

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Rok Petek

**Optična razpoznavna znakov v slikah
naravnih scen**

DIPLOMSKO DELO

VISOKOŠOLSKI STROKOVNI ŠTUDIJSKI PROGRAM PRVE
STOPNJE RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: doc. dr. Peter Peer

ASISTENT: mag. Andrej Ikica

Ljubljana, 2012

Rezultati diplomskega dela so intelektualna lastnina avtorja in Fakultete za računalništvo in informatiko Univerze v Ljubljani. Za objavljanje ali izkoriščanje rezultatov diplomskega dela je potrebno pisno soglasje avtorja, Fakultete za računalništvo in informatiko ter mentorja.

Besedilo je oblikovano z urejevalnikom besedil \LaTeX .



Št. naloge: 00228/2012

Datum: 02.04.2012

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Kandidat: **ROK PETEK**

Naslov: **OPTIČNA RAZPOZNAVA ZNAKOV V SLIKAH NARAVNIH SCEN**
OPTICAL CHARACTER RECOGNITION IN IMAGES OF NATURAL SCENES

Vrsta naloge: Diplomsko delo visokošolskega strokovnega študija prve stopnje

Tematika naloge:

Naredite pregled področja optične razpoznavne znakov v slikah naravnih scen. Implementirajte metodo na osnovi smernih segmentov nelinearne mreže. Predlagano metodo dodatno nadgradite z ostalimi popularnimi značilkami in klasifikatorji. Metodo nato testirajte ter rezultate primerjajte z rezultati drugih metod.

Mentor:


doc. dr. Peter Peer



Dekan:


prof. dr. Nikolaj Zimic

IZJAVA O AVTORSTVU DIPLOMSKEGA DELA

Spodaj podpisani Rok Petek, z vpisno številko **63090366**, sem avtor diplomskega dela z naslovom:

Razpoznava znakov v slikah naravnih scen

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal samostojno pod mentorstvom doc. dr. Petra Peera in mag. Andreja Ikice,
- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela
- soglašam z javno objavo elektronske oblike diplomskega dela v zbirki "Dela FRI".

V Ljubljani, dne 20. septembra 2012

Podpis avtorja:

Zahvalil bi se mentorju doc. dr. Petru Peeru in predvsem asistentu mag. Andreju Ikici, za dobre nasvete in motivacijo pri delu.

Posebej bi se rad zahvalil tudi staršem ter vsem ostalim, ki so mi tekom študija stali ob strani.

Kazalo

Povzetek

Abstract

| | | |
|----------|---|-----------|
| 1 | Uvod | 1 |
| 2 | Optična razpoznavna znakov | 3 |
| 2.1 | Zgodovina klasične optične razpoznavne znakov | 3 |
| 2.2 | Optična razpoznavna znakov v slikah naravnih scen | 7 |
| 2.2.1 | Pregled področja | 9 |
| 3 | Uporabljena orodja | 12 |
| 3.1 | Matlab | 12 |
| 3.2 | Zunanje knjižnice | 12 |
| 3.2.1 | Statistični programski paket | 13 |
| 3.2.2 | Programski paket za obdelavo slik | 13 |
| 3.2.3 | Programski paket za delo z nevronskimi mrežami | 13 |
| 3.2.4 | Knjižnica za uporabo metode podpornih vektorjev | 14 |
| 3.2.5 | Knjižnica za pridobivanje vizualnih značilnk | 14 |
| 4 | Zbirke slik teksta naravnih scen | 15 |
| 4.1 | ICDAR | 15 |
| 4.2 | CVL OCR DB | 16 |

| | | |
|----------|---|-----------|
| 4.3 | Hibridna zbirka | 19 |
| 5 | Predlagana metoda | 20 |
| 5.1 | Opis metode | 20 |
| 5.2 | Predprocesiranje | 22 |
| 5.3 | Segmentacija | 25 |
| 5.3.1 | FCM | 25 |
| 5.3.2 | Binarizacija Otsu | 26 |
| 5.4 | Izbira značilnk | 27 |
| 5.4.1 | Značilke smernih segmentov z nelinearno mrežo | 27 |
| 5.4.2 | SIFT | 30 |
| 5.4.3 | Gaborjev filter | 32 |
| 5.5 | Klasifikacija | 34 |
| 5.5.1 | MDC | 34 |
| | MDC s prototipnim učenjem | 35 |
| | Prerazporeditev kandidatov za razponavo | 37 |
| 5.5.2 | K-NN | 38 |
| | Izbira parametrov | 39 |
| 5.5.3 | SVM | 40 |
| | RBF jedro | 41 |
| | Najboljše tehnike SVM klasifikacije | 41 |
| | Razširitev SVM klasifikatorja | 42 |
| 5.5.4 | ANN | 43 |
| | Parametri | 46 |
| 6 | Rezultati | 48 |
| 6.1 | Predprocesiranje | 48 |
| 6.2 | Segmentacija | 49 |
| 6.3 | Pridobivanje značilnk | 49 |
| 6.4 | Klasifikacija | 51 |
| 6.5 | Evalvacije celotnega sistema | 52 |

| | |
|--------------------------|-----------|
| 6.6 Ugotovitve | 56 |
| 7 Zaključek | 59 |
| Seznam slik | 60 |
| Seznam tabel | 63 |
| Literatura | 64 |

Seznam uporabljenih kratic in simbolov

ANN – Artificial Neural Network; umetna nevrnska mreža

CVL OCR DB – Computer Vision Laboratory OCR DataBase; OCR podatkovna baza Laboratorija za računalniški vid

FCM – Fuzzy C-means; segmentacijski algoritem

ICDAR – International Conference on Document Analysis and Recognition; mednarodna konferenca razpoznavne in analize dokumentov

K-NN – K-Nearest Neighbours; k-najbližjih sosedov

MDC – Minimum Distance Classifier; klasifikator najmanjših razdalj

OCR – Optical Character Recognition; optična razpoznavna znakov

ROC – Receiver Operating Characteristic; ROC krivulja ilustrira zmogljivost klasifikacijskega sistema

SIFT – Scale Invariant Feature Transform; skalirno invariantne transformacije značilnik

SVM – Support Vector Machine; metoda podpornih vektorjev

Povzetek

V diplomskem delu sta predstavljena opis in implementacija nekaterih sodobnih tehnik in metod optične razpoznavne znakov v slikah naravnih scen. Pri izbiri metod smo se osredotočili na hitrost in natančnost. Kot osnovo smo izbrali metodo smernih segmentov nelinearne mreže, saj je bila razvita za mobilno platformo in je tako ustrezala kriteriju hitrosti. Prav tako metoda med primerljivimi metodami dosega zelo dobre rezultate. Predlagano metodo smo dodatno nadgradili z nekaterimi ostalimi popularnimi značilnkami in klasifikatorji.

Optična razpoznavna teksta v slikah naravnih scen je izredno problematična, saj se v njih tekst pojavlja v različnih velikostih, barvah, pisavah in orientacijah. Prav tako so slike naravnih scen slabše kvalitete in vsebujejo kompleksna ozadja, kar oteži proces razpoznavne. Podobno kot pri klasični optični razpoznavi znakov so sistemi za optično razpoznavo znakov v slikah naravnih scen sestavljeni iz štirih korakov: predprocesiranje, segmentacija, izračun značilnk in klasifikacija. Faza predprocesiranja je namenjena izboljšavi kvalitete slike, v fazi segmentacije pa se v sliki izberejo le slikovni elementi, ki pripadajo posameznemu znaku. Oba koraka sta zaradi že omenjenih problemov naravnih scen izredno pomembna. V fazi izračuna značilnk se izračunajo karakteristike segmentiranega znaka, ki služijo nadaljnji klasifikaciji znaka v ustrezen razred.

Vse implementirane metode smo testirali na podatkovnih zbirkah slik ICDAR in CVL OCR DB ter na hibridni zbirki, ki smo jo generirali iz obeh podatkovnih zbirk.

Nadgrajena metoda predstavljena v diplomskem delu dosega dobre rezultate in je v povezavi z ustreznim detektorjem teksta v slikah naravnih scen primerna za migracijo in uporabo na mobilni platformi.

Ključne besede:

OCR, naravne scene, računalniški vid, smerni segmenti nelinerane mreže, SIFT, Gaborjev filter, SVM, ANN, K-NN, MDC

Abstract

The diploma thesis presents a description and implementation of some modern techniques and methods for optical character recognition in images of natural scenes. When choosing methods, we focused on speed and accuracy. As a basis we have chosen the method of directional segment features with nonlinear mesh since it was developed for the mobile platform and thus meets the criteria of speed. Also, the method comparable to other methods reaches very good results. The proposed method was further upgraded with some other popular features extraction methods and classifiers.

Optical recognition of text in images of natural scenes is very problematic, because in them the text appears in a variety of sizes, colors, fonts and orientations. Also pictures of natural scenes typically have lower quality and contain complex background, which greatly complicates the process of recognition. Similar to the classic optical character recognition the systems for optical character recognition in the images of natural scenes typically consist of four steps: preprocessing, segmentation, feature extraction and classification. Preprocessing phase is designed to improve image quality, in the segmentation stage only the pixels that belong to each character are chosen in the picture. Both steps are due to the aforementioned problems of natural scenes extremely important. During the feature extraction phase the characteristics of a segmented character are calculated, which are used for further classification of the character corresponding class.

All implemented methods were tested on image databases ICDAR, CVL

OCR DB, and a hybrid collection that we have generated from the two mentioned databases.

Improved method presented in the thesis has achieved good results and is, in conjunction with the relevant text detection in images of natural scenes, suitable for migration and the use on the mobile platform.

Keywords:

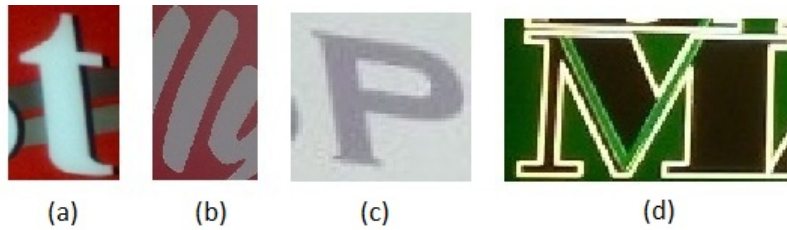
OCR, natural scenes, computer vision, directional segment features of non-linear mesh, SIFT, Gabor filter, SVM, ANN, K-NN, MDC

Poglavje 1

Uvod

Optična razpoznavna znakov (v nadaljevanju OCR) je zelo široko področje računalniškega vida, ki se aktivno razvija že od petdesetih let prejšnjega stoletja. Zgodnje verzije sistemov OCR so bile močno okrnjene in so omogočale le razpoznavo ene pisave, medtem ko današnji inteligentni sistemi omogočajo razpoznavo različnih vrst kompleksnih pisav z visoko stopnjo natančnosti. Optična razpoznavna znakov se deli na več podpodročij – od klasičnega OCR (razpoznavna črnega teksta na belem ozadju, tipično pri skeniranih dokumentih), optične razpoznavne pisanih črk, razpoznavne teksta v video posnetkih pa vse do razpoznavne teksta v slikah naravnih scen, ki se ji v diplomski nalogi posvečamo. Z izrazom tekst v slikah naravnih scen označujemo tekst poljubne velikost, stila, barve in orientacije, ki se pojavlja v raznovrstnih slikah vsakodnevnih scen s kompleksnimi ozadji, tipično zajetih s slabšo opremo za zajem (kompaktni fotoaparati, mobilni telefoni ipd.) (slika 1.1). OCR ima v naravnih scenah ogromen aplikativni potencial, saj ga je možno aplicirati v sisteme za pomoč slabovidnim, v preiskovalnike slikovnih vsebin ter nenazadanje na mobilno platformo, ki postaja zadnja leta platforma prihodnosti.

Kljub številnim komercialnim rešitvam in nenehnemu razvoju optična razpoznavna znakov v slikah naravnih scen še ne dosega optimalne natančnosti, zato področje pušča še ogromno prostora za nadaljnje raziskave in razvoj.



Slika 1.1: Primeri slik znakov (a) kompleksno ozadje, (b) kompleksen tekst, (c) nagnjen tekst, (d) različne barve teksta.

V okviru konference ICDAR (International Conference on Document Analysis and Recognition) poteka tekmovanje Robust Reading Competition [1], na katerem raziskovalci iz različnih raziskovanih ustanov s svojimi sistemi za razpoznavo tekmujejo na podatkovni zbirki slik teksta v naravnih scenah [1]. Cilj diplomske naloge je bil pregled področja optične razpoznave znakov v slikah naravnih scen, implementacija nekaterih primernih metod, njihova nadgradnja ter evalvacija na podatkovnih zbirkah teksta v slikah naravnih scen ICDAR [1] in CVL OCR DB [2]. Pri izboru metod smo se osredotočili na metode, ki so primerne za apliciranje na mobilno platformo in uporabo v praksi – torej metode, ki so ustrezno hitre in dovolj natančne.

Poglavje 2

Optična razpoznavna znakov

2.1 Zgodovina klasične optične razpoznavne znakov

Zgodovina OCR-ja sega vse nazaj v zgodnja 1950-ta leta. V tistem času so se znanstveniki trudili zajeti slike znakov in teksta tako z mehničnimi kot optičnimi sredstvi. Ta sredstva so bila vrteči diski in fotomultiplikatorji ter skener t.i flying spot scanner s katodno cevjo (temeljijo na sledenju obrisov), katerim so sledile foto-celice. Princip delovanja, ki je še danes v uporabi je ujemanje predlog/mask. Svetloba, ki pronica skozi mehansko masko je ujeta s foto-detektorjem in je tudi mehansko skenirana. Ko se ujemanje zgodi in svetloba ne pride do detektorja naprava razpozna znak, ki je natiskan. Na začetku je bilo skeniranje počasno, tako je bila lahko digitalizirana samo ena vrstica teksta v času premika skenerja ali papirnatega medija. Pozneje je napredek v digitalno-integriranih vezjih doprinesel k večji gostoti foto-celic in foto-seznamov, ki so omogočili večji prenos dokumentov ter višje hitrosti v skeniranju in digitalni pretvorbi. Skozi 60-ta in 70-ta leta se je optična razpoznavna znakov razširila v večino vej gospodarstva, kot so banke, podjetja, bolnice, pošta, časopisi ter druge [3].

Razpoznavo znakov razdelimo na tri generacije [4]. Prvi komercialni razpoznavi znakov, ki je trajala od 1960–1965 lahko rečemo prva generacija. Za to generacijo je bila značilna predvsem omejena oblika znakov za branje. Simboli so bili specifično prilagojeni mehanskemu branju in niso izgledali naravno, sčasoma so se pojavile naprave z večimi pisavami, ki so podpirali do 10 različnih pisav. Bralne naprave 2. generacije so se pojavile sredi 60–tih in na začetku 70–tih let. Te naprave so lahko razpoznale mehansko natisnjene znake kot tudi ročno napisane, a so bile zelo omejene. Vzporedno z napredki v strojni opremi za razpoznavo, so se intenzivne raziskave na področju OCR dogajale tudi na akademskem področju. Čeprav so bile tehnike razpoznave sredi 60–tih slabe, se je večina napak pojavila zaradi slabega tiska in variirajočih se pisav. Nove vrste pisav, kot je OCR–A in OCR–B so bile sprejete s strani ISO (International Standards Organization), da bi olajšale proces razpoznave znakov (slika 2.1). Rezultat teh dveh pisav je bila povečana natančnost razpoznave in njene hitrosti.

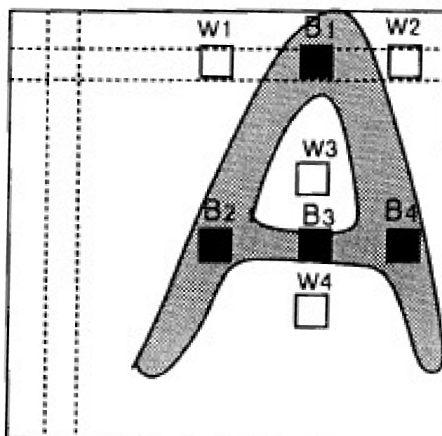


Slika 2.1: OCR-A (zgoraj), OCR-B (spodaj).

Za tretjo generacijo, ki se je pojavila v sredini 70–tih je bil izziv razpoznavna znakov predvsem v dokumentih slabe kvalitete in ročni pisavi. Enotni pedsledki v tisku in majhno število pisav, je naredilo preproste OCR naprave učinkovite za uporabo. V nadaljevanju podajamo opis nekaterih preprostih metod OCR, ki so se uporabljale na začetku.

Metoda kukala

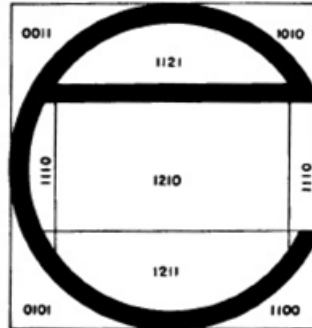
Slika 2.2 prikazuje metodo kukala [5], ki temelji na metodi ujemanja predlog, kjer je očitno, da so s konstantno velikostjo in debelino znaka, črni deli vedno črni in enako velja za belo ozadje. Nato so primerni slikovni elementi izbrani za črne in bele dele, tako da lahko izbrani slikovni elementi razlikujejo vhodni znak od ostalih.



Slika 2.2: Metoda kukala.

Analiza struktur

Poleg ujemanja predlog/mask se je pojavila tudi analiza struktur [5]. Slika 2.3 prikazuje primer analize ozadja, ki temelji na analizi struktur. Ideja analize ozadja je smatrana kot razširitev metode s sondo, kjer je vsak slikovni element ozadja, razpet po x in y osi. Bolj specifično to pomeni, da je vsak slikovni element ozadja predstavljen z 4. mestno kodo, ki je lahko binarna ali ternarna. Slika 2.3 prikazuje segmentacijo ozadja v 7 podregij. Štiriobovne regije imajo binarno kodo, ostale regije vsebujejo ternarno. Klasifikacija je tu opravljena z vektorjem značilk, kateremu zadošča 30 dimenzij.

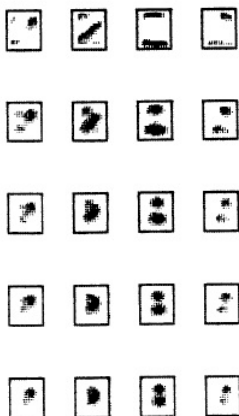


Slika 2.3: Analiza struktur.

4–smerna maska značilk

Metoda razpoznavne znakov, 4 smerna maska značilk [5], je temeljila na uje-manju značilk in je bila v tistem času smatrana za uspešno. Metoda je bila konstruirana s pridobitvijo smeri v vsakem slikovnemu elementu in vsakem vzorcu ter kvantizirana v štirih smereh, nato je bila na njej izvedena normalizacija in glajenje. Kot končni rezultat je bilo vzeto povprečje vseh procesiranih vzorcev. Slika 2.4 prikazuje značilke številke »2«, kjer prva vrstica predstavlja štiri komponente maske, kar ustreza štirim kvantiziranim smerem. Od druge do zadnje vrstice slika prikazuje zglajene in velikostno normalizirane maske, kjer se gladilni faktor povečuje z vsako vrstico. Ta metoda je prinašala razpoznavo z 97.1% natančnostjo, pri 8200 vzorcih z 41. razredi (41×200) [5].

Kjub intenzivnim raziskavam v zadnjih desetletjih, sposobnost razpoznavne znakov računalnikov še vedno zaostaja za človeško razpoznavo. Večina OCR sistemov še vedno ne more razpoznati degradiranih in lastnoročno napisanih znakov oz. besed.

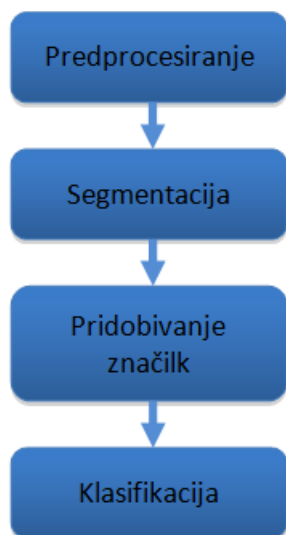


Slika 2.4: 4. smerna maska značilk.

2.2 Optična razpoznavna znakov v slikah naravnih scen

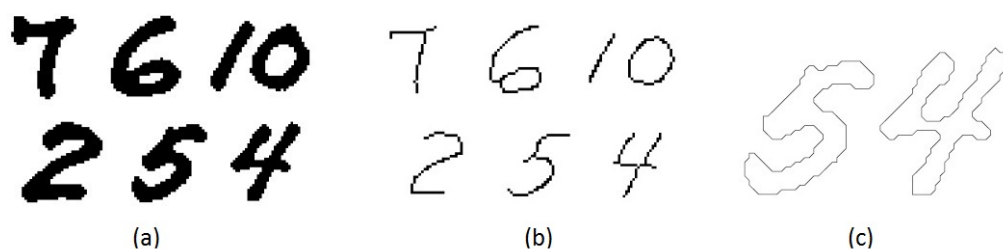
Sistemi za optično razpoznavo znakov v slikah naravnih scenah so sestavljeni iz štirih korakov, kot je prikazano na sliki 2.5.

Nekvalitetna slika znaka močno zniža natančnost razpoznave, zato je v fazi predprocesiranja potrebno sliko filtrirati. Pri predprocesiranju gre predvsem za odstranjevanje šuma z glajenjem in ostritvijo ter odstranitvijo izoliranih slikovnih elementov. Predprocesiranje izboljša rezultate segmentacije, pri kateri se slika binarizira v dva razreda: točke, ki ustrezajo znaku in točke ozadja. Pri pridobitvi značilk generiramo vektor značilk znaka, t.j. vektor najpomembnejših karakteristik, ki opisujejo določen znak. Vektor značilk služi kot vhod klasifikatorju, ki klasificira znake v ustrezne razrede. Pravilo pravi, da naj bi uporabili 5 do 10–krat več učnih vzorcev vsakega razreda, kot je dimenzionalnost našega vektorja značilk [6]. Pridobitev značilk je pomemben korak pri doseganju dobrih rezultatov v OCR sistemih, a morajo biti ostali koraki tudi optimizirani za pridobitev najboljših zmogljivosti. Metode pridobitve značilk delujejo na sivinskih slikah, na binarnih rastrskih slikah, stanjšanih simbolih



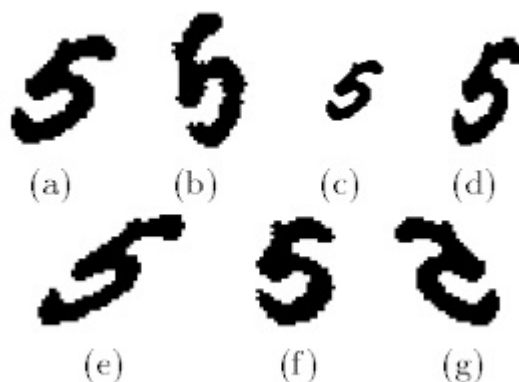
Slika 2.5: Osnovni diagram tipičnega sistema OCR.

oziroma okostju znaka ali pa na znakovnih obrisih (slika 2.6) [7].



Slika 2.6: (a) binarizacija znakov, (b) okostje znakov, (c) stanjšani znaki.

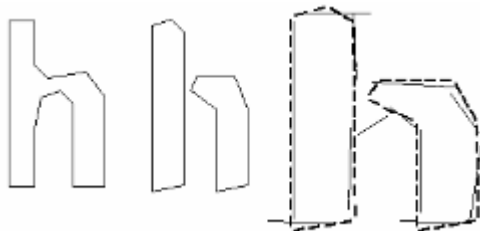
Če želimo razpoznati različne variacije istega znaka, moramo uporabiti značilke, ki so invariantne na določene transformacije (slika 2.7). Invariantne značilke so neodvisne od skaliranja, rotacije, raztega, zrcaljenja in translacije znaka.



Slika 2.7: Transformiran znak "5". (a) original, (b) rotiran, (c) skaliran, (d) raztegnjen, (e) zamaknjen, (f) zravnano, (g) zrcaljen.

2.2.1 Pregled področja

Zelo popularen sistem OCR je Tesseract [8], ki ga je razvijalo podjetje Hewlett-Packard v letih 1984–1994, sedaj pa je na voljo kot odprtokodna programska rešitev. Leta 2006 je Tesseract prevzelo podjetje Google in obenem zelo izboljšalo sistem. Tesseract-ova metoda pridobitve značilk je na začetku temeljila na topoloških značilkah, ki niso bile dovolj robustne za razpoznavo poškodovanih oz. razčlenjenih znakov (slika 2.8). Tesseract na klasičnih dokumentih deluje dobro, na slikah naravnih scen pa odpove, zato so potrebne metode prilagojene naravnim scenam. V nadaljevanju jih podajamo nekaj.



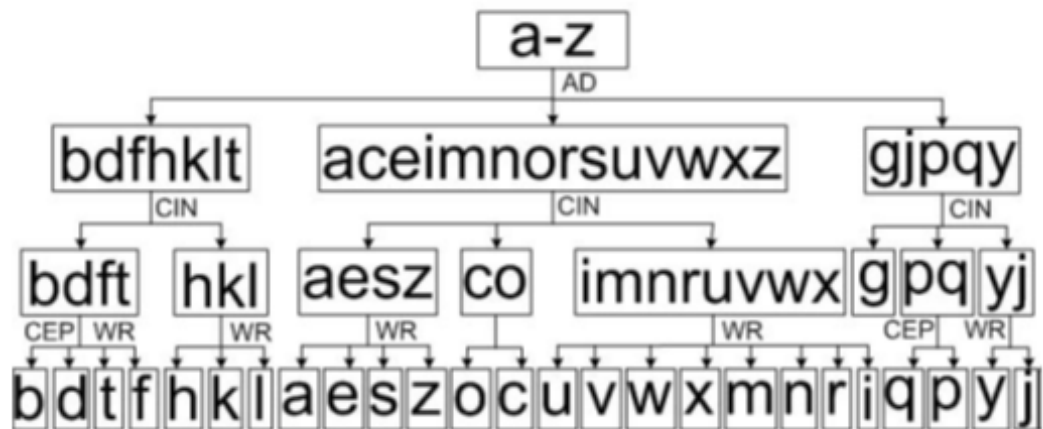
Slika 2.8: Skrajno desno, razčlenjen znak.

Ena izmed popularnejših tehnik je razpoznavna znakov s pridobitvijo Gaborjevih značilk, ki jih v svoji metodi uporabljajo tudi Zhang in sodelavci [9]. V fazi predprocesiranja histogram slike normalizirajo in s tem enakomerno porazdelijo osvetlitev slike. Po pridobitvi Gaborjevih značilk vektor značilk reducirajo z metodo LVQ (linearna vektorska kvantizacija) in LDA (linearna diskriminantna analiza). Končna klasifikacija pa temelji na klasifikatorju K-NN.

Tudi metoda, ki jo predlagajo Chen in sodelavci [10] temelji na Gaborjevih značilkah, pri čemer razpoznavajo tudi nagnjene znake, ki jih poravnajo z uporabo afininih transformacij. Sistem vključuje tudi prevod razpoznanega teksta.

Lu in sodelavci [11] se v svoji metodi osredotočajo na perspektivno invariantnost. Njihova rečitev vključuje tudi primarni nivo sodobnega OCR sistema. Predprocesiranje poteka v dveh iteracijah, pri čemer obe iteraciji vključujeta filtriranje slike. Slika je binarizirana s prilagodljivim pragom. Ekstrahirane značilke so invariantne na nagib. Z uporabo CAD (angl. Character Ascender and Descender) značilk posamezen znak razvrstijo v določene skupine, skupine pa nato razčlenijo na individualne znake (slika 2.9) [8]. Poleg CAD značilk uporabljajo tudi WR in CEP, kjer so vse invariantne na izkrivljenje perspektive.

Tekdas in sodelavci [12] predstavljajo metodo, ki temelji na oblikah konteksta (angl. Shape Context) in na značilkah valčkov. Metoda oblik konteksta [13] ekstrahira slikovne elemente glede na njihovo relativno pozicijo (na sliki procesirani s Canny operatorjem). Poleg omenjene metode pa Tekdas opisuje tudi pridobitev značilk t.i. biorthogonal spline wavelet, ki se celo bolje odrežejo kot metoda oblike konteksta. Pri klasifikaciji predlaga klasifikator SVM z RBF jedrom v povezavi s PCA (angl. principal component analysis). Pri razpoznavi znakov se je kot zelo učinkovita izkazala metoda podpornih vektorjev (SVM). Med drugim jo v svoji metodi uporabljajo tudi Campos in sodelavci [14], kot značilke pa uporabljajo oblike kontekstov (angl. Shape Context), geometrijsko



Slika 2.9: Tabela kategorizacije znakov.

zameglitev (angl. Geometric Blur), SIFT, MR8 in patch deskriptor. Ker je dimenzionalnost vektorjev velika, uporabljajo torbo vizualnih besed, kjer so vektorji kvantizirani v manj dimenzionalen prostor.

Poglavje 3

Uporabljena orodja

3.1 Matlab

Matlab (Matrix Laboratory) [15] je programski jezik in razvojno okolje za numerično računanje. Razvilo ga je podjetje MathWorks. Matlab omogoča računanje z matrikami, grafično predstavitev funkcij in podatkov, implementacijo algoritmov, izdelavo uporabniških vmesnikov in povezavo z ostalimi programskimi jeziki (C, C++, Java in Fortran). Vgrajeno razvojno okolje ima integriran urejevalnik, tolmač in veliko vizualizacijskih orodij. Razširitvena orodja ponujajo implementacijo algoritmov, ki se pogosto uporabljajo v znanosti in inženiringu. Programski jezik Matlab uporablja strukturne podatkovne tipe. Vse spremenljivke v Matlab-u so sezname oziroma strukturni sezname, kjer ima vsak element seznama enaka imena polj. Podpira tudi dinamična imena polj z imenskim poizvedovanjem posameznih polj, manipulacijo polj itd.

3.2 Zunanje knjižnice

Zaradi obsežnosti problema smo za razpoznavo znakov uporabljali tudi zunanje knjižnice in orodja, ki so del Matlaba. Te knjižnice omogočajo lažjo in hitrejšo

rešitev določenih problemov, predvsem na nivoju klasifikacije.

3.2.1 Statistični programski paket

Statistični programski paket (Statistics Toolbox) [16] vsebuje algoritme in orodja za organizacijo, analiziranje in modeliranje podatkov. Podpira tudi regresijo in klasifikacijo za modeliranje napovedi ter statistične izrise za raziskovalno podatkovno analizo.

3.2.2 Programski paket za obdelavo slik

Programski paket za obdelavo slik (Image Processing Toolbox) [17] ponuja standardne algoritme in grafična orodja za procesiranje slik, analizo, vizualizacijo in razvoj lastnih algoritmov. S paketom lahko izboljšamo kvaliteto slike, detektiramo značilke na njej, reduciramo šum, segmentiramo sliko in izvajamo geometrične transformacije. Večina funkcij tega orodja je večnitna in izkorišča večjedrne procesorje za časovno optimizacijo procesiranja slik. Poglavitne funkcije paketa, ki smo jih uporabili pri razpoznavi znakov so izboljšava kvalitete slike, filtriranje, razmeglitev, izostritev, slikovna analiza in segmentacija slike.

3.2.3 Programski paket za delo z nevronskimi mrežami

Programski paket za delo z nevronskimi mrežami (Neural Network Toolbox) [18] nam omogoča načrtovanje, implementacijo, vizualizacijo in simulacijo nevronskih mrež. Nevronske mreže se uporabljajo predvsem na področjih, kjer običajne analize ne dosega željenih rezultatov. Eno izmed takšnih področij je tudi razpoznavanje vzorcev, ki je prisotna pri problemu razpoznavanja znakov.

3.2.4 Knjižnica za uporabo metode podpornih vektorjev

Odprtokodna knjižnica za uporabo metode podpornih vektorjev (LibSVM) [19] temelji na implementaciji binarne metode podpornih vektorjev. Glavna prednost (v primerjavi z metodo podpornih vektorjev), ki je integrirana v Matlab–u, je podpora večrazredne klasifikacije, ki je nujna v primeru razpoznavne znakov.

Knjižnica vključuje tudi različne formulacije metode, učinkovito večrazredno klasifikacijo, navzkrižno validacijo za selekcijo modela, verjetnostno ocenitev, različna jedra, obtežen SVM za neuravnane podatke itd. Knjižnica podpira tudi vmesnike in razširitve za ostale programske jezike.

3.2.5 Knjižnica za pridobivanje vizualnih značilk

Knjižnica za pridobivanje vizualnih značilk (VLFeat) [20] je odprtokodna knjižnica z zbirko algoritmov s področja računalniškega vida. Poudarek knjižnice je na pridobitvi vizualnih značilk (npr. SIFT, MSER) ter grozdenju (k–means in hierarhični k–means). Knjižnica je napisana v programskem jeziku C, ki omogoča večjo učinkovitost in kompatibilnost z vmesniki v Matlabu.

Poglavje 4

Zbirke slik teksta naravnih scen

Za potrebe evalvacije metod razpoznavne znakov smo uporabili zbirke slik teksta v naravnih scenah ICDAR [1], CVL OCR DB [2] in hibridno zbirko, ki smo jo generirali iz obeh zbirk.

4.1 ICDAR

V okviru konference ICDAR [1] je organizirano tekmovanje iz razpoznavne in detekcije teksta v slikah naravnih scen Robust Reading Challenge. Tekmovanje poteka v treh stopnjah: lociranje teksta, razpoznavna znakov in razpoznavna besed. Za potrebe tekmovanja je bila generirana zbirka slik teksta v naravnih scenah [1], ki predstavlja standard za evalvacijo metod razpoznavne in detekcije teksta v slikah naravnih scen.

Zbirka vsebuje slike teksta v različnih resolucijah, zajete z različno opremo za zajem slik. Razdeljena je na tri množice – na vzorčno, učno in testno množico. Vsaka množica vsebuje datoteko XML (slika 4.1) s podatki o vsebini pripadajočih slik. Vzorčna množica je namenjena avtorjem, da lahko stestirajo delovanje svojih metod, medtem ko sta učna in testna množica namenjena samemu tekmovanju oz. evalvaciji metod. Natančneje, učna množica je namenjena učenju klasifikatorjev, testna pa evalvaciji. Vzorčna množica vsebuje

približno 800 znakov, učna in testna množica pa 6000 oz. 5400 znakov.

```
<image file="char/1/10.jpg" tag="S"/>  
<image file="char/1/11.jpg" tag="I"/>  
<image file="char/1/12.jpg" tag="S"/>  
<image file="char/1/13.jpg" tag="D"/>  
<image file="char/1/14.jpg" tag="A"/>
```

Slika 4.1: Izsek datoteke XML učne množice.

Zbirka ICDAR vsebuje raznolike znake, od zamegljenih, zamaknjenih, poševnih, do znakov z okrasno pisavo in tridimenzionalno pisavo. Slika 4.2 prikazuje nekaj primerov znakov iz zbirke ICDAR.



Slika 4.2: Primer znakov v zbirki ICDAR.

4.2 CVL OCR DB

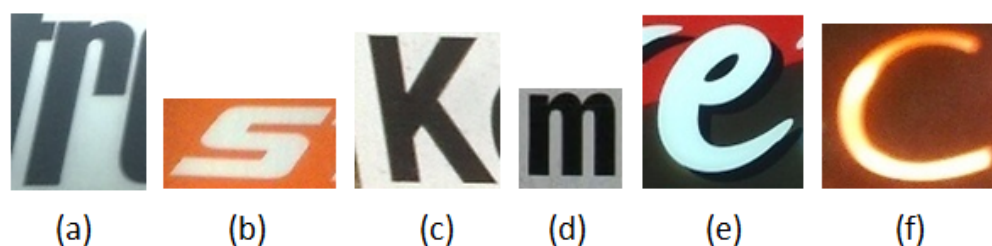
Zbirka slik teksta CVL OCR DB (angl. Computer Vision Laboratory OCR DataBase) [2] je javna zbirka anotiranih slik teksta v naravnih scenah. Zbirka vsebuje slike, zajete pod različnimi vremenskimi pogoji ter v različnih časovnih obdobjih. Posamezni vzorci znakov so ročno ekstrahirani iz večjih slik (slika 4.3) z anotacijskim programom TextAnnotator, ki je priložen zbirki. Slike v zbirki so bile zajete z različnimi fotoaparati in mobilnimi napravami, pod različnimi koti in pod spremenljivimi svetlobnimi pogoji, za boljšo simulacijo

realnih primerov. Nekaj primerov znakov iz zbirke je prikazanih na sliki 4.4. Celotna zbirka vsebuje približno 7600 slik.

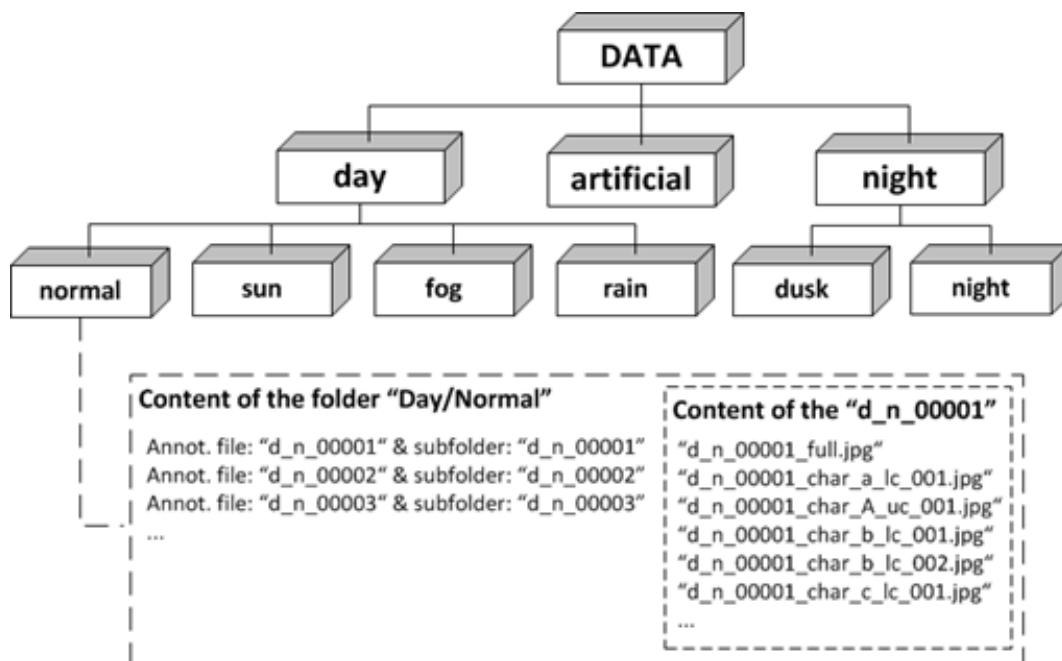


Slika 4.3: Prikaz pridobitve posameznih znakov iz besede.

Zbirka CVL OCR DB je razdeljena na tri glavne kategorije: »dan«, »noč« in »umetno«. Slike v kategoriji »dan« so bile zajete pri dnevni svetlobi. Kategorija »noč« vsebuje slike zajete v nočnem času. Kategorija »umetno« pa vsebuje slike, ki so bile zajete v prostorih z umetno osvetlitvijo (nakupovalni centri ipd.). Kategorija »dan« se dodatno deli na podkategorije »normalno«, »sončno«, »megla« in »dež«, pri čemer vsaka predstavlja vremenske pogoje ob katerih so bile slike zajete. Kategorija »noč« vsebuje podkategoriji »mrak« in »noč« (slika 4.5).



Slika 4.4: Nekaj primerov znakov iz različnih kategorij podatkovne zbirke CVL.



Slika 4.5: Datotečna organizacija zbirke slik teksta CVL OCR DB.

Slika 4.5 prikazuje datotečno strukturo zbirke CVL OCR DB. Vsaki kategoriji pripada mapa, v kateri so shranjene vse slike pripadajoče kategorije in datoteke XML, ki vsebujejo anotirane podatke (lokacija teksta v sliki, lokacija posameznih črk v sliki ipd.). Vsak znak v zbirki je predstavljen z imenom, sestavlja ga predpona kategorije (npr. kategorija dan/normal je označena z d.n, kar prikazuje slika 4.5), ki ji sledijo zaporedna številka znaka v kategoriji, koda znaka, oznaka za veliko/malo črko ter zaporedna številka znaka v sami sliki. Spodaj podajamo še nekaj primerov imen datotek znakov:

"d.f_00325_char_C_uc_001.jpg",
 "d.f_00325_char_C_uc_002.jpg",
 "d.f_00325_char_a_lc_001.jpg",
 "d.f_00325_char_A_uc_001.jpg".

4.3 Hibridna zbirka

Zaradi potrebe po boljši učni množici, smo ustvarili hibridno zbirko slik teksta med zbirkama ICDAR [1] in CVL OCR DB [2]. Pri tem postopku smo iz obeh zbirk izbrali slike najboljših kvalitet (slika 4.6). Tako smo dobili razmeroma optimalen reprezentativni učni model posameznega znaka oziroma razreda. Slike so bile izbrane ročno, na podlagi hitre ocene njihove kvalitete. Celotna hibridna zbirka znakov vsebuje preko 3800 znakov, ki so v večini primerov nepoškodovani. Vsak znak oz. razred vsebuje povprečno 70 učnih vzorcev.



Slika 4.6: Primeri slik hibridne zbirke slik teksta.

Poglavje 5

Predlagana metoda

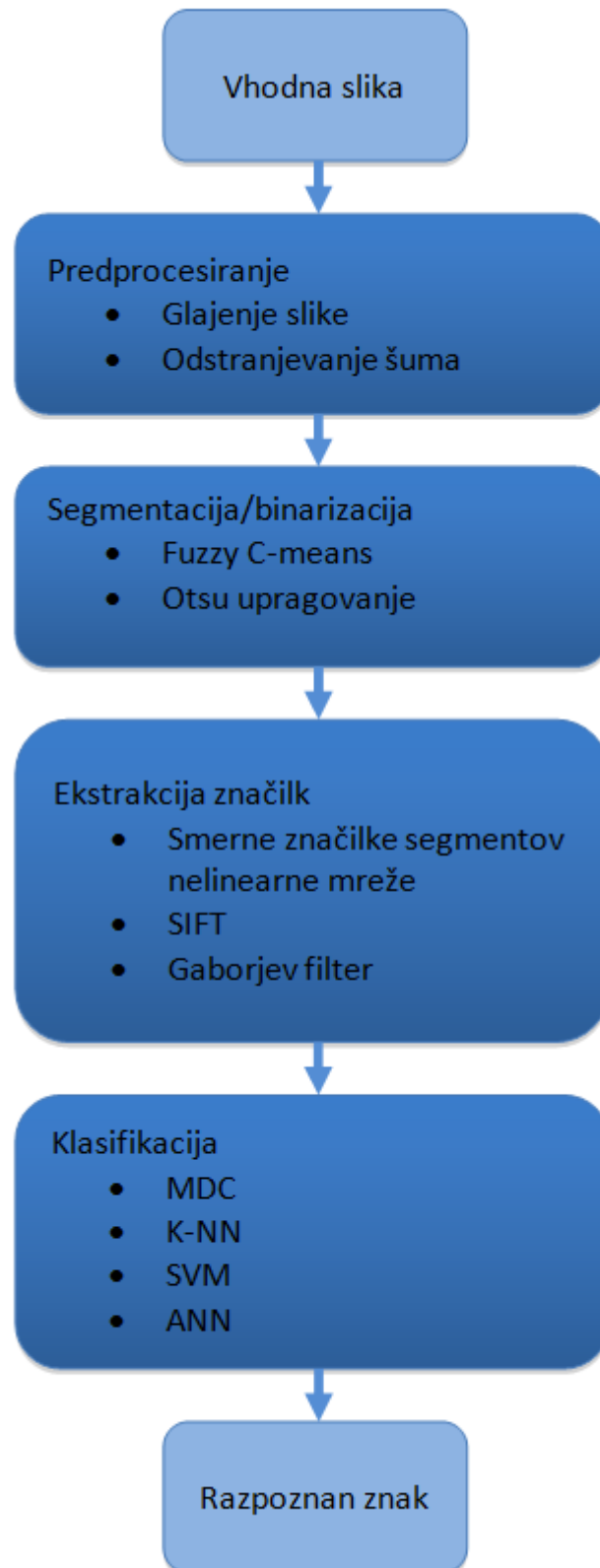
5.1 Opis metode

Implementirali in nadgradili smo metodo, ki so je predlagali Jonghyun in sodelavci [21]. Metoda zajema tri glavne korake: binarizacijo na podlagi FCM, pridobitev značilk smernih segmentov nelinearne mreže in klasifikacijo na podlagi MDC. Predlagano metodo smo nadgradili in razširili, kar prikazuje slika 5.1. Vsak nivo vsebuje več različic, celotna metoda pa deluje tako, da se na vsakem nivoju izbere le posamezna različica. Na primer, segmentacija na podlagi upragovanja Otsu, značilke na podlagi Gaborjevega filtra ter klasifikator SVM.

Predlagani metodi smo dodali nivo predprocesiranja, ki sliko zgladi in odstrani vsebujoč šum.

Segmentacija temelji na dveh metodah: na povezovanju v gruče Fuzzy C-means (FCM) in na Otsu upragovanju. FCM omogoča določitev pripadnosti delčka v sliki enemu ali večim skupkom, medtem ko se Otsujevo upragovanje izvaja avtomatično glede na histogram slike.

Poleg pridobitve značilk smernih segmentov nelinearne mreže smo uporabili še dva popularna tipa značilk: SIFT in Gaborjev filter. Značilke SIFT (angl. Scale Invariant Feature Transform) so lokalne in temeljijo na videzu objekta



Slika 5.1: Diagram metode.

ter delujejo invariantno na skaliranje slike in njeno rotacijo. Gaborjev filter [22] je linearni filter, ki se pogosto uporablja za detekcijo robov. Prav tako kot SIFT je tudi Gaborjev filter invarianten na rotacijo, skaliranje in translacijo.

Zadnji nivo predlagane metode je klasifikacija. Cilj klasifikatorja je izdelati model iz učne množice, ki lahko napove ciljne vrednosti testnih podatkov, pri čemer imamo podane samo attribute testne množice. Predlagana metoda klasifikacije je MDC, ki dodeli neznan vzorec pripadajočemu razredu z izračunom najkrajših razdalj med njunima vektorjema značilk. Podobno kot klasifikator MDC deluje tudi klasifikator K-NN (k najbližjih sosedov), pri katerem učenje deluje tako, da neznan vzorec dodeli znanemu, glede na razdaljo med njima, posamezen objekt pa klasificira z večinskim glasovanjem bližnjih sosedov. Ob MDC in K-NN smo za klasifikacijo uporabili tudi ANN ter SVM. Umetna nevronska mreža (ANN) je primer računalniškega modela, ki se zgleduje po načinu, kako biološki živčni sistem, kot so možgani, procesirajo informacije. Metoda podpornih vektorjev (SVM) pa deluje z uporabo hiperploskve, s katero ločuje vzorce različnih razredov.

5.2 Predprocesiranje

Nivo predprocesiranja je skupek operacij, ki izboljša sliko ter omogoča kvalitetnejšo binarizacijo le-te. V slikah naravnih scen se pogosto pojavljajo šumi, kot so: neenakomerno porazdeljena svetloba, zamegljenost slike, slaba osvetljenost, zamaknjenost, itd. Vsi naštetih faktorji onemogočajo kakovostno pridobitev značilk. Predprocesiranje je torej postopek, s katerim posledično izboljšamo rezultate ostalim nivojem procesiranja slike.

Pri predprocesiranju smo sliko najprej izostrili s konvolucijsko matriko v velikosti 3×3 . Izostritveno matriko imenujemo tudi visokofrekvenčni filter, saj prepušča skozi samo visoke frekvence. Jedro visokofrekvenčnega filtra je zasnovano tako, da poveča svetlost osrednjega slikovnega elementa relativno na sosednje slikovne elemente. Jedro običajno vsebuje centralno vrednost, ki

je pozitivna in obkrožena z negativnimi vrednostmi:

$$\begin{bmatrix} -1/6 & -4/6 & -1/6 \\ -4/6 & 26/6 & -4/6 \\ -1/6 & -4/6 & -1/6 \end{bmatrix}$$

Slika 5.2: Primer jedrne matrike velikosti 3×3 , ki je namenjena ostrenju slike:

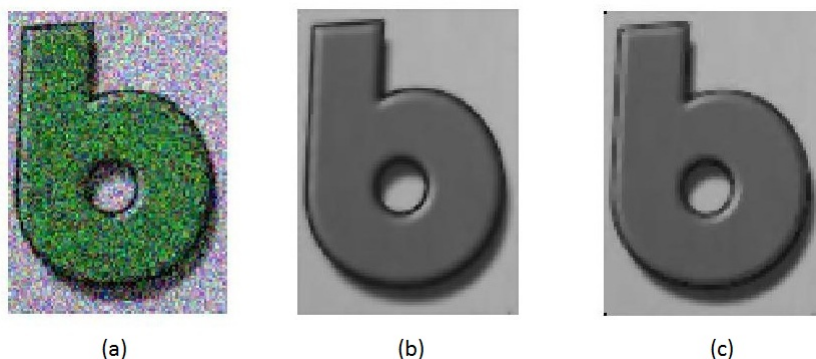
Po izostritvi slike se postopek predprocesiranja nadaljuje z Wiener dekonvolucijo, ki je v bistvu aplikacija Wiener filtra [23], ki se uporablja za odpravo šumov v sliki.

Končni korak predprocesiranja je bil glajenje slike, ki smo ga poskušali realizirati z dvema filtroma za odstranjevanje šuma (slika 5.3). Prvi je bil prilagodljiv dvodimenzionalni Wiener filter, drugi pa medianin filter. Oba filtra sta bila aplicirana z jedrom v velikosti 3×3 .

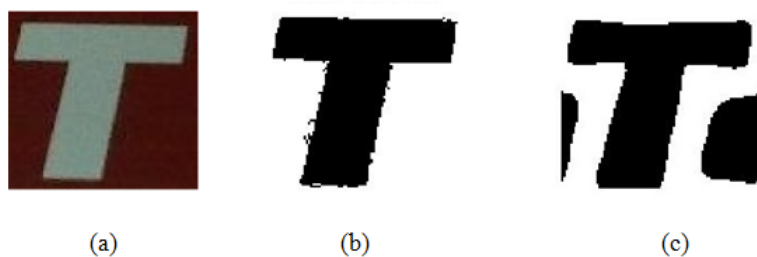
Medianin filter [24] je nelinearen filter, ki zagotavlja učinkovito odpravljanje šuma s precej manj glajenja kot filtri povprečenja. V nasprotju z medianinim filtrom, ki je nelinearen, je Wiener-jev filter linearen. Wiener filter se prilagaja sliki ter se spreminja z lokalno varianco slike. Tam kjer je varianca večja, Wienerjeva funkcija opravi manj glajenja, tam kjer je pa manjša pa se aplicira več glajenja na to območje.

Kot je razvidno iz slike 5.3 se je malenkost bolje odrezal Wiener filter, saj tudi bolje obravnava gauss-ov šum, medianin filter pa, po drugi strani bolje obravnava t.i. salt and pepper šum. Zaradi boljših rezultatov Wiener filtra smo se odločili za njegovo uporabo na nivoju predprocesiranja.

Z vsemi naštetimi postopki smo lahko uspešno konstruirali nivo predprocesiranja. Pri tem smo opazili, da se lahko nekatere slike, ki so dobro razpoznavne malce popačijo (slika 5.4) in da se pri tistih, pri katerih znaki niso bili dovolj dobro razpoznavni, predprocesiranje dobro odreže (slika 5.5).



Slika 5.3: Primerjava Wiener filtra z medianinim filtrom. Prvotni sliki je bil dodan šum z vrednostjo 0.025. (a) Originalna slika, (b) Wiener filter, (c) medianin filter.



Slika 5.4: Primer dobre slike, ki jo s predprocesiranjem malce popačimo. (a) Originalna slika, (b) binarizirana slika s FCM brez predprocesiranja, (c) binarizirana slika s FCM in predprocesiranjem.



Slika 5.5: Primer binariziranega znaka, z uporabo predprocesiranja in brez. (a) Originalna slika, (b) binarizirana slika s FCM brez predprocesiranja, (c) binarizirana slika s FCM in predprocesiranjem.

5.3 Segmentacija

V našem primeru ločimo slikovne elemente na tiste, ki pripadajo znaku in na tiste, ki pripadajo ozadju. Ker gre za dvorazredni primer segmentacije, uporabljamo izraz binarizacija¹.

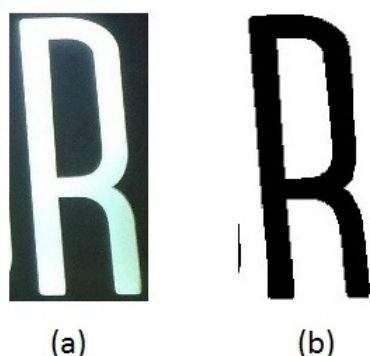
Binarizacija [25] pretvori sivinsko sliko, ki ima do 256 odtenkov sivin, v črno-belo sliko. Najpreprostejši način binarizacije je uporaba mejne vrednosti oziroma praga, pri čemer klasificiramo vse slikovne elemente, z vrednostmi večjimi od praga, kot bele in ostale kot črne slikovne elemente. Problem binarizacije je določanje praga, saj lahko z različnimi pragovi izpostavimo različne objekte na sliki. V veliko primerih je iskanje praga, ki je optimizirano za celotno sliko zelo težko, zato je najbolj primerna metoda prilagodljiva binarizacija, kjer je izbran optimalen prag za vsako lokalno območje slike. Tak primer binarizacije je dvorazredni FCM.

Poleg uporabe praga je pogosta metoda tudi povezovanje v gruče (angl. clustering). Grozdenje imenujemo tudi segmentacija podatkov, ker razdeli velike zbirke podatkov v določeno število skupin glede na njihovo podobnost. Grozdenje temelji na meritvi podobnosti med pari objektov z uporabo distančne funkcije. Tako je rezultat grozdenja niz sklopov, kjer so si objekti znotraj enega sklopa bolj podobni med seboj, kot objekti drugega sklopa. Grozdenje je uporabljeno tudi na drugih področjih, kot so procesiranje medicinskih slik, raziskave trga in analiza podatkov.

5.3.1 FCM

Fuzzy C-means (FCM) [21] je tehnika podatkovnega grozdenja, poznana tudi kot Fuzzy ISODATA, kjer je vsaka podatkovna točka pripisana nekemu skupku podatkov. FCM dobro obravnava spremembe v svetlosti (slika 5.6) in slike s slabo osvetlitvijo.

¹Skozi diplomsko delo uporabljamo namesto segmentacije tudi izraz binarizacija, kljub temu, da izraza ne označujeta iste stvari.



Slika 5.6: Primer FCM binarizacije znaka z neenakomerno osvetljenostjo. (a) originalna slika, (b) binarizirana slika

Binarizacijski algoritem Fuzzy C-means razdeli podatke na dva razreda in dodeli podatkom vsakega razreda vrednosti 0 ali 1 (črno in belo). FCM je iterativni algoritem, katerega cilj je najti centre sklopov (centroide), ki minimizirajo objektivno funkcijo. Z iterativnim posodabljanjem centrov sklopa in pripadnosti razreda za vsako podatkovno točko, FCM iterativno premika centre sklopa na pravo lokacijo v podatkovni množici. Centroida sklopa je izračunana kot povprečje vseh točk, ponderirana s stopnjo pripadnosti določenega razreda. Stopnja pripadnosti nekemu sklopu je povezana z inverzno distanco sklopa.

Fuzzy C-means algoritem uporablja nenadzorovan pristop, ki temelji na minimizaciji objektne funkcije, ki je bila uporabljena v segmentaciji slike. Algoritem dodeli slikovne elemente vsakemu sklopu z uporabo »mehke«
pripadnosti.

5.3.2 Binarizacija Otsu

Otsu upragovanje [26] uvrščamo med globalne upragovalne tehnike, ki predvidevajo, da slika vsebuje dva razreda slikovnih elementov oz. ima bimodalni histogram (ospredje in ozadje) (slika 5.7). Upragovanje Otsu je optimalno upragovanje, ki se pogosto uporablja na področju računalniškega vida. Je zelo hitro, saj operira samo s sivinskimi histogrami.



Slika 5.7: Primer binariziranega znaka z Otsu upragovanjem.

Algoritem Otsu temelji na ideji, ki poišče optimalen prag, ki minimizira ponderirano znotraj/intra razredno varianco dveh razredov oz. klasifikacijsko napako ozadja in ospredja. Kot se izkaže je minimiziranje znotraj/intra razredne variance enako kot maksimiziranje medrazredne variance.

5.4 Izbira značilk

Glavni namen pridobivanja značilk je izvleči tiste attribute vzorcev, ki so najbolj primerni za klasifikacijo in povečanje učinkovitosti razpoznave določenega vzorca. Izbor stabilnih in reprezentativnih nizov značilk je bistvenega pomena za dobro delovanje sistemov za razpoznavo znakov [3].

5.4.1 Značilke smernih segmentov z nelinearno mrežo

Značilke smernih segmentov nelinearne mreže [21] spadajo v kategorijo statističnega pristopa, ki obravnava celoten znak kot vzorec in ga poskuša razlikovati z ostalimi, z uporabo podatkov o celostni obliki znaka na podlagi nelinearne mreže. Statistični pristop je v primerjavi s strukturnim bolj primeren za slike naravnih scen, ki vsebujejo šume, kot so spremembe v svetlosti in različne vrste pisav.

Pri pridobitvi značilk smernih segmentov nelinearne mreže za vsak vzorec oziroma binarizirano sliko pripravimo $N \times M$ nelinearno mrežo, ki je invariantna na skaliranje. Segmenti mreže so razdeljeni na horizontalno (H), vertikalno

(V), levo–diagonalno (L) in desno–diagonalno (R) smer.

Predpostavljamo, da je vzorec znaka, slika višine H in širine W . Funkcija $f(i,j)$ predstavlja posamezen slikovni element (i,j) v i -ti vrstici in j -tem stolpcu, tu velja sledeč pogoj:

$$1 \leq i \leq H, \text{ in } 1 \leq j \leq W.$$

Pri konstrukciji $N \times M$ nelinearne mreže upoštevamo pogoje:

$$N \leq H, M \leq W,$$

Nelinearno mrežo izračunamo z naslednjimi koraki:

1. Najprej so izračunani horizontalni in vertikalni projekcijski profili, $h(i)$ in $v(j)$ z naslednjima enačbama:

$$h(i) = \sum_{k=1}^W f(i, k) \quad (5.1)$$

kjer velja:

$$1 \leq i \leq H$$

in

$$v(j) = \sum_{k=1}^H f(k, j) \quad (5.2)$$

kjer velja:

$$1 \leq j \leq W$$

2. Vertikalni in horizontalni profili izračunajo vsoto črnih slikovnih elementov posamezne vrstice. Vrstice vhodne slike so nato indeksirane od 1 do H (višina) in particionirane v N zaporedne trakove. Trakove delijo mejni

indeksi $[1, e_1], [e_1 + 1, e_2], \dots, [e_{N-1} + 1, H]$, kjer so v posameznem traku robni indeksi e_1, e_2, \dots, e_{N-1} določeni tako, da so gostote črnih slikovnih elementov enake za vse trakove. Določanje mejnih indeksov je definirano s spodnjo enačbo:

$$D(k) = \sum_{i=s_k}^{e_k} h(i) \quad (5.3)$$

kjer velja:

$$\begin{aligned} s_1 &= 1, s_{k+1} = e_k + 1, e_N = H \\ &\text{in} \\ 1 &\leq k \leq N. \end{aligned}$$

Tako kot vrstice, so tudi stolpci vhodne slike indeksirani od 1 do W (širina) in razdeljeni v M število trakov z vertikalno projekcijo $v(j)$. Z izračunanimi N horizontalnimi trakovi in M vertikalnimi trakovi je lahko formirana $N \times M$ mreža (sliki 5.8, primer a). Nelinearna mreža je primerna za pridobitev značilk v slikah znakov z različnimi pisavami in znakovnimi dekoracijami, ki so pogoste v naravnih scenah.

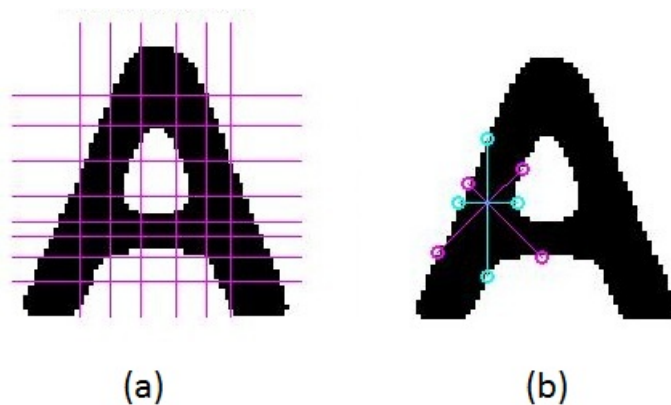
Po generiranju nelinearne mreže so 4-smerni segmenti izračunani za vsak črn slikovni element v sliki, upoštevajoč da so strukturne karakteristike slikovnega elementa predstavljene s horizontalno, vertikalno ter dvema glavnima diagonalnima linijama. Za vsak na črn slikovni element (x, y) vhodne slike, sta izračunani dolžini $RLH_{x,y}$ in $RLV_{x,y}$, z merjenjem njune horizontalne oziroma vertikalne poti. Meritev dolžine se ustavi ob dotiku prvega belega slikovnega elementa (5.8, primer b). Z uporabo teh dveh vrednosti, lahko izračunamo vrednosti horizontalnega in vertikalnega prispevka $DCH_{x,y}$ in $DCV_{x,y}$ s sledečima formulama:

$$DCH_{x,y} = RLH_{x,y} / (RLH_{x,y} + RLV_{x,y}) \quad (5.4)$$

in

$$DCV_{x,y} = RLV_{x,y} / (RLH_{x,y} + RLV_{x,y}). \quad (5.5)$$

Vrednosti smernih prispevkov $DCH_{x,y}$ in $DCV_{x,y}$ so povprečeni s številom vseh črnih slikovnih elementov v posamezni celici $N \times M$ mreže. S tem povprečenjem dobimo prvo polovico vseh značilk, torej $2 \times N \times M$ vrednosti. Druga polovica $2 \times N \times M$ značilk za levo–diagonalno in desno–diagonalno smer, je izračunana enako kot prva polovica, le da je slika znaka rotirana za 45° (slika 5.8, primer b, dolžine vijolične barve).

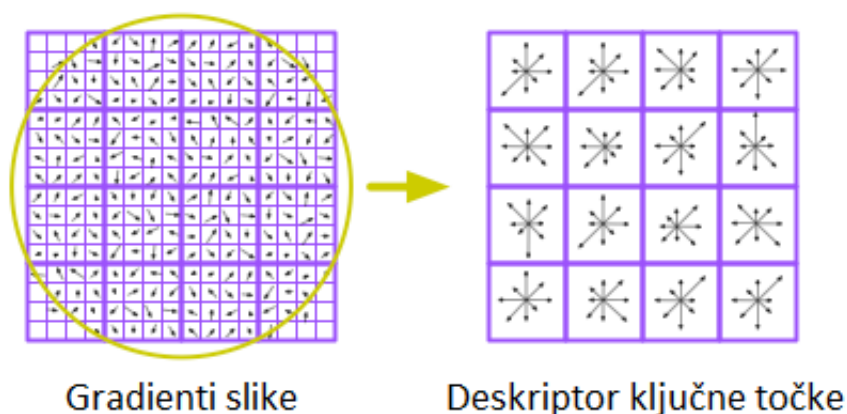


Slika 5.8: : (a) Nelinearna mreža velikosti 9×7 , (b) 4–smerna značilka, modra označuje horizontalne in vertikalne dolžine, vijolična pa predstavlja diagonalne dolžine (leve in desne).

5.4.2 SIFT

Algoritem SIFT [27, 28] pretvori sliko v zbirko vektorjev značilk, kjer je vsak vektor invarianten na translacijo, skaliranje in rotacijo slike ter deloma invarianten na spremembe osvetlitve in na lokalno geometrično popačenje. Algoritem SIFT za vsak objekt na sliki ekstrahira interesne točke objekta, ki opišejo objekt. Opis objekta je lahko uporabljen pri identifikaciji enakega objekta na drugi sliki z drugačno vsebino.

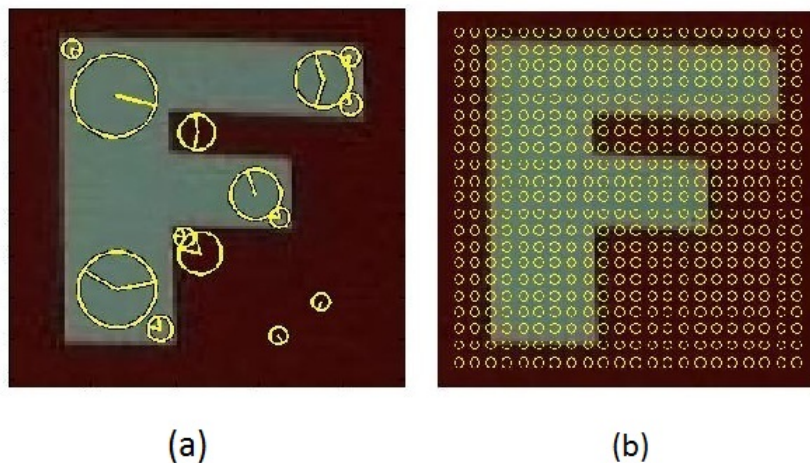
Ključne lokacijske točke (5.9) so definirane kot najvišje in najmanjše razlike Gaussove funkcije uporabljene v skalirnem prostoru. Točke s slabim kontrastom in robovi so zavržene, prevladujoče orientacije pa so zapisane v lokalne ključne točke, kar jih naredi bolj stabilne za primerjavo in razpoznavo.



Slika 5.9: Deskriptor ključne točke SIFT, dimenzije 1×128 .

Algoritem SIFT se prične s pretvorbo barvne slike v sivinsko in implementacijo Gaussovega filtra, kateremu sledi uporaba osnovne teorije skalirnega prostora. Piramidni nivo skalirnega prostora je sestavljen iz večje količine podatkov, metoda SIFT nato minimizira in določi njeno najvišjo vrednost. Ta vrednost je opredeljena z implementacijo lokalizacijske tehnike, ki primerja najvišjo in najnižjo vrednost sosednjih vrednosti vseh piramidnih nivojev. Največja in najmanjša vrednost piramidnega nivoja je lahko interpretirana tudi kot izrazit rob procesirane slike. Po dobljenih najvišjih in najmanjših vrednosti, poiščemo vrednosti orientacije gradientov za doseganje invariantnosti rotacije.

Pri pridobivanju značilik SIFT smo uporabili metodo »gosti« SIFT [20] za pridobitev večjega števila značilik (slika 5.10, primer b). Ta metoda je podobna pridobivanju značilik SIFT, vendar so značilke ekstrahirane na nasičeni mreži lokacij, pri določeni velikosti in rotaciji.



Slika 5.10: (a) SIFT s ključnimi točkami, (b) gosti SIFT.

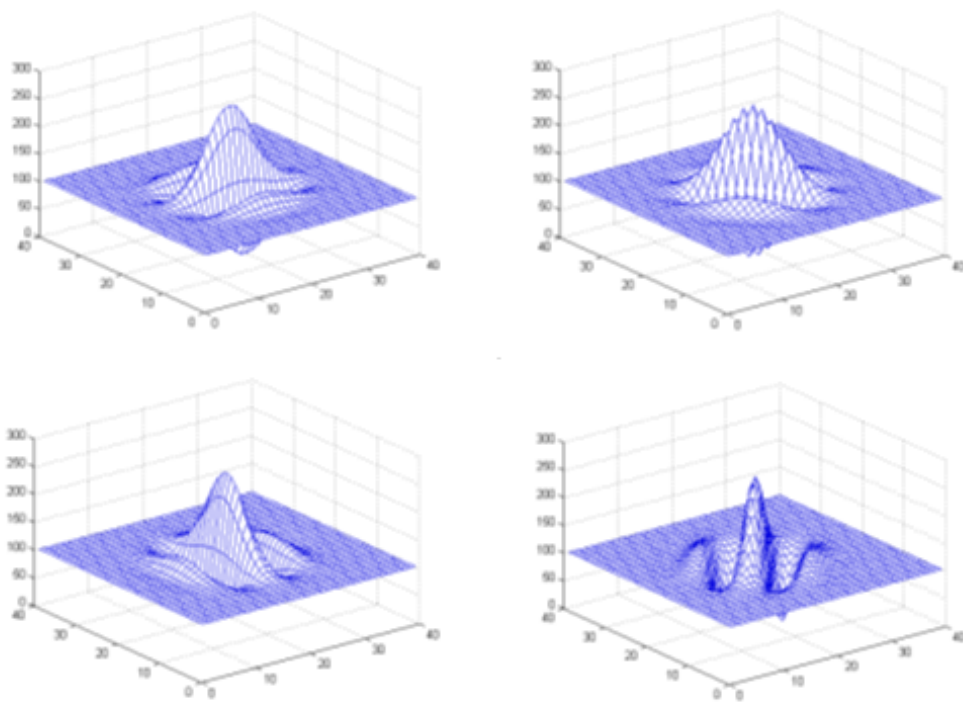
5.4.3 Gaborjev filter

Gaborjev filter [22] lahko predstavlja signale v enodimenzionalnem in dvodimenzionalnem prostoru. V prostorski domeni je apliciran dvodimenzionalen Gaborjev filter.

Gaborjev filter se neposredno nanaša na Gaborjeve valčke, saj so lahko zasnovani za različne velikosti in rotacije. Razširitev Gaborjevih filtrov z Gaborjevimi valčki se večinoma ne uporablja, saj to zahteva izračun dvo-ortogonalnih valčkov, kar je časovno potratno, zato uporabljamo skupek Gaborjevih filtrov z različnimi velikostmi in rotacijami. Gaborjev prostor je uporaben pri procesiranju slik ter razpoznavi znakov, šarenice, obrazov in prstnih odtisov [22].

Gaborjev valček je sinusna ravnina s specifično frekvenco in orientacijo, ki je modulirana z Gaussovo ovojnico. Valček lahko označi prostorsko strukturo frekvence na sliki in obenem ohrani informacijo prostorskih razmerij, tako je primeren za pridobitev orientacijsko odvisne frekvenčne vsebine vzorcev [9].

Frekvenčni odzivi in štiri orientacije (0° , 45° , 90° in 135°), uporabljene tudi pri implementaciji Gaborjevega filtra so prikazani na sliki 5.11.



Slika 5.11: : Štiri smerni Gaborjev filter.

5.5 Klasifikacija

Klasifikacija je problem identifikacije, kateri skupini vzorcev pripada nek nov neznan vzorec. Vsaka izmed skupin je označena z neko oznako, ki predstavlja dotičen klasifikacijski razred [29]. Cilj klasifikacije je določiti vsakemu vzorcu razredno oznako ali oceno pripadnosti.

Pri našem delu smo uporabili klasifikator K–NN in predlagan klasifikator MDC, ki spadata v kategorijo statističnih klasifikatorjev. Statistične klasifikacijske metode temeljijo na Bayesovi odločitveni teoriji, ki si prizadeva zmanjšati klasifikacijsko napako, obenem nam poda tudi verjetnostno oceno klasifikacije. Poleg omenjenih dveh klasifikatorjev smo uporabili tudi umetne nevronske mreže (ANN) in metodo podpornih vektorjev (angl. support vector machines), ki je v osnovi neverjetnostni binarni linearni klasifikator [3].

5.5.1 MDC

Klasifikator MDC [21] zagotavlja klasifikacijo pri minimalnem številu zahtevanih parametrov in računske zahtevnosti, vendar zaradi nizke računske zahtevnosti klasifikacijska natančnost ni optimalna. Algoritem MDC poišče srednjo vrednost oz. povprečje vzorcev posameznih razredov in izmeri razdaljo med povprečji vzorcev razreda in testnega oziroma neznanega vzorca. Testni vzorec torej dodeli tistemu razredu, katerega povprečna vrednost je najbližja neznanemu oziroma testnemu vzorcu. MDC je apliciran na področju klasifikacije vzorcev, kot je na primer diagnoza bolezni ali mamografija [30].

MDC je Bayesov klasifikator, kjer drži predpostavka, da so vzorci vsakega razreda statistično samostojni in porazdeljeni normalno s skupno varianco. Predpostavka normalne porazdelitve s skupno varianco ni splošno veljavna, zato tudi klasifikacijska natančnost klasifikacije MDC ni optimalna, a zadoštuje za predhodno klasifikacijo večjega sklopa znakov [21]. Pri implementaciji klasifikatorja najmanjših razdalj smo morali med množico K povprečnih vektorjev, ki predstavljajo posamezne razrede znakov (K razredov) $\mu_1, \mu_2, \dots, \mu_k$

in neznanim vzorcem x , izračunati Evklidsko razdaljo $\|x - \mu_i\|$ od x do i -tega povprečnega vektorja $\mu_i, i = 1, \dots, K$. Po izračunu razdalj smo dodelili neznanemu vzorec x , tistemu razredu K , ki je bil najbližji.

Evklidska razdalja je običajna razdalja med dvema točkama, ki se izračuna po Pitagorevi formuli [31]. Če sta $p = (p_1, p_2, \dots, p_n)$ in $q = (q_1, q_2, \dots, q_n)$ dve točki v Evklidskem n -prostoru je razdalja od p do q ali obratno, opisana kot:

$$d(p, q) = d(q, p) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \dots + (q_n - p_n)^2} = \sqrt{\sum_{i=1}^n (q_i - p_i)^2} \quad (5.6)$$

MDC s prototipnim učenjem

Klasifikator minimalnih razdalj (angl. minimum distance classifier), generira en prototip za vsak razred. Če imamo 52 znakov v angleški abecedi, mora biti 52 prototipov na voljo za klasifikator MDC, $\mu_1, \mu_2, \dots, \mu_{52}$.

I -ti prototip μ_i je izračunan s povprečenjem vektorjev značilnik v celotni matriki značilnik, ki pripadajo določenemu i -temu razredu (znaku):

$$\mu_i = \frac{1}{vz} \sum_{j=1}^{vz} F_j^i \quad (5.7)$$

F_j^i je j -ti vzorec i -tega razreda, vz je število vzorcev posameznega razreda.

Ob podanem neznanem vzorcem x , klasifikator MDC izmeri Evklidsko razdaljo $\|x - \mu_i\|$ od x do i -tega prototipa $\mu_i, i = 1, \dots, 52$, nato izračuna množico 5 najbolj verjetnih razredov, katerim naj bi pripadal ta vzorec. Razredi so urejeni po velikosti, naraščajoče po velikosti razdalje. Tem manjša razdalja je med neznanim vzorcem x in μ_i , tem večja je verjetnost da x pripada i -temu razredu.

V predlagani metodi klasifikacije MDC je uporabljena Evklidska razdalja, vendar smo dosegali boljše rezultate z uporabo Manhattanske razdalje oziroma

razdalje mestnega bloka (angl. cityblock distance).

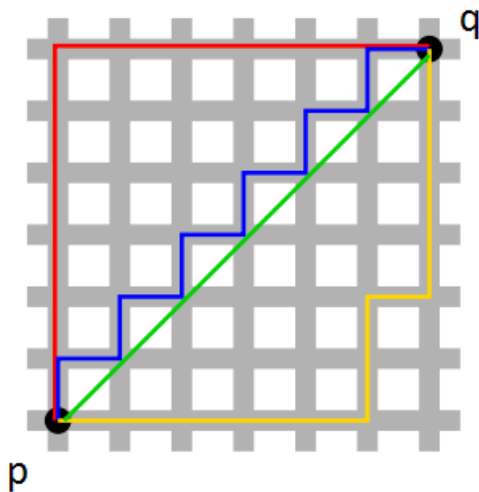
Manhattanska razdalja (slika 5.12) med dvema točkama temelji na horizontalnih in vertikalnih poteh, torej je enostavna vsota vodoravnih in navpičnih komponent. Manhattanska razdalja, med dvema vektorjema p in q v n -dimenzionalnem realnem vektorskem prostoru s fiksnim kartezičnim koordinatnim sistemom, je vsota dolžin projekcij odseka črte med točkami na koordinatnih osi [32].

Manhattansko razdaljo izračunamo z izrekom:

$$d_1(p, q) = \|p - q\|_1 = \sum_{i=1}^n |p_i - q_i|, \quad (5.8)$$

kjer je:

$$p = (p_1, p_2, \dots, p_n) \text{ in } q = (q_1, q_2, \dots, q_n)$$



Slika 5.12: Manhattanska razdalja med točkama p in q je označena z rdečo, modro in rumeno. Evklidska razdalja je predstavljena z zeleno linijo.

Prerazporeditev kandidatov za razponavo

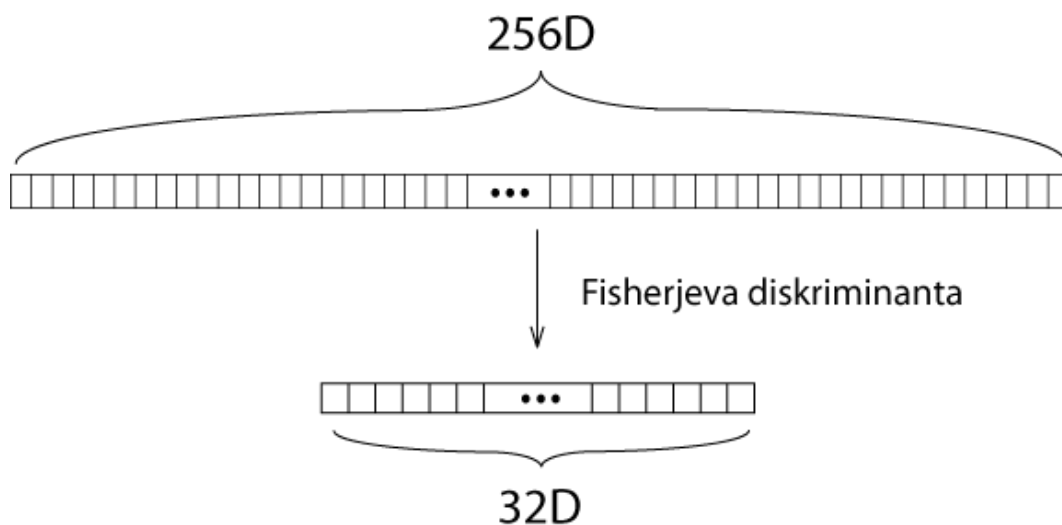
Prerazporeditveni modul ponovno razporedi razvrščene množice najboljših 5 kandidatov z uporabo navzkrižnega (angl. pairwise) klasifikatorja. Ponovno razporejanje se začne s parom dveh kandidatov največjih razdalj ali najmanjše podobnosti, nato je zmagovalec te klasifikacije in naslednji kandidat pripravljen za še eno razporejanje, itd. dokler ne dobimo zmagovalca. Ob podanih potencialnih 5 kandidatov imamo torej 4 primerjave med posameznimi kandidati.

Navzkrižni klasifikator za i -ti in j -ti razred ($1 \leq i, j \leq 52$) je klasifikator MDC, ki uporablja 32 dimenzionalne značilke in izračuna namanjšo razdaljo med reduciranim testnim vektorjem in dvema kandidatom. Če tekmujeta kandidata 5 in 4, bo izračunana najmanjša razdalja med testnim vektorjem in obema kandidatom, tisti, ki bo dosegel boljši rezultat oziroma manjšo razdaljo bo zmagal. 32 dimenzionalni vektor oziroma značilka je konstruirana kot podmnožica originalne 252 dimenzionalne ($N \times M \times 4$) smerne segmentne značilke (slika 5.13).

Selekcija vrednosti značilk temelji na Fisher-jevi diskriminantni meritvi, ki je izračunana kot:

$$F_{ij} = \frac{\sigma_{ij}(k)}{\sigma_i(k) + \sigma_j(k)}, \quad (5.9)$$

kjer je $\sigma_i(k)$ varianca k -te vrednosti vzorca, ki pripada i -temu znaku razreda, $\sigma_j(k)$ je varianca k -te vrednosti vzorca, ki pripada j -temu znaku razreda in $\sigma_{ij}(k)$ je varianca k -te vrednosti vzorca, ki pripada obema razredoma. Večja vrednost $F_{ij}(k)$ implicira, da ima k -ta značilka visoko diskriminantno težo, med i -tim in j -tim razredom. Dolžina reduciranega vektorja je približno 12% dolžine primarnega vektorja, kar pomeni, da s Fisherjevo diskriminanto izberemo 32 značilk, ki imajo najvišjo Fisherjevo meritev [21].

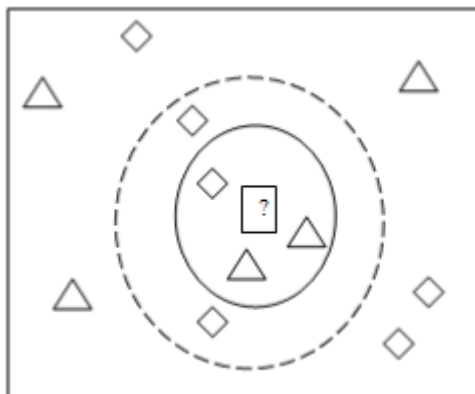


Slika 5.13: Redukcija vektorja značilnk smernih segmentov nelinearne mreže iz 256 dimenzij na 32 dimenzij s Fisherjevo diskriminanto.

5.5.2 K–NN

Klasifikator K–NN [3, 33, 34] je statistična klasifikacijska metoda, ki temelji na Bayesovi odločitveni teoriji. Učna faza klasifikatorja K–NN zajema shrambo vektorjev značilnk in razrednih oznak učnih vzorcev. V fazi klasifikacije se definira k , ki je konstanta. Klasifikator nato dodeli oznako, vsakemu testnemu vzorcu, dodeljeni znak je najpogostejši med k –učnimi vzorci, kar pomeni, da je posamezen vektor klasificiran glede na večinsko glasovanje njegovih sosedov. Slabost večinskega glasovanja je, da razredi s pogostimi vzorci prevladujejo pri napovedi oznake vektorja, saj se nagibajo k najbližjim sosedom zaradi večjega števila učnih vzorcev in ne zaradi dejanske bližine testnemu vzorcu. Ena izmed rešitev je pretehtanje klasifikacijske odločitve z upoštevanjem razdalje od testnega vzorca do k –najbližjih sosedov (slika 5.14).

Pri klasifikaciji K–NN navadno uporabljamo Evklidsko razdaljo, lahko pa tudi Hammingovo razdaljo, Manhattansko razdaljo ali razdaljo, ki jo dobimo z merjenjem kota med vektorji (angl. Cosine distance). Pri klasifikaciji značilnk



Slika 5.14: Neznani objekt (pravokotnik) je za $k = 3$ klasificiran kot trikotnik in za $k = 5$ kot objekt diamantne oblike.

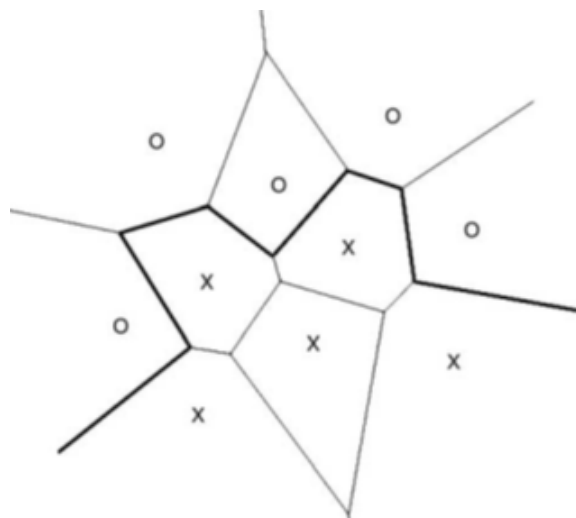
smo, tako kot pri MDC klasifikatorju, ugotovili, da smo dosegali boljšo klasifikacijsko natančnost z uporabo Manhattanske razdalje namesto Evklidske.

Izbira parametrov

Z izbiro optimalnega parametra k lahko povečamo klasifikacijsko natančnost. Ponavadi se z večjo vrednostjo k odpravi šume, vendar to ni nujno v vseh primerih. Optimalna vrednost k , se lahko izbere z navzkrižnim preverjanjem (angl. cross validation).

Ko je k enak 1, je k -NN pravilo reducirano na najbližjega sosedo oziroma 1-NN, ki klasificira vhodni vzorec x v razred, ki je najbližji učnemu vzorcu. Pri uporabi Evklidske distance se tako generira Voronojev diagram, kjer je vsaka celica definirana z učnim vzorcem in je ločena druga od druge s hiperploskvijo, ki je enako oddaljena od dveh vzorcev (slika 5.15). V našem primeru več razredne klasifikacije smo uporabili vrednost $k = 1$ [3].

Natančnost algoritma K -NN je lahko poslabšana s prisotnostjo nepomembnih in irelevantnih značilk, zato je potrebno v primeru večjih dimenzij vektorjev značilk reducirati nepomembne vrednosti. Pogost pristop k optimizaciji redukcije vektorjev značilk je evolucijski algoritem in izbira pravih informacij



Slika 5.15: Voronojev diagram.

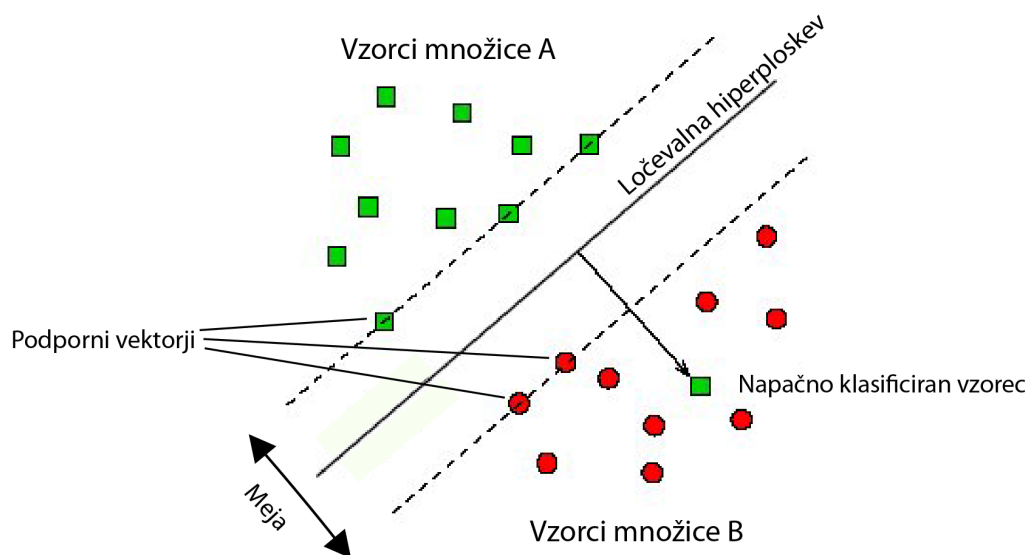
med testnim in učnimi vzorci [34]. Klasifikacijska natančnost K-NN je lahko izboljšana tudi, če so meritve razdalj izračunane s specializiranimi algoritmi kot je LMNN (angl. Large Margin Nearest Neighbour) ali analiza sosednjih komponent (angl. neighbour component analysis-NCA).

5.5.3 SVM

Metoda podpornih vektorjev [3] (SVM) je nova tehnika klasifikacije z uporabo hiperploskve. Njegova naloga je poiskati hiperravnino v n -dimenzionalnem prostoru, ki ločuje vzorce različnih razredov. Lega hiperravnine je odvisna le od njej najbližjih vektorjev, zato tem vektorjem rečemo podporni vektorji. Hiperravnine tudi ni mogoče »zviti«, zato lahko ločimo vzorce le takrat, ko so ti linearno ločljivi. To pa na resničnih primerih ni običajno, zatoorej, ko vektorji niso linearno ločljivi, uporabimo jedrni trik (angl. kernel trick), ki povečuje mejo med hiperravninami (slika 5.16) [3].

Metoda podpornih vektorjev je binarni linearni klasifikator, in je oblikovana kot utežena kombinacija jedrnih funkcij (angl. kernel function) na učnih vzorcih. Jedrna funkcija predstavlja produkt dveh vektorjev v linearnem (slika

5.17) oziroma nelinearnem prostoru značilk.



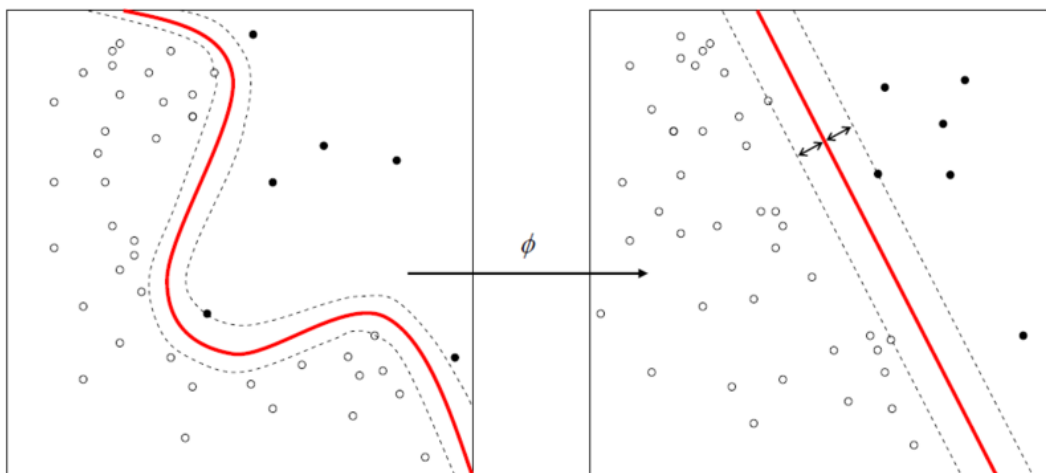
Slika 5.16: Hiperploskev binarnega SVM klasifikatorja z razdelitveno mejo.

RBF jedro

Glede na to, kako so vzorci razvrščeni v razrede, uporablja klasifikator SVM različne vrste jedr, to so linearno jedro, polinomsko jedro ter Gaussova RBF funkcija (angl. radial basis function). Uporaba različnih jedr nam prinese tudi različno klasifikacijsko natančnost. Najboljša izbira jedra je jedro RBF [35], ki nelinearno preslika vzorce v večdimenzionalni prostor. Kljub temu so primeri, ko RBF jedro ni primerno, na primer, ko je število značilk oziroma dimenzija vektorja značilk prevelika. V tem primeru lahko uporabimo tudi linearno jedro.

Najboljše tehnike SVM klasifikacije

Najboljše tehnike klasifikacije SVM [35] uporabljajo sledeč pristop:



Slika 5.17: SVM z jedrnim trikom (angl. kernel trick) pri nelinearni klasifikaciji (levo), SVM z linearnim jedrom oziroma linearni klasifikaciji (desno).

- reduciranje vrednosti značilnk na vrednosti od 0 do 1 ali od -1 do 1,
- uporaba RBF jedra,
- iskanje najboljšega C in parametra (v RBF jedru) z navzkrižnim preverjanjem.

Pri k -navzkrižnem preverjanju, razdelimo učno množico na k podmnožic enakih velikosti, nato zaporedno testiramo k -to podmnožico s preostalo $k-1$ podmnožico. Tako je vsak primer ek celotnega sklopa naučen predvidoma enkrat, torej je točnost navzkrižnega preverjanja odstotek podatkov, ki so pravilno razvrščeni.

Razširitev SVM klasifikatorja

Klasifikacija SVM je bila prvotno namenjena reševanju binarnih problemov, zato pri večrazredni klasifikaciji razgradimo večrazredni problem (slika 5.18) in s tem generiramo kombinacijo večih binarnih klasifikatorjev SVM [3]. Obstajata dve splošni obliki večrazrednega problema, »eden proti vsem« in »eden

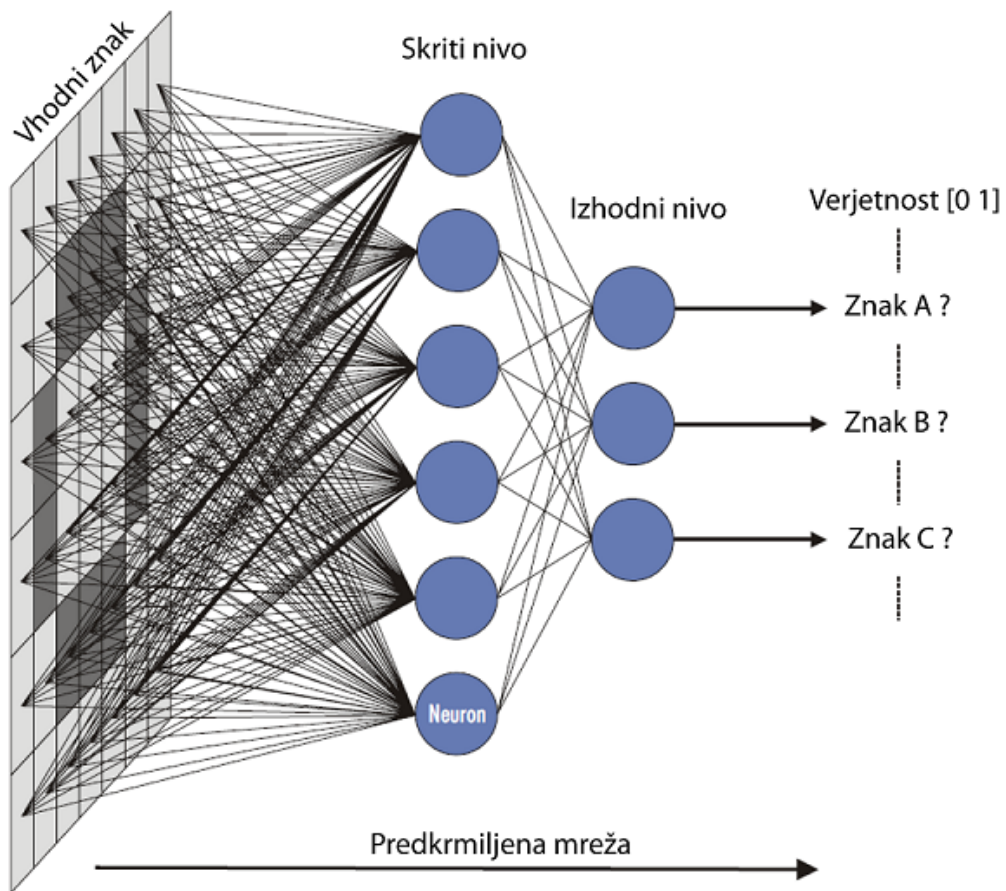
proti enemu«. Za M -razredni problem »eden proti vsem« uporabimo M -SVM klasifikatorje, kjer vsak klasifikator ločuje en razred od skupka preostalih razredov. Pri sistemu »1-proti-1«, pa je vsak klasifikator SVM uporabljen za razlikovanje med pari razredov. Pri izbiri večrazredne SVM klasifikacije smo izbrali metodo »eden proti vsem« [36].



Slika 5.18: Grafični prikaz trirazrednega SVM klasifikatorja z RBF jedrom.

5.5.4 ANN

Umetna nevronska mreža [3, 37] je vezje večih enostavnih procesnih enot t.i. nevronov. Njihovo delovanje je v osnovi podobno kompleksnim biološkim nevronom v možganih. Ključni element tega modela je nova struktura informacijsko procesnega sistema, ki je sestavljen iz večjega števila medsebojno povezanih nevronov, delujočih v simbiozi za reševanje specifičnih problemov. Način povezovanja nevronov razlikuje med seboj omrežne modele v predkrmljene mreže (slika 5.19), ponavljajoča se omrežja ali samoorganizirana omrežja.

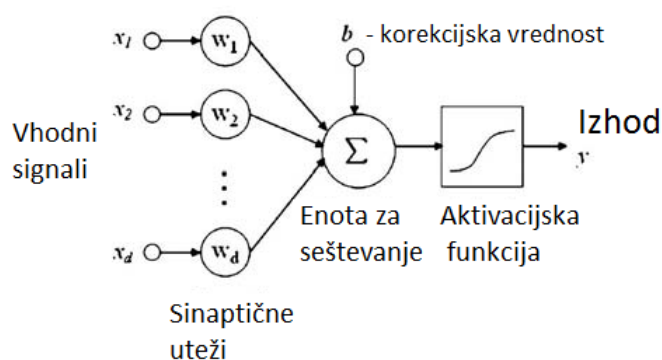


Slika 5.19: Predkrmiljena nevronska mreža.

Vsak nevron lahko sprejme signale ostalih nevronov v mreži in procesira vhodne podatke ter poda rezultate naprej ostalim nevronom v mreži. S pravo izbiro nevronov, povezav in uteži lahko optimiziramo nevronske mreže za specifičen problem. Nevronski model ima nabor uteženih povezav (kar ustreza sinapsam pri bioloških nevronih), ki manipulirajo s podatki pri izračunu klasifikacije, seštevalno enoto in aktivacijsko funkcijo (slika 5.20). Izhod seštevalne enote je utežena kombinacija vhodnih signalov oziroma značilnik, definirana z enačbo:

$$v = \sum_{i=1}^d w_i x_i + b \quad (5.10)$$

Kjer so posamezne sinaptične uteži, ki predstavljajo vhodne signale in je definiran kot korekcijska vrednost, ki omogoča premik aktivacijske funkcije v levo ali desno, kar je lahko ključnega pomena za uspešno učenje.



Slika 5.20: Zgradba posameznega nevrona umetne nevronske mreže.

Aktivacijska funkcija (slika 5.20) je lahko linearna ali nelinearna. Pogosto uporabljena nelinearna aktivacijska funkcija je sigmoida ali logistična funkcija, ki je bila uporabljena tudi v našem primeru klasifikacije z nevronskimi mrežami [3]. Formula nelinearne funkcije (sigmoide) je sledeča:

$$g(v) = \frac{1}{1+e^{-v}} \quad (5.11)$$

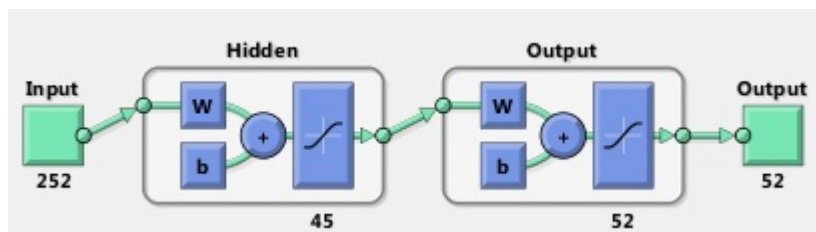
kjer v predstavlja izhod seštevalne enote.

Parametri

ANN je ponavadi definiran s tremi tipi parametrov:

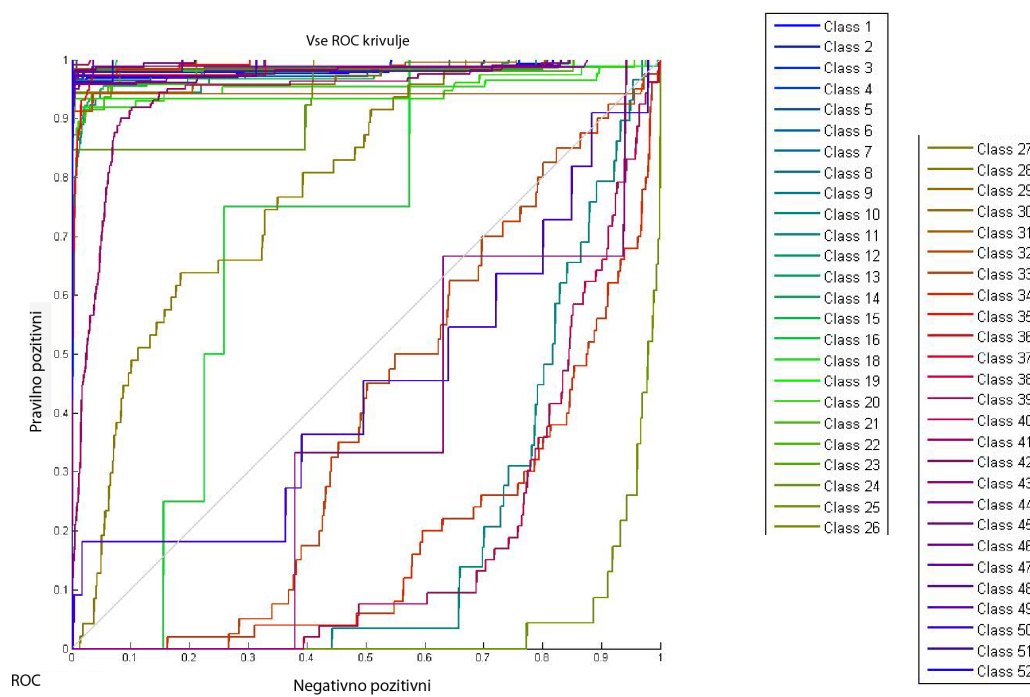
- z vzorcem medsebojne povezave med različnimi ravnmi nevronov,
- z učnim procesom za posodobitev uteži med povezavami,
- z aktivacijsko funkcijo, ki pretvori uteženi prispevek nevrona v njegovo izhodno aktivacijo.

Pri klasifikaciji ANN smo aplicirali dvonivojsko (skriti in izhodni nivo) predkrmiljeno nevronske mreže (njen graf je usmerjen aciklično). Pomemben parameter pri klasifikaciji ANN je tudi epoha (korak v učnem procesu nevronske mreže) (angl. epoch). Na splošno potrebujejo težji problemi večje število nevronov (slika 5.21).



Slika 5.21: Shema nevronske mreže, s 45 nevroni na skritem nivoju v primeru 256 dimenzionalnega vektorja.

Slika 5.22 prikazuje krivulje ROC klasifikatorja ANN pri značilkah smernih segmentov z nelinearno mrežo velikosti 9×7 in Otsu binarizacijo. Razredi so razvrščeni sledeče, od 1 do 26 so znaki od 'a' do 'z', od 27 do 52 so znaki od 'A' do 'Z'. ROC krivulja ilustrira zmogljivost klasifikacijskega sistema, tem bližje je posamezna krivulja levemu zgornjemu kotu, boljše je klasifikacijska natančnost razreda.



Slika 5.22: Krivulje ROC (angl. receiver operating characteristic) vseh razredov.

Poglavje 6

Rezultati

Celotno metodo smo implementirali v Matlabu z uporabo zunanjih knjižnic, ki so natančno opisane v poglavju 3.

Zbirka slik teksta ICDAR vsebuje tudi ločila, a smo se pri našem problemu omejili na znake angleške abecede, zato smo pri učenju in razpoznavi znakov izključili ločila. Zaradi majhne reprezentativne množice numeričnih znakov v zbirki slik teksta ICDAR in tudi CVL OCR DB le-te nismo vključili v učno in testno množico. Zbirka slik teksta CVL OCR DB vsebuje tudi nekaj šumnikov, a vseeno premalo za učinkovito klasifikacijo le-teh. Poleg števil in ločil, smo se izognili tudi šumnikom.

6.1 Predprocesiranje

Z uporabo predprocesiranja smo dobili večjo razločnost binarizirane slike, ki so bile primarno podvržene močni svetlobi ali pa minimalni svetlobi, kar je pripeljalo do povečane razpoznavnosti znaka s prostim očesom. Tu se je dobro odrezal filter Wiener za razmeglitev slike. Po drugi strani pa je ta proces malenkost popačil slike dobre kvalitete. Predprocesiranje nam je povprečno prineslo manjšo klasifikacijsko natančnost za 5%, kar je verjetno posledica tega, da predprocesiranje ni optimizirano za slike dobre razpoznavnosti, temveč

je bolj primerno za slike slabše kakovosti iz česar lahko sklepamo da postopek ni robusten in se bolje obnese na specifičnih primerih (slika 5.5).

6.2 Segmentacija

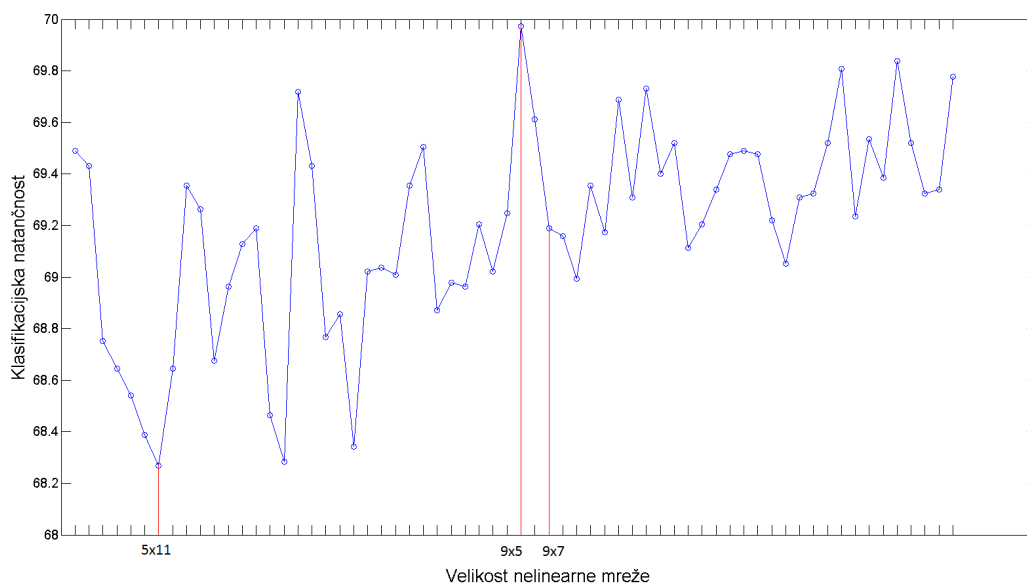
Binarizacija FCM uspešno obravnava oslABLJENE slike, zato ni bilo potrebe po predprocesiranju, čeprav je odstranjevanje šuma priporočljivo, saj nam pri binarizaciji pogosto ostanejo »otoki« slikovnih elementov, ki niso del znaka. Glavni problem, ki se je pojavil na nivoju binarizacije je bil ta, da se je slika pogosto invertirala in je bilo ozadje slike črne barve, znak pa je bil obarvan z belo. Temu problemu smo se izognili z izračunom deleža črne barve celotnega roba slike, ki je predstavljal 15% širine oziroma višine slike. V primeru večjega števila črnih slikovnih elementov kot belih smo sliko primerno invertirali, tako je bil znak črne barve in je ozadje bele barve, s čimer smo uspešno ekstrahirali značilke. Ta tehnika se zaradi svoje preprostosti ni najbolje obnesla.

V našem primeru se je izkazalo, da je primernejša tehnika binarizacije z Otsu upragovanjem, čeprav sta si bila v večini primerov algoritma Otsu in FCM zelo blizu glede natančnosti. Moramo poudariti tudi to, da je Otsu upragovanje računsko manj zahtevno, zato je bolj primerno apliciranju na mobilne platforme.

6.3 Pridobivanje značilk

Pridobitev značilk je bila razdeljena na tri različne metode in sicer značilke smernih segmentov nelinearne mreže, značilke SIFT ter aplikacijo Gaborjevega filtra na sivinsko sliko.

Slika 6.1, prikazuje, kako se klasifikacijska natančnost spreminja pri uporabi različnih vrednosti parametrov N in M , ki določata število horizontalnih in vertikalnih trakov v nelinearni mreži. Posamezne modele smo dobili s testiranjem 4-smernih segmentnih značilk z velikostjo mreže od 5×5 do 12×12 .



Slika 6.1: Graf klasifikacijske natančnosti v procentih glede na velikost nelinearne mreže značilnk smernih segmentov.

Testiranje je bilo opravljeno na 64. modelih s predprocesiranjem in binarizacijo Otsu, na zbirki ICDAR s hibridno učno množico CVL OCR DB. Kot je razvidno iz slike, dosega najboljše rezultate mreža v velikosti 9×5 (69.97%), iz česar lahko sklepamo, da je uporabljeno razmerje velikosti mreže učinkovitejše kot predlagano 9×7 (69.19%), saj je tudi razmerje latinskih znakov drugačno v primerjavi s korejskimi, pri čimer so latinski bolj pokončni in ožji kot korejski (slika 6.2). Najslabše rezultate je dosegla nelinearna mreža v velikost 5×11 in sicer 68.26%.

취 B

Slika 6.2: Primerjava korejskega in latinskega znaka.

Pri pridobitvi značilnk SIFT slike niso bile normalizirane, zato smo morali nad vektorji značilnk izvesti k-means grozdenje, kjer so SIFT deskriptorji kvantizirani (izgubna kompresija) v vizualne besede. To metodo imenujemo tudi torba vizualnih besed, ki kvantizira večdimenzionalno matriko značilnk v matriko z manj dimenzijami. V našem primeru se je izkazalo, da je podal najboljše rezultate kvantiziran vektor značilnk velikosti 1×128 za vsako sliko. Vrednosti kvantiziranih vektorjev značilnk so bile normalizirane na velikosti od 0 do 1. Posebna metoda značilnk SIFT, ki je bila uporabljena, je algoritem gosti SIFT, ki izračuna ključne točke glede na podan parameter (v našem primeru 2), ki pove za koliko slikovnih elementov so ključne točke oddaljene druga od druge. Klasifikacijska natančnost algoritma gosti SIFT je dosegala 79.45% pri K-NN klasifikaciji.

Normalizacija slik je bila tako izvedena tudi pri pridobivanju Gaborjevih značilnk, kjer so bile slike normalizirane na velikost 60×45 . Tako razmerje smo uporabili na podlagi razmerja latinskih znakov, ki so ožji. Gaborjeve značilke so dosegle najslabšo klasifikacijsko natančnost. Zaradi slabih rezultatov Gaborjevih značilnk smo ugotovili, da bi bilo potrebno reducirati vektorje značilnk z algoritmi kot so LDA in PCA, kjer bi odstranili redundantne vrednosti značilnk.

6.4 Klasifikacija

Pri K-NN klasifikaciji smo uporabili Manhattansko razdaljo, ki nam je prinesla boljše rezultate v primerjavi z Evklidsko. Z Evklidsko razdaljo smo tako dobili 74.30% (tabela 6.1) pri značilkah smernih segmentov nelinearne mreže velikosti 9×5 , z Manhattansko pa 77.96%, pri čemer je razlika skoraj 4%. S K-NN klasifikacijo smo povprečno pridobili najvišjo natančnost pri testiranju vseh zbirk slik.

Ravno tako smo Manhattansko razdaljo uporabili pri klasifikatorju MDC, kjer nam je v nasprotju z Evklidsko razdaljo prinesla višjo klasifikacijsko natančnost za približno 3%. Brez uporabe Fisherjeve diskriminante smo dobili le

70.33% klasifikacijsko natančnost z uporabo značilnk smernih segmentov nelinearne mreže, z uporabo Fisherjeve diskriminante pa se nam je ta povečala na 74.02% (tabela 6.1), kar pomeni, da je redukcija vektorja učinkovit postopek k izboljšavi klasifikacijske natančnosti.

Klasifikator SVM je bil po natančnosti v večini primerov zelo blizu klasifikatorju K-NN. Uporabljali smo predlagane parametre, kot je RBF (angl. radial basis function) jedro ter normalizacija značilnk na območje velikosti [0, 1]. Pri izbiri SVM večrazredne klasifikacije smo izbrali metodo »eden proti vsem«, ki je bila vključena v knjižnici LibSVM. Z omenjeno metodo smo dosegli največ 88.45% klasifikacijsko natančnost (slika 6.3, K-NN stolpec). Ker so nevronske mreže inicializirane z naključnimi utežmi, se rezultat po učenju vedno rahlo spremeni. Da se izognemo temu problemu, so naključne uteži statično nastavljene za pridobitev vedno enakega rezultata pri enaki učni in testni množici.

Parametri za klasifikator ANN so bili izbrani s preizkušanjem različnih epoh in števila nevronov na skritem nivoju. Ugotovili smo, da je optimalno število epoh 196 ter 51 nevronov na skritem nivoju, pri značilkah smernih segmentov nelinearne mreže v velikosti 9×5 . Ti parametri so prinesli 82.33% klasifikacijsko natančnost pri testiranju zbirke CVL OCR DB. Pri značilkah SIFT smo uporabili 35 nevronov na skritem nivoju in 196 epoh, manj nevronov je bilo potrebno zaradi manjših dimenzij vektorjev značilnk. Največje število nevronov smo morali uporabiti pri Gaborjevih značilkah in sicer 150, število epoh je ostalo enako.

6.5 Evalvacije celotnega sistema

Da bi se izognili preveliki kombinaciji metod, ki jo prinaša izbira različnih opcij na posameznih nivojih smo uporabili metodo Otsu upragovanja na nivoju segmentacije ter metodo smernih segmentov nelinearne mreže na nivoju

pridobivanja značilnk². Nato smo tako fiksirano metodo testirali pri različnih naborih klasifikatorjev.

Rezultati klasifikacijske natančnosti pri poganjanju različnih klasifikatorjev so prikazani v tabelah 6.1 do 6.4. Zgornja vrstica vsake celice rezultata tabele prikazuje klasifikacijsko natančnost pri uporabi invariantnosti glede na velikost znakov, s čimer dobimo 26 razredov. Spodnja vrstica pa prikazuje klasifikacijsko natančnost 52. razredov, ki jih ločujemo na male in velike znake. V primeru invariantnosti glede na velikost znaka se nismo omejevali na to ali je bil znak zaznan kot mala ali velika črka. V primeru, da je bil znak mali "o", je ta smatran kot pravilno klasificiran, četudi je bil klasificiran kot veliki "O".

| Hibridna zbirka (ICDAR >> CVL OCR DB) | | | | |
|---------------------------------------|------------------|------------------|------------------|------------------|
| | MDC | K-NN | SVM | ANN |
| DSF-MESH 9x5 | 74.02% 59.78% | 77.96% 63.96% | 76.20% 63.74% | 75.00% 61.50% |
| SIFT | 51.99% 42.27% | 57.77% 47.00% | 60.17% 49.75% | 57.03% 47.60% |
| Gabor | 37.33% 29.45% | 44.50% 35.46% | 41.88% 32.44% | 42.06% 33.09% |

Tabela 6.1: Tabela rezultatov hibridne zbirke, pri kateri so bili uporabljeni učni vzorci iz zbirke slik teksta ICDAR in testirani na zbirki slik teksta CVL OCR DB.

²Empirično smo ugotovili, da smo z Otsu upragovanjem ter uporabo štiri smernih segmentnih značilnk nelinearne mreže, pridobili optimalno klasifikacijsko natančnost na zbirki slik teksta CVL OCR DB [2].

| Hibridna zbirka (CVL OCR DB >> ICDAR) | | | | |
|---------------------------------------|--------|--------|--------|--------|
| | MDC | K-NN | SVM | ANN |
| DSF-mesh 9x5 | 69.09% | 71.89% | 71.24% | 66.60% |
| | 57.81% | 59.47% | 59.20% | 54.74% |
| SIFT | 43.65% | 48.50% | 50.99% | 43.58% |
| | 35.50% | 38.80% | 42.72% | 34.34% |
| Gabor | 25.73% | 32.68% | 27.47% | 29.28% |
| | 19.66% | 25.83% | 21.53% | 23.99% |

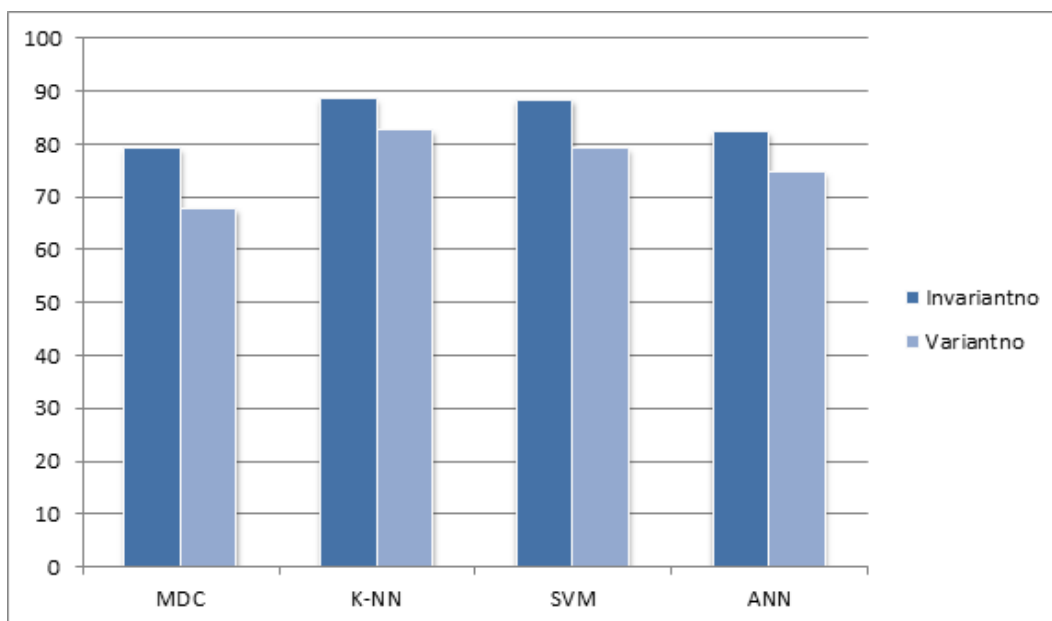
Tabela 6.2: Tabela rezultatov hibridne zbirke, pri kateri so bili uporabljeni učni vzorci zbirke slik teksta CVL OCR DB in testirani na zbirki slik teksta ICDAR.

| ICDAR | | | | |
|-----------------|--------|--------|--------|--------|
| | MDC | K-NN | SVM | ANN |
| DSF-mesh 9x5 | 38.17% | 74.68% | 37.12% | 65.54% |
| | 28.67% | 65.23% | 31.22% | 55.47% |
| SIFT | 28.39% | 61.11% | 31.24% | 57.73% |
| | 22.99% | 55.56% | 28.33% | 52.26% |
| Gabor | 20.43% | 45.33% | 21.56% | 35.06% |
| | 17.10% | 40.80% | 17.34% | 31.40% |

Tabela 6.3: Tabela rezultatov zbirke slik teksta ICDAR, ki že vsebuje ločeno testno in učno množico.

| CVL OCR DB | | | | |
|-----------------|------------------|------------------|------------------|------------------|
| | MDC | K-NN | SVM | ANN |
| DSF-mesh 9x5 | 79.24% 67.85% | 88.60% 82.73% | 88.45% 79.22% | 82.33% 74.71% |
| SIFT | 57.24% 50.37% | 79.45% 72.61% | 78.88% 70.29% | 64.79% 57.17% |
| Gabor | 41.88% 35.29% | 73.41% 65.41% | 71.93% 64.76% | 58.99% 51.52% |

Tabela 6.4: Tabela rezultatov zbirke slik teksta CVL OCR DB, pri kateri je bilo uporabljeno navzkrižno preverjanje (k-fold, za k=10).



Slika 6.3: Klasifikacijske natančnosti značilk 4-smernih segmentov nelinearne mreže velikosti 9x5, glede na invarianco oziroma varianco velikosti znaka.

6.6 Ugotovitve

Klasifikacija

Klasifikacijska natančnost predlagane metode je bila 85.97%, pri uporabi FCM binarizacije in značilk štiri smernih segmentnov nelinearne mreže velikosti 9×7 .

Naša najboljša kombinacija metod, Otsu upragovanje, značilke smernih segmentov nelinearne mreže velikosti 9×5 in K-NN klasifikacija nam je prinesla 88.6% klasifikacijsko natančnost (tabela 6.4–oranžno obarvan kvadrateg). Torej smo dosegli dokaj dobre rezultate. Povprečno se je izkazal bolje klasifikator K-NN kot SVM. Pri klasifikaciji je bila pomembna tudi hitrost, kjer je bil klasifikator K-NN najhitrejši.

Klasifikator ANN je z vidika povprečnih rezultatov primerljiv s klasifikatorjem SVM. Najslabše se je v večini primerov odrezal MDC klasifikator. Kot omenjeno so bili rezultati najvišji z uporabo metode značilke smernih segmentov nelinearne mreže, njej sledijo značilke SIFT, najslabše rezultate pa dosega Gaborjev filter. Tu moramo izpostaviti, da smo uporabljali zbirke slik teksta naravnih scen, ki vsebujejo znake, pri katerih je lahko svetloba neenakomerno porazdeljena, itd. in so na splošno slabše kakovosti. Z uporabo invariantnosti glede na velikost znakov (torej ne razlikujemo med velikimi in malimi črkami), se nam poveča natančnost za povprečno 10–15%. V večini primerov se mali znaki in veliki ne razlikujejo med seboj ali pa so zelo podobni, zato je opazno tako očitno izboljšanje.

Slika 2.9 prikazuje tabelo znakov, ki so kategorizirani v posamezne skupine. Te skupine pripadajo določenim nivojem, v katerih so si znaki med seboj zelo podobni. S slike je tudi razvidno, da so črke kot so »h« in »k« ter »c« in »o« zamenljive, kar se je najpogosteje izkazalo tudi v praksi in s tem poslabšalo rezultate.

Zbirke slik teksta

Hibridno zbirko smo ločili na dva dela, in sicer na kvalitetne vzorce slik zbirke ICDAR in zbirke CVL OCR DB. Boljše rezultate smo dosegali s hibridno učno množico ICDAR na testni množici CVL OCR DB, kot s hibridno učno množico CVL OCR DB na testni množici ICDAR. Razlika med zbirkama v klasifikacijski natančnosti v primeru hibridne učne množice je bila 6

Hibridna zbirka je povprečno prinesla najslabšo klasifikacijsko natančnost (71.89%) v primeru testiranja hibridnih učnih vzorcev zbirke CVL OCR DB na testnih vzorcih zbirke ICDAR (tabela 6.2), kar lahko pripišemo problemu zahtevnosti testnih slik zbirke ICDAR. Malenkost boljši rezultat smo dobili pri hibridni učni množici ICDAR, na kateri je bila testirana testna množica CVL OCR DB (77.96%). Iz rezultatov lahko razberemo, da zbirka CVL OCR DB vsebuje vzorce boljše kvalitete kot zbirka ICDAR.

Celotna hibridna zbirka je obsegala približno 3800 slik 52-ih razredov, kar je povprečno 36 učnih vzorcev za posamezen razred oziroma 72 vzorcev pri 26. razredih.

Na testni množici zbirke slik teksta ICDAR smo dosegli 74.68% klasifikacijsko natančnost z uporabo značilnik smernih segmentov nelinearne mreže velikosti 9×5 in K-NN klasifikatorja (tabela 6.3). Posamezen razred te učne množice je vseboval povprečno 115 vzorcev (52 razredov) oziroma 230 vzorcev (26 razredov).

Metoda Coates et al., 2010 [38] uporablja zbirko slik ICDAR, kjer dosega največ 81.4% natančnost in 85.5% pri invariantnosti velikosti znaka (62 razredov, vsebuje tudi numerične znake). Ta metoda je doslej na splošno dosegala najboljše rezultate tudi v primerjavi z drugimi metodami.

Zbirka slik teksta CVL OCR DB nam je prinesla največjo klasifikacijsko natančnost in sicer 88.6%. Zbirka je bila navzkrižno klasificirana v skupni velikosti 7600 slik, kjer smo uporabili 9/10 testnih primerov in 1/10 učnih, kar je 760 vzorcev za celotno učno množico oziroma 15 slik za vsakega izmed 52. razredov.

Na primeru najvišje klasifikacijske natančnosti 88.6%, smo ilustrativno prikazali uporabo binarizacije FCM, ki doseže 88.9% natančnost. V večini primerov se binarizacija FCM ni bolje odrezala kot Otsu in je bila povprečno slabša za 2%, a jo vendar v tem primeru prekaša za zanemarljivo vrednost v primerjavi z njeno računsko zahtevnostjo.

Ilustrativno smo prikazali tudi uporabo predprocesiranja, binarizacije Otsu in značilnik smernih segmentov nelinearne mreže velikosti 9×5 , kjer smo dosegli 83.84% klasifikacijsko natančnost s klasifikatorjem K-NN (k-fold, k=10). V primeru klasifikacije vzorceb brez predprocesiranja dobimo višji rezultat 88.6%, kjer je razlika 5%. Ugotovljeno je bilo da predprocesiranje ni bilo optimizirano za slike boljših ločljivosti in je tako nekatere slike popačilo. Predprocesiranje je prineslo dobre rezultate na primerih procesiranja slik slabe kakovosti (slika 5.3).

Poglavje 7

Zaključek

Potencialne izboljšave bi bile primerne predvsem na nivoju segmentacije oziroma binarizacije, kjer bi lahko z naprednejšimi metodami predprocesiranja dosegali boljše rezultate. Veliko slik bi lahko izboljšali z afinimi transformacijskimi tehnikami, kjer bi jim odpravili nagib in zamaknjenost, z odpravo tega problema bi dosegali večjo natančnost.

Na klasifikacijsko natančnost je vplivala tudi slaba reprezentativna testna množica, saj je ta morala vsebovati veliko različnih variacij posameznega znaka. V določenih primerih so se zaradi nekvalitetnih vzorcev učne množice rezultati poslabšali. Učno množico bi lahko generirali sintetično, tako da bi ustvarili skupek znakov istega razreda v različnih variacijah pisav, naklona, zameglitve, šuma, nagiba ter zamaknjenosti. Pri tem bi umetno reproducirali značilnosti slik naravnih scen in obenem ohranili raznolikost in razpoznavnost med posameznimi vzorci.

Potencialne izboljšave v našem delu (optična razpoznavna znakov v slikah naravnih scen) vidimo tudi kot nadgradnjo celotnega sistema v povezavi z implementacijo detekcije teksta. Koheziven sistem bi deloval na podlagi posnete fotografije in razpoznal celotno besedo, pri čimer bi bil lahko primeren tudi prevajalnik tujih besed. Omenjen sistem bi bil najprimernejši za implementacijo na mobilnih platformah.

Slike

| | | |
|-----|---|----|
| 1.1 | Primeri slik znakov (a) kompleksno ozadje, (b) kompleksen tekst, (c) nagnjen tekst, (d) različne barve teksta. | 2 |
| 2.1 | OCR-A (zgoraj), OCR-B (spodaj). | 4 |
| 2.2 | Metoda kukala. | 5 |
| 2.3 | Analiza struktur. | 6 |
| 2.4 | 4. smerna maska značilnk. | 7 |
| 2.5 | Osnovni diagram tipičnega sistema OCR. | 8 |
| 2.6 | (a) binarizacija znakov, (b) okostje znakov, (c) stanjšani znaki. | 8 |
| 2.7 | Transformiran znak "5". (a) original, (b) rotiran, (c) skaliran, (d) raztegnjen, (e) zamaknjen, (f) zravnjan, (g) zrcaljen. | 9 |
| 2.8 | Skrajno desno, razčlenjen znak. | 9 |
| 2.9 | Tabela kategorizacije znakov. | 11 |
| 4.1 | Izsek datoteke XML učne množice. | 16 |
| 4.2 | Primer znakov v zbirki ICDAR. | 16 |
| 4.3 | Prikaz pridobitve posameznih znakov iz besede. | 17 |
| 4.4 | Nekaj primerov znakov iz različnih kategorij podatkovne zbirke CVL. | 17 |
| 4.5 | Datotečna organizacija zbirke slik teksta CVL OCR DB. | 18 |
| 4.6 | Primeri slik hibridne zbirke slik teksta. | 19 |
| 5.1 | Diagram metode. | 21 |

| | | |
|------|---|----|
| 5.2 | Text excluding the matrix | 23 |
| 5.3 | Primerjava Wiener filtra z medianinim filtrom. Prvotni sliki je bil dodan šum z vrednostjo 0.025. (a) Originalna slika, (b) Wiener filter, (c) medianin filter. | 24 |
| 5.4 | Primer dobre slike, ki jo s predprocesiranjem malce popačimo. (a) Originalna slika, (b) binarizirana slika s FCM brez predprocesiranja, (c) binarizirana slika s FCM in predprocesiranjem. . . | 24 |
| 5.5 | Primer binariziranega znaka, z uporabo predprocesiranja in brez. (a) Originalna slika, (b) binarizirana slika s FCM brez predprocesiranja, (c) binarizirana slika s FCM in predprocesiranjem. . . | 24 |
| 5.6 | Primer FCM binarizacije znaka z neenakomerno osvetljenostjo. (a) originalna slika, (b) binarizirana slika | 26 |
| 5.7 | Primer binariziranega znaka z Otsu upragovanjem. | 27 |
| 5.8 | : (a) Nelinearna mreža velikosti 9×7 , (b) 4–smerna značilka, modra označuje horizontalne in vertikalne dolžine, vijolična pa predstavlja diagonalne dolžine (leve in desne). | 30 |
| 5.9 | Deskriptor ključne točke SIFT, dimenzije 1×128 | 31 |
| 5.10 | (a) SIFT s ključnimi točkami, (b) gosti SIFT. | 32 |
| 5.11 | : Štiri smerni Gaborjev filter. | 33 |
| 5.12 | Manhattanska razdalja med točkama p in q je označena z rdečo, modro in rumeno. Evklidska razdalja je predstavljena z zeleno linijo. | 36 |
| 5.13 | Redukcija vektorja značilk smernih segmentov nelinearne mreže iz 256 dimenzij na 32 dimenzij s Fisherjevo diskriminanto. . . . | 38 |
| 5.14 | Neznan objekt (pravokotnik) je za $k = 3$ klasificiran kot trikotnik in za $k = 5$ kot objekt diamantne oblike. | 39 |
| 5.15 | Voronovej diagram. | 40 |
| 5.16 | Hiperploskev binarnega SVM klasifikatorja z razdelitveno mejo. | 41 |

| | |
|---|----|
| 5.17 SVM z jedrnim trikom (angl. kernel trick) pri nelinearni klasifikaciji (levo), SVM z linearnim jedrom oziroma linearni klasifikaciji (desno). | 42 |
| 5.18 Grafični prikaz trirazrednega SVM klasifikatorja z RBF jedrom. | 43 |
| 5.19 Predkrmiljena nevronska mreža. | 44 |
| 5.20 Zgradba posameznega nevrona umetne nevronske mreže. | 45 |
| 5.21 Shema nevronske mreže, s 45 nevroni na skritem nivoju v primeru 256 dimenzionalnega vektorja. | 46 |
| 5.22 Krivulje ROC (angl. receiver operating characteristic) vseh razredov. | 47 |
| 6.1 Graf klasifikacijske natančnosti v procentih glede na velikost nelinearne mreže značilk smernih segmentov. | 50 |
| 6.2 Primerjava korejskega in latinskega znaka. | 50 |
| 6.3 Klasifikacijske natančnosti značilk 4-smernih segmentov nelinearne mreže velikosti 9×5 , glede na invarianco oziroma varianco velikosti znaka. | 55 |

Tabele

| | | |
|-----|--|----|
| 6.1 | Tabela rezultatov hibridne zbirke, pri kateri so bili uporabljeni učni vzorci iz zbirke slik teksta ICDAR in testirani na zbirki slik teksta CVL OCR DB. | 53 |
| 6.2 | Tabela rezultatov hibridne zbirke, pri kateri so bili uporabljeni učni vzorci zbirke slik teksta CVL OCR DB in testirani na zbirki slik teksta ICDAR. | 54 |
| 6.3 | Tabela rezultatov zbirke slik teksta ICDAR, ki že vsebuje ločeno testno in učno množico. | 54 |
| 6.4 | Tabela rezultatov zbirke slik teksta CVL OCR DB, pri kateri je bilo uporabljeno navzkrižno preverjanje (k-fold, za k=10). . . . | 55 |

Literatura

- [1] Simon M. Lucas, Alex Panaretos, Luis Sosa, Anthony Tang, Shirley Wong, Robert Young, Kazuki Ashida, Hiroki Nagai, Masayuki Okamoto, Hiroaki Yamamoto, Hidetoshi Miyao, Yunmin Zu, Wuwen Ou, Christian Wolf, Jean-Michel Jolion, Leon Todoran, Marcel Worring, Xiaofan Lin. ICDAR 2003 Robust Reading Competitions: Entries, Results and Future Directions. *International Journal of Document Analysis and Recognition*, 7(2-3):str. 105–122, 2005.
- [2] Andrej Ikica, Peter Peer. Cvl ocr db, an annotated image database of text in natural scenes, and its usability. *Informacije MIDE M*, 41(2):str. 150–154, 2011.
- [3] Mohammed Cheriet, Nawwaf Kharma, Cheng-lin Liu, Ching Suen. *Character Recognition Systems: A Guide for Students and Practitioners*. Wiley-Interscience, 2007.
- [4] Line Eikvil. Ocr - optical character recognition. Technical report no.876, 1993.
- [5] Shunji Mori, Ching Y. Suen, Kazuhiko Yamamoto. Historical review of ocr research and development. *Proceedings of the IEEE*, 80(7):str. 1029–1058, 1992.
- [6] Cheng-Lin Liu, Hiromichi Fujisawa. Classification and learning methods for character recognition: Advances and remaining problems. In

- Machine Learning in Document Analysis and Recognition*. Springer, str. 139–161. 2008.
- [7] Oivind Due Trier, Anil K. Jain, Torfinn Taxt. Feature extraction methods for character recognition—a survey. *Pattern Recognition*, str. 641–662, 1996.
- [8] Ray Smith. An overview of the Tesseract OCR engine. In *Proceedings of the Ninth International Conference on Document Analysis and Recognition, ICDAR '07*, zbornik 2, str. 629–633, Washington, DC, USA, 2007.
- [9] Jing Zhang, Xilin Chen, Andreas Hanneman, Jie Yang, Alex Waibel. A robust approach for recognition of text embedded in natural scenes. In *Int. Conf. on Pattern Recognition*, str. 204–207, 2002.
- [10] Piyanuch Silapachote, Jerod Weinman, Allen Hanson, Marwan A. Mattar, Richard Weiss. Automatic sign detection and recognition in natural scenes. In *IEEE Workshop on Computer Vision Applications for the Visually Impaired*, str. 27, 2005.
- [11] Shijian Lu, Chew Lim Tan. Camera text recognition based on perspective invariants. In *International Conference on Pattern Recognition*, zbornik 2, str. 1042–1045, 2006.
- [12] Onur Tekdas, Nikhil Karnad. Recognizing characters in natural scenes a feature study. Department of Computer Science and Engineering, University of Minnesota, USA, 2009.
- [13] (2012) Matching with Shape Contexts. Dostopno na:
http://www.eecs.berkeley.edu/Research/Projects/CS/vision/shape/sc_digits.html.
- [14] Teofilo E. deCampos, Bodla Rakesh Babu, Manik Varma. Character recognition in natural images. In *Proceedings of the International*

Conference on Computer Vision Theory and Applications, Lisbon, Portugal, 2009.

- [15] (2012) Matlab. Dostopno na:
<http://en.wikipedia.org/wiki/MATLAB>.
- [16] (2012) Statistics Toolbox. Dostopno na:
<http://www.mathworks.com/products/statistics>.
- [17] (2012) Image Processing Toolbox. Dostopno na:
<http://www.mathworks.com/products/image>.
- [18] (2012) Neural Network Toolbox. Dostopno na:
[http://www.mathworks.com/products/neural network](http://www.mathworks.com/products/neural_network).
- [19] (2012) LibSVM. Dostopno na:
<http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- [20] (2012) VLFeat Library. Dostopno na:
<http://www.vlfeat.org>.
- [21] Jonghyun Park, Guesang Lee, Euichul Kim, Junsik Lim, Soohyung Kim, Hyungjeong Yang, Myunghun Lee, Seongtaek Hwang. Automatic detection and recognition of korean text in outdoor signboard images. *Pattern Recognition Letters*, 12, (31): str. 1728–1739, 2010.
- [22] (2012) Gabor filter. Dostopno na:
http://en.wikipedia.org/wiki/Gabor_filter.
- [23] (2012) Wiener Filtering and Image Processing. Dostopno na:
<http://www.clear.rice.edu/elec431/projects95/lords/wiener>.
- [24] (2012) Median filter. Dostopno na:
http://en.wikipedia.org/wiki/Median_filter.

- [25] (2012) Image binarization. Dostopno na:
<http://researchweb.watson.ibm.com/haifa/projects/image/glt/binar>.
- [26] (2012) Otsu's method. Dostopno na:
http://en.wikipedia.org/wiki/Otsu's_method.
- [27] Kamarul Hawari Ghazali, Mohd. Marzuki Mustafa, Aini Hussain. Feature extraction technique using SIFT keypoint descriptors. *In Proceedings of the International Conference on Electrical Engineering and Informatics*, Electronic and Systems Engineering Faculty of Engineering, Universiti Kebangsaan Malaysia, Malaysia, 2007.
- [28] (2012) Scale invariant feature transform. Dostopno na:
http://en.wikipedia.org/wiki/Scale-invariant_feature_transform.
- [29] (2012) Statistical classification. Dostopno na:
http://en.wikipedia.org/wiki/Statistical_classification.
- [30] Alok Sharma, Kuldip K. Paliwal, Godfrey C. Onwubolu. Class-dependent PCA, MDC and LDA: A combined classifier for pattern classification. *Pattern Recognition Letters*, zbornik 7, (39): str. 1215–1229, 2006.
- [31] (2012) Euclidean distance. Dostopno na:
http://en.wikipedia.org/wiki/Euclidean_distance.
- [32] (2012) Manhattan distance. Dostopno na:
http://en.wikipedia.org/wiki/Manhattan_distance.
- [33] Gurpreet Singh Lehal, Chandan Singh. Handwritten Gurmukhi character recognition using statistical and background directional distribution features. *International Journal on Computer Science and Engineering*, str. 2–12, 1999.

- [34] (2012) K nearest neighbor algorithm. Dostopno na:
http://en.wikipedia.org/wiki/K-nearest_neighbor_algorithm.
- [35] Chih-Wei Hsu, Chih-Chung Chang, Chih-Jen Lin. A practical guide to support vector classification. Technical report, Department of Computer Science and Information Engineering, National Taiwan University, Taiwan, 2003.
- [36] Jonathan Milgram, Mohamed Cheriet, Robert Sabourin. "One Against One" or "One Against All": Which One is Better for Handwriting Recognition with SVMs? In *International Workshop on Frontiers in Handwriting Recognition*, La Baule, France: Suvisoft, 2006.
- [37] (2012) Neural networks. Dostopno na:
http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.
- [38] Adam Coates, Blake Carpenter, Carl Case, Sanjeev Satheesh, Bipin Suresh, Tao Wang, David J. Wu, Andrew Y. Ng. Text detection and character recognition in scene images with unsupervised feature learning. In *International Conference on Document Analysis and Recognition*, str. 440–445, 2011.