

UNIVERZA V LJUBLJANI  
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Aleš Bernik

# **Obogatena razglednica**

DIPLOMSKO DELO  
NA UNIVERZITETNEM ŠTUDIJU

Mentor: doc. dr. Matija Marolt

Ljubljana, 2012



Št. naloge: 01835/2012

Datum: 03.04.2012

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

Kandidat: **ALEŠ BERNIK**

Naslov: **OBOGATENA RAZGLEDNICA  
AUGMENTED POSTCARD**

Vrsta naloge: Diplomsko delo univerzitetnega študija

Tematika naloge:

V diplomski nalogi preučite sisteme za obogateno resničnost, načine njihovega delovanja in programske knjižnice, ki omogočajo enostavno izdelavo aplikacij za obogateno resničnost.

Izdelajte spletno aplikacijo obogatena razglednica, ki s pomočjo oznak omogoča obogatitev razglednice s 3D vsebino. Preučite tudi optimalno izbiro in lokacijo oznak na razglednici za robustno delovanje sistema.

Mentor:

  
doc. dr. Matija Marolt

Dekan:

  
prof. dr. Nikolaj Zimic



# IZJAVA O AVTORSTVU

## diplomskega dela

Spodaj podpisani Aleš Bernik,  
z vpisno številko 63010008,

sem avtor diplomskega dela z naslovom:

Obogatena razglednica

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal samostojno pod mentorstvom  
doc. dr. Matija Marolta
- 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 12.6.2012

Podpis avtorja:

## **Zahvala**

Za mentorstvo pri izdelavi diplomske naloge se zahvaljujem doc. dr. Matiji Maroltu.  
Zahvala gre tudi vsem, ki so me spodbujali in podpirali v času študija.

# Kazalo

<b>Povzetek</b> .....	<b>1</b>
<b>Abstract</b> .....	<b>2</b>
<b>1. Obogatena resničnost</b> .....	<b>3</b>
1.1 Uvod.....	3
1.2 Definicija .....	3
1.3 Aplikacije.....	4
1.3.1 Arhitektura.....	4
1.3.2 Medicina .....	5
1.3.3 Industrija.....	6
1.3.4 Zabava .....	6
1.4 Strojna oprema.....	7
<b>2. Značilnosti sistemov obogatene resničnosti</b> .....	<b>8</b>
2.1 Bogatenje (Augmentation).....	8
2.2 Tehnologija prikazovanja.....	9
<b>3. Obogatena resničnost danes</b> .....	<b>13</b>
3.1 Raziskovalni problemi .....	13
3.1.1 Registracija .....	13
3.1.2 Sledenje .....	14
3.2 Pristopi sledenja.....	15
3.2.1 Sledenje s pomočjo računalniškega vida .....	15
3.2.2 Sledenje s pomočjo senzorja.....	16
3.2.3 Hibridno sledenje.....	16
3.3 Obogatena resničnost z uporabo oznak.....	17
3.3.1 Odkrivanje oznake .....	17
3.3.2 Določanje položaja .....	18
3.3.3 Prepoznavanje oznake .....	18
3.3.4 Sistemi oznak.....	19
3.3.5 Sledenje oznaki.....	23
<b>4. Sistemi oznak</b> .....	<b>25</b>
4.1 Primerjava .....	25

4.2	Programska oprema.....	27
4.2.1	ARToolKit .....	27
4.2.2	ARTag.....	28
4.2.3	ARToolKitPlus.....	28
4.2.4	Studierstube Tracker .....	29
4.2.5	ALVAR.....	29
4.2.6	ArUco.....	30
4.2.7	NyARToolKit.....	31
4.2.8	FLARToolKit.....	31
4.2.9	FLARManager .....	31
4.2.10	SLARToolKit.....	32
4.3	Obogatena resničnost na spletu .....	32
<b>5.</b>	<b>Aplikacija obogatena razglednica.....</b>	<b>33</b>
5.1	Izbira knjižnice .....	34
5.2	Kakšna je lahko oznaka? .....	34
5.3	Ustvarjanje novih oznak .....	35
5.4	Izdelava vzorčne datoteke novoustvarjene oznake .....	36
5.5	Velikost oznake .....	36
5.6	Izdelava razglednice .....	38
5.6.1	Oznaka na prvi strani .....	38
5.6.2	Oznaka na zadnji strani .....	41
5.6.3	Povzetek.....	42
5.7	Izdelava internetne aplikacije .....	43
<b>6.</b>	<b>Zaključek .....</b>	<b>44</b>
	<b>Seznam slik .....</b>	<b>45</b>
	<b>Seznam tabel.....</b>	<b>46</b>
	<b>Literatura.....</b>	<b>47</b>

## **Seznam uporabljenih kratic in simbolov**

- CRC (Cyclical redundancy check) – ciklično preverjanje redundance
- FEC (Forward Error Correction) - naprejšnje popravljanje napak
- GPS (Global Positioning System) – sistem globalnega določanja položaja
- HMD (Head mounted display) – naglavni prikazovalnik
- HOM (Hoffman marker system) – hoffmanov sistem oznak
- IGD (Institut Graphische Datenverarbeitung marker system) – sistem oznak inštituta za računalniško grafiko
- SCR (Siemens Corporate Research marker system) – sistem oznak podjetja Siemens





## **Povzetek**

Cilj diplomske naloge je preučitev tehnologije obogatena resničnost, ki omogoča mešanje realnih in navideznih elementov. Obogatena resničnost je dokaj nova tehnologija, ki postaja vse bolj razširjena, zahvaljujoč tudi dokaj zmerni ceni pametnih telefonov.

Predstavljeni bodo tipi obogatene resničnosti, potrebna tehnologija in njihove prednosti in slabosti, njena trenutna uporaba v aplikacijah ter programska oprema za izgradnjo aplikacij obogatene resničnosti.

V diplomski nalogi smo se osredotočili na obogateno resničnost z uporabo oznak. Oznaka je umetna struktura, ki se uporablja za sledenje in pravilno poravnavo navideznih objektov z realnimi. Tako je v zadnjem poglavju prikazana izdelava preproste spletne aplikacije obogatena razglednica. Omenjena aplikacija na razglednici z oznako prikaže 3D model objekta, ki je za njo značilen. Preučili smo tudi, kako naj sama razglednica izgleda, kakšne vrste naj bo oznaka in kako velika je lahko in kam na razglednici jo lahko vstavimo.

## **Ključne besede:**

Obogatena resničnost, oznaka, sledenje, registracija, 3D model

## **Abstract**

The aim of this thesis is the examination of augmented reality technology, which allows us mixing real and virtual elements. Augmented reality is a relatively new technology which is becoming more widespread, thanks to a fairly reasonable price of smart phones.

Here we presents the types of augmented reality, the necessary technology and their advantages and disadvantages, its current use in applications, and software for building augmented reality applications. The thesis is mainly focused on augmented reality using the markers. Marker is an artificial structure that is used for tracking and proper registration of virtual objects. In the last chapter we make simple web application augmented postcard. This application on the postcard with the marker shows some object which is typical for postcard.

We also examine how the postcard can look like, what kind of marker can be used, how big can be and where we can put it.

## **Key words:**

Augmented reality, marker, tracking, registration, 3D model

# 1. Obogatena resničnost

## 1.1 Uvod

Živimo v svetu, kjer je tretja dimenzija vedno bolj prisotna, tako v filmski industriji kot tudi v računalništvu. Dokaj nova tehnologija se imenuje obogatena resničnost. Gre za neko dodatno 2D ali 3D informacijo, ki je uporabniku posredovana v 3D prostoru. V zadnjih časih je opazen porast aplikacij obogatene resničnosti za mobilne telefone, vendar mnogo ljudi še ni slišalo za ta izraz. Cilj diplomske naloge je predstavitev področja obogatene resničnosti. Tako bo v prvem poglavju predstavljena definicija obogatene resničnosti ter njena uporaba v aplikacijah.

V drugem poglavju opisujemo, kaj predstavlja izraz bogatenje. Predstavljena sta tudi dva načina prikazovanja obogatene resničnosti skupaj s svojimi prednostmi in slabosti.

V tretjem poglavju sledi opis raziskovalnih problemov na tem področju. Opisani so pristopi sledenja, ki se pojavljajo v obogateni resničnosti. Osredotočili smo se predvsem na pristop z uporabo oznak, ki so tudi predstavljene in jih v četrtem poglavju tudi primerjamo. Predstavljena so tudi orodja za izdelavo aplikacij obogatene resničnosti.

V zadnjem poglavju je opisana izdelava preproste spletne aplikacije obogatena razglednica, kjer razglednico določenega objekta dopolnimo s 3D modelom objekta.

## 1.2 Definicija

Na kratko, izraz obogatena resničnost predstavlja kombinacijo realnega sveta in računalniško obdelanih učinkov. Pravzaprav ni uradne definicije izraza obogatena resničnost.

Nekateri raziskovalci so obogateno resničnost definirali na način, ki zahteva uporabo naglavnega prikazovalnika (HMD).

Na tem mestu bom uporabil definicijo po raziskavi Ronalda Azuma [1]. Glede na to mora sistem obogatene resničnosti imeti naslednje značilnosti:

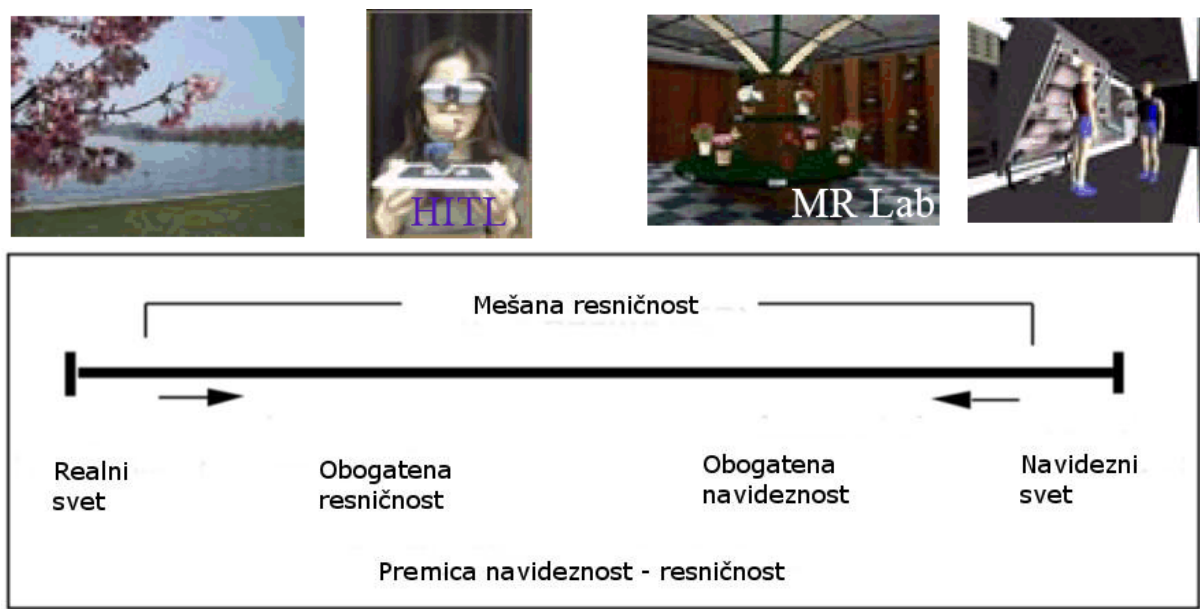
1. Mešanje realnosti in navideznosti
2. Interaktivnost v realnem času
3. Registracijo v 3D

Prva točka govori, da mora končni rezultat vsebovati navidezne objekte in resničnost samo. Zaradi druge točke filmov, kot je Avatar, ne uvrščamo med obogateno resničnost, ker navidezni objekti, mešani z realnim svetom, niso interaktivni. Zaradi točke tri tudi popularne pasice na televiziji ne spadajo v to kategorijo.

Slika 1.1 prikazuje premico navideznost – resničnost, ki si jo je zamislil Miligram [2].

Na enem koncu je realni svet, na drugem pa navidezni svet. Med obema skrajnostma je mešana resničnost, ki vsebuje obogateno resničnost in obogateno navideznost.

Pri prvi gre za realni svet, ki vsebuje navidezne objekte. Pri obogateni navideznosti pa je ravno obratno, navidezni svet vsebuje resnične objekte, npr. navidezna soba, v kateri so resnične osebe.



Slika 1.1: Premica navideznost – resničnost

### 1.3 Aplikacije

Obogatena resničnost se uporablja v aplikacijah na različnih področjih, npr. v arhitekturi, medicini, industriji, športu, vojski, za oglaševanje in na zabavnem področju.

#### 1.3.1 Arhitektura

Primer uporabe obogatene resničnosti je v opremljanju bivalnih prostorov. Aplikacija nam omogoča, da vidimo, kako določeno pohištvo izgleda v stanovanju, še preden ga kupimo.

Namesto da bi premikali pohištvo, položimo oznake na želeno mesto, da vidimo, kako izgleda nova preureditev. Slika 1.2 prikazuje primer umestitve 3D modela mize v sobo z uporabo obogatene resničnosti.



*Slika 1.2: Navidezna miza v resničnem svetu*

### **1.3.2 Medicina**

V medicini je veliko področij, kjer se lahko uporablja obogatena resničnost. Danes zelo razširjena je računalniška tomografija. To je neka vrsta medicinskega slikanja, ki ustvari tridimenzionalno sliko notranjosti telesa. Obogatena resničnost omogoča prikaz 3D modelov na mesta telesa, kamor spadajo, kot to prikazuje Slika 1.3.



*Slika 1.3: Noga, prekrita s 3D modelom kosti*

### 1.3.3 Industrija

Volkswagen je razvil aplikacijo za planiranje proizvodnje in oblikovanja. Aplikacija je že bila uporabljena za planiranje proizvodnje v različnih proizvodnih okoljih. Slika 1.4 prikazuje uporabo aplikacije.

Drug primer aplikacije v avtomobilskem sektorju bi bil označevanje delov avtomobila in dajanje navodil servisierju.



*Slika 1.4: Primer uporabe Volkswagnove aplikacije*

### 1.3.4 Zabava

Obogatena resničnost se pojavlja tudi na zabavnem področju, kot so na primer igre.

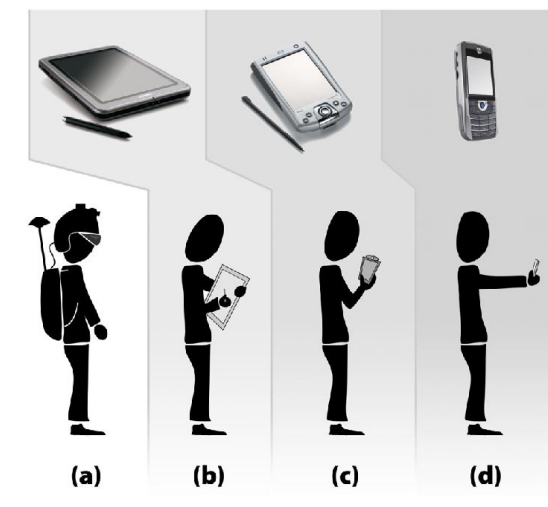
Zaradi hitrega razvoja mobilnih naprav in njihove priročnosti je opazen velik razvoj aplikacij, ki so namenjene tem napravam.

## 1.4 Strojna oprema

Za delujoč sistem obogatene resničnosti potrebujemo samo tri stvari: kamero za zajem realnega sveta, zaslon za prikaz končnega stanja ter eno izmed naprav z računsko močjo.

Na trgu obstaja veliko naprav, ki zadostujejo tem zahtevam.

Slika 1.5 prikazuje različne oblike teh naprav. Od leve proti desni: nahrbtnni računalnik z naglavnim zaslonom, tablični računalnik, PDA, pametni telefon.



Slika 1.5: Naprave za obogateno resničnost

## **2. Značilnosti sistemov obogatene resničnosti**

To poglavje obravnava značilnosti sistema obogatene resničnosti in težave, na katere lahko naletimo pri izgradnji takega sistema. Podpoglavje 2.1 opisuje glavne značilnosti obogatene resničnosti, podpoglavje 2.2 pa tehnologije prikazovanja in njihove prednosti ter pomanjkljivosti.

### **2.1 Bogatenje (*Augmentation*)**

Poleg dodajanja objektov v realno okolje, ima uporabnik tudi možnost, da jih odstrani. Trenutno je to področje obogatene resničnosti osredotočeno na dodajanje objektov v realno okolje. Čeprav, grafično prekrivanje je lahko uporabljeno za odstranjevanje ali prekrivanje delov realnega okolja. Na primer da odstranimo mizo iz sobe, moramo pripraviti sliko sten in tal, ki se nahaja za mizo in jo narisati čez njo. V filmih se ta tehnika že uporablja, toda narediti to interaktivno v sistemih obogatene resničnosti je veliko težje, kljub temu da ta odstranitev ni nujno fotorealistična.

Sistem obogatene resničnosti bi lahko vključeval vsa čutila, ne samo vid. Do zdaj so bili raziskovalci osredotočeni na mešanje realnih in navideznih slik. Sistem pa je lahko razširjen tako, da vključuje tudi sluh. Uporabnik bi nosil slušalke in mikrofona. Slušalke bi dodale usmerjen sintetični 3D zvok, medtem ko bi mikrofona zaznal zvok iz okolja. To bi sistemu dalo možnost, da zamenja ali izniči realni zvok. Čeprav to ni lahka naloga, je mogoča.

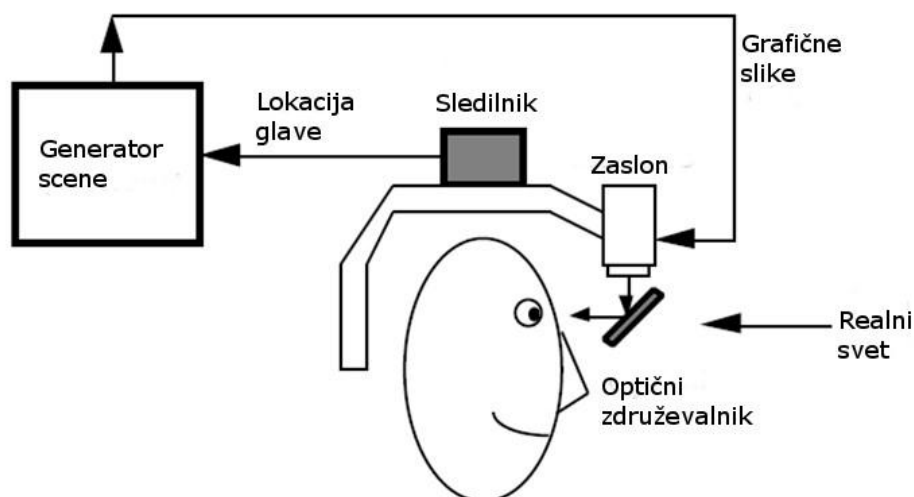
Drug primer je tip. Rokavice, ki omogočajo otipno informacijo, lahko simulirajo resnične sile v okolju. Uporabnik lahko na primer z roko drsi po resnični mizi. Navidezno simuliranje trde površine je zelo težko, vendar zelo enostavno v realnosti. Tako lahko s pomočjo rokavice dodamo občutek, da je miza na nekaterih mestih bolj groba. Ta sposobnost je lahko uporabna v določenih aplikacijah, kot na primer v primeru, kjer želimo zagotoviti dodaten namig o lokaciji navideznega objekta na resnični mizi.



## 2.2 Tehnologija prikazovanja

Osnovna odločitev pri izgradnji sistema obogatene resničnosti je, kako zagotoviti mešanje navideznega in realnega sveta. Dve osnovni možnosti sta na voljo: optična in video tehnologija. Vsaka ima določene prednosti in slabosti. To poglavje primerja obe tehnologiji in predlaga določene kompromise.

Naprava, ki omogoča prikazovanje realnega in navideznega sveta se imenuje naglavni prikazovalnik. Obstajata dve vrsti: optični naglavni prikazovalnik ter video naglavni prikazovalnik. Optični združevalnik, ki je del prikazovalnika, je delno prepusten, tako da uporabnik skozenj vidi realni svet, je pa tudi delno odsevan, kar omogoča uporabniku, da vidi odboj navidezne slike, katera prihaja iz zaslona. Slika 2.1 prikazuje konceptualni diagram optičnega naglavnega prikazovalnika. Optični združevalniki ponavadi zmanjšajo količino svetlobe, ki prihaja iz realnega sveta. Ker združevalnik deluje kot polovično posrebrnjeno ogledalo, spusti skozenj samo del svetlobe iz realnega sveta, ter tako omogoča odboj svetlobe iz monitorja v oči uporabnika. V primeru izklopa napajanja, optični prikazovalnik deluje kot sončna očala.



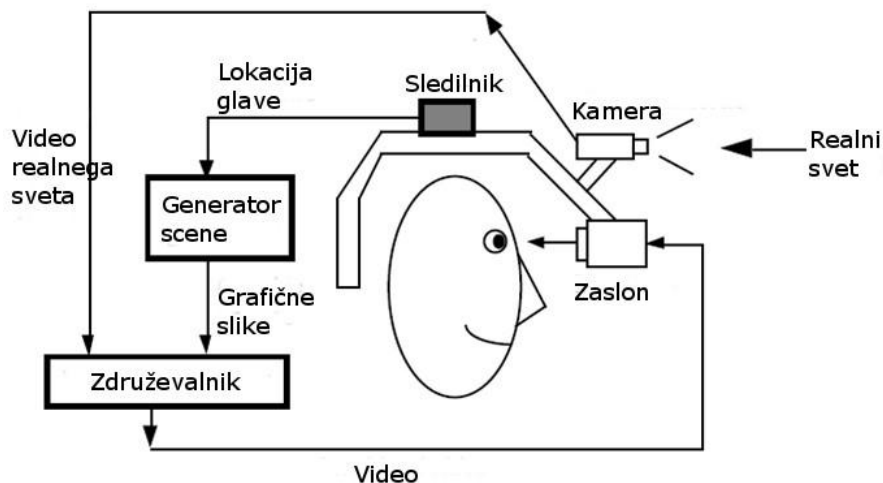
Slika 2.1: Konceptualni diagram optičnega naglavnega prikazovalnika



Slika 2.2: Optična naglavna prikazovalnika, ki ju je naredil Hughes Electronics

Naglavni video prikazovalnik vsebuje eno ali dve video kameri, katere pridobivajo uporabnikov pogled na svet. Video iz teh kamer je nato združen s slikami, ki jih generira generator scene. Rezultat je viden na zaslonu pred uporabnikovimi očmi.

Slika 2.3 prikazuje konceptualni diagram naglavnega video prikazovalnika. Video združevanje je lahko doseženo na več načinov. Lažji pristop uporablja tehniko zelenega platna. Ozadje računalniških slik je nastavljeno na določeno barvo, recimo zeleno, katere ne uporablja noben izmed navideznih objektov. Korak združevanja nato zamenja vsa območja zelene barve s pripadajočimi deli realnega sveta. Pri tem pride do prekrivanja realnega sveta z navideznimi objekti. Bolj napredna metoda uporablja globinsko informacijo. Če ima sistem za vsak piksel realnega sveta informacijo o globini, lahko združi realne in navidezne slike s primerjavo te informacije. To omogoča prekrivanje navideznih objektov z realnimi in obratno.

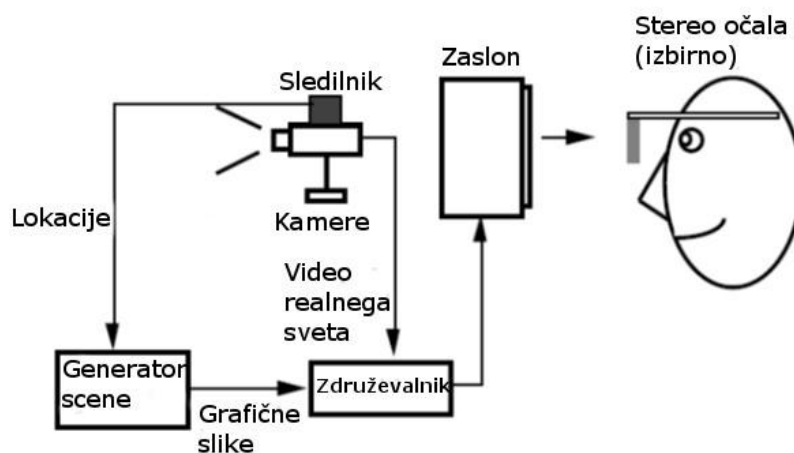


Slika 2.3: Konceptualni diagram video naglavnega prikazovalnika



Slika 2.4: Video naglavni prikazovalnik

Sistem obogatene resničnosti je lahko zgrajen tudi na osnovi monitorja. Slika 2.5 prikazuje omenjeni koncept. V tem primeru ena ali dve kameri opazujeta okolje. Kameri sta lahko statični ali mobilni. V mobilnem primeru lahko kamero premikamo s pomočjo robota, pri čemer sledimo njegovi lokaciji. Združevanje videa realnega sveta in grafičnih slik poteka kot pri video metodi, le da je slika prikazana na zaslonu pred uporabnikom. Uporabnik ne nosi nobene naprave. V primeru prikaza slike v stereo načinu, mora uporabnik uporabiti stereo očala.



Slika 2.5: Konceptualni diagram sistema obogatene resničnosti na osnovi monitorja

V ostalem delu poglavja sledi primerjava prednosti in slabosti optičnega in video pristopa. Optični pristop ima naslednje prednosti pred video pristopom.

**Enostavnost:** Optično mešanje je enostavnejše in cenejše kot video mešanje. Optični pristop ima samo en video tok, za katerega mora skrbeti, to so grafične slike. Realni svet je viden neposredno skozi združevalnik. Zakasnitev je povprečno nekaj nanosekund. Video mešanje

ima opravka z dvema video tokovoma, eden so realne, drugi pa navidezne slike. Zakasnitev takega sistema je nekaj deset milisekund. Oba tokova morata biti sinhronizirana.

**Ločljivost:** Video pristop omejuje ločljivost na ločljivost zaslona naprave. Optični pristop grafične slike tudi prikazuje v ločljivosti, ki jo ima zaslon, vendar je uporabnikov pogled na realni svet nespremenjen.

**Varnost:** V primeru prekinitve napajanja pri video pristopu postane uporabnik dobesedno slep. Pri optičnem pa je realni svet še vedno viden, uporabnik pa se počuti kot da bi imel pred očmi sončna očala.

**Ni očesnega zamika:** Pri video pristopu je uporabnikov pogled na svet zagotovljen z video kamerami. To pomeni, da so oči uporabnika tam kjer so kamere. V večini naprav so kamere na različnem mestu kot oči. To povzroča razliko med tem kar uporabnik vidi in tem kar pričakuje. Na primer, če so kamere postavljene nad očmi, bi uporabnik imel pogled na svet z malenkost večje višine, kot je navajen. Temu problemu se lahko izognemo z uporabo dodatnih ogledal, kar pa naredi sistem bolj kompleksen. Pri optičnem pristopu tega problema ni.

Prednosti video pristopa nad optičnim.

**Pravo zakrivanje:** Glavni problem pri optičnem pristopu je, da navidezni objekti ne prekrivajo realnih objektov, ker optični združevalniki dovolijo svetlobo tako iz realnega in navideznega izvora.

**Sinhronizacija:** Pri video pristopu lahko video realnega sveta namerno zakasnimo, da se ujema z zakasnitvijo navidezne slike.

**Dodatne metode registracije:** Informacijo o lokaciji uporabnika pri optičnem pristopu dobimo samo od naglavnega sledilnika. Video pristop lahko to informacijo dobi tudi iz digitalizirane slike realnega sveta. Tako lahko uporabi dodatne metode za registracijo.

### **3. Obogatena resničnost danes**

#### ***3.1 Raziskovalni problemi***

Glavna problema raziskovalcev obogatene resničnosti sta sledenje in registracija. Oba morata biti primerno rešena pri posamezni aplikaciji. Uporabnik takoj ugotovi, ali sta sledenje in registracija natančni. Glede na aplikacijo in napravami, ki jih uporablja, se zahteve spreminjajo. Aplikacija lahko zahteva robustno ali natančno sledenje, hitro ali natančno registracijo. Rešitve se morajo prilagoditi zahtevam uporabe.

##### **3.1.1 Registracija**

Registracija uporablja rezultate sledenja, da ustrezno poravna navidezno z realnim. Poravnava mora biti narejena glede na pozicijo realnih objektov in pozicijo kamere. Registracija je bodisi odprte ali zaprte zanke. Odprtozančna registracija se nanaša samo na informacije, pridobljene z sledenjem, ne uporablja povratnih informacij. Zaprtozančna registracija zagotavlja povratno informacijo o natančnosti registracije, ki se upošteva v prihodnjih izračunih. Doseči pravilno registracijo je lahko zapleteno, saj uporabnik vidi, kaj se zgodi v realnem svetu. Uporabnik vedno prepozna nepravilno poravnanost med navideznimi predmeti in realnimi objekti. Glede na vsebino nekaterih aplikacij natančnost ne sme biti prednostna naloga. Na primer, če gre za podajanje informacij o stavbah v mestu, je sprejemljivo, če to obvestilo ni prav v središču stavbe. Vendar pa ta obvestila ne smejo biti prikazana na drugih stavbah. Druge aplikacije pa zahtevajo zelo natančno registracijo. Za te vrste aplikacij ni samo problematično, če vsebujejo napačne informacije, temveč je to resen razlog, da omenjene aplikacije ne uporabljamo. Če turistična aplikacija na trenutke prikaže nepravilne informacije, je to sprejemljivo, vendar bi lahko v medicinski aplikaciji uporaba napačnih informacij resno škodovala bolniku. Ronald T. Azuma navaja, da je natančno registracijo težko doseči, saj obstajajo zelo različni viri napak. Razdeli jih na dve širši kategoriji napak: statične in dinamične. Statične napake registracije se zgodijo, ko se niti kamera niti objekti ne premikajo. Dinamične napake pa se pojavljajo, ko imamo gibanje. Povzročajo jih zakasnitve.

### 3.1.2 Sledenje

Sledenje določa položaj in orientacijo kamere v okolju. Problemi sledenja, pristopi in rešitve so predstavljeni v nadaljevanju. Sledenje je bistveni del vsake aplikacije obogatene resničnosti. Navideznih objektov ne moremo umestiti v realno okolje brez znane pozicije. Sistemi za sledenje določajo tri stopnje svobode, tako za pozicijo kot orientacijo. Pozicijo izračunajo kot 3D točko s koordinatami  $x$ ,  $y$  in  $z$ , orientacijo pa kot 3D vektor. Te vrednosti pozicije in orientacije so bodisi absolutne ali relativne glede na okolico. Sledenje mora biti natančno, točno in zanesljivo, kot je najbolj mogoče, da se ustvari iluzija, da je navidezna vsebina del realnega sveta. Idealno bi bilo, da bi postopek sledenja izračunal rezultate v realnem času, ne da bi nanj vplivalo različno okolje in različne okoliščine, na primer spreminjanje svetlobnih pogojev. Tak popoln sistem za sledenje je težko doseči. Večina sistemov je zato specializiranih in najbolje delujejo pod določenimi pogoji in omejitvami. Kot pri registraciji je ta sistem za sledenje lahko odprto ali zaprtozančni. Zhou je zaprtozančni sistem sledenja definiral kot sistem, ki svoje napake popravlja dinamično, odprtozančni pa jih ne. To je posledica dejstva, da zaprtozančni sistemi zbirajo povratne informacije o njihovih rezultatih in prilagajajo svoje izračune.

### 3.2 Pristopi sledenja

Pristope sledenja lahko razvrstimo na različne načine. Vendar pa se skoraj vsako delo na tem področju raziskav strinja z delitvijo na tri pristope:

- Sledenje s pomočjo računalniškega vida
- Sledenje s pomočjo senzorjev
- Hibridno sledenje

Pristopi se razlikujejo po načinu zbiranja podatkov o okolju.

#### 3.2.1 Sledenje s pomočjo računalniškega vida

V tej kategoriji obstajata dva pristopa. Prvi uporablja umetne oznake, drugi pa uporablja naravne značilke. Tabela 3.1 prikazuje razvrstitev pristopov, ki temeljijo na računalniškem vidu. Raziskovalci metod sledenja se predvsem osredotočajo na to kategorijo.

<b>Sledenje s pomočjo računalniškega vida</b>			
<b>Sledenje s pomočjo oznak</b>		<b>Sledenje brez oznak</b>	
Uporaba pasivnih oznak	Uporaba aktivnih oznak	Na osnovi modela	Obdelava slike

*Tabela 3.1: Pristopi sledenja s pomočjo računalniškega vida*

Raziskovanje je neenakomerno porazdeljeno med podobmočja. Sledenje na osnovi značilk je najbolj aktivno raziskovalno področje, čeprav so raziskovalci naredili veliko izboljšav, mnogo težav še vedno ostaja. Sledenje na osnovi modela je zelo novo raziskovalno področje, kateremu se namenja vedno več pozornosti v zadnjih letih. Sledenje s pomočjo oznak je zelo dobro raziskano področje in zelo malo je bilo novih premikov na tem področju v zadnjih letih. Prve aplikacije obogatene resničnosti so uporabljale sledenje s pomočjo oznak in mnoge jih še vedno. Čeprav sledenje s pomočjo oznak omejuje aplikacije, dobimo najbolj stabilne rezultate, tudi v primeru omejene računske moči.

### 3.2.2 Sledenje s pomočjo senzorja

Spodnja tabela prikazuje razdelitev področja sledenja s pomočjo senzorjev. To je grob pregled najpogostejših pristopov, ki uporabljajo senzorje. Mnogo mobilnih aplikacij obogatene resničnosti za sledenje uporablja senzorje. Kompas, akcelometer, GPS so sedaj že pogosto vgrajeni v mobilnik. Najbolj razširjeni aplikaciji v tem področju sta Wikitude World Browser in Layar, obe za mobilne telefone.

Sledenje s pomočjo senzorjev		
Magnetni senzorji	Inercialni senzorji	časovno frekvenčni senzorji
npr. kompas	npr. akcelometer, mehanični giroskop, inclinometer, pedometer	npr. GPS, ultrasonic, optični giroskop

*Tabela 3.2: Pristopi sledenja s pomočjo senzorjev*

### 3.2.3 Hibridno sledenje

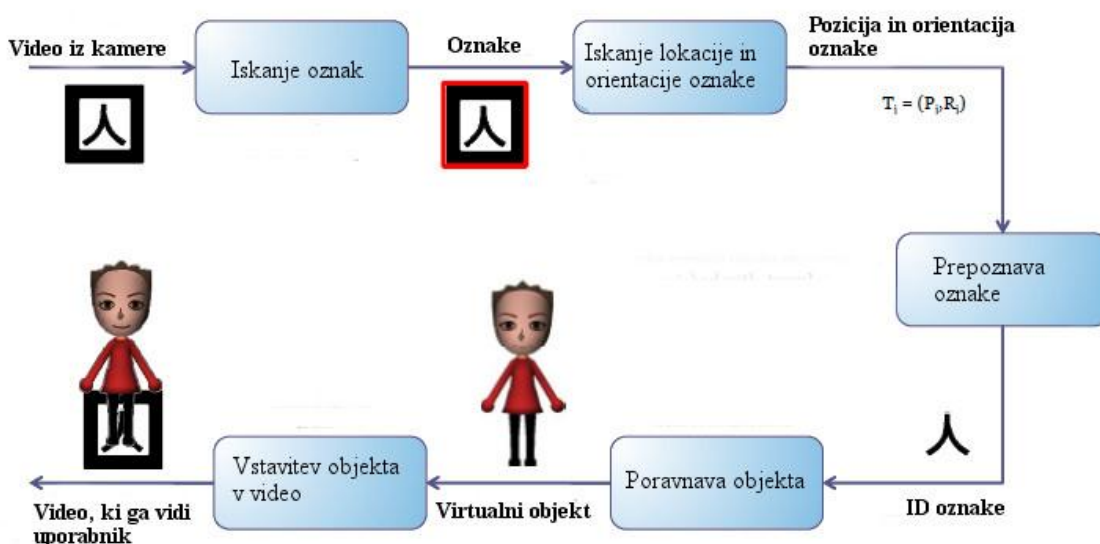
Hibridni pristop združuje različne pristope. Hibridni sistemi skušajo nadomestiti pomanjkljivosti enega pristopa z uporabo drugega oziroma več drugih. Izbrani pristop je še posebno primeren za naloge, katerih drugi pristop ne more zadovoljivo rešiti. Hibridni pristopi pogosto služijo za pridobivanje povratnih informacij sledenja in njihovo uporabo v korist sistema. Tako tvorijo zaprtizančno sledenje. Priljubljene kombinacije vključujejo sledenje na osnovi računalniškega vida in sledenje s pomočjo senzorjev. Mobilne aplikacije pogosto uporabljajo več senzorjev za sledenje. Primer je Google Sky Map. Z uporabo GPS-ja, akcelometra in kompasa ugotovi, katera ozvezdja nočnega neba posname kamera mobilnega telefona.



### 3.3 Obogatena resničnost z uporabo oznak

Za vse aplikacije obogatene resničnosti, ki temeljijo na uporabi oznak, obstaja nek skupen tok dogodkov, viden na Sliki 5.1. V sliki, ki jo zajamemo s kamero, moramo poiskati oznako. S pomočjo oznake izračunamo translacijsko matriko, s pomočjo katere navidezni objekt prikažemo nad najdeno oznako.

Oznake in algoritmi za njihovo odkrivanje in prepoznavo se med programskimi knjižnicami razlikujejo.



Slika 3.1: Tok dogodkov pri obogateni resničnosti z uporabo oznak

#### 3.3.1 Odkrivanje oznake

Prvi korak sledenja je iskanje značilik iz slike kamere. Značilke, uporabne za detekcijo, so točke, črte, regije. Najpogostejše metode vključujejo binarizacijo slike, narejeno s pomočjo fiksnega ali izračunanega pragu. Po upragovljanju, iskanje povezanih regij, robov, točk ni več tako zapleteno. Slabost upragovljanja je, da nekatere značilke na sliki ne bodo najdene, če je kontrast premajhen. Detektirane regije, robove, točke se uporabijo za določanje 2D točk, ki jih lahko uporabimo za določanje položaja. 2D točke so najdene z interpolacijo več pikslov, tako da je točnost najdenega položaja točke večja od ločljivosti slike.

### 3.3.2 Določanje položaja

Položaj se določa z uporabo več značilik iz slike, ki pripadajo znanemu 3D objektu. Za nedvoumno določitev položaja oznake potrebujemo vsaj štiri nelinearne točke iz slike in njihove korespondenčne 3D točke v koordinatnem sistemu sveta. Te štiri točke so teoretično lahko katerekoli točke, vendar iz praktičnih razlogov sledenja je bolje, če približno tvorijo kvadrat in niso preveč skupaj.

Za nedvoumno oceno položaja oznake iz več slik, potrebujemo vsaj tri nelinearne 3D točke, katere tvorijo ravnino v 3D prostoru. Z uporabo več kamer lahko uporabimo stereo tehnike za pridobitev 3D položaja značilke. To vključuje iskanje iste značilke v dveh slikah različnih kamer. Če poznamo zunanje parametre kamere, lahko geometrijsko pridobimo 3D lokacijo značilke, ne pa tudi orientacije. Z združevanjem informacij treh različnih nelinearnih točk, lahko pridobimo tudi orientacijo oznake.

Ker uporabljamo minimalno število točk za določanje položaja, lahko majhna napaka ene točke vodi do velike napake dobljenega položaja. Z uporabo dodatnih točk izboljšamo oceno položaja.

### 3.3.3 Prepoznavanje oznake

Sistem sledenja mora znati razlikovati različne oznake. Obstajata dve pomembni lastnosti: množica oznak za prepoznavanje in zamenjava oznake. Sistem sledenja naj bi bil sposoben prepoznati dovolj veliko množico različnih oznak, istočasno pa naj ne bi prišlo do zamenjave oznake. Ponavadi se ti dve lastnosti izključujeta. Večja množica oznak za prepoznavanje pomeni večjo enakost posameznih oznak in s tem večjo verjetnost zamenjave.

Obstaja več pristopov prepoznavanja oznak:

1. Ujemanje značilik (najdene značilke se primerjajo z značilkami znanih oznak)
2. Ujemanje predlog (slike se primerjajo z znanimi slikami)
3. Binarno kodirana informacija (oznaka vsebuje binarno kodirano informacijo)

Pristop ujemanja značilik oziroma predlog ima slabost v manjši množici oznak, ker morajo biti značilke oziroma predloge znane vnaprej. V primeru večje množice oznak je verjetnost zamenjave večja. Prednost pristopa ujemanja predlog je možnost uporabe uporabniku prijaznih predlog. Prav tako imamo pri pristopu ujemanja značilik veliko izbiro oznak. Pri

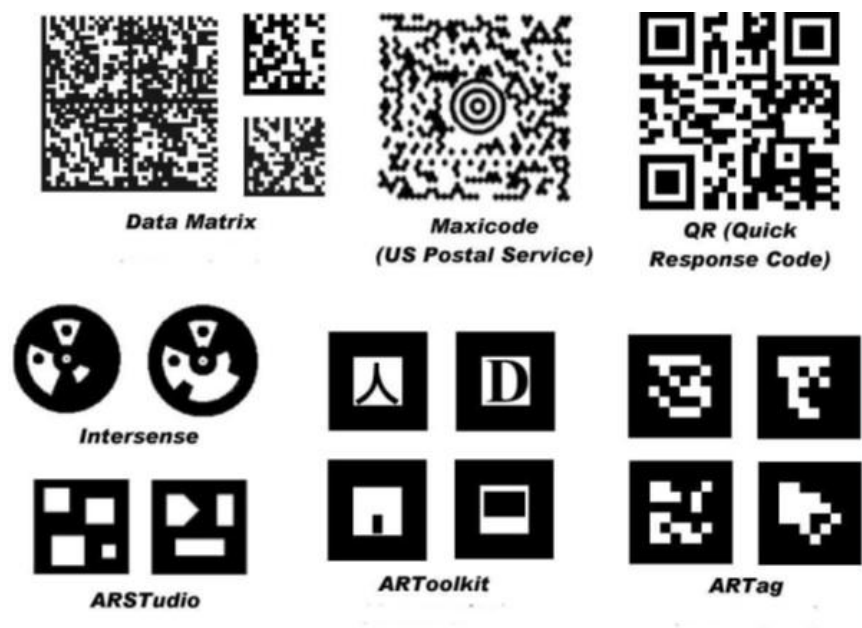
binarno kodirani informaciji nimamo znanih objektov. Tu imamo dekodirno funkcijo, ki dekodira binarno informacijo oznake, podobno kot sistem barkode. Zmožnost popravljanja napak zmanjša verjetnost zamenjave oznake.

### 3.3.4 Sistemi oznak

Dodajanje oznak v okolje je koristno, ko želimo imeti informacijo o poziciji objekta glede na kamero, na primer pri sledenju, foto-modeliranju, navigaciji robota in obogateni resničnosti. Dvodimenzionalne vzorce lahko dodajamo v okolje in jih prepoznavamo iz slik kamere. Sistemi oznak so sestavljeni iz množice dvodimenzionalnih vzorcev in algoritmov računalniškega vida za njihovo prepoznavo iz slik.

Kriteriji za ocenjevanje določenega sistema so:

1. False positive rate (oznaka je prepoznana, čeprav to ni)
2. Zamenjava oznak (oznaka je prepoznana kot neka druga oznaka)
3. False negative rate (oznaka je vsebovana v sliki, vendar ni najdena)
4. Minimalna velikost oznake za zanesljivo prepoznavo (manjša kot je velikost oznake, večja je lahko oddaljenost kamere od nje)
5. Odpornost na svetlobne razmere (dobra lastnost za prepoznavo oznak je, če jih lahko prepoznamo tudi v malo slabših pogojih)
6. Pravilnost prepoznane lokacije oznake (kako dobro se prepoznana lokacija oznake ujema z dejansko lokacijo oznake)



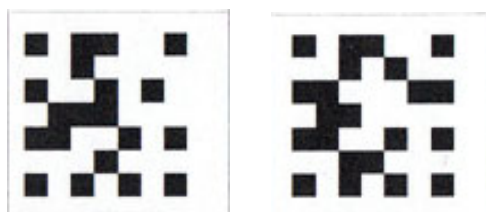
Slika 3.2: Sistemi oznak

Veliko sistemov umetnega vida v industriji uporablja dvodimenzionalne vzorce, ki vsebujejo informacijo, podobno kot barkode na izdelkih. Njihov namen je vsebovanje informacij in ne lociranje izdelka, kot je to na primer v aplikacijah obogatene resničnosti.

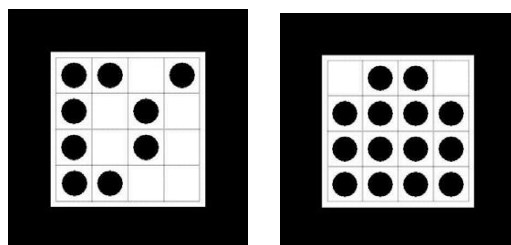
Maxicode, datamatrix in QR (Quick response) so primeri oznak, ki vsebujejo informacijo. Vsi trije uporabljajo metode za popravljanje napak v primeru nepravilno prebranih podatkov.

V splošnem so te oznake uporabne za kodiranje informacij in ne toliko za sisteme obogatene resničnosti. Prvič za to niso sploh namenjeni in ne bi dobro delovali v primerih, ko pride do velike perspektivne popačenosti. Ne zagotavljajo dovolj točk za izračun 3D pozicije (zagotavljajo tri namesto štirih). Kot zadnje pa zasedejo velik del slike, kar zmanjša njihovo področje uporabnosti.

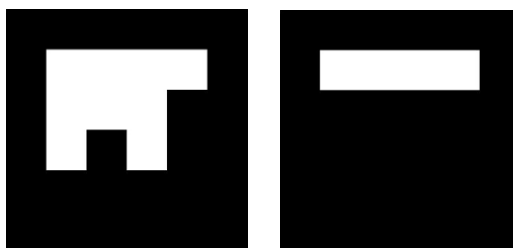
Eden prvih projektov uporabe 2D oznak, iz katerih se je lahko pridobilo vseh 6 stopenj svobode, je bil Rekimotov 2D Matrix Code, leta 1996. Leta 1999 je Kato uporabil podoben pristop za izdelavo ARToolKit-a, ki je objavljen pod GPL licenco. Od takrat je nastalo mnogo knjižnic za sledenje kvadratnim oznakam, med njimi ARStudio, Cybercode, SCR in IGD (sistema, ki se uporabljata v ARVIKA projektu), HOM, ARTag.



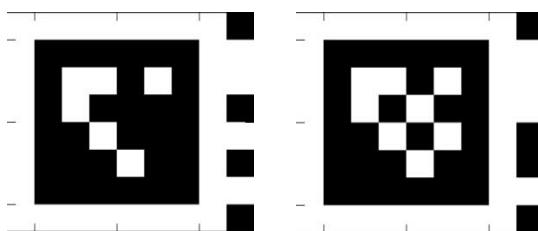
*Slika 3.3: Cybercode oznaki*



*Slika 3.4: SCR oznaki*



*Slika 3.5: IGD oznaki*



*Slika 3.6: HOM oznaki*

ARToolKit oznaka je kodirana z vzorcem v notranjem kvadratu oznake. Dekodiranje temelji na zelo poenostavljenem algoritmu ujemanja predlog, ki primerja tri geometrične invariante oznake z invariantami predhodno registriranih vzorcev. ARToolKit za vsako oznaki vrne stopnjo zaupanja ( $cf$ ). V primeru vrednosti  $cf \geq 0,5$  pomeni, da je oznaka pravilno prepoznana.

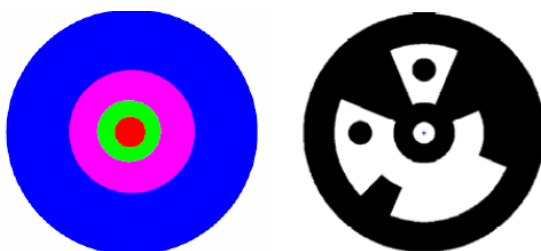
HOM sistem oznak je bil razvit leta 1994 z namenom fotogrametrije. Sistem je bil kasneje uporabljen v industrijskih programih za dokumentiranje in vzdrževanje. Poleg glavnega kvadrata je stranska vrstica, ki zagotavlja 6 bitov za izboljšanje zanesljivosti pri prepoznavanju oznake. Sistem HOM ima 7 stopenj zaupanja (od 0 do 6). 0 pomeni najvišjo, 6 pa najnižjo stopnjo zaupanja. V primeru stopnje zaupanja  $\leq 2$  je oznaka vrnjena kot prepoznana.

IGD sistem je bil razvit v IGD, ki je ARVIKA partner. ARVIKA je raziskovalni projekt za razvoj industrijskih aplikacij obogatene resničnosti, podprt s strani nemške vlade. Veliko aplikacij, povezanih s tem projektom, uporablja IGD sistem oznak. Slika 5.5 prikazuje dve taki oznaki. IGD oznaka je kvadrat, sestavljena iz polja  $6 \times 6$  enako velikih kvadratov.

Notranja 4 x 4 polja so namenjena prepoznavi oznake in njene orientacije. Knjižnice IGD sistema oznak so dostopne sodelujočim pri projektu ARVIKA.

SCR sistem je bil razvit za lokalizacijo in sledenje v različnih aplikacijah obogatene resničnosti. Sistem se uporablja tudi pri nekaterih partnerjih projekta ARVIKA. Tudi te oznake so kodirane z notranjo matriko. Vsaka SCR oznaka nudi 8 korespondenčnih značilk. Ta sistem je trenutno implementiran samo kot OCX kontrola za uporabo v Windows aplikacijah. Dostopni so tudi C++ razredi, ter primeri kode za integracijo SCR sistema z drugimi aplikacijami.

Cho in Neumann [9] sta predlagala uporabo okroglih oznak za sistem obogatene resničnosti, prikazan na Sliki 5.7. Njihov sistem uporablja oznako, sestavljeno iz obročev različnih barv. Vsaka oznaka je prepoznana po zaporedju barv. Pozicija kamere je izračunana z uporabo treh ali več oznak. Naimark in Foxter [10] pa sta ustvarila okroglo oznako na principu 2D barkode, kot to prikazuje desna oznaka na Sliki 5.7. Trenutno okrogle oznake uporabljata knjižnica Cantag [11] in VIS-Tracker [12] podjetja Intersense.



*Slika 3.7: Sistema oznak okroglih oblik*

### 3.3.5 Sledenje oznaki

Področje sledenja oznaki preučuje umetne strukture, ki so postavljene v okolje, ter kako se jih lahko uporabi za namene sledenja. Slika 5.5 prikazuje primer oznake.



*Slika 3.8: Oznaka programskega orodja ARToolKit*

Pri načrtovanju oznake je najpomembnejše, da se njegovala struktura razlikuje od okolja. Verjetnost, da bi našli podobne strukture, mora biti čim manjša.

Owen [8] je v svojem delu določil kriterije za izdelavo »dobre« oznake:

- Idealna oznaka mora zagotavljati nedvoumno določitev položaja in orientacije glede na kalibrirano kamero.
- Oznaka ne sme favorizirati neke orientacije nad drugo.
- Oznaka mora biti del množice in je težko zamenljiva z ostalimi oznakami.
- Oznaka mora biti hitro najdena in prepoznana s hitrimi in preprostimi algoritmi
- Oznaka mora biti prepoznana tudi pod večjim kotom

Preden se sledenje lahko začne, morajo biti oznake umeščene v okolje, oziroma nameščeni na objekte katerim aplikacija sledi. Aplikacija uporablja znanje o teh oznakah, na primer njihovo lokacijo in mere. Z izbiro umetnih struktur, ki se zlahka ločijo od okolja, sistemi sledenja lahko enostavno izračunajo pozicijo kamere v odvisnosti od oznake. Oznake so lahko aktivne ali pasivne. Aktivne oddajajo neke vrste signal, na primer uporaba LED diod. Video predstavlja osnovo za detekcijo značilik, ki jih nato aplikacija primerja z znanjem o obstoječih oznakah. Oznake omogočajo robustno sledenje, tudi če smo omejeni z računsko močjo. Na žalost ne moremo uporabiti nepripravljenega okolja. Ko so oznake postavljene v okolje, jih je

potrebno vzdrževati, da se zagotovi njihova razpoložljivost in celovitost. Pred uporabo sledenja s pomočjo oznak, so priprave vedno potrebne. Druga pomanjkljivost je v tem, da mora biti oznaka vedno vidna, pa še takrat lahko pride do napak pri sledenju, če je oznaka predač ali pa če pride do prekrivanja samo manjšega dela oznake. Poleg tega je težko ustvariti niz oznak, ki imajo pomen za uporabnika in da jih sledenje zlahka prepozna.



## 4. Sistemi oznak

### 4.1 Primerjava

Zang [6] je v svoji raziskavi, kjer je primerjal štiri sisteme oznak (ARToolKit, HOM, IGD, SCR), prišel do naslednjih zaključkov.

ARToolKit je združljiv z večino računalniških sistemov. Je dobro dokumentiran in enostaven za uporabo. Detekcija in dekodiranje oznake je hitra in stabilna. Najbolje se je odrezal pri hitrosti prepoznavne ene oznake ter pri prepoznavi oznake manjše velikosti.

Primeren je za prototipe, uporabljen pa je bil že v mnogih aplikacijah obogatene resničnosti. V primeru aplikacij, ki zahtevajo več sto ali tisoč oznak, je potrebno prav toliko oznak registrirati in ročno popravljati registracijsko datoteko, kar je zelo zamudno.

Sistem uporablja zelo poenostavljen algoritem ujemanja predlog za dekodiranje oznake.

Prednost takega pristopa je večja hitrost, ki pa je odvisna od števila oznak na sliki in tudi od velikosti množice oznak. Po drugi strani pa večkrat pripelje do napačne prepoznave.

Najbolje se je izkazal tudi pri prepoznavi oznake manjše velikosti (14 x 14 pikselov).

ARToolKit uporablja binarno sliko za pridobivanje značilnk. Od praga pri binarizaciji slike je odvisna detekcija lokacije robov, kar lahko pripelje do napak pri natančnosti pridobljenih značilnk. Zanimivo je, da je stopnja zaupanja pri najslabši izostritvi večja kot pri boljših. To namiguje na to, da stopnja zaupanja ni najbolj zanesljiva.

Sistem HOM deluje dobro v mnogih pogledih. Zasnovan je s sistematičnim kodiranjem, tako kot IGD in SCR sistemi oznak. Poleg tega vsaka oznaka vsebuje posebno vizualno strukturo, ki zagotavlja zanesljivost prepoznave. Dobra lastnost je 7 stopenj zaupanja (0-6), ki označujejo zanesljivost prepoznave oznake. Ko je stopnja zaupanja  $\leq 2$ , je oznaka prepoznana z visoko stopnjo zaupanja. Pri stopnji  $\geq 5$  rezultati niso preveč zanesljivi. Pri testiranju se je opazilo, da so stopnje zaupanja zanesljive in konsistentne s kakovostjo slik oznake. Pri sistemu ARToolKit smo opazili, da temu ni ravno tako.

Tudi pri tem sistemu je hitrost odkrivanja ene oznake hitra, se pa zelo poveča z večjim številom oznak na sliki. Najbolje se je izkazal pri prepoznavi oznake s perspektivno izkrivljenostjo.

Sistem SCR je najbolj počasen pri detekciji in pridobivanju značilnk iz slike ene oznake. Vendar pa uporablja začasno informacijo sledenja, kar pomeni hitrejše delovanje pri zaporedjih slik. Zmogljivost sistema se zmanjša minimalno pri večanju števila oznak na sliki. Dokaj dobro deluje tudi pri natančnosti detektiranih značilnk in pri zanesljivosti detektiranja ter dekodiranja. Sistem SCR bo vrnil oznako kot prepoznano le v primeru visoke stopnje zaupanja. Ta lastnost zmanjšuje napačno prepoznavo, vendar lahko vodi v manjšo prepoznavnost oznak. Zaradi sistematičnega kodiranja ni potrebe po predhodni registraciji oznak. Isto velja za sisteme HOM in IGD. S kodirno matriko 4 x 4 lahko ustvarimo več kot 10.000 različnih kod. Vendar pa je sistem SCR na voljo samo kot OCX kontrola za Windows aplikacije.

Sistem IGD je razširjen v mnogih ARVIKA projektih. Tudi te oznake so sistematično kodirane. Pridobivanje značilnk je natančna in hitrost sistema je spodobna. Malo nepriročno je, da mora uporabnik ustvariti wrapper knjižnico in jo prevesti z Intel C++ prevajalnikom za uporabo v Windows aplikacijah. Za nezmožnost detekcije več oznak na sliki avtorji krivijo sebe in njihovo nepoznavanje sistema IGD.

Fiala [13] je v svojem delu predstavil sistem ARTag, ki je bil narejen z namenom ponuditi boljšo rešitev od obstoječih. ARTag oznaka je ID osnovana, kar je drugačen pristop od ARToolKit-a, ki uporablja prepoznavanje na osnovi ujemanja predlog. Oznaka je kvadratne oblike, kjer je rob lahko bele ali črne barve. Njegova velikost je 10 x 10 enot. Notranjost pa je velika 6 x 6. Vsaka notranja celica je bela ali črna in nosi en bit informacije. Sistem vsebuje 2002 oznak. Uporablja metodi CRC in FEC. Verjetnost, da prepozna oznako, čeprav ne obstaja je 0,0039 %. Pri ARToolKit-u je ta verjetnost mnogo večja, kar je velika slabost omenjenega sistema. Tudi verjetnost zamenjave oznake je pri ARToolKit -u veliko večja kot pri ARTag-u. Oznake so pravilno prepoznane tudi v primeru delnega prekrivanja, česar ARToolKit ne zmore.

## 4.2 Programska oprema

Večina programske opreme obogatene resničnosti je namenjena točno določeni nalogi, zato nikoli ne postane javna. Vseeno pa obstaja nekaj javno dostopnih knjižnic. Med njimi so ARToolKit, ARTag, ALVAR, Studierstube, FLARToolKit, SLARToolKit, NyARToolKit, ArUco, FLARManager. Omenjene knjižnice zahtevajo od uporabnika znanje programiranja na nižjem nivoju. Obstajajo pa tudi programi, ki so za uporabnika bolj preprosti in po možnosti ne potrebujejo nobenega znanja programiranja, na primer DART, ATOMIC, AR-media Plugin for Google SketchUp.

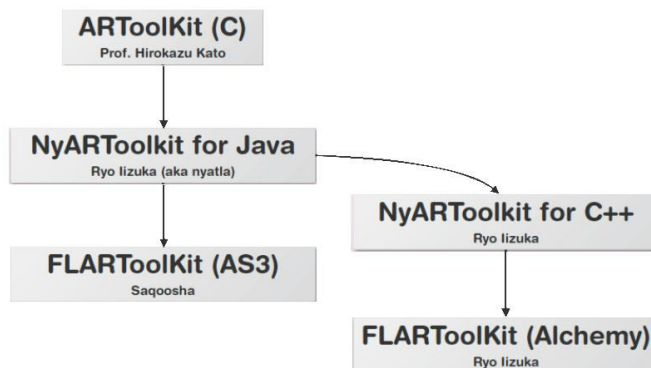
### 4.2.1 ARToolKit

Najbolj popularna knjižnica za sledenje s pomočjo oznak je ARToolKit. Razvili so jo na washingtonski univerzi. Distribuirana je v okviru licence GPL. Napisana je v C-ju in je bila zasnovana za delovanje na osebnih računalnikih. Podpira tako optično prozorno, kot video prozorno obogateno resničnost in vsebuje vse funkcije, ki so potrebne za izgradnjo enostavne aplikacije. Poleg sledenja, ki je jedro sistema, vsebuje še dodatne funkcije. ARToolKit je sposoben integrirati in kalibrirati kamero, nato pa zajemati video. Funkcije sledenja pretvorijo slike videa v binarne in poiščejo črne okvirje, ki jih vsebujejo vse oznake. Izračunajo položaj in orientacijo oznake glede na kamero. Nato primerjajo simbole oznak s predlogami. V zadnjem koraku ARToolKit prekrije video tok z navideznimi objekti. ARToolKit lahko prepozna več oznak v sliki. Kot pomoč uporabnikom, knjižnica vsebuje več primerov uporabe. Slabost knjižnice pa je, da se težko spopade s prekrivanjem in močnimi sencami.

Knjižnica zagotavlja naslednje lastnosti:

- Sledenje ene kamere
- Sledenje oznaki
- Uporabo poljubnega vzorca v oznakah
- Preprostost kode kalibracije kamere
- Dovolj hitrosti za aplikacije v realnem času
- SGI IRIX, Linux, MacOS, Windows distribucije
- Symbian, iPhone, Android, Windows phone distribucije

Obstajata dve verziji. Verzija 2.72 je prosto dostopna, verzija 4.4 pa je profesionalna verzija ARToolKit in je plačljiva. ARToolKit je bil prenesen še v druge programske jezike, kot je prikazano na Sliki 4.1.



Slika 4.1: Prenos ARToolKita v druge programske jezike

## 4.2.2 ARTag

ARTag je samosvoj toolkit, ki ga je razvil prof. Mark Fiala iz Columbia University. Ima popolnoma drugačen pristop kot ARToolKit. Oznake ima ID osnovane z zmožnostjo popravljanja napak, le da tu lahko uporabimo le 2002 oznaki. Knjižnica je bila komercialni izdelek, vendar se je njen razvoj, kot kaže, ustavil.

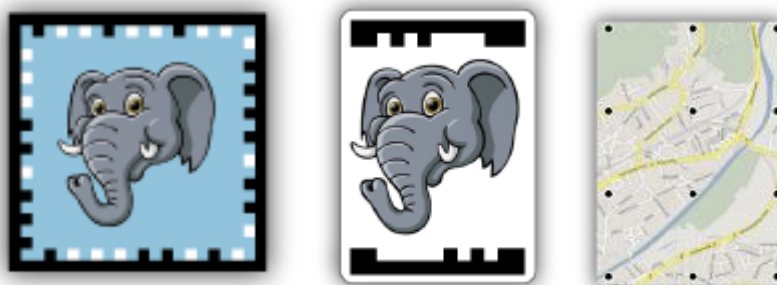
## 4.2.3 ARToolKitPlus

ARToolKitPlus knjižnica temelji na ARToolKit knjižnici, ki vsebuje tudi ID osnovane oznake po zgledu ARTag knjižnice in je prav tako distribuirana v okviru licence GPL. Razvita je bila kot del projekta »Handheld AR« tehnične univerze v Grazu. Knjižnica je bila prepisana iz C-ja v C++, optimizirana je bila za mobilne naprave in uporablja nove algoritme za oceno pozicije. Ne vsebuje pa kode za zajem slike in grafičnega izrisovanja. Njen razvoj se je končal junija 2006, ko je bila nadomeščena s knjižnico Studierstube Tracker.

#### 4.2.4 Studierstube Tracker

Studierstube Tracker je prav tako knjižnica za obogateno resničnost, ki pa ni prosto dostopna. Napisana je čisto od začetka in ne temelji na ARToolKit-u. Napisana je bila tako za osebne računalnike kot tudi za mobilne naprave, ki uporabljajo operacijski sistem Windows za pametne telefone, ne pa tudi za Android. V primerjavi s knjižnico ARToolKitPlus je na mobilnih napravah približno dvakrat hitrejša, podpira pa tudi več vrst oznak. Te so:

- Oznake knjižnice ARToolKit
- ID osnovane oznake
- DataMatrix oznake
- Uokvirjene oznake (Slika 4.2 levo)
- Razdeljene oznake (Slika 4.2 na sredini)
- Pikčaste oznake (Slika 4.2 desno)



*Slika 4.2: Oznake knjižnice Studierstube Tracker: uokvirjena oznaka, razdeljena oznaka, pikčasta oznaka*

Uokvirjena in razdeljena oznaka sta ID osnovani oznaki z zapisom ob robu oznake. Znotraj oznake se je lahko poljubna slika in ne vpliva na prepoznavo oznake.

Pikčasta oznaka se največkrat uporablja na zemljevidih. Deluje na principu ujemanja predlog, in sicer se primerja slika, ki se nahaja med pikami.

#### 4.2.5 ALVAR

ALVAR je knjižnica za ustvarjanje programov navidezne in obogatene resničnosti. Razvita je bila na Finskem v VTT Technical Research centru. Knjižnica je prosto dostopna za nekomercialno uporabo. Trenutna verzija podpira obogateno resničnost po metodi prepoznavanja oznak, kasnejše pa naj bi vsebovale tudi metode, kjer oznake ne bodo

potrebne. Deluje na operacijskih sistemih Windows, v prihodnje pa naj bi tudi na sistemu Linux ter operacijskem sistemu za mobilne telefone Symbian s60.

Knjižnica zagotavlja naslednje lastnosti:

- Sledenje na osnovi oznak;
  - Natančna ocena pozicije oznake
  - Dva tipa kvadratnih oznak
  - Enostavno dodajanje novih tipov oznak
  - Prepoznavna oznak kljub prekrivanju
- Uporabo več oznak za oceno pozicije;
  - Koordinate oznak so lahko nastavljene ročno
  - Koordinate oznak so lahko ugotovljene z avtokalibracijo
- Ostale lastnosti;
  - Odstranjevanje oznak
  - Orodja za kalibracijo kamere
  - Kalmanova knjižnica in mnogo drugih filtrov

#### **4.2.6 ArUco**

Glavne značilnosti knjižnice:

- Detekcija oznak z eno vrstico kode
- Detekcija oznak, sestavljenih iz več oznak
- Potrebuje samo knjižnico OpenCv ( $\geq 2.1$ )
- Uporaba do 1024 oznak
- Enostavna integracija z OpenGL
- Hitra, zanesljiva ter zaradi knjižnice OpenCv delujoča na več platformah
- Vključuje primere
- Ima BSD licenco

### 4.2.7 NyARToolKit

NyARToolKit je prevedena različica knjižnice ARToolKit. Napisana je v različnih programskih jezikih:

- NyARToolKit (Java)
- NyARToolKitCS (C#)
- NyARToolKitCPP (C++)
- NyAR4psg (Processing)
- NyARToolKit za Android

Knjižnica prepoznava ID osnovane oznake, imenovane NyId oznake. Lahko se jo prosto uporablja pod pogoji licence GPL, obstaja pa tudi komercialna licenca.

### 4.2.8 FLARToolKit

FLARToolKit je prevedena različica knjižnice ARToolKit, napisana v programskem jeziku AS3. V bistvu gre za prevod knjižnice NyARToolKit, javanske različice knjižnice ARToolKit. FLARToolKit je najbolj pogosto uporabljena knjižnica za izdelavo flash aplikacij obogatene resničnosti. Podpira vse večje 3D grafične motorje za flash, kot so Papervision3D, Away3D, Sandy, Alternativa3D.

### 4.2.9 FLARManager

FLARManager je ogrodje, ki združuje tri knjižnice za sledenje:

- FLARToolKit
- Flare\*tracker
- Flare\*NFT

Vsebuje tudi knjižnice 3D grafičnih motorjev Papervision3D, Away3D, Away3D Lite, Sandy, Alternativa3D.

Flare\*tracker in flare\*NFT sta knjižnici za sledenje avstrijske družbe Imagination, ki pa sta žal plačljivi oziroma na svetovnem spletu lahko dobimo preizkusno različico, ki omogoča enominutno sledenje oznakam.

Flare\*tracker za sledenje uporablja oznake, flare\*NFT pa naravne značilke. Flare\*tracker je zanimiv, ker podpira več vrst oznak in sicer ID osnovane oznake, uokvirjene oznake in datamatrix oznake. Te oznake prikazuje Slika 4.3.



Slika 4.3: Oznake knjižnice flare\*tracker

#### 4.2.10 SLARToolKit

SLARToolKit omogoča razvoj aplikacij s tehnologijo Silverlight. Knjižnica je zasnovana na podlagi knjižnic NyARToolKit in ARToolKit.

	Programski jezik	Operacijski sistem	Vrste oznak (Število oznak)	Sledenje več oznakam	Licenca
ARToolKit	C	SGI IRIX, Linux, MacOS, Windows	Poljubne	DA	GPL (verzija 2.72), komercialna (verzija 4.4)
ARTag	C++ / C#	Windows, Linux, MacOS	ID osnovane (2002)	DA	knjižnica ni več na voljo
ARToolKitPlus	C++	Windows, Linux	ID osnovane (4096)	DA	GPL
Studierstube Tracker	C++	Windows, Linux, MacoS, Windows Mobile, Windows CE, iPhone, Symbian	Poljubne, ID osnovane (4096), Okvirjene(4000000), Razdeljene, Pikčaste	DA	komercialna
ALVAR	C++	Windows	ID osnovane	DA	komercialna za polno verzijo
ArUco	C++	Windows, Linux	ID osnovane (1024)	DA	BSD
NyARToolKit	Java, C#, C++, Processing	Windows, Linux, MacOS, Android	ID osnovane (512, 5000)	DA	GPL, komercialna
FLARToolKit	ActionScript	Windows, MacOS	Poljubne, ID osnovane (512, 5000)	DA	GPL, komercialna
FLARManager FLARToolKit flare*tracker flare*NFT	ActionScript	Windows, MacOS	Poljubne, ID osnovane (4096, 512, 512,5000), DataMatrix, Okvirjene (1024)	DA	GPL, komercialna
SLARToolKit	Silverlight	Windows, MacOS, Windows Phone	Poljubne	DA	GPL, komercialna

Slika 4.4: Primerjava knjižnic za obogateno resničnost

### 4.3 Obogatena resničnost na spletu

Za ustvarjanje aplikacij za svetovni splet, lahko od prej naštetih knjižnic uporabimo FLARToolKit, FLARManager in SLARToolKit.



## 5. Aplikacija obogatena razglednica

V okviru diplomske naloge sem naredil preprosto internetno stran, katera na podlagi izbrane razglednice z vsebujočo oznako prikaže 3D model objekta, značilnega za to razglednico.

Poleg same izdelave internetne strani, sem se osredotočil tudi na sam izgled razglednice, kam je smiselno uvrstiti oznako, kako velika je lahko oznaka, da bo pravilno prepoznana, lahko oznaka vsebuje sliko, ki ima nek pomen ter ali je lahko sama slika tudi oznaka.

Spodnja slika prikazuje primer uporabe aplikacije obogatene resničnosti. Uporabnik pred spletno kamero pokaže izbrano razglednico, na kateri se nato nariše 3D model, značilen za to razglednico.



Slika 5.1: Primer uporabe aplikacije obogatene resničnosti

## 5.1 Izbira knjižnice

Glede na to da gre za izdelavo internetne strani, je bila moja prva izbira ogrodje FLARManager, ker vključuje dve knjižnici za sledenje z uporabo oznak, in sicer FLARToolKit in flare\*tracker. FLARToolKit je najbolj pogosto uporabljena knjižnica za izdelavo flash aplikacij obogatene resničnosti. Dobra stvar knjižnice flare\*tracker pa je, da vsebuje uokvirjene oznake. Glavni del naloge je bila zato izdelava flash datoteke, katero smo nato vstavili v internetno stran.

Za programiranje sem uporabil orodje Flash Builder 4.6, za grafični motor pa sem izbral knjižnico Papervision3D.

## 5.2 Kakšna je lahko oznaka?

Oznake so kvadrati, ki ji nato izbrana knjižnica za obogateno resničnost prepozna v videu. Oznaka je nek vzorec, katerega ustvarimo in natisnemo na neko podlago. Knjižnice že vsebujejo nekatere narejene oznake.

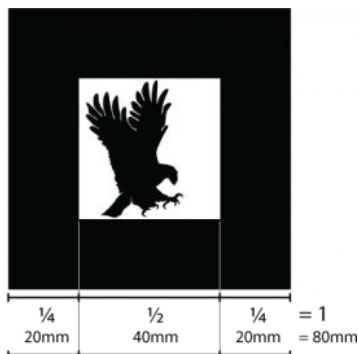
Omejitve oznak pri knjižnici FLARToolKit:

- Oznaka mora biti kvadrat
- Rob mora biti neprekinjen (po navadi cel črn ali bel) in mora ležati na podlagi kontrastne barve. Privzeta debelina robu je 25 % dolžine oznake.

Oznaka je lahko tudi barvna, vendar mora biti kontrast med robom oznake in podlago dovolj velik.

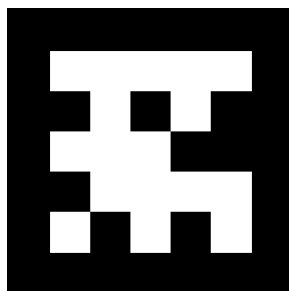
- Notranja slika oznake ne sme biti rotacijsko simetrična. Področje znotraj robov je lahko črno-belo ali barvno.

Primer oznake, ki upošteva zgornje omejitve, prikazuje Slika 5.2.



Slika 5.2: Oznaka, ki upošteva omejitve knjižnice FLARToolKit

FLARToolKit podpira tudi prepoznavo ID osnovanih oznak, imenovane NyId oznake, ki jo prikazuje Slika 5.3



*Slika 5.3: NyId oznaka, katere ID=10*

Knjižnica Flare\*tracker tudi lahko sledi ID osnovanim oznakam. Omogoča pa tudi sledenje uokvirjenim oznakam, ki je prikazana na sliki 5.4. Dobra stran teh oznak je, da v notranjosti lahko vsebujejo poljubno sliko, kar lahko naredi oznako uporabniku prijazno. Robustnost sledenja okvirnim oznakam pa je zagotovljena s črnim okvirjem okoli slike.



*Slika 5.4: Uokvirjena oznaka*

Slaba stran knjižnice flare\*tracker pa je, da je plačljiva oziroma je na svetovnem spletu dostopna samo demo verzija, katera zagotavlja samo enominutno sledenje oznakam.

### **5.3 Ustvarjanje novih oznak**

Novo oznako lahko ustvarimo s pomočjo predloge, katero dobimo v knjižnici ARToolKit. Lahko ustvarimo oznako katerekoli velikosti. Za knjižnico FLARToolKit je samo 50 % notranjosti celotne oznake mišljenih kot oznaka, kar pa lahko v konfiguracijski datoteki tudi

spremenimo. Slika je lahko barvna ali belo-črna in se lahko razteza v področje robu, vendar bo ta del izvzet pri prepoznavi oznake.

Za izdelavo nove oznake lahko uporabimo tudi program na svetovnem spletu [20].

Najbolj pogosta napaka pri ustvarjanju novih oznak je, da ustvarimo notranjost oznake s preveč detajli. Omenjena knjižnica vzorči sliko oznake z ločljivostjo 16 x 16 slikovnih točk.

V primeru podobnih oznak lahko kaj hitro pride do zamenjave.

#### **5.4 Izdelava vzorčne datoteke novoustvarjene oznake**

Ko imamo ustvarjeno novo oznako, moramo knjižnico naučiti prepoznati našo oznako. Rezultat učenja je vzorčna datoteka. Ta datoteka vsebuje podatke, ki predstavljajo notranjo sliko oznake. Ko zaženemo program obogatene resničnosti, se običajno naložijo tudi vzorčne datoteke, da program ve, katere oznake mora prepoznati. V primeru ID osnovanih oznak ne potrebujemo vzorčnih datotek, ampak samo identifikacijsko številko oznake.

Knjižnica ARToolKit vsebuje program `mk_patt.exe`, s pomočjo katerega dobimo vzorčno datoteko. Lahko pa uporabimo program na svetovnem spletu [21].

#### **5.5 Velikost oznake**

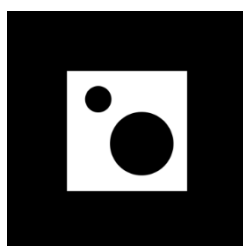
Velikost oznake je pomembna. Večja kot je oznaka, večja je oddaljenost kamere, pri kateri je oznaka prepoznana. Knjižnica ARToolKit navaja velikosti oznak in največjo razdaljo, pri kateri je oznaka še prepoznana. Gre za oznake, katere so prepoznane s pomočjo ujemanja vzorcev. Ne navaja pa, pri kakšni stopnji zaupanja so bile meritve opravljene. Omenjene meritve prikazuje Tabela 5.1.

Velikost oznake (cm)	Oddaljenost kamere (cm)
6,98	40,64
8,89	63,5
10,79	86,36
18,72	127

*Tabela 5.1: Največja razdalja kamere, pri kateri je oznaka prepoznana*

Na razdaljo, pri kateri je oznaka prepoznana, vplivajo tudi zapletenost oznake, kot med oznako in kamero ter tudi svetlobni pogoji. Bolj je oznaka zapletena, večji je kot med oznako in kamero, slabši so svetlobni pogoji, manjša mora biti oddaljenost kamere za njeno pravilno prepoznavo.

Ker ima razglednica omejeno površino, sem poskusil oznako karseda zmanjšati. Vzel sem oznako, prikazano na Sliki 5.5 ter naredil slike velikosti 1,2, 3 in 4 cm. Stopnjo zaupanja sem pustil privzeto od knjižnice FLARManager, in sicer 0,5. Meritve so prikazane v Tabeli 5.2.



*Slika 5.5: Oznaka, ki smo jo uporabili za meritve*

Velikost oznake (cm)	Oddaljenost kamere (cm)
1	15
2	25
3	50
4	70

*Tabela 5.2: Lastne meritve oddaljenosti kamere za pravilno prepoznavo oznake*

Pri tem je treba tudi poudariti, da gre za preprosto oznako in tudi meritve so se izvajale pri ugodnih svetlobnih pogojih .

Prav tako sem opravil meritve za ID osnovano oznako in uokvirjeno oznako, ki sta prikazani na Slikah 5.3 in 5.4. Meritve so prikazane v Tabelah 5.3 in 5.4.

Velikost oznake (cm)	Oddaljenost kamere (cm)
1	22
2	40
3	60
4	80

*Tabela 5.3: Največja oddaljenost kamere za ID osnovane oznake*

Velikost oznake (cm)	Oddaljenost kamere (cm)
1	10
2	20
3	35
4	45

*Tabela 5.4: Največja oddaljenost kamere za uokvirjene oznake*

Iz meritev je razvidno, da je najboljše rezultate dosegla ID osnovana oznaka. Po mojih lastnih izkušnjah bi bilo najbolje uporabiti tako velikost oznake, da smo lahko od kamere oddaljeni vsaj 50 cm. Tako je lahko izrisani 3D model večji.

## **5.6 Izdelava razglednice**

V tem podpoglavju bomo preučili, kam bi bilo najbolje umestiti oznako ter kakšna velikost je najbolj primerna oziroma še sprejemljiva. Imamo dve možnosti, oznako lahko postavimo na prvo stran razglednice, kjer je prikazana slika objekta, ali pa na zadnjo stran.

Za razglednico sem si izbral sliko Eifflovega stolpa, prikazano na Sliki 5.6.



*Slika 5.6: Prva stran razglednice*

### **5.6.1 Oznaka na prvi strani**

Pri vključevanju oznake na prvo stran razglednice sem najprej pomislil, ali sploh potrebuje oznako ali pa je lahko kar slika razglednice oznaka. Za izdelovanje vzorčnih datotek sem uporabil že prej omenjeni program na svetovnem spletu. Program razglednice ni prepoznal

kot oznako, saj razglednica ni bila uokvirjena s črnim robom na beli podlagi. Zato sem razglednico uokviril s črno-belim robom, ter tako dobil vzorčno datoteko. Opazil sem prvo slabost. Uokvirjena razglednica ne izgleda ravno najboljše. Prikazana je na Sliki 5.7.



*Slika 5.7: Obrobljena razglednica*

Naslednji poskus je bil vključiti oznako v sliko. Glede na prejšnje rezultate meritev je razvidno, da mora biti uokvirjena oznaka večja od ID osnovanih oznak in oznak, ki temeljijo na ujemanju vzorcev. Primerna velikost teh dveh vrst oznak bi bila od 3 do 4 cm, uokvirjena oznaka pa bi morala biti vsaj velikosti okoli 5 cm. Najbolj primerno mesto oznake je nekje v kotu, in sicer tam, kjer najmanj prekriva sliko razglednice.

V primeru razglednice Eifflovega stolpa smo izbrali zgornji levi kot, kot to prikazujeta Sliki 5.8 in 5.9. V primeru oznake, ki temelji na ujemanju predlog, sem izbral oznako, ki vsebuje sliko Eifflovega stolpa.



*Slika 5.8: Razglednica z ID osnovano oznako*



*Slika 5.9: Razglednica z oznako, ki temelji na ujemanju predlog*

V primeru oznak, ki temeljijo na ujemanju predlog, lahko izdelamo notranjost oznake v stilu razglednice, kot je to prikazano na Sliki 5.9.

Namesto slike lahko v notranjost oznake vstavimo ime mesta ali objekta na sliki. Pri teh oznakah moramo predvsem paziti na podobnost med njimi.

V primeru uokvirjenih oznak pa lahko v sredino oznake vstavimo karkoli, lahko pa je notranjost tudi transparentna, kot to prikazuje Slika 5.10. Edina slabost teh oznak je, da zasedejo več prostora od ostalih dveh vrst oznak.



*Slika 5.10: Razglednica z uokvirjeno oznako*

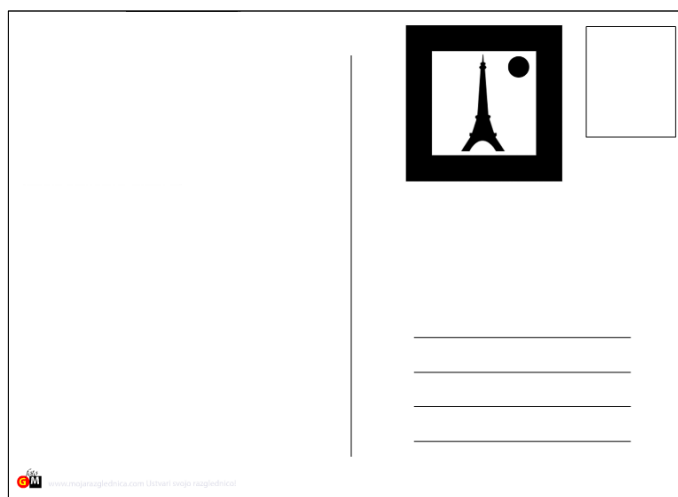


Prednost oznake na prvi strani razglednice bi lahko bila, da naredimo animacijski 3D model, ki se lahko prikaže iz slike objekta na razglednici. 3D model se privzeto vedno prikaže na sredini oznake, vendar mu lahko lokacijo prikazovanja po želji spreminjamo.

Slabost je lahko predvsem v tem, da oznaka prekriva del razglednice. Vendar to sploh ni velik problem, če izberemo sliko, ki ima dovolj prostora za oznako.

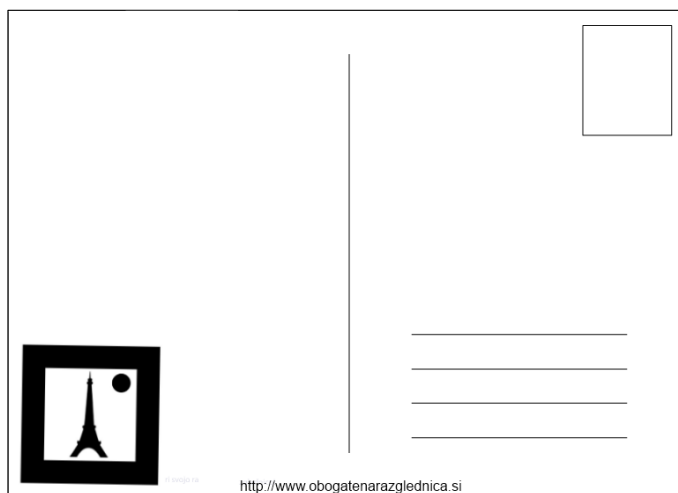
### 5.6.2 Oznaka na zadnji strani

Na zadnji strani razglednice imamo na prvi pogled več prostora. Prvo lokacijo oznake prikazuje Slika 5.11. Prostor zraven znamke je dovolj velik tudi za oznako velikosti 4 x 4 cm. Edini problem te lokacije je, da oznako lahko prelepimo z znamkami, pa tudi kakšna poštna štampljka lahko pokvari oznako. V tem primeru bi morali oznako nekako zaščititi.



*Slika 5.11: Zadnja stran razglednice z oznako zraven znamke*

Druga možna lokacija pa je nekje v kotu prostora, kamor pišemo pozdrave, kot to prikazuje Slika 5.12. Čeprav je prostora za oznako veliko, je potrebno pustiti prostor za pozdrave, tako da je tudi tu optimalna velikost od 3 do 4 cm, tako kot v primeru oznake na prvi strani.



*Slika 5.12: Zadnja stran razglednice z oznako v prostoru za pozdrave*

Tudi v primeru oznake na zadnji strani razglednice lahko uporabimo vse vrste oznak, ki smo jih uporabili v primeru oznake na prvi strani razglednice, pri čemer bi bile velikosti oznak enake.

Kot je tudi razvidno iz Slike 5.12 ob rob zapišemo še spletni naslov, kjer obogateno razglednico lahko uporabimo.

### **5.6.3 Povzetek**

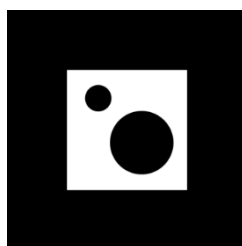
Pri lokaciji oznake, tako na prvi kot na zadnji strani lahko, uporabimo vse vrste oznak. Pri ID osnovanih oznakah in oznakah, ki temeljijo na ujemanju vzorcev, je priporočljiva velikost oznake od 3 do 4 cm, pri uokvirjenih oznakah pa 5 cm. Prav zaradi velikosti uokvirjenih oznak se nam njena uporaba za obogateno razglednico ne zdi priporočljiva.

Pri ostalih dveh vrstah oznak pa je izbira lokacije odvisna od tega, koliko slike razglednice bi prekrivala oznaka.

Pri izdelavi aplikacije obogatena razglednica, ki je predstavljena v naslednjem podpoglavju, sem uporabil oznake, ki temeljijo na ujemanju vzorcev. Za lokacijo oznake sem izbral zadnjo stran razglednice, zraven prostora za znamko.

## 5.7 Izdelava internetne aplikacije

Kot že rečeno, sem za izdelavo flash datoteke uporabil program Flash Builder 4.6 ter ogrodje FLARManager. Aplikacija je zasnovana, da podpira prepoznavo več oznak. Za vsak 3D grafični model zato potrebujemo edinstveno oznako. Na oznaki, vključeni na razglednico na Sliki 5.11, bom prikazal 3D model Eifflovega stolpa, na oznaki, prikazani na Sliki 5.13 pa bo prikazan 3D model gradu Colditz.



Slika 5.13: Slika oznake, na kateri aplikacija prikaže 3D model gradu

Vzorčne datoteke sem dobil s programom na svetovnem spletu [21]. Na obeh oznakah se je izrisal pravilni 3D model.

Slika 5.14 prikazuje enostavno internetno stran, katera nam za izbrano razglednico izriše 3D model objekta, ki je zanjo značilen.



Slika 5.14: Aplikacija obogatena razglednica

## 6. Zaključek

V diplomskem delu sem predstavil pojem obogatena resničnost. Predstavil sem vrste obogatene resničnosti in njihovo uporabo. Osredotočil sem se na obogateno resničnost z uporabo oznak, kjer sem opisal principe njenega delovanja. Raziskal sem katere knjižnice lahko uporabniki uporabijo za izdelovanje aplikacij obogatene resničnosti. Kljub temu, da je sedaj vedno več aplikacij obogatene resničnosti namenjenih pametnim telefonom, ki izkoriščajo predvsem lokacijske senzorje, smo pokazali, da še vedno obstajajo knjižnice za obogateno resničnost z uporabo oznak.

Tako sem tudi prikazal izgradnjo preproste internetne aplikacije obogatena razglednica. Prikazal sem izdelavo oznak ter njihovo umestitev na razglednico. Prišli sem do zaključkov, da so ID osnovane oznake in oznake, ki temeljijo na ujemanju vzorcev, najbolj primerne za mojo aplikacijo, saj lahko uporabimo oznako velikosti od 3 do 4 cm. Tako lahko razglednico držimo vsaj 50 cm od kamere, kar nam omogoča, da lahko opazujemo 3D model večje velikosti. Pri oznakah, ki temeljijo na ujemanju vzorcev, pazimo, da si v primeru velikega števila oznak, le-te niso preveč podobne, saj bi v tem primeru prišlo do zamenjave oznak in uporabniku bi se na razglednici izrisal napačen 3D model.

Mislím, da bi koncept aplikacije lahko zaživel v realnosti, pri čemer bi bilo treba spletno stran narediti uporabniku prijazno. Dobro bi bilo tudi preučiti obogateno resničnost brez uporabe oznak in ugotoviti njene prednosti in slabosti za uporabo v aplikaciji obogatena razglednica. Lahko bi izdelali tudi aplikacijo za mobilne telefone, saj bi bil ogled obogatene razglednice enostavnejši.

## Seznam slik

Slika 1.1: Premica navideznost – resničnost .....	4
Slika 1.2: Navidezna miza v resničnem svetu .....	5
Slika 1.3: Noga, prekrita s 3D modelom kosti .....	5
Slika 1.4: Primer uporabe Volkswagnove aplikacije.....	6
Slika 1.5: Naprave za obogateno resničnost.....	7
Slika 2.1: Konceptualni diagram optičnega naglavnega prikazovalnika.....	9
Slika 2.2: Optična naglavna prikazovalnika, ki ju je naredil Hughes Electronics.....	10
Slika 2.3: Konceptualni diagram video naglavnega prikazovalnika .....	10
Slika 2.4: Video naglavni prikazovalnik .....	11
Slika 2.5: Konceptualni diagram sistema obogatene resničnosti na osnovi monitorja .....	11
Slika 3.1: Tok dogodkov pri obogateni resničnosti z uporabo oznak .....	17
Slika 3.2: Sistemi oznak .....	19
Slika 3.3: Cybercode oznaki.....	20
Slika 3.4: SCR oznaki.....	20
Slika 3.5: IGD oznaki .....	21
Slika 3.6: HOM oznaki.....	21
Slika 3.7: Sistema oznak okroglih oblik.....	22
Slika 3.8: Oznaka programskega orodja ARToolKit .....	23
Slika 4.1: Prenos ARToolKita v druge programske jezike .....	28
Slika 4.2: Oznake knjižnice Studierstube Tracker: uokvirjena oznaka, razdeljena oznaka, pikčasta oznaka.....	29
Slika 4.3: Oznake knjižnice flare*tracker .....	32
Slika 4.4: Primerjava knjižnic za obogateno resničnost.....	32
Slika 5.1: Primer uporabe aplikacije obogatene resničnosti.....	33
Slika 5.2: Oznaka, ki upošteva omejitve knjižnice FLARToolKit.....	34
Slika 5.3: NyId oznaka, katere ID=10 .....	35
Slika 5.4: Uokvirjena oznaka .....	35
Slika 5.5: Oznaka, ki smo jo uporabili za meritve.....	37
Slika 5.6: Prva stran razglednice .....	38
Slika 5.7: Obrobljena razglednica .....	39
Slika 5.8: Razglednica z ID osnovano oznako .....	39
Slika 5.9: Razglednica z oznako, ki temelji na ujemanju predlog.....	40
Slika 5.10: Razglednica z uokvirjeno oznako .....	40
Slika 5.11: Zadnja stran razglednice z oznako zraven znamke .....	41
Slika 5.12: Zadnja stran razglednice z oznako v prostoru za pozdrave.....	42
Slika 5.13: Slika oznake, na kateri aplikacija prikaže 3D model gradu .....	43
Slika 5.14: Aplikacija obogatena razglednica .....	43

## Seznam tabel

Tabela 3.1: Pristopi sledenja s pomočjo računalniškega vida.....	15
Tabela 3.2: Pristopi sledenja s pomočjo senzorjev .....	16
Tabela 5.1: Največja razdalja kamere, pri kateri je oznaka prepoznana.....	36
Tabela 5.2: Lastne meritve oddaljenosti kamere za pravilno prepoznavo oznake.....	37
Tabela 5.3: Največja oddaljenost kamere za ID osnovane oznake .....	37
Tabela 5.4: Največja oddaljenost kamere za uokvirjene oznake .....	38

## Literatura

- [1] Ronald T. Azuma. A survey of augmented reality. Dostopno na: <http://www.cs.unc.edu/~azuma/ARpresence.pdf>, 2012
- [2] Paul Milgram, Haruo Takemura, Akira Utsumi, and Fumio Kishino. Augmented reality: A class of displays on the reality-virtuality continuum. Dostopna na: [http://etclab.mie.utoronto.ca/publication/1994/Milgram\\_Takemura\\_SPIE1994.pdf](http://etclab.mie.utoronto.ca/publication/1994/Milgram_Takemura_SPIE1994.pdf), 2012
- [3] Layar Reality Browser. Dostopno na: <http://www.layar.com>, 2012
- [4] Freek Uijtdewilligen: A framework for context-aware applications using augmented reality: A train station navigation proof-of-concept on Google Android. Dostopno na: [http://www.utwente.nl/ewi/trese/graduation\\_projects/2010/Uijtdewilligen.pdf](http://www.utwente.nl/ewi/trese/graduation_projects/2010/Uijtdewilligen.pdf), 2012
- [5] Elisa Ziegler: Real-time markerless tracking of objects on mobile devices. Dostopno na: [http://www.mobilelifecentre.org/upload/publication/140/original/BA\\_Ziegler-1\\_final.pdf](http://www.mobilelifecentre.org/upload/publication/140/original/BA_Ziegler-1_final.pdf), 2012
- [6] Zhang, X. Fronz, S. and Navab, N. Visual Marker Detection and Decoding in AR Systems: A Comparative Study, in “Proceedings of the 2002 International Symposium on Mixed and Augmented Reality”, Los Alamitos, CA:IEEE Computer Society, 2002, pp. 97-106
- [7] ALVAR. Dostopno na: <http://virtual.vtt.fi/virtual/proj2/multimedia/>, 2012
- [8] Charles B. Owen, Fan Xiao, and Paul Middlin. What is the best fiducial? In The First IEEE International Augmented Reality Toolkit Workshop, pages 98–105, Darmstadt, Germany, September 2002.
- [9] Y. Cho, J. Lee, and U. Neumann. Multi-ring color fiducial systems for scalable fiducial tracking augmented reality. Proc. of IEEE VRAIS, page 212, 1998.

- [10] L. Naimark and E. Foxlin. Circular data matrix fiducial system and robust image processing for a wearable vision-inertial self-tracker. Proceedings of the International Symposium on Mixed and Augmented Reality (ISMAR '02), pages 27–36, 2002.
- [11] A. Rice, A. Beresford, and R. Harle, “Cantag: an open source software toolkit for designing and deploying marker-based vision systems,” in Pervasive Computing and Communications, 2006.
- [12] L. Naimark and E. Foxlin, “Circular data matrix fiducial system and robust image processing for a wearable vision-inertial self-tracker,” in ISMAR, 2002.
- [13] Fiala, M.; "ARTag, a fiducial marker system using digital techniques," Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on , vol.2, no., pp. 590- 596 vol. 2, 20-25 June 2005
- [14] ARToolKit. Dostopno na:  
<http://www.hitl.washington.edu/artoolkit/>, 2012
- [15] ARToolKitPlus, Studierstube Tracker. Dostopno na:  
<http://handheldar.icg.tugraz.at/stbtracker.php>, 2012
- [16] ArUco. Dostopno na:  
<http://www.uco.es/investiga/grupos/ava/node/26>, 2012
- [17] FLARToolKit. Dostopno na:  
<http://www.libspark.org/wiki/saqoosha/FLARToolKit/en>, 2012
- [18] Generator oznak. Dostopno na:  
<http://flash.tarotaro.org/blog/2008/12/14/artoolkit-marker-generator-online-released/>, 2012



[19] ARToolworks. Dostopno na:

<http://www.artoolworks.com>, 2012

[20] Program za izdelavo oznak. Dostopno na:

<http://www.roarmot.co.nz/ar/>, 2012

[21] Program za generiranje vzorčnih datotek. Dostopno na:

<http://flash.tarotaro.org/blog/2009/07/12/mgo2/>, 2012