavljene s pomočjo sivinskih komponent različnih velikosti.

Razlog za uporabo takšnih komponent je v tem, da se pri različnih pogledih na obraz spremeni samo njihova lokacija, kar pa lahko rešimo s pomočjo prilagodljivega modela. Vendar za takšno predstavitev potrebujemo zelo raznoliko množico, ki vsebuje primere različnih pogledov in primere obrazov zajetih pod različnimi osvetlitvami.



Slika 2.9: Shema sistema za prepoznavanje obrazov, ki uporablja prilagodljiv model na osnovi videza.

Za odpravo tega problema je bil uporabljen prilagodljiv 3D model [8], ki omogoča generalizacijo slik obrazov različnih pogledov in različnih osvetlitev iz samo treh slik obrazov zajetih pod različnimi koti. Ko je 3D model zgrajen, lahko generiramo slike velikosti 85×85 slikovnih elementov s pomočjo katerih lahko potem metodo za iskanje in za prepoznavanje učimo. Za iskanje obrazov se uporablja 14 komponent, za prepoznavanje pa 9 komponent, ki se med seboj ne prekrivajo in vsebujejo sivinsko strukturo. Teh 9 komponent skupaj z regijo obraza tvori vektor značilk, ki kasneje služi za klasifikacijo s pomočjo metode podpornih vektorjev (angl. *Support Vector Machines*, SVM). Tako združena hibridna metoda, ki vsebuje komponente lokalnih značilk in celotno regijo obraza se je izkazala za zelo uspešno in dosega veliko boljše rezultate kot metoda, ki deluje samo na osnovi regije obraza.

2.3.4 Hibridni sistemi

Hibridne metode so tiste, ki kombinirajo različne tehnike za izločanje značilk, njihovo predstavitev in klasifikacijo. Hibridni sistemi pa so v glavnem sestavljeni iz različnih metod, ki lahko delujejo tudi samostojno.

V hibridni sistem lahko metode kombiniramo vzporedno ali zaporedno. Glavna prednost teh sistemov je v tem, da lahko s pravo kombinacijo odpravimo pomanjkljivosti posameznih metod. Pri vzporedno zgrajenem sistemu lahko uspešnost posameznih metod izboljšamo tako, da ustrezno združimo rezultate posamezne samostojne metode. Naslednji način delovanja hibridnih sistemov je zaporedna ali serijska ve-

zava. Pri takšnem načinu izhod prve metode predstavlja vhod v naslednjo metodo. Pri tako zgrajenih sistemih je bistveno to, da kombiniramo takšne klasifikacijske metode oziroma algoritme, ki med seboj dopolnjujejo svoje šibke točke. Postopek prepoznave mora biti zgrajen tako, da prva metoda poišče primerne kandidate, naslednja pa s pomočjo podrobnejšega pregleda določi končni rezultat.

Vzporedni hibridni sistemi

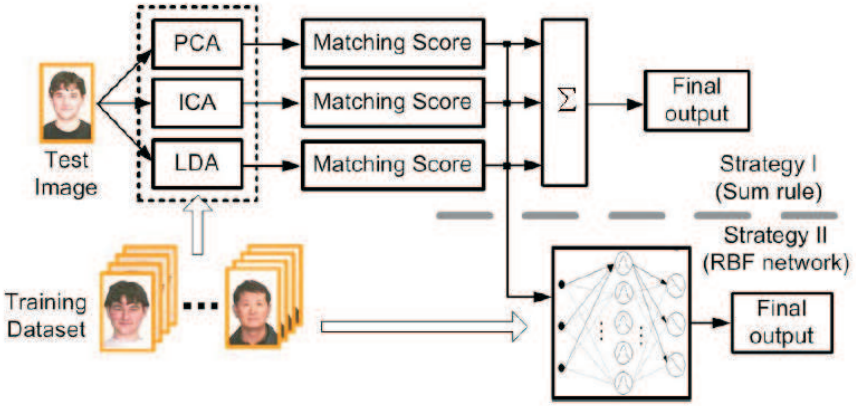
Gordon [26] je predlagal vzporedni hibridni sistem z združitvijo frontalne predstavitve obraza in predstavitve iz profila. Najprej je poiskal obrazne značilke, na podlagi katerih je obraz normaliziral in s pomočjo katerih je določil obrazne vzorce, ki so mu služili za prepoznavo regij na frontalnem obrazu in obrazu s profila (slika 2.10). Za prepoznavo je uporabil klasično metodo ujemanja z vzorcem. Po določitvi obraznih mej na frontalnem obrazu je na podlagi očesnega vzorca, iskanja zenice in strukture obraza, izločil oči. Podoben postopek je izvršil na obrazu s profila. Najprej je izločil obris profila in na podlagi tega določil lokacijo nosu in brade. Končen rezultat o odločitvi identitete osebe se tako določi na podlagi petih vzorcev obraza: levega in desnega očesa, nosu, ustnic in pogleda s profila.

Slika 2.10: Obrazni vzorci frontalnega obraza in obraza s profila vzporednega hibridnega sistema za prepoznavanje obrazov.

Rezultati so pokazali, da združitev frontalnega pogleda s pogledom s profila bolj zmanjša napako identifikacije, kot v primeru, če bi uporabili samo frontalne značilke obraza.

Avtorja Marcialis in Roli sta opisala sistem, ki združuje dve znani statistični metodi LDA in PCA [64]. S pomočjo obeh metod sta določila vektorje razdalj do vseh obrazov v podatkovni bazi. Končno odločitev sta izbrala s kombinacijo obeh vektorjev razdalj na podlagi algoritma k-najbližjih sosedov (angl. *k-nearest neighbours*). Rezultati potrjujejo večjo uspešnost kombinirane metode pred posameznima metoda PCA in LDA. Poudarila sta tudi, da bi za boljše ujemanje in s tem delovanje združenih metod potrebovala boljše poznavanje delovanja takšnega sistema. Avtor Lu et al. [60] je poleg metod LDA in PCA uporabil še metodo ICA. Za končno

odločitev je uporabili pravilo vsote in mrežo RBF (angl. *Radial Basis Function*). S kombinacijo vseh treh metod in pravilom vsote je v primerjavi s posamezno metodo dosegel najboljše rezultate. Odločanje na podlagi mreže RBF pa je rezultate še dodatno izboljšalo (slika 2.11).



Slika 2.11: Shema hibridnega sistema za prepoznavanje obrazov, ki združuje tri statistične metode: PCA, ICA in LDA in dve odločitveni strategiji: pravilo vsote in mrežo RBF.

Tudi avtorja Achermann in Bunke [1] sta za namene prepoznave obrazov združila tri metode: metodo skritih Markovih modelov (angl. *hidden Markov models*), metodo PCA in metodo na osnovi oblike profila. Tako definiran sistem dobi za odločitev informacije o določeni osebi iz popolnoma različnih virov. Rezultati so pokazali, da kombinacija dveh metod rezultate nekoliko izboljša. Dodatno izboljšanje pa dobimo, če uporabimo kombinacijo vseh treh metod.

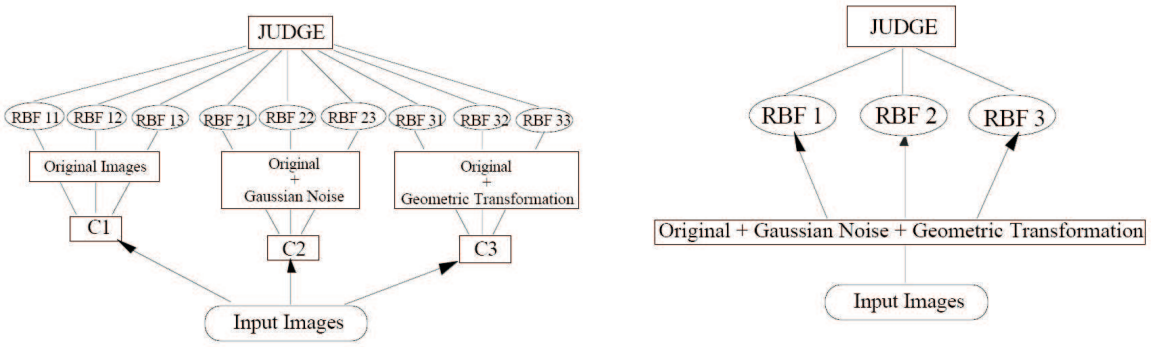
Za vse vzporedne hibridne sisteme je zelo pomembno, da ocene različnih metod, preden jih med seboj kombiniramo, ustrezno transformiramo. Šele tako transformirane ocene lahko med seboj primerjamo. Ocene so lahko razdalje, verjetnosti ali druge mere za kvaliteto. Idealno bi bilo, če bi bila razlika ocen med najboljšim ujemanjem in drugim najboljšim ujemanjem velika, kar bi bil velik dokaz, da je identifikacija pravilna. Najbolj običajne strategije za kombiniranje različnih ocen na nivoju odločitve so pravilo glasovanja, rangiranja in točkovanja. Pri pravilu glasovanja vsaka metoda dodeli en glas. Odločitev kateremu razredu pripada testna slika se določi na podlagi največjega števila glasov. Težava se lahko pojavi, če dosežeta posamezna razreda enako število glasov. V tem pogledu je boljša strategija rangiranja, kjer se za vsak razred izračuna vsota rangov. Izbran je tisti razred, ki ima najmanjšo vsoto. Pri strategiji točkovanja pa se odločitev naredi na podlagi ocene funkcije za posamezne metode. Ta strategija nam tako omogoča odločanje na podlagi dodatne informacije, ki jih pri pravilu glasovanja in rangiranja ne moremo dobiti.

Med hibridne sisteme uvrščamo tudi sisteme, ki uporabljajo poleg biometričnih

podatkov obraza tudi druge biometrične podatke kot so prstni odtisi ali govor. Tako lahko s pravo kombinacijo različnih biometričnih metod dosežemo zadovoljivo uspešnost tudi v okoljih, kjer je varnost kritična, na primer pri bančnih avtomatih. Na tržišču se že pojavljajo sistemi, ki združujejo biometrične podatke obraza, govora in prstnega odtisa, pa tudi slike šarenice, roženice in geometrijo dlani. Naslednji primer hibridnih sistemov so sistemi, ki kombinirajo sliko obraza posneto s kamero, ki sprejema vidno svetlobo in s kamero, ki sprejema infrardečo svetlobo. Prednost infrardečih slik je v tem, da niso občutljive na različne osvetlitve, kar je v veliki meri problem pri navadnih senzorjih, ki zaznavajo samo vidno svetlobo. Rezultati so pokazali veliko izboljšanje prepoznave pri uporabi kombinacije slik zajetih z navadno in infrardečo kamero [89, 87].

Zaporedni hibridni sistemi

Vse doslej predstavljene hibridne sisteme uvrščamo v skupino vzporednih hibridnih sistemov. Za konec pa predstavimo še dva zaporedna hibridna sistema. Prvi temelji na primerkih RBF mrež (angl. *Ensambles of Radial Basis Functions*, ERBF) in odločitvenih drevesih (angl. *Decision Trees*, DT) [31, 30]. Preizkušeni sta bili dve različni mrežni arhitekturi (slika 2.12). Pri prvi je bila mreža ERBF naučena na ločenih učnih množicah, pri drugi pa na eni učni množici, ki je vsebovala vse učne primere (originalne slike, slike z dodanim Gaussovim šumom in geometrijsko spremenjene slike). Izhod iz tako naučenih mrež je predstavljal vhod za odločitvena drevesa. Rezultati so pokazali, da je uporaba primerkov mrež RBF uspešnejša, kot zgolj uporaba mreže RBF in da je hibridno učenje izboljšalo uspešnost klasifikacije, ter da je učenje na združeni učni množici uspešnejše kot učenje na posamičnih množicah.

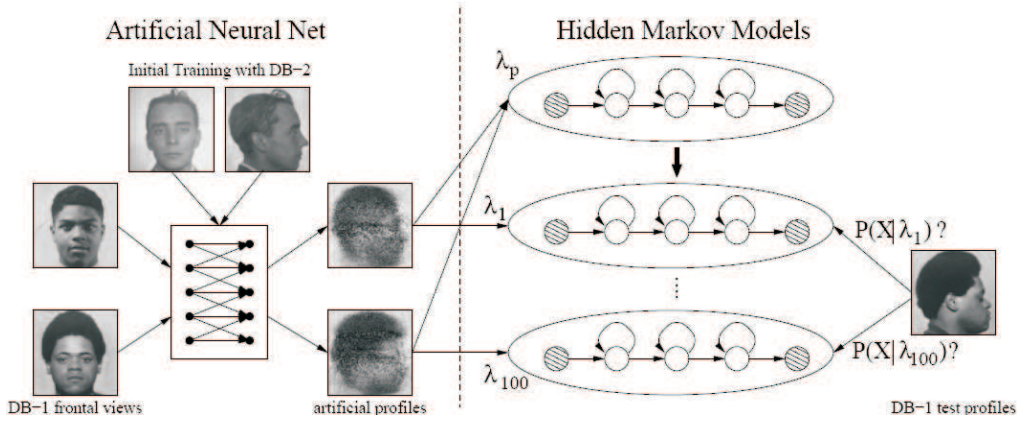


Slika 2.12: Dve shemi različnih arhitektur primerkov RBF mrež, ki sta bili uporabljeni kot prva stopnja v zaporednem hibridnem sistemu.

Avtorja Wallhoff in Rigoll [105, 104] sta preizkusila uspešnost prepoznave obrazov s profila na podlagi slike frontalnega obraza. Za ta namen sta uporabila hibridni

sistem, ki je kombiniral nevronsko mrežo in skrite Markove modele (slika 2.13). Nevronska mreža je na podlagi frontalnega obraza vrnila ustrezno predstavitev obraza s profila s pomočjo naučenih primerov. Na osnovi te predstavitve pa je s pomočjo skritih Markovih modelov poiskal ustrezen obraz.

Glavna ideja sistema je prepoznati osebo brez kakršne koli 3D informacije glave ali obraza. Zavedati se moramo, da je prepoznava osebe s profila, če imamo na voljo samo frontalno sliko, tudi za človeka zahtevna naloga. Sistem je bil preizkušen na bazi 100 ljudi in je dosegel uspešnost 60%. Uspešnost bi izboljšali, če bi bilo na voljo več učnih primerov za učenje nevronske mreže. Nikoli pa ne bi dosegli velike uspešnosti zaradi manjkajočih informacij o obrazu z različnih pogledov.



Slika 2.13: Zaporedni hibridni sistem, ki kombinira nevronsko mrežo in skrite Markove modele.

2.4 Povzetek poglavja

Prve metode za iskanje obrazov na slikah so našle na enotnem ozadju kvečjemu en obraz. Bile so zelo občutljive na različne svetlobne pogoje ter na nagib ali obrnjenost obraza. Današnje metode za iskanje obrazov pa poskušajo najti na sliki vse obraze, tudi če je ozadje različno in so obrazi različnih velikosti, ter pod različnimi koti. Uspešnost takšnih metod je okrog 90% z nekaj napačno najdenimi obrazi. Današnje metode so tudi že dovolj hitre, da jih lahko uporabljamo v realnem času.

Večji problem je pri prepoznavanju obrazov. Še vedno imajo metode za prepoznavanje težave pri obrazih, ki so različno osvetljeni ter, če je med zajetimi slikami preteklo veliko časa (podrobnejši pregled uspešnosti trenutnih metod bomo predstavili v poglavju 4). Današnje novejše metode največkrat uporabljajo za izboljšanje uspešnosti različne kombinacije že znanih metod. Pojavljajo se hibridne metode, ki kombinirajo različne tehnike izločanja značilk in različne klasifikatorje. V zadnjem

času pa se pojavljajo hibridni sistemi, ki kombinirajo različne metode prepoznavanja obrazov v vzporednem ali pa zaporednem načinu. S pomočjo prave kombinacije lahko takšni hibridni sistemi dosegajo boljše rezultati kot posamezne metode tako zgrajenega sistema.

Poglavje 3

Podatkovne baze in metodologije preizkusov

3.1 Uvod

Za preizkus metod za iskanje in prepoznavanje obrazov potrebujemo ustrezne podatkovne baze slik. Zajemanje slik obrazov je zelo zamuden in dolgotrajen proces, še posebej če hočemo posneti obraze pod določenimi pogoji in v daljših časovnih obdobjih. Skupaj z vedno novimi metodami se je pojavilo tudi vedno več podatkovnih baz. Nekatere podatkovne baze so tako namenjene samo določenim metodam in so zaradi tega specifične in največkrat zaščitene. Druge podatkovne baze pa so javno dostopne in namenjene širši množici raziskovalcev za namene preizkusa svojih metod. V tem poglavju bomo predstavili najpomembnejše javno dostopne podatkovne baze, ki se uporabljajo za namene iskanja in prepoznavanja obrazov na slikah.

Poleg vsake podatkovne baze so običajno še priporočila, kako določene slike oseb razdeliti na posamezne podmnožice. Tako lahko različni raziskovalci med seboj primerjajo različne metode. Podrobneje bomo predstavili podatkovni bazi FERET in AR ter njuni metodologiji, ki smo ju uporabili za izvedbo preizkusov.

3.2 Podatkovne baze obrazov

Obraz, kot spremenljiva in kompleksna struktura v 3D prostoru lahko pod različnimi pogoji izgleda zelo različno. Videz obraza je odvisen od samega posameznika pa tudi od položaja obraza, njegove osvetlitve, izraza na obrazu, starosti, prikrivanja in frizure. Če hočemo izdelati robustni sistem, ki je neodvisen od vseh teh sprememb, potrebujemo podatkovno bazo obrazov, ki vsebuje obraze zajete pod različnimi pogoji v skrbno kontroliranem okolju. Zelo pomembno je, da so takšne podatkovne baze tudi javno dostopne, da lahko različni raziskovalci preizkušajo svoje metode in primerjajo rezultate z ostalimi metodami za prepoznavanje obrazov.

Vzporedno z nastajanjem novih metod za prepoznavanje obrazov se je večalo

tudi število podatkovnih baz obrazov. Takšne podatkovne baze pa so bile največkrat narejene po meri specifičnim potrebam algoritmov, ki so jih uporabljali. Za namene naših preizkusov smo uporabili dve največji podatkovni bazi obrazov, ki sta javno dostopni in brezplačni v raziskovalne namene. Prva je podatkovna baza FERET [79], druga pa podatkovna baza AR [65]. Obe podatkovni bazi bomo podrobneje opisali v nadaljevanju, sedaj pa si poglejmo seznam preostalih podatkovnih baz, ki se tudi uporabljajo za preizkus uspešnosti prepoznave in iskanja obrazov na slikah.

Seznam vseh najpomembnejših podatkovnih baz, ki se uporabljajo za preizkus uspešnosti prepoznavanja obrazov prikazuje tabela 3.1. Poleg imena podatkovne baze smo navedli tudi število različnih oseb ter število vseh obrazov slik v podatkovni bazi. Slike oseb so bile zajete pod skrbno kontroliranimi pogoji kot so kontroliran položaj, osvetlitev, izraz na obrazu in v različnih časovnih intervalih. V tabeli smo navedli tudi število slik posamezne osebe, ki so bile zajete pod skrbno določenimi pogoji.

Kjer določen pogoj ni bil posebej kontroliran, ampak se slike vseeno razlikujejo, smo to označili z znakom "+". V primeru, da določen podatek ni znan smo to označili z znakom "-".

Med podatkovnimi bazami se pojavljajo tudi nekatere posebnosti. Tako podatkovni bazi *Equinox IR* in *CMU Hyper* vsebujeta slike obrazov posnete z infrardečo kamero. Podatkovni bazi *U. Texsas* (Univerza v Texasu) in *FIA* (angl. *Face In Action*) ne vsebujeta statičnih slik obrazov ampak video posnetke obrazov. Podatkovna baza *AGING* je nastala v okviru Evropskega projekta FG-NET (angl. *Face and Gesture Recognition Research Network*) in vsebuje slike oseb zajete v različnih časovnih obdobjih. Baza je zaradi tega zelo uporabna pri preučevanju obraznih sprememb v procesu staranja.

Poleg podatkovnih baz za preizkus metod za prepoznavanje obrazov pa so raziskovalci, ki se ukvarjajo z obrazno mimiko zbrali obraze različnih izrazov kot so: nevtralen, jezen, zgrožen, prestrašen, vesel, žalosten in presenečen. Nekateri raziskovalci pa so poleg osnovnih izrazov posneli tudi druge izraze in jih opisali s pomočjo obraznega sistema akcij [20]. V tabeli 3.2 so zbrane najpomembnejše podatkovne baze različnih obraznih izrazov.

Metode za iskanje obrazov na slikah uporabljajo za učenje pogosto kar podatkovne baze, ki so namenjene prepoznavanju (tabela 3.1). Med najpopularnejšimi bazami so podatkovna baza FERET, MIT, ORL, Harvard in AR. Poleg teh podatkovnih baz pa se velikokrat uporabljajo baze slik, ki niso javno dostopne. Ampak, če hočemo metode med seboj primerjati nujno potrebujemo skupne, javno dostopne podatkovne baze. Slike v takšnih bazah so velikokrat zajete pod različnimi koti, različno nagnjene, različne kvalitete in velikosti. Najpomembnejše podatkovne baze za preizkus uspešnosti metod za iskanje obrazov smo zbrali v tabeli 3.3.

Pri nekaterih metodah za iskanje obrazov potrebujemo poleg učnih slik obrazov tudi učne slike, kjer ni obrazov. Podatkovna baza *MIT CBCL1* vsebuje poleg slik obrazov tudi 28.121 primerov slik neobrazov. Naslednja zelo priljubljena zbirka slik neobrazov pa se nahaja na Univerzi v Washingtonu [99].

Baza	število		kontrolirani pogoji zajemanja			
	oseb	slik	položaj	osvetlitev	izraz	čas
AGING*	82	1002	+	+	+	+
AR*	136	3.315	1	4	4	2
BANCA	208	6.240	1	+	+	12
Caltech	27	450	-	+	+	1
CAS-PEAL	66–1040	30.900	21	9–15	6	2
CMU PIE	68	41.368	13	43	4	1
CVL	114	798	5	1	3	-
FERET*	1199	14.051	9–20	2	2	2
Georgia	50	750	+	+	+	2+
H. Scan*	23	1.518	1	-	-	1
Harvard RL	10	-	1	77–84	1	1
KFDB	1000	52.000	7	16	5	1
MIT	16	433	3	3	1	1
MIT-CBCL	10	324+200	+	+	-	1
MPI	200	-	3	3	1	1
ND HID	300+	15.000+	1	3	2	10/13
NIST	1573	-	2	-	-	-
NIST MID	1573	3.248	2	1	+	1
NLPR	22	660	+	1	-	-
ORL	40	400	1	+	+	+
PICS	100	700	1	-	-	1
	100	100	1	-	-	1
	116	690	1	2	+	2
Rich. MIT	74	444	3	-	-	1
U. Essex	395	7.900	1	+	+	1
U. Oulu	125	2.000	1	16	1	1
UMIST	20	564	+	1	-	1
VALID*	106	-	-	+	+	5
XM2VTS*	295	-	+	1	+	4
Yale B*	10	5.850	9	64	1	1
Yale*	15	165	1	3	6	1
Equinox IR	91	-	1	3	3	1
CMU Hyper	54	-	2	4	1	1–5
U. Texas	284	-	+	1	+	1
FIA	200	-	3	+	+	3

Tabela 3.1: Seznam podatkovnih baz za preizkus uspešnosti prepoznavanja obrazov. (* Podatkovna baza vsebuje označene značilke obraza.)

Baza	št. oseb	št. izrazov	velikost
JAFFE	10	7	256×256
U. Maryland	40	6	560×240
Cohn-Kanade	100	23	640×480

Tabela 3.2: Seznam podatkovnih baz za preizkus uspešnosti prepoznavanja različnih obraznih izrazov.

Mi smo za preizkus metod za prepoznavanje obrazov uporabili podatkovni bazi FERET in AR. Podatkovni bazi sta za raziskovalne namene brezplačni in vsebujeta veliko število različnih oseb in veliko število slik posamezne osebe. V nadaljevanju bomo podrobneje predstavili omenjeni podatkovni bazi. Podrobnejše opise ostalih podatkovnih baz, predstavljenih v tabeli 3.1 pa lahko najdete v [28, 27, 23, 73].

Baza	št. slik	št. obrazov	frontalni	profil	nagnjeni
UCFI	100	299	182	26	91
MIT	23	155	155	0	0
CMU	107	352	352	0	0
CMU Set II	208	441	-	347	-
MIT/CMU	180	730	507	0	223
BioID*	1521	1521	1521	0	0
MIT CBCL1	-	2901	-	-	-

Tabela 3.3: Pregled podatkovnih baz za preizkus uspešnosti iskanja obrazov. (* Podatkovna baza vsebuje označene značilke obraza.)

3.2.1 Podatkovna baza FERET in njena metodologija

Podatkovno bazo FERET (angl. *Facial Recognition Technology*) so pripravili na Univerzi George Mason s pomočjo ameriškega laboratorija za vojaške raziskave v okviru programa FERET, ki ga je sponzoriral Ameriški obrambni oddelek za razvoj tehnologije proti drogi (angl. *US Department of defense counterdrug technology development*) [79, 77].

Podatkovna baza FERET se je uporabljala za namene preizkusov uspešnosti metod za prepoznavo obrazov v preizkusnih programih FERET in FRVT 2000 [7]. Raziskovalci jo zaradi tega radi uporabljajo kot nekakšno standardno bazo za preizkus svojih metod ter za primerjavo uspešnosti različnih metod [9]. Baza vsebuje točno določene podmnožice slik, ki so namenjene učenju, testiranju in primerjavi testnih slik. Velikokrat pa raziskovalci sami definirajo svoje učne množice ali celo svoje testne množice, zato so rezultati takšnih preizkusov med seboj težko primerljivi, pa čeprav uporabljajo isto podatkovno bazo. Tudi v primeru, da so uporabljene iste učne in testne množice pa so lahko uporabljeni različni postopki geometrijske in svetlobne normalizacije, kar onemogoča neposredno primerjavo rezultatov takšnih preizkusov.

Slabost učne množice podatkovne baze FERET je veliko prekrivanje s testnimi množicami in z množico galerije, s katero izvršimo primerjavo. V tabeli 3.4 predstavljamo različne učne množice, ki jih običajno uporabljajo raziskovalci za učenje svojih metod. Prikazano je tudi prekrivanje tako definirane učnih množic s testnimi množicami in množico galerije. Nekateri raziskovalci pa se odločijo za drugačno strategijo in iz galerije slik naključno izberejo nekaj slik, ki jih uporabijo za učenje. Takšen postopek potem nekajkrat ponovijo in kot končni rezultat prikažejo povprečje rezultatov tako izbranih baz.

Zaželeno je, da je prekrivanj med posameznimi množicami čimmanj. Priporočeno je, da prekrivanj sploh ne bi bilo, kar je pri podatkovnih bazah z malo slikami zelo težko zagotoviti. Mi bomo v svojih preizkusih uporabili predlagano učno množico *CSU_training*, ki se glede na število učnih slik še najmanj prikriva s testnimi množicami. Testno množico *CSU_training* so predlagali na Univerzi Colorado in jo tudi uporabili za učenje in preizkus svojega sistema za identifikacijo [9]. Tej učni množici smo dodatno dodali še 506 slik iz podatkovne baze FERET, ki niso bile vsebovane v nobeni od uporabljenih množic. Tako vsebuje naša učna množica 1007

učna množica	št. oseb	št. slik	galerija: fa	testne množice		
				fb	fc	$dup1$
$feret_train_cd$	314	736	270 (22.5%)	-	270 (22.5%)	184 (25%)
$CSU_training$	428	501	396 (33%)	-	-	99 (13.7%)
$feret_train_x2$	72	144	70 (5.8%)	-	-	73 (10%)
$train3x255$	225	675	224 (18.7%)	224 (18.7%)	194 (100%)	3 (0.28%)

Tabela 3.4: Učne množice različnih avtorjev in njeno prekrivanje z galerijo in testnimi množicami. Baza FERET vsebuje tudi testno množico $dup2$ s katero pa ni prekrivanj.

slik obrazov 504 različnih oseb. Posamezna oseba vsebuje različno število slik obrazov (337 oseb samo eno, 3 osebe pa celo 10). V povprečju vsaka oseba vsebuje 2 sliki obraza. Prekrivanje naše učne množice je enako kot prekrivanje učne množice $CSU_training$, s tem da je od vseh 1007 učnih slik 39,3% (396) vsebovanih v galeriji in 9,8%(99) v testni množici $dup1$. Z testnimi množicami fb, fc in $dup2$ ni prekrivanj.

Posamezne slike so znotraj baze FERET razdeljene v 24 različnih kategorij (tabela 3.5). Slike so zajete v 15 različnih časovnih obdobjih med avgustom 1993 in julijem 1996. Ker so morali za zajem slik vedno znova postavljati opremo, je lahko opaziti minimalne razlike v položaju med posameznimi snemalnimi obdobji. Vse slike so zajete z 35 mm analogno kamero in kasneje digitalizirane v 8 bitne sivinske slike velikosti 256×384 slikovnih elementov.

Tako pripravljene slike so nato avtorji baze razdelili v galerijo slik fa in posamezne testne množice: fb, fc, $dup1$ in $dup2$. Slika 3.1 prikazuje slike iste osebe iz različnih testnih množic. Slike iz množic fa in fb so bile zajete ena za drugo, s tem da je morala oseba pri drugem posnetku spremeniti izraz na obrazu. Običajno imajo osebe v eni množici na obrazu nasmeh v drugi pa nevtralen izraz. Slike množice fc so bile zajete z drugo kamero in pod drugačnimi svetlobnimi pogoji. Pri nekaterih osebah so čez določen čas snemanje ponovili. Tako vsebuje množica $dup1$ slike, ki so bile zajete od 0 do 1031 dni po prvem snemanju (mediana znaša 72 dni, povprečna vrednost 251 dni). Množica $dup2$ pa vsebuje podmnožico tistih slik iz $dup1$, pri katerih je časovni zamik najmanj 18 mesecev (mediana znaša 569 dni, povprečna vrednost 627 dni).

V tabeli 3.6 prikazujemo, koliko slik vsebuje posamezna množica. Množice se med seboj ne prekrivajo, razen množice $dup2$, ki je podmnožica množice $dup1$. Množica fa predstavlja galerijo slik in v postopku prepoznave služi za primerjavo s slikami iz testnih množic. Sestavljajo jo slike iz kategorij Fa (1002 slik) in ba (194 slik) (glej tabelo 3.5). Množica fb vsebuje slike iz kategorij Fb (1001 slik) in bj (194 slik). Množica fc je podmnožica kategorije bk (194 slik). Množica $dup1$ pa vsebuje kategorije Fa (357 slik), Fb (353), ba (6 slik) in bj (6 slik). Množica $dup2$ vsebuje 234 slik množice $dup1$ in sicer 111 slik iz kategorije Fa in 111 slik iz kategorije Fb ter vse slike kategorij ba in bj, ki so prisotne v množici $dup1$.

Preostale kategorije slik pa vsebujejo obraze v različnih položajih. Slika 3.2 prikazuje primere slik osebe z različnimi položaji glave iz kategorij od ba do bi. Med

koda	poza (stopinje)	opis	št. slik	št. oseb
Fa	0 frontalen	običajni izraz na obrazu	1762	1010
Fb	0	drugačen izraz na obrazu	1518	1009
ba	0	frontalni obrazi serije “b”	200	200
bj	0	drugačen izraz kot pri ba	200	200
bk	0	drugačna osvetlitev kot pri ba	200	200
bb	+60	vidno desno lice	200	200
bc	+40		200	200
bd	+25		200	200
be	+15		200	200
bf	-15	vidno levo lice	200	200
bg	-25		200	200
bh	-40		200	200
bi	-60		200	200
ql	-22.5	za četrtino obrnjen obraz	763	508
qr	+22.5		763	508
hl	-67.5	za polovico obrnjen obraz	1246	904
hr	+67.5		1298	939
pl	-90	levi in desni profil	1318	974
pr	+90		1342	980
Ra	+45	Naključno zajete slike. Pozitivni kot označuje osebo, ki je obrnjena v desno.	322	264
Rb	+10		322	264
Rc	-10		613	429
Rd	-45		292	238
Re	-80		292	238

Tabela 3.5: 24 kategorij slik znotraj podatkovne baze FERET.

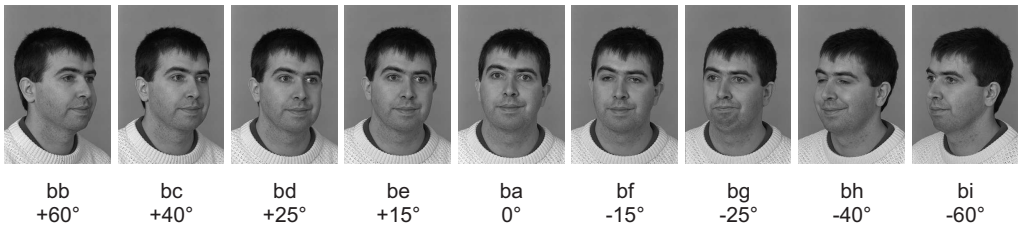
fa fb fc dup1 dup2

Slika 3.1: Slike iste osebe iz različnih testnih množic podatkovne baze FERET.

posameznimi snemanji je morala oseba obrniti glavo in telo za določen kot. Razpon v obratu je bil 60 stopinj v obeh smereh. Število oseb, ki so jih posneli na takšen način je v podatkovni bazi 200.

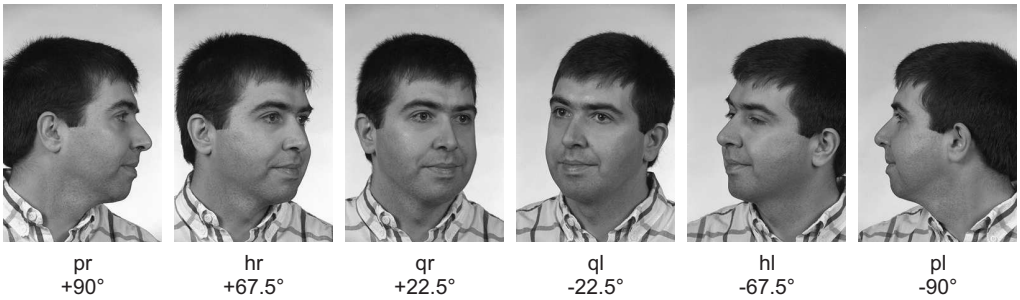
množica	št. slik
galerija slik, fa	1196
testna množica, fb	1195
testna množica, fc	194
testna množica, $dup1$	722
testna množica, $dup2$	234

Tabela 3.6: Število slik galerije in posamezne testne množice v podatkovni bazi FERET.



Slika 3.2: Primeri slik različno obrnjene glave iste osebe iz kategorij od *ba* do *bi* podatkovne baze FERET.

Slika 3.3 prikazuje drugo skupino različnih položajev. Tukaj vsebujejo posamezni položaji desni in levi profil (kategoriji *pr* in *pl*), desni in levi četrtinski profil (kategoriji *qr* in *ql*) in desni in levi polovični profil (kategoriji *hr* in *hl*). Število tako posnetih oseb je v podatkovni bazi od 580 do 980. Podatkovna baze vsebuje še slike zajete v naključnih položajih (kategorije R) in sicer od 264 do 429 različnih oseb.



Slika 3.3: Primeri slik iste osebe s profila, kategorij pr/pl, polovičnega profila kategorij hr/hl in četrtinskega profila, kategorij qr/ql iz podatkovne baze FERET.

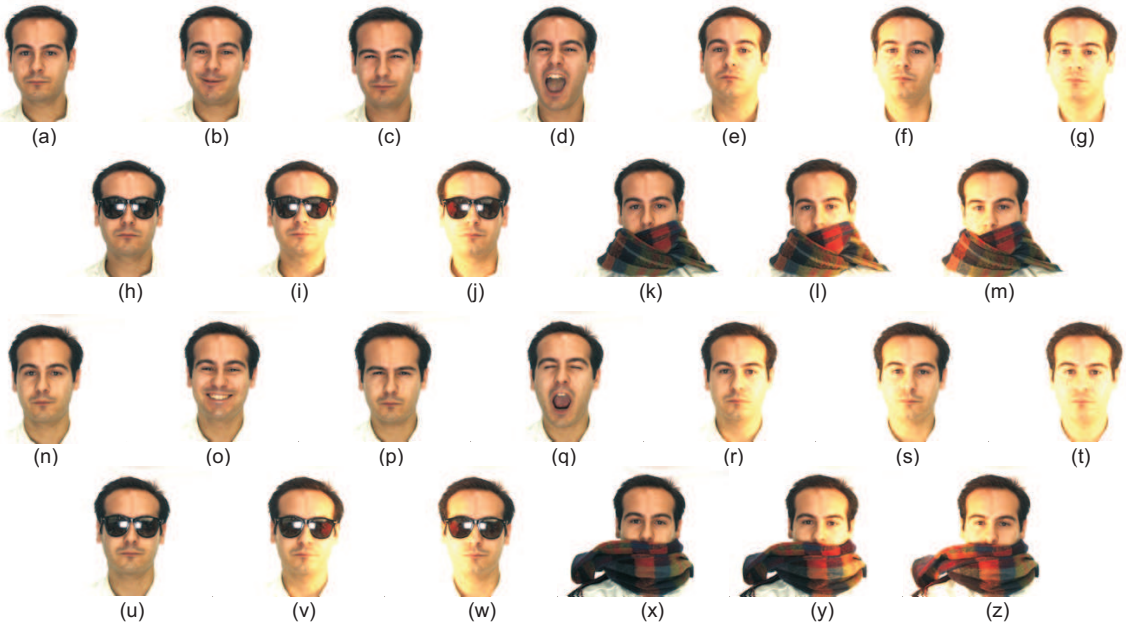
Vsaka slika podatkovne baze FERET je opremljena tudi z dodatnimi informacijami, kdaj je bila zajeta in ali oseba nosi očala ali ne. Za 3.819 slik so podane tudi ročno označene lokacije oči, nosa in ust.

Leta 2003 je bila podatkovna baza FERET dopolnjena z barvno različico istih

slik večje dimenzije 512 × 768. Ponovno so bile digitalizirane skoraj vse slike iz prejšnje sivinske različice in dopolnjene so bile nekatere informacije o slikah. Med pomembnejšimi informacijami so spol osebe, rasa ter informacija ali ima oseba brke ali brado.

3.2.2 Podatkovna baza AR in njena metodologija

Slike za podatkovno bazo AR [65] so zbrali leta 1998 v centru za računalniški vid v Barceloni v Španiji. Baza vsebuje 135 različnih oseb od tega 59 žensk in 76 moških. Med slikanjem so neprestano kontrolirali pogoje zajemanja (oddaljenost od kamere, parametre kamere, jakost osvetlitve) in s tem poskušali zagotoviti enake pogoje za vse osebe. Zajete slike so velikosti 768 × 576 slikovnih elementov. Vsako osebo so posneli dvakrat v časovnem intervalu 2 tednov. Med enim snemanjem so zajeli različne obrazne izraze, obraz pod različnimi osvetlitvami in različno prikrite obraze. Slika 3.4 prikazuje primere slik iste osebe zajete pod različnimi pogoji in v dveh različnih časovnih obdobjih.



Slika 3.4: Primeri slik iste osebe podatkovne baze AR zajetih pod različnimi pogoji in v dveh časovnih obdobjih.

Oseba je morala med snemanjem štirikrat spremeniti izraz na obrazu: (1) naravni (slika 3.4(a)), nasmeh (slika 3.4(b)), jeza (slika 3.4(c)) in krik (slika 3.4(d)). Obraz so osvetlili z različnih strani: z leve (3.4(e)), z desne (3.4(f)) in z obeh strani(3.4(g)). Oseba je morala nositi sončna očala (3.4(h)) pri normalni osvetlitvi in pri osvetlitvi z leve (3.4(i)) ter desne (3.4(j)) strani. Na koncu pa je morala oseba obraz prekriti s šalom (3.4(k)). Prekrit obraz so zajeli tudi pri osvetlitvi z leve (3.4(l)) in desne

(3.4(m)). Preostale slike obrazov od 3.4(n) do 3.4(z) pa so bile zajete po dveh tednih pod enakimi pogoji.

V okviru programa FG-NET so označili od 2 do 4 slike obrazov posamezne osebe z 22 točkami. Točke so večinoma označevale obrazne značilke in njihove robove. Označili so predvsem slike različnih izrazov (3.4(a), (b) in (c)) in z leve osvetljene obraze (3.4(e)). Da smo lahko normalizirali slike celotne baze smo morali ročno označili še preostale slike in sicer samo lokacijo oči.

Posamezno osebo predstavlja tako 26 različnih slik obraza, kar uvršča podatkovno bazo AR med podatkovne baze z največ slikami na posamezno osebo. V bazi ni posebej določenih množic za učenje in preizkus, zato si vsak avtor, ki uporablja to množico, izbere svojo razdelitev. Nekateri si izberejo samo eno ali dve sliki za učenje in preostale za preizkus. Nekateri pa si izberejo več slik za učenje in samo eno ali pa dve za preizkus. Večina avtorjev pri svojih preizkusih ne upošteva delno zakritih slik ter slik obeh množic, ki sta bili posneti v različnih časovnih obdobjih.

Mi smo v svojih preizkusih upoštevali predlog avtorja podatkovne baze AR, Alexa Martíneza [64] in v učno množico vključili slike s prvega snemanja, v testno množico pa slike z drugega snemanja. Ker naše metode predpostavljajo, da je viden celoten obraz smo iz učne in testne množice izključili delno zakrite obraze (na sliki 3.4 od (h) do (m) in od (u) do (z)). Tako vsebuje učna množica 7 slik (na sliki 3.4 obrazi od (a) do (g)), testna pa 7 slik drugega snemanja (na sliki 3.4 obrazi od (n) do (t)). Takšen preizkus bi skladno s preizkusi podatkovne baze FERET lahko označili z *dup*, ker gre za primerjavo slik zajetih v različnih časovnih obdobjih. V našem primeru je to 2 tedna. Testno množico smo dodatno razdelili na dve podmnožici. V prvo podmnožico, ki smo jo skladno s prejšnjimi oznakami označili fb_{AR}, smo vključili 4 slike različnih izrazov (na sliki 3.4 od (n) do (q)). V drugo množico (fc_{AR}) smo vključili preostale 3 slike različne osvetlitve. Kot smo že predhodno omenili lahko skupen rezultat obeh testnih množic označimo kot dup_{AR}.

3.3 Povzetek poglavja

Med vsemi podatkovnimi bazami, ki so javno dostopne, se za preizkus različnih metod največkrat uporablja podatkovna baza FERET. Glavna prednost te baze je, da vsebuje zelo veliko število oseb. Na osnovi te podatkovne baze je bilo izdelanih kar nekaj standardnih preizkusov, ki jih bomo podrobneje opisali v naslednjem poglavju, zato je uporabnost te baze še toliko večja. Avtorji lahko svojo metodo preizkusijo tako, da upoštevajo metodologijo, ki jo določa protokol FERET in primerjajo svoje rezultate z ostalimi. Slabost te podatkovne baze pa je v tem, da vsebuje zelo malo slik, ki pripadajo posamezni osebi (število slik znotraj razreda). Baza, ki vsebuje veliko število slik posamezne osebe pa je podatkovna baza AR. Zaradi tega smo poleg podatkovne baze FERET za naše preizkuse uporabili tudi bazo AR, s pomočjo katere smo poskušali ugotoviti ali število slik na osebo vpliva na uspešnost metod.

Poglavje 4

Pregled standardnih preizkusov FERET in FRVT

4.1 Uvod

Metode za prepoznavanje obrazov so bile v zgodovini že velikokrat preizkušene na organiziranih neodvisnih preizkusih – FERET [80, 79, 77] in preizkusih namenjenim komercialnim izdelkom – FRVT (angl. *Face Recognition Vendor Tests*) [7, 78].

Prvi preizkus FERET je bil izveden avgusta 1994, naslednji marca 1995 in zadnji septembra 1996. Pri teh preizkusih so sodelovale različne raziskovalne inštitucije s svojimi metodami za prepoznavanje obrazov. Prvi preizkus komercialnih izdelkov za prepoznavanje obrazov je bil opravljen leta 2000, naslednji je bil izveden leta 2002, zadnji pa leta 2006. Vsi ti preizkusi so pokazali določen napredek na področju prepoznavanja obrazov, izpostavili so najobetavnejše metode in s tem začrtali smer razvoja metod za prepoznavanje v prihodnosti.

Pri vsakemu preizkusu je zelo pomemben protokol preizkusa. Protokol preizkusa določa kako naj preizkus poteka in kako naj se izračunajo rezultati. V nadaljevanju bomo podrobneje predstavili protokola FERET in FRVT 2002. Protokol preizkusa FRVT 2002 bazira na protokolu preizkusa FERET in za razliko od protokola FERET določa splošen postopek, ki se lahko uporablja tudi za druge biometrične podatke in ne samo za obraze.

S pomočjo preizkusov lahko izračunamo različne statistike in tako prikažemo rezultate uspešnosti za različne scenarije. Ugotovimo lahko uspešnost metod za identifikacijo na odprti ali zaprti množici ter uspešnost verifikacije.

4.2 Standardni kazalci uspešnosti

Pri prepoznavanju obrazov oziroma na splošno pri biometriji lahko merimo uspešnost treh nalog: verifikacije, identifikacije na odprti in identifikacije na zaprti množici. Vsaka naloga ima svoje mere uspešnosti. Med seboj so si vse tri naloge zelo podobne.

Na identifikacijo na odprti množici pa lahko gledamo kot na splošno nalogo, ki pokriva obe ostali nalogi.

Biometrični sistem temelji na biometričnih vzorcih, ki označujejo posamezne lastnosti človeka. Primeri biometričnih vzorcev so slika obraza, prstni odtis ali slika šarenice. Biometrični vzorec lahko sestavljajo podatki iste biometrične lastnosti, na primer 5 slik obraza zajetih pod različnimi pogoji, lahko pa ga sestavlja kombinacija različnih biometričnih podatkov, na primer obraz in prstni odtis.

Za izračun uspešnosti moramo ločiti 3 različne množice slik. Prva množica je galerija G, ki vsebuje biometrične vzorce poznanih oseb. Drugi dve množici sta testni množici (P_G in P_N). Slika iz testne množice je primer biometričnega vzorca, za katerega mora sistem prepoznati komu pripada, pri čemer je prepoznava lahko identifikacija ali verifikacija. Prva testna množica P_G vsebuje biometrične vzorce o ljudeh, katerih vzorce imamo tudi v galeriji (vzorci v testni množici so različni od tistih v galeriji). Druga testna množica P_N pa vsebuje biometrične vzorce tistih ljudi, ki jih ni v galeriji.

Identifikacija na zaprti množici je standardna naloga za preizkus uspešnosti metod za prepoznavanje obrazov in jo zato imenujemo krajše kar identifikacija ali prepoznava. Pri tej nalogi je tipično vprašanje, ki ga zastavimo sistemu: Čigav obraz je to? To vprašanje je primerno za identifikacijo na zaprti množici, kajti testna slika zagotovo pripada eni izmed oseb v galeriji.

Pri identifikaciji na odprti množici lahko testna slika ne pripada nobeni osebi iz galerije. Osnovno vprašanje, ki si ga zastavimo je: Ali poznamo ta obraz? Pri identifikaciji na odprti množici mora sistem najprej ugotoviti ali testna slika pripada osebi iz galerije. Če se sistem odloči, da je oseba v galeriji mora sporočiti njeno identiteto. Glede na velikost galerije ločimo dve specifični nalogi. Pri prvi nalogi, kjer je galerija slik majhna, iščemo po seznamu (angl. *watch list task*). V drugem primeru, ko je galerija slik velika, pa gre za preiskovanje velike baze podatkov (angl. *mugshot book searching*). Pri identifikaciji prstnih odtisov imenujemo takšen sistem preiskovanja velike baze podatkov krajše kar AFIS (angl. *Automatic Fingerprint Identification System*). Na identifikacijo lahko gledamo tudi kot na primerjavo eden z mnogim (1:N), ker gre za primerjavo testnega vzorca osebe z vsemi vzorci v galeriji.

Pri nalogi verifikacije, ponudimo sistemu biometrični vzorec poznane osebe in zahtevamo potrditev njene identifikacije. Naloga sistema je preveriti ali vzorec res pripada tej osebi. Pri verifikaciji je osnovno vprašanje: Ali je to res oseba za katero se izdaja? Na verifikacijo lahko gledamo tudi kot na primerjavo eden z enim (1:1).

4.2.1 Identifikacija na odprti množici

Na identifikacijo na odprti množici lahko gledamo tudi kot na posplošen primer naloge, ki vsebuje verifikacijo in identifikacijo na zaprti množici kot posebna primera. Pri nalogi identifikacije na odprti množici sistem preveri, če testna slika p_j pripada osebi v galeriji G. Če testna slika pripada eni izmed oseb v galeriji, algoritem prepozna kateri osebi pripada.

Galerija G vsebuje množico biometričnih vzorcev $g_1, ..., g_{|G|}$, pri čemer en biometrični vzorec pripada eni osebi. Kadar testni vzorec p_j podamo sistemu, se ta primerja s celotno galerijo vzorcev. Na podlagi primerjave med testnim vzorcem p_j in vzorcem iz galerije g_j dobimo oceno podobnosti s_{ij} (angl. *similarity score*). Večja kot je ocena podobnosti bolj sta si vzorca med seboj podobna (Razdaljo med vzorcema lahko spremenimo v oceno podobnosti z negacijo le-te). Kadar vzorca g_i in p_j pripadata isti osebi, imenujemo oceno podobnosti s_{ij} tudi ocena ujemanja (angl. *match score*), kadar pripadata vzorca različnima osebama, pa ocena neujemanja (angl. *nonmatch score*). Če biometrični vzorec p_j pripada osebi iz galerije, definirajmo g^* kot najboljše ujemanje v galeriji. Oceno podobnosti med vzorcema p_j in g^* definirajmo s s_{*j} in funkcija id$(\cdot)$ nam vrne identiteto biometričnega vzorca tako, da velja id(p_j)=id(g^*). Pri identifikaciji moramo izračunati vse ocene podobnosti med testnim vzorcem in galerijo ter jih urediti po velikosti. Testni vzorec p_j ima rang n, če je s_{*j} v tako urejenem seznamu ocen podobnosti na n-tem mestu. Določimo funkcijo rang(p_j)=n, ki vrne rang za izbrani testni vzorec. Rang 1 imenujemo tudi največje ujemanje (angl. *top match*). Uspešnost identifikacije na odprti množici določata dve statistiki: stopnja detekcije in identifikacije (angl. *the detection and idetification rate*) ter stopnja lažnega alarma (angl. *false alarm rate*). Najprej si poglejmo primer, ko pripada testni vzorec eni izmed oseb v galeriji ($p_j \in P_G$). Testni vzorec je najden in pravilno prepoznan, če sistem najde identiteto testnega vzorca in če je ocena ujemanja nad določeno mejno vrednostjo τ. Formalno lahko pogoja zapišemo kot:

- rang(p_j)=1 in
- $s_{*j} \geq \tau$ za oceno ujemanja, kjer je id(p_j)=id(g^*) ,

za vnaprej določeno mejno vrednost τ. Stopnjo detekcije in identifikacije določa tisti del testnih primerov v P_G, ki so bili uspešno najdeni in prepoznani. Stopnja detekcije in identifikacije je funkcija mejne vrednosti τ. Tako lahko zapišemo stopnjo detekcije in identifikacije P_{DI} kot funkcijo parametra τ

$$P_{DI}(\tau, 1) = \frac{|\{p_j : \text{rank}(p_j) = 1 \text{ in } s_{*j} \geq \tau\}|}{|P_G|} \quad . \tag{4.1}$$

Namesto stopnje detekcije in identifikacije se uporablja tudi stopnja napačne zavrnitve (angl. *False Rejection Rate, FRR*) P_{FR}, ki določa da je sistem zavrnil znan obraz (velja namreč $P_{DI} = 1 - P_{FR}$).

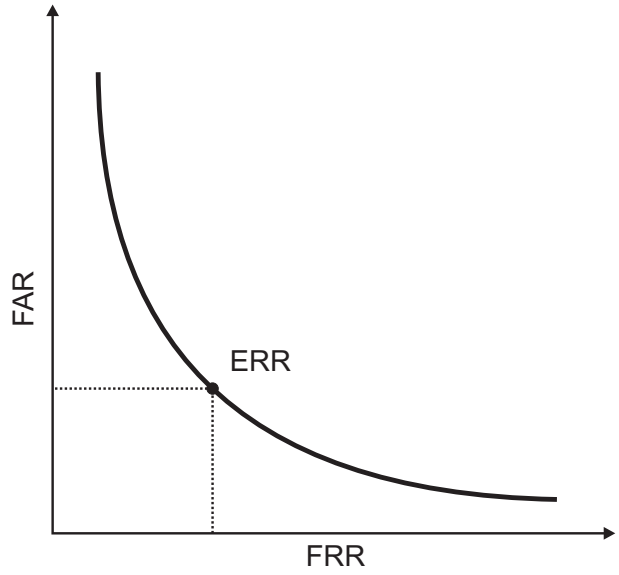
Drugi kazalec uspešnosti je stopnja lažnega alarma, ki mu pravimo tudi stopnja napačne potrditve (angl. *False Acceptance Rate*, FAR). Zgodi se v primeru, ko testnemu vzorcu ne pripada nobena oseba iz galerije ($p_j \in P_N$), sistem pa ga vseeno pripiše osebi iz galerije G. Takšen testni vzorec imenujemo tudi "vsiljivec" (angl. *imposter*). Lažni alarm se sproži v primeru, ko je najvišja ocena ujemanja za testni vzorec nad določeno mejno vrednostjo:

$$\max_i s_{ij} \geq \tau \ .$$

Stopnjo lažnega alarma določajo tisti testni vzorci p_j iz P_N pri katerih se sproži lažni alarm. To lahko formalno zapišemo kot:

$$P_{FA}(\tau) = \frac{|\{p_j : \max_i s_{ij} \geq \tau\}|}{|P_N|} \ . \tag{4.2}$$

Idealni sistem za prepoznavanje obrazov mora tako imeti stopnjo detekcije in identifikacije enako 1.0 in stopnjo lažnega alarma enako 0. To pomeni, da so bile vse osebe iz galerije pravilno najdene in prepoznane ter da se ni nikoli sprožil lažni alarm. Pri sistemih, ki delujejo v resničnem okolju pa gre vedno za neko razmerje med stopnjo detekcije in identifikacije ter stopnjo lažnega alarma. S spreminjanjem mejne vednosti lahko to razmerje spreminjamo. S povečevanjem mejne vrednosti obe stopnji zmanjšamo. Hkrati pa ne moremo uspešnosti obeh stopenj povečati, ampak gre vedno za obratno sorazmerje med njima. To razmerje lahko prikažemo s pomočjo krivulje ROC (angl. *Receiver Operator Characteristic*). Običajno pa s krivuljo ROC prikažemo razmerje obeh napak: stopnjo napačne zavrnitve – FRR in stopnjo napačne potrditve – FAR (slika 4.1).



Slika 4.1: Primer krivulje ROC za primer identifikacije na odprti množici.

Če je mejna vrednost zelo nizka, potem je vrednost napake FAR zelo visoka, vrednost napake FRR pa nizka. V tem primeru se nahajamo v skrajnem levem zgornjem kotu krivulje ROC. Če mejno vrednost povečujemo se napaka FAR zmanjšuje, povečuje pa se napaka FRR. V skrajnem desnem kotu je vrednost napake FRR največja, vrednost napake FAR pa blizu vrednosti 0. Točka na krivulji pri kateri sta

vednosti napak FRR in FAR enaki imenujemo stopnja enake napake (angl. *Equal Error Rate*, EER). Pozorni moramo biti tudi na to, da poleg stopenj uspešnosti podamo tudi velikosti obeh testnih množic in velikost galerije.

V splošnem primeru identifikacije na odprti množici sistem pregleda n najboljših ujemanj z vzorci galerije. Tako govorimo o uspešni detekciji in identifikaciji testnega vzorca pri rangu n, če se testni vzorec nahaja na n-tem ali nižjem mestu in je ocena ujemanja nad določeno mejo. Pogoja lahko formalno zapišemo kot:

- rank(p_j)$\leq$ n in
- $s_{*j} \geq \tau$ za oceno ujemanja, kjer je id(p_j)=id(g^*) .

Stopnja detekcije in identifikacije pri rangu n je tisti del vzorcev iz testne množice P_G, ki so pravilno najdeni in prepoznani v območju n. Stopnjo detekcije in identifikacije pri rangu n in mejni vrednosti τ lahko zapišemo kot:

$$P_{DI}(\tau, n) = \frac{|\{p_j : \text{rank}(p_j) \leq n \text{ in } s_{*j} \geq \tau\}|}{|P_G|} \quad . \tag{4.3}$$

Izračun lažnega alarma pri rangu n pa je enak kot za rang 1 (enačba 4.2).

Uspešnost splošnega sistema za prepoznavanje obrazov lahko tako predstavimo s krivuljo v trirazsežnem prostoru, kjer posamezne osi predstavljajo stopnjo detekcije in identifikacije, stopnjo lažnega alarma in rang. Običajno pa uspešnost predstavimo v dvodimenzionalnem prostoru pri konstantnem rangu ali pri konstantni stopnji lažnega alarma. Pri predstavitvi stopnje uspešnosti pri določenem rangu (enačba 4.3) smo predpostavili, da so vse ocene podobnosti med seboj različne. Posebno pozornost moramo nameniti primerom, če so ocene podobnosti med seboj tudi enake. Takšne ocene podobnosti, ki so med seboj enake imenujemo tudi vezane podobnosti (angl. *tied similarity scores*). Takšne primere vezanih podobnosti lahko rešimo na tri različne načine, s pomočjo optimističnega, pesimističnega ali povprečnega ranga. Optimistični rang je število ocen podobnosti, ki so striktno večje (>) od s_{*j} povečano za 1. V tem primeru prepišemo testnemu primeru najboljši možni rang. Pesimistični rang pa določa število ocen podobnosti, ki so večje ali enake ($\geq$) od s_{*j}. V tem primeru prepišemo testnemu primeru najslabši možni rang. Pri povprečnem rangu pa izračunamo povprečje med optimističnim in pesimističnim rangom. Pri preizkusu FRVT 2002 je bil uporabljen povprečni rang. Pri uporabi optimističnega ranga lahko dobimo nepričakovane rezultate. Če na primer, vsebuje matrika podobnosti samo enake vrednosti dobimo pri optimističnemu rangu za stopnjo identifikacije rezultat 100%.

4.2.2 Verifikacija

Pri postopku verifikacije oseba predloži svoj biometrični vzorec sistemu in trdi, da je ena izmed oseb v galeriji. Na biometrični vzorec osebe lahko gledamo kot na

testni vzorec. Sistem nato primerja testni biometrični vzorec s shranjenim vzorcem osebe v galeriji. Rezultat primerjave je ocena podobnosti. Sistem odobri zahtevi osebe, če je ocena podobnosti večja od sistemsko določene mejne vrednosti (mejna vrednost je določena s strani programa in je lahko za različne programe različna). V nasprotnem primeru pa sistem osebo zavrne.

Obstajata dva postopka za izračun stopnje pravilne verifikacije. Prva je metoda *round-robin*. Pri postopku metode *round-robin* sta obe testni množici P_N in P_G enaki in ju lahko označimo kot testno množico P. Najprej se izračunajo vse ocene podobnosti med testno množico in galerijo. Vse tako dobljene ocene ujemanja se nato upoštevajo za izračun stopnje pravilne verifikacije (angl. *verification rate*) in vse ocene neujemanja za izračun stopnje napačne verifikacije ali potrditve (angl. *False Acceptance Rate*, FAR). Stopnjo pravilne verifikacije P_V po round-robinovi metodi izračunamo kot:

$$P_V(\tau) = \frac{|\{p_j : s_{ij} \geq \tau,\ \mathrm{id}(g_i) = \mathrm{id}(p_j)\}|}{|P|} \ , \tag{4.4}$$

stopnjo napačne potrditve pa kot:

$$P_{FA}(\tau) = \frac{|\{p_j : s_{ij} \geq \tau,\ \mathrm{id}(g_i) \neq \mathrm{id}(p_j)\}|}{(|P| - 1)|G|} \ . \tag{4.5}$$

Pomanjkljivost metode *round-robin* je, da uporablja za izračun stopnje pravilne in napačne verifikacije isto testno množico, kar ne posnema resnične situacije, ko verifikacijo zahtevajo osebe, ki niso v galeriji. To pomanjkljivost odpravi drugi postopek, ki stopnjo pravilne in napačne verifikacije določi s pomočjo dveh testnih množic P_G in P_N. Stopnja pravilne verifikacije se izračuna s pomočjo ocen ujemanja med vzorci galerije G in množice P_G. Število vseh ocen ujemanja je velikost testne množice P_G. Stopnjo lažnega alarma pa se izračuna izmed vseh ocen neujemanja med vzorci galerije in P_N. Ocene neujemanja veljajo za resnične vsiljivce, ker oseb iz testne množice P_N ni v galeriji. Zaradi tega se temu postopku reče tudi postopek resničnih vsiljivcev (angl. *true imposter protocol*). Število vseh ocen neujemanj je v tem primeru enako $|P_N||G|$. Stopnjo pravilne verifikacije po postopku resničnih vsiljivcev izračunamo kot:

$$P_V(\tau) = \frac{|\{p_j : s_{ij} \geq \tau,\ \mathrm{id}(g_i) = \mathrm{id}(p_j)\}|}{|P_G|} \tag{4.6}$$

in stopnjo napačne potrditve kot:

$$P_{FA}(\tau) = \frac{|\{p_j : s_{ij} \geq \tau\}|}{|P_N||G|} \ . \tag{4.7}$$

4.2.3 Identifikacija na zaprti množici

Kazalci uspešnosti pri identifikaciji na zaprti množici so klasični kazalci za merjenje uspešnosti prepoznavanja obrazov. Pri identifikaciji na zaprti množici se ne vprašamo samo, ali je najboljše ujemanje pravilno, ampak nas zanima tudi, ali je pravilno prepoznana oseba znotraj območja n (rang n) najboljših ujemanj. Za izračun uspešnosti moramo najprej urediti ocene podobnosti med testnim vzorcem p_j in galerijo G ter izračunati rang(p_j). Stopnja identifikacije (angl. *identification rate*) $P_I(n)$ pri rangu n so tisti testni vzorci, katerih rang je n ali nižji. Definirajmo seštevek testnih vzorcev ranga n ali manj kot:

$$C(n) = |\{p_j : \mathrm{rank}(p_j) \leq n\}| \ . \tag{4.8}$$

Stopnjo identifikacije pri rangu n pa izračunajmo kot:

$$P_I(n) = \frac{C(n)}{|P_G N|} \ . \tag{4.9}$$

Funkciji $C(n)$ in $P_I(n)$ s povečevanjem ranga n naraščata. Stopnjo identifikacije pri rangu 1, $P_I(1)$ imenujemo tudi stopnja pravilne identifikacije ali stopnja ali ocena najboljšega ujemanja. Uspešnost identifikacije na zaprti množici običajno prikažemo s pomočjo grafa CMC (angl. *Cumulative Match Characteristic*), kjer narišemo $P_I(n)$ kot funkcijo ranga n. Običajno pa uspešnost prikažemo samo pri rangu 1. Če pa uporabimo različne range so običajne vrednosti ranga, pri katerih prikažemo uspešnost 5, 10 in 20. Slabost takšnega kazalca uspešnosti je njena odvisnost od velikosti galerije $|G|$. Zaradi tega uspešnost identifikacije večkrat prikažemo pri rangu 1 v odvisnosti od velikosti galerije. Če se hočemo izogniti odvisnosti pa prikažemo uspešnost identifikacije kot odstotek ranga (na primer, za pravilno identifikacijo označimo tiste primere, katerih ujemanje je znotraj prvih 10% vseh primerov).

Identifikacija na zaprti množici je poseben primer identifikacije na odprti množici kjer je testna množica P_N prazna in sistemska mejna vrednost $\tau = -\infty$. Pri mejni vrednosti $\tau = -\infty$ imamo stopnjo lažnega alarma enako 1 (FAR=1). To pomeni, da velja $s_{*j} \geq \tau$ za vse ocene ujemanja in vsi primeri sprožijo alarm. Torej velja za vsak n: $P_{DI}(-\infty, n) = P_I(n)$.

4.3 Preizkusna protokola FERET in FRVT 2002

Preizkusna protokola FERET in FRVT 2002 [77, 78] veljata za nekakšen standard na področju preizkušanja uspešnosti metod za prepoznavanje obrazov oziroma na splošno za preizkušanje metod, ki temeljijo na biometričnih podatkih. Protokol FRVT 2002 temelji na protokolu, ki je bil predstavljen septembra 1996 v okviru preizkusa FERET s to razliko, da je zasnovan za preizkus splošnih biometričnih

podatkov in ne samo biometričnih podatkov obraza. V protokolu FRVT 2002 je bil dodan še postopek za normalizacijo ocen podobnosti ter specifikacija XML datoteke, ki določa splošni format zapisa rezultatov za katerekoli biometrične podatke.

Pri preizkusih FRVT 2002, 2000 in FERET veljajo sledeča navodila:

1. Preizkuse so pripravljale in izvajale skupine ljudi, ki so bile neodvisne od razvijalcev algoritmov ali podjetji, ki so sodelovala na preizkusih.

2. Podatke tesnih množic so predhodno ločili in niso bili predhodno na vpogled sodelujočim pri preizkusih.

3. Predhodno so objavili ureditev preizkusa, preizkusne postopke (protokole) in metodologijo.

4. Rezultate preizkušenih metod so uredili tako, da so se pokazale razlike med udeleženci.

Prvi dve navedeni točki zagotavljata nepristranskost preizkusa. Druga točka zagotavlja tudi to, da niso bile metode predhodno prilagojene testni množici ampak preizkus res odraža uspešnost metode na novi neznani testni množici. Za razumevanje in interpretacijo rezultatov moramo zelo dobro poznati pogoje pod katerimi so bile metode preizkušene. Ti pogoji so podrobno opisani v opisih ureditve preizkusa, njegovih postopkih in metodologiji, ki je bila uporabljena. Protokol preizkusa določa tako postopke samega preizkusa kot tudi postopke predstavitve rezultatov. Nujno je, da so vsi postopki preizkusov predhodno znani, da lahko kasneje razumemo predstavljene rezultate. Točka 4 pa poskuša prilagoditi težavnost preizkusa tako, da bodo rezultati med posameznimi metodami ter med preizkusi znotraj posamezne metode dovolj reprezentativni. Če je na primer problem prelahek in so rezultati vseh preizkusov zelo dobri, je zelo težko določiti katera metoda je najboljša. V primeru, da bi bili rezultati zelo slabi pa sploh ne bi mogli določiti kakšne probleme smo z metodami rešili.

Pri preizkusih FERET in FRVT so načrtovalci protokolov ustrezno težavnost dosegli s pomočjo skrbno izbranih testnih množic. Težavnost so spreminjali z izbiro različno velikih podatkovnih baz, z vključevanjem slik, ki so bile zajete v različnih časih in z uporabo slik, zajetih tako v prostoru kot na prostem. To je imelo za posledico spremenljive pogoje osvetlitve, velikosti in ozadja.

Vhod v algoritem ali preizkusni sistem, sta dve množici biometričnih vzorcev: ciljna množica T (angl. *target set*) in poizvedovalna množica Q (angl. *query set*). Množica galerije G je sestavljena iz ciljne množice, testne množice pa iz poizvedovalne množice. Izhod iz algoritma so mere podobnosti s_{ij} med vsemi pari slik t_i iz ciljne množice in slik q_j iz poizvedovalne množice. Mera podobnosti nam pove koliko sta si dva obraza med seboj podobna. Tako dobljene mere podobnosti med vsemi pari tvorijo tako imenovano matriko podobnosti (angl. *similarity matrix*), iz katere se lahko izračuna statistike uspešnosti metode. Prvo pravilo preizkusnega postopka

FERET in FRVT 2002 je, da se mora izračunati celotna matrika podobnosti. To zagotavlja izračun posameznih statistik uspešnosti za vse algoritme.

Da lahko izračunamo uspešnost za različne galerije in različne tesne množice moramo imeti v ciljni in poizvedovalni množici več biometričnih vzorcev iste osebe. To vodi k drugemu pravilu preizkusnega postopka: vsak biometrični vzorec v ciljni ali poizvedovalni množici mora biti edinstven. To pomeni, da se isti vzorec ne pojavlja v različnih množicah.

Tretje pravilo pa določa, da se učenje izvede pred samim preizkusom metode. To pravilo prisili metode, da imajo neko splošno predstavo obrazov, ne pa predstavo, ki je prilagojena galeriji slik. V primeru, da bi učenje izvajali na galeriji tudi ne bi bilo mogoče preizkusiti različnih množic galerij samo z enim samim poskusom. V postopku FRVT 2002 je dovoljena normalizacija ocen podobnosti in s tem je do neke mere omogočeno prilagajanje vzorcem galerije.

S pomočjo ciljne in poizvedovalne množice lahko izvajamo preizkuse na različnih kategorijah biometričnih vzorcev. Pri prepoznavanju obrazov bi lahko izbrali sledeče kategorije testnih primerov:

1. slike iz galerije in testne množice zajete na isti dan,
2. testne slike zajete s tedenskim zamikom od slik iz galerije,
3. slike, med katerimi je časovni razpon več kot leto dni,
4. testne slike zajete na prostem, slike galerije v prostoru,
5. slike, ki se razlikujejo v osvetlitvi,
6. slike, ki se razlikujejo v obrazni mimiki.

S primerno razporeditvijo množic lahko preizkusimo tudi vpliv starosti in spola na uspešnost prepoznavanja. Protokol preizkusov FERET in FRVT 2002 je zelo prilagodljiv in omogoča, da izračunamo različne statistike. Tako lahko preizkusimo vpliv velikosti galerije na uspešnost. Če hočemo ugotoviti kako se uspešnost spreminja zaradi različnega okolja, lahko metodo preizkusimo s primerno izbranimi galerijami.

4.4 Preizkusi FERET in FRVT

Pred preizkusi FERET ni obstajal preizkusni postopek, ki bi vseboval veliko podatkovno bazo obrazov in standardizirane postopke. Starejši preizkusi niso pokazali na resnično stanje na področju metod za prepoznavanje obrazov. Nekateri sistemi za prepoznavanje obrazov so namreč obljubljali idealne rezultate, ker so bili preizkušeni na ozko usmerjenih majhnih bazah podatkov.

Prvi preizkus FERET, ki je bil izveden avgusta 1994 [79], je postavil temelje za preizkuse vseh algoritmov za prepoznavanje obrazov. Preizkusil je uspešnost algoritmov pri avtomatskem iskanju, normalizaciji in prepoznavanju obrazov. Vseboval je tri različne preizkuse: prvi preizkus je meril uspešnost identifikacije pri galeriji velikosti 316 oseb, pri čemer je vsaki osebi pripadala ena slika, z drugim preizkusom so preizkusili lažni alarm algoritmov in tretji preizkus je meril vpliv položaja obraza na uspešnost prepoznave.

Drugi preizkus FERET je bil organiziran marca 1995 in je vseboval samo en preizkus. Ta preizkus je meril uspešnost identifikacije pri galeriji 817 oseb in pri 463 osebah, ki so bile posnete v časovnem zamiku. Tretji in zadnji preizkus FERET, ki so ga izvajali septembra 1996 in marca 1997, pa bomo podrobneje predstavili v nadaljevanju. Na koncu bomo predstavili še preizkusa FRVT iz leta 2000 in leta 2002, na katerih so preizkusili uspešnost komercialnih izdelkov.

4.4.1 Preizkusi FERET

Za zadnji preizkus FERET je bila uporabljena podatkovna baza FERET, ki so jo kasneje leta 2000 tudi javno objavili in s tem omogočili tudi drugim raziskovalcem, da so preizkusili svoje metode in primerjali svoje rezultate s tistimi, ki so bili dobljeni na preizkusu FERET. Opis podatkovne baze FERET najdete v poglavju 3.2.1.

Na septembrskem preizkusu FERET leta 1996 je galerija vsebovala 1196 slik posameznikov, katerim je pripadala ena frontalna slika (fa). Množica slik galerije je temeljna množica, ki služi za primerjanje slik iz štirih testnih množic:

- testna množica fb – slike galerije in testne slike so bile zajete na isti dan, pod enako osvetlitvijo (1195 slik)
- testna množica fc – slike galerije in testne slike so bile zajete na isti dan, pod različno osvetlitvijo (194 slik)
- testna množica $dup1$ – slike galerije in testne slike so bile zajete na različne dneve (722 slik)
- testna množica $dup2$ – slike galerije in testne slike so bile zajete več kot leto narazen (234 slik)

Preizkušenih je bilo 10 različnih algoritmov. Uspešnost je bila izračunana za nalogo identifikacije na zaprti množici in verifikacije. Najbolje so se odrezali trije algoritmi: verjetnostni PCA iz laboratorija MIT [67], metoda, ki je bazirala na algoritmu LDA z Univerze Maryland [112, 114] in algoritem na osnovi ujemanja elastičnih grafov (EGM) z Univerze Southern California (USC) [74].

S pomočjo preizkusa so ugotovili, da je uspešnost zelo odvisna od testne množice. Pokazala se je tudi velika razlika med uspešnostjo najboljšega algoritma in povprečno uspešnostjo vseh algoritmov. Uspešnost algoritmov se je razlikovala tudi glede naloge: ali so morali ugotoviti identiteto neznane osebe (identifikacija) ali jo samo

algoritem	fb	fc	dup1	dup2
verjet. PCA iz MIT	94.8%	32.0%	57.6%	34.2%
LDA iz UMD	96.2%	58.8%	47.2%	20.9%
EGM iz USC	95.0%	82.0%	59.1%	52.1%

Tabela 4.1: Rezultati uspešnosti treh najboljših algoritmov na preizkusu FERET septembra 1996 in marca 1997.

preveriti (verifikacija). Tabela 4.1 prikazuje rezultate preizkusa identifikacije treh najuspešnejših algoritmov za vse štiri testne množice.

Iz rezultatov lahko vidimo, da predstavljata največji izziv testni množici *dup1* in *dup2*, ki vsebujeta slike obrazov zajete v drugem časovnem obdobju kot slike iz galerije. Vidimo lahko tudi, da različne osvetlitve (testna baza fc) najslabše vplivajo na algoritem PCA, najbolje pri tej nalogi pa se je izkazal algoritem EGM. Poleg časovnih sprememb in sprememb v osvetlitvi so na preizkusu FERET ugotovili tudi veliko težav pri prepoznavanju obrazov slikanih iz različnih pogledov.

4.4.2 Preizkus FRVT 2000

V septembrskem preizkusu FERET so bili v glavnem preizkušeni prototipi sistemov za prepoznavanje obrazov. Po zadnjem preizkusu marca 1997 pa se je napredek in razvoj sistemov za prepoznavanje obrazov zelo povečal, tako da so se leta 2000 na tržišču že pojavili prvi komercialni sistemi (za pregled komercialnih sistemov glej poglavje 10).

Preizkus FRVT 2000 [7] je bil v glavnem namenjen preizkusu uspešnosti trenutno najuspešnejših komercialnih sistemov, ki so se v tistem času pojavili na tržišču. Organizatorji so za preizkus uporabili enak postopek kot pri zadnjem preizkusu FERET. Na preizkusu je sodelovalo pet različnih komercialnih produktov podjetji: Banque-Tec International iz Avstralije, C-VIS Computer Vision und Automation GmbH iz Nemčije ter podjetja Miros, Lau technologies in Visionics Corporation iz Amerike.

Pri preizkusu FRVT 2000 je bilo uporabljenih veliko več različnih kategorij testnih množic kot pri prejšnjih preizkusih. Preizkusili so uspešnost osmih različnih kategorij: različno stisnjeni podatki, različna oddaljenost od osebe, različni izrazi na obrazih, različni medijski nosilci, položaj, velikost in različen čas zajemanja. Posamezna kategorija ni imela skupne galerije slik, ampak se je velikost galerije in testnih množic za posamezno kategorijo spreminjala.

Rezultati na slikah, ki so bile zajete na dva različna medijska nosilca (navadni film in digitalni medij) so pokazali, da ni razlike v uspešnosti. Tudi pri slikah, ki so bile stisnjene do razmerja 40:1 se uspešnost ni poslabšala. Za stiskanje je bil uporabljen algoritem JPEG. Na preizkusu so preverili tudi uspešnost prepoznave za slike zajete z različnih pogledov. Izkazalo se je, da uspešnost ne pade drastično do zornega kota 25°, pri obratu, ki je večji od 40° pa se uspešnost močno zmanjša. Za različno

osvetlitev sta bila izvedena dva preizkusa. Za prvi preizkus so bile slike zajete v prostoru, kar sovpada s testno množico fc pri preizkusih FERET. Najboljšemu sistemu se v tej kategoriji uspešnost ni opazno spremenila. V drugemu preizkusu so preizkusili uspešnost nad slikami, ki so bile zajete na prostem. Rezultati so pokazali, da se uspešnost v tem primeru močno zmanjša. Najboljši sistem je dosegel v tej kategoriji samo 55% uspešnost. Preizkus časovne zakasnitve je bil enak kot pri prejšnjih preizkusih FERET. Pokazal se je napredek v primerjavi s preizkusom FERET. Na množici *dup1* je bila uspešnost na preizkusu FRVT 2000 63% medtem, ko je bila uspešnost na preizkusu FERET 58%. Na množici *dup2* pa je bila uspešnost 64%, na preizkusu FERET pa 52%.

4.4.3 Preizkus FRVT 2002

Glavni cilj preizkusa FRVT 2002 je bil ugotoviti, kako primerni so trenutni sistemi za prepoznavanje obrazov v realnih okoljih. Na preizkus se je prijavilo 10 podjetji: AcSys Biometrics Corp., Cognitec Systems GmbH, C-VIS Computer Vision und Automation GmbH, Dream Mirh, Eyematics Interfaces, Iconquest, Identix, Imagis Technologies, Viisage Technology in VisionSphere Technologies. Preizkus je potekal v dveh delih. Pri prvem preizkusu HCInt (angl. *high computational intesity test*) so uspešnost preizkusili na ogromni množici obrazov, ki je vsebovala 121.589 slik, 37.437 različnih oseb. Preizkuse so izvedli za naloge verifikacije, identifikacije na zaprti množici in identifikacije na odprti množici. Drugi preizkus MCint (angl. *medium computational intensity test*) pa je vseboval manjšo podatkovno bazo slik različnih kategorij in video posnetke. Slike so bile posnete na različne načine: v stilu frontalna slika/slika s profila (angl. *mugshot style*), z različnih pogledov, v prostoru in na prostem. Uporabljene pa so bile tudi modelirane nefrontalne slike.

Rezultati preizkusa FRVT 2002 na množici HCInt so pokazali, da zmerna sprememba osvetlitve prostora najboljšim sistemom ne povzroča težav. Skoraj enake rezultate so dobili pri preizkusu dveh testnih množic različnih osvetlitev. Najboljša sistema Cognitec in Identix sta dosegla stopnjo verifikacije 90% pri 1% stopnji napačne potrditve. Ostali sistemi so dosegli sledeče uspešnosti: 87% sistem podjetja Eyematic, 65% C-VIS, 64% Viisage, 59% Imagis, 53% VisionSphare in 34% Dream Mirih. Preostali trije pri preizkusih HCint niso sodelovali.

Z zmanjševanjem napake pa se je zmanjševala tudi uspešnost, tako so najboljši sistemi dosegali pri napaki 0.1% uspešnost 82% in pri napaki 0.01% samo še 70% uspešnost. Lahko pa opazimo napredek v primerjavi s preizkusom FRVT iz leta 2000. Pri enaki uspešnosti se je napaka zmanjšala za 50%.

Sisteme so preizkusili tudi na spremenljivih osvetlitvah na prostem, kjer je najboljši sistem dosegel uspešnost 50% pri napaki 1%, kar dokazuje, da ta kategorija ostaja še vedno izziv raziskovalcem.

Pomembno vprašanje pri sistemih, ki delujejo v realnih okoljih je tudi, za koliko se uspešnost zmanjša, če povečujemo čas med zajetjem slike iz galerije in zajetjem nove slike, ki jo mora sistem prepoznati. Na preizkusu FRVT 2002 so ugotovili, da

uspešnost s povečanjem časa med zajetimi slikami pada in sicer za vsako dodatno leto se uspešnost poslabša za približno 5%.

Na preizkusu FRVT 2002 so poskušali odgovoriti tudi na drugo pomembno vprašanje, kako se uspešnost spreminja s spreminjanjem velikosti podatkovne baze oziroma s spreminjanjem velikosti seznama. Najboljši sistem je dosegel uspešnost 85% na podatkovni bazi 800 ljudi, uspešnost 83% na bazi 1600 ljudi in 73% na bazi 37.437 ljudi. Glede na rezultate je bilo ugotovljeno, da se pri vsaki podvojitvi baze uspešnost zmanjša za dva ali tri odstotke. Ali bolj splošno povedano, uspešnost identifikacije na zaprti množici pada linearno z logaritmov velikosti baze.

Preizkusili so tudi uspešnost pri spremenljivi velikosti seznama oseb za nalogo identifikacije na odprti množici. Pri stopnji napačnega alarma 1% je bila uspešnost najboljšega sistema 77%, če je bilo v seznamu 25 oseb, pri 300 ljudeh je uspešnost padla na 69% in pri 3000 na 56%. Izkazalo se je, da je uspešnost nad seznamom oseb velikosti od 25 do 50 ljudi veliko boljša kot pri velikem seznamu ljudi.

Na preizkusu so prvič preizkusili, kako vplivajo demografske spremembe na uspešnost prepoznave. Ugotovljeno je bilo, da je uspešnost prepoznave moških za 6% do 9% boljša kot prepoznava žensk. Najboljši sistem je dosegel stopnjo identifikacije na podatkovni bazi moških 78%, pri podatkovni bazi žensk pa 69%. Izkazalo se je tudi, da sistemi veliko lažje prepoznajo starejše osebe kot mlajše. Pri osebah starosti od 18 do 22 let so najboljši sistemi dosegli uspešnost 62%, pri osebah med 38 in 24 let pa 74%. Na vsakih 10 let se uspešnost poveča za 5%, preizkušeno do 63 leta.

Preizkusili so tudi dve novi tehniki. Prva je bila uporaba 3D modeliranja za prepoznavo obrazov z različnih pogledov. Uspešnost sistemov nad obrazi, ki niso frontalni, je še vedno nizka. Najboljši sistemi so pri obrazih, ki so bili obrnjeni do 45 stopinj levo in desno dosegli uspešnost 42%, pri obrazih obrnjenih za 45 stopinj gor in dol pa 53%. Z uporabo 3D modelov avtorjev Blanza in Vetterja [8] pa so uspešnost izboljšali na 80% oziroma 85%.

Drugo, kar so preizkusili je bila uporaba tehnik prepoznavanja obrazov na video posnetkih. Ugotovljeno je bilo, da je uspešnost prepoznavanja na video posnetkih primerljiva s tisto, ki jo dosegajo sistemi na statičnih slikah.

4.5 Povzetek poglavja

Periodični preizkusi FERET in FRVT so pripomogli k pospešeni rasti sistemov za prepoznavanje obrazov. Zagotovili so standarde na področju preizkusov in vrednotenju rezultatov. S širjenjem področja prepoznave obrazov so se tudi razvijali vedno novi preizkusi, ki so poskušali odgovoriti na aktualna vprašanja.

Preizkusi FERET so morali kot prvi preizkusi na tem področju zagotoviti ustrezno težavne testne množice. Izkazalo se je, da največ težav povzroča kategorija, kjer so testne slike zajete v drugem času kot slike iz galerije.

Na preizkusu FRVT 2000 se je izkazalo, da medijski nosilec in nivo stiskanja ne

vplivata na uspešnost. Veliko težav pa še vedno povzročajo spremenljiva osvetlitev na prostem, različni pogledi in časovna razlika med zajemom slik.

Iz pregleda preizkusa FRVT 2002 smo lahko ugotovili, da je uspešnost verifikacije najboljših sistemov 90% pri 1% napaki in na to uspešnost ne vplivajo zmerne spremembe v osvetlitvi znotraj prostora. Ostaja pa še nerešen problem prepoznave obrazov na prostem. 3D modeli so se izkazali za zelo učinkovite pri izboljšanju prepoznave pri obrazih z različnih pogledov. Ugotovljeno je bilo tudi, da uspešnost identifikacije pada linearno z logaritmom velikosti podatkovne baze obrazov, ter da demografske spremembe kot so spol in starost tudi vplivajo na uspešnost sistemov.

Ko so se začele pojavljati težnje po avtomatiziranem sistemu za prepoznavanje obrazov, je bil njegov glavni cilj prepoznati neznan obraz. Z razvojem vedno naprednejših sistemov pa se je temu cilju pridružil še cilj razumevanja takšnih sistemov in postopkov za prepoznavo. Če hočemo boljše razumeti same lastnosti algoritmov, moramo poleg neodvisnih preizkusov v prihodnje izvesti še veliko dodatnih računskih eksperimentov.

Poglavje 5

Normalizacija slik obrazov

5.1 Uvod

Prvi korak pred postopkom učenja oziroma izločanja značilk je normaliziranje posamezne slike obraza. Zagotoviti namreč moramo, da so vse slike, ki vstopijo v sistem, enako urejene. Postopek normalizacije moramo izvesti na vseh slikah: najprej na učnih slikah, nato pa na slikah, ki vstopajo v sistem v fazi prepoznavanja.

Glavna naloga normalizacije je, da uredimo različno velike, obrnjene in nagnjene obraze. Vhodne slike obrazov moramo ustrezno pomanjšati, odrezati in obrniti, da so vse slike s katerimi operiramo v sistemu za prepoznavanje obrazov enake velikosti ter da imajo obrazne značilke kar se da enako lokacijo na sliki.

Pri normalizaciji sliko obraza tudi obrežemo. Običajno obrežemo iz slike tisti del, ki predstavlja ozadje ali pa nam nepomembne in moteče elemente, ki za samo prepoznavo niso pomembni. Takšni deli so običajno pokrivala ali deli obleke. S tem, ko sliko obrežemo, zmanjšamo tudi informacijo in povečamo hitrost ter največkrat tudi točnost samega postopka prepoznave obrazov.

Zaradi ovalne oblike obraza pa slike ne moremo natančno obrezat tako, da bi ohranili samo obraz. V tem primeru uporabimo različne ovalne maske, ki navadno s črno barvo prekrijejo ozadje obraza, ki ni pomembno.

Naloga normalizacije je tudi odprava različne osvetlitve. Za odpravo osvetlitve se lahko uporabljajo različni postopki. Razvite so bile prav posebne metode, ki vključujejo model osvetlitve na podlagi katerega se izvrši normalizacija. Največkrat pa se pri metodah za prepoznavo obrazov uporabljajo preprosti postopki, ki niso računsko zahtevni.

Točnost normalizacijskega postopka je za samo uspešnost prepoznave zelo pomembna in si ga bomo zato v nadaljevanju podrobneje ogledali. Predstavili bomo osnovni postopek normalizacije, ki vključuje poravnavo slik z ustrezno rotacijo, zmanjševanjem in rezanjem. Predstavili bomo tudi dva normalizacijskega postopka na nivoju slikovnih elementov, s katerima zmanjšajo vpliv različne osvetlitve. Predlagali bomo tudi postopek uporabe maske, ki zakrije del ozadja. V poglavju 9 pa bomo poskušali ugotoviti, kako različni normalizacijski postopki vplivajo na samo

uspešnost prepoznave.

5.2 Slike obrazov

Slike iz podatkovne baze FERET [77] so sivinske in velikosti 256×384 slikovnih elementov. Obrazi na slikah so različnih velikosti in različno obrnjeni (slika 5.1).

(a) (b) (c)

Slika 5.1: Trije primeri nenormaliziranih slik obrazov iz podatkovne baze FERET: (a) manjši obraz, (b) večji obraz in (c) obrnjen obraz.

Slike iz podatkovne baze AR so barvne, s 24 bitno barvno globino velikosti 768×576 slikovnih elementov. Čeprav na prvi pogled slike obrazov izgledajo poravnane, jih je potrebno normalizirati, saj se obrazi malenkostno razlikujejo v velikosti, pa tudi v nagibu (slika 5.2).

(a) (b) (c)

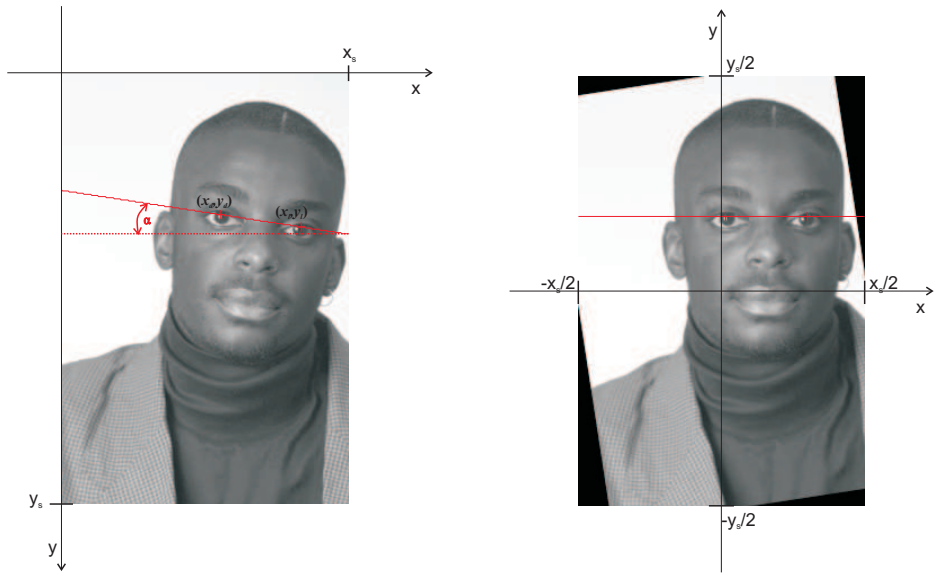
Slika 5.2: Trije primeri nenormaliziranih slik obrazov iz podatkovne baze AR: (a) manjši obraz, (b) večji obraz in (c) nagnjen obraz.

5.3 Rotacija slike obraza

Prvi korak normalizacije je rotacija slike glede na pozicijo oči. To pomeni, da horizontalno poravnamo levo in desno oko. Poleg vsake slike imamo zapisane koordinate x in y posameznega očesa. Za desno oko osebe x_d in y_d ter za levo oko osebe x_l in y_l. Sliko rotiramo za kot α, ki ga izračunamo na podlagi očesnih koordinat:

$$\alpha = \arctan \frac{v_{raz}}{h_{raz}} \quad , \tag{5.1}$$

kjer $v_{raz} = y_l - y_d$ določa vertikalno razliko med očesoma in $h_{raz} = x_l - x_d$ horizontalno razliko med očesoma. V primeru, ko je kot α pozitiven, rotiramo sliko v nasprotni smeri urinega kazalca, sicer pa v smeri urinega kazalca. Kot α je pozitiven samo v primeru, ko je vrednost $y_l - y_d > 0$ pozitivna (to je, če je desno oko nad levim očesom). Na podlagi tako izračunanega kota izvedemo rotacijo slike in dobimo horizontalno poravnan obraz glede na oči (slika 5.3).



(a) (b)

Slika 5.3: Primer rotacije slike obraza na podlagi koordinat oči: (a) originalna slika obraza z označenimi očmi in kotom nagiba α in (b) rotirana slika obraza.

Na podlagi tako izračunanega kota α lahko določimo rotacijsko matriko $\mathbf{R}$:

$$\mathbf{R} = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{bmatrix} \quad , \tag{5.2}$$

s pomočjo katere nato izračunamo nove koordinate oči na tako poravnani sliki:

$$\begin{bmatrix} x'_{nl} \\ y'_{nl} \end{bmatrix} = \mathbf{R} \begin{bmatrix} x_l - \frac{x_s}{2} \\ \frac{y_s}{2} - y_l \end{bmatrix}, \qquad \begin{bmatrix} x'_{nd} \\ y'_{nd} \end{bmatrix} = \mathbf{R} \begin{bmatrix} x_d - \frac{x_s}{2} \\ \frac{y_s}{2} - y_d \end{bmatrix} \quad , \tag{5.3}$$

kjer x_s in y_s označujeta širino oziroma višino slike. S tem, ko odštejemo polovično širino oziroma odštejemo y koordinato očesa od polovične višine, premaknemo koordinatno izhodišče iz zgornjega levega kota na sredino slike, ki predstavlja tudi središče rotacije (slika 5.3). Da dobimo prave koordinate levega in desnega očesa, moramo na tako rotirani sliki polovično širino zopet prišteti oziroma koordinato y očesa odšteti od polovične višine:

$$x_{nl} = x'_{nl} + \frac{x_s}{2}, \qquad x_{nd} = x'_{nd} + \frac{x_s}{2}, \tag{5.4}$$

$$y_{nl} = \frac{y_s}{2} - y'_{nl}, \qquad y_{nd} = \frac{y_s}{2} - y'_{nd} \ . \tag{5.5}$$

Po tej transformaciji imata koordinati y_{nd} in y_{nl} enake vrednosti.

5.4 Sprememba velikosti slike obraza

Naslednji korak pri normalizaciji je sprememba velikosti slike tako, da bo razdalja med očmi $d0$ enotna za vse slike. Razdaljo med očmi izračunamo s pomočjo novih koordinat oči kot: $d = x_{nl} - x_{nd}$. Na podlagi tako izračunane razdalje določimo faktor spremembe velikosti kot:

$$f = \frac{d0}{d} \ . \tag{5.6}$$

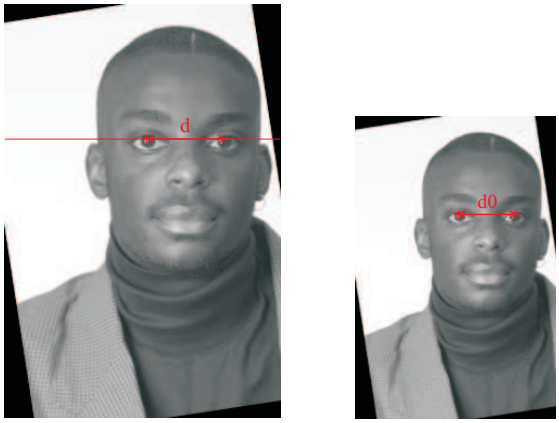
S pomočjo faktorja f nato spremenimo velikost celotne slike obraza. Če je faktor f večji od ena ($f > 1$), če je torej razdalja med očmi manjša od prej definirane $d < d0$, sliko povečamo, sicer jo zmanjšamo. Če moramo sliko povečati, zgubimo na kvaliteti in ne pridobimo nobene nove informacije, zato slike pri normalizaciji večinoma zmanjšamo. Slika 5.4 prikazuje tako zmanjšan obraz na predhodno definirano horizontalno razdaljo $d0$ med očmi.

Po tej transformaciji je razdalja med očmi $d0 = x_{nl} * k - x_{nd} * k$ na vseh normaliziranih slikah enaka.

5.5 Rezanje slike obraza

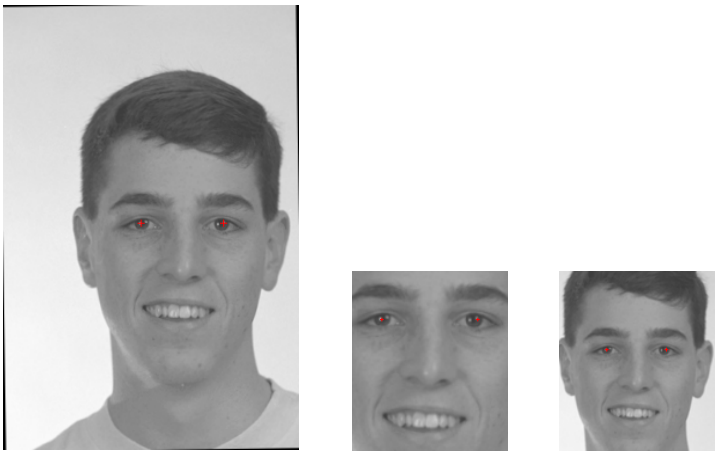
Naslednji korak je obrez slike na prej definirano višino v_{norm} in širino s_{norm} slike. S pomočjo rezanja odstranimo ozadje slike obraza, ki je velikokrat različno in zaradi tega moteče za samo prepoznavo obraza. Sliko lahko obrežemo tako, da odstranimo samo del ozadja in ohranimo cel obraz, lahko pa ohranimo samo osrednji del obraza z glavnimi obraznimi značilkami (slika 5.5).

S tem ko iz slike odrežemo samo obraz zagotovimo, da se nahajajo oči osebe vedno na istem mestu na sliki. Važen parameter pri rezanju je tudi, kako visoko od pozicije oči sliko odrežemo. Tako lahko ohranimo del pričeske ali pa jo popolnoma odrežemo. Lasje lahko služijo kot dobra informacija za ločevanje med različnimi



(a) (b)

Slika 5.4: Primer zmanjšanja slike obraza na podlagi razdalje med očmi: (a) rotirana originalna slika in (b) zmanjšana slika z razdaljo $d0$ med očmi.



(a) (b) (c)

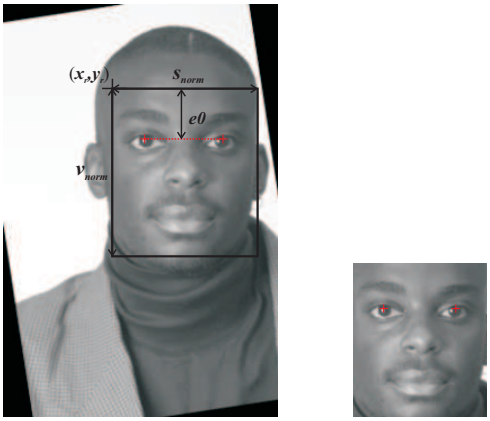
Slika 5.5: Primer različno rezanega obraza iz slike: (a) originalna slika obraza pri kateri (b) ohranimo samo osrednji del obraza ali (c) ohranimo celoten obraz in s tem del okolice.

osebami, vendar samo v primeru, ko ima oseba vedno enako pričesko, v nasprotnem primeru je lahko moteča. Definirajmo parameter $e0$, ki določa vertikalno razdaljo med zgornjim robom rezanja in očmi. Običajno nastavimo parameter $e0$ tako, da so na sliki še vidne značilke kot so oči na zgornjem robu ter usta ali brada na spodnjem robu. Levi rob rezanja pa nastavimo tako, da je pri dani širini s_{norm} sredinska točka med očmi na sredini slike. Začetno točko rezanja (x_r, y_r) definiramo kot:

$$x_r = x_{nd} * k - (s_{norm} - d0)/2 \ , \tag{5.7}$$

$$y_r = y_{nd} * k - e0 \ . \tag{5.8}$$

Slika 5.6 prikazuje primer iz slike odrezanega obraza, kjer sta označena parameter $e0$ in začetna točka rezanja (x_r, y_r).



(a) (b)

Slika 5.6: Primer odrezanega obraza iz slike glede na pozicijo oči: (a) originalna slika iz katere odrežemo sliko v točki $(x_r,\ y_r)$, (b) odrezana slika velikosti $v_{norm} \times s_{norm}$.

5.6 Maskiranje slike obraza

Pri tem koraku maskiramo odrezano sliko obraza na takšen način, da izločimo moteče ozadje. Maska se uporablja predvsem pri slikah, ki smo jih obrezali tako, da prikazujejo celoten obraz (slika 5.5(c)) in s tem tudi del ozadja. Maske so običajno ovalne oblike in enakih velikosti kot obrezana slika. Primer takšne maske prikazuje slika 5.7(b). Masko lahko predstavimo kot polje enic in ničel. Mesta, kjer se nahajajo slikovni elementi, ki bi jih radi ohranili predstavimo z enicami, na mestih ozadja katerega se bi pa radi znebili pa zapišemo ničle. S tako pripravljeno masko nato pomnožimo obrezano sliko (slika 5.7).

(a) (b) (c)

Slika 5.7: Primer maskiranja obrezane slike obraza: (a) odrezana slika obraza, (b) ovalna maska in (c) maskiran obraz.

S pomočjo maskiranja odstranimo nezaželeno ozadje na sliki obraza, ki se velikokrat spreminja in je zaradi tega moteče za samo prepoznavo.

5.7 Svetlobna normalizacija

Za svetlobno normalizacijo obstajajo zelo kompleksni in učinkoviti postopki, ki jih tukaj ne bomo obravnavali. Skoraj vsi normalizacijski postopki pa vključujejo vsaj dve preprosti tehniki svetlobne normalizacije. Prva je svetlobna poravnava slike s pomočjo izravnave histograma (angl. *histogram equalization*), drugi postopek pa temelji na normiranju vektorja slike s pomočjo povprečne vrednosti in standardne deviacije. Pri postopku izravnave histograma gre za postopek enakomerne porazdelitve različnih odtenkov sivine v histogramu glede na prisotnost le te v sami sliki. Na takšen način dele slike, ki vsebujejo večjo frekvenco barvnih razlik bolj poudarimo, medtem ko dele slik z manjšo frekvenco izenačimo. Primer slike normalizirane po postopku izravnave histograma prikazuje slika 5.8.

(a) (b)

Slika 5.8: Primer normaliziranega obraza s pomočjo postopka izravnave histograma: (a) pomanjšan in odrezan obraz in (b) svetlobno normaliziran obraz.

S pomočjo drugega postopka normaliziramo sliko obraza tako, da je njena povprečna vrednost enaka nič in njena standardna deviacija enaka ena. Najprej sliko pretvorimo v vektor $\mathbf{x}$ tako, da združimo vrstice oziroma stolpce. Nato izračunamo povprečno vrednost μ_x vektorja $\mathbf{x}$ in njegovo standardno deviacijo σ_x. Vektor $\mathbf{x}$ nato normaliziramo po sledečem postopku:

$$\mathbf{x}_n = \frac{\mathbf{x} - \mu_x}{\sigma_x} \quad . \tag{5.9}$$

Tako izračunani vektor $\mathbf{x}_n$ ima povprečno vrednost enako nič in standardno deviacijo enako ena. Primer tako normaliziranega obraza prikazuje slika 5.9.

(a) (b)

Slika 5.9: Primer normaliziranega obraza na osnovi povprečne vrednosti in standardne deviacije: (a) pomanjšan in odrezan obraz, (b) normaliziran obraz s povprečno vrednostjo nič ($\mu_{x_n} = 0$) in enotsko standardno deviacijo ($\sigma_{x_n} = 1$)

5.8 Različna normalizacija in različni deli obraza

Različni avtorji uporabljajo za predpripravo slik obrazov pred samim postopkom prepoznave različne normalizacijske postopke. Prvi korak normalizacije, to je poravnava slike glede na pozicijo oči, je nujen korak in je enolično določen. Ostali koraki pa se lahko med seboj razlikujejo. Rezultat tega so različno normalizirane slike obrazov in posledično tudi različni rezultati uspešnosti prepoznavanja obrazov pri enakih metodah.

Nekateri avtorji uporabljajo zelo različne razdalje med očmi kar pomeni tudi različno velike obraze pri enaki velikosti normalizirane slike. Velikost razdalje določimo s pomočjo parametra $d0$. Metodologija FERET priporoča velikost 70 slikovnih elementov pri velikosti normalizirane slike 150×130. Slika 5.10 prikazuje primere normaliziranega obraza z različnimi razdaljami med očmi pri enaki velikosti slike obraza.

Slika 5.10: Primeri normaliziranega obraza z različnimi razdaljami med očmi: $d0 = \{22, 24, 26, 28, 30\}$ pri normalizirani velikosti slike 60×50 slikovnih elementov.

Tudi velikosti same normalizirane slike se zelo razlikujejo. Nekateri avtorji upoštevajo priporočila FERET in uporabljajo velikost 150×130 slikovnih elementov. Ostale velikosti, ki se pogosto uporabljajo so še 128×128, 60×50 in 96×64 slikovnih elementov.

Drugi spremenljivi parameter je $e0$, ki določa rob rezanja glede na pozicijo oči. S tem parametrom določimo, na kakšni višini se nahajajo oči normalizirane slike. Avtorji različnih metod za prepoznavanje obrazov običajno ne navajajo tega podatka. Vsi pa poskrbijo, da so s pravilno izbiro parametra $e0$ pri dani velikosti slike na sliki vidne vse obrazne značilke. Slika 5.11 prikazuje primere normaliziranega obraza z različnimi vrednostmi parametra $e0$.

Slika 5.11: Primeri normaliziranega obraza z različnimi pozicijami oči, kar določa parameter $e0 = \{18, 21, 24, 27, 30\}$ pri normalizirani velikosti slike 60×50 slikovnih elementov.

V večini primerov se postopek normalizacije z rezanjem obraza tudi zaključi. V nekaterih primerih pa normalizacija vključuje tudi maskiranje in svetlobno normalizacijo. Pri svetlobni normalizaciji se velikokrat uporablja en od zgoraj predstavljenih postopkov. Nekateri avtorji pa uporabljajo oba. Predvsem v sistemih, kjer je pomembna hitrost, se ne uporablja normalizacije s pomočjo histogramov, ker je računsko bolj zahtevna.

V poglavju 9 smo preizkusili različne normalizacijske postopke s katerimi smo poskušali ugotoviti kako le-ti vplivajo na samo uspešnost metod. Poskušali smo ugotoviti vpliv parametrov kot sta $d0$ in $e0$ ter kako vpliva maska ter različni postopki svetlobne normalizacije na uspešnost prepoznavanja obrazov.

Nekateri avtorji, ki so pri normalizaciji uporabljali samo zgornji del obraza, so pri prepoznavanju dosegali boljše rezultate. Predvsem je bilo to opaziti pri metodah, ki so občutljive na različne obrazne izraze. Tudi pri osebah, posnetih v daljših časovnih presledkih se je uspešnost povečala, če je normalizacijski postopek odrezal pričesko osebe, saj se je le-ta po vsej verjetnosti v tem času spremenila.

Zaradi teh ugotovitev smo tudi mi preizkusili, kako se uspešnost metode spremeni, če upoštevamo samo del obraza. Preizkusili smo uspešnost na slikah, ki so vsebovale samo desno ali levo polovico obraza ali samo zgornjo, srednjo ali spodnjo polovico obraza (slika 5.12).

Slika 5.12: Primeri delno zakritega normaliziranega obraza: leva, desna polovica, spodnji, srednji, zgornji del.

Naslednjo skupino testnih slik smo pripravili tako, da so vsebovale samo dele obraza, kot to prikazuje slika 5.13. S pomočjo preizkusa metod na posameznih delih obraza smo tudi poskušali ugotoviti, kateri del obraza je za posamezno metodo najpomembnejši ter na splošno po katerem delu obraza lahko osebe najbolje ločimo.

Slika 5.13: Primeri posameznih delov normaliziranega obraza: čelo, oči, nos in usta.

5.9 Povzetek poglavja

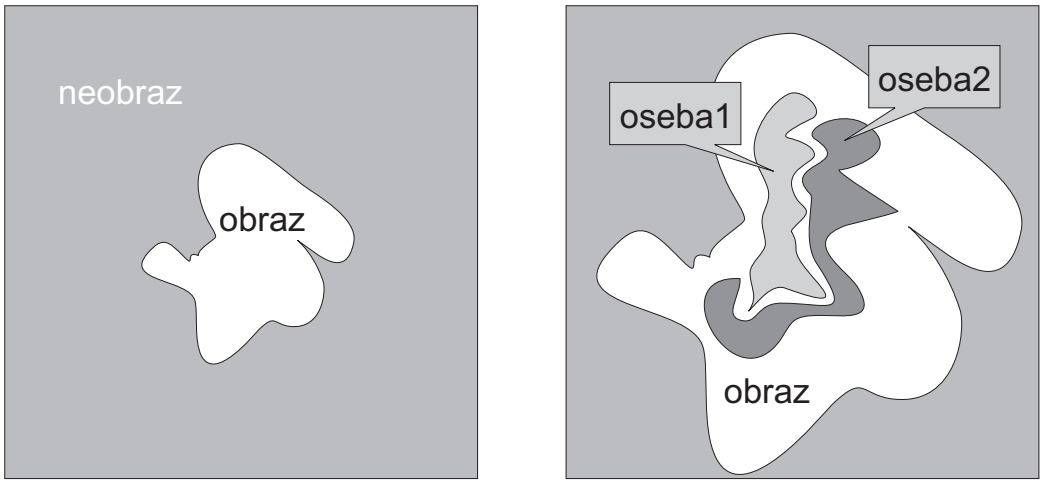
V tem poglavju smo predstavili različne postopke, ki se navadno uporabljajo pri normalizaciji slik obrazov. Predstavili smo geometrijsko normalizacijo obraza, ki obsega rotacijo, rezanje in spremembo velikosti. Predstavili smo tudi dva postopka svetlobne normalizacije: histogramsko izravnavo in normalizacijo na podlagi povprečne vrednosti in standardne deviacije. Opisali smo tudi postopek maskiranja slike obraza, s katerim lahko dodatno odstranimo moteče ozadje. Pogledali smo si tudi, kako se normaliziran obraz spremeni pri različnih parametrih razdalje med očmi in z različnimi pozicijami oči na sliki. Na koncu smo si pogledali normalizacijo različni delov obrazov in sicer različne polovice: leva, desna, zgornja, srednja in spodnja ter posamezne dele obraza. S pomočjo tako definiranih postopkov normalizacije smo pri preizkusih preverili kako le-ti vplivajo na uspešnost metod za prepoznavanje obrazov.

Poglavje 6

Metode osnovane na videzu

6.1 Tehnični izzivi in obrazni podprostori

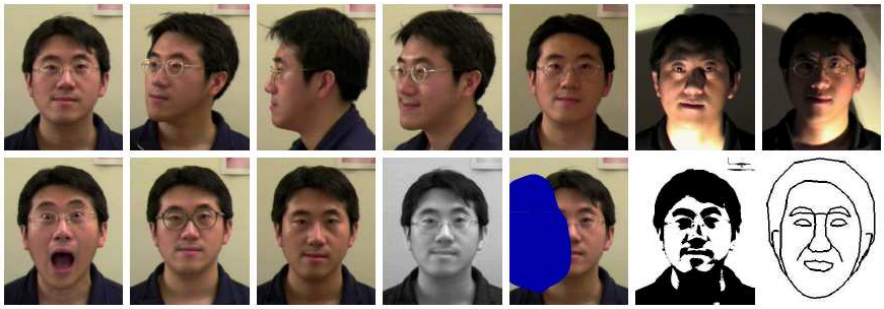
Na obraze različnih izgledov lahko gledamo tudi kot na obrazno mnogoterost (angl. *manifold*), medtem ko vse ostalo kar ni obraz predstavljajo neobrazne mnogoterosti. Če predstavimo obrazne mnogoterosti v slikovnem prostoru vidimo, da so nelinearne in nekonveksnih oblik. Slika 6.1(a) prikazuje mnogoterosti obraza in neobraza predstavljeni v slikovnem prostoru. Na sliki 6.1(b) pa prikazujemo dve različni osebi znotraj obrazne mnogoterosti. Na nalogo iskanja obraza na sliki lahko tako gledamo kot na razločevanje med obrazno in neobrazno mnogoterostjo, na nalogo prepoznave pa kot na ločevanje med posameznimi elementi (obrazi) znotraj obrazne mnogoterosti.



(a) (b)

Slika 6.1: (a) Mnogoterost obraza in neobraza. (b) Mnogoterosti dveh različnih oseb znotraj obrazne mnogoterosti.

Klasifikacijski problem, povezan z iskanjem obrazov, torej ni linearen in tudi ni konveksen. Toliko bolj to velja za klasifikacijo različnih oseb znotraj razreda obrazov. Mnogoterost posameznega subjekta pa postane še bolj kompleksna, če se spremenijo določene okoliščine, kot so osvetlitev, velikost ali zorni kot pogleda. Neodvisne raziskave najuspešnejših metod [79, 77] za prepoznavo obrazov so pokazale, da njihova uspešnost močno pade, če se ti pogoji spremenijo. Medtem ko sta oblika in odbojna jakost obraza bistveni lastnosti, se videz obraza lahko močno spreminja zaradi različnih pogojev: pogled kamere, osvetlitev, izraz na obrazu (slika 6.2).



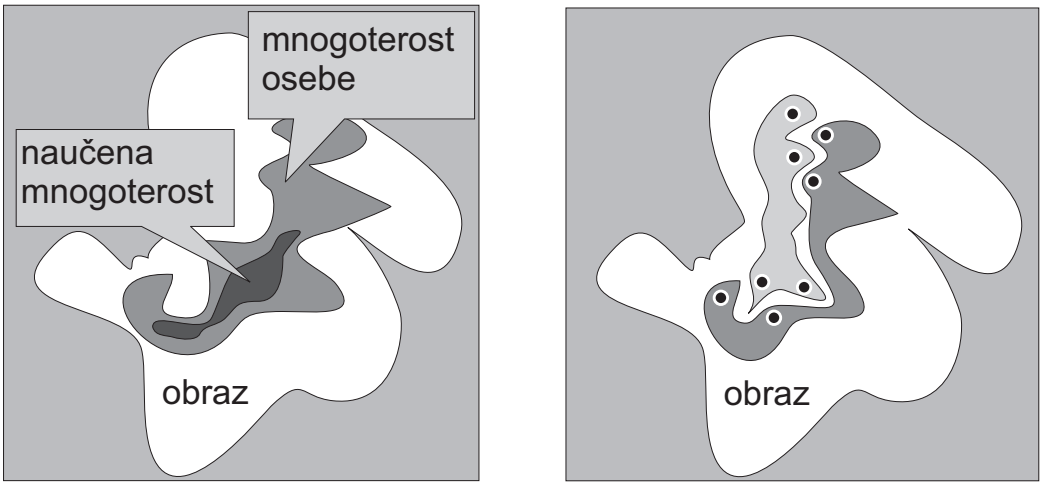
Slika 6.2: Variacije istega obraza zaradi različnega položaju, različne osvetlitve, izraza na obrazu, zakrivanja, dodatkov in barve.

Različen videz iste osebe je lahko tudi posledica naprave same: različen ekspoziciji čas, odprtost zaslonke, barva beline, različne leče ter napake leč, različni senzorji za zajemanje itd. Nenazadnje pa težavo povzročajo tudi osebe, ki so si med seboj zelo podobne, kot so to na primer dvojčki ali podobnosti med očetom in sinom. Vsi ti faktorji vplivajo na različne izglede iste osebe in lahko povzročijo, da so razlike med slikami iste osebe večje kot razlike med različnimi osebami. Zaradi tega je težko izbrati takšne značilke na obrazu, ki enolično določajo obraz posameznika.

Druga izziv pri prepoznavi obrazov je zmožnost generalizacije. Tipična slika obraza velikosti 112×92 slikovnih elementov leži v 10.304 dimenzijskem prostoru značilk. Običajno je največje število učnih primerov za posamezno osebo največ 10, največkrat pa samo ena ali dve. Takšna velikost je za učenje mnogoterosti, ki opisuje posamezno osebo, mnogo premajhna v primerjavi z dimenzijo slikovnega prostora. Sistem, ki ga naučimo s pomočjo tako majhnega števila učnih primerov za posamezen razred, namreč ni sposoben dovolj dobre generalizacije, da bi prepoznal nov, predhodno neznan primer obraza (slika 6.3(a)).

Zaradi velike kompleksnosti in nelinearnosti obraznih mnogoterosti posamezne osebe ne moremo med seboj linearno ločiti in za merjenje ujemanja preprosto uporabiti Evklidske razdalje. Na sliki 6.3(b) lahko vidimo, da je lahko Evklidska razdalja med posameznimi primeri obraza znotraj različnih mnogoterosti manjša, kot pa razdalja znotraj iste mnogoterosti.

Obstajata dve strategiji spopadanja z zgoraj opisanimi problemi: (1) ustrezna izbira značilnih točk in (2) ustrezna klasifikacijska metoda, ki upošteva nelinearnost



(a) (b)

Slika 6.3: Izzivi pri prepoznavanju obrazov: (a) klasifikator ne more dovolj dobro predstaviti (generalizirati) novega neznanega obraz iste osebe s pomočjo naučene mnogoterosti, (b) s pomočjo evklidske razdalje ne moremo ločiti med različnima osebama.

in kompleksnost obraznih mnogoterosti. Cilj prve strategije je zgraditi takšen prostor značilk (angl. *feature space*), v katerem bodo mnogoterosti manj kompleksne in bolj linearne ter s tem med seboj bolj ločljive. Takšen prostor zgradimo z ustrezno geometrijsko in svetlobno normalizacijo in ustreznim izborom takšnih značilk, ki so znotraj posameznega razreda iste osebe čimbolj stabilne. V nadaljevanju si bomo pogledali, kako lahko zgradimo takšen podprostor s pomočjo različnih metod. V tem poglavju bomo predstavili metodo glavnih komponent – PCA.

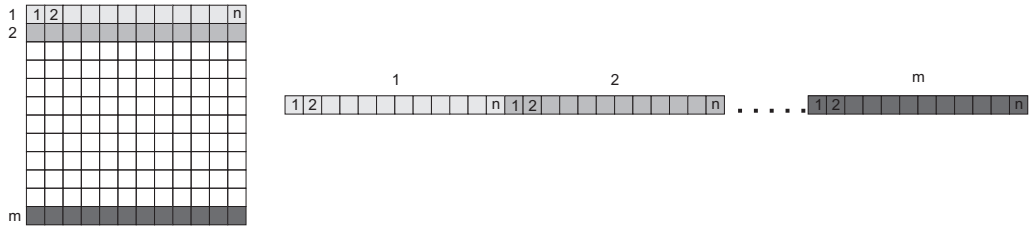
6.1.1 Obrazni podprostori

Prepoznava obrazov na podlagi analize podprostorov bazira na dejstvu, da lahko razred nam zanimivih vzorcev, v našem primeru obrazov, predstavimo v veliko manjšem podprostoru vhodnega prostora slik. S 4096 slikovnimi elementi, ki jih vsebuje slika velikosti 64×64 slikovnih elementov lahko na primer prikažemo veliko število najrazličnejših vzorcev, kot so drevesa, hiše, avtomobili ali obrazi. Med svojimi $256^{4096} > 10^{9864}$ različnimi "konfiguracijami" jih samo nekaj pripada vzorcem obraza. Zaradi tega je takšna predstavitev slike zelo redundantna, še posebej, če nas zanima samo določen vzorec. V takem primeru lahko dimenzijo takšne predstave močno zmanjšamo.

Pri digitalni obdelavi slik obrazov imamo opravka z vizualnimi signali (svetloba, ki se odbije od površine obraza), ki jih digitalni senzor registrira kot matriko slikovnih elementov. Slikovni elementi lahko nosijo barvno informacijo ali samo barvno

intenziteto. V našem delu se bomo omejili na slednje, torej sivinske slike, kjer nas bo zanimala samo intenziteta posameznega slikovnega elementa. Po ustrezni normalizaciji in zmanjšanju velikosti na $m \times n$ slikovnih elementov, lahko matriko slikovnih elementov predstavimo kot točko oziroma vektor $\mathbf{x}$ v mn dimenzijskem slikovnem prostoru. Postopek zapisa posameznega slikovnega elementa matrike v vektor prikazuje slika 6.4.

Kritična točka tako predstavljenih slik je ravno njihova dimenzija (velikost vektorja) – to je število koordinat (slikovnih elementov), ki jih potrebujemo, da določimo posamezno točko, ki predstavlja sliko obraza.



Slika 6.4: Predstavitev slike $m \times n$ v vektor $\mathbf{x}$ velikosti $M = mn$.

Za opis posamezne slike moramo torej določiti vrednosti vseh slikovnih točk. Število vseh slikovnih elementov je za sliko velikosti $m \times n$ enako mn, kar je veliko, tudi če imamo opraviti z majhnimi slikami. Metode za prepoznavanje, ki delujejo neposredno na slikah predstavljenih s pomočjo slikovnih elementov, se srečujejo z velikimi težavami. Prva težava je računska kompleksnost, kajti obdelava in primerjava slik v tako velikem dimenzijskem prostoru je tako računsko kot prostorsko kompleksna. Pri parametričnih metodah potrebujemo za predstavitev slik zelo veliko parametrov, saj število parametrov eksponentno narašča z velikostjo dimenzije. Običajno je število parametrov, ki jih potrebujemo za opis, veliko večje od števila učnih primerov. Podobno je pri neparametričnih metodah, kjer je kompleksnost vzorca nedosegljivo visoka. Če bi hoteli namreč opisati porazdelitev podatkov, bi potrebovali ogromno število učnih primerov.

Če pogledamo na obrazne vzorce, lahko ugotovimo, da zavzema veliko površine koža, ki je gladka in ima značilno teksturo. To pomeni, da je posamezna vrednost slikovnega elementa močno odvisna od vrednosti sosednjih elementov. Razen tega pa ima vsak obraz tipičen videz. To pomeni, da je simetričen, na straneh ima dvoje oči, na sredini nos, itd. Velik del slikovnih elementov obraza torej ne doprinese veliko oziroma skoraj nič pri sami predstavitvi obraza v slikovnem prostoru. Zaradi tega se je porodila ideja, da lahko slike obrazov predstavimo v veliko manjšem prostoru, ki ga imenujemo tudi obrazni prostor ali podprostor (angl. *face space*).

S pomočjo analize glavnih komponent (angl. *Principal Component Analysis*, PCA) [49, 98] lahko iz učne množice obrazov izločimo majhno število tako imenovanih lastnih-obrazov (angl. *eigenfaces*). Tako lahko sliko obraza s pomočjo lastnih obrazov predstavimo kot vektor značilk manjše dimenzije. Značilke tako izbra-

nega podprostora predstavljajo boljšo osnovo za prepoznavo, kot to omogoča slika, predstavljena v originalnem prostoru slikovnih elementov. Ravno uporaba tehnik zmanjševanja prostora je veliko pripomogla k napredku samih tehnologij za prepoznavanje obrazov. V nadaljevanju bomo podrobneje spoznali metodo PCA, ki se najpogosteje uporablja za zmanjševanje dimenzije slikovnega prostora. V naslednjih poglavjih pa bomo spoznali še metodi LDA (poglavje 6.3) in ICA (poglavje 6.4).

6.2 Metoda glavnih komponent

Metoda glavnih komponent (angl. *Principal Component Analysis*, PCA) je linearna transformacija iz več-dimenzionalnega vhodnega prostora v nižje-dimenzijski prostor značilk. Med vsemi linearnimi transformacijami zagotavlja najboljšo predstavitev visoko-dimenzijskih vektorjev vhodnih podatkov v nižje-dimenzijski prostor značilk. Koordinatni sistem zavrti na takšen način, da lahko raznolikost podatkov ustrezno predstavimo že z majhnim številom baznih vektorjev ali tako imenovanimi glavnimi komponentami (angl. *principal components*). Prva glavna komponenta je linearna kombinacija dimenzij originalnega prostora, ki imajo največjo varianco. Vsaka naslednja pa ima naslednjo najvišjo varianco in mora biti pravokotna na vse prejšnje.

Metodo je prvič formuliral Hotelling [34] leta 1933 in jo zaradi tega imenujemo tudi transformacija Hotelling. Metoda PCA pa je tesno povezana tudi s Karhunen-Loève transformacijo (angl. *Karhunen-Loève transform*, KLT) poimenovano po avtorjih Kariju Karhunenu in Michelu Loèvelu. Transformacija KLT velja v kontekstu procesiranja signalov kot ortogonalna transformacija, ki minimizira kvadratno rekonstrukcijsko napako. Pod predpostavko, da imajo vhodni podatki povprečno vrednost enako nič (angl. *zero-mean*), pa sta formulaciji metod PCA in KLT popolnoma enaki.

Metoda mora izbrati glavne komponente na takšen način, da minimizira kvadratno rekonstrukcijsko napako med vhodnimi podatki in njihovo predstavitvijo ter zagotoviti, da je njihova varianca največja. Izkaže se, da sta oba kriterija ekvivalentna in ju lahko dosežemo z uporabo metode PCA. Uspešnost metode je močno odvisna od korelacije vhodnih več-dimenzionalnih podatkov. Stopnja korelacije označuje odvisnosti med podatki in določa koliko odvečne informacije vsebujejo podatki. Metoda PCA namreč odstrani odvečno informacijo na takšen način, da zmanjša korelacijo vhodnih vektorjev. Nove koordinate vhodnih vektorjev (glavne komponente) so zaradi tega med seboj nekorelirane. Posledično lahko metodo PCA uporabimo tudi kot zelo močno orodje za zgoščevanje koreliranih podatkov.

6.2.1 Matematična definicija

Postopek za izračun glavnih komponent metode PCA lahko izpeljemo tako, da poiščemo takšno linearno transformacijo, pri kateri je varianca transformiranih po-

datkov največja.

Predpostavimo, da imamo N M-dimenzionalnih vektorjev $\mathbf{x}_j$, ki jih zapišemo v stolpce matrike $\mathbf{X} \in \mathbb{R}^{M \times N}$. Naj vektor $\mathbf{u}$ označuje smer (smerni vektor dolžine 1) v $\mathbb{R}^M$. Projekcijo j-tega vektorja $\mathbf{x}_j$ na vektor $\mathbf{u}$ izračunamo na naslednji način:

$$a_j = \langle \mathbf{x}_j, \mathbf{u} \rangle = \mathbf{u}^\top \mathbf{x}_j = \sum_{i=1}^{M} u_i x_{ij} \ . \tag{6.1}$$

Naš cilj je poiskati takšen smerni vektor $\mathbf{u}$, ki maksimizira varianco vseh projiciranih vhodnih vektorjev $\mathbf{x}_j, j = 1 \ldots N$.

Povprečje projiciranih podatkov je:

$$\bar{a} = \frac{1}{N} \sum_{j=1}^{N} a_j = \frac{1}{N} \sum_{j=1}^{N} \sum_{i=1}^{M} u_i x_{ij} = \sum_{i=1}^{M} u_i \mu_i \tag{6.2}$$

in varianca je

$$\begin{aligned} \sigma^2 &= \frac{1}{N} \sum_{j=1}^{N} (a_j - \bar{a})^2 = \frac{1}{N} \sum_{j=1}^{N} \left(\sum_{i=1}^{M} u_i x_{ij} - \sum_{i=1}^{M} u_i \mu_i \right)^2 = \\ &= \frac{1}{N} \sum_{j=1}^{N} \left(\sum_{i=1}^{M} u_i \hat{x}_{ij} \right)^2 = \frac{1}{N} \sum_{j=1}^{N} \sum_{i=1}^{M} \sum_{l=1}^{M} u_i \hat{x}_{ij} u_l \hat{x}_{lj} = \\ &= \sum_{i=1}^{M} \sum_{l=1}^{M} u_i u_l \frac{1}{N} \langle \hat{\mathbf{x}}_{i:}, \hat{\mathbf{x}}_{l:} \rangle = \sum_{i=1}^{M} \sum_{l=1}^{M} u_i c_{il} u_l = \mathbf{u}^\top \mathbf{C} \mathbf{u} \ , \end{aligned} \tag{6.3}$$

kjer μ_i označuje povprečno vrednost i-te vrstice v podatkovni matriki $\mathbf{X}$ in $\hat{x}_{ij}$ označuje vrednost spremenljivke x_{ij} kateri je odšteto povprečje μ_i. Oznaka s podpisom x_i označuje i-ti stolpec matrike $\mathbf{X}$, medtem ko $x_{i:}$ označuje i-to vrstico. Z vektorjem $\boldsymbol{\mu}$ označimo povprečne vrednosti vseh vrstic matrike $\mathbf{X}$:

$$\boldsymbol{\mu} = [\mu_1, \ldots, \mu_M]^\top = \frac{1}{N} \sum_{j=1}^{N} \mathbf{x}_j \ , \tag{6.4}$$

in z matriko $\hat{\mathbf{X}}$ centralizirane vhodne podatke, t.j. od povprečne vrednosti odštete posamezne stolpce matrike $\mathbf{X}$

$$\hat{\mathbf{X}} = \mathbf{X} - \boldsymbol{\mu} \mathbf{1}_{1 \times N} \ , \tag{6.5}$$

kjer $\mathbf{1}_{1 \times N}$ označuje matriko enic velikosti $1 \times N$. Matrika $\mathbf{C}$ pa določa kovariančno matriko vhodnih podatkov $\mathbf{X}$:

$$\mathbf{C} = \frac{1}{N} \hat{\mathbf{X}} \hat{\mathbf{X}}^\top \ . \tag{6.6}$$

Naš cilj je maksimizirati varianco σ^2 pri pogoju, da je $\|\mathbf{u}\| = 1$. Gre za maksimizacijo z omejitvijo, ki se običajno rešuje z metodo Langrangevega multiplikatorja. V tem primeru določimo Langrangejevo funkcijo takole:

$$F(\mathbf{u};\lambda) = \mathbf{u}^\top \mathbf{C}\mathbf{u} - \lambda(\mathbf{u}^\top \mathbf{u} - 1) = \sum_{i=1}^{M}\sum_{j=1}^{M} u_i c_{ij} u_j - \lambda\left(\sum_{i=1}^{M} u_i^2 - 1\right) \quad , \tag{6.7}$$

kjer je λ neznana konstanta, znana pod imenom Langrangev multiplikator in $(\mathbf{u}^\top\mathbf{u} - 1)$ predstavlja omejitev. λ in $\mathbf{u}$, za katere bo imela funkcija F največjo vrednost, so rešitve postavljenega problema glavnih komponent.

Rešitev tega problema dobimo s pomočjo parcialnega odvoda:

$$\begin{aligned}
\frac{\partial F}{\partial u_l} = \sum_{j=1}^{M} c_{lj} u_j + \sum_{i=1}^{M} u_i c_{il} - \lambda 2 u_l &= 0 \ ; \ l = 1 \ldots M \\
\sum_{i=1}^{M} c_{li} u_i &= \lambda u_l \ ; \ l = 1 \ldots M \\
\mathbf{C}\mathbf{u} &= \lambda \mathbf{u} \ .
\end{aligned} \tag{6.8}$$

Če hočemo torej najti $\mathbf{u}$ in λ, pri katerih je vrednost enačbe 6.7 največja, moramo izračunati lastne vektorje in lastne vrednosti kovariančne matrike $\mathbf{C}$. Največja lastna vrednost, ki jo dobimo, pripada največji varianci, medtem ko pripadajoči lastni vektor določa smer z največjo varianco.

Lastne vrednosti in vektorje lahko izračunamo s pomočjo razcepa z lastnimi vrednostmi (angl. *Eigenvalue Decomposition*, EVD) ali s pomočjo razcepa s singularnimi vrednostmi (angl. *Singular Value Decomposition*, SVD) kovariančne matrike $\mathbf{C}$. S pomočjo razcepa s singularnimi vrednostmi lahko matriko $\mathbf{C} \in \mathbb{R}^{M\times M}$ razcepimo kot:

$$\mathbf{C} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top \quad , \tag{6.9}$$

da vsebuje ortonormirana matrika $\mathbf{U} = [\mathbf{u}_1, \ldots \mathbf{u}_M] \in \mathbb{R}^{M\times M}$ po stolpcih lastne vektorje $\mathbf{u}_1, \ldots, \mathbf{u}_M$ in diagonalna matrika $\mathbf{\Lambda} \in \mathbb{R}^{M\times M}$ na diagonali lastne vrednosti $\lambda_1, \ldots, \lambda_M$. Tako dobljeni lastni vektorji sestavljajo bazo ortonormiranih (ortogonalnih in enotskih) vektorjev. Ker pa je kovariančna matrika oblike $\mathbf{C} = \hat{\mathbf{X}}\hat{\mathbf{X}}^\top$, kjer je $\hat{\mathbf{X}} \in \mathbb{R}^{M\times N}$ in ker je običajno število učnih primerov manjše od dimenzije podatkov ($N < M$), je rang matrike $\mathbf{C}$ lahko manjši ali največ enak N. Iz tega sledi, da ima matrika $\mathbf{C}$ največ N nenegativnih lastnih vrednosti. Potemtakem lahko s pomočjo metode SVD razcepimo matriko $\mathbf{C} \in \mathbb{R}^{M\times M}$ na manjšo ortonormirano matriko lastnih vektorjev $\mathbf{U} = [\mathbf{u}_1, \ldots \mathbf{u}_N] \in \mathbb{R}^{M\times N}$ in diagonalno matriko $\mathbf{\Lambda} \in \mathbb{R}^{N\times N}$ lastnih vrednosti $\lambda_1, \ldots, \lambda_N$.

Predpostavimo, da so lastne vrednosti in pripadajoči lastni vektorji urejeni po padajočem vrtnem redu lastnih vrednosti, torej $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_N$. Tako večino

variabilnosti vhodnih vektorjev vsebujejo že prvi lastni vektorji, ki jih imenujemo tudi glavni vektorji (angl. *principal vectors*) ali glavne osi.

Izračun glavnih vektorjev po opisanem postopku pa je možen samo v primeru, če velikost vhodnega vektorja ni prevelika. Na področju računalniškega vida, ko je vhodni podatek slika, imamo opraviti z veliko dimenzijo M vhodnih podatkov in zaradi tega ogromno kovariančno matriko $\mathbf{C} \in \mathbb{R}^{M \times M}$ (6.6). V tem primeru, ko velja $N << M$, lahko namesto kovariančne matrike $\mathbf{C} \in \mathbb{R}^{M \times M}$ izračunamo matriko notranjega produkta $\mathbf{C}' \in \mathbb{R}^{N \times N}$:

$$\mathbf{C}' = \frac{1}{N} \hat{\mathbf{X}}^\top \hat{\mathbf{X}} \ . \tag{6.10}$$

Lastne vrednosti in lastne vektorje kovariančne matrike $\mathbf{C}$ nato dobimo s pomočjo lastnih vrednosti λ'_i in lastnih vektorjev $\mathbf{u}'_i$ matrike $\mathbf{C}'$:

$$\lambda_i = \lambda'_i \tag{6.11}$$

$$\mathbf{u}_i = \frac{\hat{\mathbf{X}}\mathbf{u}'_\mathbf{i}}{\sqrt{N\lambda'_i}} \ , \ i = 1 \ldots N \ . \tag{6.12}$$

Glede na zgornje izpeljave, lahko standardni pristop za izračun glavnih komponent metode PCA zapišemo kot algoritem 2.

Algoritem 2 : Metoda glavnih komponent – PCA

Vhod: podatkovna matrika $\mathbf{X}$

Izhod: vektor povprečnih vrednosti $\boldsymbol{\mu}$, lastni vektorji $\mathbf{U}$, lastne vrednosti $\boldsymbol{\lambda}$.

1: Določi vektor povprečnih vrednosti: $\boldsymbol{\mu} = \frac{1}{N} \sum_{j=1}^{N} \mathbf{x}_j$.
2: Centriraj podatke okrog povprečja: $\hat{\mathbf{X}} = \mathbf{X} - \boldsymbol{\mu}\mathbf{1}_{1 \times N}$.
3: **če** $M \leq N$ **potem**
4: Določi kovariančno matriko: $\mathbf{C} = \frac{1}{N}\hat{\mathbf{X}}\hat{\mathbf{X}}^\top$.
5: Izvedi SVD na matriki $\mathbf{C}$. Dobimo lastne vektorje $\mathbf{U}$ in lastne vrednosti $\boldsymbol{\lambda}$.
6: **sicer**
7: Določi matriko notranjega produkta: $\mathbf{C}' = \frac{1}{N}\hat{\mathbf{X}}^\top\hat{\mathbf{X}}$.
8: Izvedi SVD na matriki $\mathbf{C}'$. Dobimo lastne vektorje $\mathbf{U}'$ in lastne vrednosti $\boldsymbol{\lambda}'$.
9: Določi glavne vektorje $\mathbf{U}$: $\mathbf{u}_i = \frac{\hat{\mathbf{X}}\mathbf{u}'_\mathbf{i}}{\sqrt{N\lambda'_i}}$, $i = 1 \ldots N$.
10: Določi lastne vrednosti $\boldsymbol{\lambda} = \boldsymbol{\lambda}'$.
11: **konec če**

Glavne vektorje in lastne vrednosti pa lahko izračunamo tudi s pomočjo razcepa s singularnimi vrednostmi centralizirane matrike $\hat{\mathbf{X}}$ vhodnih podatkov. Razcep s singularnimi vrednostmi matrike $\hat{\mathbf{X}}$ velikosti $M \times N$, kjer je $M \geq N$ zapišemo kot

$$\hat{\mathbf{X}} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{V}^\top \ , \tag{6.13}$$

kjer ima matrika $\mathbf{U} \in \mathbb{R}^{M \times N}$ in $\mathbf{V} \in \mathbb{R}^{N \times N}$ ortonormirane stolpce in matrika $\boldsymbol{\Lambda}$ singularne vrednosti matrike $\mathbf{X}$ na glavni diagonali, drugod pa ničle. Dokažemo

lahko, da vsebujejo stolpci matrike $\mathbf{U}$ lastne vektorje in da so singularne vrednosti matrike $\mathbf{X}$ kvadratni koreni lastnih vrednosti kovariančne matrike $\mathbf{C}$ (6.6). Iz enačb 6.6 in 6.13 in ob upoštevanju ortonormiranosti matrike $\mathbf{V}$ (torej $\mathbf{V}^\top\mathbf{V} = \mathbf{I}$) velja

$$\begin{aligned}
\mathbf{C} &= \frac{1}{N}\hat{\mathbf{X}}\hat{\mathbf{X}}^\top = && (6.14)\\
&= \frac{1}{N}(\mathbf{U}\boldsymbol{\Lambda}\mathbf{V}^\top)\cdot(\mathbf{U}\boldsymbol{\Lambda}\mathbf{V}^\top)^\top = \\
&= \frac{1}{N}\mathbf{U}\boldsymbol{\Lambda}\mathbf{V}^\top\mathbf{V}\boldsymbol{\Lambda}\mathbf{U}^\top = \\
&= \frac{1}{N}\mathbf{U}\boldsymbol{\Lambda}\boldsymbol{\Lambda}\mathbf{U}^\top = \\
&= \frac{1}{N}\mathbf{U}\boldsymbol{\Lambda}^2\mathbf{U}^\top \ , && (6.15)
\end{aligned}$$

kar je v bistvu ekvivalentno enačbi 6.9. Iz tega sledi, da lahko glavne komponente metode PCA učinkovito določimo že s samim razcepom vhodne matrike $\hat{\mathbf{X}}$, namesto da računamo kovariančno matriko $\mathbf{C}$ (6.6), ki je lahko zelo velika v primeru, ko je število učnih primerov N veliko manjše od dimenzije posameznega učnega primera M, kar je na področju računalniškega vida pogosto.

6.2.2 Lastnosti metode PCA

Ortonormirana matrika $\mathbf{U}$, ki vsebuje glavne vektorje, lahko služi kot linearna transformacijska matrika, ki projicira iz več-dimenzijskega vhodnega prostora v nižje-dimenzijski podprostor značilk in obratno. Stolpci matrike $\mathbf{U}$ so tako bazni vektorji novega nižje-dimenzijskega koordinatnega sistema, ki je izražen z večdimenzionalnimi koordinatami. Tako lahko vhodni vektor $\mathbf{x}$, od katerega odštejemo povprečni vektor $\boldsymbol{\mu}$, projiciramo v glavni podprostor s pomočjo transformacijske matrike $\mathbf{U}^\top : \mathbb{R}^M \rightarrow \mathbb{R}^N$:

$$\mathbf{a} = \mathbf{U}^\top\hat{\mathbf{x}} \ . \tag{6.16}$$

Na takšen način se koeficienti a_j izračunajo kot projekcija vhodne slike v posamezen glavni vektor:

$$a_j = \langle\hat{\mathbf{x}}, \mathbf{u}_j\rangle = \sum_{i=1}^{M} u_{ij}\hat{x}_i \ , \qquad j = 1\ldots N \ . \tag{6.17}$$

Vse vhodne vektorje, ki jih vsebuje vhodna matrika $\hat{\mathbf{X}}$ lahko projiciramo kot $\mathbf{A} = \mathbf{U}^\top\hat{\mathbf{X}}$. Medtem, ko je matrika $\mathbf{A}$ ortonormirana transformacija centralizirane matrike $\hat{\mathbf{X}}$ so tudi glavne komponente, ki jih dobimo centrirane okrog vrednosti nič:

$$\boldsymbol{\mu}_{\mathbf{A}} = \frac{1}{N}\sum_{j=1}^{N}\mathbf{a}_j = \frac{1}{N}\sum_{j=1}^{N}\mathbf{U}^\top\hat{\mathbf{x}} = \mathbf{U}^\top\frac{1}{N}\sum_{j=1}^{N}\hat{\mathbf{x}} = \mathbf{0} \ . \tag{6.18}$$

Izračunamo še kovariančno matriko tako dobljene matrike $\mathbf{A}$:

$$\mathbf{C_A} = \frac{1}{N}\mathbf{A}\mathbf{A}^\top = \frac{1}{N}\mathbf{U}^\top\hat{\mathbf{X}}(\mathbf{U}^\top\hat{\mathbf{X}})^\top = \mathbf{U}^\top\frac{1}{N}\hat{\mathbf{X}}\hat{\mathbf{X}}^\top\mathbf{U} \;=$$
$$= \mathbf{U}^\top\mathbf{C}\mathbf{U} = \mathbf{U}^\top\mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top\mathbf{U} \;=\; \mathbf{\Lambda}\ . \tag{6.19}$$

V zgornji enačbi smo matriko $\mathbf{C}$ zapisali v diagonalizirani obliki (6.9) in upoštevali ortonormiranost matrike $\mathbf{U}$ (torej $\mathbf{U}^\top\mathbf{U} = \mathbf{I}$). Izkaže se, da je kovariančna matrika transformiranih podatkov diagonalna matrika $\mathbf{\Lambda}$, ki vsebuje na glavni diagonali lastne vrednosti. Ta lastnost ima dve pomembni posledici. Prvič, potrjuje da, so transformirani vektorji med seboj nekorelirani. To pomeni, da je bila redundanca, kot posledica korelacije med vhodnimi vektorji, odpravljena. Drugič, pa dokazuje, da je varianca v smeri i-te glavne osi enaka i-ti lastni vrednosti λ_i, tako da velja $\frac{1}{N}\sum_{j=1}^{N} a_{ij}^2 = \lambda_i$.

Pomembna lastnost diagonalizacije (6.9) je tudi ta, da ohrani sled (angl. *trace*) matrike, ki jo diagonaliziramo. Medtem, ko je vsota diagonalnih elementov kovariančne matrike vsota variance vhodnih vektorjev, se je tudi celotna varianca vhodnih podatkov ohranila in je enaka vsoti lastnih vrednosti:

$$\mathrm{VAR}(\mathbf{X}) = \sum_{i=1}^{M}\frac{1}{N}\hat{\mathbf{x}}_{i:}\hat{\mathbf{x}}_{i:}^\top = \sum_{i=1}^{M} c_{ii} \;=$$
$$= \sum_{i=1}^{N}\lambda_i = \sum_{i=1}^{N}\frac{1}{N}\mathbf{a}_{i:}\mathbf{a}_{i:}^\top \;=\; \mathrm{VAR}(\mathbf{A})\ . \tag{6.20}$$

Poglejmo si sedaj kako lahko matriko $\mathbf{U}$ uporabimo kot transformacijsko matriko, s katero projiciramo vektor koeficientov nazaj v vhodni prostor. To operacijo imenujemo rekonstrukcija. Vektor koeficientov $\mathbf{a}$ lahko rekonstruiramo s pomočjo matrike $\mathbf{U} : \mathbb{R}^N \to \mathbb{R}^M$:

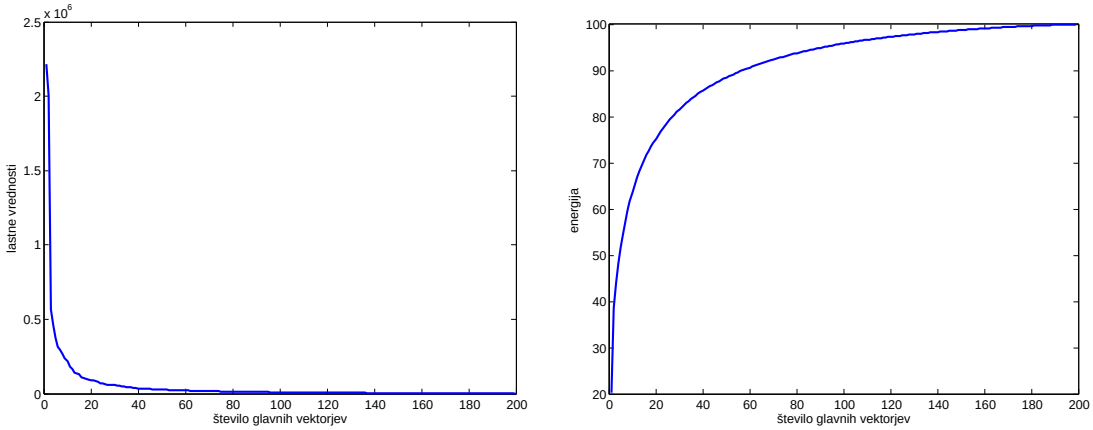
$$\hat{\mathbf{y}} = \mathbf{U}\mathbf{a} = \sum_{j=1}^{N} a_j\mathbf{u}_j\ . \tag{6.21}$$

Ker N lastnih vektorjev v matriki $\mathbf{U} \in \mathbb{R}^{M\times N}$ obsega enak podprostor v $\mathbb{R}^M$ kot vseh N vhodnih slik matrike $\mathbf{X} \in \mathbb{R}^{M\times N}$, lahko vsako vhodno sliko iz $\mathbf{X}$ popolnoma rekonstruiramo brez izgube. Kar je za nas bolj pomembno je to, kako dobro lahko rekonstruiramo vhodno sliko, če uporabimo samo podmnožico glavnih komponent.

Da to ugotovimo moramo najprej pogledati kako je varianca porazdeljena po glavnih oseh. To porazdeljenost imenujemo lastni spekter (angl. *eigenspectrum*) in jo običajno predstavimo kot graf lastnih vrednosti urejenih po padajočem vrstnem redu. Tipični lastni spekter je predstavljen na sliki 6.5(a). Kot lahko vidimo, večino variance vsebuje nekaj prvih lastnih vektorjev. Podobno lahko to ugotovimo z merjenjem energije, ki je določena kot delež celotne variance. Energijo vsebovano v prvih k lastnih vektorjih lahko tako izračunamo kot:

$$en_k = \frac{\sum_{i=1}^{k}\lambda_i}{\sum_{i=1}^{N}\lambda_i}\ . \tag{6.22}$$

Tako lahko energijo lastnih vrednosti, katerih spekter smo prikazali na sliki 6.5(a) prikažemo kot graf na sliki 6.5(b). Ponovno lahko vidimo, da večino energije ohrani že nekaj prvih lastnih vektorjev.



(a) (b)

Slika 6.5: Tipični (a) lastni spekter in (b) energija.

Iz tega lahko zaključimo, da lahko dobimo dober približek vhodnih slik z upoštevanjem zgolj podmnožice lastnih vektorjev, ki jim pripadajo največje lastne vrednosti. Tako bomo od sedaj naprej upoštevali samo k ($k << N$) glavnih osi, tako da bo transformacijska matrika $\mathbf{W}$ vsebovala samo prvih k glavnih vektorjev: $\mathbf{W} = [\mathbf{u}_1, \ldots, \mathbf{u}_k] \in \mathbb{R}^{M \times k}$.

Sedaj se vhodni vektor projicira v k-dimenzionalni glavni podprostor s pomočjo transformacijske matrike $\mathbf{W}^\top : \mathbb{R}^M \to \mathbb{R}^k$:

$$\begin{aligned} \mathbf{a} &= \mathbf{W}^\top \hat{\mathbf{x}} = \mathbf{W}^\top (\mathbf{x} - \boldsymbol{\mu}) \\ a_j &= \langle \hat{\mathbf{x}}, \mathbf{u}_j \rangle = \sum_{i=1}^{M} u_{ij} \hat{x}_i = \sum_{i=1}^{M} u_{ij}(x_i - \mu_i) \ , \ \ j = 1 \ldots k \end{aligned} \tag{6.23}$$

in rekonstruira s pomočjo transformacijske matrike $\mathbf{W} : \mathbb{R}^k \to \mathbb{R}^M$:

$$\begin{aligned} \hat{\mathbf{y}} &= \mathbf{W}\mathbf{a} = \sum_{j=1}^{k} a_j \mathbf{u}_j \\ \mathbf{y} &= \hat{\mathbf{y}} + \boldsymbol{\mu} \ . \end{aligned} \tag{6.24}$$

Na takšen način lahko vhodno sliko aproksimiramo z linearno kombinacijo prvih k glavnih vektorjev.

Rekonstrukcijska napaka (vektor preostanka) je enaka razliki med vhodnim in rekonstruiranim vektorjem:

$$\mathbf{e} = \hat{\mathbf{x}} - \hat{\mathbf{y}} = \sum_{j=1}^{N} a_j \mathbf{u}_j - \sum_{j=1}^{k} a_j \mathbf{u}_j = \sum_{j=k+1}^{N} a_j \mathbf{u}_j \ . \tag{6.25}$$

Običajno pa se za merilo napake raje uporablja kvadratna rekonstrukcijska napaka e, ki je določena kot kvadrat vektorja, ki ga zavržemo oziroma ne upoštevamo. Ob upoštevanju ortonormiranosti lastnih vektorjev $\mathbf{u}_j$ imamo

$$e = \|\mathbf{e}\|^2 = \left\| \sum_{j=k+1}^{N} a_j \mathbf{u}_j \right\|^2 = \sum_{j=k+1}^{N} a_j^2 \ . \tag{6.26}$$

Potemtakem je kvadratna rekonstrukcijska napaka enaka vsoti kvadratov neuporabljenih glavnih komponent. Ker pa jih običajno vnaprej ne poznamo, lahko pričakovano napako približno določimo s pomočjo pričakovanih vrednosti variance med zavrženimi lastnimi vektorji, kar je enako pripadajočim lastnim vrednostim:

$$\mathcal{E}(e) = \sum_{j=k+1}^{N} \lambda_j \ . \tag{6.27}$$

Potemtakem je pričakovana napaka enaka vsoti zavrženih ali neupoštevanih lastnih vrednosti. Ta ugotovitev potrjuje tudi dejstvo, da s povečevanjem variance prvih (padajoče urejenih) lastnih vektorjev, hkrati zmanjšujemo tudi kvadratno rekonstrukcijsko napako. To dejstvo pa je tudi ena izmed pomembnih lastnosti metode PCA.

Za dano dimenzijo k podprostora, torej metoda PCA poišče takšne glavne vektorje $\mathbf{u}_l, l = 1 \ldots k$ in vektorje koeficientov ali glavne komponenete $\mathbf{a}_j \in \mathbb{R}^k, j = 1 \ldots N$, ki zmanjšajo celotno kvadratno rekonstrukcijsko napako

$$e = \sum_{i=1}^{M} \sum_{j=1}^{N} \left(\hat{x}_{ij} - \sum_{l=1}^{k} u_{il} a_{lj} \right)^2 \ . \tag{6.28}$$

Tako lahko kot alternativo maksimizacije variance, določimo glavne vektorje in glavne komponente z minimizacijo kvadratne rekonstrukcijske napake (6.28). Tukaj imamo opraviti z nelinearnim minimizacijskim problemom, ki ga lahko rešimo z enim od predlaganih algoritmov, kot je gradientno iskanje (angl. *gradient descend*) [54] ali nevronske mreže [18].

Takšna predstavitev je še posebej primerna za modeliranje množice vhodnih vektorjev, ki so med seboj zelo korelirani, saj jih dekorelira in s tem odpravi redundanco v vhodnih podatkih ter tako zmanjša količino potrebnih podatkov za predstavitev. Tako je metoda PCA optimalna v smislu zgoščevanja podatkov, saj pri določeni stopnji kompresije (številu baznih vektorjev) zagotavlja minimalno rekonstrukcijsko napako (enačba 6.28).

Na PCA pa lahko gledamo tudi kot na mejni primer Gaussovega modela, ko je šum neskončno majhen in enak v vseh smereh. Roweis [81] je iz te opazke izpeljal algoritem za izračun glavnega podpostora, ki temelji na EM algoritmu [16, 81, 97]. Algoritem izračuna glavne smeri in glavne komponente na iterativen način z izmeničnim izvajanjem naslednjih E in M korakov:

- E-korak: $\mathbf{A} = (\mathbf{U}^\top\mathbf{U})^{-1}\mathbf{U}^\top\hat{\mathbf{X}}$
- M-korak: $\mathbf{U} = \hat{\mathbf{X}}\mathbf{A}^\top(\mathbf{A}\mathbf{A}^\top)^{-1}$.

6.2.3 Enostaven 2D primer

Poskušajmo sedaj na primeru dvodimenzionalnih vhodnih podatkov izračunati glavne vektorje in glavne komponente ter preveriti zgoraj opisane lastnosti. Naša vhodna matrika $\mathbf{X} \in \mathbb{R}^{2\times 10}$ vsebuje 10 različnih primerov podatkov, ki so predstavljeni z dvema koordinatama. Ekvivalentno zgornji definiciji označimo število primerov z $N = 10$ in dimenzijo prostora z $M = 2$.

$$\mathbf{X} = \begin{bmatrix} 2,5 & 0,5 & 2,2 & 1,9 & 3,1 & 2,3 & 2 & 1 & 1,5 & 1,1 \\ 2,4 & 0,7 & 2,9 & 2,2 & 3,0 & 2,7 & 1,6 & 1,1 & 1,6 & 0,9 \end{bmatrix} \tag{6.29}$$

Najprej izračunamo vektor povprečnih vrednosti $\boldsymbol{\mu}$ in vhodne podatke centraliziramo $\hat{\mathbf{X}}$ (6.5).

$$\begin{aligned} \boldsymbol{\mu}^\top &= \begin{bmatrix} 1,81 & 1,91 \end{bmatrix} \qquad (6.30) \\ \hat{\mathbf{X}} &= \begin{bmatrix} 0,69 & -1,31 & 0,39 & 0,09 & 1,29 & 0,49 & 0,19 & -0,81 & -0,31 & -0,71 \\ 0,49 & -1,21 & 0,99 & 0,29 & 1,09 & 0,79 & -0,31 & -0,81 & -0,31 & -1,01 \end{bmatrix} \end{aligned}$$

Tako centralizirane podatke prikažemo kot črne točke na sliki 6.6. Cilj metode PCA je najti nov koordinacijski sistem, v katerem bodo te točke kar najbolje predstavljene s pomočjo enega samega glavnega vektorja. Glavne osi novega koordinatnega sistema določimo tako, da izračunamo lastne vektorje in lastne vrednosti kovariančne matrike $\mathbf{C}$ (6.9, 6.6):

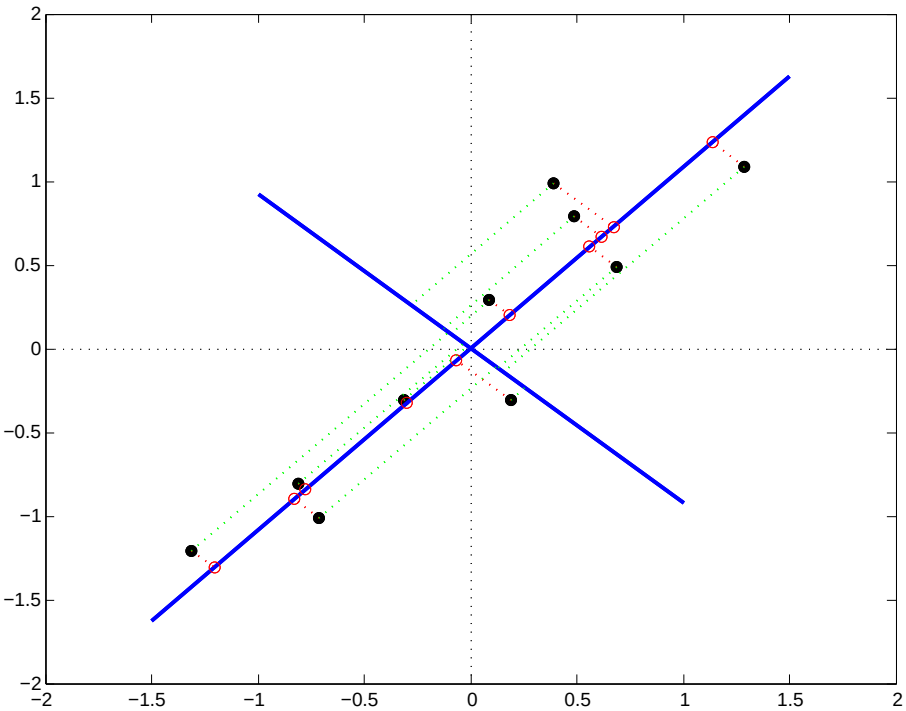
$$\mathbf{C} = \frac{1}{N}\hat{\mathbf{X}}\hat{\mathbf{X}}^\top = \begin{bmatrix} 0,55 & 0,55 \\ 0,55 & 0,64 \end{bmatrix} \tag{6.31}$$

$$\mathbf{C} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^\top = \begin{bmatrix} -0,68 & -0,73 \\ -0,74 & 0,68 \end{bmatrix} \begin{bmatrix} 1,16 & 0 \\ 0 & 0,04 \end{bmatrix} \begin{bmatrix} -0,68 & -0,74 \\ -0,74 & 0,68 \end{bmatrix} \tag{6.32}$$

Tako dobljeni glavni osi (glavna vektorja) sta na sliki 6.6 označeni z modrima črtama. V našem primeru imamo dva takšna vektorja, pri čemer je prvi glavni vektor ($\mathbf{u}_1^\top = (-0.68 - 0.74)$) tisti, ki mu pripada največja lastna vrednost ($\lambda_1 = 1.16$) (nosi največjo varianco). Zaradi korelacije med elementi vhodnih podatkov, vsebuje prvi glavni vektor največ variabilnosti vhodnih točk. Tako dobimo s projekcijo vhodnih točk na prvo glavno os enodimenzionalne glavne komponente, ki so hkrati tudi najboljša možna enodimenzionalna predstavitev vhodnih podatkov.

$$\begin{aligned} \mathbf{a} &= \mathbf{u}_1^\top\hat{\mathbf{X}} = \qquad (6.33) \\ &= [0,83, -1,78, 0,99, 0,27, 1,68, 0,91, -0,10, -1,14, -0,44, -1,22] \end{aligned}$$

Glavne komponente, imenovane tudi koeficienti so, na sliki 6.6 prikazane z rdečimi točkami. Glavne osi so določene na takšen način, da minimizirajo kvadratno rekonstrukcijsko napako med vhodnimi točkami in njihovo predstavitvijo ter maksimizirajo varianco glavnih komponent. Glede na sliko 6.6 je cilj zavrteti modro črto v takšno smer, da so rdeče črte čim krajše in zelene črte čim daljše. Izkaže se, da sta oba kriterija, tako minimizacija napake kot maksimizacija variance, ekvivalentna in ju lahko dosežemo s pomočjo metode PCA.



Slika 6.6: Delovanje metode PCA za primer 2D podatkov.

Preverimo sedaj, če veljajo vse lastnosti, ki smo jih opisali zgoraj. Najprej preverimo, če je povprečje transformiranih centraliziranih vhodnih podatkov še vedno enako nič:

$$\begin{aligned} \mathbf{A} &= \mathbf{U}^\top \hat{\mathbf{X}} = \qquad (6.34) \\ &= \begin{bmatrix} 0,83 & -1,78 & 0,99 & 0,27 & 1,68 & 0,91 & -0,10 & -1,14 & -0,44 & -1,22 \\ -0,18 & 0,14 & 0,38 & 0,13 & -0,21 & 0,18 & -0,35 & 0,05 & 0,02 & -0,16 \end{bmatrix} \\ \boldsymbol{\lambda}_A &= \frac{1}{N}\sum_{j=1}^{N} \mathbf{a}_j = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \qquad (6.35) \end{aligned}$$

Preverimo, če je korelacijska matrika $\mathbf{C_A}$, matrike glavnih komponent $\mathbf{A}$ res enaka matriki lastnih vrednosti:

$$\mathbf{C_A} = \frac{1}{N}\mathbf{A}\mathbf{A}^\top = \begin{bmatrix} 1,16 & 0 \\ 0 & 0,04 \end{bmatrix} \quad . \qquad (6.36)$$

Iz tega lahko tudi vidimo, da so transformirane točke med seboj nekorelirane in da nosi največjo varianco prva glavna komponenta.

Preverimo, če se je celotna varianca vhodnih podatkov $\mathbf{X}$ po transformaciji ohranila (6.20):

$$\begin{aligned} \mathrm{VAR}(\mathbf{X}) &= 0,555 + 0,645 = 1,20 \\ \mathrm{VAR}(\mathbf{A}) &= 1,16 + 0,04 = 1,20 \end{aligned} \tag{6.37}$$

Vidimo, da se je varianca transformiranih podatkov $\mathbf{A}$ ohranila in da je enaka vsoti lastnih vrednosti $(1,16 + 0,04 = 1,20)$.

Transformacijsko matriko $\mathbf{U}$ lahko uporabimo tudi za projekcijo vektorja koeficientov nazaj v vhodni prostor. Če uporabimo vse glavne vektorje, lahko vhodne podatke rekonstruiramo brez napake. Če uporabimo samo lastni vektor, ki mu pripada največja lastna vrednost, pa je napaka rekonstrukcije za posamezno točko:

$$\mathbf{a}_2 = \mathbf{u}_2^\top \hat{\mathbf{X}} \tag{6.38}$$

$$\mathbf{E} = \hat{\mathbf{X}} - \hat{\mathbf{Y}} = \mathbf{u}_2 \mathbf{a}_2 = \tag{6.39}$$

$$= \begin{bmatrix} 0,13 & -0,11 & -0,29 & -0,10 & 0,15 & -0,13 & 0,26 & -0,03 & -0,01 & 0,12 \\ -0,12 & 0,10 & 0,26 & 0,09 & -0,14 & 0,12 & -0,24 & 0,03 & 0,01 & -0,11 \end{bmatrix}$$

Oziroma kvadratna rekonstrukcijska napaka za posamezno točko znaša:

$$\begin{aligned} e_j &= ||\mathbf{e}_j||^2 \quad j = 1, \ldots N \\ e_j &= \begin{Bmatrix} 0,03 & 0,02 & 0,15 & 0,02 & 0,04 & 0,03 & 0,12 & 0,00 & 0,00 & 0,03 \end{Bmatrix} , \end{aligned} \tag{6.40}$$

kar je približno enako pričakovani napaki, ki jo lahko določimo kot vsoto neuporabljenih lastnih vrednosti $(\lambda_2 = 0,04)$.

Izračunamo lahko tudi, da nosi prva glavna komponenta več kot 96% celotne variance:

$$en_1 = \frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{N} \lambda_i} = \frac{\lambda_1}{\lambda_1 + \lambda_2} = 0,96 \tag{6.41}$$

6.2.4 Uporaba metode PCA za prepoznavanje obrazov

Domnevo, da človeški zaznavni sistem uporablja za pomnjenje in iskanje obrazov samo majhen nabor parametrov, sta avtorja Sirovich in Kirby [49, 50] hotela preveriti z zgoščevanjem slik obrazov. Za to nalogo sta uporabila metodo PCA. Metoda namreč omogoča, da lahko vsak primer $\mathbf{s}_i \in \mathbb{R}^d$ $(i = 1, \ldots, n)$ predstavimo samo s $p << d$ komponentami oziroma koordinatami nove baze, ki jo tvorijo vektorji $\mathbf{u}_i \in \mathbb{R}^d$ $(j = 1, \ldots, p)$. Seveda moramo shraniti tudi te vektorje, kar skupno znese $(n + d)p$, v nasprotju s prvotno potrebo po prostoru, ki je znašala nd. Sirovich in Kirby sta vektorje $\mathbf{u}_i$ poimenovala lastne slike, ker imajo enako dimenzijo kot

vhodne slike. Njun preizkus je pokazal, da lahko s pomočjo 40-ih lastnih slik uspešno rekonstruiramo 115 slik velikosti 128×128 slikovnih elementov s samo 3% napako. Opozoriti je potrebno, da s takšno predstavitvijo zmanjšamo prvotni prostor slik za več kot polovico[1]:

$$\begin{aligned} nd &= 115 * 128 * 128 * 8 = 15.073.280 \text{ bitov} = 1840 \text{ KB} \\ (n+d)p &= (115 + 128 * 128 * 8) * 40 = 5.247.480 \text{ bitov} = 641 \text{ KB} \end{aligned}$$

Avtorja sta pokazala tudi, da je napaka rekonstrukcije slike obraza, ki ni del podatkovne baze, manjša od 8% celo pri velikih svetlobnih odstopanjih kar dokazuje na veliko ekstrapolacijsko moč.

Čeprav sta idejo o uporabi lastnih slik za razpoznavanje obrazov nakazala v svojem delu že Sirovich in Kirby leta 1987, pa sta prvo realizacijo opisala avtorja Turk in Pentland leta 1991 [98]. Predstavila sta avtomatiziran sistem za prepoznavanje obrazov s pomočjo metode PCA in klasifikacije po metodi najbližjih sosedov. Lastne slike sta v svojem delu poimenovala lastni obrazi. V nadaljevanju bomo takšen postopek učenja in prepoznavanja obrazov s pomočjo metode PCA podrobneje opisali.

Postopek učenja

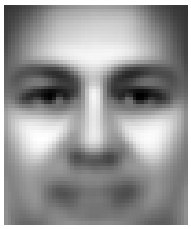
Sliko obraza $\mathbf{I}(x, y)$ si lahko predstavljamo kot dvodimenzionalno polje velikosti $m \times n$ 8 bitnih vrednosti. Lahko pa jo predstavimo tudi kot vektor dimenzije $M = nm$. Tako lahko sliko obraza velikosti 60×50 predstavimo kot vektor z dimenzijo 3000 ali ekvivalentno kot točko v 3000-dimenzijskem prostoru. Podobno lahko tudi na galerijo slik obrazov gledamo kot na množico točk v takšnem ogromnem prostoru.

Slike obrazov, ki so si po svoji strukturi zelo podobne, pa ne tvorijo naključne porazdeljenosti po celotnem slikovnem prostoru in jih lahko zato predstavimo v veliko manjšem podprostoru. Osnovna ideja metode glavnih komponent je poiskati takšne vektorje, ki najbolje opisujejo porazdeljenost obraznih slik v celotnem prostoru slik. Takšni vektorji določajo podprostor slik obrazov, ki ga zaradi tega imenujemo tudi obrazni prostor (angl. *face space*). Vsak takšen vektor dolžine M predstavlja sliko velikosti $m \times n$ in je linearna kombinacija originalnih slik obrazov. Ker so ti vektorji lastni vektorji kovariančne matrike vhodnih slik in ker so po izgledu podobni obrazom, jih imenujemo tudi lastni obrazi (angl. *eigenfaces*). Nekaj takšnih lastnih obrazov prikazuje slika 6.9.

Predpostavimo, da imamo učno množico N-tih obraznih slik: $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$ dimenzije $M = n \times m$. Povprečno sliko obraza učne množice določimo kot $\boldsymbol{\mu} = \frac{1}{N}\sum_{j=1}^{N} \mathbf{x}_j$. Vhodne podatke najprej centriramo, tako da od vsake slike odštejemo povprečni obraz $\boldsymbol{\mu}$: $\hat{\mathbf{x}}_i = \mathbf{x}_i - \boldsymbol{\mu}$. Nekaj primerov slik iz učne množice prikazuje slika 6.7. Na sliki 6.8 pa je prikazan povprečni obraz $\boldsymbol{\mu}$ učne množice.

[1] Pri izračunu smo upoštevali, da je slika sivinska in je vsaka slikovna točka predstavljena z 8 biti.

Slika 6.7: Primeri slik obrazov $\mathbf{x_i}$ iz učne množice.



Slika 6.8: Slika povprečnega obraza $\boldsymbol{\mu}$.

Na tako centraliziranih podatkih nato s pomočjo analize glavnih komponent poiščemo množico N-tih ortonormiranih vektorjev $\mathbf{u}_i$, ki najbolje opišejo porazdeljenost podatkov. I-ti vektor $\mathbf{u}_i$ je izbran tako, da je vrednost enačbe

$$\boldsymbol{\lambda}_i = \frac{1}{N}\sum_{j=1}^{N}(\mathbf{u}_i^\top \hat{\mathbf{x}}_j)^2 \tag{6.42}$$

največja ob predpostavki, da velja

$$\mathbf{u}_l^\top \mathbf{u}_k = \delta_{lk} = \begin{cases} 1, & \text{če je } l = k \\ 0, & \text{sicer}\,. \end{cases} \tag{6.43}$$

Vektorji $\mathbf{u}_i$ in skalarji λ_i so lastni vektorji in lastne vrednosti kovariančne matrike

$$\mathbf{C} = \frac{1}{N}\sum_{j=1}^{N}\hat{\mathbf{x}}_j\hat{\mathbf{x}}_j^\top = \frac{1}{N}\hat{\mathbf{X}}\hat{\mathbf{X}}^\top\,, \tag{6.44}$$

kjer matrika $\hat{\mathbf{X}} = (\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, \ldots, \hat{\mathbf{x}}_N)$ v stolpcih vsebuje centrirane slike obrazov. Matrika $\mathbf{C}$ je velikosti $M \times M$, kar je za našo velikost slike (M=3000) računsko prezahteven problem. Ker pa je število učnih primerov N veliko manjše od dimenzije M ($N \ll M$), lahko uporabimo predhodno predstavljen algoritem 2 in izračunamo matriko notranjega produkta. Velja namreč, da obstaja samo N namesto M pomembnih lastnih vektorjev (v našem primeru celo $N-1$, ker so vhodni podatki centrirani). Vsem ostalim lastnim vektorjem namreč pripadajo lastne vrednosti, ki imajo vrednost enako nič. S pomočjo tega postopka lahko poiščemo lastne vektorje dimenzije M tako, da najprej izračunamo lastne vektorje matrike notranjega produkta $\mathbf{C}' = \frac{1}{N}\hat{\mathbf{X}}^\top\hat{\mathbf{X}}'$, nato pa določimo lastne vektorje $\mathbf{u}'_\mathbf{i}$ in lastne vrednosti λ'_i matrike $\mathbf{C}' \in \mathbb{R}^{N\times N}$ da velja:

$$\hat{\mathbf{X}}^\top\hat{\mathbf{X}}\mathbf{u}'_\mathbf{i} = \lambda_i\mathbf{u}'_\mathbf{i} \ . \tag{6.45}$$

Enačbo pomnožimo z leve strani z matriko $\hat{\mathbf{X}}$ in dobimo:

$$\hat{\mathbf{X}}\hat{\mathbf{X}}^\top(\hat{\mathbf{X}}\mathbf{u}'_\mathbf{i}) = \lambda_i(\hat{\mathbf{X}}\mathbf{u}'_\mathbf{i}) \ . \tag{6.46}$$

Vidimo, da so lastni vektorji matrike $\mathbf{C}$ enaki

$$\mathbf{u}_\mathbf{i} = \hat{\mathbf{X}}\mathbf{u}'_\mathbf{i} \ , \tag{6.47}$$

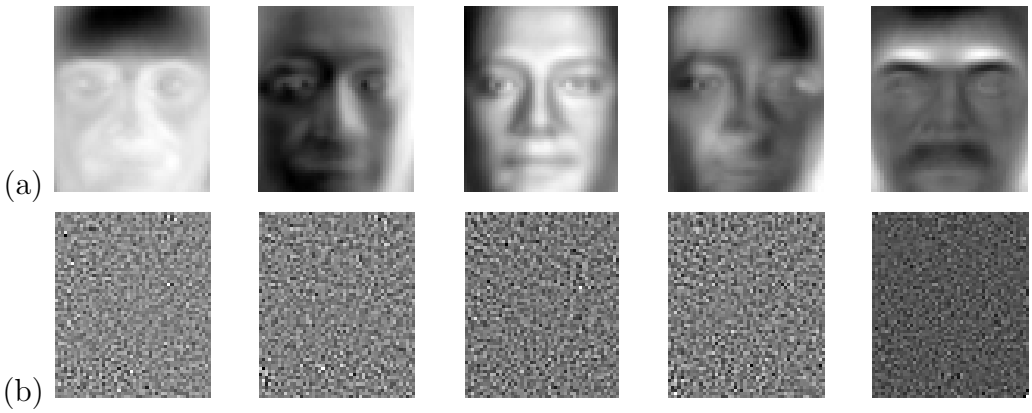
lastne vrednosti λ pa so enake lastnim vrednostim λ'. Kar pa tako dobljeni vektorji niso enotski jih moramo še normirati. Izkaže se, da je dolžina vektorjev $\hat{\mathbf{X}}\mathbf{u}'_\mathbf{i}$ enaka kvadratnemu korenu pripadajoče lastne vrednosti pomnožene s številom učnih primerov N (v primeru, da števila N pri računanju notranjega produkta ne upoštevamo (torej $\mathbf{C}' = \hat{\mathbf{X}}^\top\hat{\mathbf{X}}$), ga tudi tukaj izpustimo).

$$\mathbf{u}_\mathbf{i} = \frac{1}{\sqrt{N\lambda'_i}}\hat{\mathbf{X}}\mathbf{u}'_\mathbf{i} \ . \tag{6.48}$$

Na takšen način se računska zahtevnost močno zmanjša iz velikosti reda, ki je enak številu slikovnih elementov M, na velikost reda enakega številu učnih primerov N. Pripadajoče lastne vrednosti λ_i nam omogočajo, da uredimo lastne vektorje po pomembnosti opisa variabilnosti vhodnih slik. Slika 6.9(a) prikazuje prvih 5 lastnih vektorjev ali tako imenovanih lastnih obrazov, ki jim pripadajo največje lastne vrednosti. Slika 6.9(b) pa prikazuje zadnjih 5 lastnih obrazov. Vidimo lahko, da prvi lastni obrazi zelo dobro opisujejo slike, medtem ko se obrazne značilnosti pri lastnih obrazih manjših lastnih vrednosti porazgubijo. Lastni obrazi z manjšimi lastnimi vrednostmi največkrat predstavljajo šum na slikah in pri nadaljnji uporabi niso pomembni oziroma ne doprinesejo bistvene dodatne informacije.

Postopek učenja s pomočjo metode PCA opisuje algoritem 3.

Zelo pomemben, v glavnem nerešen problem pri zmanjševanju dimenzije, je pravilna izbira parametra k, to je števila uporabljenih glavnih komponent. Za izbiro števila k namreč ne obstaja analitična izpeljava. Da problem poenostavimo, se navadno predvideva, da zajet signal (v našem primeru slika obraza) vsebuje šum, tako



Slika 6.9: Primeri (a) prvih petih in (b) zadnjih petih lastnih obrazov.

Algoritem 3 : Postopek učenja z metodo PCA

Vhod: podatkovna matrika učnih slik obrazov $\mathbf{X}$,
število ohranjenih glavnih vektorjev k

Izhod: povprečna slika obraza $\boldsymbol{\mu}$, transformacijska matrika $\mathbf{W}$, lastne vrednosti $\boldsymbol{\lambda}$.

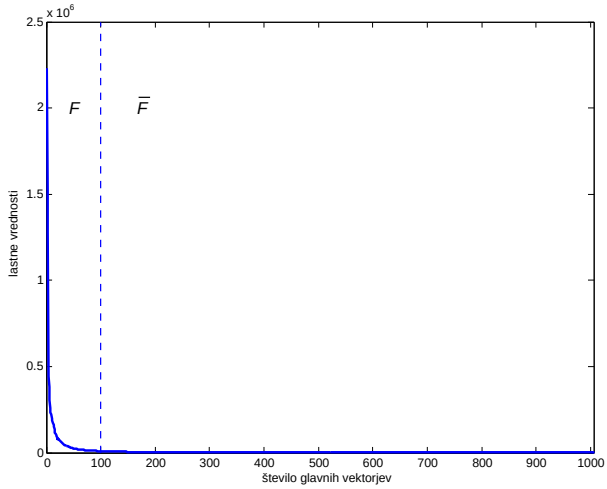
1: Določi povprečno sliko: $\boldsymbol{\mu} = \frac{1}{N}\sum_{j=1}^{N}\mathbf{x}_j$.
2: Centriraj slike obrazov: $\hat{\mathbf{X}} = \mathbf{X} - \boldsymbol{\mu}\mathbf{1}_{1\times N}$.
3: Določi matriko notranjega produkta: $\mathbf{C}' = \frac{1}{N}\hat{\mathbf{X}}^\top\hat{\mathbf{X}}$.
4: Izvedi SVD na matriki $\mathbf{C}'$. Dobimo lastne vektorje $\mathbf{U}'$ in lastne vrednosti $\boldsymbol{\lambda}'$.
5: Določi glavne vektorje $\mathbf{U}$: $\mathbf{u}_i = \frac{\hat{\mathbf{X}}\mathbf{u}'_i}{\sqrt{N\lambda'_i}}$, $i = 1\ldots N$.
6: Določi lastne vrednosti: $\boldsymbol{\lambda} = \boldsymbol{\lambda}'$.
7: Določi transformacijsko matriko: $\mathbf{W} = [\mathbf{u_1}, \mathbf{u_2}, \ldots, \mathbf{u_k}]$.

da je varianca podatkov zajeta s prvimi glavnimi komponentami veliko večja kot varianca med ostalimi komponentami. Te ugotovitve lahko vidimo tudi na lastnem spektru (angl. *eigenspectrum*), ki določa lastne vrednosti v padajočem vrstnem redu (slika 6.10). Kot vemo, določa i-ta lastna vrednost varianco vzdolž i-te glavne komponente. Potemtakem je primeren algoritem za iskanje vrednosti k takšen, da izbere vrednost pri tisti lastni vrednosti λ_i, pri kateri se vrednost nenadno najbolj zniža. Na takšen način lahko razdelimo prostor $\mathbb{R}^N$ na glavni podprostor F in njemu ortogonalni dopolnjujoč podprostor $\bar{F}$. Za naš primer, ko imamo $N = 1007$ učnih slik, bi na podlagi slike 6.10 lahko določili mejno vrednost $k = 100$. Pri tej vrednosti je skupna varianca, ki jo določa prvih k glavnih komponent enaka: $\sum_{j=1}^{100}\lambda_j = 0,910$.

Po drugi strani pa je s strani metodologije FERET [77] priporočljivo število komponent enako 40% vseh lastnih vektorjev, kar v našem primeru znaša $k = (1007) * 0.4 = 403$ in $\sum_{j=1}^{403}\lambda_j = 0,985$.

V svojem delu pa je avtor Swets [93] izbral toliko glavnih komponent k, da je ohranil najmanj 95% celotne energije.

V praksi pa se pri izbiri števila komponent gleda tudi na računsko zahtevnost,



Slika 6.10: Lastni spekter za primer obraznih slik in razdelitev na dva ortogonalna podprostora F in $\bar{F}$.

ki je večja pri večjem številu komponent in na število slik obrazov, ki jih imamo na voljo. V poglavju 9 bomo pri preizkusih ohranili različno število komponent in poskušali ugotoviti, kako ta izbira vpliva na samo uspešnost prepoznave, ter katero število komponent je v našem primeru optimalno.

Klasifikacija neznanega obraza

V prejšnjem podpoglavju smo določili k lastnih obrazov s pomočjo kovariančne matrike obrazov učne množice. Tako dobljeni lastni obrazi določajo bazne vektorje, s katerimi lahko predstavimo posamezno sliko obraza. Sliko obraza velikosti M lahko torej predstavimo s pomočjo k komponent ali koeficientov.

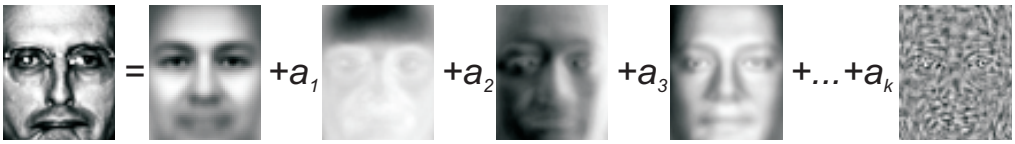
Za samo identifikacijo oziroma klasifikacijo moramo torej vse slike obrazov najprej predstaviti s pomočjo komponent lastnih obrazov, oziroma jih moramo projicirati v obrazni prostor. Posamezno sliko obraza $\mathbf{x}$ projiciramo na sledeči način:

$$\mathbf{a}_j = \mathbf{w}_j^\top(\mathbf{x} - \boldsymbol{\mu}), \qquad j = 1, \ldots, k \ , \tag{6.49}$$

kjer $\mathbf{w}_j$ določa posamezen glavni vektor transformacijske matrike $\mathbf{W}$. Slika 6.11 prikazuje originalno sliko obraza in njeno projekcijo v k-dimenzionalni obrazni prostor. Glavne komponente predstavljajo vektorji uteži $\mathbf{a}^\top = [a_1, a_2, \ldots, a_k]$, ki določajo prispevek posameznega lastnega obraza k predstavitvi vhodne slike obraza (slika 6.12).

Na takšen način projiciramo vse slike znanih obrazov. V nekaterih primerih so to kar učne slike, po metodologiji FERET pa je temu namenjena posebna množica slik imenovana galerija slik, ki jo bomo označili kot matriko $\mathbf{G}$. Matrika $\mathbf{G}$ vsebuje v svojih stolpcih posamezne slike obrazov, katerih identiteto poznamo. S projekcijo slik galerije dobimo matriko $\mathbf{P}$, ki vsebuje v svojih stolpcih koeficiente posamezne

Slika 6.11: Originalna slika obraza in njena projekcija v obrazni prostor s pomočjo prvik k lastnih obrazov.



Slika 6.12: Predstavitev slike obraza s posameznimi prispevki lastnih obrazov, ki jih določajo glavne komponente a_j.

slike obraza:

$$\mathbf{P} = \mathbf{W}^{\top}(\mathbf{G} - \boldsymbol{\mu} 1_{1\times N_G}) \ . \tag{6.50}$$

V primeru, če imamo v galeriji $\mathbf{G}$ več slik iste osebe c določimo povprečni vektor $\boldsymbol{\mu}_c$ tako, da izračunamo povprečje vseh predstavitev $\mathbf{a}_i$ določene osebe:

$$\boldsymbol{\mu}_c = \frac{1}{c}\sum_{i=1}^{c} \mathbf{a}_i \ . \tag{6.51}$$

Tako določeni vektorji matrike $\mathbf{P}$ predstavljajo posamezno osebo oziroma v smislu klasifikacije posamezen razred c. Tako dobljene vektorje glavnih komponent ali koeficientov pa lahko imenujemo tudi vektorji značilk, ker določajo posamezne značilne točke vhodnih slik. Takšen vektor značilk lahko tako uporabimo v standardnem algoritmu prepoznavanja vzorcev, ki poišče tistega od predhodno določenih razredov, ki najbolje opisuje iskani obraz. Metoda najbližjega soseda izbere tisti razred c, katerega razdalja med vhodno sliko in izbranim razredom c je najmanjša. Razdaljo med posameznimi vektorji lahko merimo na različne načine, za naš primer bomo uporabili Evklidsko razdaljo, ki se tudi najpogosteje uporablja:

$$r_c^2 = ||(\mathbf{a} - \boldsymbol{\mu}_c)||^2 \ , \tag{6.52}$$

kjer vektor $\boldsymbol{\mu}_c$ opisuje posamezen razred c oziroma posamezno osebo. Za merjenje razdalje pa lahko uporabimo tudi druge mere, kot so Kosinusna, Mahalanobis ali razdalja City blok. V poglavju 9 smo preizkusili 14 različnih razdalj in poskušali ugotoviti, kako izbira različne mere za razdaljo vpliva na uspešnost prepoznave in katera je za posamezno metodo najuspešnejša. Celoten postopek prepoznave neznanega obraza opisuje algoritem 4.

Algoritem 4 : Postopek prepoznave z metodo PCA

Vhod: vhodna slika obraza $\mathbf{x}$, galerija slik $\mathbf{G}$, povprečna slika obraza $\boldsymbol{\mu}$, transformacijska matrika $\mathbf{W}$, lastne vrednosti $\boldsymbol{\lambda}$.

Izhod: razred c, vrednost razdalje r_c,

1: Centriraj galerijo slik obrazov: $\hat{\mathbf{G}} = \mathbf{G} - \boldsymbol{\mu} 1_{1\times N_G}$.
2: Projiciraj slike iz galerije v obrazni prostor: $\mathbf{P} = \mathbf{W}^\top \hat{\mathbf{G}}$.
3: Centriraj vhodno sliko $\mathbf{x}$ neznanega obraza: $\hat{\mathbf{x}} = \mathbf{x} - \boldsymbol{\mu}$.
4: Projiciraj sliko v obrazni prostor: $\mathbf{a} = \mathbf{W}^\top \hat{\mathbf{x}}$.
5: **za vse** razrede c iz matrike $\mathbf{P}$ transformiranih slik galerije **delaj**
6: Izračunaj razdaljo za posamezen razred c: $r_c^2 = ||\mathbf{a} - \mathbf{p_c}||^2$.
7: Izberi najmanjšo razdaljo r_c in pripadajoči razred c .
8: **konec za**

Rekonstrukcija obraza

S tem, ko določimo vektor glavnih komponent, preslikamo originalno sliko obraza v nižje-dimenzijski obrazni prostor. Na takšen način pa lahko preslikamo katerokoli sliko, tudi takšno, na kateri ni obraza. Z izračunom razdalje r med originalno sliko in njeno rekonstrukcijo lahko ugotovimo, ali tako predstavljena slika v obraznem prostoru res predstavlja obraz:

$$r^2 = ||\hat{\mathbf{x}} - \hat{\mathbf{y}}||^2 \ , \tag{6.53}$$

kjer je vektor $\hat{\mathbf{y}} = \sum_{j=1}^{k} a_j \mathbf{u_j}$ rekonstrukcija vhodne slike $\mathbf{x}$ iz obraznega prostora.

Avtorja Turk in Pentland sta razdaljo r uporabila za predhodno kontrolo ali vhodna slika res predstavlja obraz ali ne. Če je bila razdalja r večja od predhodno določene meje, sta sliko zavrnila, drugače pa sta izračunala razdalje r_c do posameznega razreda v galeriji slik obrazov. Če je bila razdalja r_c večja od dovoljene meje, sta obraz označila kot neznan, drugače sta vrnila identiteto osebe, ki je pripadala razredu z najmanjšo razdaljo med vektorji komponent. Sistem za prepoznavo pa lahko dodatno razširimo tako, da v primeru neznanega obraza, le-tega dodamo v učno množico. Sistem lahko potem ponovno naučimo na novi učni množici, lahko pa uporabimo inkrementalno metodo učenja in sistem učimo sproti.

Turk in Pentland pa sta razdaljo r, ki določa oddaljenost vhodne slike od obraznega prostora, uporabila tudi za iskanje obraza na sliki. Za posamezno podsliko sta določila razdaljo in tako za celotno sliko dobila matriko razdalj, ki je določala, kje na sliki se nahaja obraz.

6.3 Linearna diskriminantna analiza

Če se vhodne slike iste osebe večinoma razlikujejo zaradi različne osvetlitve ali zaradi različnega izraza na obrazu, se variacija v podatkih nahaja predvsem zaradi teh razlik. Pomanjkljivost metode PCA, ki smo jo predstavili v prejšnjem poglavju 6.2 je ravno v tem, da podobnost med podatki v obraznem prostoru, ki ga določa, ne predstavlja nujno iste osebe, ampak je lahko posledica istih pogojev osvetlitve ali izraza na obrazu.

Ko projiciramo podatke v novi podatkovni prostor, jih hočemo čimbolj razpršiti, da posamezne razrede, ki v našem primeru predstavljajo posamezne osebe, med seboj ločimo. Razpršenost med podatki pa se lahko pojavi tako med razredi (angl. *between-class scatter*), kot tudi znotraj posameznega razreda (angl. *within-class scatter*). Razpršenost, ki se pojavi med razredi, je uporabna za samo klasifikacijo. Razpršenost znotraj posameznega razreda pa je pri klasifikaciji nezaželena. Veliko variacije med posameznimi slikami se pojavi ravno zaradi sprememb v osvetlitvi [71]. Če izvedemo metodo PCA nad takimi slikami, bo projekcijska matrika $\mathbf{W}$ vsebovala glavne vektorje (t.i. lastne obraze), ki bodo v projiciranem obraznem prostoru ohranjali večino variacije predvsem zaradi osvetlitve. Posledično to pomeni, da točke v takšnem prostoru ne bodo dobro klasificirane, ali še slabše, da bodo razredi med seboj pomešani.

Predlagan je bil način zmanjševanja odvisnosti zaradi različne osvetlitve tako, da zavržemo prvih nekaj glavnih komponent pri metodi PCA.

V primeru, da prve komponente vsebujejo variacijo zaradi osvetlitve, s takšnim načinom dosežemo boljšo klasifikacijo, kot v primeru če jih upoštevamo. Po drugi strani pa ne moremo zagotovo vedeti ali vsebujejo prve komponente res variacijo osvetlitve in jih lahko zaradi tega zanemarimo ali pa vsebujejo informacije, ki so za samo razločevanje lahko zelo pomembne. Kako vplivajo prve glavne komponente na samo uspešnost prepoznave bomo preverili v preizkusu metode PCA v poglavju 9.

Belhumeur et al. [4] je kot rešitev pomanjkljivosti metode PCA predlagal uporabo linearne diskriminantne analize (angl. *Linear Diskriminant Analysis*, LDA) imenovane tudi Fisherjeva linearna diskriminantna metoda (angl. *Fisher's linear diskriminant analysis*, FLD). Lastne obraze je v tem primeru poimenoval Fisherjevi obrazi (angl. *fisherfaces*). S pomočjo metode LDA lahko izberemo takšen podprostor, ki poleg porazdelitve podatkov upošteva tudi porazdelitev znotraj razredov in porazdelitev med razredi. Porazdelitev med razredi mora biti namreč največja, da se posamezne razrede, v našem primeru osebe, med seboj kar se da dobro loči.

6.3.1 Matematična definicija

Predpostavimo, da imamo množico N M-dimenzionalnih vhodnih podatkov $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$. Od tega jih je N_1 v podmnožici X_1, ki pripadajo razredu c_1 in N_2 v podmnožici X_2, ki pripadajo drugemu razredu c_2. S pomočjo skalarnega produkta

$$y = \mathbf{w}^\top \mathbf{x} \tag{6.54}$$

tvorimo linearno kombinacijo komponent $\mathbf{x}$ in dobimo N primerov $y_1, y_2, \ldots, y_N$ razdeljenih v dve podmnožici Y_1 in Y_2. Če velja $||\mathbf{w}|| = 1$, potem je posamezna točka y_i projekcija pripadajočega vhodnega vektorja $\mathbf{x}_i$ na črto v smeri $\mathbf{w}$. Velikost vektorja $\mathbf{w}$ pravzaprav ni toliko pomembna, ker določa samo velikost projekcije. Večjega pomena je sama smer vektorja $\mathbf{w}$. Če si predstavljamo, da vhodni podatki, ki pripadajo razredu c_1, tvorijo gručo in podatki razreda c_2 pripadajo drugi gruči, potem si želimo, da projicirane podatke posameznega razreda med seboj kar najbolje ločimo. Slika 6.13 prikazuje izbiro dveh različnih vrednosti smernega vektorja $\mathbf{w}$ v primeru dvodimenzionalnih podatkov.

Cilj metode LDA je poiskati takšen smerni vektor $\mathbf{w}$, ki bo zagotovil najboljšo klasifikacijo. Merilo za ločevanje med projiciranimi podatki je razlika njihovih povprečnih vrednostih. Če določimo vektor $\mathbf{m}_i$, ki predstavlja M-dimenzionalno povprečno vrednost vhodnih podatkov posameznega razreda kot:

$$\mathbf{m}_i = \frac{1}{N_i} \sum_{\mathbf{x} \in X_i} \mathbf{x} \ , \tag{6.55}$$

potem lahko povprečno vrednost projiciranih podatkov določimo kot:

$$\tilde{m}_i = \frac{1}{N_i} \sum_{y \in Y_i} y = \tag{6.56}$$

$$= \frac{1}{N_i} \sum_{\mathbf{x} \in X_i} \mathbf{w}^\top \mathbf{x} = \mathbf{w}^\top \mathbf{m}_i \tag{6.57}$$

in vidimo, da je to kar projekcija same povprečne vrednosti $\mathbf{m}_i$.

Iz tega sledi, da je razdalja med projiciranimi povprečnimi vrednostmi enaka:

$$|\tilde{m}_1 - \tilde{m}_2| = |\mathbf{w}^\top (\mathbf{m}_1 - \mathbf{m}_2)| \tag{6.58}$$

in da lahko naredimo razliko večjo, če povečamo velikost vektorja $\mathbf{w}$. Za dobro ločevanje projiciranih podatkov pa želimo, da je razlika projiciranih povprečnih vrednosti relativno večja glede na standardno deviacijo posameznega razreda. Namesto variance vseh podatkov definirajmo raje porazdelitev projiciranih podatkov za posamezen razred c_i kot:

$$\tilde{s}_i^2 = \sum_{y \in Y_i} (y - \tilde{m}_i)^2 \ . \tag{6.59}$$

Tako lahko določimo varianco vseh podatkov kot $(\frac{1}{N})(\tilde{s}_1^2 + \tilde{s}_2^2)$, kjer $\tilde{s}_1^2 + \tilde{s}_2^2$ določa celotno porazdelitev znotraj razredov projiciranih podatkov (angl. *total within-class scatter*). Pri metodi LDA uporabimo takšno linearno funkcijo $\mathbf{w}^\top \mathbf{x}$ pri kateri je kriterijska funkcija

$$J(\mathbf{w}) = \frac{|\tilde{m}_1 - \tilde{m}_2|^2}{\tilde{s}_1^2 + \tilde{s}_2^2} \tag{6.60}$$

največja in neodvisna od same velikosti $||\mathbf{w}||$. S tem ko vektor $\mathbf{w}$ maksimizira funkcijo $J(\cdot)$ določa najboljšo možno razdelitev med dvema projiciranima množicama. Za določitev klasifikatorja potrebujemo samo še mejno vrednost, ki najbolje razmeji posamezna razreda med seboj.

Da določimo $J(\cdot)$ kot eksplicitno funkcijo vrednosti $\mathbf{w}$, definirajmo matrike razpršenosti podatkov $\mathbf{S}_i$ in matriko $\mathbf{S}_W$ kot:

$$\mathbf{S}_i = \sum_{\mathbf{x} \in X_i} (\mathbf{x} - \mathbf{m}_i)(\mathbf{x} - \mathbf{m}_i)^\top \tag{6.61}$$

in

$$\mathbf{S}_W = \mathbf{S}_1 + \mathbf{S}_2 \ . \tag{6.62}$$

Nato lahko zapišemo

$$\begin{aligned} \tilde{s}_i^2 &= \sum_{\mathbf{x} \in X_i} (\mathbf{w}^\top \mathbf{x} - \mathbf{w}^\top \mathbf{m}_i)^2 = \\ &= \sum_{\mathbf{x} \in X_i} \mathbf{w}^\top (\mathbf{x} - \mathbf{m}_i)(\mathbf{x} - \mathbf{m}_i)\mathbf{w} = \\ &= \mathbf{w}^\top \mathbf{S}_i \mathbf{w} \end{aligned} \tag{6.63}$$

in vsoto posameznih razpršenosti lahko potemtakem zapišemo kot:

$$\tilde{s}_1^2 + \tilde{s}_2^2 = \mathbf{w}^\top \mathbf{S}_W \mathbf{w} \ . \tag{6.64}$$

Podobno, za razmejitev projiciranih povprečij definiramo

$$(\tilde{m}_1 - \tilde{m}_2)^2 = (\mathbf{w}^\top \mathbf{m}_1 - \mathbf{w}^\top \mathbf{m}_2)^2 = \tag{6.65}$$

$$= \mathbf{w}^\top (\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^\top \mathbf{w} \tag{6.66}$$

$$= \mathbf{w}^\top \mathbf{S}_B \mathbf{w} \ , \tag{6.67}$$

kjer je

$$\mathbf{S}_B = (\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^\top \ . \tag{6.68}$$

Matriko $\mathbf{S}_W$ imenujemo matrika razpršenosti znotraj razreda (angl. *within-class scatter matrix*). Njena vrednost je sorazmerna kovariančni matriki vhodnih podatkov. Je simetrična, pozitivno semidefinitna in nesingularna, če velja $N > M$. Matriko $\mathbf{S}_B$ imenujemo matrika razpršenosti med razredi (angl. *between-class scatter matrix*) in je prav tako simetrična in pozitivno semidefinitna. Ker je definirana kot zunanji produkt dveh vektorjev je njen rang v primeru dveh razredov največ ena in je zaradi tega tudi singularna.

Glede na definicijo matrik $\mathbf{S}_B$ in $\mathbf{S}_W$ lahko funkcijo $J(\cdot)$ sedaj zapišemo kot:

$$J(\mathbf{w}) = \frac{\mathbf{w}^\top \mathbf{S}_B \mathbf{w}}{\mathbf{w}^\top \mathbf{S}_W \mathbf{w}} \ . \tag{6.69}$$

Vektor $\mathbf{w}$, ki maksimizira vrednost funkcije $J(\cdot)$ mora zadostiti enačbi:

$$\mathbf{S}_B\mathbf{w} = \lambda\mathbf{S}_W\mathbf{w} \ , \tag{6.70}$$

za določeno konstanto λ, kar je v bistvu generaliziran problem lastnih vrednosti. V primeru, če je matrika $\mathbf{S}_W$ nesingularna, lahko prevedemo generaliziran problem lastnih vrednosti v običajen problem lastnih vrednosti:

$$\mathbf{S}_W^{-1}\mathbf{S}_B\mathbf{w} = \lambda\mathbf{w} \ . \tag{6.71}$$

Zgoraj smo definirali rešitev linearne diskriminantne metode v primeru dveh razredov. Če pa imamo c različnih razredov, potem linearna diskriminantna analiza vsebuje $c-1$ diskriminantnih funkcij. Tako imamo projekcijo iz vhodnega N-dimenzijskega prostora v $(c-1)$-dimenzionalni prostor. Generalizacija razpršenosti znotraj posameznega razreda $\mathbf{S}_W$ je podobno kot v primeru dveh razredov enaka:

$$\mathbf{S}_W = \sum_{i=1}^{c} \mathbf{S}_i \ , \tag{6.72}$$

kjer je kot prej

$$\mathbf{S}_i = \sum_{\mathbf{x}\in X_i} (\mathbf{x}-\mathbf{m}_i)(\mathbf{x}-\mathbf{m}_i)^\top \tag{6.73}$$

in

$$\mathbf{m}_i = \frac{1}{N_i}\sum_{\mathbf{x}\in X_i} \mathbf{x} \ . \tag{6.74}$$

Definicija generalizacije za matriko $\mathbf{S}_B$ ni tako očitna. Definirajmo najprej vektor povprečne vrednosti $\mathbf{m}$ in matriko razpršenosti $\mathbf{S}_T$, za celotne podatke:

$$\mathbf{m} = \frac{1}{N}\sum_{\mathbf{x}} \mathbf{x} = \frac{1}{N}\sum_{i=1}^{c} N_i\mathbf{m}_i \tag{6.75}$$

$$\mathbf{S}_T = \sum_{\mathbf{x}} (\mathbf{x}-\mathbf{m})(\mathbf{x}-\mathbf{m})^\top \ . \tag{6.76}$$

Iz tega sledi

$$\begin{aligned}
\mathbf{S}_T &= \sum_{i=1}^{c}\sum_{\mathbf{x}\in X_i} (\mathbf{x}-\mathbf{m}_i+\mathbf{m}_i-\mathbf{m})(\mathbf{x}-\mathbf{m}_i+\mathbf{m}_i-\mathbf{m})^\top = \\
&= \sum_{i=1}^{c}\sum_{\mathbf{x}\in X_i} (\mathbf{x}-\mathbf{m}_i)(\mathbf{x}-\mathbf{m}_i)^\top + \sum_{i=1}^{c}\sum_{\mathbf{x}\in X_i} (\mathbf{m}_i-\mathbf{m})(\mathbf{m}_i-\mathbf{m})^\top \\
&= \mathbf{S}_W + \sum_{i=1}^{c} N_i(\mathbf{m}_i-\mathbf{m})(\mathbf{m}_i-\mathbf{m})^\top \ .
\end{aligned} \tag{6.77}$$

Naravno je, da definiramo drugi del enačbe 6.77 kot generalno matriko razpršenosti med posameznimi razredi $\mathbf{S}_B$ tako, da je celotna razpršenost podatkov enaka vsoti matrik razpršenosti znotraj razredov in razpršenosti med razredi:

$$\mathbf{S}_B = \sum_{i=1}^{c} N_i(\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^\top \tag{6.78}$$

in

$$\mathbf{S}_T = \mathbf{S}_W + \mathbf{S}_B \ . \tag{6.79}$$

Če primerjamo rešitev za razpršenost med razredi $\mathbf{S}_B$ z rešitvijo v primeru dveh razredov (enačba 6.68) lahko vidimo, da je vrednost tukaj večja za N_1N_2/N, kar pa ne vpliva na končno rešitev in ne spremeni naše definicije.

Projekcijo iz N-dimenzijskega prostora v $(c-1)$-dimenzionalni prostor določimo s pomočjo $c-1$ diskriminantnih funkcij

$$y_i = \mathbf{w}_i^\top \mathbf{x} \ , \qquad i = 1, 2, \ldots, c-1 \ . \tag{6.80}$$

Če gledamo na vrednosti y_i kot na komponente vektorja $\mathbf{y}$ in na vektorje uteži $\mathbf{w}_i$ kot na posamezne stolpce matrike $\mathbf{W} \in \mathbb{R}^{N\times(c-1)}$, potem lahko projekcijo zapišemo tudi kot:

$$\mathbf{y} = \mathbf{W}^\top \mathbf{x} \ . \tag{6.81}$$

Posamezni primeri $x_1, x_2, \ldots, x_N$ se projicirajo v pripadajočo množico $y_1, y_2, \ldots, y_N$, ki jo lahko opišemo s povprečnim vektorjem in matrikama razpršenosti kot:

$$\tilde{\mathbf{m}}_i = \frac{1}{N_i} \sum_{\mathbf{y}\in Y_i} \mathbf{y} \tag{6.82}$$

$$\tilde{\mathbf{m}} = \frac{1}{N} \sum_{i=1}^{c} N_i \tilde{\mathbf{m}}_i \tag{6.83}$$

$$\tilde{\mathbf{S}}_W = \sum_{i=1}^{c} \sum_{\mathbf{y}\in Y_i} (\mathbf{y} - \tilde{\mathbf{m}}_i)(\mathbf{y} - \tilde{\mathbf{m}}_i)^\top \tag{6.84}$$

$$\tilde{\mathbf{S}}_B = \sum_{i=1}^{c} N_i (\tilde{\mathbf{m}}_i - \tilde{\mathbf{m}})(\tilde{\mathbf{m}}_i - \tilde{\mathbf{m}})^\top \ . \tag{6.85}$$

Iz tega sledi, da velja:

$$\tilde{\mathbf{S}}_W = \mathbf{W}^\top \mathbf{S}_W \mathbf{W} \tag{6.86}$$

$$\tilde{\mathbf{S}}_B = \mathbf{W}^\top \mathbf{S}_B \mathbf{W} \ . \tag{6.87}$$

Ti dve enačbi nam določata transformacijo matrike razpršenosti znotraj razredov in matrike razpršenosti med razredi v nižje dimenzijski prostor. Mi pa iščemo takšno

transformacijsko matrika $\mathbf{W}$, ki bo maksimizirala razmerje med razpršenostjo med razredi in razpršenostjo znotraj razredov. Enostavno skalarno merilo razpršenosti je determinanta matrike razpršenosti. Determinanta je v bistvu produkt lastnih vrednosti in v tem primeru je to kar produkt variance glavnih smeri. Z uporabo tega merila, lahko zapišemo kriterijsko funkcijo $J(\mathbf{W})$ kot:

$$J(\mathbf{W}) = \frac{|\tilde{\mathbf{S}}_B|}{|\tilde{\mathbf{S}}_W|} = \frac{|\mathbf{W}^\top \mathbf{S}_B \mathbf{W}|}{|\mathbf{W}^\top \mathbf{S}_W \mathbf{W}|} \ . \tag{6.88}$$

Optimalna matrika $\mathbf{W}$, ki maksimizira funkcijo $J(\cdot)$ ima v svojih stolpcih generalizirane lastne vektorje, ki pripadajo največjim lastnim vrednostim:

$$\mathbf{S}_B \mathbf{w}_i = \lambda_i \mathbf{S}_W \mathbf{w}_i \ . \tag{6.89}$$

Obstaja kar nekaj rešitev generaliziranega problema lastnih vrednosti. V primeru, če je matrika $\mathbf{S}_W$ nesingularna, lahko problem, enako kot prej, prevedemo v običajni problem lastnih vrednosti, tako da izračunamo inverz matrike $\mathbf{S}_W$. Poleg tega pa lahko izračunamo lastne vrednosti tako, da določimo ničle karakterističnemu polinomu

$$|\mathbf{S}_B - \lambda_i \mathbf{S}_W| = 0 \tag{6.90}$$

in potem izračunamo

$$(\mathbf{S}_B - \lambda_i \mathbf{S}_W)\mathbf{w}_i = 0 \tag{6.91}$$

neposredno za lastni vektor $\mathbf{w}_i$.

Ker je matrika $\mathbf{S}_B$ vsota c matrik ranga ena ali manj, ter ker jih je med njimi samo $c-1$ neodvisnih, ima matrika $\mathbf{S}_B$ rang največ $c-1$. Tako ne obstaja več kot $c-1$ neničelnih lastnih vrednosti in njim pripadajočih lastnih vektorjev. V posebnem primeru, ko je matrika $\mathbf{S}_W$ isotropična, pa lastni vektorji pripadajo predvsem matriki $\mathbf{S}_B$ in jih lahko enostavno določimo s pomočjo ortogonalizacije Gram-Schmidt. V splošnem pa takšna rešitev ni unikatna.

6.3.2 Enostaven 2D primer

Poskusimo sedaj na primeru dvodimenzionalnih podatkov izračunati glavne vektorje s pomočjo metode LDA in pokazati njene prednosti pred metodo PCA. Vhodni podatki $\mathbf{X} \in \mathbb{R}^{2\times 10}$ pripadajo dvema različnima razredoma in so razporejeni tako, da ležijo blizu linearnega podprostora. Za naš primer torej velja, da je število vseh primerov enako $N = 10$, dimenzija podatkov je $M = 2$, število razredov je $c = 2$, največja možna dimenzija podprostora je $k = c - 1 = 1$ in število primerov znotraj posameznega razreda je $N_i = 5$. Podatke $\mathbf{X}$ razdelimo tako, da jih prvih pet pripada prvemu razredu, zadnjih pet pa drugemu.

$$\mathbf{X} = \begin{bmatrix} 9 & 2 & 2 & 3 & 6 & 9 & 8 & 10 & 4 & 4 \\ 10 & 4 & 3 & 6 & 8 & 5 & 7 & 8 & 1 & 4 \end{bmatrix} \tag{6.92}$$

Najprej izračunamo vektor $\boldsymbol{\mu}$ povprečnih vrednosti celotnih podatkov ter povprečni vrednosti $\boldsymbol{\mu}_1$ in $\boldsymbol{\mu}_2$ posameznega razreda:

$$\boldsymbol{\mu}^\top = \begin{bmatrix}5,7 & 5,6\end{bmatrix} \tag{6.93}$$

$$\boldsymbol{\mu}_1^\top = \begin{bmatrix}4,4 & 6,2\end{bmatrix} \tag{6.94}$$

$$\boldsymbol{\mu}_2^\top = \begin{bmatrix}7 & 5\end{bmatrix} \tag{6.95}$$

Na podlagi izračunanih povprečnih vrednosti lahko izračunamo matriko $\mathbf{S}_B$ (6.78), ki določa porazdelitev podatkov med razredi.

$$\begin{aligned}\mathbf{S}_B &= \sum_{i=1}^{2} N_i(\boldsymbol{\mu}_i - \boldsymbol{\mu})(\boldsymbol{\mu}_i - \boldsymbol{\mu})^\top = \\ &= 5\begin{bmatrix}1,69 & -0,78\\ -0,78 & 0,36\end{bmatrix} + 5\begin{bmatrix}1,69 & -0,78\\ -0,78 & 0,36\end{bmatrix} = \\ &= \begin{bmatrix}16,9 & -7,8\\ -7,8 & 3,6\end{bmatrix}\end{aligned} \tag{6.96}$$

Za izračun matrike $\mathbf{S}_W$ pa moramo izračunati za posamezen razred kovariančno matriko, ki ju označimo s $\mathbf{S}_{W1}$ in $\mathbf{S}_{W2}$.

$$\mathbf{S}_{W1} = (\mathbf{X_1} - \boldsymbol{\mu}_1)(\mathbf{X_1} - \boldsymbol{\mu}_1)^\top = \begin{bmatrix}37,2 & 33,6\\ 33,6 & 32,8\end{bmatrix} \tag{6.97}$$

$$\mathbf{S}_{W2} = (\mathbf{X_2} - \boldsymbol{\mu}_2)(\mathbf{X_2} - \boldsymbol{\mu}_2)^\top = \begin{bmatrix}32 & 26\\ 26 & 30\end{bmatrix} \tag{6.98}$$

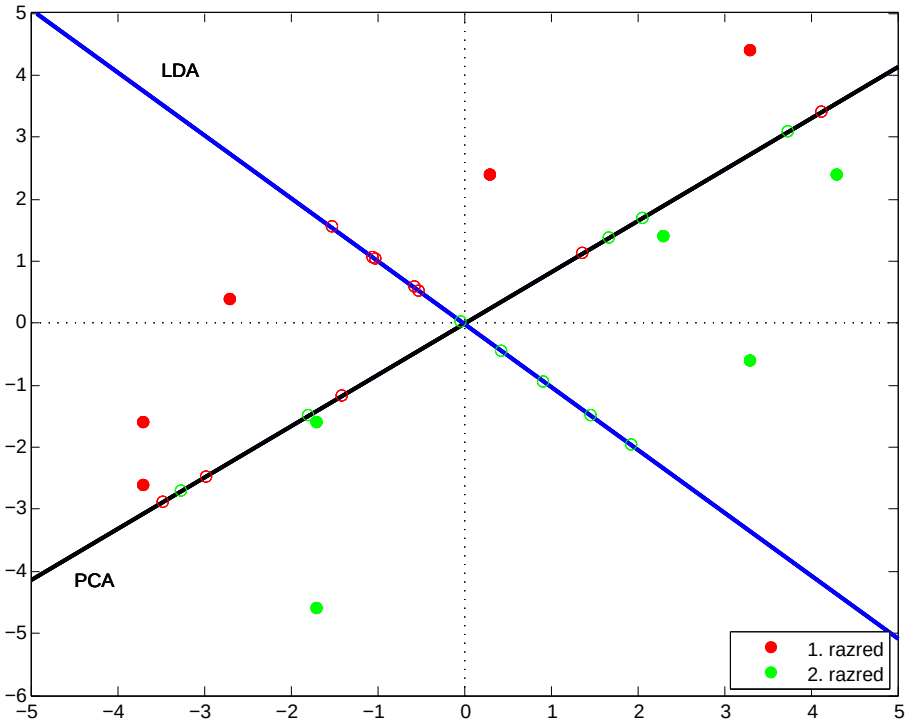
Matriko $\mathbf{S}_W$ (6.72), ki določa porazdelitev podatkov znotraj posameznega razreda, dobimo tako, da seštejemo vse kovariančne matrike razredov.

$$\mathbf{S}_W = \sum_{i=1}^{2} \mathbf{S}_{Wi} = \mathbf{S}_{W1} + \mathbf{S}_{W2} = \begin{bmatrix}69,2 & 59,6\\ 59,6 & 62,8\end{bmatrix} \tag{6.99}$$

Ko imamo izračunani obe matriki $\mathbf{S}_W$ in $\mathbf{S}_B$ dobimo LDA projekcijo kot rešitev generaliziranega problema lastnih vrednosti (6.89) ali kot rešitev lastnih vektorjev in lastnih vrednosti matrike $\mathbf{S}_W^{-1}\mathbf{S}_B$. Ker imamo v našem primeru samo dva razreda ($c = 2$) je največje število lastnih vrednosti enako $k = c - 1 = 1$. V našem primeru sta izračunani lastna vrednost λ_1 in lastni vektor $\mathbf{w_1}$ enaka:

$$\begin{aligned}\lambda_1 &= 2,82\\ \mathbf{w}_1^\top &= [0,70, -0,71]\end{aligned}$$

Slika 6.13 prikazuje, za primer 2D podatkov, prvo komponento izračunano po metodi PCA in glavni vektor $\mathbf{w_1}$, izračunan s pomočjo metode LDA. Z rdečo so



Slika 6.13: Delovanje metode LDA za primer 2D podatkov.

označeni podatki, ki pripadajo prvemu razredu, z zeleno pa podatki drugega razreda. Posamezna glavna vektorja metod PCA in LDA pa sta predstavljena s črno oziroma modro barvo. Vidimo lahko, da glavni vektor metode PCA določa smer, kjer je varianca podatkov največja, pri tem pa ne upošteva kateremu razredu pripada kakšen podatek. Zaradi tega se podatki posameznega razreda med seboj prekrivajo. S pomočjo metode LDA pa smo izračunali takšen vektor, ki upošteva tudi pripadnost razredom in vidimo lahko, da podatke posameznega razreda med seboj loči.

6.3.3 Uporaba metode LDA za prepoznavanje obrazov

Obstaja kar nekaj načinov za izračun generaliziranega lastnega problema (6.89). Najenostavnejši je, da izračunamo inverz matrike $\mathbf{S}_W$ in nato izračunamo, v splošnem nesimetričen, lastni problem za matriko $\mathbf{S}_W^{-1}\mathbf{S}_B$. Ampak takšen način je tudi numerično nestabilen, ker vključuje inverz zelo velike matrike $(M \times M)$, ki je po vsej verjetnosti singularna. Opisan postopek izračuna po metodi LDA v primeru, ko je matrika $\mathbf{S}_W$ nesingularna opisuje algoritem 5.

Drugi način, da rešimo simetrični lastni problem, je s pomočjo dekompozicije Cholesky ali tako, da diagonaliziramo obe simetrični matriki $\mathbf{S}_W$ in $\mathbf{S}_B$ in na takšen način dobimo numerično stabilen postopek za izračun lastnega sistema [93].

Najprej izračunamo lastni problem za matriko $\mathbf{S}_W$. Ker je matrika $\mathbf{S}_W$ simetrična, obstajata ortogonalna matrika $\mathbf{H}$ in diagonalna matrika $\mathbf{\Lambda}$, da velja $\mathbf{S}_W =$

Algoritem 5 : Linearna diskriminantna metoda – LDA

Vhod: učna množica $\mathbf{X}$ z c razredi: $\mathbf{X} = \{\mathbf{X}_i\}_{i=1}^{c}$, vsak razred vsebuje $X_i = \{x_{ij}\}_{j=1}^{N_i}$ slik obrazov, kjer je $x_{ij} \in \mathbb{R}^M$.

Izhod: k-dimenzijski LDA podprostor, ki ga določa transformacijska matrika $\mathbf{W} \in \mathbb{R}^{M\times k}$.

1: Določi vektor povprečnih vrednosti: $\boldsymbol{\mu} = \frac{1}{N}\sum_{j=1}^{N}\mathbf{x}_j$.

2: Določi vektor povprečij za posamezen razred c_i: $\boldsymbol{\mu}_i = \frac{1}{N_i}\sum_{j=1}^{N_i}\mathbf{x}_{ij}$.

3: Izračunaj matriko razpršenosti znotraj posameznega razreda: $\mathbf{S}_W = \sum_{i=1}^{c}\sum_{j=1}^{N_i}(\mathbf{x}_{ij} - \boldsymbol{\mu}_i)(\mathbf{x}_{ij} - \boldsymbol{\mu}_i)^\top$

4: Izračunaj matriko razpršenosti med posameznimi razredi: $\mathbf{S}_B = \sum_{i=1}^{c} N_i(\boldsymbol{\mu}_i - \boldsymbol{\mu})(\boldsymbol{\mu}_i - \boldsymbol{\mu})^\top$

5: Določi kriterijsko funkcijo $J(\mathbf{W}) = \frac{|\mathbf{W}^\top\mathbf{S}_B\mathbf{W}|}{|\mathbf{W}^\top\mathbf{S}_W\mathbf{W}|}$

6: Določi maksimum funkcije $J(\mathbf{W})$ z izračunom generaliziranega lastnega problema: $\mathbf{S}_B\mathbf{w}_i = \lambda_i\mathbf{S}_W\mathbf{w}_i$

7: Generaliziran lastni problem prevedemo v običajni problem lastnih vrednosti matrike $\mathbf{S}_W^{-1}\mathbf{S}_B$: $\mathbf{S}_W^{-1}\mathbf{S}_B\mathbf{w}_i = \lambda_i\mathbf{w}_i$

8: Dobimo lastne vektorje $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_M]$ in lastne vrednosti $\boldsymbol{\lambda}$.

9: Izberi $k \leq c-1$ lastnih vektorjev, katerim pripadajo neničelne lastne vrednosti: $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_k]$.

$\mathbf{H}\boldsymbol{\Lambda}\mathbf{H}^\top$. Potem velja tudi $(\mathbf{H}\boldsymbol{\Lambda}^{-\frac{1}{2}})^\top\mathbf{S}_W\mathbf{H}\boldsymbol{\Lambda}^{-\frac{1}{2}} = \mathbf{I}$. Nato izračunamo ortogonalno matriko $\mathbf{U}$ in diagonalno matriko $\boldsymbol{\Sigma}$ tako, da je $(\mathbf{H}\boldsymbol{\Lambda}^{-\frac{1}{2}})^\top\mathbf{S}_B\mathbf{H}\boldsymbol{\Lambda}^{-\frac{1}{2}} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{U}^\top$. Tako imamo:

$$\mathbf{S}_B = \mathbf{H}\boldsymbol{\Lambda}^{\frac{1}{2}}\mathbf{U}\boldsymbol{\Sigma}\mathbf{U}^\top\boldsymbol{\Lambda}^{\frac{1}{2}}\mathbf{H}^\top \quad \text{in} \tag{6.100}$$

$$\mathbf{S}_W = \mathbf{H}\boldsymbol{\Lambda}^{\frac{1}{2}}\mathbf{U}\mathbf{I}\mathbf{U}^\top\boldsymbol{\Lambda}^{\frac{1}{2}}\mathbf{H}^\top \ . \tag{6.101}$$

Definirajmo $\boldsymbol{\nabla} = \mathbf{H}\boldsymbol{\Lambda}^{-\frac{1}{2}}\mathbf{U}$, kjer $\boldsymbol{\nabla}$ istočasno diagonalizira $\mathbf{S}_B$ in $\mathbf{S}_W$. Ker velja:

$$\mathbf{S}_W^{-1} = \mathbf{H}\boldsymbol{\Lambda}^{-1}\mathbf{H}^\top \ , \tag{6.102}$$

lahko iz enačb 6.100 in 6.102 dobimo:

$$\mathbf{S}_W^{-1}\mathbf{S}_B = \mathbf{H}\boldsymbol{\Lambda}^{-1}\mathbf{H}^\top\mathbf{H}\boldsymbol{\Lambda}^{\frac{1}{2}}\mathbf{U}\boldsymbol{\Sigma}\mathbf{U}^\top\boldsymbol{\Lambda}^{\frac{1}{2}}\mathbf{H}^\top = \tag{6.103}$$

$$= \mathbf{H}\boldsymbol{\Lambda}^{-\frac{1}{2}}\mathbf{U}\boldsymbol{\Sigma}\mathbf{U}^\top\boldsymbol{\Lambda}^{\frac{1}{2}}\mathbf{H}^\top = \tag{6.104}$$

$$= \boldsymbol{\nabla}\boldsymbol{\Sigma}\boldsymbol{\nabla}^{-1} \ . \tag{6.105}$$

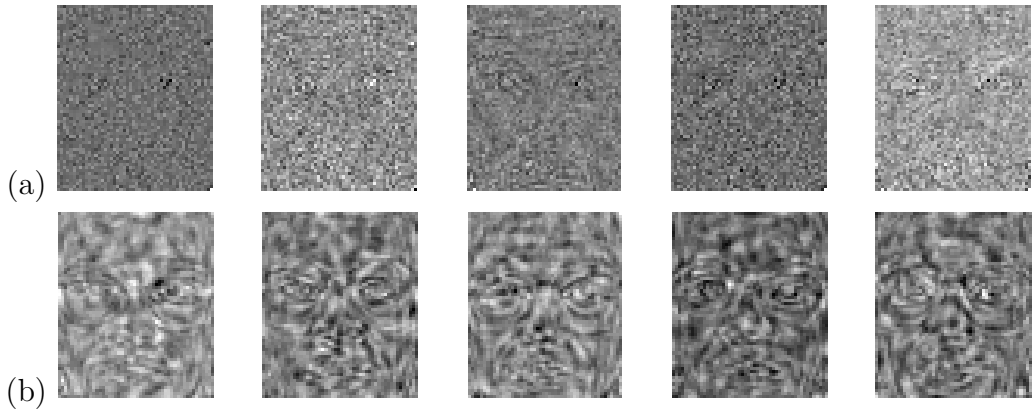
Tako vsebuje matrika $\boldsymbol{\nabla}$ po stolpcih lastne vektorje matrike $\mathbf{S}_W^{-1}\mathbf{S}_B$ in matrika $\boldsymbol{\Sigma}$ vsebuje na diagonali njene lastne vrednosti. Tako smo z upoštevanjem simetričnih lastnosti matrik razpršitve izpeljali metodo za iskanje lastnih vektorjev in lastnih vrednosti matrike $\mathbf{S}_W^{-1}\mathbf{S}_B$, ki je stabilna.

Pri aplikacijah, ki operirajo s slikami, je značilno, da je število učnih slik N veliko manjše od števila slikovnih elementov slike oziroma njene dimenzije prostora

M. Med te aplikacije spada tudi aplikacija prepoznavanja obrazov, kjer je vhodni podatek slika obraza. V takem primeru je matrika $\mathbf{S}_W$ zelo velika, reda $M \times M$ in običajno singularna zaradi napake zaokroževanja (angl. *round-off error*). Da lahko rešimo takšen numeričen problem, predlaga avtor Zhao v svojem delu [113], da matriko $\mathbf{S}_W$ spremenimo tako, da ji dodamo $\mathbf{S}_W + \Delta\mathbf{I}$, kjer Δ označuje zelo majhno pozitivno vrednost. Matriko spreminjamo toliko časa, da le ta postane nesingularna.

Avtorji Belhumeur, Swets in Zhao [4, 93, 113] pa so predlagali, da namesto ene projekcije s pomočjo metode LDA naredimo dve projekciji. S pomočjo metode PCA najprej zmanjšamo dimenzijo vhodnega prostora in nato v tako dobljenem prostoru izvršimo še metodo LDA. S takšnim postopkom rešimo poleg problema singularnosti matrike $\mathbf{S}_W$ tudi problem generalizacije oziroma problem prevelikega prilagajanja (angl. *over-fitting*). Sama metoda LDA se namreč zelo prilagodi učnim primerom, katere hoče kar se da dobro razmejiti. Iz tega sledi, da je lahko metoda zelo uspešna na učnih primerih. V primerih, ko imamo slike obrazov, ki niso iz učne množice ali pa so drugačne od učnih (različno ozadje, različna osvetlitev itd.), pa je uspešnost slabša. Tako kombinirano metodo LDA skupaj z metodo PCA je prvi predlagal Belhumeur [4], ki je tako dobljene diskriminantne vektorje poimenoval Fisherjevi obrazi (angl. *Fisherfaces*). Avtorja Swets in Weng [93] pa sta enako metodo poimenovala metoda najbolj diskriminantnih značilk (angl. *Most Discriminant Features*, MDF).

Slika 6.14 prikazuje prvih pet glavnih vektorjev, ki smo jih dobil samo s pomočjo metode LDA in prvih pet glavnih vektorjev (t.i. fisherjevih obrazov), ki smo jih dobil s pomočjo kombinacije metod PCA in LDA. Tudi iz slike lahko vidimo da predstavlja kombinacija metod reprezentativnejšo različico same metode LDA in zaradi tega omogoča boljšo generalizacijo naučenih in nenaučenih primerov podatkov.



Slika 6.14: Primeri prvih petih lastnih obrazov (a) samo metode LDA in (b) kombinacije metod PCA in LDA ali tako imenovanih Fisherjevih obrazov.

Postopek tako kombinirane metode LDA podrobneje opisuje algoritem 6.

Metodo Fisherjevih obrazov sestavljata dve tehniki zmanjševanja dimenzije. Vhodno sliko dimenzije M najprej preslikamo v k-dimenzionalni prostor, ki ga določa metoda PCA. Velikost prostora k izberemo tako, da je večji od števila razredov c

Algoritem 6 : Metoda Fisherjevih obrazov – LDA+PCA

Vhod: učna množica $\mathbf{X}$ z c razredi: $\mathbf{X} = \{\mathbf{X}_i\}_{i=1}^{c}$, vsak razred vsebuje $X_i = \{x_{ij}\}_{j=1}^{N_i}$ slik obrazov, kjer je $x_{ij} \in \mathbb{R}^M$.

Izhod: k-dimenzijski LDA podprostor, ki ga določa transformacijska matrika $\mathbf{W} \in \mathbb{R}^{M\times k}$.

1: S pomočjo metode PCA izračunaj matriko lastnih vektorjev $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2]$, ki pripadajo neničelnim lastnim vrednostim in povprečni vektor $\boldsymbol{\mu}$.

2: Določi transformacijsko matriko $\mathbf{W}_{pca} = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_m]$, kjer je število glavnih komponent m večje od c in manjše od $N - c$.

3: Transformiraj vhodne podatke $\mathbf{X}$ s pomočjo transformacijske matrike $\mathbf{W}_{pca}$ v PCA podprostor: $\mathbf{X}' = \mathbf{W}_{pca}^{\top}\mathbf{X}$

4: Določi vektor povprečnih vrednosti: $\boldsymbol{\mu}' = \frac{1}{N}\sum_{j=1}^{N}\mathbf{x}'_j$.

5: Določi vektor povprečji za posamezen razred c_i: $\boldsymbol{\mu}'_i = \frac{1}{N_i}\sum_{j=1}^{N_i}\mathbf{x}'_{ij}$.

6: Izračunaj matriko razpršenosti znotraj posameznega razreda: $\mathbf{S}'_W = \sum_{i=1}^{c}\sum_{j=1}^{N_i}(\mathbf{x}'_{ij} - \boldsymbol{\mu}'_i)(\mathbf{x}'_{ij} - \boldsymbol{\mu}'_i)^{\top}$

7: Izračunaj matriko razpršenosti med posameznimi razredi: $\mathbf{S}'_B = \sum_{i=1}^{c} N_i(\boldsymbol{\mu}'_i - \boldsymbol{\mu}')(\boldsymbol{\mu}'_i - \boldsymbol{\mu}')^{\top}$

8: Določi kriterijsko funkcijo $J(\mathbf{W}) = \frac{|\mathbf{W}^{\top}\mathbf{S}'_B\mathbf{W}|}{|\mathbf{W}^{\top}\mathbf{S}'_W\mathbf{W}|} = \frac{|\mathbf{W}^{\top}\mathbf{W}_{pca}^{\top}\mathbf{S}_B\mathbf{W}_{pca}\mathbf{W}|}{|\mathbf{W}^{\top}\mathbf{W}_{pca}^{\top}\mathbf{S}_W\mathbf{W}_{pca}\mathbf{W}|}$

9: Določi maksimum funkcije $J(\mathbf{W})$ z izračunom problema lastnih vrednosti matrike $\mathbf{S}'^{-1}_W\mathbf{S}'_B$: $\mathbf{S}'^{-1}_W\mathbf{S}'_B\mathbf{w}_i = \lambda_i\mathbf{w}_i$

10: Dobimo lastne vektorje $\mathbf{W}_{lda} = [\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_m]$ in lastne vrednosti $\boldsymbol{\lambda}$.

11: Izberi $k \leq c - 1 \leq m$ lastnih vektorjev, katerim pripadajo neničelne lastne vrednosti: $\mathbf{W}_{lda} = [\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_k]$.

12: Določi transformacijsko matriko $\mathbf{W} = \mathbf{W}_{pca}\mathbf{W}_{lda}$.

ter manjši od ranga matrike S_W, ki je $N - c$, kjer N označuje število vseh primerov. Nato pa preslikamo sliko iz prostora PCA s pomočjo metode LDA v nov prostor, ki ima največjo dimenzijo $c - 1$.

Formalno lahko transformacijsko matriko $\mathbf{W}$ zapišemo kot:

$$\mathbf{W} = \mathbf{W}_{pca}\mathbf{W}_{lda} \ , \tag{6.106}$$

kjer je

$$\mathbf{W}_{pca} = \arg\max_{\mathbf{W}} |\mathbf{W}^{\top}\mathbf{S}_T\mathbf{W}| \tag{6.107}$$

$$\mathbf{W}_{lda} = \arg\max_{\mathbf{W}} \frac{|\mathbf{W}^{\top}\mathbf{W}_{pca}^{\top}\mathbf{S}_B\mathbf{W}_{pca}\mathbf{W}|}{|\mathbf{W}^{\top}\mathbf{W}_{pca}^{\top}\mathbf{S}_W\mathbf{W}_{pca}\mathbf{W}|} \ . \tag{6.108}$$

S tem postopkom odpravimo singularnost razpršene matrike $\mathbf{S}_W$ in opazimo lahko, da zavržemo v primeru, ko ohranimo največje število komponent, to je $N - c$, samo c najmanjših komponent metode PCA.

Avtorji Chen, Yu in Lu [14, 111, 42] pa trdijo, da vsebuje teh c lastnih vektorjev, ki jih završemo, pomembne diskriminantne informacije, zato predlagajo uporabo direktne metode LDA (angl. *direct LDA*).

Chen et al. [14] predlaga metodo, ki jo bomo tukaj imenovali CLDA. Njegova direktna metoda LDA odpravi problem singularnosti, ki nastane pri aplikacijah, ki imajo majhno število učnih primerov, torej je primerna tudi za aplikacije prepoznavanja obrazov.

Glavna ideja metode CLDA je uporabiti diskriminantno informacijo nultega prostora (angl. *null space*) matrike razpršenosti $\mathbf{S}_W$ za namene maksimizacije razpršenosti med razredi $\mathbf{S}_B$. V primeru nesingularnosti matrike $\mathbf{S}_W$ pa uporabiti lastne vektorje, ki pripadajo največjim lastnim vrednostim matrike $(\mathbf{S}_B + \mathbf{S}_W)^{-1}\mathbf{S}_B$. Fukunaga [24] je namreč dokazal, da pripadajo matrikam $(\mathbf{S}_B + \mathbf{S}_W)^{-1}\mathbf{S}_B$ in $\mathbf{S}_W^{-1}\mathbf{S}_B$ enaki lastni vektorji. Algoritem CLDA za izračun projekcijske matrike $\mathbf{W}_{clda}$ lahko opišemo v sledečih točkah:

1. Izračunaj rang r matrike razpršitve znotraj razredov $\mathbf{S}_W$,
2. če je $\mathbf{S}_W$ nesingularna (t.j. $r = M$), potem matriko $\mathbf{W}_{clda}$ določajo lastni vektorji, ki pripadajo največjim lastnim vrednostim matrike $(\mathbf{S}_B + \mathbf{S}_W)^{-1}\mathbf{S}_B$,
3. drugače, če je $\mathbf{S}_W$ singularna, izračunaj lastne vektorje $\mathbf{V} = [v_1, \ldots, v_r, v_{r+1}, \ldots, v_M]$ singularne matrike $\mathbf{S}_W$. Naj matrika $\mathbf{Q}$ določa nulti prostor, tako da je $\mathbf{Q} = [v_{r+1}, v_{r+2}, \ldots, v_M]$ podmatrika dimenzije $M \times (M - r)$ matrike $\mathbf{V}$,
4. projekcijsko matriko $\mathbf{W}_{clda}$ tako določajo lastni vektorji, ki pripadajo največjim lastnim vrednostim matrike $\mathbf{Q}\mathbf{Q}^\top\mathbf{S}_B(\mathbf{Q}\mathbf{Q}^\top)^\top$.

Chen je dokazal, da so tako definirani lastni vektorji, ki jih dobimo s pomočjo transformacije $\mathbf{Q}\mathbf{Q}^\top$ najboljši diskriminantni vektorji originalnega prostora. Čeprav njegovi rezultati preizkusov izboljšajo uspešnost prepoznave obrazov, pa je v primeru nesingularnosti matrike $\mathbf{S}_W$ rezultat še vedno enak rezultatu standardne metode LDA.

Avtorja Yu in Yang [111] sta predstavila svojo različico direktne metode LDA, ki jo bomo imenovali DLDA. Preizkusila sta jo na visoko-dimenzijskih podatkih na primeru prepoznavanja obrazov. Njuna metoda deluje tako, da simultano diagonalizira obe simetrični matriki $\mathbf{S}_W$ in $\mathbf{S}_B$.

Glavna ideja metode DLDA je zavreči ničelni prostor matrike $\mathbf{S}_B$ s pomočjo diagonalizacije matrike $\mathbf{S}_B$ in nato diagonalizacije matrike $\mathbf{S}_W$. Pozorni moramo biti na to, da je običajni postopek metode LDA takšen, da najprej diagonalizira matriko $\mathbf{S}_W$ in nato matriko $\mathbf{S}_B$ (6.100, 6.101), s tem pa posledično zavrže ničelni prostor matrike $\mathbf{S}_W$, ki vsebuje pomembne diskriminantne informacije [14].

S pomočjo predlaganega postopka DLDA se poleg ohranitve ničelnega prostora matrike $\mathbf{S}_W$ izognemo tudi problemu singularnosti. Predlagan postopek za izračun projekcijske matrike $\mathbf{W}_{dlda}$ s pomočjo predlagane direktne metode LDA opisuje algoritem 7.

Algoritem 7 : Direktna metoda LDA – DLDA

Vhod: učna množica $\mathbf{X}$ z C razredi: $\mathbf{X} = \{\mathbf{X}_i\}_{i=1}^{C}$, vsak razred vsebuje $X_i = \{x_{ij}\}_{j=1}^{N_i}$ slik obrazov, kjer je $x_{ij} \in \mathbb{R}^M$.

Izhod: k-dimenzijski LDA podprostor, ki ga določa transformacijska matrika $\mathbf{W} \in \mathbb{R}^{M \times k}$.

1: Določi vektor povprečnih vrednosti: $\boldsymbol{\mu} = \frac{1}{N}\sum_{j=1}^{N} \mathbf{x}_j$.

2: Določi vektor povprečji za posamezen razred C_i: $\boldsymbol{\mu}_i = \frac{1}{N_i}\sum_{j=1}^{N_i} \mathbf{x}_{ij}$.

3: Izračunaj matriko razpršenosti znotraj posameznega razreda: $\mathbf{S}_W = \sum_{i=1}^{C}\sum_{j=1}^{N_i}(\mathbf{x}_{ij} - \boldsymbol{\mu}_i)(\mathbf{x}_{ij} - \boldsymbol{\mu}_i)^\top$

4: Izračunaj matriko razpršenosti med posameznimi razredi: $\mathbf{S}_B = \sum_{i=1}^{C} N_i(\boldsymbol{\mu}_i - \boldsymbol{\mu})(\boldsymbol{\mu}_i - \boldsymbol{\mu})^\top$

5: Poišči $m < C - 1$ lastnih vektorjev $\mathbf{U}$, katerim pripadajo največje lastne vrednosti $\boldsymbol{\Lambda}_B$, matrike $\mathbf{S}_B = \boldsymbol{\Phi}_B\boldsymbol{\Phi}_B^\top$: $\mathbf{U}^\top\mathbf{S}_B\mathbf{U} = \boldsymbol{\Lambda}_B$,

- Poišči m lastnih vektorjev matriki $\boldsymbol{\Phi}_B\boldsymbol{\Phi}_B^\top$, katerim pripadajo neničelne lastne vrednosti in jih označi kot $\mathbf{E} = [\mathbf{e}_1, \ldots, \mathbf{e}_m]$, kjer je $\boldsymbol{\Phi}_B = [\boldsymbol{\Phi}_{B,1}, \ldots, \boldsymbol{\Phi}_{B,C}]$, $\boldsymbol{\Phi}_{B,i} = (N_i/N)^{(1/2)}(\boldsymbol{\mu}_i - \boldsymbol{\mu})$,
- Izračunaj prvih m najpomembnejših lastnih vektorjev $\mathbf{U}$ matrike $\mathbf{S}_B$ in pripadajoče lastne vrednosti $\boldsymbol{\Lambda}_B$ kot: $\mathbf{U} = \boldsymbol{\Phi}_B\mathbf{E}$ in $\boldsymbol{\Lambda}_B = \mathbf{U}^\top\mathbf{S}_B\mathbf{U}$

6: Naj bo $\mathbf{Z} = \mathbf{U}\boldsymbol{\Lambda}_B^{-1/2}$ transformacija beljenja matrike $\mathbf{S}_B$, ki je hkrati tudi transformacija, ki zmanjša dimenzijo prostora iz M na m, velja $\mathbf{Z}^\top\mathbf{S}_B\mathbf{Z} = \mathbf{I}$,

7: Diagonaliziraj matriko $\mathbf{Z}^\top\mathbf{S}_W\mathbf{Z}$ tako, da izračunaš lastne vektorje $\mathbf{P} = [\mathbf{p}_1, \ldots, \mathbf{p}_m]$ in lastne vrednosti $\boldsymbol{\Lambda}_W$ da velja: $\mathbf{W}^\top(\mathbf{Z}^\top\mathbf{S}_W\mathbf{Z})\mathbf{W} = \boldsymbol{\Lambda}_W$,

8: Izberi tistih prvih $k \leq m$ lastnih vektorjev matrike $\mathbf{P}$ katerim pripadajo najnižje lastne vrednosti $\boldsymbol{\Lambda}_W$ in jo označi s $\mathbf{P}_k = [\mathbf{p}_1, \ldots, \mathbf{p}_k]$.

9: Transformacijsko matriko $\mathbf{W}$ določimo kot: $\mathbf{W} = \mathbf{Z}\mathbf{P}_k\boldsymbol{\Lambda}_W$

Z uporabo te tehnike sta avtorja Yu in Yang v svojih preizkusih pokazala, da lahko direktno metodo LDA uporabimo na originalnih podatkih slik obrazov brez predhodnega zmanjševanja dimenzije. Pokazala sta tudi, da z zamenjavo matrike $\mathbf{S}_B$ v Fisherjevi kriterijski funkciji (6.88) z matriko $\mathbf{S}_T$, kjer je $\mathbf{S}_T = \mathbf{S}_B + \mathbf{S}_W$ prvi del algoritma DLDA predstavlja ravno metodo PCA.

Avtor Juwei Lu et al [42] je za izračun direktne LDA uporabil enak postopek, le da je namesto funkcije $J(\mathbf{W})$ (6.88) predlagal kriterijsko funkcijo:

$$J(\mathbf{W}) = \frac{|\mathbf{W}^\top\mathbf{S}_B\mathbf{W}|}{|\mathbf{W}^\top(\mathbf{S}_W + \mathbf{S}_B)\mathbf{W}|} \ , \tag{6.109}$$

kjer imamo v imenovalcu namesto matrike $\mathbf{S}_W$ matriko razpršenosti celotnih podatkov $\mathbf{S}_T = \mathbf{S}_W + \mathbf{S}_B$. Kasneje je avtor Juwei Lu et al [59] predlagal regularizirano

metodo LDA (angl. *regularized LDA*, RLDA). Postopek se razlikuje v kriterijski funkciji, ki jo določa regularizacijski prameter η s katerim določamo, koliko bo funkcija pristranska (angl. *biased*).

$$J(\mathbf{W}) = \frac{|\mathbf{W}^\top \mathbf{S}_B \mathbf{W}|}{|\eta(\mathbf{W}^\top \mathbf{S}_B \mathbf{W}) + (\mathbf{W}^\top \mathbf{S}_W \mathbf{W})|} \quad . \tag{6.110}$$

Vidimo lahko, da v primeru, ko je parameter $\eta = 0$ dobimo ravno metodo DLDA 7, ki jo predlagata avtorja Yu in Yang [111]. V tem primeru se bo metoda LDA maksimalno prilagodila učnim podatkom. Temu pa se lahko izognemo tako, da povečamo parameter, tako da bo $\eta > 0$. Če povečamo parameter na $\eta = 1$, potem dosežemo drugo skrajnost, kjer dobimo ravno originalno direktno metodo avtorja Juwei Lu et al [42], ki jo opisuje kriterijska funkcija 6.109.

6.4 Metoda neodvisnih komponent

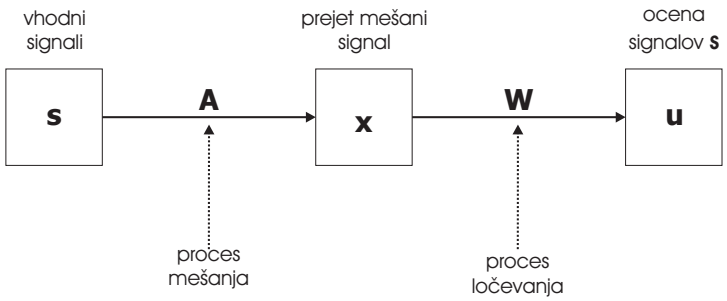
Medtem ko metoda PCA dekorelira vhodne podatke z uporabo drugo-nivojske statistike in tako generira stisnjene podatke z minimalno srednjo kvadratno napako projekcije, metoda ICA (angl. *Independent Component Analysis*) minimizira tako drugo-nivojske kot tudi višje-nivojske odvisnosti v vhodnih podatkih. Metodo se največkrat uporablja pri problemu slepega ločevanja izvorov (angl. *Blind Source Separation*, BSS), kjer je cilj razčleniti sprejeti signal v linearno kombinacijo nam neznanih neodvisnih signalov. Naj bo $\mathbf{s}$ vektor neznanih izvornih signalov in $\mathbf{x}$ vektor sprejete mešanice signalov, ki jih moramo ločiti. Če je matrika $\mathbf{A}$ neznana mešalna matrika (angl. *mixing matrix*), potem lahko sistem v katerem se signali mešajo, zapišemo kot:

$$\mathbf{x} = \mathbf{A}\mathbf{s} \ . \tag{6.111}$$

Predpostavimo, da so izvorni signali med seboj neodvisni in da je mešalna matrika $\mathbf{A}$ obrnljiva (lahko izračunamo njen inverz). Glede na dane predpostavke in sprejeti mešani signal $\mathbf{x}$, poskuša metoda ICA poiskati takšno mešalno matriko $\mathbf{A}$ oziroma ločevalno matriko $\mathbf{W}$, da bodo signali $\mathbf{u}$ med seboj kar se da neodvisni:

$$\mathbf{u} = \mathbf{W}\mathbf{x} = \mathbf{W}\mathbf{A}\mathbf{s} \tag{6.112}$$

Slika 6.15 prikazuje model mešanja in ločevanja signalov.



Slika 6.15: Model slepega ločevanja izvorov – BSS.

Na metodo ICA lahko gledamo tudi kot na posplošeno metodo PCA. Kot smo podrobneje opisali v poglavju 6.2, metoda PCA dekorelira učne podatke tako, da je njihova kovarianca enaka nič. Beljenje (angl. *whitening*) je strožji pogoj, ki zahteva, da so podatki med seboj nekorelirani in imajo enotno varianco. Transformacijo beljenja lahko definiramo kot: $\mathbf{D}^{1/2}\mathbf{R}^{\top}$, kjer je matrika $\mathbf{D}$ diagonalna matrika lastnih vrednosti in matrika $\mathbf{R}$ vsebuje ortogonalne lastne vektorje kovariančne matrike vhodnih podatkov. S pomočjo beljenja lahko sprejet mešani signal transformiramo tako, da je ortogonalen. S pomočjo metode ICA pa gremo še en korak dlje; tako pobeljene podatke transformiramo v množico statistično neodvisnih signalov.

Signali so statistično neodvisni, ko velja:

$$\mathbf{f_u}(\mathbf{u}) = \prod_i \mathbf{f_{u_i}}(\mathbf{u}_i) \ , \tag{6.113}$$

kjer je f_u verjetnostna funkcija spremenljivke $\mathbf{u}$. Ekvivalentno lahko rečemo, da je vektor $\mathbf{u}$ normalno porazdeljen. Lahko se zgodi, da ne obstaja takšna matrika $\mathbf{W}$, ki bi popolnoma zadostila pogoju neodvisnosti in ne obstaja enolično določena formula za rešitev matrike $\mathbf{W}$. Zaradi tega pa obstaja nekaj algoritmov, ki iterativno aproksimirajo matriko $\mathbf{W}$ in s tem neposredno večajo neodvisnost.

Ker je problem neodvisnosti direktno težko rešiti, vsi predlagani algoritmi preoblikujejo problem v iterativno optimizacijo gladke funkcije, ki doseže globalni optimum, ko so izhodni vektorji $\mathbf{u}$ med seboj neodvisni.

Algoritem InfoMax [5] predpostavlja, da je neodvisnost maksimalna, ko je entropija $\mathbf{H}(\mathbf{u})$ največja. Entropijo določimo kot:

$$\mathbf{H}(\mathbf{u}) \equiv -\int \mathbf{f_u}(\mathbf{u}) \log \mathbf{f}_u(\mathbf{u}) d\mathbf{u} \ . \tag{6.114}$$

Algoritem InfoMax izvaja gradientno dviganje posameznih elementov w_{ij} tako, da maksimizira $\mathbf{H}(\mathbf{u})$. Algoritem je dobil ime, ker pri izračunu upošteva predpostavko, da maksimizacija entropije $\mathbf{H}(\mathbf{u})$ hkrati tudi maksimizira vzajemno informacijo $\mathbf{I}(\mathbf{x}, \mathbf{y})$ med vhodnimi in izhodnimi vektorji. Signale ločuje v časovnem prostoru. Algoritem zahteva predhodno nastavitev ustreznih parametrov in iterativno učenje ločevalnih filtrov.

Algoritem JADE (angl. *Joint Approximate Diagonalization of Eigen-matrices*) [12] ne zahteva nastavljanja parametrov, upošteva statistične lastnosti signalov – kumulante četrtega reda in ne deluje na samih vhodnih vzorcih. Algoritem lahko povzamemo v naslednjih korakih: (1) dekorelacija vhodnih podatkov, (2) izračun matrik kumulantov četrtega reda, (3) skupna diagonalizacija matrik kumulantov in (4) izračun ločevalne matrike.

Algoritem FastICA [38] je med vsemi algoritmi gotovo najsplošnejši, ker maksimizira funkcijo:

$$J(y) \approx c[E\{G(y)\} - E\{G(v)\}]^2 \ , \tag{6.115}$$

kjer je G nekvadratična funkcija, v Gaussova naključna spremenljivka in c pozitivna konstanta. Dokazano je namreč, da z maksimizacijo funkcije predlagane oblike maksimiziramo tudi neodvisnosti.

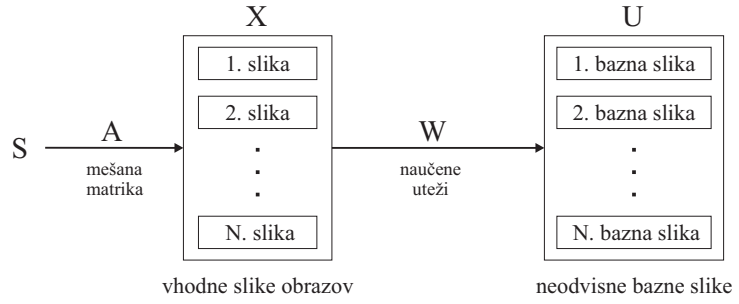
Vsi trije najpogosteje uporabljeni algoritmi: InfoMax, JADE in FastICA maksimizirajo funkcije z enakim globalnim optimumom. Kot rezultat tega morajo vsi trije algoritmi konvergirati k isti rešitvi za dano množico podatkov. V praksi pa vsak algoritem uporablja nekoliko drugačno formulacijo pogojev neodvisnosti, ki se uporabljajo pri različnih tehnikah aproksimacije, zaradi česar se tudi rezultati posameznih algoritmov med seboj razlikujejo. Razlike so minimalne in so odvisne

predvsem od vhodnih podatkov. Moghaddam [70] in Lee et al. [57] sta preizkusila algoritma JADE in FastICA na slikovnih podatkih in ugotovila, da med njima ni razlike. Draper et al. [17] je prav tako na slikah preizkusil algoritma InfoMax in FastICA in ugotavlja, da je razlika uspešnosti med njima zelo majhna in je odvisna predvsem od arhitekture, ki se uporablja. Algoritem FastICA je bolje deloval pri drugi arhitekturi, medtem ko se je algoritem InfoMax izkazal za boljšega pri prvi arhitekturi. Algoritma JADE pa avtor ni preizkusil, ker ni primeren za podatke, ki imajo dimenzijo večjo od 50.

Ne glede na to, kateri algoritem uporabimo za realizacijo metode ICA, jo lahko na primeru prepoznavanja obrazov uporabimo na dva različna načina. Bartlett et al. [2] predlaga dve arhitekturi: prva arhitektura (ICA I) določa statistično neodvisne bazne slike, medtem ko določa druga arhitektura (ICA II) neodvisne parametre, s katerimi predstavimo slike obrazov. V nadaljevanju bomo obe arhitekturi podrobneje predstavili.

6.4.1 Prva arhitektura metode ICA – ICA I

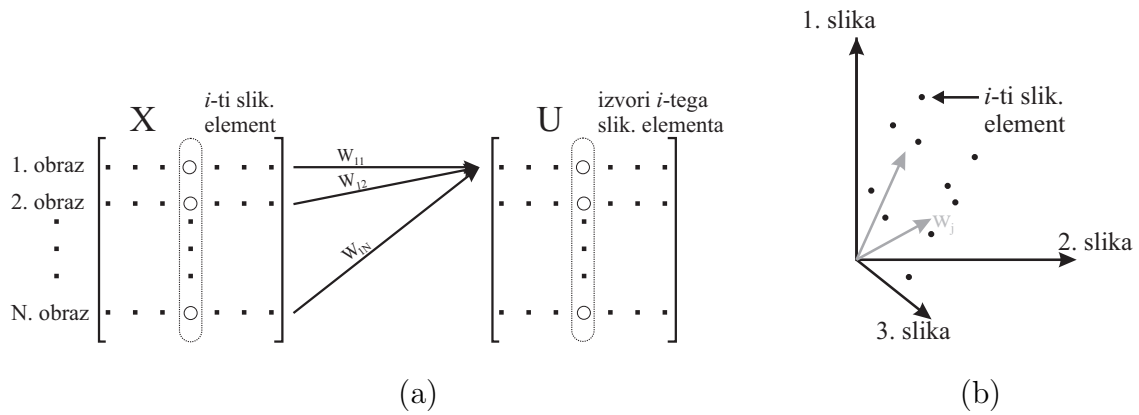
Cilj uporabe te arhitekture je dobiti statistično neodvisne bazne slike. Vhodno podatkovno matriko $\mathbf{X}$ organiziramo tako, da so posamezne slike obrazov v vrsticah in slikovni elementi v stolpcih. Tako ima matrika $\mathbf{X}$ N vrstic in M stolpcev. Vsaka slika obraza je normalizirana tako, da ima povprečno vrednost enako nič. V prvi arhitekturi so vhodne slike obrazov v matriki $\mathbf{X}$ mišljene kot linearna mešanica statistično neodvisnih baznih slik $\mathbf{S}$, ki jih določa neznana mešalna matrika $\mathbf{A}$. S pomočjo algoritma ICA moramo določiti matriko uteži $\mathbf{W}$, ki jo nato uporabimo, da dobimo množico neodvisnih baznih slik v posameznih vrsticah matrike $\mathbf{U}$ (slika 6.16). Metodo ICA s to arhitekturo sta uporabila tudi avtorja Bell in Sejnowski za ločevanje izvornih signalov [5].



Slika 6.16: Postopek iskanja statistično neodvisnih baznih slik s pomočjo metode ICA.

Pri tej arhitekturi so obrazne slike naključne spremenljivke, slikovni elementi pa določajo posamezen učni primer. V tem primeru je ustrezneje govoriti o neodvisnosti slik oziroma funkcij slik. Sliki i in j sta neodvisni v primeru, ko se

premaknemo čez vse slikovne elemente in pri tem ne moremo napovedati vrednosti slikovnega elementa iz slike j glede na vrednost istega slikovnega elementa iz slike i. Slika 6.17(a) prikazuje razporeditev vhodnih podatkov v matriki $\mathbf{X}$ za prvo arhitekturo. S pomočjo metode ločevanja obraznih slik dobimo v vrsticah matrike $\mathbf{U}$ neodvisne slike. Slika 6.17(b) pa prikazuje vrednosti posameznega slikovnega elementa i v koordinatnem sistemu, ki ga določajo slike. Metoda ICA v tej arhitekturi poišče vektorje uteži v smereh statističnih odvisnosti med posameznimi slikovnimi elementi.



Slika 6.17: Predstavitev prve arhitekture metode ICA na slikah obrazov: (a) predstavitev vhodne matrike in iskanje neodvisnih baznih slik, (b) predstavitev slikovnih elementov v koordinatnem sistemu slik.

Bartlett et al. [2] je nad podatki predhodno izvedla metodo PCA, da je vhodne podatke projicirala v manjši podprostor dimenzije m. Metodo ICA je nato izvedla na m slikah namesto na vseh N originalnih slikah. Model ločevanja izvorov predpostavlja, da so slike v matriki $\mathbf{X}$ linearna kombinacija statistično neodvisnih izvornih slik, zato ni nič narobe, če originalne slike nadomestimo z drugo linearno kombinacijo slik. Izkazalo se je celo, da metoda PCA izboljša učinkovitost metode ICA, s tem ko odstrani manjše lastne vrednosti pred beljenjem in s tem, ko zmanjša računsko zahtevnost z odpravo drugo nivojskih odvisnosti. Metoda PCA vhodne podatke predhodno dekolerira, medtem ko metoda ICA loči preostale višje nivojske odvisnosti.

V nadaljevanju bomo opisali matematično definicijo prve arhitekture metode ICA, ki jo bomo v nadaljevanju imenovali metoda ICA I.

Matematična definicija metode ICA I

Imejmo N vhodnih slik obrazov, ki predstavljajo učne primere. Predstavimo jih kot vektorje: $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N \in \mathbb{R}^M$. Definirajmo matriko $\mathbf{X}$, ki ima v svojih stolpcih vhodne vektorje slik: $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N)$ in njeno transponirano matriko $\mathbf{Y} = \mathbf{X}^\top$. Pri prvi arhitekturi gledamo na slike obrazov kot na naključne spremenljivke in slikovni

elementi predstavljajo posamezne primere teh spremenljivk. To pomeni, da metodo ICA izvedemo na vrsticah matrike $\mathbf{X}$, ki predstavljajo slike obrazov. Označimo z matriko $\mathbf{Y} = (\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_M)$ naše opazovane vektorje na podlagi katerih bomo določili ločevalno matriko za model ICA I.

Prvi korak je centriranje vhodnih podatkov v opazovanem vektorskem prostoru $\mathbb{R}^N$. Vektor povprečja izračunamo kot: $\mathbf{m}_I = \frac{1}{M}\sum_{j=1}^{M}\mathbf{y}_j$ in označimo $\mathbf{m}_I = (m_1, m_2, \ldots, m_N)^\top$. Posamezna vrednost m_j je pravzaprav povprečje vseh slikovnih elementov j-te slike. Podatke centriramo tako, da od vsakega opazovanega vektorja $\mathbf{y}_j$ odštejemo povprečni vektor $\mathbf{m}_I$: $\mathbf{y}_j \leftarrow (\mathbf{y}_j - \mathbf{m}_I)$. Tako dobimo matriko $\mathbf{Y} = (\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_M)$, ki vsebuje v svojih vrsticah centrirane slike. Označimo matriko kot $\mathbf{X}_v = \mathbf{Y}^\top$ in jo poimenujmo vertikalna centrirana vhodna matrika, ki vsebuje po stolpcih centrirane vhodne slike obrazov.

Naslednji korak pred samim postopkom ICA je proces beljenja podatkov. Na centriranih vektorjih $\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_M$ izvedemo metodo PCA. To naredimo tako, da najprej izračunamo kovariančno matriko:

$$\mathbf{\Sigma}_I = \frac{1}{M}\sum_{j=1}^{M}\mathbf{y}_j\mathbf{y}_j^\top = \frac{1}{M}\mathbf{Y}\mathbf{Y}^\top \ . \tag{6.116}$$

Označimo $\mathbf{G}_v = \mathbf{Y}\mathbf{Y}^\top$ in izračunamo njene ortonormirane lastne vektorje $\boldsymbol{\gamma}_1, \boldsymbol{\gamma}_2, \ldots, \boldsymbol{\gamma}_m$, ki jim pripada m največjih pozitivnih lastnih vrednosti $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_m$. Tako dobimo m največjih lastnih vrednosti kovariančne matrike $\mathbf{\Sigma}_I$ kot: $\frac{\lambda_1}{M}, \frac{\lambda_2}{M}, \ldots, \frac{\lambda_m}{M}$ in njeni pripadajoči ortonormirani lastni vektorji so: $\boldsymbol{\gamma}_1, \boldsymbol{\gamma}_2, \ldots, \boldsymbol{\gamma}_m$. Označimo $\mathbf{V} = (\boldsymbol{\gamma}_1, \boldsymbol{\gamma}_2, \ldots, \boldsymbol{\gamma}_m)$ in $\mathbf{\Lambda} = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_m)$ in definirajmo matriko beljenja:

$$\mathbf{P} = \mathbf{V}(\frac{1}{M}\mathbf{\Lambda})^{-\frac{1}{2}} = \sqrt{M}\mathbf{V}\mathbf{\Lambda}^{-\frac{1}{2}} \ , \tag{6.117}$$

tako, da velja $\mathbf{P}^\top\mathbf{\Sigma}_I\mathbf{P} = \mathbf{I}$. Podatkovno matriko $\mathbf{Y} = (\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_M)$ lahko pobelimo s pomočjo transformacije:

$$\mathbf{R} = \mathbf{P}^\top\mathbf{Y} \ . \tag{6.118}$$

Metodo ICA nato izvršimo na tako pobeljenih podatkih $\mathbf{R}$. Rezultat metode je m neodvisnih baznih slik v vrsticah matrike $\mathbf{U}_I$ tako da velja:

$$\mathbf{U}_I = \mathbf{W}_I\mathbf{R} \ , \tag{6.119}$$

kjer $\mathbf{W}_I$ označuje ločevalno matriko. Transformacijsko matriko W_{ICA_I} metode ICA I nam tako predstavlja kar matrika U_I.

Da dobimo predstavitvene koeficiente $\mathbf{b}$ s katerimi lahko predstavimo sliko lahko uporabimo dve metodi. Prva metoda, ki bazira na projekciji [44], direktno projicira sliko na osi, ki jo določajo bazne slike metode ICA. Druga metoda pa bazira na povprečni kvadratni napaki (angl. *Mean Square Error*, MSE) [2]. Obe metodi

sta ekvivalentni v primeru, ko je ločevalna matrika $\mathbf{W}_I$ ortogonalna, kar bi moralo teoretično veljati. V praksi pa lahko ortogonalnost dosežemo samo s pomočjo algoritma FastICA. Po prvi metodi tako dano sliko $\mathbf{x}$ vertikalno centriramo in jo nato projiciramo v m neodvisnih baznih slik kot:

$$\mathbf{b} = \mathbf{W}_{ICA_I}\mathbf{x} = \mathbf{U}_I\mathbf{x} = \mathbf{W}_I\mathbf{R}\mathbf{x} \ . \tag{6.120}$$

Lahko jo razčlenimo v dve transformaciji:

$$\mathbf{c} = \mathbf{R}\mathbf{x} = \mathbf{P}^\top\mathbf{Y}\mathbf{x} = (\sqrt{M}\mathbf{V}\mathbf{\Lambda}^{-\frac{1}{2}})^\top\mathbf{Y}\mathbf{x} = \sqrt{M}\mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{V}^\top\mathbf{Y}\mathbf{x} \tag{6.121}$$

$$\mathbf{b} = \mathbf{W}_I\mathbf{c} \tag{6.122}$$

Po drugi metodi pa določimo koeficiente s pomočjo minimalne povprečne kvadratne napake. Rekonstrukcijo dane slike $\mathbf{x}$ lahko zapišemo kot:

$$\hat{\mathbf{x}} = \mathbf{R}^\top\mathbf{c} \ , \tag{6.123}$$

kjer vektor $\mathbf{c} = \mathbf{R}\mathbf{x}$ vsebuje PCA koeficiente, enako kot v enačbi 6.121. Iz enačbe $\mathbf{U} = \mathbf{W}\mathbf{R}$ in predpostavke, da je matrika $\mathbf{W}$ obrnljiva sledi:

$$\mathbf{R} = \mathbf{W}_I^{-1}\mathbf{U}_I \ . \tag{6.124}$$

Iz enačbe 6.124 in enačbe 6.124 dobimo:

$$\hat{\mathbf{x}} = \mathbf{U}_I^\top(\mathbf{W}_I^{-1})^\top\mathbf{c} \ . \tag{6.125}$$

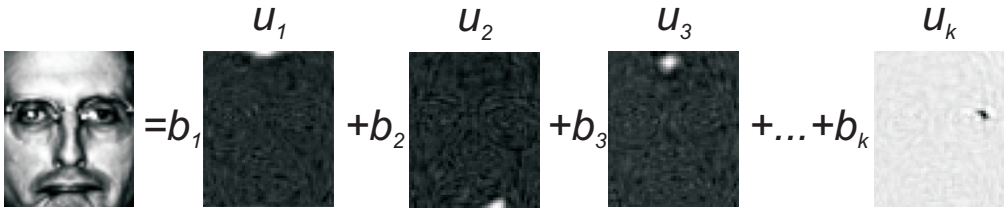
Tako smo dobili koeficiente $\mathbf{b}$, s katerimi lahko predstavimo dano sliko $\mathbf{x}$ v prostoru, ki ga določajo statistično neodvisne bazne slike, ki smo jih dobili z metodo ICA I:

$$\mathbf{b} = (\mathbf{W}_I^{-1})^\top\mathbf{c} = (\mathbf{W}_I^{-1})^\top\mathbf{R}\mathbf{x} \ . \tag{6.126}$$

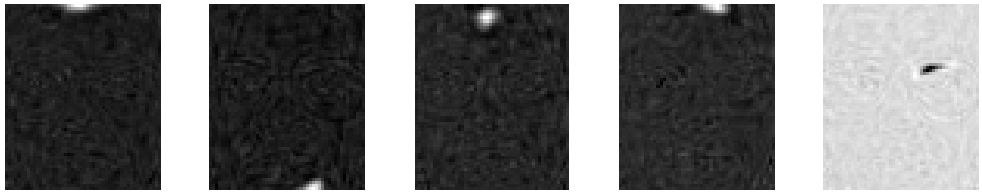
Vidimo lahko, da je takšna predstavitev konsistentna s predstavijo, ki jo predstavlja enačba 6.120. Če pa je ločevalna matrika $\mathbf{W}_I$ ortogonalna: $\mathbf{W}_I^{-1} = \mathbf{W}_I^\top$, potem sta enačbi 6.120 in 6.126 enaki. V primeru, če bi uporabili samo metodo ICA brez predhodne metode PCA, pa bi koeficiente $\mathbf{b}$ vsebovala kar inverzna ločevalna matrika $\mathbf{W}^{-1}$ in koeficiente posamezne slike $\mathbf{x}$ bi določili kot: $\mathbf{b} = \mathbf{U}\mathbf{x}$ (slika 6.18).

Slika 6.19 nam prikazuje prvih pet baznih slik, ki smo jih dobili s pomočjo metode ICA in predstavljene prve arhitekture v posameznih vrsticah matrike $\mathbf{U}$.

Vsaka vrstica mešalne matrike $\mathbf{W}$, ki smo jo našli s pomočjo metode ICA I, predstavlja skupek slikovnih elementov, ki imajo enake lastnosti pri vseh slikah. Posamezna vrstica matrike $\mathbf{U}$ pa nam pove, kako blizu je posamezen slikovni element določeni i-ti gruči. Ker smo uporabili model za razpršitev in neodvisnost izvornih signalov, so tudi naše bazne slike razpršene in neodvisne. Razpršenost se kaže po tem, da ima vsaka bazna slika veliko število slikovnih elementov, ki so blizu ničle in samo nekaj takšnih, ki so velika pozitivna ali negativna števila. Opazimo lahko tudi, da so bazne slike lokalne, kar pomeni, da se slikovni elementi z ekstremnimi vrednostmi držijo skupaj. Zaradi tega je največ odvisnosti med tistimi slikovnimi elementi, ki so si blizu.



Slika 6.18: Predstavitev slike obraza s pomočjo koeficientov **b** in linearne kombinacije neodvisnih baznih slik **u**.



Slika 6.19: Primeri prvih petih lastnih obrazov metode ICA I.

Metoda PCA I

Iz metode ICA, ki smo jo uporabili v prvi arhitekturi (ICA I), lahko pred samim ločevanjem statistično neodvisnih slik izločimo samo prvi korak . Izkaže se, da je ta predkorak metode ICA I ravno PCA metoda vertikalno centriranih podatkov (PCA I). Tako določeno metodo PCA I bomo v poglavju 9 preizkusili in primerjali s samo metodo ICA I ter poskušali ugotoviti, kolikšen del uspešnosti prepoznave obrazov doprinese postopek ločevanja na statistično neodvisne bazne slike s pomočjo metode ICA.

Najprej izvršimo PCA nad vertikalno centriranimi podatki $\mathbf{X}_v = \mathbf{Y}^\top$, nato izračunamo kovariančno matriko:

$$\mathbf{\Sigma}_v = \frac{1}{N}\mathbf{X}_v\mathbf{X}_v^\top = \frac{1}{N}\mathbf{Y}^\top\mathbf{Y} \ . \tag{6.127}$$

Iz teorema o dekompoziciji singularnih vrednosti – SVD vemo, da imata matriki $\mathbf{G}_v = \mathbf{X}_v\mathbf{X}_v^\top$ in $\mathbf{Y}^\top\mathbf{Y}$ enake neničelne lastne vrednosti. Predpostavimo, da ortonormirani lastni vektorji $\boldsymbol{\gamma}_1, \boldsymbol{\gamma}_2, \ldots, \boldsymbol{\gamma}_m$ matrike $\mathbf{G}_v$ pripadajo m-tim največjim pozitivnim lastnim vrednostim $\lambda_1 \geq \lambda_2 \geq \ldots \lambda_m$. V naslednjem koraku lahko določimo pripadajoče ortonormirane lastne vektorje matrike $\mathbf{Y}^\top\mathbf{Y}$ kot:

$$\boldsymbol{\beta}_j = \frac{1}{\sqrt{\lambda}}\mathbf{Y}^\top\boldsymbol{\gamma}_j \ , \qquad j = 1, \ldots, m \ . \tag{6.128}$$

Tako je m največjih pozitivnih lastnih vrednosti kovariančne matrike $\mathbf{\Sigma}_v$ enakih: $\frac{\lambda_1}{M}, \frac{\lambda_2}{M}, \ldots, \frac{\lambda_m}{M}$ in pripadajoči ortonormirani lastni vektorji so: $\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \ldots, \boldsymbol{\beta}_m$. Naj bo $\mathbf{P}_v = (\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \ldots, \boldsymbol{\beta}_m) = \mathbf{Y}^\top\mathbf{V}\mathbf{\Lambda}^{-\frac{1}{2}}$. Pripadajoča PCA transformacija s predhodnim

vertikalnim centriranjem je tako definirana kot:

$$\mathbf{c} = \mathbf{P}_v^\top \mathbf{x} = (\mathbf{Y}^\top \mathbf{V} \mathbf{\Lambda}^{-\frac{1}{2}})^\top \mathbf{x} = \mathbf{\Lambda}^{-\frac{1}{2}} \mathbf{V}^\top \mathbf{Y} \mathbf{x} \ . \tag{6.129}$$

Če primerjamo enačbi 6.129 in 6.121 lahko vidimo, da je prva transformacija pri metodi ICA I enaka PCA transformaciji na vertikalno centriranih podatkih. Konstanta $\sqrt{M}$ za katero se transformacijski matriki razlikujeta, namreč nima vpliva na rezultat pri izločanju značilk. Tako lahko zapišemo transformacijsko matriko $\mathbf{R}$ tudi kot $\mathbf{R} = \sqrt{M}\mathbf{P}_v^\top$. Vidimo, da vsebuje matrika v vrsticah lastne vektorje kovariančne matrike $\mathbf{\Sigma}_v$ ali tako imenovane lastne slike. Cilj metode ICA je ponovno dobiti množico neodvisnih baznih slik iz tako določenih lastnih slik.

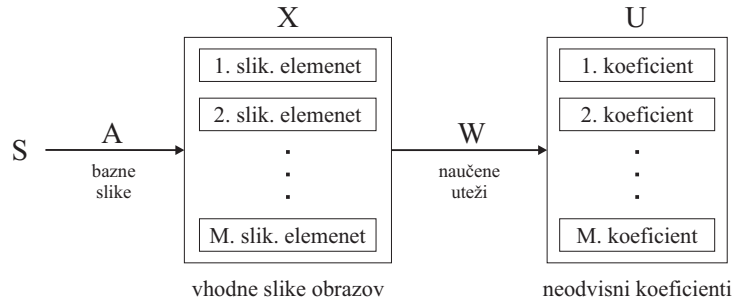
Slika 6.20: Primeri prvih petih lastnih obrazov metode PCA I.

Slika 6.20 prikazuje prvih pet lastnih slik oziroma lastnih obrazov metode PCA I. Vidimo lahko, da opisujejo lastni obrazi globalne lastnosti in so zaradi tega bolj podobni lastnim obrazom metode PCA (slika 6.9) kot tistim, ki jih dobimo z metodo ICA I.

6.4.2 Druga arhitektura metode ICA – ICA II

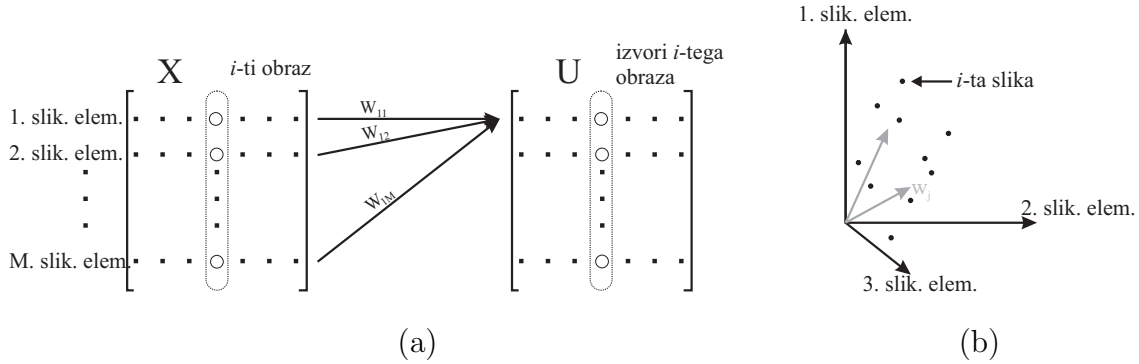
Medtem ko so bazne slike v prvi arhitekturi statistično neodvisne, pa koeficienti s katerimi predstavimo vhodne slike v obraznem podprostoru, niso. Cilj metode ICA druge arhitekture (ICA II) je najti statistično neodvisne koeficiente, s katerimi predstavimo vhodne podatke. Pri tej arhitekturi je v primerjavi s prvo arhitekturo vhodna matrika obrnjena: slikovni elementi so spremenljivke in slike so opazovani primeri. Ločevanje izvorov se vrši na slikovnih elementih in posamezna vrstica naučene matrike uteži $\mathbf{W}$ predstavlja sliko. Matrika $\mathbf{A}$, ki predstavlja inverz matrike $\mathbf{W}$ ima v svojih stolpcih bazne slike. Statistično neodvisne izvorne koeficiente, ki jih vsebuje izvorna matrika $\mathbf{S}$ in predstavljajo vhodne slike, poskušamo s pomočjo metode ICA II obnoviti v stolpcih matrike $\mathbf{U}$ (slika 6.21). Ta arhitektura je bila uporabljena tudi za iskanje slikovnih filtrov, ki so na naravnih slikah tvorili statistično neodvisne izhode [6].

Pri tej arhitekturi torej organiziramo vhodne podatke tako, da so slike v posameznih vrsticah matrike $\mathbf{X}$. Grafično predstavitev druge arhitekture predstavlja slika 6.22. V tem primeru so slikovni elementi naključne spremenljivke in slike posamezni primeri. Tukaj je ustrezneje govoriti o neodvisnosti slikovnih elementov oziroma funkcij slikovnih elementov. Slikovna elementa i in j sta neodvisna, če se



Slika 6.21: Postopek iskanja statistično neodvisnih koeficientov s pomočjo metode ICA.

premaknemo čez vse slike in ne moremo napovedati vrednosti slikovnega elementa i glede na slikovni element j iste slike.



Slika 6.22: Predstavitev druge arhitekture metode ICA na slikah obrazov: (a) predstavitev vhodne matrike in iskanje neodvisnih koeficientov, (b) predstavitev slik v koordinatnem sistemu slikovnih elementov.

Podobno kot pri prvi arhitekturi tudi tukaj predhodno zmanjšamo dimenzijo prostora. Namesto, da izvršimo metodo ICA neposredno na M-tih slikovnih elementih, zmanjšamo dimenzijo s pomočjo metode PCA na $m < M$. Metodo ICA tako izvršimo na m PCA koeficientih, s katerimi predstavimo vhodne slike. Ti koeficienti določajo stolpce vhodne matrike in morajo biti predhodno centrirani.

Matematična definicija metode ICA II

Imejmo N vhodnih slik obrazov, ki predstavljajo učne primere. Predstavimo jih kot vektorje: $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N \in \mathbb{R}^M$. Definirajmo matriko $\mathbf{X}$, ki ima v svojih stolpcih vhodne vektorje slik: $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N)$.

Cilj metode ICA pri drugi arhitekturi (ICA II) je najti statistično neodvisne koeficiente za vhodne slikovne podatke. Pri tej arhitekturi gledamo na slike obrazov kot na posamezne primere in na slikovne elemente kot na naključne spremenljivke.

Metodo ICA II uporabimo neposredno na matriki vhodnih podatkov $\mathbf{X} \in \mathbb{R}^{M \times N}$. Z drugimi besedami povedano, vektorji $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$ so naši opazovani vektorji, s pomočjo katerih bomo določili ločevalno matriko modela ICA II.

Najprej moramo podatke centrirati v opazovanem vektorskem prostoru $\mathbb{R}^M$. Vektor povprečja izračunamo kot: $\mathbf{m}_{II} = \frac{1}{N}\sum_{j=1}^{N} \mathbf{x}_j$. Od vsakega primera odštejemo povprečno vrednost: $\mathbf{x}_j \leftarrow (\mathbf{x}_j - \mathbf{m}_{II})$ in dobimo centrirano podatkovno matriko $\mathbf{X}_h = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N)$, ki vsebuje v svojih stolpcih slike obrazov, katerim je bil odštet povprečni obraz ($\mathbf{m}_{II}$). V primerjavi z vertikalnim centriranjem, ki smo ga uporabili pri metodi ICA I, pravimo takšnemu načinu centriranja tudi horizontalno centriranje.

Sledi proces beljenja. Da dobimo matriko beljenja moramo najprej izvršiti metodo PCA na centriranih podatkovnih vektorjih $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$. Podroben postopek beljenja bomo opisali v naslednjem poglavju 6.4.2, za sedaj samo predpostavimo, da je matrika beljenja $\mathbf{P}_w$. Matriko pobeljenih podatkov $\mathbf{Y} = (\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N)$ pa dobimo s pomočjo transformacije:

$$\mathbf{Y} = \mathbf{P}_w^\top \mathbf{X}_h \ . \tag{6.130}$$

Metodo ICA nato izvršimo na tako pobeljenih podatkih $\mathbf{Y}$:

$$\mathbf{U} = \mathbf{W}_{II} * \mathbf{Y} \ , \tag{6.131}$$

kjer matrika $\mathbf{W}_{II}$ označuje ločevalno matriko. Po vrsticah matrike $\mathbf{U}_{II}$ dobimo neodvisne vektorje, po stolpcih pa predstavitev posamezne slike s pomočjo neodvisnih koeficientov. Transformacijsko matriko $\mathbf{W}_{ICA_{II}}$ pa dobimo kot:

$$\mathbf{W}_{ICA_{II}} = \mathbf{W}_{II}\mathbf{P}_w^\top \ . \tag{6.132}$$

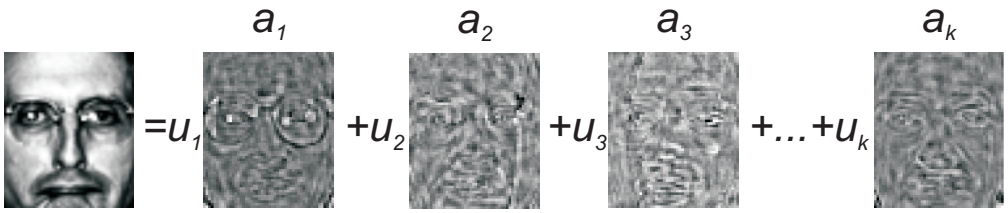
Posamezne vrstice matrike $\mathbf{W}_{ICA_{II}}$ označujejo bazne slike metode ICA II. Te bazne slike niso nujno neodvisne, vendar pa tvorijo koordinatne osi tako, da so koeficienti s katerimi so predstavljene slike med seboj, kar se da neodvisni. Dano sliko $\mathbf{x} \in \mathbb{R}^{M \times 1}$ po horizontalnem centriranju (odštejemo povprečno sliko) projiciramo v m bazne slike kot:

$$\mathbf{b} = \mathbf{W}_{ICA_{II}}\mathbf{x} = \mathbf{W}_{II}\mathbf{P}_w^\top \mathbf{x} \ . \tag{6.133}$$

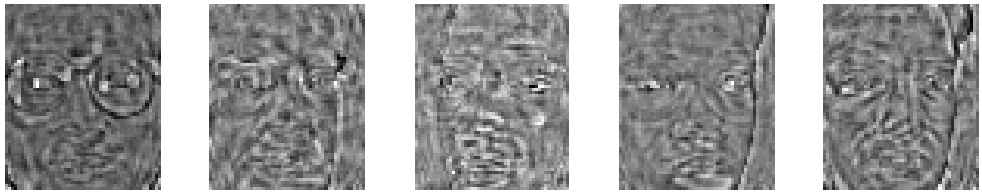
Vektor $\mathbf{b}$ vsebuje množico neodvisnih koeficientov, s katerimi predstavimo sliko $\mathbf{x}$ (slika 6.23). Očitno je, da lahko enačbo 6.133 razčlenimo v dve: transformacijo PCA beljenja $\mathbf{c} = \mathbf{P}_w^\top \mathbf{x}$, ki ji sledi ICA transformacija $\mathbf{b} = \mathbf{W}_{II}\mathbf{c}$.

V primeru, če ne bi uporabili metode PCA kot predkoraka metode ICA, bi dobili bazne slike kot stolpce matrike $\mathbf{A} = \mathbf{W}^{-1}$. V našem primeru pa dobimo bazne slike v stolpcih matrike $\mathbf{W}_{ICA_{II}}$. Prvih pet baznih slik prikazuje slika 6.24.

Pri tej arhitekturi se vsak stolpec mešalne matrike $\mathbf{A} = \mathbf{W}^{-1}$, ki ga dobimo s pomočjo metode ICA, poskuša približati gruči slik, ki imajo podoben videz vseh slikovnih elementov. Metoda generira bazne slike, ki so po izgledu bolj podobne



Slika 6.23: Predstavitev slike obraza s pomočjo neodvisnih koeficientov $\mathbf{u}$ in linearne kombinacije baznih slik matrike $\mathbf{A}$.



Slika 6.24: Primeri prvih petih lastnih obrazov metode ICA II.

obrazom kot pri metodi PCA. Bazne slike, ki jih najde matoda ICA II so torej povprečje med seboj podobnih slik. Za razliko od ICA I, se algoritem ne trudi, da bi bili stolpci matrike $\mathbf{A}$ čimbolj razpršeni in med seboj neodvisni. Zaradi tega predstavljajo bazne slike matrike $\mathbf{A}$ bolj globalne lastnosti v primerjavi z baznimi slikami metode ICA pri prvi arhitekturi.

Metoda PCA II

Podobno, kot smo naredili pri prejšnji arhitekturi, bi radi tudi tukaj izločili samo prvi korak pred statističnim ločevanjem koeficientov metode ICA II. V nadaljevanju bomo torej matematično definirali metodo PCA, ki smo jo uporabili kot predkorak metode ICA II. Tako definirano metodo bomo poimenovali PCA II in jo bomo v poglavju 9 preizkusili in ugotovili, kako se razlikuje uspešnost prepoznave obrazov v primerjavi z metodo ICA II.

Pri drugi arhitekturi izvršimo metodo PCA na horizontalno centriranih podatkih, kar je v bistvu navadna metoda PCA. Najprej torej izračunamo kovariančno matriko opazovanih vektorjev $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$:

$$\mathbf{\Sigma}_{II} = \frac{1}{N}\mathbf{X}_h\mathbf{X}_h^\top \ . \tag{6.134}$$

Označimo $\mathbf{G}_h = \mathbf{X}_h^\top\mathbf{X}_h$. Iz teorema o singularni dekompoziciji vemo, da imata matriki $\mathbf{G}_h$ in $\mathbf{X}_h\mathbf{X}_h^\top$ enake neničelne lastne vrednosti. Predpostavimo, da so $\boldsymbol{\gamma}_1$, $\boldsymbol{\gamma}_2$, $\ldots$, $\boldsymbol{\gamma}_m$ ortonormirani lastni vektorji matrike $\mathbf{G}_h$, ki pripadajo m največjim pozitivnim lastnim vrednostim $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_m$. Potem so ortonormirani lastni

vektorji matrike $\mathbf{X}_h\mathbf{X}_h^\top$:

$$\boldsymbol{\beta}_j = \frac{1}{\sqrt{\lambda_j}}\mathbf{X}_h\boldsymbol{\gamma}_j \ , \qquad j = 1, \ldots, m \ . \tag{6.135}$$

Tako je m največjih pozitivnih lastnih vrednosti matrike $\boldsymbol{\Sigma}_{II}$ enakih: $\frac{\lambda_1}{N}, \frac{\lambda_2}{N}, \ldots \frac{\lambda_m}{N}$ in pripadajoči lastni vektorji so: $\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \ldots, \boldsymbol{\beta}_m$. Označimo z $\mathbf{V} = (\boldsymbol{\gamma}_1, \boldsymbol{\gamma}_2, \ldots, \boldsymbol{\gamma}_m)$, $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_m)$ in $\mathbf{P}_h = (\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \ldots, \boldsymbol{\beta}_m) = \mathbf{X}_h\mathbf{V}\boldsymbol{\Lambda}^{-\frac{1}{2}}$. Potemtakem je standardna PCA transformacija definirana kot:

$$\mathbf{c} = \mathbf{P}_h^\top\mathbf{x} \ . \tag{6.136}$$

Kot vhod v metodo ICA pa potrebujemo pobeljene vhodne podatke. Preverimo, če transformacijska matrika $\mathbf{P}_h$ zadošča metriki beljenja:

$$\mathbf{P}_h^\top\boldsymbol{\Sigma}_{II}\mathbf{P}_h = \mathrm{diag}(\frac{\lambda_1}{N}, \frac{\lambda_2}{N}, \ldots, \frac{\lambda_m}{N}) = \frac{1}{N}\boldsymbol{\Lambda} \ . \tag{6.137}$$

Vidimo, da rezultat ni enotska matrika. Matriko beljenja P_w določimo kot:

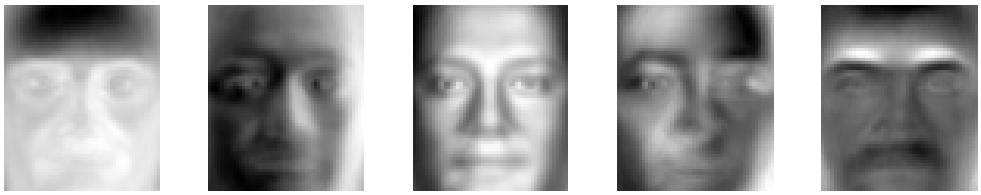
$$\mathbf{P}_w = \mathbf{P}_h(\frac{1}{N}\boldsymbol{\Lambda})^{-\frac{1}{2}} = \sqrt{N}\mathbf{X}_h\mathbf{V}\boldsymbol{\Lambda}^{-1} \ , \tag{6.138}$$

in lahko preverimo: $\mathbf{P}_w\boldsymbol{\Sigma}_{II}\mathbf{P}_w = \mathbf{I}$. Vhodne podatke lahko tako pobelimo s pomočjo transformacije:

$$\mathbf{c} = \mathbf{P}_w^\top\mathbf{x} \ . \tag{6.139}$$

Matrika beljenja $\mathbf{P}_w$ je v bistvu pobeljena PCA transformacijska matrika. Pobeljena PCA transformacija poleg tega, da odstrani korelacijo med spremenljivkami, tudi normalizira deviacijo posamezne spremenljivke tako, da je deviacija enaka 1.

Slika 6.25 prikazuje prvih pet lastnih slik ali obrazov metode PCA II. Vidimo lahko, da so lastni obrazi zelo podobni tistim, ki jih dobimo z metodo PCA (slika 6.9).



Slika 6.25: Primeri prvih petih lastnih obrazov metode PCA II.

6.5 Povzetek poglavja

Metode osnovane na videzu so zelo razširjene na področju prepoznavanja obrazov. Njihova skupna lastnost je, da predstavijo slike obrazov v obraznem prostoru, ki ima nižjo dimenzijo kot originalni prostor. V tako definiranem obraznem prostoru je posamezna slika obraza predstavljen s koeficienti, ki določajo linearno kombinacijo lastnih obrazov.

S pomočjo metode PCA definiramo takšen podprostor v katerem predstavljeni podatki ohranjajo največjo varianco. Vendar tako preslikani podatki ne upoštevajo kateremu razredu pripada posamezen podatek. To pomanjkljivost reši metoda LDA. S pomočjo metode LDA upoštevamo tako razpršenost razreda samega kot tudi razpršenost med razredi. Metoda ICA pa dodatno pri definiranju podprostora upošteva poleg odvisnosti drugega reda tudi višje nivojske odvisnosti. Slike obrazov lahko predstavimo po prvi arhitekturi (ICA I) s pomočjo linearno neodvisnimi lastnimi slikami ali po drugi arhitekturi (ICA II) s pomočjo linearno neodvisnimi koeficienti. Iz vsake arhitekture ICA I in ICA II smo izločili metodi PCA (PCA I in PCA II), ki sta služili za prepripravo podatkov pred procesom ločevanja iskanja neodvisnih komponent. Metoda PCA I določa navadno metodo PCA na vertikalno centriranih podatkih, metoda PCA II pa pobeljeno metodo PCA na horizontalno centriranih podatkih.

Poglavje 7

Metoda Gaborjevih valčkov

7.1 Uvod

Za razliko od metod osnovanih na videzu, ki za predstavitev obrazov upoštevajo celoten videz, pa metoda na osnovi Gaborjevih valčkov upošteva samo določene dele slike. Zaradi tega to metodo uvrščamo v skupino metod na podlagi značilk.

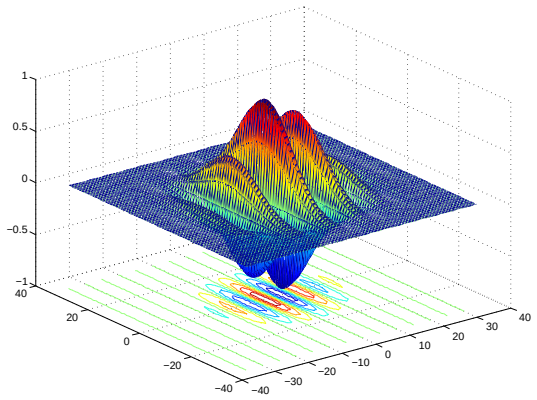
Metoda valčkov (angl. *wavelets*) predstavlja način razčlenitve kompleksnega signala v vsoto baznih funkcij. V tem pogledu je podobna Fourier-jevi analizi, vendar z bistveno razliko. S Fourier-jevimi funkcijami lahko točno določimo frekvenco signala, ne moramo pa točno določiti prostora, kjer se pojavi. To pomeni, da bodo že majhne spremembe v Fourier-jevi transformaciji povzročile spremembe vsepovsod v časovni domeni. S pomočjo valčkov pa lahko določimo tako čas, kot tudi frekvenco pojavljajočega se signala. Čas oziroma koordinate v 2D prostoru določimo z različnimi orientacijami valčka, frekvenco pa z različnimi velikostmi valčka. Zaradi tega lahko s pomočjo valčkov veliko bolje analiziramo podatke različnih velikosti, kot bi to naredili s sinusnimi in kosinusnimi funkcijami. Tako na primer impulz, ki je zelo lokalno določen, zelo težko modeliramo s pomočjo vsote neskončnih funkcij. Medtem ko so funkcije, ki so že v osnovi lokalne, nadvse primerne za to nalogo. Tako potrebujemo za predstavitev ostrih impulzov in nezveznosti manj baznih valčkov, kot bi za to potrebovali baznih funkcij na osnovi sinusov in kosinusov.

7.2 Gaborjevi valčki

Znano je, da je človeški zaznavni sistem predvsem dovzeten na poudarjene robove na slikah. Psihološke študije so našle v človeških možganih preproste celice, ki se posebej odzivajo tako na orientacijo slikovnih značilk, kot tudi na spremembe prostorske frekvence slik. Dovzetnost teh celic lahko približno opišemo s pomočjo dvodimenzionalnih Gaborjevih filtrov. Gaborjeve funkcije je prvi predlagal Dennis Gabor, kot orodje za iskanje signala v šumu. Gaborjeve valčke pa je v računalniškem vidu prvi uporabil Daugman leta 1980. Na osnovi te predstavitve so avtorji Manjunath,

Chellappa in Malsburg razvili sistem za prepoznavo obrazov [63]. Delo je nadaljeval Wiskott et al. [108], ki je predstavil elastičen graf množic (angl. *Elastic Bunch Graph*, EBG), ki so ga sestavljali koeficienti odzivov Gaborjevih filtrov. Lades et al. [55] pa je Gaborjeve valčke uporabil v dinamični povezovalni arhitekturi (angl. *Dinamic Link Architecture*, DLA). Gaborjeve filtre se danes velikokrat uporablja v najrazličnejših metodah na področju računalniškega vida.

Dvodimenzionalne Gaborjeve funkcije predstavljajo obrise poudarjenih robov, kot tudi obrise dolin in grebenov na slikah. Na sliki obraza tako poudarijo najpomembnejše obrazne značilke kot so oči, usta in konica nosu. Poudarijo pa tudi kožna znamenja, lične jamice in brazgotine. Gaborjevi valčki imajo obliko valovite ravnine omejene z Gaussovo ovojno funkcijo (slika 7.1). Odziv Gaborjevega valčka na sliki predstavlja njena konvolucija z Gaborjevim filtrom različnih prostorskih frekvenc/velikosti in orientacij/zasukov.



Slika 7.1: Primer predstavitve Gaborjevega valčka v 3D prostoru.

S pomočjo Gaborjevih valčkov dobimo tako amplitudo kot tudi fazo, ki opisujeta prostorsko frekvenčno strukturo in prostorske relacije. Najenostavnejša tehnika za primerjavo različnih slik s pomočjo Gaborjevih valčkov je, da prekrijemo sliko z mrežo in v vsakem vozlišču izračunamo množico Gaborjevih filtrov različnih orientacij in frekvenc. Primerjava tako dobljenih konvolucijskih koeficientov zna biti zelo težavna. Zaradi zasuka faze se lahko tako dobljeni koeficienti zelo razlikujejo, čeprav so narazen samo nekaj slikovnih elementov in celo predstavljajo isto lokalno značilko obraza. Zaradi tega pri sami primerjavi, faze raje ne upoštevamo ali izravnamo njeno odstopanje. Fazno informacijo potrebujemo, če hočemo ločevati med dvema vzorcema, ki imata podobne amplitude. Ker se faza zelo hitro spreminja, je uporabna tudi za natančnejšo lociranje Gaborjevih filtrov na določeno mesto na sliki.

7.2.1 Matematična definicija Gaborjevih valčkov

Da bi lažje razumeli 2D Gaborjeve valčke, ki se uporabljajo za analizo slik, moramo najprej spoznati osnovne pojme valčne in Fourier-jeve analize. Obe analizi se

uporabljata za analiziranje frekvenčnega prostora slik. Glavna razlika med njima je v tem, da delujejo valčki na določenem delu slike, medtem ko deluje Fourier-jeva analiza na celotni sliki.

Fourier-jeva transformacija razčleni signal tako, da ga lahko predstavimo kot kombinacijo sinusoidov. Pri procesiranju signalov je analiza frekvenčnega prostora zelo pomembna, ker lahko z njeno pomočjo odkrijemo veliko informacij o signalu, ki iz samih podatkov niso očitne. Enodimenzionalno Fourier-jevo transformacijo funkcije x izračunamo s pomočjo enačbe:

$$F(x(t))(\omega) = \int_{-\infty}^{\infty} x(t)e^{-i\omega t}\mathrm{d}t \ . \tag{7.1}$$

S pomočjo transformacije dobimo novo funkcijo, ki je odvisna od frekvence in ne več od časa. Kaj se s funkcijo dogaja vidimo lažje, če transformacijo razbijemo na njen realni in imaginarni del:

$$F(x(t))(\omega) = \int_{-\infty}^{\infty} x(t)\cos(\omega t)(d)t - i\int_{-\infty}^{\infty} x(t)\sin(\omega t)(d)t \ . \tag{7.2}$$

Vidimo lahko, da Fourier-jeva transformacija razčleni funkcijo v sinusne in kosinusne funkcije.

Z izračunom vrednosti funkcije $F(x(t))(\omega)$ pri dani frekvenci, dobimo kompleksno število, ki ustreza amplitudi kosinusnega (realnega) in sinusnega (imaginarnega) dela originalne funkcije pri tej frekvenci.

S seštevkom vseh kosinusnih in sinusnih valov vseh frekvenc lahko rekonstruiramo originalni signal. Ta postopek imenujemo tudi inverzna Fourier-jeva transformacija, ki jo izračunamo s pomočjo enačbe:

$$x(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} F(x(t))(\omega)e^{i\omega t}(d)\omega \ . \tag{7.3}$$

Kadar izvedemo Fourier-jevo transformacijo realne funkcije, je rezultat kompleksno število. Za vsako frekvenco dobimo realno ($a_{\omega,r}$) in imaginarno ($a_{\omega,i}$) komponento originalne funkcije:

$$x_\omega(t) = a_{\omega,r}\cos(t) + a_{\omega,i}\sin(t) \ . \tag{7.4}$$

Če obe komponenti seštejemo, dobimo nov sinusoid, ki vsebuje vse informacije te frekvence. Tako dobljeni sinusoid lahko predstavimo tudi samo s kosinusno funkcijo z ustrezno amplitudo in fazo:

$$x_\omega(t) = a_{\omega,r}\cos(t) + a_{\omega,i}\sin(t) = a\cos(t+\phi) \ . \tag{7.5}$$

Tako definirana enačba transformira Fourier-jevo informacijo iz kartezijske predstavitve v polarno predstavitev. Nova predstavitev določa velikost in fazni kot sinusoida.

Sedaj, ko smo spoznali osnove Fouriere-jeve teorije, lahko definiramo Gaborjeve valčke. Valčki so v veliko pogledov podobni Fourier-jevi analizi, čeprav imajo omejeno območje. Gaborjevi valčki so v osnovi sinusoidi pomnoženi z Gaussovo krivuljo. Kadar izračunamo konvolucijo funkcije z Gaussovim valčkom, se frekvenčna informacija blizu centra Gaussove krivulje ohrani. Frekvenčna informacija daleč proč od centra pa ima zanemarljiv efekt.

Enačbo, ki definira enodimenzionalni Gaborjev valček lahko zapišemo kot:

$$W(t, t_0, \omega) = e^{-\sigma(t-t_0)^2} e^{-i\sigma(t-t_0)} \ . \tag{7.6}$$

Konvolucijo za valčno transformacijo pa definiramo kot:

$$C(x(t)(t_0, \omega)) = \int_{-\infty}^{\infty} x(t) W(t, t_0, \omega) \mathrm{d}t \ . \tag{7.7}$$

Če valčno funkcijo razširimo, je konvolucija zelo podobna Fourier-jevi transformacij:

$$C(x(t))(t_0, \omega) = \int_{-\infty}^{\infty} x(t) e^{-\sigma(t-t_0)^2} e^{i\omega(t-t_0)} \tag{7.8}$$

$$C(x(t))(t_0, \omega) = \int_{-\infty}^{\infty} x(t) e^{-\sigma(t-t_0)^2} \cos(\omega(t-t_0)) + i \int_{-\infty}^{\infty} x(t) e^{-\sigma(t-t_0)^2} \sin(\omega(t-t_0)) \tag{7.9}$$

Integral tukaj določa kompleksni koeficient $C(x(t))(t_0, \omega)$, ki opisuje lokalno frekvenčno informacijo funkcije x določene frekvence ω in časa t_0. Podobno kot Fourier-jeva transformacija ima tudi tako določen valčni koeficient realni in imaginarni del, ki pripadata kosinusni in sinusni funkciji:

$$C(x(t))(t_0, \omega) = a_{real} + i a_{imag} \ . \tag{7.10}$$

Edina razlika je v tem, da so v tem primeru sinusoidi pomnoženi še z Gaussovo funkcijo. Ob upoštevanju enačbe 7.5 lahko kompleksni koeficient predstavimo tudi v polarnih koordinatah tako, da a predstavlja velikost in ϕ fazni kot:

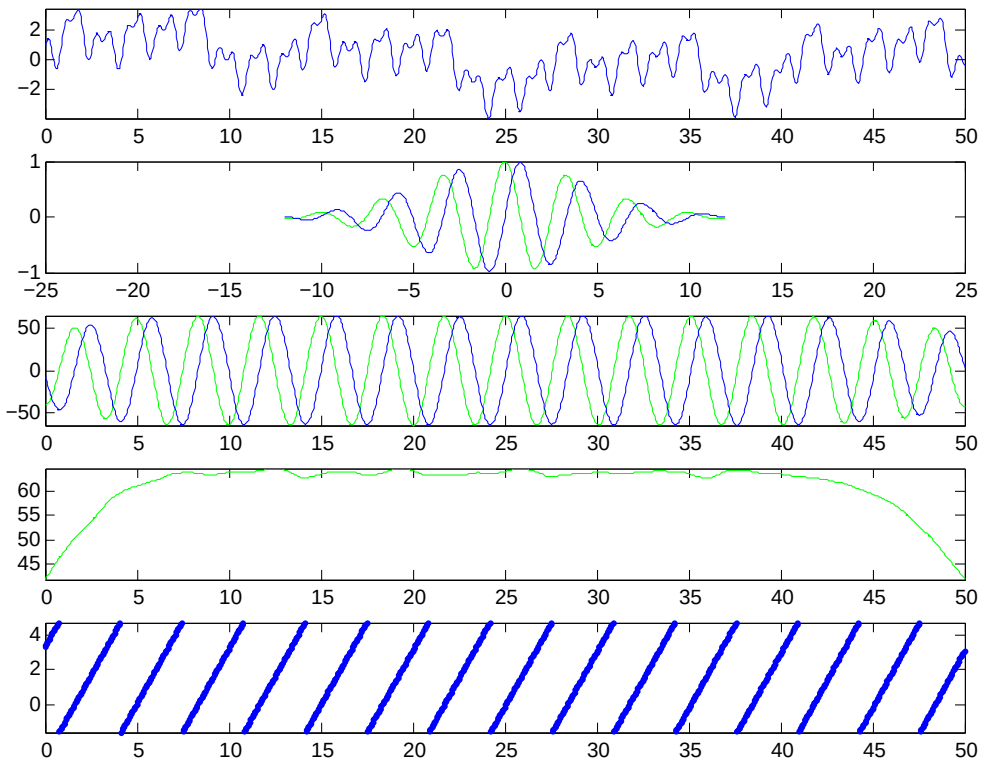
$$a_{real} = a \cos \phi \tag{7.11}$$

$$a_{imag} = a \sin \phi \tag{7.12}$$

Velikost a in fazni kot ϕ pa določimo s pomočjo enačb:

$$a = \sqrt{a_{real}^2 + a_{imag}^2} \tag{7.13}$$

$$\phi = \begin{cases} \arctan(a_{imag}/a_{real}) \,, & \text{če je } a_{real} > 0 \,, \\ \pi + \arctan(a_{imag}/a_{real}) \,, & \text{če je } a_{real} < 0 \,, \\ \pi/2 \,, & \text{če je } a_{real} = 0 \text{ in } a_{imag} \geq 0 \,, \\ -\pi/2 \,, & \text{če je } a_{real} = 0 \text{ in } a_{imag} < 0 \ . \end{cases} \tag{7.14}$$



Slika 7.2: Primer konvolucije z Gaborjevim valčkom.

Pomembno je tudi vedeti, kako sta med seboj povezana časovni zamik in fazni kot koeficienta. Če izračunamo valčne koeficiente za dve različni točki, ki sta med seboj zelo blizu, je fazni kot koeficientov sorazmeren zamiku teh dveh konvolucijskih točk. Poglejmo si to na primeru.

Slika 7.2 prikazuje primer funkcije sinusoida in njen rezultat konvolucije z Gaborjevim valčkom. Prva vrstica prikazuje graf funkcije, sestavljene iz desetih naključnih sinusoidov. Sestavljena je iz vsote sinusnih in kosinusnih valov različnih frekvenc z naključno amplitudo v območju od -1 do 1. Druga vrstica prikazuje funkcijo Gaborjevega valčka in sicer njen realni (zelen graf) in imaginarni (moder graf) del. Frekvenca Gaborjevega valčka je enaka eni izmed frekvenc osnovne funkcije. Tretji graf prikazuje rezultat konvolucije valčka za vsako točko originalne funkcije. Vidimo lahko odziv valčka na sinusoid originalne funkcije: kadar fazi funkcije in valčka sovpadati je vrednost konvolucije visoka, kadar pa fazi nista sinhronizirani za vrednost π potem je konvolucijska vrednost negativna.

Predzadnji graf prikazuje vrednost velikosti odziva valčka. Vidimo, da je energija izbrane frekvenčne komponente funkcije konstantna blizu centra same funkcije. Padec velikosti ob straneh je pričakovan, ker funkcija na robovih ni definirana.

Zadnji graf prikazuje fazo koeficientov valčka. Vidimo lahko, da faza koeficientov zaključuje zaključen cikel med periodami sinusoida.

Dvodimenzionalni Gaborjevi valčki

V prejšnjem razdelku smo definirali enodimenzionalne Gaborjeve valčke. Na slikah pa se uporablja dvodimenzionalno obliko Gaborjevih valčkov. Dvodimenzionalni valček je sestavljen iz ravninskega sinusoida pomnoženega z dvodimenzionalnim Gaussom. Sinusni val se odziva na frekvenčno informacijo slike. Gaussova porazdelitev pa zagotavlja, da prevlada konvolucija tistih regij na sliki, ki so bližje centru valčka. V tem poglavju bomo spoznali različne parametre, s katerimi lahko opišemo Gaborjev valček.

Gaborjevi valčki lahko zavzamejo različne oblike. S pomočjo različnih parametrov pa lahko valčku spreminjamo orientacijo, frekvenco, fazo, velikost in razmerje valčka. Za natančen opis frekvenčne informacije značilke na sliki je potrebno izračunati konvolucije dane lokacije značilke z valčki različnih oblik. Različne oblike določajo predvsem različne frekvence in orientacije.

Slika 7.3 prikazuje rezultat konvolucije slike obraza z realnim in imaginarnim delom valčka. Dvodimenzionalni Gaborjev valček se odzove na slikovne značilke, ki imajo enako orientacijo in frekvenco, kot jih ima sam valček. Slika prikazuje originalno sliko in dve maski Gaborjevega valčka (Gaborjeva filtra), ki se uporabljata za konvolucijo. Sliki v zadnji vrstici pa prikazujeta rezultat velikosti in fazo konvolucije za posamezno točko slike.

Na sliki 7.3(c) lahko vidimo, da se je valček odzval na obrazne značilke, kot so oči in usta. Vidimo tudi, da velikost odziva pojema zelo počasi.

Posamezna maska je dvodimenzionalno polje, ki se uporablja za konvolucijo nad sliko. Najprej se masko centrira nad dano lokacijo, nato se med seboj zmnožijo slikovni elementi, ki se prekrivajo. Tako zmnoženi elementi se nato seštejejo in njihova vsota predstavlja konvolucijsko vrednost dane lokacije. Da dobimo realni in imaginarni del valčka je potrebno izračunati konvolucijo z dvema maskama, ki imata fazi zamaknjeni za $\pi/2$. To ravno ustreza uporabi sinusa in kosinusa valčne transformacije.

Definirajmo enačbo dvodimenzionalnega Gaborjevega valčka z vsemi parametri kot:

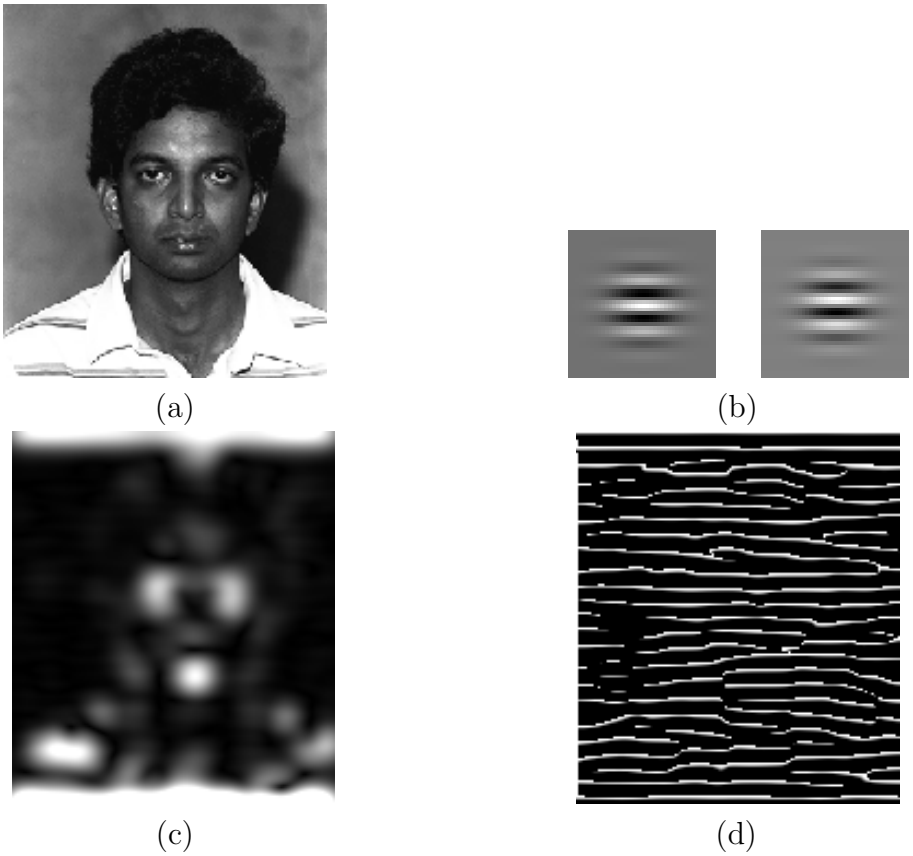
$$W(x, y, \theta, \lambda, \varphi, \sigma, \gamma) = e^{-\frac{x\prime^2+\gamma^2 y\prime^2}{2\sigma^2}} \cos(2\pi \frac{x\prime}{\lambda} + \varphi) \tag{7.15}$$

$$x\prime = x\cos\theta + y\sin\theta \tag{7.16}$$

$$y\prime = -x\sin\theta + y\cos\theta \tag{7.17}$$

S pomočjo petih parametrov θ, λ, φ, σ, γ lahko popolnoma kontroliramo obliko valčka. V nadaljevanju bomo opisali posamezen parameter in na primeru pokazali, kako se valček spreminja pri različnih vrednostih parametra.

- Parameter θ definira orientiranost valčka. S pomočjo tega parametra lahko zasukamo valček okrog svojega centra. Orientacija valčka definira kot robov oziroma prog, na katere se bo valček odzval. V večini primerov je vrednost

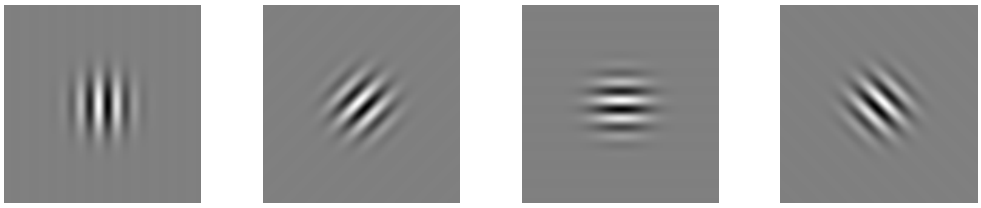


Slika 7.3: Primer konvolucije slike z dvodimenzionalnim Gaborjevim valčkom: (a) originalna slika obraza, (b) maski Gaborjevega valčka, (c) velikost konvolucije in (d) faza konvolucije.

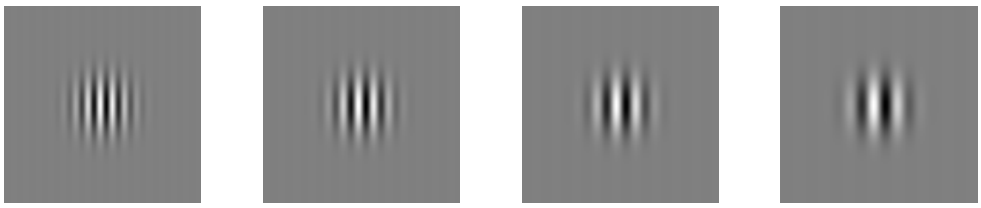
parametra od 0 do π. Vrednosti od π do 2π so enakovredne zaradi simetrije valčka. Kot primer vzemimo valček, ki bazira na kosinusu, tako da je $\varphi = 0$. Takšen valček je simetričen glede na izhodišče. V tem primeru, če se vrednosti parametrov θ_i za posamezna valčka razlikujeta za π (na primer $\theta \in \{0, \pi\}$), bosta takšna valčka identična. Če pa po drugi strani vzemimo valček, ki bazira na sinusu ($\varphi = \pi/2$), je vrednost konvolucije enako velika ampak z nasprotnim predznakom. Slika 7.4 prikazuje primere valčkov za vrednosti parametrov $\theta \in \{0, \pi/4, \pi/2, 3\pi/4\}$

- Parameter λ definira valovno dolžino kosinusnega vala oziroma frekvenco valčka. Valčki z veliko valovno dolžino se bodo odzvali na postopne spremembe v intenziteti slike. Valčki s kratkimi valovnimi dolžinami pa se bodo odzvali na ostre robove ali proge na sliki. Štirje različni valčki na sliki 7.5 prikazujejo efekt, ko parameter λ počasi narašča iz 8 na 16 slikovnih elementov.

- Parameter φ definira fazo sinusoida. Tipični Gaborjevi valčki bazirajo na sinusnih ali kosinusih valih. Kosinusni valčki predstavljajo realni del valčka in

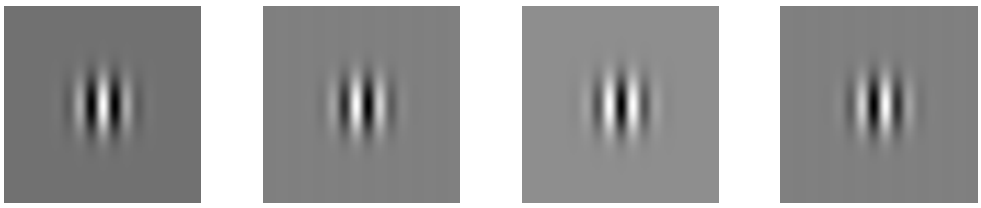


Slika 7.4: Primeri valčkov različnih orientacij.



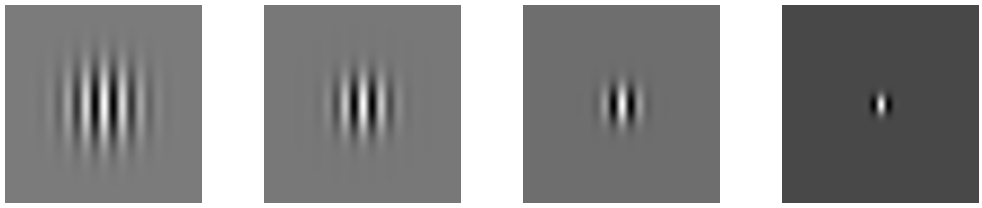
Slika 7.5: Primeri valčkov različnih valovnih dolžin.

sinusni valčki imaginarni del. Potemtakem rezultat konvolucije z obema fazama vrne kompleksen koeficient. Osnovna matematična definicija zahteva kompleksen koeficient, ki bazira na dveh valčkih, ki imata fazni zamik za $\pi/2$, to pomeni $\varphi \in \{0, \pi/2\}$. Pravzaprav lahko uporabimo katerikoli par, samo da se fazi realnega in imaginarnega dela valčka razlikujeta za $\pi/2$. Zaradi tega vrednost parametra φ večinoma določimo kot vrednost 0 ali $\pi/2$. Slika 7.6 prikazuje primere valčkov z vrednostmi parametra $\varphi \in \{0, \pi/2, \pi, 3\pi/2\}$. Konvoluciji z zadnjima dvema filtroma sta, zaradi istih razlogov, kot smo jih opisali pri parametru θ, odvečni.



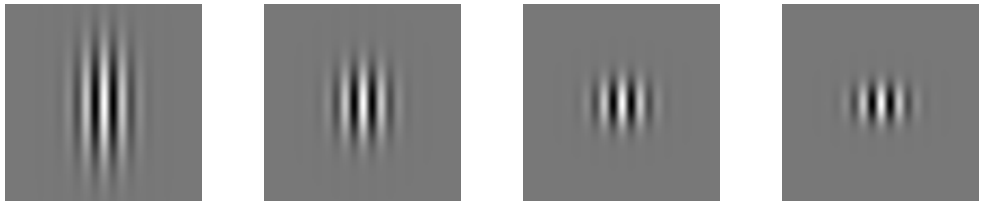
Slika 7.6: Primeri valčkov z različnimi fazami.

- Parameter σ definira radij Gaussove funkcije v prostoru. Velikost Gaussove funkcije včasih označimo tudi kot podporo valčni bazi. Z velikostjo Gaussa določimo velikost dela slike, nad katerim se bo izvršila konvolucija. Teoretično bi morala celotna slika učinkovati na konvolucijo, a ko se s konvolucijo premaknemo iz centra, je preostalo računanje nepomembno. Ta parameter je običajno sorazmeren z valovno dolžino, tako da so valčki različnih velikosti in frekvenc samo njihova povečana različica: $\sigma = c\lambda$. Tudi velikost maske je tesno povezana z velikostjo Gaussa: izbrana je tako, da zajame pomembnejši del Gaussa. Slika 7.7 prikazuje valčke pri parametrih $\sigma \in \{16, 12, 8, 4\}$.



Slika 7.7: Primeri valčkov z različnimi velikostmi Gaussa.

- Parameter γ določa razmerje velikosti Gaussa. Ta parameter smo vključili zato, da se lahko s pomočjo valčkov približamo tudi biološkim modelom. Večinoma pa valčki uporabljajo vrednost razmerja enako 1. Slika 7.8 prikazuje štiri različna razmerja v območju med 0.5 do 1.5.



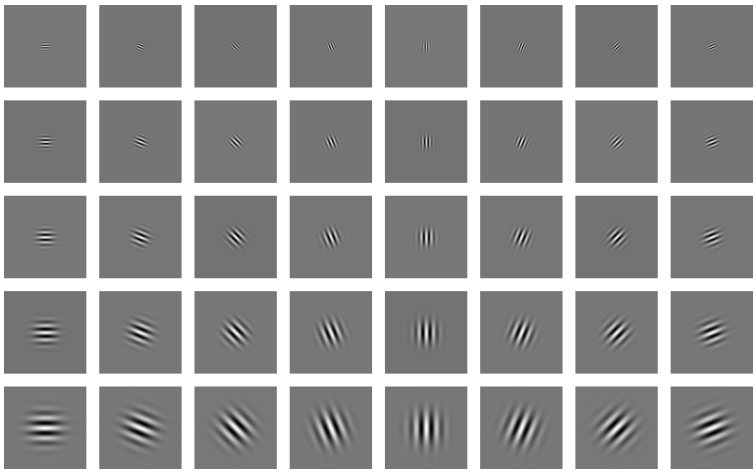
Slika 7.8: Primeri valčkov z različnimi razmerji Gaussa.

7.3 Uporaba Gaborjevih filtrov za prepoznavanje obrazov

V prejšnjih poglavjih smo opisali postopke za prepoznavanje obrazov s pomočjo metod na osnovi celotnega videza obraza, kot so PCA, LDA in ICA. Te metode spadajo med statistične metode, ki potrebujejo za svoje delovanje učno množico s pomočjo katere jih predhodno naučimo. Metoda Gaborjevih valčkov pa se od teh metod bistveno razlikuje po tem, da za svoje delovanje ne potrebuje učne množice. Ker deluje samo nad posameznimi deli slike, jo uvrščamo v skupino metod na osnovi značilk in geometrije. Za to skupino metod je značilno, da izluščijo iz slike značilke, s pomočjo katerih nato predstavimo sliko. Z metodo Gaborjevih valčkov dobimo značilke v posamezni točki (x,y) slike I s pomočjo konvolucije z Gaborjevimi maskami oziroma filtri:

$$O(x, y, \theta, \lambda) = I(x, y) * W(x, y, \theta, \lambda) \ , \tag{7.18}$$

kjer z $*$ označujemo konvolucijski operator in z $O(x, y, \theta, \lambda)$ rezultat konvolucije z Gaborjevim jedrom pri določeni orientaciji θ in velikosti λ. Definirajmo množico $\mathcal{S}$ različnih Gaborjevih valčkov, s katerimi bomo predstavili sliko $I(x, y)$: $\mathcal{S}$=$\{O(x, y, \theta, \lambda)$: $\theta \in \{0, \pi/8, 2\pi/8, 3\pi/8, 4\pi/8, 5\pi/8, 6\pi/8, 7\pi/8\}$, $\lambda \in \{4, 4\sqrt{2}, 8, 8\sqrt{2}, 16\}\}$.



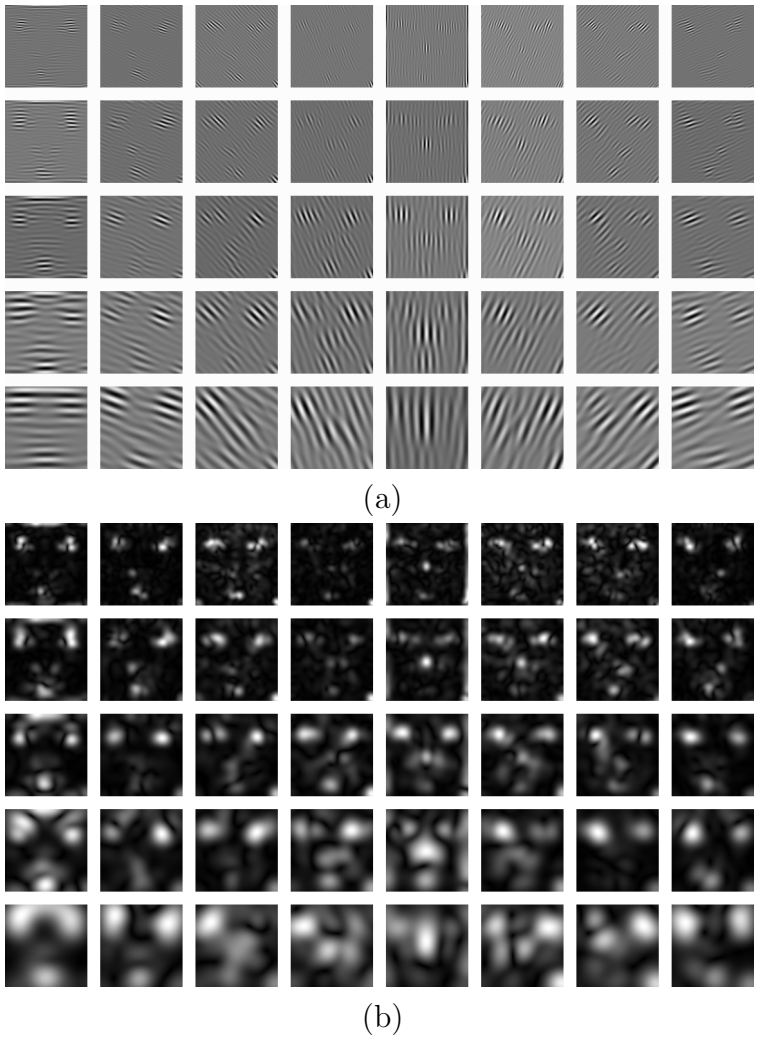
Slika 7.9: Realni del množice Gaborjevih valčkov petih različnih velikosti in osmih orientacij.

Slika 7.9 prikazuje tako definirano množico $\mathcal{S}$ različnih Gaborjevih valčkov pri vrednosti parametra $\varphi = 0$, ki določa realni del kompleksnega koeficienta oziroma kosinusni del valčka. Za popolno predstavitev slike pa moramo izračunati še lihi del valčka, ki določa imaginarni del koeficienta ($\varphi = pi/2$). Tako dobimo pri osmih različnih velikostih ($\theta \in \{0, \pi/8, 2\pi/8, 3\pi/8, 4\pi/8, 5\pi/8, 6\pi/8, 7\pi/8\}$), štirih različnih orientacijah ($\lambda \in \{4, 4\sqrt{2}, 8, 8\sqrt{2}, 16\}$) in dveh fazah ($\varphi = \{0, pi/2\}$), $5*8*2 = 80$ različnih Gaborjevih valčkov. Na sliki 7.10(a) lahko vidimo rezultat konvolucije slike obraza z realnim delom Gaborjevih filtrov. Slika 7.10(b) pa prikazuje magnitudo konvolucije, kjer sta upoštevana realni in imaginarni del (enačba 7.13).

Za predstavitev slike $\mathbf{x}$ velikosti 60×50 slikovnih elementov torej potrebujemo vektor značilk velikosti 60*50*80=240.000 slikovnih elementov, kar pa je veliko preveč. Zaradi tega metode, ki predstavijo slike s pomočjo Gaborjevih valčkov, izračunajo konvolucijo samo v določenih točkah. Najenostavneje določimo točke tako, da čez sliko postavimo mrežo in stičišča črt nam tako predstavljajo točke. Koliko točk bo predstavljajo sliko, določimo z gostoto mreže. Predpostavimo, da imamo gostoto mreže 8 slikovnih elementov. Takšna mreža določa 56 različnih točk na sliki velikosti 60×50 slikovnih elementov. Torej lahko vsako sliko predstavimo z vektorjem značilk velikosti 56*80=4.480 koeficientov. Takšen vektor značilk za primer obraza prikazuje slika 7.11.

Vektor raje predstavimo v obliki posameznih slik, kot v vektorski obliki. Posamezni rezultat konvolucije slike za posamezen Gaborjev filter smo pred združitvijo v vektor značilk zmanjšali za določen faktor s pomočjo mreže in ga normalizirali tako, da smo odšteli povprečje vseh elementov in delili s standardno deviacijo vseh elementov vektorja značilk.

Ko imamo tako zgrajene vektorje značilk za vsako posamezno sliko obraza, lahko za vse slike izvedemo primerjavo neznanih slik z znanimi slikami galerije. Pri konvo-



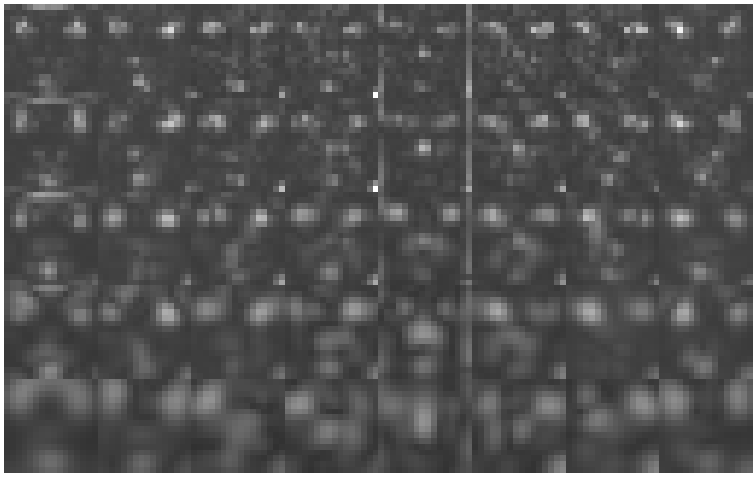
Slika 7.10: Predstavitev slike obraza s pomočjo Gaborjevih valčkov: (a) realni del predstavitve in (b) magnituda predstavitve.

luciji dobimo kot rezultat velikost a in fazo ϕ (enačba 7.13). Ker pa se faza spreminja zelo hitro z lokacijo, jo za samo primerjavo izpustimo in upoštevamo samo velikost a:

$$a(x,y) = \sqrt{O(x,y,\theta,\lambda,\varphi=0)^2 + O(x,y,\theta,\lambda,\varphi=\pi/2)^2} \ . \tag{7.19}$$

Tako lahko vsak vektor značilk predstavimo s pomočjo 40 vrednosti. Za primerjavo med njimi pa lahko uporabimo različna merjenja razdalj, ki so predstavljena v poglavju 8. Običajno pa se za primerjavo uporablja kosinusno razdaljo:

$$d(O_{I_1}, O_{I_2}) = 1 - \frac{\sum_{i=1}^{M} a_{I_1} a_{I_2}}{\sqrt{\sum_{i=1}^{M} a_{I_1}^2 \sum_{i=1}^{M} a_{I_2}^2}} \ , \tag{7.20}$$



Slika 7.11: Primer Gaborjevega vektorja značilk za sliko obraza.

kjer O_I določa rezultat konvolucije družine Gaborjevih filtrov nad sliko I in a_I velikost konvolucije v posamezni točki. Vrednost M označuje velikost vektorja značilk.

7.4 Povzetek poglavja

Metoda Gaborjevih valčkov je na področju prepoznavanja obrazov ali splošneje na področju razpoznavanja vzorcev zelo uspešna. S pomočjo družine Gaborjevih valčkov poudarimo na sliki robove različnih orientacij in velikosti. Psihološke študije so pokazale, da je človeški zaznavni sistem za takšne spremembe še posebej dovzeten. Tako poudarjene značilke, ki jih izločimo iz slike, določajo novo predstavitev slike. Prednost Gaborjeve metode, pred metodami osnovanimi na videzu je, da ta metoda ne potrebuje predhodnega učenja. Torej lahko sistemu za prepoznavanje obrazov dodajamo nove osebe brez, da bi morali predhodno sistem ponovno naučiti. Ker deluje ta metoda na posameznih značilkah in ne na celotnem videzu obraza, je metoda tudi manj odvisna od različne osvetlitve obraza. Po drugi strani pa je metoda Gaborjevih valčkov močno odvisna od različnih izrazov na obrazu, kar pa metode osnovane na videzu s pomočjo učnega postopka delno omilijo.

Metoda je bila uspešno uporabljena tudi v grafu, kjer je vsako vozlišče določalo del obraza različnih oseb, predstavljen z družino Gaborjevih valčkov [107]. Takšen graf so poimenovali elastičen graf množic, ker se lahko prilagaja obraznim značilkam na podlagi izračunanih vrednosti Gaborjevih valčkov. Tako poravnana mreža se nato uporablja za primerjavo in s tem za prepoznavo neznanega obraza. V tem primeru se za primerjavo uporabijo samo točke v vozliščih in ne več enakomerno razdeljene točke mreže, kot smo te definirali v našem primeru. Prednost tako predstavljenega obraza s pomočjo elastičnega grafa je tudi v tem, da posamezno osebo predstavimo z različnimi značilkami, kot na primer z odprtimi/zaprtimi usti, odprtimi/zaprtimi očmi. Tako definirana metoda temelječa na mreži značilk je tudi manj občutljiva na

različne spremembe na obrazu, kot so na primer različni izrazi.

Poglavje 8

Različne mere razdalj metode najbližjega soseda

8.1 Uvod

V poglavjih 6 in 7 smo spoznali različne metode, s pomočjo katerih dobimo iz slik obrazov njene značilne točke ali značilke, ki to sliko najbolje predstavljajo, oziroma slike med seboj najbolje razlikujejo. Pri metodah osnovanih na videzu, smo takšno predstavitev dobili s pomočjo učnega postopka, pri metodi Gaborjevih filtrov pa s pomočjo konvolucije množice različnih filtrov na izbranih lokacijah slike. Tako definirane značilke določala vektor značilk, s pomočjo katerega smo predstavili vse slike obrazov. Po učenju smo najprej predstavili slike znanih obrazov galerije in jih predhodno shranili. Pred postopkom identifikacije pa smo na enak način predstavili tudi vhodno sliko neznanega obraza.

Pri postopku verifikacije moramo primerjati vhodni vektor značilk neznanega obraza z vektorjem obraza znane osebe, za katero se oseba izdaja. Na podlagi podobnosti potem sporočimo ali je ujemanje dovolj veliko, da je oseba res ta, za katero se izdaja. Pri nalogi identifikacije pa moramo primerjati vhodno sliko obraza z vsemi znanimi obrazi iz galerije. Na podlagi podobnosti izberemo tisto osebo, pri kateri je podobnost z vhodno sliko največja. Takšen postopek se imenuje izbira na podlagi najbližjega soseda (angl. *nearest-neighbour matching*). Ker je dimenzija vektorja značilk, s katerim je predstavljena slika, običajno veliko manjša od dimenzije originalne slike, je takšna primerjava računsko tudi učinkovitejša. Ker vsebujejo vektorji s katerimi so predstavljeni obrazi najpomembnejše značilke, je takšen postopek tudi uspešnejši, kot če bi med seboj primerjali vse slikovne elemente slike.

Predpostavimo, da imamo vektorja značilk $\mathbf{x}$ in $\mathbf{y}$ dolžine M. Funkcija $d(\mathbf{x}, \mathbf{y})$ pa bo označevala razdaljo med njima. Čim manjša bo razdalja med njima, tem bolj sta si podobna oziroma tem bolj sta si podobni pripadajoči originalni sliki obraza.

Seveda lahko za merjenje razdalj (funkcija d) izberemo različne mere. Različne mere razdalj nam vrnejo različne rezultate in s tem tudi različne vrednosti ujemanja. Zaradi tega lahko s pomočjo različnih mer razdalj dobimo različne uspešnosti

prepoznavanja neznanega obraza. V nadaljevanju bomo predstavili različne mere za merjenje razdalj, ki se lahko uporabljajo pri metodi najbližjega soseda.

8.1.1 Različne mere razdalj

Predstavili bomo matematično definicijo posamezne mere razdalj ter oznako, na katero se bomo v nadaljevanju sklicevali.

1. **Razdalja CityBlock (Manhattan) – $\mathbf{L_1}$**

$$d(\mathbf{x}, \mathbf{y}) = |\mathbf{x} - \mathbf{y}| = \sum_{i=1}^{M} |x_i - y_i| \ , \tag{8.1}$$

kjer $|\cdot|$ označuje absolutno vrednost in x_i, y_i i-te komponente vektorja.

2. **Evklidska razdalja (angl.** *Euclidean distance*) **– $\mathbf{L_2}$**

$$d(\mathbf{x}, \mathbf{y}) = ||\mathbf{x} - \mathbf{y}|| = \sqrt{\sum_{i=1}^{M} (x_i - y_i)^2} \ , \tag{8.2}$$

kjer $||\cdot||$ označuje normo. Evklidska razdalja in razdalja Cityblock sta samo poseben primer **razdalje Minkowski**:

$$d(\mathbf{x}, \mathbf{y}) = ||\mathbf{x} - \mathbf{y}||_p = \sqrt[p]{\sum_{i=1}^{M} |x_i - y_i|^p} \ ,$$

pri katerih imamo pri razdalji Cityblock parameter $p = 1$ in pri Evklidski razdalji $p = 2$.

3. **Kosinusna razdalja (angl.** *Cosine distance*) **– COS**

$$d(\mathbf{x}, \mathbf{y}) = 1 - \cos(\mathbf{x}, \mathbf{y}) \ , \tag{8.3}$$

$$\cos(\mathbf{x}, \mathbf{y}) = \frac{\mathbf{x} \cdot \mathbf{y}}{||\mathbf{x}|| \cdot ||\mathbf{y}||} = \frac{\sum_{i=1}^{M} x_i y_i}{\sqrt{\sum_{i=1}^{M} x_i^2 \sum_{i=1}^{M} y_i^2}} \ .$$

Pri tej razdalji se meri velikost kota med dvema vektorjema. Če sta vektorja enaka, je razdalja 1, če pa sta pravokotna je 0.

4. **Korelacijska razdalja (angl.** *Correlation distance*) – **Kor**

$$d(\mathbf{x},\mathbf{y}) = 1 - r(\mathbf{x},\mathbf{y}) \ , \tag{8.4}$$
$$r(\mathbf{x},\mathbf{y}) = \frac{\sum_{i=1}^{M}(x_i-\bar{x})(y-\bar{y})}{M\sqrt{\frac{1}{M}\sum_{i=1}^{M}(x_i-\bar{x})^2}\sqrt{\frac{1}{M}\sum_{i=1}^{M}(y_i-\bar{y})^2}}$$

V primeru, če so vektorji centrirani, to pomeni, da je njihova povprečna vrednost enaka nič ($\bar{x}=0, \bar{y}=0$), je korelacijska razdalja enaka kosinusni razdalji (enačba 8.3).

Razdalje Mahalanobis

Za razumevanje razdalj Mahalanobis moramo predhodno razumeti transformacijo med slikovnim prostorom in prostorom Mahalanobis. S pomočjo metode PCA dobimo poleg baznih vektorjev še varianco podatkov za vsako dimenzijo posebej. Kot izhod iz metode PCA dobimo lastne vektorje, ki določajo takšno rotacijo v prostoru, da je skupna kovarianca med dimenzijami enaka nič. Poleg tega dobimo še lastne vrednosti, ki določajo varianco vzdolž posamezne dimenzije.

Prostor Mahalanobis je definiran kot prostor kjer so variance vzdolž posamezne dimenzije enake ena. Da to dosežemo, izvedemo transformacijo vektorja iz slikovnega prostora v prostor značilk tako, da posamezne koeficiente vektorja delimo z njeno standardno deviacijo. S pomočjo takšne transformacije dobimo prostor značilk, ki ima za posamezno dimenzijo enotsko varianco.

Za izračun razdalje Mahalanobis moramo torej namesto vektorjev $\mathbf{x}$ in $\mathbf{y}$ uporabiti njima pripadajoča vektorja $\mathbf{u}$ in $\mathbf{v}$, ki pripadata tako definiranemu prostoru Mahalanobis. Vektorja $\mathbf{u}$ in $\mathbf{v}$ pa določimo kot:

$$u_i = \frac{x_i}{\sigma_i} = \frac{x_i}{\sqrt{\lambda_i}}; \qquad v_i = \frac{y_i}{\sigma_i} = \frac{y_i}{\sqrt{\lambda_i}} \ ,$$

kjer σ_i označuje standardno deviacijo vzdolž posamezne dimenzije in je enaka korenu variance, ki jo določajo lastne vrednosti λ_i. Ta postopek imenujemo tudi beljenje. Torej lahko izračunamo posamezne različice razdalje Mahalanobis tudi tako, da podatke predhodno pobelimo. Tako poznamo Mahalanobis Cityblock razdaljo, Mahalanobis Evklidsko razdaljo, ki ji krajše rečemo kar Mahalanobis razdalja in Mahalanobis kosinusno razdaljo.

Lastne vrednosti λ_i pa si lahko predstavljamo tudi kot uteži, s katerimi obtežimo posamezno dimenzijo. Zaradi tega imenujemo izpeljanke Mahalanobis razdalj tudi obtežene razdalje. Tako lahko prejšnje Mahalanobis razdalje poimenujemo tudi kot: obtežena Cityblock razdalja, obtežena Evklidska razdalja

in obtežena kosinusna razdalja. V nadaljevanju bomo predstavili posamezne različice razdalj Mahalanobis.

5. **Mahalanobis Cityblock razdalja (obtežena Cityblock razdalja) – $\mathbf{wL_1}$**

$$d(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{M} \frac{1}{\sqrt{\lambda_i}} |x_i - y_i| \ , \tag{8.5}$$

6. **Mahalanobis razdalja (Mahalanobis Evklidska razdalja, obtežena Evklidska razdalja) – MAH**

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^{M} \frac{1}{\sqrt{\lambda_i}} (x_i - y_i)^2} \ , \tag{8.6}$$

7. **Mahalanobis kosinusna razdalja (obtežena kosinusna razdalja) – wCOS**

$$d(\mathbf{x}, \mathbf{y}) = 1 - \frac{\sum_{i=1}^{M} \frac{1}{\sqrt{\lambda_i}} x_i y_i}{\sqrt{\sum_{i=1}^{M} x_i^2 \sum_{i=1}^{M} y_i^2}} \ . \tag{8.7}$$

8. **Razdalja Yambor – wYam**

$$d(\mathbf{x}, \mathbf{y}) = 1 - \sum_{i=1}^{M} \frac{1}{\sqrt{\lambda_i}} x_i y_i \tag{8.8}$$

Veliko avtorjev napačno označuje Yambor razdaljo kot Mahalanobis razdaljo. Razdalja Yambor je posplošena razdalja, ki jo je definiral Moon [77]:

9. **Moon-ova razdalja - wMoon**

$$d(\mathbf{x}, \mathbf{y}) = 1 - \sum_{i=1}^{M} \sqrt{\frac{\lambda_i}{\lambda_i - \alpha^2}} x_i y_i \ , \tag{8.9}$$

kjer je namesto vrednosti $\frac{1}{\sqrt{\lambda_i}}$ uporabljena zapletenejša funkcija uteži: $\sqrt{\frac{\lambda_i}{\lambda_i + \alpha^2}}$, kjer je $\alpha = 0.25$.

V primeru, če so vektorji normalizirani (to pomeni, da je njihova dolžina enaka ena), je razdalja Yambor ekvivalentna Mahalanobis kosinusni razdalji (enačba 8.7).

10. **Razdalja Canberra (angl.** *Canberra distance*) – **Can**

$$d(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{M} \frac{|x_i - y_i|}{|x_i| + |y_i|} \quad . \tag{8.10}$$

11. **Spremenjena Cityblock razdalja – sL$_1$**

$$d(\mathbf{x},\mathbf{y}) = \frac{\sum_{i=1}^{M} |x_i - y_i|}{\sum_{i=1}^{M} |x_i| \sum_{i=1}^{M} |y_i|} \quad . \tag{8.11}$$

12. **Spremenjena Evklidska razdalja – sL$_2$**

$$d(\mathbf{x},\mathbf{y}) = \sqrt{\frac{\sum_{i=1}^{M} (x_i - y_i)^2}{\sum_{i=1}^{M} (x_i)^2 \sum_{i=1}^{M} (y_i)^2}} \quad . \tag{8.12}$$

13. **Obtežena spremenjena Cityblock razdalja – wsL$_1$**

$$d(\mathbf{x},\mathbf{y}) = \frac{\sum_{i=1}^{M} \sqrt{\frac{1}{\lambda_i}} |x_i - y_i|}{\sum_{i=1}^{M} |x_i| \sum_{i=1}^{M} |y_i|} \quad . \tag{8.13}$$

14. **Obtežena spremenjena Evklidska razdalja – wsL$_2$**

$$d(\mathbf{x},\mathbf{y}) = \sqrt{\frac{\sum_{i=1}^{M} \sqrt{\frac{1}{\lambda_i}} (x_i - y_i)^2}{\sum_{i=1}^{M} (x_i)^2 \sum_{i=1}^{M} (y_i)^2}} \quad . \tag{8.14}$$

Če imamo vektorje značilk $\mathbf{x}_i$ predhodno shranjene v podatkovni bazi, lahko nekatere komponente: $\sum_{i=1}^{M} x_i^2$, $(\sum_{i=1}^{M} x_i)^2$, $\sum_{i=1}^{M} |x_i|$ omenjenih razdalj izračunamo vnaprej in shranimo v bazo. Na ta način pospešimo primerjanje in iskanje po bazi. Kot smo že povedali pri razdaljah Mahalanobis, lahko namesto, da uporabimo lastne vednosti λ_i kot uteži pri računanju razdalje, le-te prehodno vključimo že v samo

transformacijo. Tako lahko namesto, da uporabimo obteženo razdaljo Cityblock (8.5), predhodno uporabimo proces beljenja in izračunamo razdaljo s pomočjo navadne razdalje Cityblock (8.1):

$$\begin{aligned} d_{wL1}(\mathbf{x},\mathbf{y}) &= \sum_{i=1}^{M} \frac{1}{\sqrt{\lambda_i}} |x_i - y_i| \\ &= \sum_{i=1}^{M} |\frac{1}{\sqrt{\lambda_i}} x_i - \frac{1}{\sqrt{\lambda_i}} y_i| \\ &= \sum_{i=1}^{M} \frac{1}{\sqrt{\lambda_i}} |u_i - v_i| \ , \end{aligned} \tag{8.15}$$

kjer je $u_i = \frac{1}{\sqrt{\lambda_i}} x_i$ in $v_i = \frac{1}{\sqrt{\lambda_i}} y_i$.

Opazimo lahko tudi, da moramo vrednosti nekaterih razdalj premakniti za določeno konstanto, če hočemo dobiti pozitivne vrednosti. Nekatere pa moramo ustrezno pretvoriti, da jih dobimo znotraj območja $[0, 1]$. Takšna normalizacija samo poveča računsko zahtevnost in za samo primerjavo ni pomembna. V primeru, če potrebujemo za prikaz normalizirane vrednosti, lahko na koncu normaliziramo samo nekaj najboljših rezultatov, da jih prikažemo uporabniku.

8.2 Povzetek poglavja

V poglavju smo predstavili različne razdalje, ki se lahko uporabljajo pri metodi najbližjega soseda. Izračunane razdalje nam dajo informacijo o tem, koliko sta vzorca, ki ju primerjamo, med seboj oddaljena. Na podlagi tega izvemo, koliko sta si vzorca podobna oziroma koliko sta si sliki, katerima vzorca pripadata, podobna.

Največkrat uporabljene razdalje so Evklidska, kosinusna in Mahalanobis razdalja. Mi smo predstavili še 11 drugih razdalj, ki jih tudi lahko uporabljamo za primerjavo. Vendar nekaterih med njimi ne moremo uporabljati v kombinaciji z vsemi metodami. Tako na primer obteženih razdalj ne moremo uporabljati z metodami, pri katerih iz samega izračuna ne dobimo uteži. Takšne metode so direktna LDA, ICA1, ICA2 in metoda Gaborjevih valčkov.

Različne razdalje določajo različne ocene ujemanja in posledično so tudi rezultati prepoznavanja obrazov različni. V poglavju 9 bomo na različnih metodah uporabili različne mere za merjenje razdalje in poskušali ugotoviti, katera kombinacija nam daje najboljše rezultate.

Poglavje 9

Preizkusi metod in hibridni sistemi

9.1 Uvod

V tem poglavju bomo predstavili preizkuse posameznih metod na dveh podatkovnih bazah obrazov, to je na podatkovni bazi FERET [77] in na podatkovni bazi AR [65]. Vse metode smo preizkusili z vsemi možnimi merami razdalj. Pri metodah, kjer dobimo poleg transformacijske matrike tudi uteži (PCA, LDA+PCA, PCA1 in PCA2), smo preizkus izvedli v kombinaciji z vsemi štirinajstimi razdaljami, v ostalih primerih pa s sedmimi. Rezultate podamo za najboljše ujemanje (rang 1) in za višje range v obliki krivulje CMS (angl. *Cumulative Match Score*). Na podlagi rezultatov lahko izberemo najboljše kombinacije metod in razdalj. Tako dobljene rezultate bomo na koncu primerjali in predlagali najboljše metode, ki smo jih kombinirali v hibridni sistem. Predlagani hibridni sistem, ki smo ga tudi preizkusili na obeh podatkovnih bazah bomo na koncu tudi podrobno opisali. Preden pa začnemo z opisi samih rezultatov preizkusov, definirajmo najprej njihovo metodologijo.

9.1.1 Metodologija preizkusov

Podatkovna baza

Za preizkus metod smo izbrali dve podatkovni bazi obrazov, ki smo ju podrobneje opisali v poglavjih 3.2.1 in 3.2.2. Podatkovna baza FERET že vsebuje štiri pripravljene testne množice: $fafb$, $fafc$, $dup1$ in $dup2$. Testne slike v postopku prepoznavanja primerjamo s slikami iz galerije G, ki vsebuje 1196 slik različnih oseb. To je tudi ena glavnih razlik s podatkovno bazo AR, pri kateri je galerija enaka učni množici slik. Testna množica $fafb$ vsebuje 1195 slik oseb, ki so bile zajete ob istem času kot slike iz galerije, s to razliko, da imajo osebe na obrazu drugačen izraz. Množica $fafc$ vsebuje 194 slik obrazov, zajetih pod različnimi svetlobnimi pogoji. Množica $dup1$ pa vsebuje 722 slik, ki so bile zajete v časovnem razponu od 1 minute

do 1331 dni po zajetju slik iz galerije. Testne slike množice *dup*2, ki so podmnožica množice *dup*1, so bile zajete najmanj 18 mesecev za slikami iz galerije.

Tudi slike iz podatkovne baze AR smo razdelili na učno in testno množico. Testna množica vsebuje enako število slik kot učna, s to razliko, da so bile vse slike zajete v drugem časovnem obdobju. Množici vsebujeta slike 119 različnih oseb in sicer v vsaki množici 7 različnih slik za eno osebo. Testno množico smo dodatno razdelili na dve podmnožici. Prva testna množica fb_{AR} vsebuje 476 (4*119) slik oseb z različnim izrazom na obrazu, množica fc_{AR} pa 357 (3*119) slik, posnetih pod različnimi svetlobnimi pogoji. Kot smo že omenili, se učna množica baze AR uporablja tudi kot galerija znanih slik pri postopku primerjave. Druga pomembna razlika v primerjavi s podatkovno bazo FERET je ta, da vsebuje podatkovna baza AR veliko več slik iste osebe. Za naš preizkus smo 7 slik izbrali za učenje in 4 oziroma 3 za primerjavo pri testiranju. Pri bazi FERET pa se za učenje uporabljata v povprečju 2 sliki in 1 slika na osebo za primerjavo pri testiranju.

Normalizacija

Vse slike moramo, preden vstopijo v sistem za prepoznavo, ustrezno normalizirati. Operacije nad slikami smo izvedli s pomočjo programskega paketa MATLAB. Za rotacijo smo uporabili funkcijo imrotate, za zmanjševanje slik pa imresize z bikubično interpolacijo. Slike smo ustrezno poravnali glede na pozicijo oči in jih obrezali. Pri rezanju smo odstranili odvečno ozadje, tako da smo na sliki ohranili samo obrazne značilke. Tako obrezanih slik tudi ni bilo potrebno dodatno maskirati. Slike smo pred rezanjem zmanjšali, da je bila razdalja med očmi 26 slikovnih elementov, očesi pa sta bili od zgornjega roba oddaljeni 24 slikovnih elementov. Tako zmanjšane slike so imele enotno velikost 60×50 slikovnih elementov. Slika 9.1 prikazuje nekaj tako normaliziranih slik obrazov iz podatkovne baze FERET in AR. Na koncu smo tako normalizirane slike še svetlobno normalizirali, tako da smo jim na podlagi histograma izenačili intenziteto.

9.1.2 Postopek učenja

Za postopek učenja smo pri podatkovni bazi FERET izbrali $N = 1007$ slik $c = 504$ različnih oseb. Učna množica se je uporabljala samo za učenje in je v postopku prepoznavanja nismo več uporabljali. Vsebovala je 396 slik iz galerije G in 99 slik iz testne množice *dup*1, ostale slike niso bile prisotne v nobeni od testnih množic. Pri podatkovni bazi AR smo za učni postopek izbrali $N = 833$ slik $c = 119$ različnih oseb. Torej je bilo število slik na osebo 7, v primeru baze FERET pa v povprečju 2 sliki na osebo.

S pomočjo tako določenih učnih množic smo lahko teoretično določili največji podprostor velikosti $(N - 1)$, torej za bazo FERET 1006 in za bazo AR 832. Po metodologiji FERET, ki določa, da izberemo 40% od vseh lastnih komponent, to znaša za bazo FERET 403 glavnih komponent in za bazo AR 333 glavnih komponent.

(a)

(b)

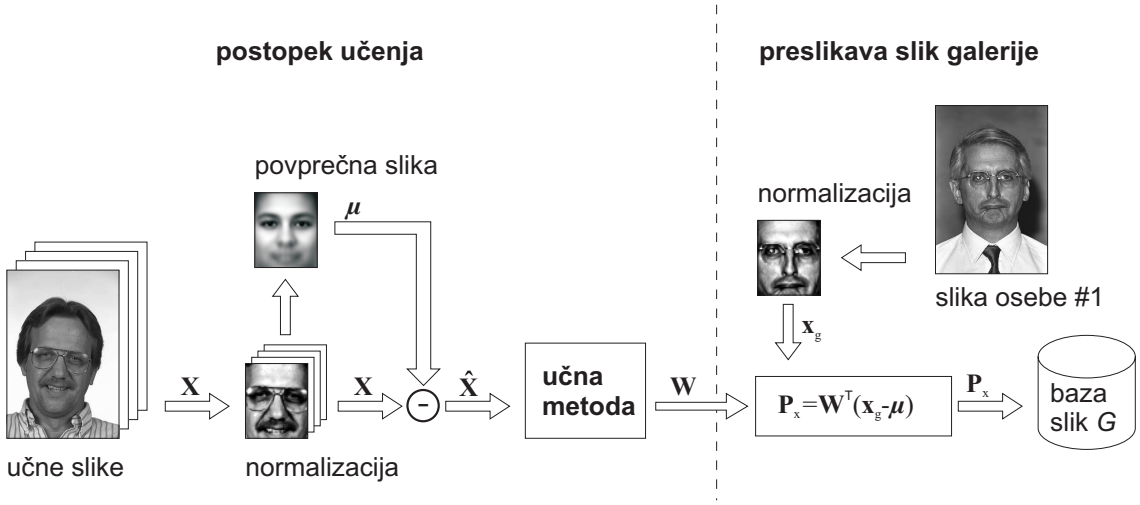
Slika 9.1: Primeri normaliziranih slik obrazov podatkovnih baz: (a) FERET in (b) AR.

Mi smo se raje odločili, da določimo optimalno število komponent s pomočjo metode PCA na podlagi preizkusa. Zaradi primerljivosti rezultatov z ostalimi metodami smo tako določeno število komponent uporabili tudi za vse ostale preizkuse.

Pri metodi LDA pa omejitev števila glavnih komponent določa število razredov c. V našem primeru število razredov predstavlja število različnih oseb. Zato je največje število komponent metode LDA pri bazi FERET $c - 1 = 503$, pri bazi AR pa 118. Paziti moramo tudi, če uporabljamo metodo PCA kot predkorak kakšne od metod (LDA, ICA). V tem primeru mora biti število komponent metode PCA enako ali večje od števila komponent teh metod. Pri metodi LDA moramo še posebej paziti, da število komponent metode PCA ni preveliko, ker se lahko zgodi, da postane matrika $\mathbf{S}_W$ singularna. Tako znaša za metodo LDA največje število komponent metode PCA za bazo FERET $N - c = 1007 - 504 = 503$, za bazo AR pa $N - c = 833 - 119 = 714$.

Tako izbrane učne slike potrebujemo za vse metode osnovane na videzu, da v po-

stopku učenja lahko zgradimo slikovni podprostor. V tako zgrajen podprostor nato preslikamo znane slike obrazov, s katerimi v postopku prepoznavanja primerjamo testne slike. Postopek učenja in preslikave prikazuje slika 9.2.



Slika 9.2: Postopek učenja metod osnovanih na videzu in preslikava znanih obrazov iz galerije G.

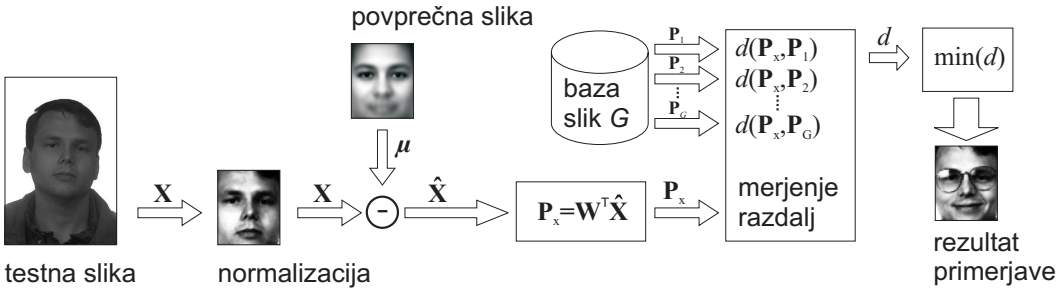
Pri postopku učenja vhodne slike iz učne množice najprej normaliziramo, nato od vsake slike odštejemo povprečno sliko, ki smo jo izračunali na podlagi vseh učnih slik. Tako pripravljene učne slike predstavljajo vhod v učno metodo. Rezultat učne metode je transformacijska matrika $\mathbf{W}$, ki določa obrazni slikovni podprostor. Pri postopku preslikave vse slike znanih oseb iz galerije preslikamo s pomočjo transformacijske matrike $\mathbf{W}$ v podprostor in s tem določimo njen vektor značilk $\mathbf{P}_X$. Tako izračunane vektorje značilk za posamezno sliko shranimo v galerijo G znanih obrazov.

9.1.3 Postopek prepoznave

V postopku prepoznave primerjamo vhodni vzorec neznane osebe (testna slika) z vsemi vzorci iz baze znanih oseb. Proces prepoznave za metode osnovane na videzu prikazuje slika 9.3.

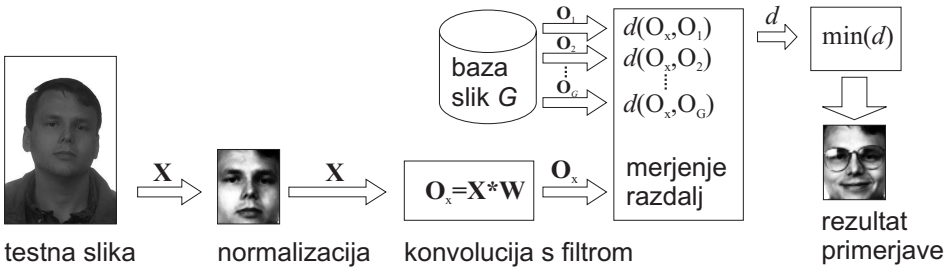
Testno sliko najprej normaliziramo in od nje odštejemo povprečni obraz, ki smo ga izračunali v postopku učenja. S pomočjo transformacijske matrike $\mathbf{W}$ nato izločimo iz slike obraza značilke, ki jih nato primerjamo z vsemi vektorji značilk iz galerije znanih obrazov G, ki smo jo zgradili v postopku preslikave. Razdaljo med vektorji določimo na podlagi metode za merjenje razdalj in najmanjša razdalja, ki določa najboljše ujemanje, predstavlja rezultat primerjave oziroma identifikacijo osebe iz vhodne testne slike.

Pri metodi Gaborjevih valčkov je postopek podoben. Po normalizaciji vhodne slike obraza dobimo kot rezultat njene konvolucije z Gaborjevimi filtri vektor značilk,



Slika 9.3: Postopek prepoznavanja testne slike neznanega obraza s pomočjo metod osnovanih na videzu.

ki določa vhodni vzorec testne slike. Ta vzorec nato primerjamo z vsemi vzorci znanih obrazov, ki smo jih predhodno shranili v galerijo G. Postopek prepoznave s pomočjo metode Gaborjevih valčkov prikazuje slika 9.4.



Slika 9.4: Postopek prepoznavanja testne slike neznanega obraza s pomočjo metode Gaborjevih valčkov.

9.2 Rezultati nad podatkovno bazo FERET

Prvi sklop preizkusov smo izvedli nad podatkovno bazo FERET in sicer nad posameznimi testnimi množicami: $fafb$, $fafc$, $dup1$ in $dup2$. Poleg rezultatov na posameznih testnih množicah bomo prikazali tudi skupen rezultat na celotni testni množici ($fafb + fafc + dup1$). Za posamezno metodo smo preizkusili uspešnost prepoznave pri rangu 1 (najboljše ujemanje) in pri višjih rangih (krivulja CMS). Pred samimi preizkusi pa smo s pomočjo metode PCA določili optimalno število glavnih komponent, katero smo nato uporabljali pri vseh nadaljnjih preizkusih.

9.2.1 Potek preizkusov posamezne metode

Nad vsemi metodami smo izvedli dva preizkusa. Pri prvem preizkusu smo preverili uspešnost identifikacije neznanega obraza na zaprti množici. Pri identifikaciji preverimo najboljše ujemanje neznanega obraza s slikami znanih obrazov, to je uspešnost pri rangu 1. Pri drugem preizkusu pa smo preizkusili uspešnost pri višjih rangih. Rezultati uspešnosti drugega preizkusa so še posebej pomembni pri aplikacijah, kjer hočemo nadaljnje preiskovanje samo omejiti in naš glavni cilj ni identifikacija osebe. Rezultat tega preizkusa je seznam oseb, med katerimi je tudi obraz iskane osebe. Oba preizkusa smo izvedli z različnimi merami razdalj in s tem poskušali ugotoviti, katera mera je najugodnejša za posamezno metodo. V nadaljevanju bomo podrobneje prikazali potek preizkusov za metodo PCA. V naslednjem poglavju pa bomo predstavili rezultate preizkusov še ostalih metod. Najprej pa opišimo postopek določitve optimalnega števila glavnih komponent.

Določitev optimalnega števila komponent

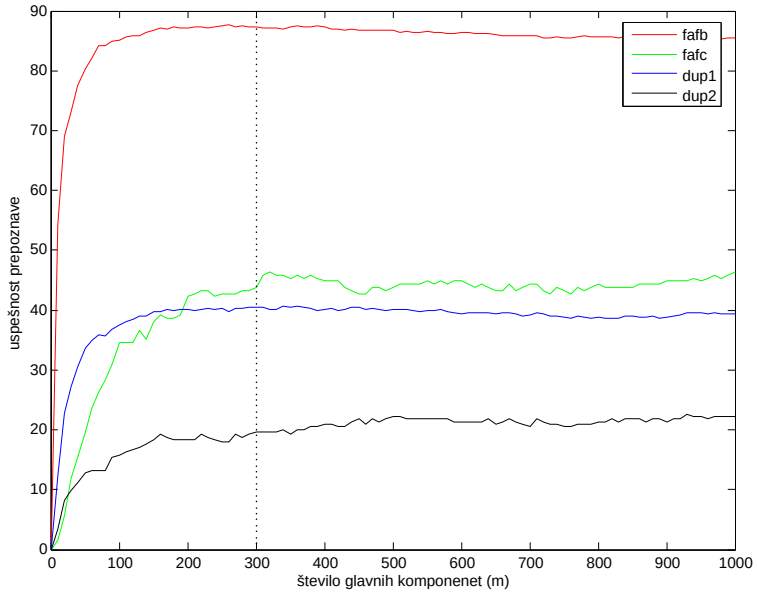
Določitev optimalnega števila glavnih komponent je zahtevna naloga. Predlaganih je bilo kar nekaj tehnik, ki smo jih podrobno opisali v poglavju 6 pri učnemu postopku metode PCA. Mi smo poskušali poiskati optimalno število s pomočjo preizkusa uspešnosti metode PCA za testne množice podatkovne baze FERET. Preizkus smo izvedli tako, da smo za posamezno število komponent izračunali uspešnost metode nad posamezno testno množico. Tako izračunane vrednosti uspešnosti prikazujejo krivulje na sliki 9.5. Uspešnost smo preizkusili pri vseh merah razdalj in ugotovili, da je karakteristika krivulje za vse mere podobna, zato tukaj prikazujemo karakteristično krivuljo uspešnosti samo za razdaljo $L1$.

Vidimo lahko, da uspešnost narašča do določenega števila komponent. V našem primeru bi lahko zaključili, da je to število za množici $fafb$ in $dup1$ 150. Uspešnost prepoznave za testno množico $fafc$ pa narašča približno do števila 300. Po priporočilu metodologije FERET bi morali število komponent določiti kot 40% vseh možnih komponent, kar v našem primeru znaša 1006*0.4=403 komponente. Iz preizkusa pa smo ugotovili, da je dovolj že 300 komponent, za bazi $fafb$ in $dup1$ pa še manj.

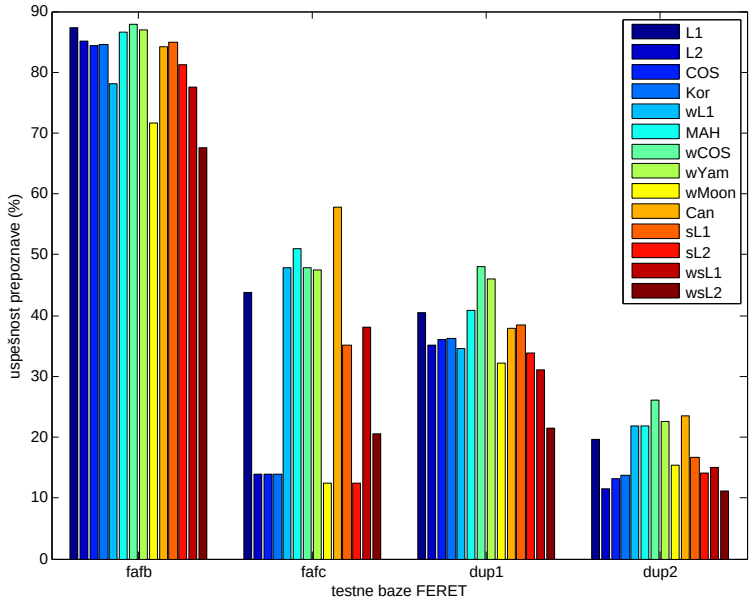
V vseh naslednjih preizkusih metode PCA smo tako uporabljali prvih 300 glavnih komponent. Da je primerjava med posameznimi metodami čimbolj enakovredna, smo tudi za druge metode izbrali enako število komponent. Na ta način zagotovimo tudi enako časovno kompleksnost prepoznavanja obrazov za različne metode.

Preizkus uspešnosti metode PCA za rang 1

V naslednjem preizkusu smo metodo PCA preizkusili nad vsemi testnimi množicami baze FERET in sicer z različnimi merami razdalj. V poglavju 8 smo opisali 14 različnih načinov, kako lahko izmerimo razdaljo med dvema vektorjema, ki smo jih tudi praktično preizkusili. Rezultate preizkusa za metodo PCA prikazuje slika 9.6.



Slika 9.5: Določitev števila komponent na podlagi metode PCA in baze FERET.



Slika 9.6: Rezultati uspešnosti prepoznave pri različnih merah razdalj za metodo PCA na bazi FERET.

Najboljše rezultate smo pričakovano dobili nad testno množico $fafb$, ker je ta množica najenostavnejša, saj vsebuje samo različne izraze na obrazu. Nekoliko slabše rezultate smo dobili nad množico $fafc$, ki vsebuje obraze pri različnih osvetlitvah. Najslabše rezultate pa smo dosegli na množici $dup1$ in še slabše na množici $dup2$. Ti dve množici sta se izkazali za najzahtevnejši za prepoznavo, saj vsebujeta slike

oseb, ki so bile zajete v drugem časovnem obdobju. V tem času so se osebe lahko zelo spremenile, pa tudi sami pogoji zajemanja so bili različni od prvotnih.

Na testni množici $fafb$ lahko vidimo, da se je večina mer za razdalje izkazala za uspešne, razen razdalj $wL1$, $wMoon$ in $wsL2$, kar je presenetljivo, ker smo s pomočjo njunih različici $L1$ in $wYam$ dobili zelo dobre rezultate. Najboljši rezultat smo dobili s pomočjo utežene kosinusne razdalje ($wCOS$). Presenetljivo pa je tudi, da se je tudi preprosta razdalja $L1$ izkazala za zelo uspešno, celo bolje kot nekatere druge razdalje, ki se običajno največkrat uporabljajo v kombinaciji z metodo PCA ($L2$, $wL2$, COS in MAH).

Na testni množici $fafc$ se je za daleč najboljšo izkazala kombinacija z razdaljo Canberra (Can). Zelo slabo pa so se izkazale razdalje $L2$, COS, Kor, $wMoon$ in $sL2$. Zaradi tega omenjene razdalje v kombinaciji z metodo PCA niso primerne, še posebej, če prepoznavamo obraze različnih osvetlitev.

Nad testnima množicama $dup1$ in $dup2$ pa sta bili najboljši razdalji $wCOS$ in $wYam$.

Vse rezultate metode PCA prikazuje tudi tabela 9.1. V tabelo smo dodali tudi rezultat uspešnosti za celotno testno bazo ($fafb$+$fafc$+$dup1$). Vidimo lahko, da se je nad celotno bazo za najboljšo razdaljo izkazala utežena kosinusna razdalja ($wCOS$).

	fafb	fafc	dup1	dup2	vse
L1	87,28	43,81	40,44	19,66	67,27
L2	85,19	13,92	35,04	11,54	61,49
COS	84,44	13,92	36,01	13,25	61,39
Kor	84,52	13,92	36,29	13,68	61,53
wL1	78,08	47,94	34,63	21,79	60,45
MAH	86,53	51,03	40,86	21,79	67,65
wCOS	**87,95**	47,94	**48,06**	**26,07**	**70,63**
wYam	86,95	47,42	45,98	22,65	69,30
wMoon	71,72	12,37	32,13	15,38	52,72
Can	84,18	**57,73**	37,81	23,50	65,89
sL1	84,85	35,05	38,50	16,67	64,42
sL2	81,17	12,37	33,93	14,10	58,69
wsL1	77,57	38,14	31,02	14,96	58,03
wsL2	67,53	20,62	21,47	11,11	47,47

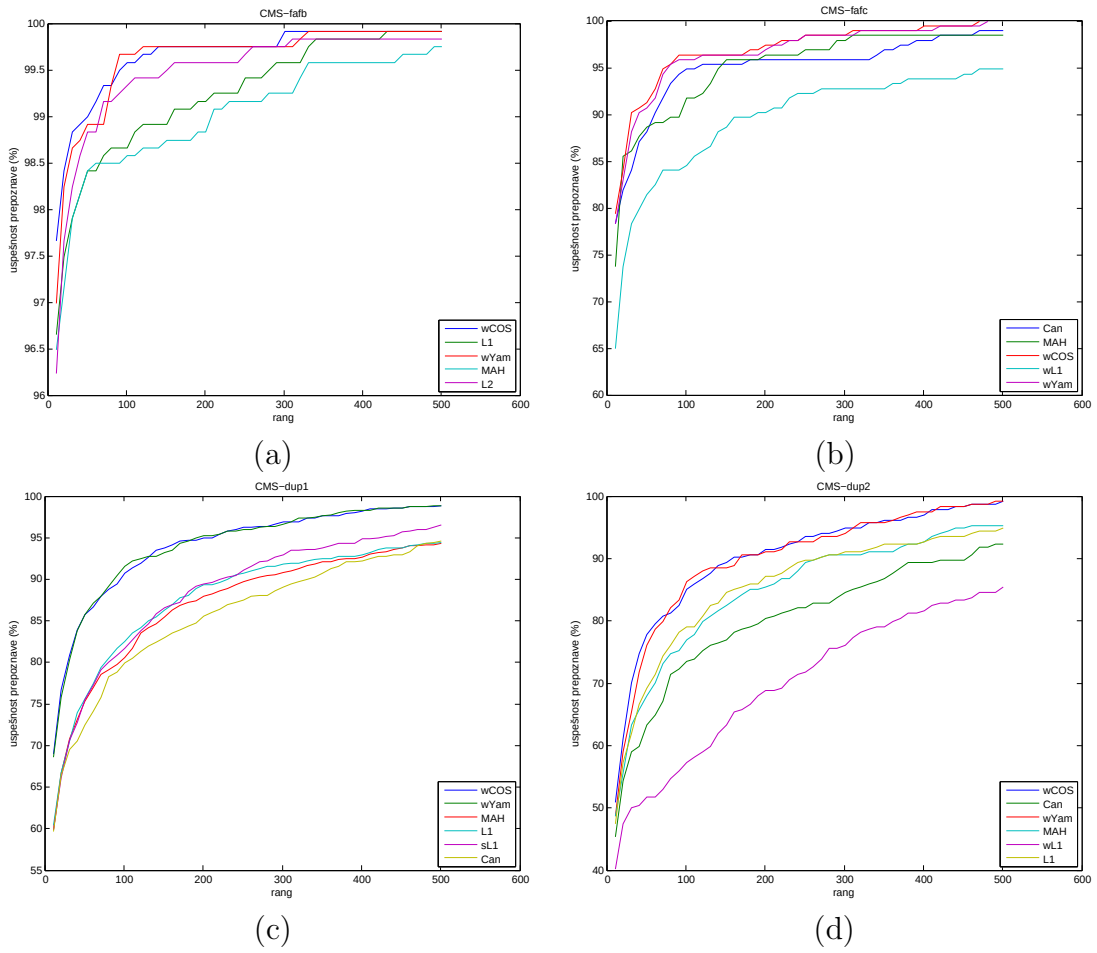
Tabela 9.1: Rezultati preizkusa metode PCA nad podatkovno bazo FERET za rang 1 pri različnih merah razdalje. Najboljši rezultati nad posamezno testno množico so poudarjeni.

Preizkus uspešnosti metode PCA pri različnih rangih

V poskusu, ki smo ga predstavili v prejšnjem poglavju smo preizkusili uspešnost metode PCA pri rangu 1, oziroma najboljše ujemanje. Nato pa smo preverili, kakšna je uspešnost metode pri različnih rangih večjih od ena. Uspešnost metode nam določa, ali je iskana oseba znotraj določenega območja oziroma ranga. Takšna uspešnost je pomembna pri sistemih, kjer ne potrebujemo točnega odgovora o identifikaciji iskane osebe ampak je dovolj, da sistem samo omeji preiskovalno območje in nam

kot rezultat vrne seznam, znotraj katerega je tudi iskana oseba. Kot bomo videli v nadaljevanju, je rezultat zelo pomemben tudi pri gradnji zaporednega hibridnega sistema.

Na sliki 9.7 prikazujemo grafe uspešnosti nad vsemi štirimi tesnimi množicami baze FERET in sicer za prvih nekaj razdalj, ki so se v prejšnjem preizkusu izkazale za najuspešnejše pri rangu 1.



Slika 9.7: Rezultati uspešnosti metode PCA pri višjih rangih na testnih množicah baze FERET: (a) $fafb$, (b) $fafc$, (c) $dup1$ in (d) $dup2$.

Iz slik lahko vidimo, da ne velja vedno, da je najuspešnejša razdalja pri rangu 1 tudi najuspešnejša razdalja za celotni rang. Na sliki 9.7 smo rang omejili samo na določeno območje z razlogom, da lažje vidimo, katere mere razdalj so najboljše. V tabeli 9.2 prikazujemo v zadnjih treh vrsticah prve tri najboljše mere razdalj za range večje od ena. V oklepaju pa je zapisano mesto, ki ga je razdalja zasedla pri rangu ena. Za primerjavo smo v prve tri vrstice dopisali tudi najboljše tri mere razdalj in njihovo uspešnost pri rangu 1.

Nad testno množico $fafb$ je še vedno najuspešnejša razdalja $wCOS$, razdalja $L1$

	$fafb$		$fafc$		$dup1$		$dup2$		vse	
rang 1, 1.	wCOS	87.95	Can	57.73	wCOS	48.06	wCOS	26.07	wCOS	70.63
rang 1, 2.	L1	87.28	MAH	51.03	wYam	45.98	Can	23.50	wYam	69.30
rang 1, 3.	wYam	86.95	wCOS	47.94	MAH	40.86	wYam	22.65	MAH	67.65
CMS, 1.	wCOS (1.)		wCOS (3.)		wCOS (1.)		wCOS (1.)		wCOS (1.)	
CMS, 2.	wYam (3.)		wYam (5.)		wYam (2.)		wYam (3.)		wYam (1.)	
CMS, 3.	L2 (5.)		Can (1.)		L1 (4.)		L1 (6.)		L1 (4.)	

Tabela 9.2: Rezultati uspešnosti metode PCA na bazi FERET za rang ena in pri višjih rangih.

pa ni več druga najboljša, ampak sta od nje bolšji razdalja $wYam$ in razdalja $L2$, ki je bila, na primer pri rangu ena šele na petem mestu (slika 9.7 (a)).

Pri množici $fafc$ se je za zelo uspešno izkazala razdalja $wYam$, ki jo po uspešnosti lahko primerjamo z najboljšo razdaljo $wCOS$. Zanimivo je to, da razdalja $wYam$ ni bila preveč uspešna pri rangu ena (5. mesto), enako tudi razdalja $wCOS$, ki je bila pri rangu ena šele na tretjem mestu. Razdalja, ki se je izkazala za najboljšo pri rangu ena, to je razdalja Canberra (Can), pa se je tukaj izkazala kot tretja najboljša (slika 9.7 (b)).

Za množico $dup1$ velja, da sta tudi pri višjih rangih najboljši razdalji $wCOS$ in $wYam$, ki sta se za najboljši izkazali že pri rangu ena. Edini razliki z rangom ena sta razdalji $L1$, ki se je izkazala za boljšo od MAH in $sL1$, ki se je od ranga 250 naprej izkazala za boljšo kot razdalja $L1$ (slika 9.7 (c)).

Pri množici $dup2$ je najboljša razdalja ostala $wCOS$, enako uspešna pa je tudi razdalja $wYam$. Presenetljivo pa se je za zelo uspešno razdaljo izkazala $L1$, ki je bila pri rangu ena na šestem mestu. Razdalja Can, ki je bila pri rangu ena na drugem mestu, pa pri višjih ni bila uspešna (slika 9.7 (d)).

Po preizkusu uspešnosti pri različnih rangih nad celotno testno bazo se je za najboljšo razdaljo izkazala razdalja $wCOS$, ki ji sledi razdalja $wYam$. Presenetljivo sta se tudi tukaj, v primerjavi z rangom ena, izkazali za zelo uspešni razdalji $L1$ in $sL1$.

V nadaljevanju bomo predstavili in interpretirali rezultate še ostalih metod, ki smo jih preizkusili z različnimi razdaljami za rang ena in pri višjih rangih na vseh testnih množicah baze FERET.

9.2.2 Rezultati preizkusov vseh metod na bazi FERET

V tem poglavju bomo predstavili rezultate uspešnosti posameznih metod z različnimi merami razdalj za rang 1 in za višje range na različnih testnih množicah baze FERET ter nad celotno testno bazo. V prvih treh vrsticah bomo navedli mere razdalj, pri katerih je bila metoda najuspešnejša za rang ena. Poleg imena razdalje bomo navedli še rezultat uspešnosti. V naslednjih treh vrsticah pa bomo predstavili tri najuspešnejše razdalje pri višjih rangih. Poleg imena razdalje bomo v oklepaju zapisali tudi mesto, ki ga je zasedla razdalja pri rangu ena.

Metoda PCA

Za začetek ponovno predstavimo rezultate uspešnosti za metodo PCA (tabela 9.3). Pri rangu ena je najuspešnejša metoda $wCOS$, razen na testni množici $fafc$, kjer sta boljši razdalji Can in MAH. Pri višjih rangih pa lahko zaključimo, da je metoda PCA najuspešnejša v kombinaciji z razdaljo $wCOS$.

	$fafb$		$fafc$		$dup1$		$dup2$		vse	
rang 1, 1.	wCOS	87,95	Can	57,73	wCOS	48,06	wCOS	26,07	wCOS	70,63
rang 1, 2.	L1	87,28	MAH	51,03	wYam	45,98	Can	23,50	wYam	69,30
rang 1, 3.	wYam	86,95	wCOS	47,94	MAH	40,86	wYam	22,65	MAH	67,65
CMS, 1.	wCOS (1.)		wCOS (3.)		wCOS (1.)		wCOS (1.)		wCOS (1.)	
CMS, 2.	wYam (3.)		wYam (5.)		wYam (2.)		wYam (3.)		wYam (1.)	
CMS, 3.	L2 (5.)		Can (1.)		L1 (4.)		L1 (6.)		L1 (4.)	

Tabela 9.3: Rezultati uspešnosti metode PCA na bazi FERET za rang ena in pri višjih rangih.

Metoda PCA brez prvih treh komponent

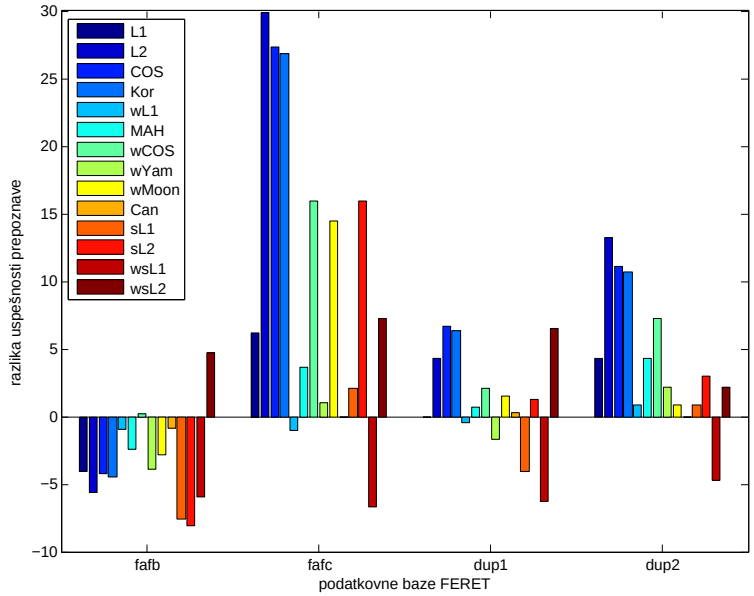
Ugotovljeno je bilo, da lahko nosijo prve glavne komponente metode PCA tudi informacije, ki so za samo ločevanje med posameznimi osebami moteče. Še posebej to velja za slike oseb, ki so bile zajete pod različnimi svetlobnimi pogoji (testna množica $fafc$). Zaradi tega smo tudi mi preizkusili metodo PCA tako, da smo ji odstranili prve tri glavne komponente. Zaradi same primerljivosti z ostalimi metodami smo še vedno ohranili 300 komponent, s tem da smo manjkajoče tri komponente nadomestili z naslednjimi tremi 301, 302 in 303. Preizkus smo izvedli z različnimi merami razdalj tako za rang ena kot tudi pri višjih rangih. Rezultate preizkusa prikazuje tabela 9.4. Vidimo lahko, da se je razdalja $wCOS$, podobno kot pri osnovni metodi

	$fafb$		$fafc$		$dup1$		$dup2$		vse	
rang 1, 1.	wCOS	88,12	wCOS	63,92	wCOS	50,14	wCOS	33,33	wCOS	72,90
rang 1, 2.	MAH	84,10	Can	57,73	wYam	44,32	MAH	26,07	MAH	66,84
rang 1, 3.	Can	83,35	MAH	54,64	Kor	42,66	wYam	24,79	wYam	66,60
CMS, 1.	wCOS (1.)		wCOS (1.)		wCOS (1.)		wCOS (1.)		wCOS (1.)	
CMS, 2.	wYam (5.)		wYam (5.)		wYam (2.)		wYam (3.)		wYam (3.)	
CMS, 3.	COS (6.)		L2 (7.)		Kor (3.)		Kor (5.)		COS (6.)	

Tabela 9.4: Rezultati uspešnosti metode PCA brez prvih treh komponent na bazi FERET pri rangu ena in pri višjih rangih.

PCA, izkazala za najuspešnejšo. Opazimo lahko tudi, da se je uspešnost na splošno izboljšala tudi pri razdaljah, ki so se pri osnovni metodi PCA izkazale za najslabše. Podobno kot pri osnovni PCA se je pri višjih rangih presenetljivo dobro izkazala tudi razdalja $wYam$, ki pri rangu ena ni izstopala.

Ker je preizkušena metoda v bistvu metoda PCA, ki smo ji odvzeli prve tri glavne komponente, jo je smiselno primerjati z osnovno metodo PCA. Razliko rezultatov prepoznave prikazuje slika 9.8, kjer pozitivne vrednosti prikazujejo stopnjo izboljšanja predlagane metode PCA v primerjavi z osnovno metodo PCA. Največje iz-



Slika 9.8: Razlika uspešnosti prepoznave za metodo PCA in spremenjeno PCA brez prvih treh komponent.

boljšanje smo, po pričakovanjih, dosegli na množici $fafc$, ki vsebuje obraze različnih osvetlitev. Izboljšanje pri razdalji $L2$ je skoraj 30%. Tudi kombinacija z razdaljo $wCOS$ je še dodatno izboljšala uspešnost v primerjavi z osnovno metodo in se je tako izkazala za najboljšo razdaljo. Tudi na drugih testnih množicah je metoda z nekaterimi razdaljami uspešnejša, tako da lahko zaključimo, da je metoda PCA, ki smo ji odstranili prve tri glavne komponente, v splošnem uspešnejša od osnovne metode PCA. V našem primeru predstavljajo torej prve tri komponente nezaželeno informacijo v smislu razlikovanja med različnimi osebami.

Metodi LDA+PCA in direktna LDA

Tukaj bomo predstavili rezultate dveh različic metode LDA. Pri prvi metodi smo predhodno za zmanjšanje visoko dimenzijskega slikovnega prostora uporabili metodo PCA [113], pri drugi pa smo izvedli metodo LDA neposredno na slikah, kot to predlaga avtor Juwei Lu et al. [42]. Največje število glavnih komponent za metodo LDA je $c - 1$, kjer je c število razredov oziroma v našem primeru število različnih oseb ($c = 504$). Mi smo za obe različici izbrali, enako kot pri metodi PCA, prvih 300 glavnih komponent. Pri metodi LDA+PCA smo tudi sam slikovni prostor s pomočjo metode PCA predhodno zmanjšali na dimenzijo 300.

Tabela 9.5 prikazuje rezultate za metodo LDA+PCA ali tako imenovano metodo Fisherjevih obrazov. Pri rangu ena in pri višjih rangih sta se za najboljši razdalji izkazali razdalja Kor in COS. Za uspešni sta se izkazali tudi razdalji Can in $L2$. Razdalji $wYam$ in $wMoon$ pa sta se izkazali kot neprimerni razdalji v kombinaciji

	$fafb$		$fafc$		$dup1$		$dup2$		vse	
rang 1, 1.	Kor	90,46	Kor	68,56	Kor	59,00	Kor	41,88	Kor	77,69
rang 1, 2.	COS	90,38	COS	68,56	COS	59,00	COS	41,88	COS	77,64
rang 1, 3.	L2	84,35	Can	59,28	L2	50,14	L2	39,32	L2	69,73
CMS, 1.	Kor (1.)		Kor (1.)		Kor (1.)		Kor (1.)		Kor (1.)	
CMS, 2.	COS (2.)		COS (2.)		COS (2.)		COS (2.)		COS (2.)	
CMS, 3.	Can (4.)		Can (3.)		Can (5.)		L2 (3.)		Can (4.)	

Tabela 9.5: Rezultati uspešnosti metode LDA+PCA na bazi FERET za rang ena in pri višjih rangih.

s to metodo.

Tabela 9.6 prikazuje rezultate preizkusa direktne metode LDA. Na testnih množicah

	$fafb$		$fafc$		$dup1$		$dup2$		vse	
rank 1, 1.	sL1	81,26	sL1	46,39	COS	57,48	L1	56,84	sL1	69,59
rank 1, 2.	sL2	80,25	sL2	43,30	Kor	57,34	COS	56,41	sL2	67,65
rank 1, 3.	COS	77,99	Kor	40,21	sL1	56,51	Kor	55,98	COS	67,46
CMS, 1.	sL1 (1.)		sL1 (1.)		sL1 (3.)		Can (6.)		sL1 (1.)	
CMS, 2.	sL2 (2.)		sL2 (2.)		Can (7.)		sL1 (5.)		sL2 (2.)	
CMS, 3.	COS (3.)		Kor (3.)		COS (1.)		L1 (1.)		COS (3.)	

Tabela 9.6: Rezultati uspešnosti direktne metode LDA na bazi FERET za rang ena in pri višjih rangih.

$fafb$ in $fafc$ je bila najboljša razdalja $sL1$, na množicah $dup1$ in $dup2$ pa COS oziroma $L1$. Pri višjih rangih je pri vseh testnih množicah prevladovala razdalja $sL1$. Na množicah $dup1$ in $dup2$ se je za uspešno izkazala tudi razdalja Can, čeprav je bila pri rangu ena med slabšimi. Če primerjamo med seboj obe različici metode LDA lahko vidimo, da deluje metoda LDA v kombinaciji z metodo PCA bolje, kot njena direktna različica.

Metodi ICA1 in PCA1

Metoda ICA1 določa prvo arhitekturo metode ICA. Pri tej metodi nimamo lastnih vrednosti, zato smo tudi preizkus izvedli samo z neuteženimi razdaljami ($L1$, $L2$, COS, Kor, Can, $sL1$ in $sL2$). Število komponent metode ICA izberemo enako kot pri drugih metodah, torej 300. Ker vsebuje metoda ICA1 kot predkorak metodo PCA, je pomembna tudi izbira števila komponent metode PCA. Na preizkusih se je izkazalo, da je metoda najuspešnejša pri izbiri najmanjšega števila komponent metode PCA, kar je v našem primeru prvih 300 glavnih komponent.

Tabela 9.7 prikazuje rezultate metode ICA1. Metoda ICA1 se je na splošno izkazala zelo slabo. Dobre rezultate smo dobili samo na testni množici $fafb$ in sicer z razdaljo $L1$. Tudi na drugih testnih množicah in pri višjih rangih se je za najboljšo izkazala razdalja $L1$. Presenetljivo se je v primerjavi z rangom ena pri višjih rangih na množici $fafc$ izkazala za uspešno tudi razdalja $L2$.

Različico metode PCA, ki smo jo uporabili kot predkorak metodi ICA1, lahko preizkusimo kot samostojno metodo. S tem preizkusom lahko ugotovimo, koliko

	$fafb$		$fafc$		$dup1$		$dup2$		vse	
rang 1, 1.	L1	89,12	L1	15,46	L1	37,53	L1	14,10	L1	64,71
rang 1, 2.	sL1	89,12	sL1	14,43	sL1	36,57	sL1	13,68	sL1	64,28
rang 1, 3.	Can	88,12	sL2	13,92	sL2	35,60	Can	13,68	Can	62,81
CMS, 1.	L1 (1.)		L1 (1.)		L1 (1.)		L1 (1.)		L1 (1.)	
CMS, 2.	sL1 (2.)		sL1 (2.)		sL1 (2.)		Can (3.)		sL1 (2.)	
CMS, 3.	Can (3.)		L2 (7.)		Can (7.)		sL1 (2.)		Can (3.)	

Tabela 9.7: Rezultati uspešnosti metode ICA1 na bazi FERET za rang ena in pri višjih rangih.

doprinese k uspešnosti sam postopek ločevanja neodvisnih komponent metode ICA. Rezultate metode PCA1 prikazuje tabela 9.8. Iz rezultatov metode PCA1 lahko

	$fafb$		$fafc$		$dup1$		$dup2$		vse	
rang 1, 1.	wYam	87,36	Can	57,22	wYam	47,37	Can	23,93	wYam	70,06
rang 1, 2.	L1	87,36	MAH	50,00	wCOS	46,95	wYam	23,50	wCOS	69,78
rang 1, 3.	wCOS	87,11	wL1	50,00	sL1	41,00	wsL2	23,08	MAH	67,22
CMS, 1.	wYam (1.)		wYam (6.)		wYam (1.)		wYam (2.)		wYam (1.)	
CMS, 2.	L2 (7.)		wCOS (7.)		wCOS (2.)		wCOS (4.)		wCOS (2.)	
CMS, 3.	sL2 (8.)		Can (1.)		sL1 (3.)		wsL2 (3.)		sL1 (6.)	

Tabela 9.8: Rezultati uspešnosti metode PCA1 na bazi FERET za rang ena in pri višjih rangih.

vidimo, da so le-ti, razen pri množici $fafb$, boljši od metode ICA1. Pri metodi PCA1 so se izkazale za boljše utežene različice razdalj. Pri rangu ena se je za najboljšo izkazala razdalja $wYam$, nad testno množico $fafc$ pa razdalja Can. Nad to množico je bila tudi uspešnost metode v primerjavi z ICA1 veliko boljša. Pri višjih rangih je nad vsemi testnimi množicami prevladovala razdalja $wYam$ in to ne glede na to, kako uspešna je bila pri rangu ena. Za zelo uspešno se je izkazala tudi razdalja $wCOS$, čeprav pri rangu ena ni bila med najboljšimi.

Metodi ICA2 in PCA2

Metoda ICA2 določa drugo arhitekturo metode ICA. Podobno kot pri metodi ICA1 smo tudi tukaj izbrali enako število glavnih komponent ICA ter enako število komponent za metodo PCA2, ki je predkorak metode ICA2. Metodo ICA2 smo preizkusili s sedmimi neuteženimi merami razdalj, metodo PCA2 pa v kombinaciji z vsemi razdaljami. Tabela 9.9 prikazuje rezultate za metodo ICA2. Za najboljši razdalji sta se izkazali razdalji COS in Kor, s katerima smo dobili zelo podobne rezultate. Ti dve razdalji tudi zelo izstopata v primerjavi z drugimi razdaljami. Razdalji sta se izkazali za najuspešnejši tudi pri višjih rangih.

V tabeli 9.10 predstavljamo rezultate za metodo PCA2, ki je predkorak same metode ICA2. Na podlagi primerjave teh rezultatov z rezultati metode ICA2 smo lahko ugotovili, koliko doprinese k uspešnosti prepoznave sam postopek ICA druge arhitekture. Rezultati za metodo PCA2 so pokazali nad vsemi testnimi množicami enotne rezultate glede izbire za mero razdalje. Tako za rang ena, kot tudi pri višjih

	$fafb$		$fafc$		$dup1$		$dup2$		vse	
rang 1, 1.	COS	88,37	Kor	70,10	Kor	46,26	Kor	30,34	COS	72,29
rang 1, 2.	Kor	88,28	COS	70,10	COS	46,26	COS	30,34	Kor	72,24
rang 1, 3.	L2	78,49	Can	56,19	L2	34,76	L2	23,08	L2	60,87
CMS, 1.	COS (1.)		Kor (1.)		Kor (1.)		Kor (1.)		COS (1.)	
CMS, 2.	Kor (2.)		COS (2.)		COS (2.)		COS (2.)		Kor (2.)	
CMS, 3.	L2 (3.)		Can (3.)		Can (4.)		Can (4.)		L2 (3.)	

Tabela 9.9: Rezultati uspešnosti metode ICA2 na bazi FERET za rang ena in pri višjih rangih.

	$fafb$		$fafc$		$dup1$		$dup2$		vse	
rang 1, 1.	Kor	88,45	Kor	70,10	Kor	46,40	Kor	30,77	Kor	72,38
rang 1, 2.	COS	88,37	COS	70,10	COS	46,26	COS	30,34	COS	72,29
rang 1, 3.	Can	84,18	wCOS	61,86	wMoon	41,00	wCOS	28,21	wMoon	67,17
CMS, 1.	Kor (1.)		Kor (1.)		Kor (1.)		Kor (1.)		Kor (1.)	
CMS, 2.	COS (2.)		COS (2.)		COS (2.)		COS (2.)		COS (2.)	
CMS, 3.	wMoon (4.)		wMoon (4.)		wMoon (3.)		wMoon (4.)		wMoon (3.)	

Tabela 9.10: Rezultati uspešnosti metode PCA2 na bazi FERET za rang ena in pri višjih rangih.

rangih se je za najboljšo izkazala razdalja Kor, za drugo najboljšo pa razdalja COS. V primerjavi z metodo ICA2 so rezultati zelo podobni, kar dokazuje, da sam postopek ICA tudi pri drugi arhitekturi ne izboljša uspešnosti prepoznave.

Metoda Gaborjevih valčkov

Rezultate uspešnosti prepoznave za metodo Gaborjevih valčkov prikazuje tabela 9.11. Pri rangu ena se je za najboljšo razdaljo izkazala Can, razen na testni bazi $fafc$,

	$fafb$		$fafc$		$dup1$		$dup2$		vse	
rang 1, 1.	Can	92,72	L1	88,14	Can	51,66	Can	38,89	Can	77,97
rang 1, 2.	L1	89,37	Can	85,05	sL1	47,78	sL1	37,18	L1	74,66
rang 1, 3.	sL1	88,37	sL2	83,51	L1	46,68	L1	31,20	sL1	73,76
CMS, 1.	Can (1.)		L1 (1.)		Can (1.)		sL1 (2.)		Can (1.)	
CMS, 2.	L1 (2.)		Can (2.)		sL1 (2.)		Can (1.)		L1 (2.)	
CMS, 3.	sL1 (3.)		sL2 (3.)		L1 (3.)		Kor (4.)		sL1 (3.)	

Tabela 9.11: Rezultati uspešnosti metode Gaborjevih valčkov na bazi FERET za rang ena in pri višjih rangih.

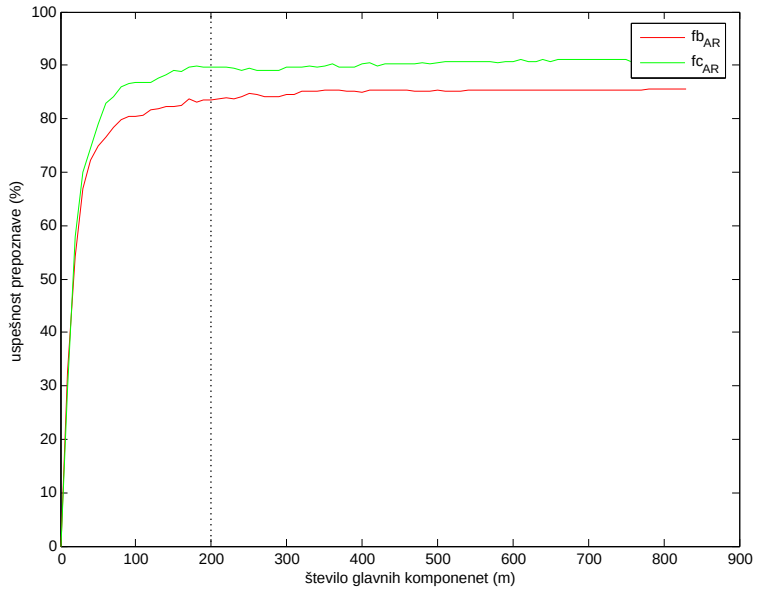
kjer je bila uspešnejša razdalja $L1$. Podobno se je tudi pri višjih rangih na množicah $fafb$ in $dup2$ izkazala za najboljšo razdalja Can, na množicah $fafc$ in $dup2$ pa $L1$ oziroma njena različica $sL1$.

9.3 Rezultati nad podatkovno bazo AR

Drugi sklop preizkusov smo izvedli nad podatkovno bazo AR in sicer nad testnima bazama fb_{AR} in fc_{AR}. Nad podatkovno bazo AR smo izvedli iste preizkuse, kot smo jih izvršili nad podatkovno bazo FERET. Najprej smo s pomočjo metode PCA določili optimalno število glavnih komponent za vse preizkuse na podatkovni bazi AR.

9.3.1 Določitev optimalnega števila komponent

Podobno kot na podatkovni bazi FERET, tudi tukaj preizkusimo uspešnost metode PCA z različnim številom glavnih komponent. Graf uspešnosti v odvisnosti od števila uporabljenih komponent prikazuje slika 9.9.



Slika 9.9: Določitev števila komponent za metodo PCA na bazi AR.

Vidimo lahko, da uspešnost narašča do števila 200. Od tu naprej je vrednost uspešnosti konstantna. Na podlagi tega preizkusa lahko zaključimo, da je optimalno število glavnih komponent 200. Po predlogu metodologije FERET, ki izbere 40% vseh glavnih komponent, bi v našem primeru imeli 833*0.4=333 glavnih komponent. Iz preizkusa pa vidimo, da dosežemo enako uspešnost že s prvimi 200-timi glavnimi komponentami.

V nadaljevanju bomo opisali preizkuse kjer smo na podatkovni bazi AR preizkusili posamezne metode v kombinaciji z različnimi merami razdalj za rang 1 in pri višjih rangih. Pri vsaki metodi smo upoštevali samo prvih 200 komponent.

9.3.2 Rezultati preizkusov vseh metod na bazi AR

Tabela 9.12 prikazuje rezultate metode PCA nad testnima množicama fb in fc podatkovne baze AR. Zadnji stolpec prikazuje skupni rezultat nad obema množicama. Prve tri vrstice prikazujejo najboljše ujemanje pri rangu ena, zadnje tri vrstice pa imena treh razdalj, ki so se izkazale za najuspešnejše pri višjih rangih. Velja namreč, da najuspešnejša razdalja pri rangu ena ni nujno tudi najuspešnejša pri višjih rangih. Za najboljše razdalje pri rangu ena so se v kombinaciji z metodo PCA izkazale

	fb_{AR}		fc_{AR}		*skupaj*	
rang 1, 1.	wCOS	88,45	wCOS	91,32	wCOS	89,68
rang 1, 2.	MAH	86,76	wYam	90,76	wYam	88,36
rang 1, 3.	wYam	86,55	MAH	90,20	MAH	88,24
CMS, 1.	wCOS (1.)		wCOS (1.)		wCOS (1.)	
CMS, 2.	wYam (3.)		wYam (2.)		wYam (2.)	
CMS, 3.	MAH (2.)		MAH (3.)≈ Can (6.)		MAH (3.)	

Tabela 9.12: Rezultati uspešnosti metode PCA na bazi AR za rang ena in pri višjih rangih.

razdalje $wCOS$, $wYam$ in MAH. Vidimo tudi, da so rezultati nad testno množico fc, ki vsebuje obraze različnih osvetlitev boljši. Pri višjih rangih se je izkazala za najboljšo razdalja $wCOS$. Pri množici fb je razdalja $wYam$ delovala boljše kot razdalja MAH, čeprav je pri rangu ena veljalo obratno. Pri množici fc pa se pri višjih rangih vrstni red razdalj ni spremenil. Za zelo uspešno se je izkazala tudi razdalja Can, ki se lahko po uspešnosti primerja z razdaljo MAH.

Naslednji preizkus smo naredili s pomočjo metode PCA, s tem da smo ji odstranili prve tri glavne komponente. Rezultate tako spremenjene metode PCA prikazuje tabela 9.13. Pri rangu ena so se ponovno za najboljše izkazale razdalje $wCOS$, $wYam$

	fb_{AR}		fc_{AR}		*skupaj*	
rang 1, 1.	wCOS	88,24	wCOS	93,00	wCOS	90,28
rang 1, 2.	wYam	86,76	wYam	92,16	wYam	89,08
rang 1, 3.	MAH	85,92	MAH	89,08	MAH	87,27
CMS, 1.	wYam (2.)		wCOS (1.)		wYam (2.)	
CMS, 2.	wCOS (1.)		wYam (2.)		wCOS (1.)	
CMS, 3.	Kor (6.)		Kor (6.)		Kor (6.)	

Tabela 9.13: Rezultati uspešnosti metode PCA brez prvih treh komponent na bazi AR za rang ena in pri višjih rangih.

in MAH. Pri višjih rangih sta se za najboljši izkazali razdalji $wCOS$ in $wYam$. Presenetljivo dobro se je izkazala tudi razdalja Kor. V primerjavi z osnovno metodo PCA se je metoda PCA brez prvih treh komponent nad množico fb izkazala za ekvivalentno, na množici fc pa se je uspešnost izboljšala. Množica fc vsebuje namreč obraze različnih osvetlitev, zaradi česar deluje metoda PCA brez prvih treh komponent boljše, saj vsebujejo v tem primeru prve tri komponente predvsem informacije osvetlitve, ki so za samo razlikovanje nepomembne.

Pri metodi LDA je največje število komponent $c-1$, kjer c določa število različnih oseb. V podatkovni bazi AR je število različnih oseb 119, zato je največje število komponent, ki jih lahko izberemo, enako 118. Pri metodi LDA+PCA najprej slikovni prostor zmanjšamo s pomočjo metode PCA na dimenzijo velikosti 200 nato pa s pomočjo metode LDA določimo 118 komponent. Rezultate metode LDA+PCA prikazuje tabela 9.14. Za najboljšo razdaljo se je tako pri rangu ena kot pri višjih

	fb_{AR}		fc_{AR}		*skupaj*	
rang 1, 1.	COS	89,71	Kor	92,16	COS	90,76
rang 1, 2.	Kor	89,29	COS	92,16	Kor	90,52
rang 1, 3.	wCOS	87,82	wCOS	91,88	wCOS	89,56
CMS, 1.	COS (1.)		Kor (1.)		COS (1.)	
CMS, 2.	Kor (2.)		COS (2.)		Kor (2.)	
CMS, 3.	wCOS (3.)		wCOS(3.)		wCOS (3.)	

Tabela 9.14: Rezultati metode LDA+PCA na bazi AR za rang ena in pri višjih rangih.

rangih izkazala razdalja COS. Zelo podobne rezultate smo dosegli tudi z razdaljo Kor. Tretja najuspešnejša je bila razdalja $wCOS$.

Tabela 9.15 prikazuje rezultate direktne metode LDA. Pri direktni metodi je

	fb_{AR}		fc_{AR}		*skupaj*	
rang 1, 1.	sL1	86,76	sL1	91,60	sL1	88,84
rang 1, 2.	Kor	85,50	Kor	90,48	Kor	87,64
rang 1, 3.	COS	85,50	sL2	90,20	COS	87,52
CMS, 1.	sL1 (1.)		sL1 (1.)		sL1 (1.)	
CMS, 2.	sL2 (4.)		sL2 (3.)		sL2 (5.)	
CMS, 3.	COS (3.)		Kor (2.)		COS (3.)	

Tabela 9.15: Rezultati uspešnosti direktne metode LDA na bazi AR pri rangu ena in pri višjih rangih.

prevladovala razdalja $sL1$ tako pri rangu ena kot pri višjih rangih. Ravno tako sta pri vseh rangih dosegali podobne rezultate razdalji Kor in COS. Pri višjih rangih pa je bila od teh dveh razdalj uspešnejša razdalja $sL2$.

Tabela 9.16 prikazuje rezultate prve arhitekture metode ICA na testni bazi AR. Pri rangu ena sta bili na testni množici fb najboljši razdalji Kor in COS, na množici

	fb_{AR}		fc_{AR}		*skupaj*	
rang 1, 1.	Kor	77,10	L1	73,67	L1	74,91
rang 1, 2.	COS	77,10	sL1	68,91	Can	67,23
rang 1, 3.	L2	76,68	Can	68,35	sL1	66,63
CMS, 1.	COS (2.), > 30 L1		L1 (1.),> 35 sL1		L1 (1.), > 45 sL1	
CMS, 2.	Kor (1.), > 30 sL1		Can (3.), > 35 L1		sL1 (3.), > 60 Can	
CMS, 3.	L2 (3.), > 30 sL2		sL1 (2.), > 35 Can		Can (2.), > 60 L1	

Tabela 9.16: Rezultati uspešnosti metode ICA1 na bazi AR za rang ena in pri višjih rangih.

fc pa razdalja $L1$. Pri višjih rangih pa se uspešnost nekaterih razdalj spremeni: za

množico fb po rangu večjem od 30, za množico fc pa po rangu 35. Pri nižjih rangih prevladujejo iste razdalje kot pri rangu ena, pri višjih pa se izkažejo za boljše razdalje $L1$, $sL1$ in $sL2$.

Pred samim postopkom ICA zmanjšamo slikovni prostor s pomočjo spremenjene metode PCA, ki smo jo poimenovali PCA1. Tabela 9.17 prikazuje rezultate metode PCA1, torej brez postopka ločevanja izvorov. Metoda PCA1 se je izkazala za boljšo

	fb_{AR}		fc_{AR}		*skupaj*	
rang 1, 1.	wCOS	87,39	wCOS	92,44	wCOS	89,56
rang 1, 2.	MAH	87,18	wYam	92,16	wYam	88,84
rang 1, 3.	wYam	86,34	MAH	89,64	MAH	88,24
CMS, 1.	wCOS (1.)		wCOS (1.)		wCOS (1.)	
CMS, 2.	wYam (3.)		wYam (2.)		wYam (2.)	
CMS, 3.	MAH (2.), > 45 sL2, wMoon		MAH (3.)		MAH (3.)	

Tabela 9.17: Rezultati uspešnosti metode PCA1 na bazi AR za rang ena in pri višjih rangih.

od metode ICA1. Najbolje je delovala v kombinaciji z razdaljo $wCOS$ tako za rang ena kot pri višjih rangih. Druga najboljša razdalja je bila $wYam$, tretja pa MAH in sicer pri množici fb samo do ranga 45. Po tem rangu sta bili boljši razdalji $sL2$ in $wMoon$.

Tabeli 9.18 in 9.19 prikazujeta rezultate metode ICA2 in metode PCA2, ki je predkorak same metode ICA2. V kombinaciji z metodo ICA2 sta bili najboljši

	fb_{AR}		fc_{AR}		*skupaj*	
rang 1, 1.	Kor	88,66	Kor	91,88	Kor	90,04
rang 1, 2.	COS	88,45	COS	91,88	COS	89,92
rang 1, 3.	L2	82,98	L2	89,08	L2	85,59
CMS, 1.	Kor (1.)		Kor (1.)		Kor (1)	
CMS, 2.	COS (2.)		COS (2.)		COS (2.)	
CMS, 3.	L2 (3.), > 30 L1		L1 (4.)		L1 (4.)	

Tabela 9.18: Rezultati uspešnosti metode ICA2 na bazi AR pri rangu ena in pri višjih rangih.

razdalji Kor in COS tako pri rangu ena kot pri višjih rangih. Pri rangu ena je bila tretja najuspešnejša razdalja $L2$ pri višjih rangih pa razdalja $L1$. Pri metodi PCA2

	fb_{AR}		fc_{AR}		*skupaj*	
rang 1, 1.	wMoon	89,08	wMoon	93,28	wMoon	90,88
rang 1, 2.	Kor	88,45	Kor	91,88	Kor	89,92
rang 1, 3.	COS	88,45	COS	91,88	COS	89,92
CMS, 1.	wMoon (1.)		wMoon (1.)		wMoon (1.)	
CMS, 2.	Kor (2.)		Kor (2.)		Kor (2.)	
CMS, 3.	COS (3.)		COS (3.)		COS (3.)	

Tabela 9.19: Rezultati uspešnosti metode PCA2 na bazi AR za rang ena in pri višjih rangih.

se je najbolje izkazala razdalja $wMoon$ in dosegla na obeh testnih množicah boljše

rezultate kot metoda ICA2. Iz tega lahko sklepamo, da lahko dosežemo že s samo metodo PCA2 brez postopka ICA primerljive ali celo boljše rezultate.

Tabela 9.20 prikazuje rezultate metode Gaborjevih valčkov na testnih množicah baze AR. Na testni množici fb je bila pri rangu ena najuspešnejša razdalja $sL1$ in

	fb_{AR}		fc_{AR}		*skupaj*	
rang 1, 1.	sL1	92,86	Can	95,80	Can	93,88
rang 1, 2.	L1	92,86	L1	94,96	L1	93,76
rang 1, 3.	Can	92,44	sL1	94,40	sL1	93,52
CMS, 1.	sL1 (1.)		Can (1.)		L1 (2.)	
CMS, 2.	L1 (2.)		L1 (2.)		Can (1.)	
CMS, 3.	sL2 (4.)		sL1 (3.)		sL1 (3.)	

Tabela 9.20: Rezultati uspešnosti metode Gaborjevih valčkov na bazi AR pri rangu ena in pri višjih rangih.

njena različica $L1$. Na množici fc pa je najboljše rezultate dosegla metoda Gaborjevih valčkov v kombinaciji z razdaljo Can, pa tudi v kombinaciji z razdaljama $L1$ in $sL1$ je delovala zelo dobro. Pri višjih rangih ni bilo sprememb, razen tega, da se je na skupni testni bazi od razdalje Can bolje odrezala razdalja $L1$, ki je bila pri rangu ena slabša.

9.4 Primerjava rezultatov različnih metod in baz

V tem poglavju bomo primerjali med seboj najboljše kombinacije metod in mere razdalj. Rezultate bomo primerjali pri najboljšem ujemanju, torej za rang 1 in pri višjih rangih. Rezultate pri rangu 1 bomo predstavili numerično in v obliki grafikona, za višje range pa s pomočjo krivulj CMS (angl. *Comulative Match Score*). S pomočjo krivulje CMS prikažemo uspešnost metode za različne vrednosti ranga. Metoda je pri določenem rangu uspešna če je rezultat, v našem primeru iskana oseba, znotraj tega ranga.

9.4.1 Rezultati za rang 1

V prejšnjih poglavjih smo predstavili preizkuse pri katerih smo preverili vse možne mere razdalj v kombinaciji s posamezno metodo. Na podlagi rezultatov smo ugotovili, katera kombinacija je na posamezni testni množici najboljša. V tabeli 9.21 predstavljamo rezultate tako dobljenih najboljših kombinacij metod in razdalj za posamezno testno množico podatkovnih baz FERET in AR. Na podlagi rezultatov smo poskušali ugotoviti, katera razdalja in katera metoda sta primernejša za posamezno nalogo, ki jo določa posamezna testna množica.

	$fafb$	$fafc$	$dup1$	$dup2$	fb_{AR}	fc_{AR}
PCA	87,95 wCOS	57,73 Can	48,06 wCOS	26,07 wCOS	88,45 wCOS	91,32 wCOS
PCAwo3	88,12 wCOS	63,92 wCOS	50,14 wCOS	33,33 wCOS	88,24 wCOS	93,00 wCOS
LDA+PCA	90,46 Kor	68,56 Kor	**59,00** **Kor**	41,81 Kor	89,71 COS	92,16 Kor
LDA	81,26 sL1	46,39 sL1	57,48 COS	**56,84** **L1**	86,76 sL1	91,60 sL1
ICA1	89,12 sL1	15,46 L1	37,53 L1	14,10 L1	77,10 Kor	73,67 L1
PCA1	87,36 wYam	57,22 Can	47,37 wYam	23,93 Can	87,39 wCOS	92,44 wCOS
ICA2	88,37 COS	70,10 Kor	46,26 Kor	30,34 Kor	88,66 Kor	91,88 Kor
PCA2	88,45 Kor	70,10 Kor	46,40 Kor	30,77 Kor	89,08 wMoon	93,28 wMoon
Gabor	**92,72** **Can**	**88,14** **L1**	51,66 Can	38,89 Can	**92,86** **sL1**	**95,80** **Can**

Tabela 9.21: Rezultati najboljših kombinacij metod in razdalj za posamezno testno množico podatkovnih baz FERET in AR. Najboljši rezultati nad posamezno testno množico so poudarjeni.

Iz tabele 9.21 lahko vidimo, da je utežena kosinusna razdalja $wCOS$ najboljša

v kombinaciji z metodama PCA in PCAwo3 (metoda PCA brez prvih treh komponent). Metodi sta se v kombinaciji z razdaljo $wCOS$ izkazali za najboljši nad vsemi testnimi množicami, razen pri množici $fafc$, kjer je bila uspešnejša razdalja Canberra. Ampak tudi v tem primeru je bila razdalja $wCOS$ tretja najboljša z uspešnostjo 47,94%.

Korelacijska razdalja (Kor) se je najbolje odrezala pri metodah LDA+PCA, ICA2 in PCA2. Razdalja Kor je bila pri teh metodah uspešna pri večini testnih množic. Razen pri testni množici fb_{AR}, kjer je v kombinaciji z metodo LDA+PCA bolje delovala razdalja COS in pri $fafb$, kjer je bila z metodo ICA2 znova boljša razdalja COS. Vendar je bila v obeh primerih druga najboljša razdalja Kor, tako da lahko zaključimo, da delujeta metodi LDA+PCA in ICA2 najbolje v kombinaciji s Korelacijsko razdaljo. Presenetljivo je na podatkovni bazi AR metoda PCA2 delovala bolje z razdaljo $wMoon$, kar pa ne velja za podatkovno bazo FERET. Razdalja $wMoon$ se je pri podatkovni bazi FERET v kombinaciji z metodo PCA2 izkazala za dobro samo pri višjih rangih. Ker pa se je kot druga najboljša razdalja nad testnima množicama baze AR izkazala zopet razdalja Kor, lahko tudi za metodo PCA2 zaključimo, da deluje ne glede na testno množico zelo dobro v kombinaciji s Korelacijsko razdaljo (Kor).

Naslednja zanimiva ugotovitev je pri razdalji $L1$ in njeni spremenjeni različici $sL1$. Ta mera za razdaljo je zelo enostavna in se v kombinaciji z metodami za prepoznavanje obrazov ne uporablja pogosto, kljub temu pa se je v našem primeru izkazala za zelo uspešno v kombinaciji z metodama LDA in ICA1. Razdalja $L1$ se je izkazala za najboljšo z metodo ICA1, razen na testni množici fb baze AR, kjer je bila boljša razdalja Kor. Na tej testni množici se je razdalja $L1$ izkazala za boljšo šele pri višjih rangih. Kljub temu lahko sklenemo, da je najboljša kombinacija metoda ICA1 z razdaljo $L1$. Spremenjena različica razdalje $L1$ ($sL1$) pa se je najbolje odrezala z direktno metodo LDA. Nad testnima množicama $dup1$ in $dup2$ razdalja $sL1$ ni bila najboljša vendar je vseeno delovala zelo dobro.

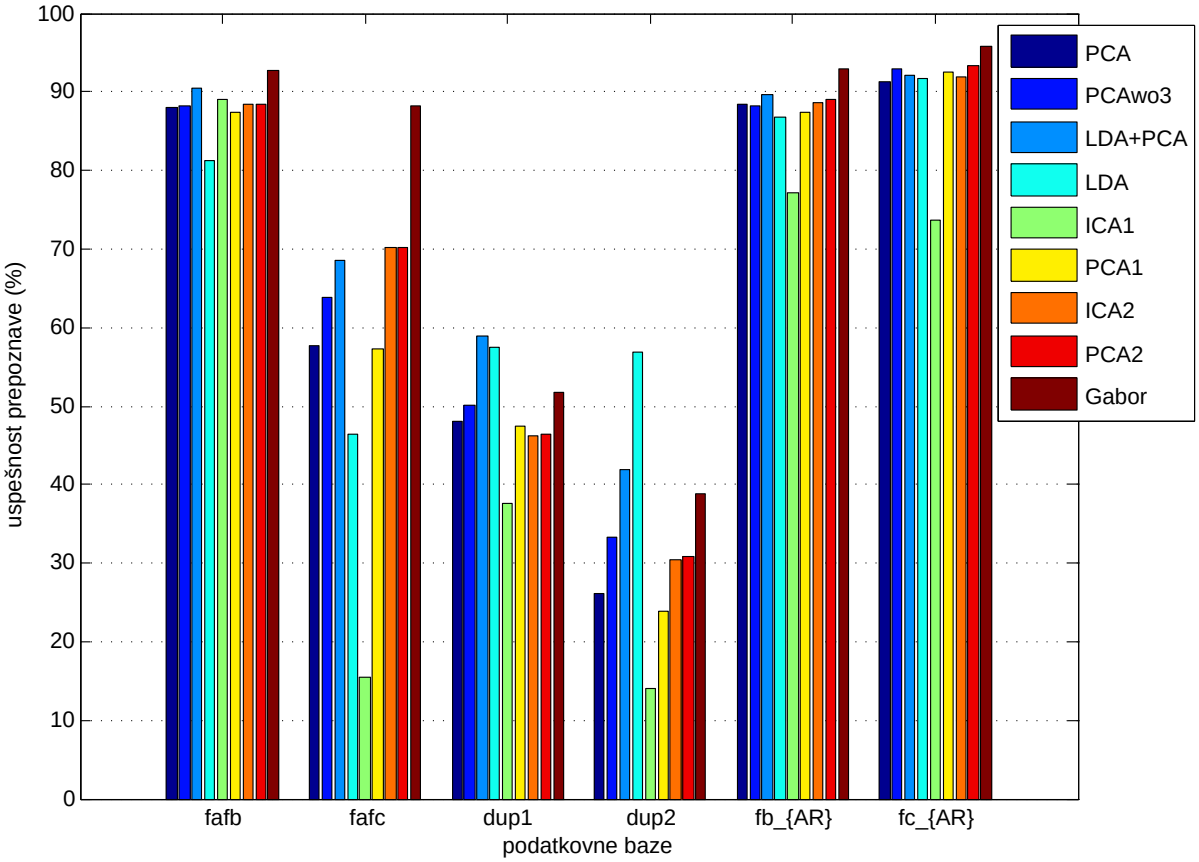
Za metodo PCA1 ne moremo zaključiti, s katero razdaljo deluje najbolje. Pri podatkovni bazi FERET se je v nekaterih primerih izkazala za boljšo razdalja $wYam$, v drugih pa razdalja Can. Čeprav deluje metoda na podatkovni bazi AR zelo dobro tudi z razdaljo $wYam$, pa se je za najboljšo kombinacijo izkazala razdalja $wCOS$.

Za metodo Gaborjevih valčkov se je izkazalo, da deluje najbolje v kombinaciji z razdaljo Canberra, v nekaterih primerih pa jo za malenkost prehiti razdalja $L1$ ali pa razdalja $sL1$.

Rezultate najboljših kombinacij različnih metod za posamezno testno množico prikazujemo še grafično na sliki 9.10.

Iz slike 9.10 lažje vidimo, katera metoda je najuspešnejša nad posamezno testno množico. Na testno množico lahko gledamo tudi kot na točno določeno nalogo, kjer vsebuje vsaka testna množica skrbno izbrane slike obrazov. Na ta način lahko ugotovimo, katera metoda je primernejša za določeno nalogo.

Prva testna množica $fafb$ vsebuje slike obrazov različnih izrazov. Ta testna množica v glavnem ni nobeni izmed metod povzročala prevelikih težav. Vidimo



Slika 9.10: Rezultati najboljših kombinacij različnih metod za posamezno testno bazo FERET in AR.

lahko, da je samo metoda LDA delovala slabše, vse ostale metode pa so dosegle uspešnost skoraj 90%. Za najboljši sta se izkazali metoda Gaborjevih valčkov in metoda LDA v kombinaciji z metodo PCA (LDA+PCA).

Naslednja testna množica $fafc$ je vsebovala slike obrazov različnih osvetlitev. Tukaj je bil padec uspešnosti v primerjavi z množico $fafb$ precejšen. Nekatere metode, kot so LDA, ICA1 in PCA1, so popolnoma odpovedale. Spet se je za najboljšo izkazala metoda Gaborjevih valčkov, ki je tudi zaradi tega, ker deluje na lokalnih značilkah, najmanj občutljiva na različne svetlobne pogoje. Izkazali sta se tudi metodi PCA2 in ICA2, pa tudi metoda LDA+PCA ni za njima veliko zaostajala.

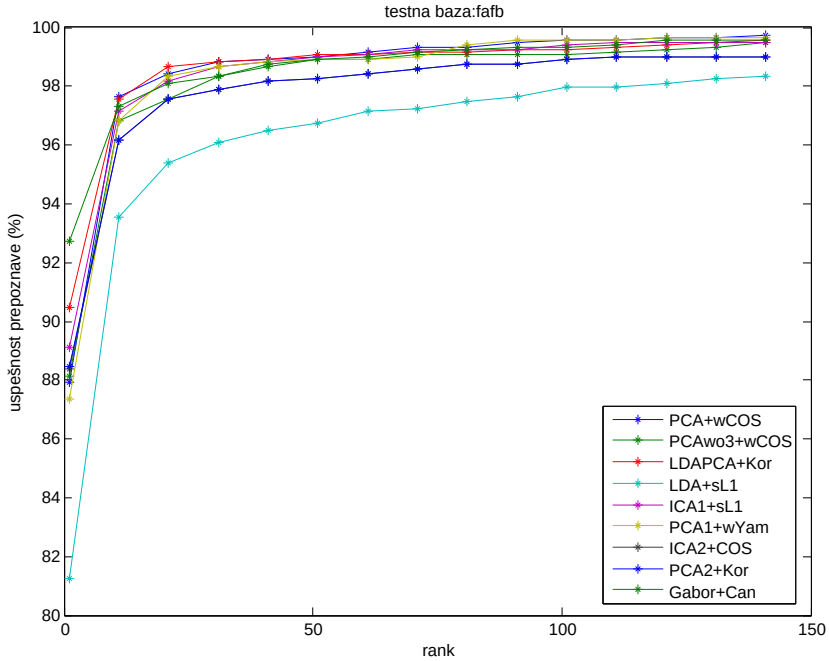
Naslednje dve testni množici $dup1$ in $dup2$ sta se izkazali za najzahtevnejši. Množici namreč vsebujeta slike obrazov, ki so bile zajete po daljšem časovnem obdobja. Med vsemi metodami najbolj izstopa metoda LDA. Na testni množici $dup1$, kjer je časovno obdobje krajše, pa se je izkazala za malenkost boljšo metoda LDA+PCA.

Testni množici podatkovne baze AR fb in fc sta se izkazali za lažji. Prva množica

fb je vsebovala slike obrazov z različnimi izrazi, zajetimi pod enakimi svetlobnimi pogoji. Testna množica fc pa je vsebovala slike obrazov zajete pod različnimi svetlobnimi pogoji. Slike obeh testnih množic so bile posnete po dveh tednih od zajetja učnih slik. Zanimivo je to, da smo dosegli boljše rezultate nad drugo testno množico, čeprav je vsebovala obraze različnih osvetlitev. V tem primeru se je namreč testna množica različnih izrazov izkazala za težjo v smislu prepoznavanja. Najboljša metoda za obe testni množici je bila ponovno metoda Gaborjevih valčkov. Pri množici fb ji po uspešnosti sledijo metode LDA+PCA, PCA2 in ICA2, pri množici fc pa PCA2, PCAwo3 in PCA1. Na obeh množicah se je najslabše odrezala metoda ICA1.

9.4.2 Rezultati za višje range

Do sedaj smo si pogledali najuspešnejše kombinacije metod in razdalj pri najboljšem ujemanju torej pri rangu ena. Na slikah 9.11– 9.16 prikazujemo rezultate najboljših kombinacij metod nad posameznimi testnimi množicami za višje range. Rezultate prikažemo s pomočjo tako imenovanih krivulj CMS (angl. *Cumulative Match Score*).

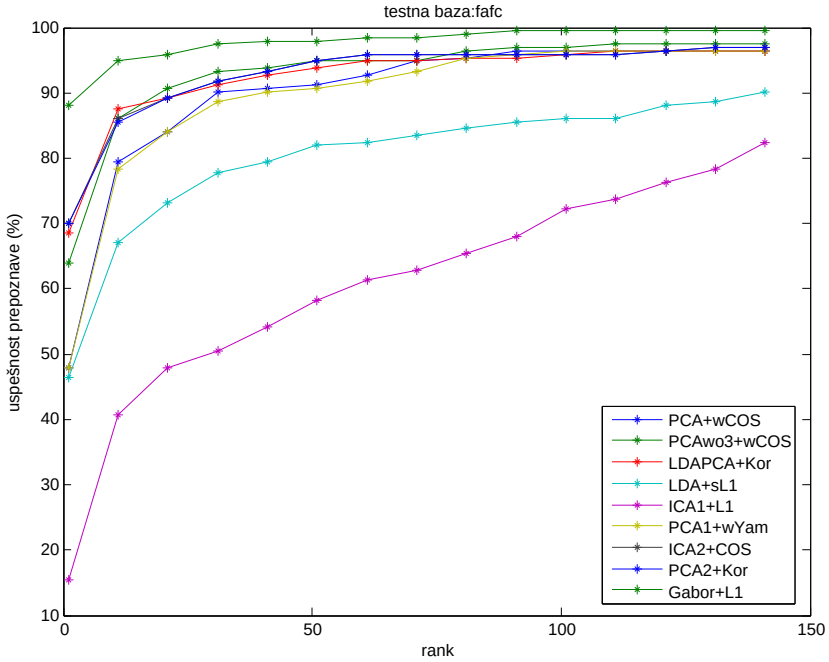


Slika 9.11: Krivulja CMS za najboljše kombinacije metod in razdalj testne množice $fafb$.

Na testni množici $fafb$ se je pri višjih rangih za najboljšo kombinacijo izkazala metoda PCA+$wCOS$. Takoj za njo so se po uspešnosti uvrstile metode LDA+PCA, ICA1, PCA1 in Gaborjeva metoda. Uspešnost teh metod je med seboj zelo podobna, zaradi česar je težko ločiti, katera med njimi je najuspešnejša. Če natančneje pogledamo pa lahko vidimo, da je pri nižjem rangu od 10 do 30 najboljša kombinacija

metode LDA+PCA in razdalje Kor, od ranga 80 do 150 pa prevladuje kombinacija metode PCA1 in razdalje $wYam$. Ostale tri kombinacije PCA2+Kor, ICA2+COS in LDA+sL1 so se nad bazo $fafb$ izkazale za slabše.

Slika 9.12 prikazuje krivulje najboljših kombinacij metod in razdalj za testno množico $fafc$. Na testni množici $fafc$ se je pri višjih rangih najbolje odrezala

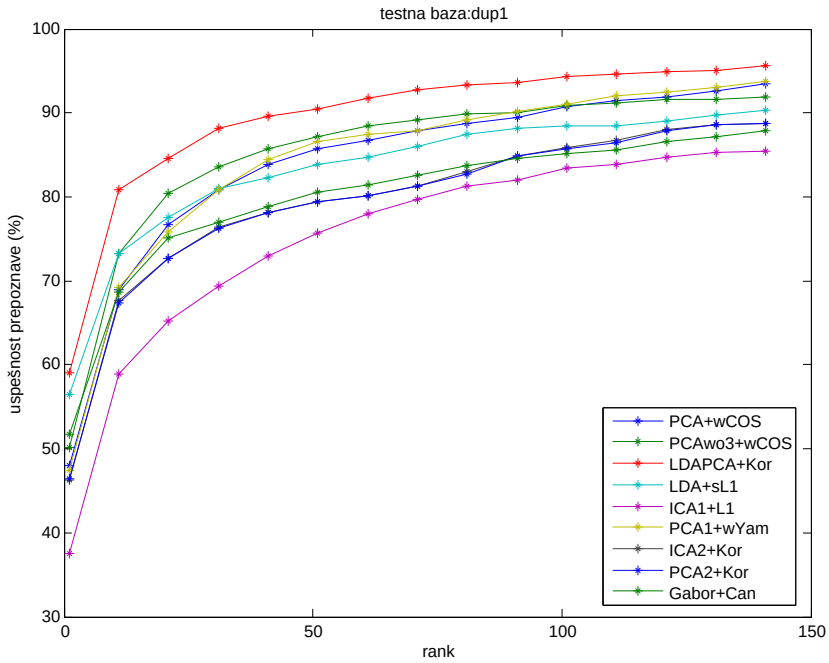


Slika 9.12: Krivulja CMS za najboljše kombinacije metod in razdalj testne množice $fafc$.

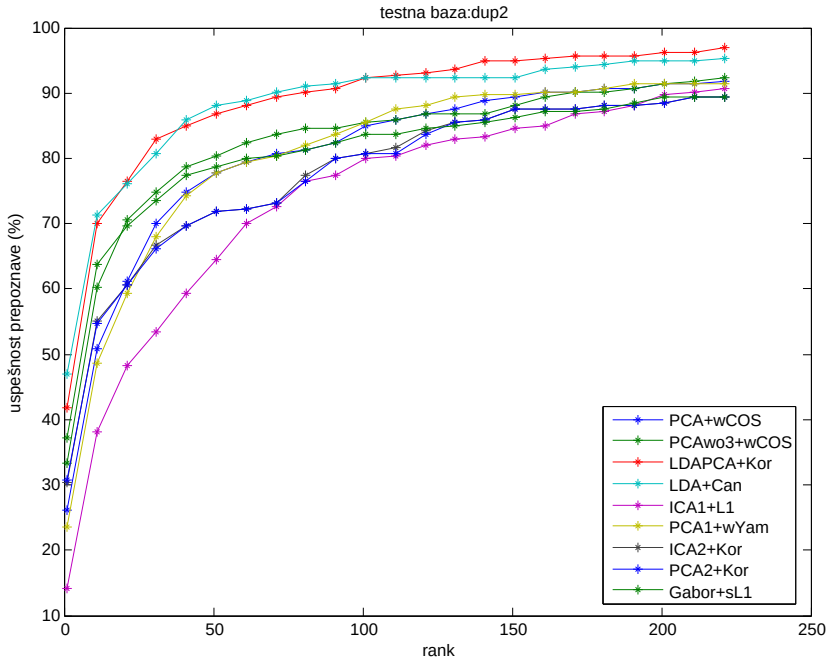
metoda Gaborjevih valčkov v kombinaciji z razdaljo $L1$, naslednja najboljša kombinacija je bila PCA brez prvih treh komponent+$wCOS$. Nato so si tri metode po uspešnosti zelo podobne, to so LDAPCA+Kor, ICA2+COS in PCA2+Kor. Slabše sta se uvrstili kombinaciji PCA+$wCOS$ in PCA1+$wYam$ ter najslabše LDA+$sL1$ in ICA1+$L1$.

Na sliki 9.13 prikazujemo krivulje CMS za testno množico $dup1$. Pri testni množici $dup1$ prevladuje kombinacija LDAPCA+Kor. Do ranga 100 je druga najboljša metoda PCA brez prvih treh komponent (PCAwo3+$wCOS$), po rangu 100 pa jo po uspešnosti prehitita metodi PCA1 v kombinaciji z razdaljo $wYam$ in metoda PCA+$wCOS$. Do ranga 100 pa sta ti dve metodi na tretjem in četrtem mestu po uspešnosti. Sledi kombinacija LDA+$sL1$, ki je pri nizkih rangih (do približno 30) celo na tretjem mestu po uspešnosti. Gaborjeva metoda je do ranga 100 boljša od metod ICA2 in PCA2, po tem rangu pa sta od nje ti dve metodi boljši. Najslabše pa se je izkazala kombinacija ICA1+$L1$.

Na sliki 9.14 prikazujemo krivulje CMS za testno množico $dup2$. Tudi pri testni množici $dup2$ lahko ločimo dve območji, na katerih je uspešnost metod različna.



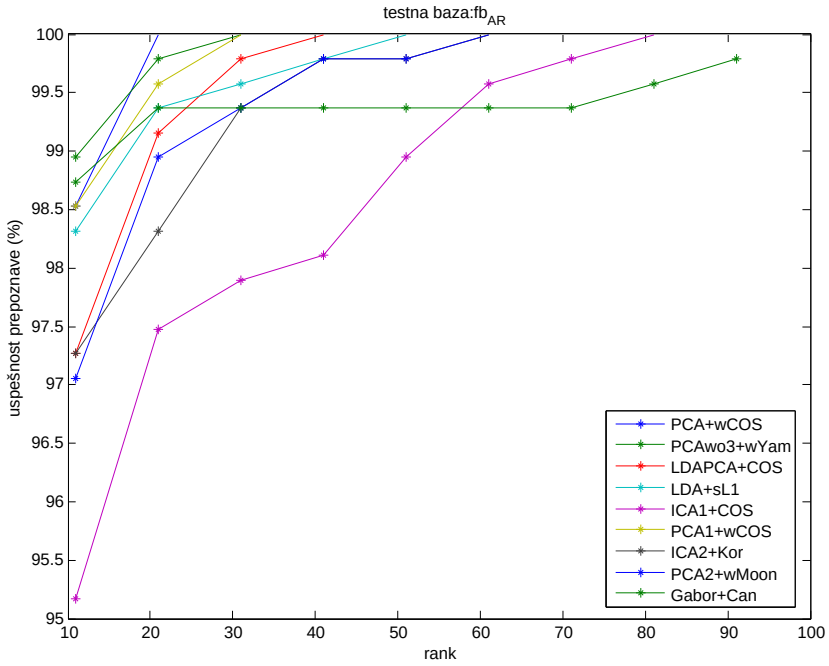
Slika 9.13: Krivulja CMS za najboljše kombinacije metod in razdalj testne množice $dup1$.



Slika 9.14: Krivulja CMS za najboljše kombinacije metod in razdalj testne množice $dup2$.

Ločnica je približno rang 100. Do ranga 100 je najboljša metoda LDA, po tem rangu pa jo prehiti metoda LDA+PCA. Sledijo ji tri metode: PCA, PCAwo3 in PCA1, med katerimi je do ranga 100 najboljša PCAwo3, po tem rangu pa PCA1. Metoda Gaborjevih valčkov je do ranga 70 celo boljša od metod PCA in PCA1, pri višjih rangih pa mu uspešnost prepočasi narašča, tako da je pri višjih rangih celo slabša od metod ICA2 in PCA2. Najslabše pa se je odrezala metoda ICA1+$L1$.

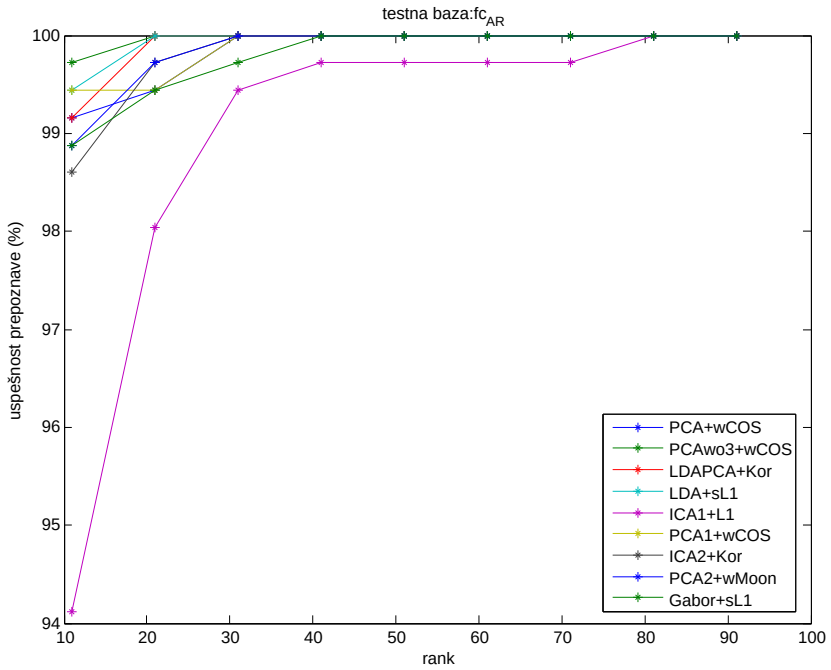
Slika 9.15 prikazuje uspešnost metod pri višjih rangih na testni množici fb podatkovne baze AR. Za podatkovno bazo AR je značilno, da vse metode zelo hitro



Slika 9.15: Krivulja CMS za najboljše kombinacije metod in razdalj testne množice fb_{AR}.

dosežejo najboljši rezultat, to je 100%. Vseeno pa lahko določimo, da je pri testni množici fb najboljša metoda PCA, kateri sledi metoda PCAwo3. Tretja najboljša je metoda PCA1. Do ranga 25 je četrta najboljša metoda LDA, po tem rangu pa jo prehiti metoda LDA+PCA. Sledijo metode PCA2, ICA2 in ICA1. Metoda Gaborjevih valčkov deluje do ranga 20 celo boljše kot metodi LDA in LDA+PCA, po tem rangu pa uspešnost ne narašča več, zato se po rangu 55 izkaže za najslabšo metodo in po rangu 120 kot zadnja metoda doseže tudi najvišjo vrednost.

Slika 9.16 prikazuje rezultate uspešnosti metod pri višjih rangih na testni množici fc podatkovne baze AR. Večinoma so tudi pri testni množici fc vse metode zelo uspešne in dosežejo maksimalno vrednost do 30 ranga, malce slabše se zopet izkaže metoda Gaborjevih valčkov, najslabše pa se je odrezala metoda ICA1. Po uspešnosti si metode sledijo tako: PCAwo3, LDA, LDA+PCA, PCA2, ICA2, PCA1, PCA, Gabor in ICA1.



Slika 9.16: Krivulja CMS za najboljše kombinacije metod in razdalj testne množice fc_{AR}.

9.4.3 Zaključek in ugotovitve

S pomočjo preizkusov nad podatkovnima bazama FERET in AR smo ugotovili, da so trenutne metode za prepoznavanje obrazov zelo uspešne (več kot 90%) v primeru, ko neznani obrazi vsebujejo samo različen izraz na obrazu (testni množici $fafb$ in fb_{AR}). Nekoliko slabšo uspešnost dosežemo na obrazih različnih osvetlitev (testna množica $fafc$). Na tej testni množici je bila najuspešnejša metoda Gaborjevih filtrov, ki je v primerjavi z metodami na osnovi videza dosegala veliko boljše rezultate (uspešnost 88%). Rezultate nad množico obrazov različnih osvetlitev (množica fc_{AR}) pa lahko izboljšamo tudi z reprezentativnejšo učno množico. Učna množica podatkovne baze AR namreč vsebuje za posamezno osebo 7 različnih slik obrazov in tako omogoča, da smo tudi z metodami na podlagi videza nad množico fc_{AR} dosegli več kot 90% uspešnost. Na splošno je bila uspešnost vseh metod boljša nad podatkovno bazo AR kot nad podatkovno bazo FERET. Najslabše so se metode odrezale na obrazih, pri katerih je preteklo že veliko časa od prvega snemanja (testni bazi $dup1$ in $dup2$). Na preizkusu smo za ti dve množici namreč dosegli uspešnost pod 60%.

Rezultate najboljših kombinacij metode in razdalje za posamezno podatkovno bazo prikazujemo v tabeli 9.22.

testna množica	$fafb$	$fafc$	$dup1$	$dup2$	fb_{AR}	fc_{AR}
metoda	Gabor	Gabor	LDA+PCA	LDA	Gabor	Gabor
razdalja	Can	L1	Kor	L1	sL1	Can
uspešnost	92,72%	88,14%	59,00%	56,84%	92,86%	95,80%

Tabela 9.22: Najboljši rezultati uspešnosti prepoznave obrazov za posamezno testno množico.

Ugotovili smo tudi, da deluje metoda PCA brez prvih treh komponent bolje od osnovne metode PCA. Metoda LDA v kombinaciji z metodo PCA se je izkazala za boljšo kot različica metode LDA, ki deluje direktno nad slikovnimi podatki. Dokazali smo, da lahko samo s pomočjo metod PCA1 in PCA2, ki sta predkoraka posamezne arhitekture metode ICA, dosežemo enake ali celo boljše rezultate, kot s pomočjo časovno zahtevnega postopka iskanja neodvisnih komponent. Metoda Gaborjevih valčkov se je izkazala za zelo uspešno metodo zlasti na obrazih različnih osvetlitev. Metoda ICA1 pa se je na splošno izkazala za najslabšo metodo.

Od razdalj so se kot zelo obetavne izkazale utežena kosinusna razdalja ($wCOS$) v kombinaciji z metodama PCA in PCAwo3, korelacijska razdalja (Kor) predvsem v kombinaciji z metodami LDA+PCA, ICA2 in PCA2 ter razdalja Canberra v kombinaciji z metodo Gaborjevih valčkov. V nekaterih primerih se je za zelo uspešno izkazala tudi najpreprostejša razdalja $L1$ ali njena spremenjena različica $sL1$.

S pomočjo preizkusov pri višjih rangih smo ugotovili, da ne velja vedno, da je najboljša metoda pri rangu ena tudi najboljša metoda pri višjih rangih. Rezultate najuspešnejših kombinacij metode in razdalje za posamezno testno množico pri višjih rangih prikazuje tabela 9.23.

testna baza	$fafb$	$fafc$	$dup1$	$dup2$	fa_{AR}	fc_{AR}
metoda	PCA	Gabor	LDA+PCA	< 100 : LDA > 100 : LDA+PCA	PCA	PCAwo3
razdalja	wCOS	L1	Kor	Can Kor	wCOS	wCOS
enako kot rang 1	NE	DA	DA	NE	NE	NE

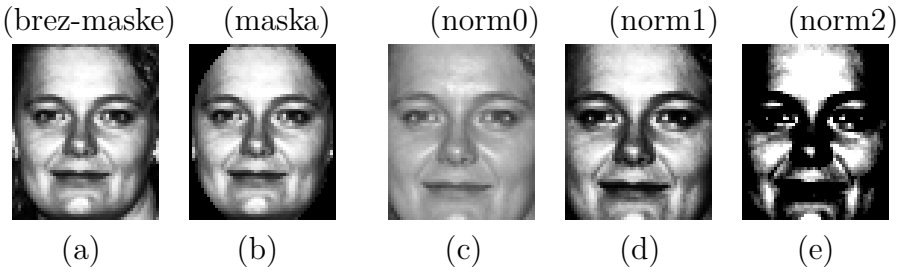
Tabela 9.23: Najboljše kombinacije metode in razdalje za posamezne testne množice pri višjih rangih.

9.5 Rezultati preizkusov pri različnih normalizacijah

S pomočjo naslednjih preizkusov smo poskušali ugotoviti, kako vpliva različen normalizacijski postopek na uspešnost prepoznavanja obrazov. Različne postopke normalizacije smo podrobneje predstavili v poglavju 5. V preizkusih smo uporabili najboljše kombinacije metod in razdalj, ki smo jih določili v prejšnjem poglavju. Vse metode lahko razdelimo v dve glavni skupini, na osnovi videza in na osnovi značilk, zato tudi predvidevamo, da bo normalizacijski postopek različno vplival na metode iz različnih skupin. Kot predstavnika prve skupine, na osnovi videza, smo preizkusili metodo PCA, kot predstavnika druge skupine pa metodo na osnovi Gaborjevih valčkov. Obe metodi smo preizkusili na testnih množicah podatkovne baze FERET. Metodo PCA smo preizkusili v kombinaciji z uteženo kosinusno razdaljo ($wCOS$), metodo Gaborjevih valčkov pa z razdaljo Canberra (Can).

9.5.1 Maskiranje in svetlobna normalizacija

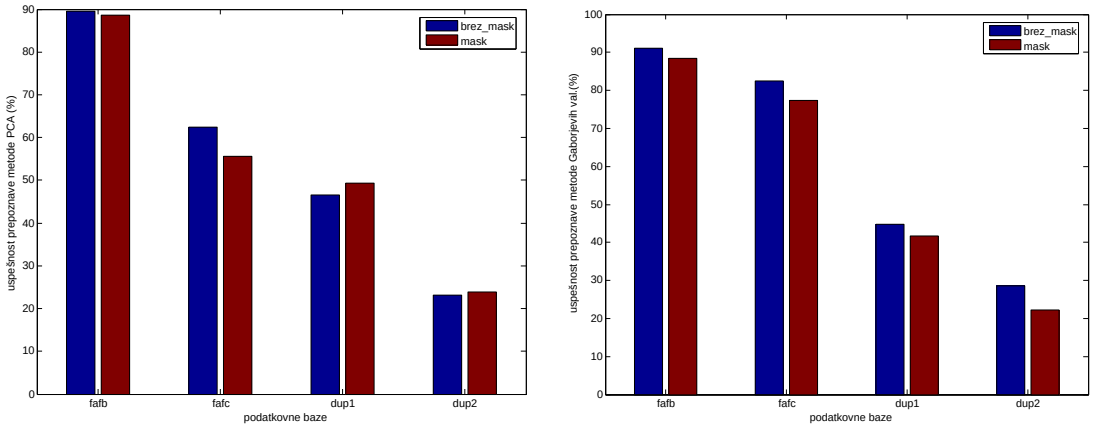
S pomočjo prvega preizkusa smo preverili, ali z dodajanjem maske na slike obraza izboljšamo uspešnost metod za prepoznavanje obrazov. Slika 9.17 (a) prikazuje primer slike obraza, na kateri je viden celoten obraz in slika 9.17 (b) primer slike obraza, na kateri obraz maskiramo z ovalno masko.



Slika 9.17: Primer (b) maskirane slike obraza in različno svetlobno normaliziranih slik: (c) brez normalizacije, (d) izravnava histograma in (c) normalizacija s pomočjo povprečja in standardne deviacije.

Slika 9.18 (a) prikazuje rezultate uspešnosti prepoznave metode PCA na vseh štirih testnih množicah baze FERET za sliki 9.17 (a) in (b). Slika 9.18 (b) pa prikazuje rezultate uspešnosti, ki smo jih dobili s pomočjo Gaborjeve metode.

Pri metodi PCA vidimo, da dobimo z uporabo maske na testnih množicah $fafb$ in $fafc$ malenkost slabše rezultate. Uspešnost se malenkostno izboljša na bazah $dup1$ in $dup2$. Pri metodi Gaborjevih valčkov pa so se maskirani obrazi izkazali slabše na vseh testnih množicah. Zaključimo lahko, da maskiranje obrazov na podatkovni bazi FERET ne doprinese izboljšanja pri prepoznavanju obrazov. Takšen zanemarljiv vpliv maskiranja lahko pripišemo tudi dejstvu, da so bili vsi obrazi po-



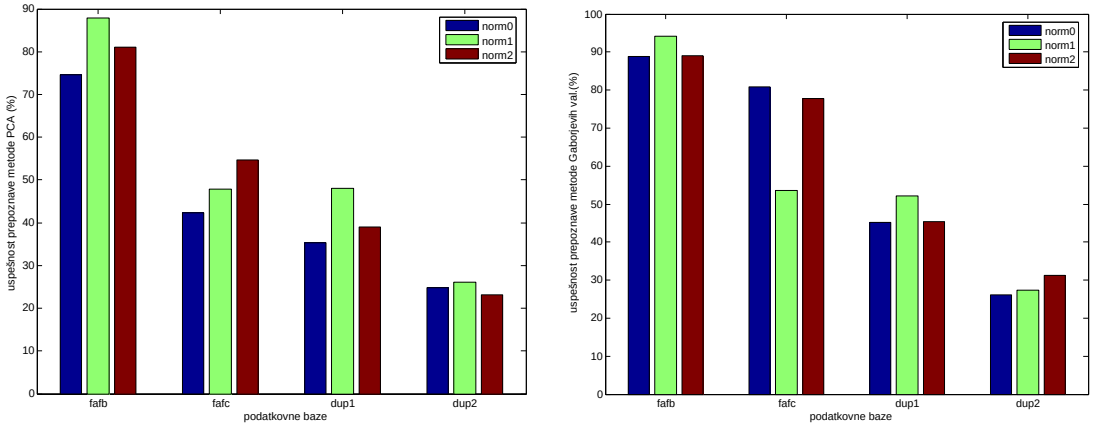
(a) (b)

Slika 9.18: Primerjava rezultatov uspešnosti na maskiranih slikah obrazov podatkovne baze FERET s pomočjo metode (a) PCA in (b) Gaborjeve metode.

datkovne baze FERET zajeti na enotnem ozadju, zato ozadje v našem primeru ni moteče za prepoznavanje obrazov.

S pomočjo drugega preizkusa smo preverili, kako na uspešnost vpliva svetlobna normalizacija. V ta namen smo sliko obraza (slika 9.17 (c)) svetlobno normalizirali s pomočjo dveh različnih postopkov: z izravnavo histograma (slika 9.17 (d)) in s postopkom, pri katerem smo od slike odšteli povprečno vrednost in jo delili s standardno deviacijo (slika 9.17 (e)).

Slika 9.19 (a) prikazuje rezultate uspešnosti prepoznave metode PCA za različno svetlobno normalizirane slike obrazov (sliki 9.17 (d) in (e)). Slika 9.19 (b) pa rezultate, ki smo jih dobili s pomočjo Gaborjeve metode.



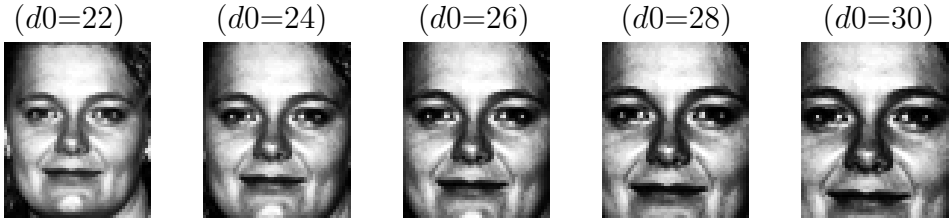
(a) (b)

Slika 9.19: Primerjava uspešnosti različno svetlobno normaliziranih slik obrazov baze FERET s pomočjo metode (a) PCA in (b) Gaborjeve metode.

Vidimo, da v vseh primerih svetlobna normalizacija slik obrazov izboljša uspešnost prepoznave. Izjema je samo testna množica $fafc$ pri Gaborjevi metodi. Izkaže se namreč, da je za ločevanje svetlobno različnih obrazov bolje uporabiti metodo Gaborjevih valčkov na originalnih slikah, kot pa na svetlobno normaliziranih s pomočjo postopka izravnave histograma. Zaradi tega je pri Gaborjevi metodi bolje uporabiti drug normalizacijski postopek, pri katerem odštejemo od slike obraza njeno povprečno vrednost in jo delimo z njeno standardno deviacijo (norm2). Pri metodi PCA pa je boljše uporabiti postopek s pomočjo izravnave histograma (norm1).

9.5.2 Različne vrednosti parametra $d0$ in $e0$

Pri postopku normalizacije moramo določiti razdaljo med očmi (parameter $d0$) in razdaljo med očmi in zgornjim robom slike (parameter $e0$). Parameter $d0$ pri enaki velikosti slike obraza določa tudi "približevanje" obraza. Večji kot je parameter, bolj se obrazu približamo, ter manjši kot je, bolj se oddaljimo od obraza in s tem zajamemo večji del obraza. Primere slike obraza z različnimi vrednostmi parametra $d0$, nad katerimi smo izvršili preizkus, prikazuje slika 9.20. Nad slikami obrazov so zapisane tudi dejanske vrednosti parametra $d0$ v slikovnih točkah pri velikosti slike 60×50 slikovnih elementov.

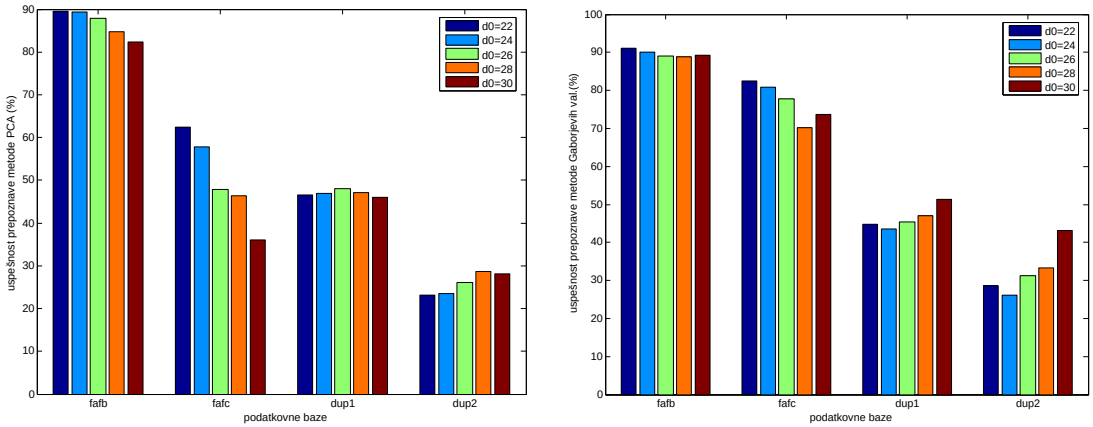


(a) (b) (c) (d) (e)

Slika 9.20: Primeri različno normalizirane slike obraza za različne razdalje $d0$ med očmi. Vrednost parametra določa število slikovnih elementov med očmi pri velikosti slike 60×50 slikovnih elementov.

Slika 9.21 prikazuje rezultate uspešnosti za metodo PCA in metodo Gaborjevih valčkov nad podatkovno bazo FERET. Iz rezultatov uspešnosti lahko vidimo, da velja za obe metodi, da sta uspešnejši pri manjši razdalji med očmi. V tem primeru vsebuje slika celoten obraz z delno okolico in ne samo značilk obraza. Izjema je samo testna množica $dup2$, ki je tudi sicer najzahtevnejša. Slike te množice večinoma vsebujejo zelo različne pričeske, posledica česar so boljši rezultati, če se omejimo samo na obrazne značilke. Na splošno pa se izkaže, da metode lažje razlikujejo obraze, če imajo na voljo poleg obraznih značilk še druge značilnosti, kot so brada, pričeska in oblika obraza.

Naslednji preizkus smo izvedli nad slikami z različnim parametrom $e0$. Parameter $e0$ določa razdaljo med očmi in zgornjim robom slike. Z večanjem parametra pomikamo obraz navzdol in s tem zakrivamo spodnje značilke obraza (brada, usta),



(a) (b)

Slika 9.21: Primerjava uspešnosti prepoznave obrazov z različnimi razdaljami med očmi podatkovne baze FERET s pomočjo metode (a) PCA in (b) Gaborjeve metode.

z manjšanjem pa pomikamo obraz navzgor in s tem prikrivamo del lasišča in čelo. Primere slik, ki smo jih uporabili v tem preizkusu prikazuje slika 9.22. Nad posamezno sliko obraza je zapisana tudi dejanska vrednost parametra $e0$ v slikovnih točkah pri velikosti obraza 60×50 slikovnih elementov.

($e0$=18) ($e0$=21) ($e0$=24) ($e0$=27) ($e0$=30)

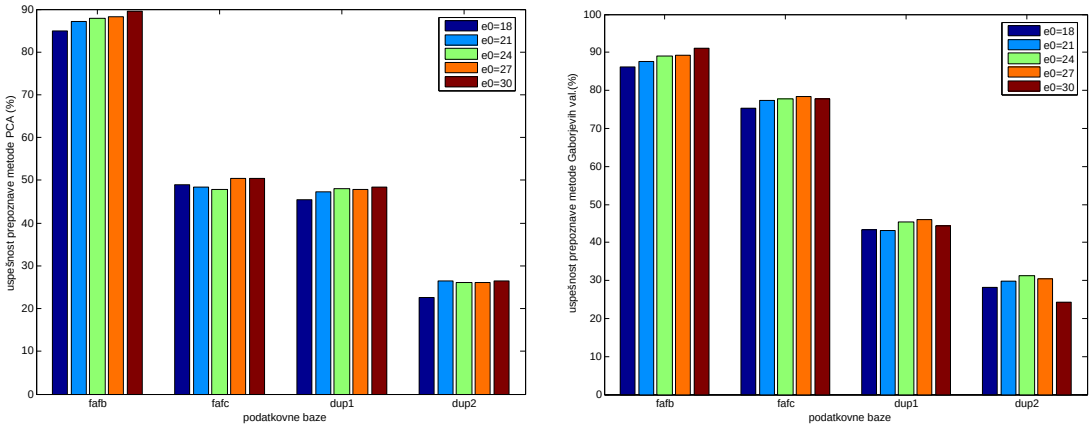
(a) (b) (c) (d) (e)

Slika 9.22: Primeri različno normalizirane slike obraza z različno zamaknjenimi obrazi. Zamik določa parameter $e0$, ki določa razdaljo med očmi in zgornjim robom.

Slika 9.23 prikazuje rezultate preizkusa za metodo PCA in metodo Gaborjevih valčkov nad slikami normaliziranimi z različnimi vrednostmi parametra $e0$. Iz rezultatov lahko vidimo, da je vpliv parametra $e0$ zelo majhen. Samo pri testni množici $fafb$ (obrazi različnih izrazov) se za uspešnejše izkažejo slike z večjim parametrom $e0$. V tem primeru se izkaže, da je bolj pomembna informacija iz zgornjega dela obraza (na primer lasišče), kot pa spodnji del obraz (na primer brada).

9.5.3 Različni deli obraza

Do sedaj smo v preizkusih upoštevali celoten obraz. Pojavi pa se vprašanje ali je kakšen del obraza za prepoznavanje pomembnejši od ostalih. Mogoče pa je kakšen



(a) (b)

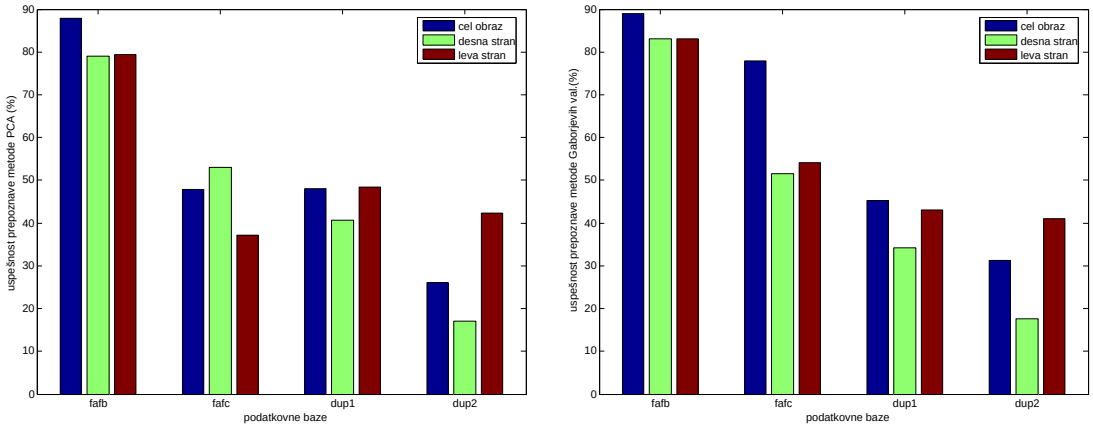
Slika 9.23: Primerjava uspešnosti različno zamaknjenih obrazov podatkovne baze FERET s pomočjo metode (a) PCA in (b) Gaborjeve metode.

del obraza za samo prepoznavanje moteč in je bolje, da ga pri prepoznavanju ne upoštevamo. V prvem delu preizkusa smo preizkusili uspešnost prepoznave samo na levi oziroma desni polovici obraza. Preverili smo tudi uspešnost prepoznave na samo zgornjem delu obraza, samo spodnjem delu in samo sredinskem delu. Posamezne tako določene dele obraza prikazuje slika 9.24.

(a) (b) (c) (d) (e)

Slika 9.24: Primeri različnih delov obraza: (a) desna, (b) leva stran, (c) zgornji, (d) srednji in (e) spodnji del.

Slika 9.25 prikazuje rezultate uspešnosti za desno in levo stran obraza. Zaradi same primerljivosti prikazujemo tudi uspešnost nad celim obrazom. Pri testni množici $fafb$ lahko vidimo, da s pomočjo leve ali desne polovice obraza dosežemo enake rezultate, kar potrjuje simetričnost obraza. Pri ostalih množicah so rezultati različni, verjetno zaradi tega, ker so posamezni deli obraza različno osvetljeni. Najbolj pride razlika do izraza pri metodi PCA, ki deluje na osnovi videza celotnega obraza. Iz rezultata na testni množici $fafc$ lahko sklepamo, da ima leva stran obrazov različno osvetlitev in posledično zaradi tega dosegamo na tej polovici slabšo prepoznavo. Enako lahko zaključimo pri množici $dup2$, da se je desna stran po preteku časa med snemanji bolj spremenila kot leva stran. Pri metodi PCA opazimo tudi, da se uspešnost metode ne spremeni veliko, če izvedemo prepoznavo samo na polovici obraza. Na takšen način lahko zmanjšamo čas prepoznave brez prevelikega

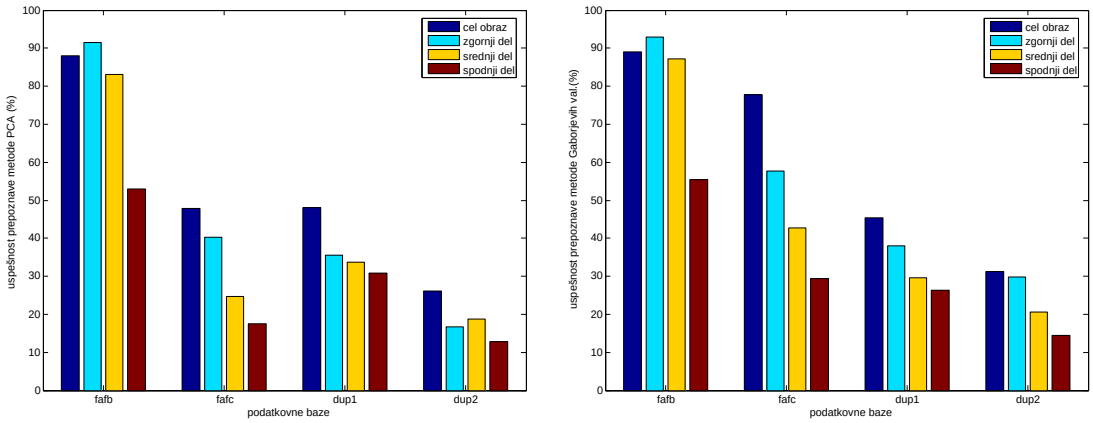


(a) (b)

Slika 9.25: Primerjava uspešnosti za desno in levo stran obrazov podatkovne baze FERET s pomočjo metode (a) PCA in (b) Gaborjeve metode.

zmanjšanja uspešnosti. Pri Gaborjevi metodi je razlika med rezultati uspešnosti nad celim obrazom in samo polovico opaznejša predvsem na testni množici $fafc$.

Slika 9.26 prikazuje rezultate uspešnosti za zgornji, srednji in spodnji del obraza. Tudi tukaj zaradi primerljivosti rezultatov dodamo rezultat uspešnosti za cel obraz. Pri vseh preizkusih je uspešnost nad celim obrazom še vedno najboljša. Vidimo pa,



(a) (b)

Slika 9.26: Primerjava uspešnosti za zgornji, srednji in spodnji del obrazov podatkovne baze FERET s pomočjo metode (a) PCA in (b) Gaborjeve metode.

da je najpomembnejši zgornji del obraza in sicer pri obeh metodah in nad vsemi testnimi množicami. Sledi mu osrednji del obraza. Na podlagi spodnjega dela obraza pa metode najtežje ločijo obraze med seboj. Opazimo lahko tudi, da na testni množici $fafb$ dosegamo boljše rezultate s slikami samo zgornjega dela obraza, kot s slikami, na katerih je cel obraz. Razlago lahko poiščemo v tem, da vsebuje

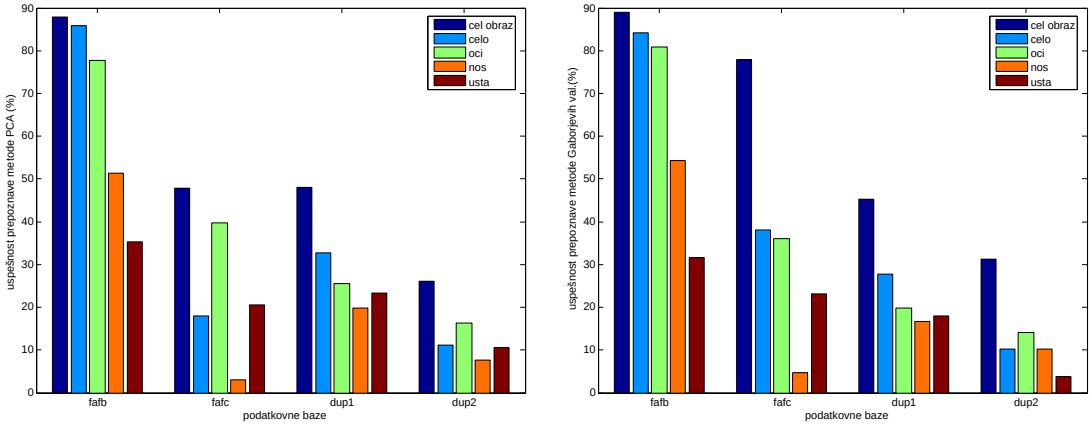
množica $fafb$ obraze različnih izrazov in ker se v tem primeru najbolj spremenijo prav usta lahko sklepamo, da je spodnji del obraza za samo prepoznavo moteči del.

Naslednji preizkus smo izvršili za posamezne dele obraza in sicer samo na čelu s pričesko, očeh, nosu, ustih in bradi. S pomočjo tega preizkusa poskušamo ugotoviti, kateri del obraz je za razlikovanje med različnimi osebami najpomembnejši. Slika 9.27 prikazuje tako določene posamezne dele obraza.

(a) (b) (c) (d)

Slika 9.27: Primeri posameznih delov obraza: (a) čelo, (b) oči, (c) nos in (d) usta.

Rezultate nad posameznimi deli obraza prikazuje slika 9.28. Za primerjavo smo rezultatom dodali tudi rezultate uspešnosti za cel obraz. Na splošno ne moremo



(a) (b)

Slika 9.28: Primerjava uspešnosti za posamezen del obrazov podatkovne baze FERET s pomočjo metode (a) PCA in (b) Gaborjeve metode.

zaključiti, kateri del je za prepoznavo najpomembnejši. Vidimo pa, da s posameznim delom ne moremo doseči enake uspešnosti kot s pomočjo celotnega obraza. Izjema je testna množica $fafb$, kjer lahko s pomočjo samo zgornjega dela, ki nosi informacijo o čelu in pričeski osebe, zelo dobro razlikujemo posamezne osebe. Zelo pomemben del so tudi oči. Ti dve ugotovitvi veljata tako za metodo PCA kot za Gaborjevo metodo. Za to testno množico tudi velja, da so si rezultati na posameznem delu zelo podobni za obe metodi. Naslednji najpomembnejši del za čelom in očmi je nos in na koncu usta. Usta so se v tem primeru izkazale za najmanj uspešen del predvsem zaradi dejstva, da vsebuje testna množica $fafb$ slike obrazov na katerih imajo osebe različen izraz na obrazu, kar se najbolj odraža prav na ustih.

Na testni množici $fafc$ se je s pomočjo obeh metod izkazalo za zelo uspešno področje na katerem so oči. Iz tega lahko sklepamo, da ima osvetlitev najmanjši

vpliv na izgled oči. Pri Gaborjevi metodi se je zelo uspešno izkazal tudi skrajno zgornji del obraza. Pri obeh metodah pa se je za najslabše razločevalen del izkazal predel nosu. Iz tega lahko sklepamo, da ima različna osvetlitev največ vpliva na predel okrog nosu, pri katerem nastane tudi največ senc.

Na testni množici $dup1$ je bil zgornji del obraza najuspešnejši za obe metodi, na testni množici $dup2$ pa predel oči. Testna množica $dup2$ namreč vsebuje obraze, pri katerih je preteklo že veliko časa od prvega zajemanja in se je v tem času pričeska že zelo spremenila. V tem primeru so se osebe med seboj najboljše ločile na podlagi oči.

9.5.4 Zaključek in ugotovitve

Na podlagi preizkusov z različnimi postopki normalizacije smo prišli do zelo zanimivih ugotovitev. Ugotovili smo, da v našem primeru pri podatkovni bazi FERET maskiranje slike obraza ne izboljša rezultatov. Svetlobna normalizacija se je izkazala za zelo pomembno, ker pripomore k boljši uspešnosti prepoznave. Za metodo PCA je boljša normalizacija z izravnavo histograma, pri Gaborjevi metodi pa normalizacija, kjer smo od slike odšteli povprečno vrednost in jo delili s standardno deviacijo. Parameter $d0$ je pomemben predvsem pri obrazih različnih osvetlitev in je bolje, da je večji. Pri parametru $e0$ pa se je pokazalo, da nima velikega vpliva na samo uspešnost prepoznave. Ugotovili smo tudi, da lahko dosežemo primerljive rezultate uspešnosti, če upoštevamo samo levo ali desno polovico obraza, s čimer lahko močno zmanjšamo čas prepoznavanja obrazov. Za prepoznavanje se je izkazal za pomembnejšega zgornji del obraza. Pri slikah obrazov različnih izrazov dosegamo samo z zgornjim delom obraza boljše rezultate, kot če bi upoštevali celoten obraz. Tudi na podlagi preizkusa posameznih delov obraza se je izkazal zgornji del obraza, ki vsebuje čelo in pričesko, za najbolj informativnega. Kot zelo pomembne za razlikovanje med posameznimi osebami so se izkazale tudi oči, še posebej pri različno osvetljenih obrazih. Za najmanj pomemben del obraza pa se je izkazal predel nosu.

9.6 Vzporedni in zaporedni hibridni sistemi

V prejšnjem poglavju smo preizkusili uspešnost posameznih metod za prepoznavanje obrazov. Vsaka metoda ima svoje značilnosti, ki se odražajo v različnih uspešnostih v določeni situaciji. Metode so bile namreč različno uspešne nad posameznimi testnimi množicami, nekatere so delovale uspešneje nad enostavnimi množicami, druge so bile uspešnejše nad množico z različnimi osvetlitvami, itd. Na splošno pa ni nobena metoda delovala nad vsemi testnimi množicami najboljše. Najboljša metoda Gaborjevih valčkov se je sicer izkazala za zelo uspešno, vendar se je nad množicami, ki so vsebovale obraze iz daljših časovnih obdobji, izkazala za slabšo. Naslednja pomanjkljivost metode Gaborjevih valčkov je njena časovna kompleksnost v primerjavi z metodami, ki temeljijo na osnovi videza.

Rezultate posamezne metode bi bilo seveda možno še dodatno izboljšali, tako da bi natančno preučili posamezne testne množice in nastavili parametre posamezne metode. S tem bi morebiti izboljšali rezultate za nekaj odstotkov, vendar bi kmalu dosegli omejitve posamezne metode. Izgubili bi tudi na splošnosti metode zaradi česar bi tako prilagojena metoda najverjetneje delovala na drugi testni množici slabše.

Vsaka metoda ima svoje značilnosti, ki so še posebej izstopajoče med metodami, ki temeljijo na osnovi videza obraza (PCA, LDA, ICA) in metodami, ki temelji na posameznih značilkah (Gaborjevi filtri). Zato jih raje, kot da posamezne metode optimiziramo na določeno testno množico, med seboj smiselno združimo, tako da ohranimo njihove dobre značilnosti in odpravimo ali vsaj omilimo slabe. Poznamo dva načina združevanja: vzporedno in zaporedno. Pri prvem načinu delujejo posamezne metode istočasno in na koncu njihove rezultate združimo, pri drugem načinu pa metode nanizamo eno za drugo, kjer končni rezultat predstavlja rezultat zadnje metode. V nadaljevanju bomo predstavili hibridna sistema, ki smo ju na omenjena načina tudi zgradili in preizkusili na testnih množicah baze FERET in AR.

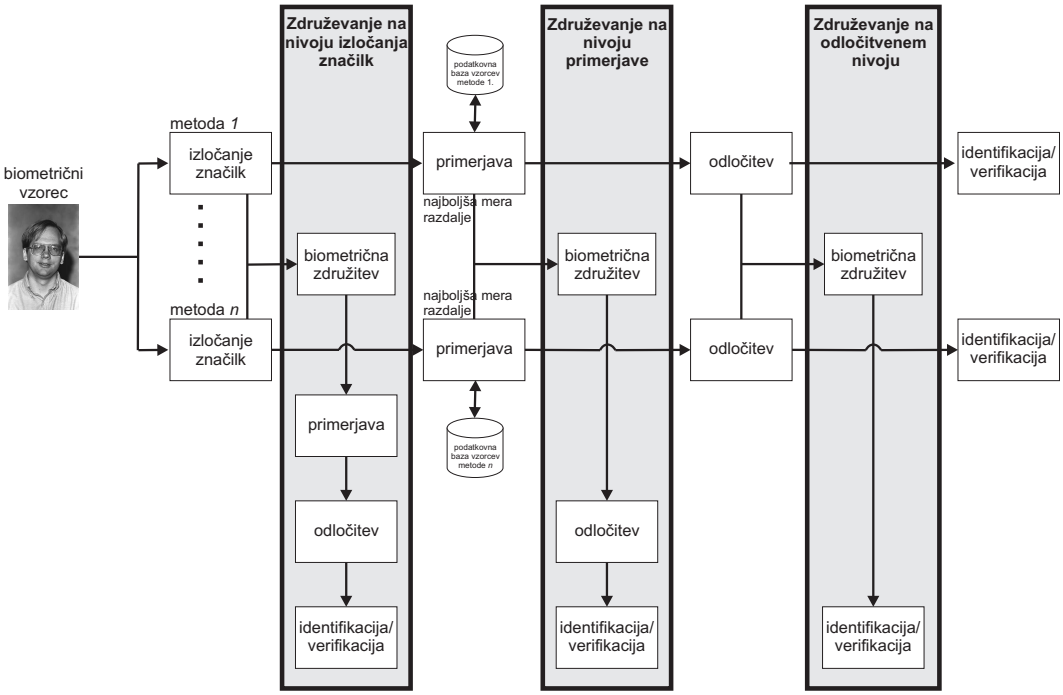
9.6.1 Vzporedni hibridni sistem

Združevanje v vzporedne hibridne sisteme je na področju biometrije zelo razširjeno. Hibridni sistemi, ki jih imenujemo tudi večmodalni biometrični sistemi (angl. *multimodal biometric systems*) vsebujejo večinoma različne biometrične vzorce, katerih analize so potem združene in določajo končno odločitev pri primerjavi z znanimi vzorci. Vzporedni hibridni sistem lahko zgradimo na pet različnih načinov: a) uporabimo lahko različne senzorje, s pomočjo katerih zajamemo isti biometrični vzorec, b) zajamemo lahko različne biometrične vzorce, c) zajamemo lahko več vzorcev, č) zajamemo lahko več različnih vzorcev iste biometrične lastnosti (prstni odtisi različnih prstov, obraz z različnih strani) ali d) uporabimo različne metode za izločanje značilnosti istega vzorca. Običajno se pričakuje večje izboljšanje pri združevanju različnih biometričnih lastnosti človeka, kot če uporabimo zgolj različne metode nad enakim biometričnem vzorcem [43].

Združevanje lahko med samim procesom prepoznavanja izvedemo na različnih stopnjah, kar določa tri različne metodologije združevanja vzporednih hibridih sistemov: a) združevanje na nivoju izločanja značilk, b) združevanje na nivoju primerjave in c) združevanje na nivoju odločitve. Slika 9.29 nam prikazuje vse tri možne načine združevanja biometričnih podatkov.

Pri združevanju na nivoju izločanja značilk posamezna metoda iz vzorca najprej izloči določene značilke. V idealnem primeru lahko že na podlagi značilk, ki jih dobimo iz posamezne metode enolično določimo osebo, običajno pa izločene značilke določajo samo podmnožico vseh oseb. Šele s pomočjo združevanja značilk iz posamezne metode dobimo vektor višje dimenzije, s pomočjo katerega lahko osebo identificiramo z večjo verjetnostjo.

Pri združevanju na nivoju primerjave pa posamezna metoda svoje vektorje značilk



Slika 9.29: Tri različne metodologije združevanja vzporednega hibridnega sistema: na nivoju izločanja značilk, na nivoju združevanja in na odločitvenem nivoju.

preda posameznemu algoritmu za primerjavo vektorja s shranjenimi vektorji vzorcev v podatkovni bazi. Posamezne ocene podobnosti se nato združijo v enotno oceno na podlagi katere se nato izvrši odločanje. Za združevanje se najpogosteje uporabljajo metode nevronskih mrež, diskriminantna analiza ali pa pravilo vsote.

Pri združevanju na nivoju odločitve pa združimo končne odločitve posamezne metode. Običajno se pri tem uporablja pravilo glasovanja. V primerih, če kateri od metod bolj zaupamo kot ostalim, pa lahko uporabljamo tudi uteži, na podlagi katerih končni rezultat kakšne od metod bolj utežimo od ostalih.

Večje izboljšanje lahko pričakujemo pri združevanju na nivoju izločanja značilk, kot pa na drugih dveh nivojih. Pri združevanju na prvem nivoju imamo namreč opraviti s podatki, ki vsebujejo največ informacije in tako imajo posamezne metode na nivoju primerjave najmanj vpliva na končno odločitev. Združevanje na nivoju značilk pa velja za najtežje, ker je velikokrat zelo težko združiti različne vrste značilk, še posebej če gre za različne biometrične modalnosti, na primer obraz v kombinaciji s šarenico ali prstnim odtisom. Pri združevanju je zelo pomembno tudi, da so biometrični podatki med seboj kar se da neodvisni. To pomeni, da lahko pričakujemo večje izboljšanje, če združujemo različne modalnosti, kot na primer sliko obraza in prstni odtis, kot pa če združimo več različnih slik obraza. V našem primeru imamo kot primer biometrične modalnosti opraviti s frontalnimi slikami obrazov, zato večjega izboljšanja ne pričakujemo.

Problem združevanja na nivoju značilk je opazen tudi v literaturi, kjer se večinoma

uporablja združevanje na nivoju primerjave ali na nivoju odločitve. Tako sta avtorja Marcialis in Roli [64] na nivoju primerjave uspešno združila metodi LDA in PCA in dosegla največje izboljšanje s postopkom k-najbližjih sosedov. Avtor Lu et al. [60] je združil poleg metod LDA in PCA še metodo ICA. Kot pravilo združevanja je uporabil pravilo vsote in mrežo RBF. S kombinacijo vseh treh metod in pravilom vsote je v primerjavi s posamezno metodo dosegel boljše rezultate. Po drugi strani pa je avtor Daugman [15] v svojem delu pokazal, da lahko dosežemo boljše rezultate na podlagi samo ene metode, kot pa z združitvijo več metod na nivoju odločitve.

Za izgradnjo našega hibridnega sistema smo se odločili, da podatke združimo na nivoju primerjave. Kot smo že omenili, lahko na nivoju primerjave uporabimo različne postopke združevanja. Odločili smo se za pravilo vsote, ki se v primerjavi z ostalimi postopki izkaže za najuspešnejše [51]. Pravilo vsote definiramo kot:

$$s = \frac{1}{2}(\beta s^{(m1)} + (1-\beta)s^{m2}), \tag{9.1}$$

kjer sta $s^{(m1)}$ in $s^{(m2)}$ oceni izhoda iz posamezne metode, končno oceno pa določa parameter s, ki ga izračunamo tako, da z utežjo β ustrezno utežene končne ocene med seboj seštejemo. Pred samim združevanjem pa moramo posamezne ocene normalizirati. Za normalizacijo smo uporabili pravilo min-max:

$$\mathbf{s} = \frac{\mathbf{s}' - \min(\mathbf{s}')}{\max(\mathbf{s}') - \min(\mathbf{s}')} \ , \tag{9.2}$$

kjer vektor $\mathbf{s}'$ določa množico ocen, ki jih dobimo na izhodu algoritma za primerjavo testnega vzorca z vsemi vzorci iz podatkovne baze in vektor $\mathbf{s}$ normalizirane ocene na območju [0,1].

Tako naš vzporedni hibridni sistem vsebuje na vhodu biometrični vzorec obraza. Za združevanje smo uporabili vse predstavljene metode za prepoznavanje obrazov, ki uporabljajo različne tehnike izločanja značilk. Združevali smo končne ocene primerjave, za postopek združevanja pa smo uporabili pravilo vsote. Najboljše izboljšanje smo pričakovali pri metodah, katerih rezultati so si med seboj bolj neodvisni, to je v našem primeru med metodo Gaborjevih valčkov in metodami iz skupine na osnovi videza.

9.6.2 Preizkus in rezultati vzporednega hibridnega sistema

Hibridni sistem smo zgradili tako, da smo združili rezultate primerjave posameznih metod za prepoznavanje obrazov. V tabeli 9.24 prikazujemo rezultate uspešnosti tako zgrajenega sistema, kjer posamezen stolpec predstavlja različno število združenih metod. Tako prvi stolpec prikazuje rezultate sistema, kjer smo združili samo dve metodi, drugi kjer smo združili tri metode, itd. Preizkusili smo vse možne kombinacije metod, v tabeli pa navajamo samo najboljše. Poleg rezultata tako navajamo

še imena metod, katere smo v preizkušenem sistemu združili (znak + označuje, da smo omenjeno metodo v sistem dodali, znak - pa da smo jo odvzeli).

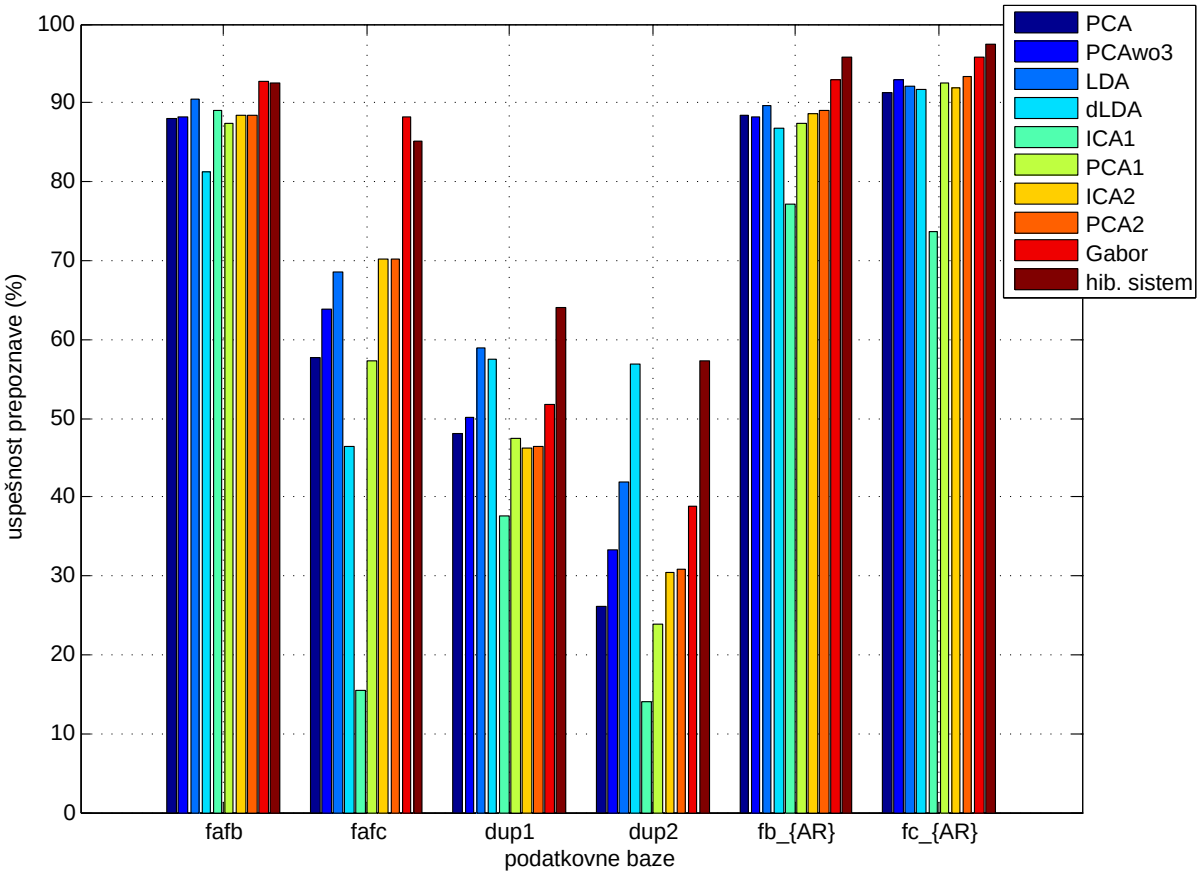
	vzporedni hibridni sistem: število združenih metod							
	2	3	4	5	6	7	8	9
$fafb$	92,47	92,38	91,97	91,80	91,30	90,96	90,63	90,29
metode	LDA+ICA1	+dLDA	+PCA	+ICA2	+PCA2	+PCAwo3	+PCA1	+Gabor
$fafc$	85,05	82,99	80,41	78,87	77,32	76,29	75,26	71,65
metode	LDA+Gabor	+PCAwo3	+ICA2	+PCA2	+dLDA	+PCA1	+PCA	+ICA1
$dup1$	63,99	63,85	60,39	58,03	56,93	55,54	54,71	53,88
metode	LDA+dLDA	+PCAwo3	+Gabor	+PCA	+PCA1	+ICA2	-PCA +ICA1 +PCA2	+PCA
$dup2$	57,27	46,58	42,74	41,03	38,46	36,75	35,90	35,47
metode	LDA+dLDA	+PCAwo3	+Gabor	+ICA2	+PCA2	+ICA1	+PCA1	+PCA
fb_{AR}	95,80	95,38	94,96	94,12	93,70	93,28	93,07	92,86
metode	PCA2+Gabor	+PCA1	+LDA	+PCAwo3	+dLDA	+ICA1	+ICA2	+PCA
fc_{AR}	96,64	97,48	97,20	96,64	96,64	96,64	96,64	95,52
metode	PCA2+Gabor	+dLDA	-PCA2 +PCAwo3 +PCA1	-PCA1 +PCA +PCA2	+LDA	+PCA1	+ICA2	+ICA1

Tabela 9.24: Rezultati uspešnosti različnih vzporednih hibridnih sistemov z različnim številom združenih metod nad posameznimi testnimi množicami podatkovnih baz FERET in AR. Oznaka dLDA označuje direktno metodo LDA, ter oznaka LDA metodo Fisherjevih obrazov, kjer se pred samo metodo LDA uporablja metodo PCA (LDA+PCA).

Iz rezultatov (tabela 9.24) vidimo, da deluje vzporedni hibridni sistem najboljše v primeru, kjer smo združili dve metodi. Z združevanjem več metod namreč ne pridobimo nobenega izboljšanja več, oziroma se uspešnost celo poslabša. Če primerjamo uspešnost hibridnega sistema z uspešnostjo posameznih metod (tabela 9.21) lahko zaključimo, da s pomočjo hibridnega sistema v splošnem rezultate izboljšamo. Največje izboljšanje dobimo pri najzahtevnejših testnih množicah ($dup1$ in $dup2$). Na množicah $fafb$ in $fafc$ baze FERET pa je uspešnost presegla uspešnosti posameznih metod na podlagi videza, vendar se je metoda Gaborjevih valčkov še vedno izkazala za najuspešnejšo.

Če pogledamo, katere metode so se izkazale za najuspešnejše pri združevanju, lahko opazimo določene razlike med podatkovnima bazama FERET in AR. Pri podatkovni bazi FERET se je za zelo uspešno izkazala metoda LDA (metoda Fisherjevih obrazov), ki je bila pri vseh različicah izbrana kot del hibridnega sistema. Sledi ji direktna metoda LDA (dLDA) in spremenjena metoda PCA brez prvih treh komponent (PCAwo3). Pri združevanju se v nasprotju z našimi pričakovanji ni bolje izkazala Gaborjeva metoda, ki je bila za bazo FERET del hibridnega sistema samo pri testni množici $fafc$. Zanimivo je tudi, da so se pri testni množici $fafb$ izkazale za najboljšo kombinacijo samo metode na osnovi videza, čeprav je bila najuspešnejša nad to množico Gaborjeva metoda. Pri bazi AR pa je najuspešnejša kombinacija metode PCA2 in Gaborjeve metode. Na množici fc_{AR} je uspešna tudi direktna metoda LDA (dLDA).

Na koncu še grafično predstavimo rezultate vzporednega hibridnega sistema, ki smo ga zgradili s pomočjo najboljše kombinacije različnega števila posameznih metod v primerjavi z rezultati posameznih metod za različne testne množice.



Slika 9.30: Rezultati uspešnosti vzporednega hibridnega sistema v primerjavi s posameznimi metodami na testnih množicah baze FERET in baze AR.

Iz grafa 9.30 lahko vidimo, da je vzporedni hibridni sistem uspešnejši predvsem na testnih množicah $dup1$ in $dup2$. Izboljšanje smo dosegli tudi na testnih množicah baze AR. Pri testnih množicah ($fafb$, $fafc$) pa je hibridni sistem sicer deloval boljše kot posamezne metode na osnovi videza, vendar uspešnosti Gaborjeve metode ni presegel.

9.6.3 Zaporedni hibridni sistem

Pri zaporednem hibridnem sistemu gre za združevanje različnih metod na način, da izhod prve metode predstavlja vhod v drugo metodo. Takšen sistem se največkrat uporablja za omejevanje preiskovalnega območja. Deluje tako, da s pomočjo prve metode zožimo preiskovalno območje na samo izbrane kandidate med katerimi je tudi

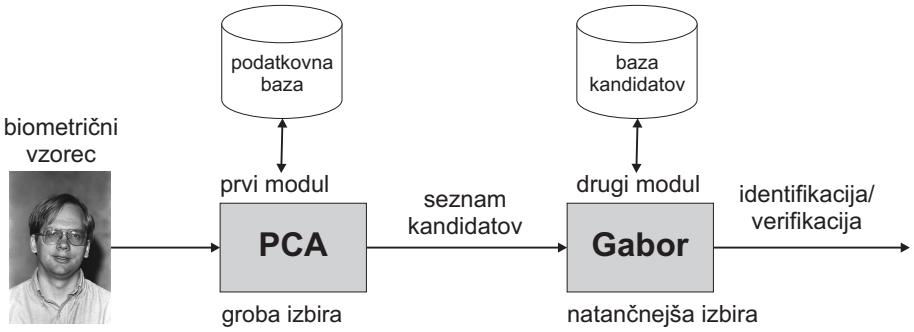
iskana oseba. Nato z nadaljnjimi metodami natančneje preiščemo znotraj obetajočih kandidatov in kot končen rezultat dobimo osebo, ki jo iščemo.

Zaporedni hibridni sistemi so v zadnjem času postali zelo popularni, še posebej tam, kjer imamo opraviti z velikimi bazami podatkov ter tam, kjer sta pomembna tako hitrost kot uspešnost prepoznave. Tako uporablja najuspešnejši komercialni sistem za prepoznavo obrazov FaceIt [39] najrazličnejše metode, ki delujejo v zaporednem hibridnem sistemu. Na prvi stopnji metoda uporablja vektorski vzorec, nato lokalno analizo značilk in na koncu še najzahtevnejšo metodo na osnovi teksture.

Primere zaporednih sistemov lahko zasledimo tudi med drugimi raziskovalci. Prvi takšen primer temelji na primerkih RBF mrež (angl. *Ensambles of Radial Basis Functions*, ERBF) in odločitvenih drevesih (angl. *Decision Trees*, DT). Gutta et al. [31, 30] je zgradil hibridni sistem, kjer je združil primerke RBF mrež z odločitvenimi drevesi. Ugotovil je, da tako zgrajeni hibridni sistem izboljša uspešnost klasifikacije. Tudi avtorja Wallhoff in Rigoll [105, 104] sta za namene prepoznave frontalnega obraza iz slike profila zgradila hibridni sistem, ki je kombiniral nevronsko mrežo in skriti Markov model. Uspešnost takšnega sistema ni bila velika, kar dokazuje, da je zelo težko prepoznati osebo samo na podlagi slike profila.

S pomočjo prave kombinacije metod lahko zgradimo hiter in zelo uspešen zaporedni hibridni sistem. Pomembno je, da je prva metoda, ki omeji množico zelo hitra, ter da je uspešna pri višjih rangih. Na drugi stopnji pa moramo izbrati takšno metodo, ki je najuspešnejša pri rangu ena.

Na podlagi rezultatov preizkusa različnih metod pri rangu ena (tabela 9.22) ter pri različnih rangih (tabela 9.23) smo ugotovili, da na splošno za rang ena deluje najbolje metoda Gaborjevih valčkov, pri višjih rangih pa deluje dovolj dobro že osnovna metoda PCA. Nad testnima množicama $dup1$ in $dup2$ je nekoliko boljša metoda LDA+PCA, vendar lahko, če vzamemo dovolj velik rang dosežemo enako uspešnost tudi z metodo PCA. Na podlagi teh ugotovitev smo za naš zaporedni hibridni sistem uporabili na prvi stopnji metodo PCA, na drugi stopnji pa Gaborjevo metodo. Tako zgrajen sistem prikazuje slika 9.31.



Slika 9.31: Shema zaporednega hibridnega sistema.

Vhod v prvi modul je testna slika obraza, ki predstavlja naš biometrični vzorec. Metoda PCA nato primerja značilke testnega vzorca z vsemi predhodno shranje-

nimi vzorci iz podatkovne baze obrazov. Na podlagi najboljšega ujemanja izbere podmnožico najbolj obetajočih kandidatov, ki predstavlja nato vhod v drugi modul. V drugem modulu metoda Gaborjevih valčkov med izbranimi kandidati izbere najverjetnejšega, ki tudi predstavlja končni rezultat sistema.

Za metodo na prvi stopnji je pomembno, da izbere takšno množico kandidatov, da je med njimi zagotovo tudi iskani obraz. Torej moramo izbrati takšen rang, pri katerem je uspešnost metode PCA najboljša oziroma v najboljšem primeru 100%. Po drugi strani pa moramo paziti, da ni število izbranih kandidatov preveliko, ker v nasprotnem primeru ne bi sistem deloval nič bolje od same Gaborjeve metode, časovna kompleksnost takšnega sistema pa bi bila slabša. S pravo izbiro ranga lahko dosežemo, da je zgrajeni hibridni sistem uspešnejši tako po času kot po učinku. V tabelah 9.25 in 9.26 navajamo uspešnosti metode PCA pri različnih vrednostih ranga za posamezni testni množici baz FERET in AR.

delež celotne baze	rang (FERET)	$fafb$	$fafc$	$dup1$	$dup2$
1%	11	97,66	79,38	68,98	50,85
10%	121	99,67	96,39	91,97	86,75
20%	241	99,75	97,94	95,98	92,74
50%	601	99,92	100	99,17	99,15

Tabela 9.25: Uspešnost metode PCA pri različnih rangih za testne množice baze FERET.

delež celotne baze	rang (AR)	fb_{AR}	fc_{AR}
1%	1	88,45	91,32
10%	12	98,53	99,44
20%	24	100	100
50%	60	100	100

Tabela 9.26: Uspešnost metode PCA pri različnih rangih za testne množice baze AR.

Pri izbiri ranga se moramo odločiti za kompromis med uspešnostjo in časovno zahtevnostjo. Če izberemo visok rang zagotovimo, da ne bomo na prvi stopnji izpustili nobenega iskanega obraza, vendar bo časovna kompleksnost takšnega sistema velika. V primeru majhnega ranga pa se lahko zgodi, da na prvi stopnji izpustimo iskani obraz, ker uspešnost ni 100%, vendar bo število kandidatov manjše in zaradi tega bo tudi časovna kompleksnost manjša. Pri preizkusu različnih rangov se je za najoptimalnejšega izkazal rang 121 pri bazi FERET in 12 pri bazi AR, kar pomeni, da na prvi stopnji zmanjšamo preiskovalno množico obrazov na 10%. V naslednjem koraku lahko torej s pomočjo Gaborjeve metode preiskujemo samo znotraj tako izbranih kandidatov. Velika prednost zaporednega hibridnega sistema je v njeni časovni nezahtevnosti. Za prvo stopnjo smo izbrali metodo PCA, ki deluje zelo

hitro, na drugi stopnji pa zahtevnejšo Gaborjevo metodo, ki potrebuje za primerjavo med različnimi značilkami obraza več časa. V tabeli 9.27 navajamo posamezne čase, ki jih potrebuje posamezna metoda za prepoznavo 100 obrazov v primerjavi s časom, ki ga za to nalogo potrebuje tako zgrajen zaporedni hibridni sistem.

metoda	normalizacija	izloč. značilk	prepoznava	skupaj
PCA	17,2 sek.	0,7 sek.	22,6 sek.	40,5 sek.
Gabor	17,2 sek.	204 sek.	295 sek.	516,2 sek.
Hibrid	17,2 sek.	0,7+204 sek.	22,6+22,9 sek.	267,4 sek.

Tabela 9.27: Časi prepoznave obrazov s pomočjo metode PCA, metode Gaborjevih valčkov in zaporednega hibridnega sistema. Metode so bile izdelane s pomočjo programskega paketa Matlab 7.1 in preizkušene na računalniku s procesorjem AMD Athlon 64 X2 Dual Core 4200+, 2.21GHz, z 2GB spomina.

Vidimo, da se je v našem primeru čas prepoznave enega obraza v primerjavi z Gaborjevo metodo razpolovil iz 5 sekund $(516,2/100)$ na približno 2,5 sekunde $(267,4/100)$. Torej deluje hibridni sistem še enkrat hitreje od Gaborjeve metode. Razlika je toliko večja, če imamo opraviti z večjimi bazami podatkov ali s kompleksnejšimi metodami za prepoznavanje obrazov. V naslednjem poglavju pa bomo predstavili rezultate preizkusa hibridnega sistema nad posameznimi testnimi množicami obeh podatkovnih baz in preverili ali je sistem tudi uspešnejši v primerjavi s posamezno metodo.

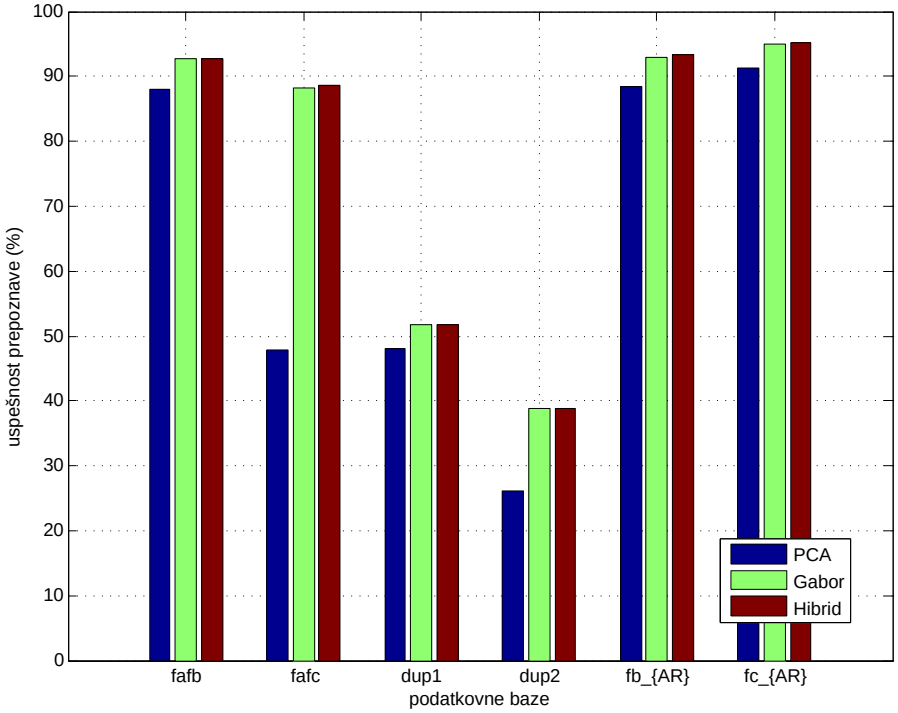
9.6.4 Preizkus in rezultati zaporednega hibridnega sistema

Zaporedni hibridni sistem smo sestavili s pomočjo metode PCA v kombinaciji z razdaljo wCOS na prvem nivoju in metode Gaborjevih valčkov v kombinaciji z razdaljo Can na drugem nivoju. Pri testni množici $fafc$ smo za Gaborjevo metodo uporabili razdaljo $L1$, ki se je pri rangu ena izkazala bolje kot razdalja Can. Pri metodi PCA smo uporabili takšen rang, da smo ohranili 10% celotne baze slik in s tem drastično zmanjšali potreben čas prepoznavanja obrazov (tabela 9.27). Tako zgrajen zaporedni hibridni sistem smo preizkusili na vseh testnih množicah podatkovnih baz FERET in AR. Rezultate uspešnosti prikazujemo numerično (tabela 9.28) in grafično (slika 9.32).

Iz rezultatov lahko vidimo, da je uspešnost zaporednega hibridnega sistema zelo podobna najuspešnejši metodi, to je Gaborjevi metodi. V primerih, kjer metoda PCA izbere drugačne kandidate, kot bi jih izbrala sama Gaborjeva metoda pa so rezultati celo boljši. V primeru, če izgubimo pravega kandidata že na prvi metodi (zaradi prenizke izbire ranga metode PCA) pa je rezultat lahko celo nekoliko slabši (množica $dup2$). V splošnem pa lahko zaključimo, da je zaporedni hibridni sistem po uspešnosti primerljiv z Gaborjevo metodo, deluje pa enkrat hitreje od nje.

testna množica	PCA	Gabor	Hibrid
$fafb$	87,95	92,72	92,80
$fafc$	47,94	88,14	88,66
$dup1$	48,06	51,66	51,66
$dup2$	26,07	38,89	38,89
fb_{AR}	88,45	92,86	93,28
fc_{AR}	91,32	94,96	95,24

Tabela 9.28: Rezultati uspešnosti zaporednega hibridnega sistema v primerjavi z rezultati metode PCA in Gaborjeve metode.



Slika 9.32: Rezultati uspešnosti zaporednega hibridnega sistema v primerjavi z rezultati metode PCA in Gaborjeve metode metodami na testnih množicah baze FERET in baze AR.

9.6.5 Zaključek in ugotovitve

Spoznali smo, da lahko s pomočjo združevanja različnih metod v hibridne sisteme dosežemo izboljšanje uspešnosti sistemov za prepoznavanje obrazov. Pri vzporednih sistemih se sicer čas izvajanja povečuje s številom združenih metod, vendar lahko s pravo računalniško arhitekturo (vzporedno izvajanje na več računalnikov) časovno kompleksnost tudi zmanjšamo. Preizkusili smo različne kombinacije različnega števila metod in ugotovili, da sistem deluje najbolje če združimo samo dve metodi, ki imata med seboj najbolj neodvisne rezultate. Rezultati ostalih metod so si med seboj tako

podobni, da z nadaljnjim dodajanjem ne pripomorejo več k boljši uspešnosti, oziroma jo celo malenkostno poslabšajo. Izboljšanje vzporednih hibridnih sistemov je opazno predvsem pri težjih nalogah, kjer delujejo posamezne metode zelo slabo ($dup1$ in $dup2$). Pričakovali smo, da bo Gaborjeva metoda bolj pogosto del sistema zaradi svoje različne tehnike izločanja značilk, vendar smo jo združili samo pri množici $fafc$ baze FERET in množicah fc in fb baze AR. Zanimivo je hibridni sistem na množici $fafb$ deloval najbolje s kombinacijo samo metod na osnovi videza.

S pomočjo zaporednega hibridnega sistema pa smo s pravo izbiro metod na prvem in drugem nivoju dosegli precejšno pohitritev prepoznavanja obrazov. Za enako nalogo namreč naš sistem potrebuje pol manj časa. Z izbiro Gaborjeve metode na drugem nivoju pa smo zagotovili, da je tudi uspešnost takšnega sistema primerljiva z najuspešnejšo metodo. Kjer pa metodi vračata med seboj neodvisne rezultate lahko dosežemo s pomočjo takšnega sistema celo boljše rezultate. V našem primeru smo dobili boljše rezultate na množicah $fafb$, $fafc$, fb_{AR} in fc_{AR}. To se zgodi v primeru, ko prva metoda izbere takšne kandidate, ki pri drugi metodi ne bi prišli v ožji izbor.

Če torej hočemo povečati uspešnost je smiselno metode združevati v vzporedne hibridne sisteme. Večje izboljšanje bi seveda dosegli, če bi združevali različne modalnosti ampak tudi pri dovolj različnih metodah dosežemo izboljšanje, še posebej pri zahtevnejših nalogah prepoznave. S pomočjo zaporednih hibridnih sistemih pa lahko preiskovanje baze biometričnih podatkov drastično zmanjšamo in s tem tudi čas, ki je potreben za identifikacijo neznane osebe.

9.7 Povzetek poglavja

Izvedli smo neodvisen preizkus za identifikacijo na zaprti množici vseh najpopularnejših metod za prepoznavanje obrazov (PCA, LDA, ICA, Gabor) in njenih različic (dLDA, ICA1, ICA2, PCA1, PCA2) na enotni podatkovni bazi (FERET in AR) ter z enotnim normalizacijskim postopkom. Na ta način smo ugotovili katera metoda deluje najboljše pri določeni nalogi, ki jo določa testna množica. Metode smo pri postopku primerjave preizkusili z različnimi merami razdalje in ugotovili, da delujejo nekatere razdalje bolje v kombinaciji z določenimi metodami. Najboljše kombinacije metod in razdalj smo poleg najboljšega ujemanja (rang 1) preizkusili tudi pri višjih rangih in ugotovili, da ne velja vedno, da je metoda pri rangu 1 tudi najuspešnejša za višje range. Na uspešnost prepoznave vpliva tudi normalizacija, kar smo potrdili z uporabo različnih normalizacijskih postopkov nad bazo FERET. Preverili smo tudi kakšno uspešnost prepoznave dosežemo samo na posameznem delu obraza. S tem smo ugotovili kateri del obraza je za prepoznavo pomembnejši. Z vzporednim hibridnim sistemom smo uspešnost prepoznave nad vsemi testnimi množicami še dodatno izboljšali. Z zaporednim hibridnim sistemom pa smo izboljšali časovno kompleksnost sistema, kar je še posebej pomembno, če imamo opraviti z veliko podatkovno bazo biometričnih podatkov.

Poglavje 10

Uporaba sistemov za prepoznavanje obrazov

10.1 Uvod

Eden izmed razlogov, zakaj se metode za prepoznavanje obrazov v zadnjih letih tako hitro razvijajo, je tudi njihov velik potencial uporabe v vladne in komercialne namene. Že leta 1995 je Chellappa et al. [13] predstavil nekaj sistemov, ki so uporabljali tehnologijo za prepoznavo obrazov ter naštel njihove slabe in dobre lastnosti. Sisteme je kasneje [115] razdelil v 5 glavnih kategorij uporabe. Že leta 1997 je bilo na voljo najmanj 25 sistemov za prepoznavo obrazov 13 različnih podjetji [11]. Od takrat se je zaradi vedno novih področjih uporabe ter izboljšav obstoječih metod za prepoznavanje obrazov število komercialnih izdelkov samo še povečevalo. Najopaznejša podjetja in njihovi izdelki so gotovo tista, ki so se prijavila na zadnji preizkus uspešnosti FRVT (angl. *Face Recognition Vendor Test*) 2002 [78]. Pri teh sistemih največkrat ni znano, katere tehnologije se uporabljajo za detekcijo, prepoznavo in nazadnje za primerjavo, kar dodatno dokazuje, da so ti sistemi vključeni v komercialne izdelke in zaradi tega zaščiteni [37].

Med vsemi biometričnimi metodami velja prepoznava obraza za najmanj vsiljivo tehniko. To dokazuje tudi dejstvo, da je bila metoda za prepoznavo obrazov izbrana s strani mednarodne organizacije letalstva kot najbolj primerna metoda za kontrolo v letalskem prometu.

V nadaljevanju bomo predstavili obstoječe sisteme za prepoznavo obrazov po posameznih področjih uporabe in na koncu trenutne omejitve oziroma težave, s katerimi se takšni sistemi srečujejo. Poglavje bomo zaključili z idejnim primerom uporabe, pametnimi oglasi. Primer pametnih oglasov predstavlja novost na področju oglaševanja, ker se sistem oglaševanja prilagaja potencialnemu kupcu.

10.2 Področja uporabe in tipične aplikacije

V poglavju bomo predstavili najbolj tipične aplikacije oziroma sisteme, ki uporabljajo tehnologijo prepoznavanja obrazov za najrazličnejše namene. Sisteme smo razdelil v 10 različnih kategorij uporabe. Predstavili bomo sisteme s področij identifikacije obraza, kontrole dostopa, varnosti, nadzora, pametnih kartic, organov pregona, digitalnih zbirk, upravljanja z večpredstavnostnimi vsebinami in na področju interakcije med človekom in računalnikom. V vsaki kategoriji bomo predstavili nekaj tipičnih predstavnikov sistemov, ki delujejo v realnem okolju ter opisali njihove omejitve.

10.2.1 Identifikacija obraza

Osnova sistemov za identifikacijo ljudi na podlagi obraza je slika obraza. V nasprotju z ostalimi sistemi za identifikacijo je za te sisteme potrebna zgolj prisotnost osebe. Uporaba biometričnih podatkov za preverjanje identitete ima tudi določene prednosti glede varnosti. Tako lahko z zamenjavo numeričnih gesel z biometričnimi karakteristikami naredimo sisteme bolj prijazne in varne za uporabnika. Zmanjša se tveganje zlorab ukradenih gesel ali izgubljenih kartic. Dobrodošla izboljšava za operaterje je tudi sistem za avtomatsko kontrolo dostopa v varovana poslopja. Največ sistemov za identifikacijo obraza, ki skrbijo za kontrolo dostopa, je nameščenih v prilagojenih okoljih, kjer je število oseb, ki lahko vstopijo, omejeno. Takšne sisteme bomo podrobneje predstavili v naslednjem poglavju. Najprej pa predstavimo dva sistema, ki delujeta na drugih področjih.

Leta 2000 je bil za iskanje duplikatov v registru volivcev Združenih držav Amerike uporabljen program FaceIt podjetja Identix [39]. Obstajali so namreč primeri, ko je bila oseba registrirana več kot enkrat. Med primeri, ki jih je sistem našel, so nato ročno preverili ali sta osebi res enaki.

S sistemom faceFinder podjetja Viisage [95] so opremili številne uprave javnih zaporov in pisarne za izdajo vozniških dovoljenj. Ta program je bil uporabljen tudi za preverjanje naključno izbranih prosilcev za izdajo permanentne vize za vstop v Združene države Amerike. Tako zajete slike prosilcev nadalje uporabljajo pri preverjanju potencialnih oseb, ki bi lahko ogrozile varnost države.

10.2.2 Kontrola dostopa

Sistemi za kontrolo dostopa v prostore oziroma računalniške sisteme navadno uporabljajo zelo majhno podatkovno bazo ljudi, katerih identiteto je potrebno preveriti. Tudi slika obraza je v takšnih sistemih navadno zajeta pod prilagojenimi pogoji (frontalna slika, enakomerna osvetlitev). Zaradi tega so sistemi v takšnih okoljih tudi brez sodelovanja uporabnika s sistemom zelo uspešni. Uporabniku se na primer ni potrebno dotakniti sistema za potrditev prstnega odtisa ali približati očesa za preverjanje očesne šarenice. Še večjo zanesljivost dosežemo, če sistem za prepo-

znavo obrazov združimo z ostalimi sistemi za preverjanje pristnosti, kot so sistemi za preverjanje prstnih odtisov ali očesne šarenice. Raven zadovoljstva med uporabniki takšnih sistemov je zelo visoka. Na področju kontrole dostopa je bilo razvitih veliko komercialnih izdelkov. V nadaljevanju bomo nekatere od njih podrobneje predstavili.

Leta 2000 je podjetje IBM začelo prodajati prenosne računalnike Thinkpad serij A, T in X, z vgrajeno kamero in programom FaceIt za prepoznavo obrazov. Program je nenehno preverjal, kdo je pred zaslonom računalnika. Ko je uporabnik zapustil računalniški sistem, se je aktiviral ohranjevalnik zaslona in s tem se je dostop do sistema onesposobil. Pri vrnitvi uporabnika pred sistem, je program preveril pristnost uporabnika in mu omogočil dostop. Če bi pred sistem pristopil drug uporabnik bi mu sistem na podlagi preverjanja dostop zavrnil.

Izdelek FaceGate [21] je primer še enega izdelka za kontrolo dostopa. Za kontrolo dostopa potrebuje sistem poleg vstopne kode ali kartice tudi sliko iz kamere. Sistem prilagodi matematični model obrazu na vhodni sliki in s tem ustvari identifikacijsko biometrično kodo posameznika. Ko želi oseba vstopiti v zgradbo, sistem preveri vstopno kodo in primerja sliko obraza s shranjeno biometrično kodo. Če se slika obraza ujema, osebi vstop omogoči, v nasprotnem primeru pa zavrne.

Sistem FaceKey [22] kombinira za kontrolo dostopa sistem za prepoznavo obrazov in sistem za prepoznavo prstnih odtisov. S tem se še dodatno poveča zanesljivost sistema. Takšen sistem ne potrebuje vstopnih kod, gesel ali kartic. Prednost takšnih sistemov, ki kombinirajo dva biometrična podatka je tudi v tem, da se pri njih zelo zmanjša verjetnost napake. Sistem lahko deluje kot samostojen sistem ali kot sistem, povezan v lokalno ali celo globalno mrežo.

Sistem FaceVACS [25] omogoča, da lahko običajne sisteme za kontrolo dostopa nadgradimo s sistemom za prepoznavo obraza. Pri vstopu se obraz obiskovalca zajame s pomočjo video kamere. Iz tako dobljene slike se izločijo obrazne značilke, ki se nato primerjajo s shranjenimi značilkami osebe. Dostop je omogočen le v primeru zadostnega ujemanja med značilkami. Za območja, kjer je potrebna večja varnost, se lahko sistem kombinira z vstopnimi karticami. Prilagodljiv komunikacijski vmesnik omogoča enostavno vgradnjo v obstoječe sisteme. Sistemi so med seboj povezljivi, slike vseh obrazov se shranjujejo in zgodovinski dnevnik omogoča časovni pregled vseh vstopov.

V Sloveniji je podjetje Sistemi TAB [94] razvilo integrirani multimedijski sistem za nadzor pristopa in evidenco prisotnosti Smarti. Sitem prepozna človeka po treh biometričnih parametrih, in sicer po obrazu, premikanju ustnic in po glasu. Kot druge podobne naprave omogoča tudi vrsto povezav z različnimi drugimi tehničnimi sistemi, kot so ključavnice, luči, alarmi, stroji in naprave itd. Sestavni deli sistema so enota za prepoznavanje, podatkovni strežnik in odjemalski uporabniški program, ki je lahko nameščen kjer koli v računalniškem omrežju. Pošiljati je mogoče tudi video sporočila; v trenutku, ko sistem identificira osebo, ki ji je posneto sporočilo namenjeno, denimo ob prihodu v zgradbo, ga naslovniku tudi predvaja. Sistem poleg biometrične prepoznave omogoča še uporabo osebne identifikacijske številke (angl.

Personal Identification Number, PIN), ki jo vtipkamo neposredno na zaslon. Sistem omogoča priklop katere koli dodatne opreme (čitalnik prstnih odtisov, magnetnih ali brezkontaktnih kartic itd), tako da lahko stopnjo varnosti po potrebi še povečamo.

Poleg komercialnih izdelkov se v praksi uporablja kar nekaj sistemov, ki so bili razviti v okviru univerz. Omenimo sistem, ki združuje prepoznavo obraza in govora, ki so ga razvili na univerzi Illinois [109]. Vsak uporabnik ima v podatkovni bazi shranjene slike obraza in zvočni zapis stavka, ki ga je v fazi učenja prebral. Za vstop v sistem se preveri vhodna slika iz kamere in ujemanje prebranega stavka s podatki, shranjenimi v podatkovni bazi.

10.2.3 Varnost

Različni varnostni sistemi, ki uporabljajo tehnologijo prepoznave obrazov so bili nameščeni že na številna letališča po vsem svetu. Čeprav lahko pri takšnih sistemih do določene meje kontroliramo pozicijo obraza ter pogoje osvetlitve, predstavlja največjo oviro še vedno preveliko število lažnih alarmov (napačno prepoznanih obrazov). Večina sistemov za prepoznavo obrazov namreč še vedno ne more zagotoviti majhnega števila napačno prepoznanih oseb pri še tako majhnem deležu lažnih alarmov. Zaradi tega je stopnja zaupanja v takšne sisteme pri ljudeh še vedno zelo nizka. V nadaljevanju bomo našteli nekaj primerov programov, ki skrbijo za varnost na letališčih in stadionih.

Oktobra 2001 so za namene varnosti na mednarodnem letališču FYI v Kaliforniji namestili sistem za prepoznavo obrazov podjetja Viisage [95]. Sistem deluje tako, da obvesti uslužbence varnostne službe letališča, če se slika osebe, ki vstopi na letališče, ujema s sliko katere od oseb na seznamu teroristov.

Na letališču v Sydneyu v Avstraliji skrbi za varnost sistem SmartFace podjetja Visionics [39]. Širokokotne kamere zajamejo vse prispele potnike in podatke pošljejo v računalnik, ki te slike obdela in obraze primerja z obrazi na listi nezaželenih, ki jih ima shranjene v podatkovni bazi. Če se kakšen obraz ujema z znanim obrazom, se sproži tihi alarm osebju, ki skrbi za varnost.

Sistem podjetja Imagis [40] pa uporabljajo v sobi za zasliševanje na letališču Oakland v Kaliforniji. Sumljive osebe se primerja s slikami iskanih kriminalcev.

V Maleziji se na 16 letališčih uporablja sistem FaceIt [39], ki skrbi za večjo varnost potnikov in njihove prtljage. Pri kontroli prtljage se s pomočjo zelo majhne kamere zajame video posnetek potnika. Ta video posnetek se shrani na posebno pametno kartico, ki je del karte ter del prtljage. Sistem zagotovi, da lahko samo oseba, ki je predala prtljago, vstopi v prostor za odhode in se kasneje tudi vkrca na letalo. Prav tako se kontrolira tudi prtljaga, če je naložena na pravo letalo. Med vkrcavanjem sistem v realnem času preveri sliko potnika s sliko shranjeno na karti. Dokler se sliki ne ujemata, se tudi prtljaga ne naloži na letalo.

Sistem faceFINDER podjetja Viisage [95] so uporabljali tudi za preverjanje občinstva na stadionu. Na vhodu stadiona so bile nameščene kamere, ki so posnele vsakega obiskovalca, ki je vstopil. Kamere so bile povezane s posebnim kontrolnim

centrom, kjer so se obrazi na slikah primerjali s slikami kriminalcev. Isti sistem je uporabila tudi policija v Avstraliji, ko je na nogometni tekmi preverjala obiskovalce z znanimi huligani, ki so imeli prepoved obiska tekme.

10.2.4 Nadzor

Podobno kot varnostni sistemi, dosegajo tudi nadzorni sistemi, ki uporabljajo tehnologijo prepoznave obrazov, nizko zadovoljstvo pri uporabnikih. Različni svetlobni pogoji, različno obrnjeni obrazi ter drugi spremenljivi faktorji vplivajo na to, da je prepoznava obraza v sistemih za nadzor velike množice ljudi še vedno težavna naloga.

Leta 1998 je podjetje Visionics [39] prvič predstavilo svoj sistem FaceIt za nadzor mestnega središča v Londonu. Po središču so postavili 300 kamer, ki so bile v zaključenem krogu povezane s kontrolno sobo. Mestne oblasti so zatrdile, da je takšen sistem pripomogel k zmanjšanju kriminala za 34%. Podoben sistem so nato postavili še v Birminghamu. Program FaceIt uporabljajo tudi policisti na Floridi, da s pomočjo video kamer preiskujejo ulice in iščejo spolne iztirjence. V Virginiji so program namestili na javno ulico, da pregleduje obraze mimoidočih in jih primerja s 2500 slikami aktualnih kriminalcev, pogrešanih in pobeglih oseb. V New Yorku s pomočjo sistema za nadzor kontrolirajo obiskovalce kipa svobode. V pristanišču kjer obiskovalci zapustijo Manhattan jih posnameta dve kameri in njihove slike primerjata s slikami znanih teroristov. Kameri spremljajta turiste v vrsti, ki čakajo na ladjo.

10.2.5 Pametne kartice

Pametne kartice, ki vsebujejo spominske module in miniaturne procesorje, imajo že dovolj velike zmogljivosti, da jih lahko opremimo z biometričnimi podatki osebe. Večina kartic ima tudi že vgrajeno zaščito ter radiofrekvenčni vmesnik za oddaljeno komunikacijo s bralniki. Tako lahko postanejo pametne kartice s svojo zmožnostjo prenosljivosti in nadgradnje nekakšen vezni člen med virtualnim in fizičnim svetom. Sistem integriran na takšni kartici tako skrbi za preverjanje biometričnih podatkov kot tudi za nadgradnjo z novimi podatki v vsakem primeru ujemanja.

Podjetje Maximus [66] je na pametni kartici združilo biometrične podatke obraza ter podatke prstnega odtisa. S pomočjo takšne kartice lahko sistemi hitreje preverjajo identiteto potnikov, s čimer se skrajša čakanje. Bralniki kartic so namreč nameščeni v posebnih hitrih vrstah, kjer se preveri identiteta potnikov s primerjavo podatkov shranjenih na kartici s tistimi, ki jih naprava zajame med prehodom. Pred izdajo takšne kartice potnika preverijo in mu na podlagi slike obraza in njegovega prstnega odtisa izdelajo kartico.

Sistem ZN-Face [52], ki prav tako združuje biometrične podatke obraza in prstnega odtisa na pametni kartici, se uporablja na Berlinskem letališču za zaščito varovanih območji. Potencialne grožnje kriminalcev, ki so do sedaj uspešno vsto-

pali v zavarovana območja s pomočjo ustreznih preoblek (pilotske uniforme) lahko takšni sistemi uspešno izključijo. Na vsakem vstopnem mestu se zajete značilnosti oseb preverijo s tistimi, ki so shranjene na pametni kartici in tako služijo kot verifikacija.

Tudi v Sloveniji se bo obrazna prepoznava uporabljala za namene izdelave biometričnih potnih listov. V začetku leta 2005 je namreč stopila v veljavo obvezujoča Uredba Sveta (ES) o standardih za varnostne značilnosti in biometrične podatke v potnih listih in potovalnih dokumentih, ki jih izdajajo države članice. Uredba opredeljuje implementacijo dveh biometričnih elementov kot obveznih: obrazno prepoznavo in prepoznavo dveh prstnih odtisov. Biometrični podatki bodo zapisani na brezkontaktnem čipu, ki bo vgrajen v potni list.

Pametne kartice se večinoma uporabljajo za primere verifikacije. Natančnost ujemanja na kartici shranjene slike s tisto, ki jo zajamejo naprave, je odvisna predvsem od časa, ki je minil med zajetjema obeh slik. S tem, ko podatke na kartici posodobimo z novimi, lahko to časovno obdobje skrajšamo. V primeru podatkovnih baz z majhnim številom ljudi, je zanesljivost takšnih sistemov zadovoljiva.

10.2.6 Organi pregona

S pomočjo tehnologije prepoznavanja obrazov in informacij o kriminalcih lahko preiskovalci hitro najdejo in identificirajo osumljenca celo v primerih, ko nimajo popolnih podatkov. Včasih imajo o osumljencu znano samo skico očividca ali slab posnetek iz nadzorne video kamere. Prav zaradi tega je lahko uspešnost sistemov za prepoznavo obrazov v takšnih primerih zelo slaba. Vsekakor pa so sistemi za prepoznavo obrazov lahko zelo dober pripomoček za policijo.

Podjetje Imagis [40] je za varnostne organe pripravilo poseben program, ki omogoča kalifornijskim policistom in detektivom s pomočjo prenosnih naprav preko zaščitene internetne povezave ažuriran dostop do informacij o aretiranih osebah. Sistem, ki vključuje tudi prepoznavo obrazov, upravljanje s slikami in podatkovno bazo, predstavlja policijskim delavcem dodatno pomoč pri njihovem delu. S pomočjo sistema za prepoznavo obrazov in vedno ažurirane podatkovne baze obrazov, policistom ni potrebno več izgubljati ure in ure, da bi identificirali sumljivo osebo. Policist enostavno zajame sliko osumljenca, ki jo lahko na terenu primerja z osebami v zbirki podatkov. Podobno poizvedbo lahko naredi za osebe, ki jih je zajela nadzorna kamera ali zmanjša množico možnih osumljencev na podlagi dodatnega opisa očividca.

Sistem podjetja Imagis so namestili tudi v igralnico za potrebe nadzora in kontrole obiskovalcev. V kombinaciji z globalno podatkovno bazo nezaželenih obiskovalcev in hazarderjev omogoča sistem igralnicam zelo učinkovito varnostno rešitev.

10.2.7 Zbirke digitalnih fotografij

Tehnologije za pridobivanje slik na osnovi opisnega besedila niso več kos vse večjim podatkovnim zbirkam digitalnih fotografij. To poskušajo rešiti nove tehnologije,

ki slike kategorizirajo na podlagi njihove vsebine. Slik ne rabimo več opremljati z besedilom, ampak se le-te avtomatsko opremijo s ključnimi besedami glede na to, kaj prikazujejo. Najenostavnejše metode uporabljajo za ugotavljanje vsebine kar lastnosti slike kot so barva in tekstura. Takšne splošne značilke pa imajo svoje pomanjkljivosti, zato jih poskušajo raziskovalci uporabljati v kombinacijami z drugimi metodami za analizo slik in s tem povečati točnost označevanja slik. Med takšne metode spada tudi metoda odkrivanja in prepoznave obrazov na sliki. Čeprav se metoda prepoznave obrazov večinoma uporablja za pridobivanje in indeksiranje obraznih slik (baza osumljencev), se jo uporablja tudi za preiskovanje podatkovnih zbirk, ki vsebujejo tako slike obrazov kot tudi slike neobrazov (družinski albumi).

Uspešnost takšnih sistemov je predvsem zaradi zelo velikega števila slik in s tem obrazov, ki jih moramo poiskati, še vedno relativno nizka. Uspešnost je zmanjšana tudi zaradi različnih pogojev, pod katerimi se obrazi na slikah pojavljajo.

Poznamo dva tipa sistemov, ki se uporabljata nad podatkovnimi zbirkami digitalnih fotografij. Prvi sistemi uporabljajo metode prepoznavanja obrazov za označevanje slik, drugi pa tehnike označevanja slik uporabljajo kot pomoč pri iskanju osebe.

Tako nam sistemi prvega tipa omogočajo avtomatsko organizacijo in s tem enostavnejše iskanje po veliki zbirki fotografij domačega albuma. Primer takšnega programa je FotoFile [53], ki s pomočjo prepoznave obraza avtomatsko izvede identifikacijo vseh oseb na fotografijah in na podlagi tega fotografije samodejno označi in združi v skupine. Uporabnik lahko takšno izbiro sistema potrdi ali ovrže in s tem sistem dodatno uči.

Sistemi drugega tipa pa nam omogočijo, da v veliki zbirki obrazov poiščemo osebo samo po spominu. Avtorja Navarrete in del Solar [72] sta uporabila metodo, ki temelji na odgovoru, ki ga vrne uporabnik, glede na podobnost prikazanih slik s tisto, ki jo ima v spominu. Idejo sta implementirala s pomočjo drevesne strukture samo-organizacijskih mrež (angl. *Tree-Structured Self-Organizing Map*, TS-SOM), ki samodejno organizira slike obrazov v podatkovni bazi. Podobni obrazi se tako nahajajo na sosednjih pozicijah v drevesu mreže. Da pridemo do želene lokacije, kjer se nahaja obraz, ki ga iščemo, nam sistem ponudi naključno izbrane slike, med katerimi moramo izbrati tiste, ki so po našem mnenju najbolj podobne obrazu, ki ga imamo v mislih. Nato nam sistem ponudi naslednjo skupino slik, ki jo skrbno izbere glede na našo predhodno odločitev. Tako nam sistem ponuja nove in nove skupine slik dokler proces iskanja ne konvergira do želene slike. Drugi takšen sistem avtorjev Eickeler in Birlinghoven [19] temelji na skritem Markovem modelu. Raziskave, ki sta jih avtorja naredila, delujejo na podatkovni bazi s 250.000 slikami, kar dokazuje, da je metoda primerna tudi za velike zbirke slik.

Tudi sami smo s pomočjo instalacije 15 sekund slave izdelali veliko podatkovno bazo za namene preizkusov in nadaljnje uporabe [90]. Vsaka slika vsebuje najmanj en obraz, ki ga je program za iskanje našel. Tako najdene obraze naprej uporabimo za izgradnjo mozaika slavnih oseb ter shranimo v podatkovno bazo za kasnejše pošiljanje zahtevanega portreta uporabniku. Sistem za iskanje obraza lahko nadgradimo tudi s sistemom za prepoznavanje mimike in s tem vplivamo na umetniško

transformacijo slike izbranega obraza. Nadalje pa lahko obraz obiskovalca tudi prepoznamo in s tem podatkovno bazo slik dodatno organiziramo in omogočimo, da sistem v prihodnje obiskovalca prepozna in mu pri pošiljanju omogoči dostop do vseh njegovih portretov.

10.2.8 Upravljanje z večpredstavnostnimi vsebinami

Obrazi se pojavljajo povsod okoli nas, pri napovedih novic, pri športu, v filmih ter v drugih večpredstavnih vsebinah. Če hočemo takšne večpredstavne vsebine razvrstiti glede na to, kaj prikazujejo, moramo uporabiti metode kot so iskanje, sledenje obrazov in njihova prepoznava ter spremembe skozi čas. Na takšen način lahko video vsebine organiziramo v posamezne segmente in s tem omogočimo preiskovanje, hitro pregledovanje in izdelavo povzetkov. Metode za preiskovanje video vsebin na podlagi obraza so poleg metod, kot so analiza govora, prepoznava govora in drugih tehnik analize slik, zelo močno orodje pri avtomatskem indeksiranju in označevanju večpredstavnih vsebin.

Ena izmed težav pri uporabi metod za prepoznavo obrazov na video vsebinah je ta, da nimamo vnaprej podane galerije slik in tako ne vemo, kdo je najdena oseba na sliki. Metode morajo identiteto najdenih oseb poiskati iz same vsebine.

Haughton [35] je problem rešil tako, da imena nastopajočih poišče na internetnih straneh, ki opisujejo preiskovano video vsebino. Dodatno poišče imena v spremnem tekstu na samem posnetku s pomočjo tehnike OCR (angl. *Optical Character Recognition*). Tako najdena imena potem poveže z osebami, ki nastopajo v videu in s tem zgradi galerijo. Za iskanje obrazov uporablja program FaceIt.

Avtorja Ma in Zhang [62] sta razvila interaktivni uporabniški vmesnik, ki omogoča uporabniku, da predhodno označi osebe na izbranih video segmentih, ki jih sistem izbere tako, da vsebujejo še neznane osebe. Za izbiro ključnih segmentov je uporabljen program za iskanje obrazov. Kasneje pa se za preiskovanje video vsebin uporablja metode prepoznave.

Integriran sistem za upravljanje z večpredstavnimi vsebinami je tako imenovan sistem Infomedia, ki je nastal v okviru projekta na univerzi Carnegi Mellon [103]. Cilj projekta je ustvariti informacijsko digitalno video knjižnico, ki bo pomagala pri učenju. Uporabniki lahko po indeksiranih in arhiviranih video vsebinah hitro iščejo in poizvedujejo z opisnimi poizvedbami kot so: "poišči izseke, na katerih ljudje govorijo", (metoda za iskanje obrazov) ali "poišči intervjuje z določeno osebo" (metoda za iskanje obrazov v kombinaciji s spremnim tekstom).

Sistem iVIEW je še en primer digitalne video knjižnice, ki podpira več jezikov in več načinov delovanja, izdelali pa so ga na Kitajski univerzi v Hong Kongu [61]. Shema prepoznavanja obrazov je zelo podobna tisti, ki jo je predstavil avtor Haughton. Zasnova sistema iVIEW temelji na internetnem uporabniškem vmesniku, ki je povezan s strežnikom.

Avtorja Wang in Chang [106] sta izdelala sistem, ki omogoča iskanje, sledenje in izgradnjo povzetkov iz stisnjenih video vsebin v realnem času. Za iskanje obrazov

na video posnetku, avtorja izkoriščata informacije zapisane v MPEG stisnjenem videu, ki sta jih nadgradila s sledenjem. S sledenjem posameznim osebam skozi čas in prostor se lahko naučita tudi, kako so te osebe med seboj povezane. Orientacije obrazov in svetlobni pogoji so v večini večpredstavnih vsebin zelo različni, kar prispeva k nizki uspešnosti takšnih sistemov.

10.2.9 Interakcija med človekom in računalnikom

Za učinkovito in uporabniku prijazno interakcijo med človekom in računalnikom lahko običajne naprave, kot so miška in tipkovnica, zamenjajo bolj naravne vhodne "naprave", kot so človeški deli telesa (med njimi tudi glava). To dejstvo je motiviralo številne raziskovalce na področjih, kot so sledenje, analiza mimike, sinteza in animacija človeškega obraza.

Čeprav je glavni cilj takšnih uporabniških vmesnikov prepoznava in razumevanje posamezne kretnje človeka, je prvi korak takšnih sistemov posamezne človeške dele poiskati in jim slediti. Kot najbolj razlikovalna značilka se tukaj ponuja barva kože, ki omogoča učinkovito lokalizacijo in slednje objektom, kot so roke in obraz. Posamezne dele telesa najdemo s pomočjo segmentacije tako, da združimo posamezne barvne regije, ki vsebujejo kožno barvo. Kljub temu, da je bilo na področju prilagodljivih barvnih modelov že kar nekaj narejenega, je potrebno to področje še dodatno raziskati.

Od računalnika, ki bi imel sposobnost interakcije s človekom se pričakuje, da ima podobne komunikacijske sposobnosti kot jih ima človek. Ena takšnih sposobnosti je tudi prepoznava razpoloženja človeka. Človek pa svojo razpoloženje najbolj izraža s pomočjo mimike svojega obraza.

Realistična slika 3D modela glave je eden ključnih faktorjev na poti k bolj naravni interakciji med človekom in računalnikom. Grafični model človeka je prava rešitev za vizualno predstavitev informacij. Primeri okolij kjer se takšen model uporablja, so kodiranje videa za prenos po ozki pasovni širini za namen vizualne telekomunikacije, prepoznava na podlagi avdio in vizualne informacije ter animacija govorečega modela glave pri računalniških agentih. V okoljih, kjer je veliko šuma, takšni modeli pripomorejo k boljšemu razumevanju govora in s tem k boljšemu odzivu uporabnika pri interakciji. Izkazalo se je, da virtualni agenti, ki skrbijo za prodajo, zbujajo pri kupcih zaupanje ter da govoreči modeli človeške glave pripomorejo k boljšemu učenju na računalniških sistemih.

10.2.10 Druga področja uporabe

Veliko sistemov za prepoznavo obrazov potrebuje za realizacijo sodelovanje s strokovnjaki na izbranem področju. Največkrat delujejo kot pomočniki ekspertom na tem področju.

Prvi primer takšnih sistemov je sistem za verifikacijo starinskih fotografij. Za zgodovinarje, življenjepisce in zbiratelje starin je izrednega pomena, da potrdimo

identiteto osebe na starinski sliki. Zaradi velike časovne razlike med slikama, ki ju primerjamo ter zaradi večkrat zelo slabe kvalitete starinske slike, predstavlja takšen sistem še vedno izziv.

Drugo področje uporabe so sistemi za prenos obrazne slike. Sliko lahko zakodiramo s pomočjo samo nekaj parametrov, kar je zelo uporabno za prenos slike obraza pri nizkih pasovnih širinah pri video telefoniji ter pri telekonferencah. Namesto, da bi se pošiljale celotne slike ali video, se pošiljajo samo parametri, ki se nato na drugi strani prilagodijo modelu obraza in kar se da čimbolj rekonstruirajo pravo podobo originalnega obraza osebe. Sistem ima potencialno uporabo tudi v namene izobraževanja (v muzejih, v razredu) ter zabave (animirani junaki v računalniških igricah, interaktivni filmi, digitalni fotoaparati).

Sistemi za prepoznavo obraza se uporabljajo tudi kot pomoč ekspertom pri rekonstrukciji obraza iz znanih lastnosti ali ostankov, pri izdelavi fotorobotov ali kot pomoč pri simulaciji staranja skozi čas.

10.2.11 Omejitve trenutnih sistemov

Čeprav se tehnologija za prepoznavo obrazov že veliko uporablja, je to območje uporabe še vedno zelo omejeno. Da bi to spremenili, je potrebno rešiti vsaj še dva problema. Prvi problem je robustnost sistemov v spremenljivih okoliščinah.

Uspešnost prepoznave je v primerih, ko imamo opraviti z veliko količino podatkov, še vedno premajhna. Spremembe osvetlitve, različni položaji obraza in časovna razlika med zajemom testne slike in slike iz galerije to še dodatno poslabšajo. Uspešnost prepoznave pri spremenljivih okoliščinah so preizkusili tudi na zadnjem velikem preizkusu FRVT leta 2002, kjer so preizkusili 10 najboljših komercialnih izdelkov [78]. Med najboljše tri so se uvrstila podjetja Cognitec [25], Visionics (zdaj Identix) [39], Eyematic (zdaj Neven Vision) [41].

Verifikacijo so preizkusili nad množico iz galerije, ki je vsebovala 37.437 oseb, testna množica pa je vsebovala 74.854 oseb, kjer je bila oseba predstavljena s po dvema slikama. Prvi trije najboljši sistemi so na preizkusu, kjer so bili pogoji osvetlitve kontrolirani v zaprtem prostoru, dosegli uspešnost verifikacije 90% pri 1% napake, 80% pri napaki 0.1% in 70% pri napaki 0.01%. Vedeti moramo, da lahko samo v nekaterih sistemih zagotovimo kontrolirano okolje ter da je takšna uspešnost še vedno premajhna na letališčih, kjer imamo opraviti z velikim številom ljudi. Takšen sistem bi lahko uporabili na primer za kontrolo dostopa v situacijah, kjer imamo največ sto ljudi in lahko zagotovimo kontrolirano okolje. Pri preizkusu, kjer se je položaj obraza spreminjal, je na podatkovni bazi s 87 osebami najboljši sistem dosegel identifikacijsko uspešnost 42%, ko se je položaj spreminjal za 45 stopinj levo in desno, ter 52% uspešnost pri spremembi 45 stopinj gor in dol. Pri preizkusu, kjer je bila časovna razlika med slikami eno leto, je uspešnost padla za 5%. Pri različnih osvetlitvah, kjer je bila slika iz galerije posneta v zaprtem prostoru, testna slika pa na prostem, je uspešnost padla pri napaki 1% iz 90% na 60%.

Sistemi za namene nadzora v javnih prostorih so vzbudili tudi skrb o poseganju

v posameznikovo zasebnost. Ameriško združenje civilnih svoboščin (angl. *American Civil Liberties Union*, ACLU) namreč nasprotuje uporabi sistemov za prepoznavo obrazov na letališčih zaradi zelo majhne učinkovitosti in zaradi vmešavanja v zasebnost ljudi.

10.3 Primer uporabe: pametni oglasi

Danes je eden izmed največjih izzivov na področju računalniškega vida, vključevanje vizualnih sistemov v nekontrolirano okolje in s tem povečanje kvalitete življenja. Končni rezultat takšne združitve bi moralo biti okolje v katerem bi bila prisotnost tehnologije nemoteča in s tem tudi računalniki neopazni. Del takšnega okolja bi bili seveda tudi sistemi za prepoznavanje, klasifikacijo in analizo obrazov.

Razširjen sistem, ki smo ga predstavili v uvodu namreč ponuja dodatne možnosti uporabe tudi v okoljih, kjer takšni sistemi do sedaj niso bili prisotni. S pomočjo klasifikacije in analize lahko sisteme uporabimo tudi tam, kjer ne zahtevamo zgolj nalog identifikacije in verifikacije. V nadaljevanju bomo predstavili primer uporabe takšnega sistem za namene oglaševanja.

Spletni prodajalci poskušajo s pomočjo najrazličnejših tehnologij izoblikovati profil kupca. Oglaševalci se poskušajo s svojim oglaševanjem čimbolj približati potencialnemu potrošniku. S pomočjo našega sistema se bodo lahko oglaševalci še dodatno približali kupcu. Ideja našega sistem je v tem, da združimo tehnologijo iskanja, prepoznavanja, klasifikacije in analize obraza z oglaševalnimi panoji. Današnja tehnologija že omogoča, da postavimo v zunanje okolje ekrane velikega formata in visoke ločljivosti. Takšne reklamni panoji omogočajo tudi povezovanje z medmrežjem. V drugih pogledih pa ne prinašajo nobenega napredka pred statičnimi oglasnimi tablami, razen tega, da se lahko vsebina hitreje in lažje menjuje.

S pomočjo združitve vizualnega sistema in tehnologije prepoznavanja obrazov pa lahko dodamo takšnim oglasnim panojem "inteligenco". Tako imenovani "pametni" oglaševalni panoji bodo lahko zamenjali vsebino glede na to, kdo jih opazuje. Sistem bo na podlagi klasifikacije in analize obraza ugotovil, ali pred panojem stoji moški ali ženska, mladoletnik ali starejša oseba, ali oseba nosi očala, ali ima brado, itd. Pri tako zasnovanem sistemu tudi zasebnost ne predstavlja problema. Vsi zajeti podatki osebo bodo namreč služili samo za namene prilagajanja vsebine oglasa in se bodo po obdelavi tudi zbrisali. V okoljih kjer pa zasebnost ne predstavlja problema (domače okolje, skrbno varovano območje, ustrezno označeno in prijavljeno območje) pa bo imel sistem tudi možnost pomnjenja in si bo osebo lahko zapomnil. Naslednjič, ko se bo oseba vrnila, bo poznal vse njene lastnosti, ki jih je že spoznal. Na podlagi prepoznavanja mimike obraza pa bomo lahko dobili tudi zelo pomembno povratno informacijo.

S pomočjo takšnih sistemov bodo tudi oglasni panoji postali bolj zanimivi, tako za ponudnike kot tudi za potrošnike.

10.4 Povzetek poglavja

V poglavju smo naredili pregled nad sistemi za prepoznavanje obrazov v različnih kategorijah uporabe. Za vsako kategorijo smo izpostavili omejitve sistemov, ki se trenutno uporabljajo. Lahko vidimo, da so se metode za prepoznavo obrazov iz akademskih raziskovalnih laboratorijev že preselile v komercialne sisteme. Najuspešnejše aplikacije so še vedno tiste, ki uporabljajo majhno do srednje veliko podatkovno bazo oseb. Takšni sistemi se največkrat uporabljajo za kontrolo dostopa ali vstopa v računalniški sistem. Sistemi za kontrolo na letališčih ali za nadzor na javnih prostorih pa še vedno ostajajo izziv za raziskovalce. Takšni sistemi so pri velikem številu oseb, ki jih morajo pregledati, ter zaradi zelo različnih pogojev zajemanja, še vedno nezanesljivi. S tesnejšim sodelovanjem med industrijskimi in univerzitetnimi raziskovalci lahko tudi takšni sistemi postanejo bolj zanesljivi in robustni.

Druga možnost izboljšave učinkovitosti prepoznave pa leži v kombinaciji različnih biometričnih in varnostnih metod. Tako lahko v kombinacijah s sistemi za prepoznavo obrazov uporabimo tudi prepoznavo zvoka, prstnih odtisov ali očesne šarenice. Kot varnostne metode pa lahko uporabimo rentgensko kontrolo, detektor kovin ali detektor kemičnih snovi.

V prihodnosti lahko pričakujemo, da se bo področje uporabe sistemov za prepoznavo obrazov razširilo še na področja, kot so avtomobilska industrija, industrija zabave, domači sistemi za zaščito računalniških sistemov, alarmni sistemi, področja internetne zaščite, itd. Predlagamo tudi novo področje uporabe za namene pametnega oglaševanja. Vsekakor pa bo prisotnih vse več sistemov za nadzor in kontrolo pri vstopih v zavarovana območja, za kontrolo vstopnic, za dodatno kontrolo na bankomatih in v potniškem prometu.

Poglavje 11

Zaključek in nadaljnje delo

11.1 Zaključek

V doktorski disertaciji smo predstavili celoten pregled metod za iskanje in prepoznavanje človeških obrazov na slikah. Podrobneje smo predstavili dve najuspešnejši metodi za iskanje obrazov na slikah, metodo temelječo na algoritmu AdaBoost in Bayesovo metodo diskriminantnih značilk – BDF. Metodi omogočata, da v realnem času na vhodni sliki najdemo vse frontalne obraze. Na koncu navedemo primerjavo najuspešnejših metod za iskanje, ki so bile preizkušene na skupni podatkovni bazi obrazov.

Za preizkus metod za iskanje in prepoznavanje obrazov so zelo pomembne podatkovne baze. V disertaciji navedemo vse pomembne podatkovne baze, ki se uporabljajo za samo iskanje, pa tudi za prepoznavanje obrazov. Podrobneje opišemo podatkovni bazi FERET in AR, ki ju tudi uporabimo v naših preizkusih. Podrobno si ogledamo tudi metodologijo preizkusov in predstavimo najpomembnejše rezultate standardnih preizkusov FERET IN FRVT.

Na uspešnost metod za prepoznavanje obrazov vpliva tudi normalizacijski postopek. Podrobneje predstavimo posamezne korake postopka in s pomočjo preizkusov preverimo, kako pomembna sta tako geometrijski kot svetlobni normalizacijski postopek. Pri normalizaciji preizkusimo tudi uspešnost postopka prepoznave za različne dele obraza in na podlagi tega ugotovimo, kateri del obraza je najpomembnejši.

V disertaciji podrobno predstavimo štiri glavne metode za prepoznavanje obrazov. Prve tri (PCA, LDA in ICA) temeljijo na celotnemu videzu obraza, zadnja (Gaborjeva metoda) pa temelji na posameznih značilkah obraza. Pri vsaki metodi opišemo splošno matematično definicijo, enostaven primer uporabe in primer uporabe na slikah obrazov za namen prepoznavanja. Pri metodi PCA ločimo še metodo PCAwo3, pri kateri izpustimo prve tri glavne komponente, pri metodi LDA predstavimo poleg običajne Fisherjeve metode, ki uporablja metodo PCA kot predkorak, tudi neposredno metodo LDA – direktna metoda LDA, pri metodi ICA pa ločimo dve arhitekturi. Iz metode ICA izpeljemo še dve metodi, ki se uporabljata kot pred-

priprava podatkov pred samim ločevanjem neodvisnih komponent – metodo PCA1 in metodo PCA2. Na takšen način dobimo 9 različnih metod za prepoznavanje obrazov.

Za postopek primerjave smo uporabili metodo najbližjih sosedov, ki izbere tisti vzorec, ki se najbolj ujema s testnim (razdalja med njima je najmanjša). Za merjenje razdalje med vzorci, ki jih predstavimo z vektorji značilk lahko uporabimo različne mere. Mi smo jih predstavili 14, ki se najpogosteje uporabljajo na področju razpoznavanja vzorcev.

Vse predstavljene metode in razdalje nato v vseh možnih kombinacijah preizkusimo na dveh podatkovnih bazah obrazov. Pri preizkusu uporabimo točno določen postopek, ki je skladen s postopkom preizkusa FERET. Uporabljen je enotni postopek normalizacije, enako število glavnih komponent in enak postopek primerjave. S tako skrbno zastavljenim preizkusom smo prišli do različnih ugotovitev. Na podlagi različnih testnih množic smo ugotovili za katere naloge metode delujejo najbolje, ter katere naloge še vedno povzročajo probleme, katera mera razdalje je najboljša za posamezno metodo, katera metoda velja na splošno za najuspešnejšo in katera mera razdalje je najuspešnejša. Pri preizkusih smo izvedli identifikacijo na zaprti množici in sicer pri najboljšem ujemanju (rang 1) in pri višjih rangih. Najboljše ujemanje določa uspešnost, če je rezultat iskana oseba, pri različnih rangih pa preverjamo če je iskana oseba znotraj določenega območja oziroma ranga. Drugi rezultat uspešnosti prikažemo s pomočjo krivulje CMS, ki določa uspešnost pri določenemu rangu. Izkaže se, da metoda, ki je uspešna pri rangu 1 ni nujno uspešna tudi pri višjih rangih. Iz preizkusa smo ugotovili, katere metode in razdalje so najuspešnejše pri višjih rangih.

Na podlagi vseh ugotovitev preizkusov predstavimo dva hibridna sistema. S pomočjo prvega vzporednega sistema izdelamo različne konfiguracije in ugotovimo, da deluje sistem najbolje v kombinaciji z dvema metodama. Za najboljši se izkažeta kombinacija obeh različic LDA in kombinacija metode PCA2 in Gaborjeve metode. Ugotovimo, da lahko s pravo kombinacijo izdelamo vzporedni hibridni sistem, ki izboljša uspešnost posameznih metod, ki so del sistema. Na podlagi rezultatov uspešnosti za rang 1 in pri višjih rangih izdelamo tudi zaporedni hibridni sistem. Medtem, ko je uspešnost tako zasnovanega zaporednega hibridnega sistema primerljiva z najuspešnejšo metoda v sistemu, deluje hibridi sistem enkrat hitreje.

Na koncu naredimo pregled najpomembnejših področij uporabe sistemov za prepoznavanje obrazov. Znotraj posameznih področij predstavimo težave in primere delujočih sistemov, ki uporabljajo poleg obraza največkrat še druge biometrične podatke. Kot poseben primer uporabe predstavimo tudi področje oglaševanja, kjer predstavimo naš predlagan modularni sistem za kategorizacijo, analizo in prepoznavo človeških obrazov.

11.2 Prispevki k znanosti

Glavne prispevke k znanosti lahko predstavimo v naslednjih točkah:

1. **Predstavitev sistema za prepoznavanje, klasifikacijo in analizo obrazov.** Splošni sistem za prepoznavanje smo razširili z metodami za klasifikacijo in analizo človeškega obraza. Predstavili smo sistem, ki je zasnovan modularno in omogoča poleg iskanja in prepoznavanja tudi kategorizacijo in analizo človeških obrazov. Tako zasnovan sistem lahko uporabimo na novih področjih, kot je tudi interakcija sistema s človekom.

2. **Implementacija in primerjava različnih metod za prepoznavanje obrazov.** Na podlagi pregleda vseh metod za prepoznavanje obrazov smo izbrali osem najpomembnejših metod, ki temeljijo na celotni sliki obraza ter metodo Gaborjevih valčkov, ki temelji na posameznih značilkah obraza. Preizkusili smo jih na dveh podatkovnih bazah z enotnim postopkom normalizacije in primerjave. Na podlagi preizkusa smo ugotovili, katere kombinacije tehnik izločanja značilk in tehnik primerjave so najuspešnejše, ter katera metoda za prepoznavanje obrazov deluje najbolje nad obrazi z določenimi lastnostmi.

3. **Preizkus različnih razdalj za merjenje oddaljenosti med vektorji značilk.** Za primerjavo vektorjev značilk smo uporabili metodo najbližjega soseda pri kateri lahko uporabimo različne razdalje za merjenje podobnosti med vzorci. Vse izbrane metode smo preizkusili v kombinaciji s 14 različnimi merami za razdalje. Preizkusili smo uspešnost najboljšega ujemanja (rang 1) in uspešnost pri višjih rangih.

4. **Ugotavljanje vpliva postopka normalizacije na uspešnost prepoznave obrazov.** Preizkusili smo različne normalizacijske postopke in predlagali najboljši postopek za različne metode in naloge. Preizkusili smo tudi uspešnost na posameznih delih obraza in ugotovili, kateri del obraza je za prepoznavanje najpomembnejši.

5. **Izdelava hibridnega sistema za prepoznavanje obrazov.** Sistemi, ki dandanes obetajo najboljšo uspešnost prepoznavanja obrazov so ravno hibridni sistemi, ki združujejo več različnih metod, s čemer lahko odpravijo pomanjkljivosti posameznih metod. Na podlagi rezultatov uspešnosti za posamezne metode smo predlagali najustreznejše metode za hibridni sistem. Izdelali smo tako vzporedni kot zaporedni hibridni sistem. Medtem, ko vzporedni sistem odpravlja določene slabosti posamezne metode in s tem dosega boljše rezultate, pa lahko z zaporednim hibridnim sistemom močno zmanjšamo čas prepoznavanja. Ti dve ugotovitvi smo potrdili s preizkusom na podatkovni bazi.

6. **Inovativna uporaba sistema za prepoznavanje, kategorizacijo in analizo na področju oglaševanja.** Predstavili smo inovativno idejo uporabe sistema za kategorizacijo, analizo in prepoznavo obrazov na področju oglaševanja.

Oglaševalski panoji opremljeni s takšnim sistemom bodo lahko svojo vsebino prilagodili gledalcu, potrošniku in mu tako prikazali to, kar ga zanima.

11.3 Nadaljnje delo

Pri preizkusih metod smo upoštevali preizkusni postopek FERET, ki je bil predstavljen in uporabljen za primerjavo različnih metod za prepoznavanje obrazov na standardnem preizkusu FERET. Postopek FERET s svojimi različnimi testnimi množicami omogoča, da preizkusimo različne lastnosti metod za prepoznavanje. Glavno orodje, ki ga postopek FERET uporablja za primerjavo uspešnosti različnih metod so krivulje CMS, ki prikazujejo uspešnost pri različnih rangih. Za najboljše ujemanje pri rangu 1 pa rezultate uspešnosti različnih metod primerjamo numerično v obliki tabel ali s pomočjo grafikonov. Za pravo primerjavo rezultatov uspešnosti pa bi potrebovali naprednejšo statistično orodje, s čemer bi preverili ali je razlika med rezultati različnih metod tudi statistično pomembna. Uporabili bi lahko splošen z-test ali McNemar-jev test, ki je posebej namenjen za podatke v paru, ki jih dobimo tudi v naših primerih, ko primerjamo metodi na isti testni množici.

Pri preizkusu metod smo predstavili najpomembnejše metode na podlagi videza: PCA, LDA in ICA. Poleg teh metod obstajajo tudi njihove nelinearne različice. Zanimivo bi bilo preizkusiti tudi te in jih primerjati z ostalimi. Za primerjavo izločenih značilk smo izbrali metodo najbližjih sosedov, ki se najpogosteje uporablja v kombinaciji s preizkušenimi metodami. Nekateri raziskovalci pa uporabljajo na stopnji primerjave naprednejše metode, kot so metoda podpornih vektorjev, nevronske mreže, itd. S pomočjo različnih kombinacij metod za izločanje značilk in metod za primerjavo bi lahko zgradili različne hibridne metode in jih preizkusili, če doprinesejo k izboljšanju.

Glavna pozornost biometričnih sistemov je namenjena združevanju več modalnosti med seboj. V disertaciji smo preizkusili hibridna sistema, ki smo ju zgradili tako, da smo združili med seboj različne metode. Dodatno pa bi bilo smiselno zgraditi hibridni sistem, ki bi združeval različne biometrične podatke in ne samo sliko obraza. Že združevanje slik zajetih z navadno kamero in slik zajetih z infrardečo kamero bi nam omogočilo, da bi zgradili hibridni sistem, ki bi bil manj občutljiv na osvetlitev.

Zahvala

Za vso pomoč pri izdelavi doktorske disertacije in ideje pri raziskavah se zahvaljujem mentorju prof. dr. Francu Solini. Zahvaljujem se mu tudi za razumevanje, zaupanje in podporo pri mojem delu, ki sem ga opravljal kot mladi raziskovalec v Laboratoriju za računalniški vid na Fakulteti za računalništvo in informatiko v Ljubljani.

Hvala tudi vsem sodelavcem, ki so mi delali družbo na delovnem mestu. Še posebej dr. Katarini Mele, mag. Mihu Peternelu in doc. dr. Petru Peeru. Hvala tudi članom komisije doc. dr. Alešu Jakliču, prof. dr. Petru Kokolu in prof. dr. Francu Solini za trud in čas, ki so ga vložili v pregledovanje moje disertacije.

Rad bi se zahvalil tudi staršem za vso podporo v času šolanja in študija. Posebna zahvala pa gre ženi Tadeji, ki me je v času pisanja stala ob strani, čeprav sem veliko časa preživel na delovnem mestu in seveda hčerki Emi, ki me je dan za dnem presenečala s svojim napredovanjem v razvoju, tudi na področju prepoznavanja obrazov.

Izjava

Izjavljam, da sem doktorsko disertacijo izdelal samostojno pod mentorstvom prof. dr. Franca Solina. Izkazano pomoč drugih sodelavcev sem v celoti navedel v zahvali.

Literatura

[1] B. Achermann in H. Bunke. Combination of classifiers on the decision level for face recognition. Tech. rep. IAM-96-002, Institute of Computer Science and Applied Mathematics, University of Bern, Bern, Switzerland, 1996.

[2] M. S. Bartlett, J. R. Movellan, in T. J. Sejnowski. Face recognition by independent component analysis. *IEEE Transactions on Neural Networks*, 13(6):1450–1464, 2002.

[3] Borut Batagelj. *Iskanje obrazov na osnovi barv s pomočjo statističnih metod razpoznavanja vzorcev*. Magistrsko delo, Fakulteta za računalništvo in informatiko Univeza v Ljubljani, 2004.

[4] Peter N. Belhumeur, Joao Hespanha, in David J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):711–720, 1997.

[5] Anthony J. Bell in Terrence J. Sejnowski. An information-maximization approach to blind separation and blind deconvolution. *Neural Computation*, 7(6):1129–1159, 1995.

[6] Anthony J. Bell in Terrence J. Sejnowski. The independent components of natural scenes are edge filters. *Vision Research*, 37(23):3327–3338, 1997.

[7] D.M. Blackburn, J.M. Bone, in P.J. Phillips. Facial recognition vendor test 2000: Evaluation report. Tehnično poročilo, DARPA, 2000.

[8] Volker Blanz in Thomas Vetter. A morphable model for the synthesis of 3D faces. V *Siggraph 1999, Computer Graphics Proceedings*, strani 187–194, 1999.

[9] D.S. Bolme, J. R. Beveridge, M. Teixeira, in B.A. Draper. The CSU face identification evaluation system: Its purpose, features, and structure. V *International Conference on Vision Systems, ICVS*, strani 304–313, 2003.

[10] J. Buhmann, M. Lades, in C. von der Malsburg. Size and distortion invariant object recognition by hierarchical graph matching. V *Proceedings of the IJCNN International Joint Conference on Neural Networks*, strani 411–416, 1990.

[11] C. Bunney. Survey: face recognition system. *Biometric Technology Today*, strani 8–12, 1997.

[12] J. F. Cardoso in A. Souloumiac. Blind beamforming for non gaussian signals. V *IEE Proceedings-F*, letnik 140, strani 362–370, december 1993.

[13] R. Chellappa, C. Wilson, in S. Sirohey. Human and machine recognition of faces: a survey. *Proceedings of the IEEE*, 83(5):704–740, 1995.

[14] Li-Fen Chen, Hong-Yuan Mark Liao, Ming-Tat Ko, Ja-Chen Lin, in Gwo-Jong Yu. A new lda-based face recognition system which can solve the small sample size problem. *Pattern Recognition*, 33(10):1713–1726, 2000.

[15] John Daugman. Combining multiple biometrics. *The Computer Laboratory, Cambridge University*, 2004.

[16] A. P. Dempster, N. M. Laird, in D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society series B*, 39:1–38, 1977.

[17] Bruce A. Draper, Kyungim Baek, Marian Stewart Bartlett, in J. Ross Beveridge. Recognizing faces with pca and ica. *Comput. Vis. Image Underst.*, 91(1-2):115–137, 2003.

[18] Richard O. Duda, Peter E. Hart, in David G. Stork. *Pattern Classification (2nd Edition)*. Wiley-Interscience, 2000.

[19] S. Eickeler in S. Birlinghoven. Face database retrieval using pseudo 2d markov models. V *Proceeding of IEEE Int. Conf. On Face and Gestures (FG 2002)*, 2002.

[20] P. Ekman in W. Friesen. The facial action coding system: A technique for the measurement of facial movement. *Consulting Psychologists Press*, 1978.

[21] FaceGate. http://www.premierelect.co.uk/faceaccess.html, april 2006.

[22] FaceKey. http://www.facekey.com, april 2006.

[23] R. Frischholz. Face detection homepage - datasets. http://www.facedetection.com/, marec 2006.

[24] Keinosuke Fukunaga. *Introduction to statistical pattern recognition*. Boston: Academic Press, second edition izdaja, 1990.

[25] Cognitec Systems GmbH. http://www.cognitec-systems.de, april 2006.

[26] G. G. Gordon. Face recognition from frontal and profile views. V M. Bichsel, urednik, *Proceedings of the International Workshop on Automatic Face- and GestureRecognition, IWAFGR*, strani 47–52, 1995.

[27] M. Grcic in K. Delac. Face recognition homepage - databases. http://www.face-rec.org/databases/, marec 2006.

[28] R. Gross. Face databases. V S.Z. Li in A.K. Jain, uredniki, *Handbook of Face Recognition*, poglavje 13, strani 301–327. Springer, 2004.

[29] L. Gu, S. Li, in H. Zhang. Learning probabilistic distribution model for multi-view face dectection. V *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2001.

[30] S. Gutta, J. Huang, I. Imam, in H. Weschler. Face and hand gesture recognition using hybrid classifiers. V *Proceedings of the Second International Conference on Automatic Face and Gesture Recognition, FG '96*, strani 164–169, Washington, DC, USA, 1996. IEEE Computer Society.

[31] S. Gutta in H. Wechsler. Face recognition using hybrid classifier systems. V *Proceedings of the IEEE International Conference on Neural Networks*, strani 1017–1022, 1996.

[32] R. Hietmeyer. Biometric identification promises fast and secure processing of airline passengers. *The international Civil Aviation Organization Journal*, 55(9):10–11, 2000.

[33] E. Hjelmås in B-K Low. Face detection: A survey. *Computer Vision and Image Understanding*, 83(3):236–274, 2001.

[34] H. Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24(6):417–441, 1933.

[35] R. Houghton. Named faces: putting names to faces. *IEEE Intelligence Systems*, 14(5):45–50, 1999.

[36] J. Huang, B. Heisele, in V. Blanz. Component-based face recognition with 3d morphable models. V *4th Conference on Audio- and Video-Based Biometric Person Authetication, AVBPA*, 2003.

[37] T. Huang, Z. Xiong, in Z. Zhang. Face recognition applications. V S.Z. Li in A.K. Jain, uredniki, *Handbook of Face Recognition*, poglavje 16, strani 371–390. Springer, 2004.

[38] A. Hyvärinen in E. Oja. A fast fixed-point algorithm for independent component analysis. *Neural Computation*, 9:1483–1492, 1997.

[39] Visionics Identix. http://www.identix.com, april 2006.

[40] Imagis Technologies Inc. http://www.imagistechnologies.com, april 2006.

[41] Eyematic Interfaces Inc. http://www.eyematic.com, april 2006.

[42] A.N. Venetsanopoulos J. Lu, K.N. Plataniotis. Face recognition using lda-based algorithms. *IEEE Transactions on Neural Networks*, 14(1):195–200, 2003.

[43] A.K. Jain, A. Ross, in S. Prabhakar. An introduction to biometric recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(1):4–20, 2004.

[44] Zhong Jin in F. Davoine. Orthogonal ica representation of images. V *Control, Automation, Robotics and Vision Conference, ICARCV 2004 8th*, strani 369–374, december 2004.

[45] R.A. Johnston in H.D. Ellis. The development of face recognition. *Cognitive and Computational Aspects of Face Processing*, strani 1–23, 1995.

[46] Michael Jones in Paul Viola. Fast multi-view face detection. V *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2003.

[47] Takeo Kanade. *Picture Processing System by Computer Complex and Recognition of Human Faces*. PhD thesis, Kyoto University, november 1973.

[48] Michael David Kelly. *Visual identification of people by computer*. PhD thesis, Stanford University, 1971.

[49] M. Kirby in L. Sirovich. Low-dimensional procedure for the characterization of human faces. *Journal of the Optical Society of America A*, 4(3):519–524, 1987.

[50] M. Kirby in L. Sirovich. Application of the karhunen-lo'eve procedure for characterization of human faces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(1):103–108, 1990.

[51] J. Kittler, M. Hatef, R.P.W. Duin, in J. Matas. On combining classifiers. *IEEE Trans. Pattern Analysis and Machine Intelligence, PAMI*, 20(3):226–239, 1998.

[52] W. Konen in E. Schulze-Kruger. Zn-face: a system for access control using automated face recognition. V *Poceedings of the International Workshop on Automatic Face and Gesture Recognition*, strani 18–23, 1995.

[53] A. Kudhinsky, C. Pering, M.L. Creech, D. Freeze, B. Serra, in J. Gvvizdka. Fotofile: a consumer multimedia organization and retrieval system. V *Proceedings of CHI 1999*, strani 496–503, 1999.

[54] Fernando De la Torre in Michael Black. Robust principal component analysis for computer vision. V *IEEE Int. Conference on Computer Vision (ICCV)*, letnik I, strani 362–369, 2001.

[55] M. Lades, J. C. Vorbrüggen, J. Buhmann, J. Lange, C. von der Malsburg, R. P. Würtz, in W. Konen. Distortion invariant object recognition in the dynamic link architecture. *IEEE Transactions on Computers*, 42(3):300–311, 1993.

[56] A. Lanitis, C.J. Taylor, in T.F. Cootes. Automatic face identification system using flexible appearance models. *Image and Vision Computing, IVC*, 13(5):393–401, junij 1995.

[57] T.-W. Lee, T. Wachtler, in T. J. Sejnowski. Color opponency is an efficient representation of spectral properties in natural scenes. *Vision Research*, 42:2095–2103, 2002.

[58] C. Liu. A bayesian discriminating features method for face detection. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 25(6):725–740, 2003.

[59] Juwei Lu, K. N. Plataniotis, in A. N. Venetsanopoulos. Regularization studies of linear discriminant analysis in small sample size scenarios with application to face recognition. *Pattern Recognition Letter*, 26(2):181–191, 2005.

[60] Xiaoguang Lu, Yunhong Wang, in Anil K. Jain. Combining classifiers for face recognition. V *IEEE International Conference on Multimedia and Expo (ICME'03)*, letnik 3, strani 13–16, 2003.

[61] M.R. Lyu, E. Yau, in S. Sze. A multilingual, multimodal digital video library system. V *ACM/IEEE Joint Conf. On Digital Libraries, (JCDL 2002)*, strani 145–153, 1998.

[62] W.-Y. Ma in H. Zhang. An indexing and browsing system for home video. V *Proc. Of 10th European Signal Processing Conference*, 2000.

[63] B.S. Manjunath, R. Chellappa, in C. von der Malsburg. A feature based approach to face recognition. V *Computer Vision and Pattern Recognition, CVPR*, strani 372–378, 1992.

[64] G. L. Marcialis in F. Roli. Fusion of LDA and PCA for face recognition. V *Proceedings of the Workshop on Machine Vision and Perception, 8th Workshop of the Italian Association for Artificial Intelligence (AIIA '02)*, 2002.

[65] A.R. Martínez in R. Benavente. The AR face database. Technical Report 24, Computer Vision Center (CVC), Barcelona, 1998.

[66] Maximus. http://www.maximus.com/corporate/pages/, april 2006.

[67] B. Moghaddam, C. Nastar, in A. Pentland. Bayesian face recognition using deformable intensity surfaces. V *IEEE Conference on Computer Vision and Pattern Recognition*, strani 638–645, 1996.

[68] B. Moghaddam in A. Pentland. Face recognition using view-based and modular eigenspaces. V *Automatic Systems for the Identification and Inspection of Humans, SPIE'94*, letnik 2257, 1994.

[69] B. Moghaddam in A. Pentland. Probabilistic visual learning for object representation. *Pattern Analysis and Machine Intelligence, PAMI*, 19(7):696–710, 1997.

[70] Baback Moghaddam. Principal manifolds and probabilistic subspaces for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(6):780–788, 2002.

[71] Y. Moses, Y. Adini, in S. Ullman. Face recognition: The problem of compensating for changes in illumination direction. *Pattern Analysis and Machine Intelligence, PAMI*, 19(7):721–732, 1997.

[72] P. Navarrete in J. del Solar. Interactive face retrieval using self-organizing maps. V *Proceedings, 2002 Int. Joint Conf. On Neural Networks, (IJCNN 2002)*, 2002.

[73] Massachusetts Institute of Technology. Face databases. http://web.mit.edu/emeyers/www/face_databases.html, marec 2006.

[74] K. Okada, J. Steffens, T. Maurer, H. Hong, E. Elagin, H. Neven, in C. von der Malsburg. The Bochum/USC Face Recognition System And How it Fared in the FERET Phase III test. V H. Wechsler, P. J. Phillips, V. Bruce, F. Fogeman Soulié, in T. S. Huang, uredniki, *Face Recognition: From Theory to Applications*, strani 186–205. Springer-Verlag, 1998.

[75] E. Osuna, R. Freund, in F. Girosi. Training support vector machines: An application to face detection. V *Proceedings of the Conference on Computer Vision and Pattern Recognition, (CVPR'97)*, strani 130–136, 1997.

[76] P. S. Penev in J. J. Atick. Local feature analysis: A general statistical theory for object representation. *Network: Computation in Neural Systems*, 7(3):477–500, 1996.

[77] P. J. Phillips, H. Moon, P. J. Rauss, in S. Rizvi. The feret evaluation methodology for face recognition algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(10):1090–1104, oktober 2000.

[78] P.J. Phillips, P. Grother, R.J. Micheals, D.M. Blackburn, E. Tabassi, in M. Bone. Face recognition vendor test 2002. Tehnično poročilo NIST IR 6965, NIST, 2003. AMFG03(44-44), AVBPA03(937-945).

[79] P.J. Phillips, H.Wechsler, J.Huang, in P.Rauss. The feret database and evaluation procedure for face-recognition algorithms. *Image and Vision Computing*, 16(5):295–306, 1998.

[80] S.A. Rizvi, P.J.Phillips, in H. Moon. A verification protocol and statistical performance analysis for face recognition algorithms. V *Proceedings of Conference on Computer Vision and Pattern Recognition CVPR'98*, strani 833–838, Santa Barbara, CA, USA, 1998.

[81] S. Roweis. EM algorithms for PCA and SPCA. V *Neural Information Processing Systems 10 (NIPS'97)*, strani 626–632, 1997.

[82] H. A. Rowley, S. Baluja, in T. Kanade. Rotation invariant neural network-based face detection. V *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, (CVPR'98)*, stran 38. IEEE Computer Society, 1998.

[83] Henry A. Rowley, Shumeet Baluja, in Takeo Kanade. Human face detection in visual scenes. V David S. Touretzky, Michael C. Mozer, in Michael E. Hasselmo, uredniki, *Advances in Neural Information Processing Systems*, letnik 8, strani 875–881. The MIT Press, 1996.

[84] Henry A. Rowley, Shumeet Baluja, in Takeo Kanade. Neural network-based face detection. V *Proceedings of the 1996 Conference on Computer Vision and Pattern Recognition (CVPR '96)*, stran 203. IEEE Computer Society, 1996.

[85] H. Schneiderman in T. Kanade. A statistical model for 3d object detection applied to faces and cars. V *IEEE Conference on Computer Vision and Pattern Recognition*, strani 1746–1752, 2000.

[86] Henry Schneiderman in Takeo Kanade. Probabilistic modeling of local appearance and spatial relationships for object recognition. V *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '98)*, strani 45–51, julij 1998.

[87] Andrea Selinger in Diego A. Socolinsky. Appearance-based facial recognition using visible and thermal imagery: A comparative study. Technical report 02-01, Equinox Corporation, 2002.

[88] P. Sinha, B.J. Balas, Y. Ostrovsky, in R. Russell. Face recognition by humans. In Face Recognition: Advanced Modeling and Methods, 2006.

[89] Diego A. Socolinsky, Lawrence B. Wolff, Joshua D. Neuheisel, in Christopher K. Eveland. Illumination invariant face recognition using thermal infrared imagery. V *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, strani 527–534, 2001.

[90] F. Solina, P. Peer, B. Batagelj, S. Juvan, in J. Kovač. Color-based face detection in the "15 seconds of fame" art installation. V *Conference on Computer Vision / Computer Graphics Collaboration for Model-based Imaging, Rendering, image Analysis and Graphical special Effects*, strani 38–47, 2003.

[91] Kah Kay Sung. *Learning and Example Selection for Object and Pattern Detection.* PhD thesis, Massachusetts Inst. of Technolog, 1996.

[92] Kah Kay Sung in Tomaso Poggio. Example-based learning for view-based human face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1):39–51, 1998.

[93] Daniel L. Swets in Juyang Weng. Using discriminant eigenfeatures for image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(8):831–836, 1996.

[94] Sistemi TAB. http://www.tab-systems.com, april 2006.

[95] Viisage Technology. http://www.viisage.com, april 2006.

[96] P. Thompson. Margaret thatcher: a new illusion. *Perception*, 9(4):483–484, 1980.

[97] M. E. Tipping in C. M. Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society, Series B*, 61(3):611–622, 1999.

[98] M. Turk in A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

[99] Washington University. Archive of nonface images. http://wuarchive.wustl.edu/pub/aminet/pix/, marec 2006.

[100] Paul Viola in Michael Jones. Rapid object detection using a boosted cascade of simple features. V *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'01)*, letnik 1, strani 511–518, 2001.

[101] Paul Viola in Michael Jones. Robust real-time object detection. V *IEEE ICCV Workshop on Statistical and Computational Theories of Vision*, Vancouver, 2001.

[102] Paul Viola in Michael Jones. Robust real-time object detection. *International Journal of Computer Vision*, 57(2):137–154, 2004.

[103] H. Wactlar, T.K.M. Smith, in S. Stevens. Intelligence access to digital video: informedia project. *IEEE Computer*, 29(5):46–52, 1996.

[104] Frank Wallhoff, Stefan Müller, in Gerhard Rigoll. Hybrid Face Recognition Systems for Profile Views Using The MUGSHOT Database. V *Workshop RAT-FG in conjunction with the IEEE Int. Conference on Computer Vision (ICCV)*, Vancouver, Canada, 2001.

[105] Frank Wallhoff in Gerhard Rigoll. A Novel Hybrid Face Profile Recognition System Using The FERET And MUGSHOT Databases. V *IEEE Int. Conference on Image Processing (ICIP)*, Thessaloniki, Greece, 2001.

[106] H. Wang in S.-F. Chang. A highly efficient system for automatic face region detection in mpg video sequence. *IEEE tras. On Circuits and Systems for Video Technology, Special Issue on Multimedia Systems and Technologies*, 7(4):615–628, 1997.

[107] Laurenz Wiskott, Jean-Marc Fellous, Norbert Krüger, in Christoph von der Malsburg. Face recognition by elastic bunch graph matching. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(7):775–779, 1997.

[108] Laurenz Wiskott, Jean-Marc Fellous, Norbert Krüger, in Christoph von der Malsburg. Face recognition by elastic bunch graph matching. V Gerald Sommer, Kostas Daniilidis, in Josef Pauli, uredniki, *Proceedings 7th International Conference on Computer Analysis of Images and Patterns, CAIP'97, Kiel*, strani 456–463, Heidelberg, 1997. Springer-Verlag.

[109] Z. Xiong, Y. Chen, R. Wang, in T. Huang. Improved information maximization based face and facial feature detection from real-time video and application in a multi-modal person identification system. V *Proceedings of Fourth International Conference on Multimodal Interfaces (ICMI 2002)*, strani 511–516, 2002.

[110] Ming-Hsuan Yang, David Kriegman, in Narendra Ahuja. Detecting faces in images: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 24(1):34–58, 2002.

[111] Hua Yu in Jie Yang. A direct LDA algorithm for high-dimensional data - with application to face recognition. *Pattern Recognition*, 34(10):2067–2070, 2001.

[112] W. Zhao, R. Chellappa, in A. Krishnaswamy. Discriminant analysis of principal components for face recognition. V *3rd International Conference on Automatic Face and Gesture Recognition*, strani 336–341, 1998.

[113] W. Zhao, R. Chellappa, in P. Phillips. Subspace linear discriminant analysis for face recognition. Tehnično poročilo CAR-TR-914, Center for Automation Research, University of Maryland, College Park, 1999.

[114] W. Zhao, A. Krishnaswamy, R. Chellappa, D. Swets, in J. Weng. Discriminant analysis of principal components for face recognition. V H. Wechsler, P.J. Phillips, V. Bruce, F.F. Soulie, in T.S. Huang, uredniki, *Face Recognition: From Theory to Applications*, strani 73–85. Springer-Verlag, 1998.

[115] W. Zhao, R.Chellappa, A. Rosenfelda, in J. Phillips. Face recognition: a literature survey. Technical report, CS-TR4167R, University of Maryland, 2000.