

UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY

DETEKCIA A ROZPOZNÁVANIE DOPRAVNÝCH
ZNAČIEK
BAKALÁRSKA PRÁCA

2017

Anikó Szabóová

UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY

DETEKCIA A ROZPOZNÁVANIE DOPRAVNÝCH
ZNAČIEK
BAKALÁRSKA PRÁCA

Študijný program: Informatika
Študijný odbor: 2508 Informatika
Školiace pracovisko: Katedra informatiky
Školiteľ: RNDr. Zuzana Černeková, PhD.

Bratislava, 2017

Anikó Szabóová



Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta: Anikó Szabóová
Študijný program: informatika (Jednoodborové štúdium, bakalársky I. st., denná forma)
Študijný odbor: informatika
Typ záverečnej práce: bakalárska
Jazyk záverečnej práce: slovenský
Sekundárny jazyk: anglický

Názov: Detekcia a rozpoznávanie dopravných značiek
Detection and recognition of traffic signs

Cieľ: Naštudovať metódy zaoberajúce sa rozpoznávaním dopravných značiek a s tým súvisiace metódy výberu vhodných príznakov ako farba, geometrické charakteristiky a iné, používané v skúmanej oblasti. Vybrať a implementovať jednu metódu. Otestovať aplikáciu a vyhodnotiť dosiahnuté výsledky.

Vedúci: RNDr. Zuzana Černeková, PhD.
Katedra: FMFI.KAI - Katedra aplikovanej informatiky
Vedúci katedry: prof. Ing. Igor Farkaš, Dr.
Dátum zadania: 18.10.2016

Dátum schválenia: 25.10.2016

doc. RNDr. Daniel Olejár, PhD.
garant študijného programu

študent

vedúci práce

Čestné prehlásenie:

Čestne prehlasujem, že túto bakalársku prácu som vypracovala samostatne s použitím uvedených zdrojov literatúry.

V Bratislave dňa:

.....
Anikó Szabóová

Pod'akovanie:

Ďakujem vedúcej bakalárskej práce RNDr. Zuzane Černekovej, PhD. za cenné rady a pripomienky pri vypracovávaní tejto práce.

Abstrakt

V súčasnej dobe môžeme pozorovať rýchly nárast motorových vozidiel v cestnej premávke. Aby sa zvýšila bezpečnosť v dopravnej infraštruktúre, vyvíjajú sa rôzne inteligentné bezpečnostné systémy.

Táto práca sa zaoberá problematikou detekcie a rozpoznávania dopravných značiek. Okrem opísania existujúcich techník a postupov je súčasťou aj popis implementácie konkrétnej aplikácie na tento účel. Na konci sú zhrnuté dosiahnuté výsledky získané pomocou tejto aplikácie.

Kľúčové slová: dopravné značky, spracovanie obrazu, detekcia, rozpoznávanie, MATLAB

Abstract

Nowadays, we can observe the rapid growth of motor vehicles in the traffic. Different intelligent security systems are being developed to increase the security in the transport infrastructure.

This document deals with detection and identification of the traffic signs. In addition to describing existing techniques and procedures, the description of implementation of a specific application for this purpose is also included. At the end, the results obtained by this application are summarized.

Keywords: traffic signs, image processing, detection, recognition, MATLAB

Obsah

Úvod	1
1 Prehľad problematiky	2
1.1 História dopravných značiek	2
1.2 Rozdelenie dopravných značiek	3
1.3 Dopravné značky v rôznych krajinách	4
1.4 Bezpečnostné systémy do auta	5
1.5 Rozpoznávanie značiek	6
1.5.1 Detekcia podľa farby	7
1.5.2 Detekcia podľa tvaru	7
1.5.3 Detekcia na základe strojového učenia	8
2 Základné pojmy	9
2.1 Farebné modely	9
2.1.1 Farebný model RGB	9
2.1.2 Farebný model HSV	10
2.1.3 Farebný model CIE $L^*a^*b^*$	10
2.2 Odstraňovanie šumu	12
2.3 Hranové detektory	13
2.4 Morfológické operácie	13
2.5 Identifikácia tvaru	14
2.5.1 Kruhovitosť	14
2.5.2 Houghova transformácia	14

2.6	Template matching	14
2.6.1	Harrisov rohový detektor	15
2.6.2	SIFT	15
2.6.3	SURF	17
2.6.4	MSER	17
3	Postup pri riešení problematiky	18
3.1	Snímanie obrazu	18
3.2	Predspracovanie obrazu	19
3.3	Segmentácia významných oblastí	19
3.4	Detekcia dopravnej značky	20
3.5	Rozpoznávanie konkrétnej značky	20
4	Návrh riešenia	21
4.1	Získavanie obrazu	21
4.2	Nástroje	22
4.2.1	MATLAB	22
4.2.2	Wondershare Video Editor	22
4.3	Predspracovanie obrazu	23
4.4	Segmentácia modrých oblastí	23
4.5	Morfologické operácie	23
4.6	Detegovanie tvaru	24
4.7	Databáza značiek	24
4.8	Druhá fáza - detekcia	25
4.9	Užívateľské prostredie	25
5	Implementácia	27
6	Výsledky a testovanie	34
6.1	SIFT	35
6.2	SURF	35
6.3	Porovnávanie pixelov	36

<i>OBSAH</i>	vii
Záver	39
Prílohy	44

Zoznam obrázkov

1.1	Rozdiely v značkách medzi štátmi ktoré podpísali Viedenskú konvenciu (Chorvátsko, Nemecko, Španielsko, Francúzsko, Taliansko, Poľsko) [2]	5
2.1	RGB jednotková kocka [14]	10
2.2	HSV ihlan: H - farebný tón, S - sýtosť a V - jasová hodnota [5]	11
2.3	CIE L*a*b* farebný model [9]	12
4.1	Ukážka použitých šablón [30]	24
4.2	Postup pri riešení problematiky	26
5.1	Segmentovanie pomocou RGB, CIE L*a*b* a HSV farebného modelu	28
5.2	Obrázok pred a po úprave morfológickými operáciami	29
5.3	Významné body získané pomocou rôznych algoritmov	30
5.4	Používateľské rozhranie č.1	32
5.5	Používateľské rozhranie č.2	33
5.6	Používateľské rozhranie č.3	33
6.1	Príklad zlej klasifikácie	36
6.2	Zle segmentovaná oblasť kvôli oblohe	37
6.3	Zle segmentovaná oblasť kvôli reklamnej tabuli	37

Zoznam tabuliek

6.1	Zdetegované dopravné značky z testovacieho videa	35
6.2	Úspešnosť klasifikácie pomocou algoritmu SIFT	35
6.3	Úspešnosť klasifikácie pomocou algoritmu SURF	36
6.4	Úspešnosť klasifikácie pomocou porovnávania pixelov	38
6.5	Úspešnosť klasifikácie pri kombinácii dvoch techník	38
6.6	Celková úspešnosť rozpoznávania	38

Úvod

Dopravné značky sú dôležitým elementom dopravnej infraštruktúry. Znalosť ich významu a dodržiavanie ich pravidiel je pre účastníka cestnej premávky nevyhnutná. Keďže hrajú takú dôležitú úlohu, je potrebné, aby boli výrazné a jednoznačné. Pre tento účel majú dopravné značky špecifický tvar, farbu a znakovú informáciu.

V posledných rokoch sa začali vyvíjať systémy, ktoré majú za úlohu zvýšiť bezpečnosť vodiča počas jazdy. Medzi tieto systémy patria aj systémy na detekciu a rozpoznávanie dopravných značiek. Vďaka špecifickým vlastnostiam týchto tabúl je ich možné identifikovať v prírode.

Cieľom tejto bakalárskej práce je popísať metódy zaoberajúce sa rozpoznávaním dopravných značiek a s tým súvisiace metódy. Okrem toho implementovať jednu z metód a vyhodnotiť dosiahnuté výsledky. Väčšina prác zaoberajúca sa s touto problematikou rieši detekciu červených dopravných značiek. Preto naším cieľom je vytvoriť aplikáciu, ktorá dokáže rozpoznať modré dopravné značky, ktoré sa oveľa ťažšie segmentujú.

V prvej kapitole opíšeme základné rozdelenie značiek a spôsoby detekcie značiek. V ďalšej kapitole si vysvetlíme základné pojmy pri riešení tejto problematiky. V tretej kapitole uvedieme postup riešenia a následne v štvrtej kapitole opíšeme návrh riešenia. Piata kapitola predstavuje implementovanú aplikáciu na detekciu modrých dopravných značiek. V poslednej kapitole zhrnieme dosiahnuté výsledky.

Kapitola 1

Prehľad problematiky

Dopravné značky sú dôležitou súčasťou cestnej infraštruktúry. Kontrolujú dopravu, udávajú smer a rýchlosť jazdy, informujú účastníkov cestnej premávky o vzdialenosti a smere cieľa, upozorňujú na nebezpečné úseky atď.

Znalosť významu všetkých dopravných značiek je pre jazdu nevyhnutná. Existujú aplikácie, ktoré vedia rozpoznať značky zo snímaného videa a upozorňujú vodiča na aktuálne dopravné značenie.

1.1 História dopravných značiek

Rast populácie, objavenie motorových vozidiel a rozvoj automobilizmu viedli k výraznému vývoju verejnej dopravy na konci devätnásteho storočia. Stále rastúce množstvo dopravných prostriedkov bolo potrebné nejakým spôsobom riadiť. Prvá zmluva o úprave cestnej premávky bola podpísaná v roku 1909 v Paríži. Okrem podmienok, ktoré museli splňať vodiči aj motorové vozidlá, zaviedli štyri výstražné dopravné značky [3].

Najdôležitejšia dohoda o dopravnom označení vznikla v roku 1949 v Ženeve. Táto zmluva tvorí základ súčasného európskeho systému. Dopravné značky definuje ako znakové symboly, ktoré neobsahujú slová, aby boli zrozumiteľné aj pre negramotných ľudí a cudzincov [26].

V roku 1968 sa konala Viedenská konvencia, ktorú podpísala väčšina

európskych krajín, vrátane Československa. Táto zmluva bola rozšírením Ženevskej zmluvy [27].

1.2 Rozdelenie dopravných značiek

Dopravné značky delíme na zvislé a vodorovné. Zvislé dopravné značky delíme ďalej na výstražné, zákazové, príkazové, informatívne prevádzkové, informatívne smerové, informatívne iné a na značky upravujúce prednosť v jazde. Z pohľadu spracovávania obrazu ich delíme podľa farebnosti a geometrického tvaru.

Rozdelenie podľa farebnosti:

- **Červené** - najdôležitejšie a zároveň najbežnejšie dopravné značky sú červenej farby. Táto farba je najlepšie rozoznateľná vo farebnom prostredí, preto sú výstražné, zákazové a značky upravujúce prednosť v jazde červené.
- **Oranžové a žlté** - podobne teplé farby sa používajú na informovanie vodičov o dočasných alebo trvalých nebezpečenstvách. Dočasné upozornenia, prevažne z dôvodu výstavby, sú značky oranžovej farby s čiernym textom. Na trvalé upozornenia sa používa čierny text, alebo znak na žltom podklade pre maximálnu čitateľnosť a viditeľnosť.
- **Modré** - tieto značky obsahujú menej zásadné informácie ako červené. Toto sfarbenie je menej rušivé, aby neodviedlo pozornosť vodičov. Charakter týchto značiek je informatívny. Väčšina príkazových a informatívnych dopravných značiek je modrej farby.
- **Zelené** - zelené značky obsahujú informácie, ktoré závisia čisto na zámere vodiča (napríklad výjazdové cesty na diaľnici). Patria sem informatívne smerové tabule.

- **Čierne a biele** - primárny cieľ čiernych a bielych tabúl je jasne a zreteľne informovať vodiča bez rozptyľovania. Patria sem dodatkové tabule, ktoré obsahujú čierny text alebo znak na bielom podklade.

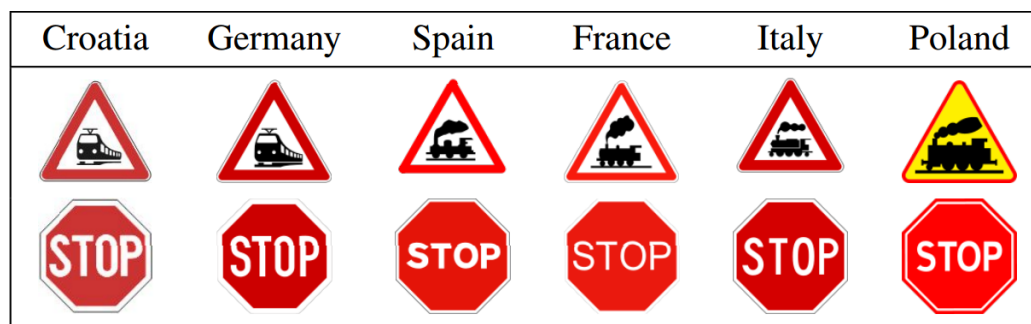
Rozdelenie podľa tvaru:

- **Kruhové značky** - označujú zákaz alebo obmedzenie (zákazové, príkazové značky).
- **Trojuholníkové značky** - varujú pred nebezpečenstvom (výstražné značky).
- **Osemuholník** - tento tvar má iba jediná značka, značka Stoj.
- **Obdĺžnikové alebo štvorcové značky** - obsahujú doplnkové informácie (informatívne značky a dodatkové tabule).
- **Kosoštvorec** - značka označujúca hlavnú cestu, alebo koniec hlavnej cesty.

1.3 Dopravné značky v rôznych krajinách

Väčšina európskych krajín a mnoho ďalších krajín podpísali Viedenskú konvenciu, ktorá obsahuje striktné predpísané pravidlá, ako majú dopravné značky vyzerieť. Avšak stále sa nájdu rozdiely medzi značkami týchto štátov. Tieto odlišnosti sú pre človeka nepodstatné, ale môžu spôsobovať ťažkosti pre algoritmus na rozpoznávanie značiek (Obr. 1.1).

V Spojených štátoch amerických, v Austrálii a na Novom Zélande používajú značky s textom. Podobná štruktúra ako predstavuje Viedenská konvencia bola zavedená v Strednej a v Južnej Amerike a v niektorých ázijských krajinách. V afrických štátoch sa používa systém, ktorý je kombináciou symbolov a textu [26].



Obr. 1.1: Rozdiely v značkách medzi štátmi ktoré podpísali Viedenskú konvenciu (Chorvátsko, Nemecko, Španielsko, Francúzsko, Taliansko, Poľsko) [2]

1.4 Bezpečnostné systémy do auta

Kvôli zvýšeniu bezpečnosti cestnej premávky sa začali vyvíjať systémy, ktoré pomáhajú vodičovi pri riadení motorového vozidla. Tieto systémy sa spoločne nazývajú ADAS (*Advanced Driver Assistance Systems*) - pokročilé asistenčné systémy vodiča [8]. Bezpečnostné funkcie boli vyvinuté tak, aby dokázali predísť nehodám, tým, že upozornia vodiča na potencionálne nebezpečenstvo. Niektoré dokážu prebrať kontrolu nad vozidlom z dôvodu zabránenia zrážke.

ADAS prvky môžu byť zabudované do auta, alebo sú k dispozícii ako doplnkové funkcie. Patria sem funkcie ako automatické parkovanie, parkovacie senzory, adaptívny tempomat (*Adaptive cruise control (ACC)*), odhalenie ospalosti vodiča (*Driver drowsiness detection*), alebo blokovanie možnosti naštartovania auta pri detekcii alkoholu v dychu (*Ignition interlock device*). Ďalším príkladom ADAS systémov je aplikácia, ktorá dokáže upozorniť vodiča na aktuálne dopravné značenie. Je to takzvaný TSR systém, skratka z anglického názvu *Traffic Sign Recognition* - systém na rozpoznávanie dopravných značiek.

1.5 Rozpoznávanie značiek

Hlavným cieľom TSR systémov je zvýšiť bezpečnosť jazdy vodiča. Preto sa vývojári týchto systémov skôr zameriavajú na výstražné, zákazové a na značky upravujúce prednosť v jazde. Z nich hlavne na značky udávajúce najvyššiu povolenú rýchlosť, prednosť v jazde, zákaz predbiehania a na značku Stoj.

Tieto systémy sa dajú využiť aj na ukladanie dopravných značiek do mapových súborov podľa GPS súradníc (*Global Positioning System*). Následne sa tieto údaje použijú pri hľadaní optimálnej trasy v navigačných systémoch, alebo na úradoch na zapísanie všetkých dopravných značiek v regióne.

Výskumy týkajúce sa detekcie dopravných značiek sú ťažko porovnateľné, pretože účel jednotlivých systémov je rôzny. Metódy na detekciu sú rozdelené do troch kategórií: rozpoznávanie na základe **farby**, **tvaru** alebo **strojového učenia** [2].

Faktory v ktorých sa jednotlivé metódy líšia sú:

- Typy vstupu: video alebo obraz.
- Oblasti použitia: metóda je buď použiteľná len na jednu triedu dopravných značiek, alebo na viacero tried.
- Podmienky natáčania: systém môže byť určený na používanie výnimočne len počas denného svetla, počas noci alebo počas celého dňa. Okrem toho je ešte otázne, či systém funguje aj za nepriaznivých poveternostných podmienok ako je dážď, sneh alebo hmla.
- Typy senzora: video môže byť nasnímané s vysokým alebo s nízkym rozlíšením, ďalej rozhoduje, či sa pracuje s farebným alebo s čiernobielym obrazom, či sa dáta zbierajú len pomocou jednej kamery atď.
- Požiadavky na spracovanie: detekcia buď prebieha v reálnom čase, alebo je určená na offline spracovanie.

- Povaha problému: sú akceptovateľné drobné nepresnosti, alebo je potrebné aby aplikácia fungovala bezchybne.

Na základe týchto faktorov sa vyberie najvhodnejšia metóda pre riešenie danej oblasti. Napríklad rozpoznávanie na základe farby je bezpredmetné v prípade, ak pracujeme s čiernobielym obrazom. Na druhej strane, táto metóda môže byť veľmi užitočná, ak sa snažíme rozpoznať dopravné značky z videa vo vysokom rozlíšení, s farebným obrazom zachyteným za denného svetla s kvalitnou kamerou. Detekcia pomocou tvaru nefunguje v prípade ak používame kameru s prekrývaním (*interlacing*). Metódy pomocou strojového učenia vyhovujú, ak máme veľa označených dát, avšak v prípade, keď dáta nemáme k dispozícii, táto metóda je nevyužiteľná.

1.5.1 Detekcia podľa farby

Najbežnejšia metóda pri rozpoznávaní dopravných značiek je detekcia na základe farby. Značky boli navrhnuté tak, aby sa dali ľahko odlíšiť od okolitej prírody. Táto ich vlastnosť umožňuje používať túto metódu.

Najprv sa pomocou prahovania a segmentácie nájdu oblasti na snímke, kde sa nachádza žiadúca farba. Tieto časti sú buď hneď vyhlásené za dopravnú značku, alebo ešte prechádzajú ďalšou fázou overovania (napríklad či sa táto oblasť nachádza na pravej časti snímky, ako značky vedľa cesty). Hlavná nevýhoda tohto spôsobu spočíva v tom, že výsledky príliš závisia od poveternostných podmienok, tieňov, keď sa osvetlenie scény môže značne líšiť. Farebný model RGB je veľmi citlivý na osvetlenie, preto je pri tejto metóde vhodnejšie použiť segmentáciu pomocou iného farebného modelu, napríklad modelu HSI alebo L^*a^*b .

1.5.2 Detekcia podľa tvaru

Pri tejto metóde sa najčastejšie používa Houghova transformácia. Pomocou tejto techniky vieme nájsť ľubovoľné tvary v obraze. Pri implementácii je potrebné poznať parametrický popis tvaru hľadaného objektu. Preto sa využíva

hlavne pre detekciu jednoduchých objektov v obraze, ktorých hranice vieme určiť jednoduchými krivkami ako sú priamky, kružnice, elipsy, trojuholníky, štvorce atď.

1.5.3 Detekcia na základe strojového učenia

V predošlých metódach sa do algoritmu vkladali nejaké znalosti (farba alebo tvar dopravných značiek), avšak tieto poznatky by sa dali získať aj strojovým učením. P. Viola a M. J. Jones predstavili algoritmus, ktorý dokázal spoľahlivo zdetegovať rôzne objekty v reálnom čase. Detektor bol natrémovaný pomocou sady pozitívnych a negatívnych príkladov. Pôvodne bol tento algoritmus určený na detekciu tváre, ale neskôr sa začal používať aj v iných oblastiach, napríklad na detekciu dopravných značiek [21].

Kapitola 2

Základné pojmy

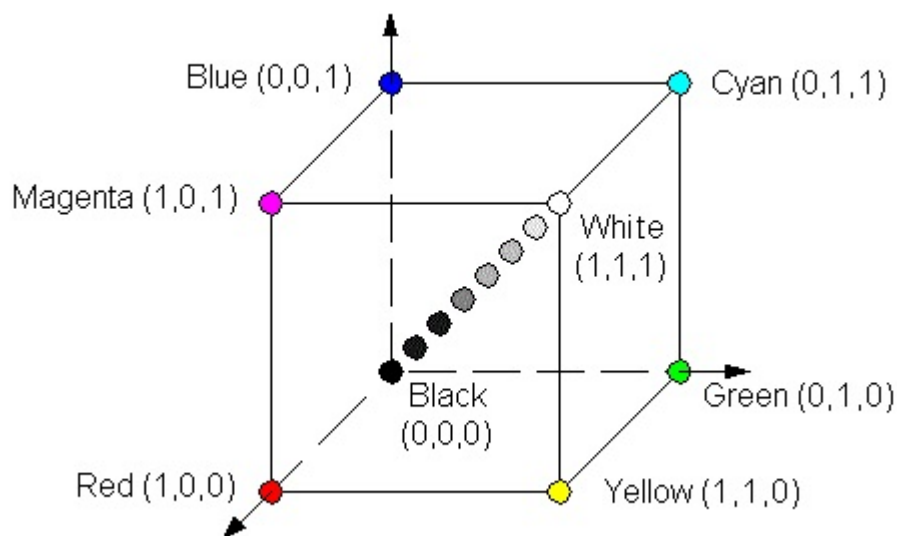
2.1 Farebné modely

2.1.1 Farebný model RGB

RGB (Red, Green, Blue) - je základný a najpoužívanější farebný model v počítačovej grafike. Základnou vlastnosťou tohto modelu je aditívne skladanie farieb. Základné zložky pre tento model sú R - červená, G - zelená a B - modrá farba. Pomocou týchto zložiek vieme vytvoriť ďalšie farby, pričom najnižšia hodnota reprezentuje čiernu a najvyššia hodnota reprezentuje bielu farbu. Rozsah týchto hodnôt je na intervale $\langle 0,1 \rangle$, alebo sa často používa rozsah 0 až 255.

Farebný model RGB sa reprezentuje ako jednotková kocka (Obr. 2.1). Vo vrcholoch tejto kocky sa nachádzajú základné farby: červená $[1,0,0]$, zelená $[0,1,0]$, modrá $[0,0,1]$, čierna $[0,0,0]$, biela $[1,1,1]$, žltá $[1,1,0]$, fialová $[1,0,1]$ a tyrkysová $[0,1,1]$. Na diagonále medzi čiernou a bielou ležia šedotónové farby.

Hlavnou nevýhodou tohto modelu je to, že sa ťažko určujú jednotlivé farby a ich odtiene.



Obr. 2.1: RGB jednotková kocka [14]

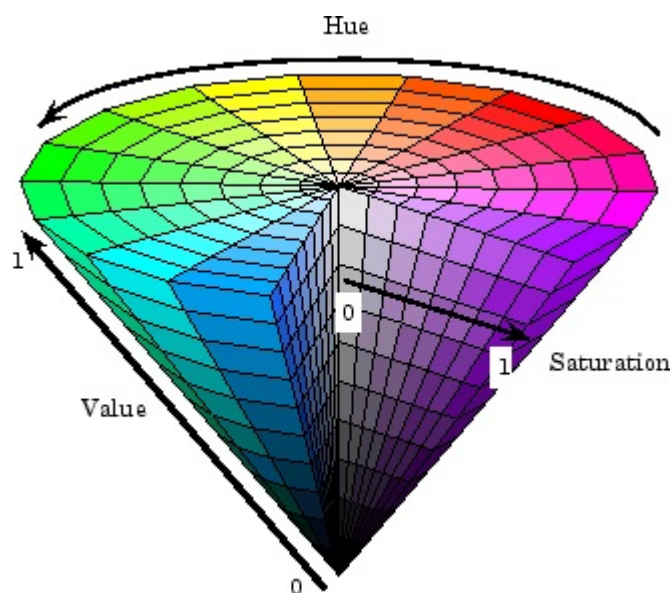
2.1.2 Farebný model HSV

Tento model definuje farbu pomocou troch zložiek, ktoré však už nepredstavujú základné farby a sú bližšie k ľudskému vnímaniu farieb. Hlavnými parametrami modelu HSV sú H - farebný tón, S - sýtosť a V - jasová hodnota.

Na priestorové zobrazenie modelu sa používa ihlan so šesťuholníkovou podstavou (Obr. 2.2). Súradnice S a V sa menia od 0 po 1, súradnica H reprezentuje uhol, ktorý nadobúda hodnotu z intervalu $(0^\circ, 360^\circ)$. Vrchol ihlana predstavuje čiernu farbu a stred šesťuholníka bielu farbu, keďže jas rastie smerom k podstave. Sýtosť odpovedá vzdialenosti bodu od osi ihlana. Dominantné farby tak ležia na plášti, čisté farby sú na obvodě podstavy [13] [28].

2.1.3 Farebný model CIE $L^*a^*b^*$

CIE $L^*a^*b^*$ je farebný model ktorý bol navrhnutý na CIE konferencii (*Commission internationale de l'éclairage*) v roku 1976 ako štandard základných



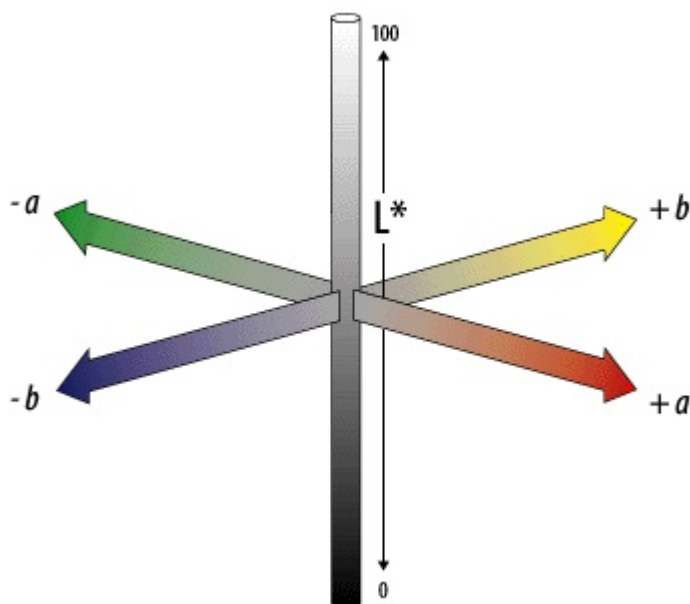
Obr. 2.2: HSV ihlan: H - farebný tón, S - sýtosť a V - jasová hodnota [5]

farieb. Pri definovaní tohto modelu sa vychádzalo z hlavného CIE XYZ farebného modelu z roku 1931. Ten definuje všetky farby váženým súčtom troch základných farieb. Každá skutočná farba je určená množstvom základných farieb potrebných na jej vytvorenie. V normalizovanom tvare ich zapíšeme ako:

$$x = \frac{X}{X + Y + Z} \quad y = \frac{Y}{X + Y + Z} \quad z = \frac{Z}{X + Y + Z}$$

Pretože platí $x+y+z = 1$, na vyjadrenie každej farby nám stačia dve ľubovoľné zložky. Pomocou zložky x a y vieme reprezentovať všetky farby dvojrozmerným diagramom.

Preto je farebný model CIE $L^*a^*b^*$ reprezentovaný vo forme kocky, kde zložka L^* (*lightness*) reprezentuje jas a zložky a^* b^* reprezentujú farbu. Hodnota $L^* = 0$ predstavuje najtmavšiu čiernu a hodnota $L^* = 100$ je najsvetlejšia biela. Farebné kanály a^* a b^* reprezentujú odtiene šedej pri hodnote $a^*=0$ a $b^*=0$. Kontrastné farby červená a zelená sa nachádzajú na osi a^* ,

Obr. 2.3: CIE $L^*a^*b^*$ farebný model [9]

pričom v záporných hodnotách a^* sa nachádzajú odtiene zelenej a v kladných hodnotách odtiene červenej farby. Na osi b^* sa od záporných ku kladným hodnotám nachádzajú odtiene modrej farby k odtieňom žltej [13] (Obr. 2.3).

2.2 Odstraňovanie šumu

Odstraňovanie šumu je veľmi užitočná operácia pri digitálnych upravách obrazu. Pri tejto operácii zaleží na spôsobe získania obrazu a na charakteristike šumu. Ak máme k dispozícii viacero exemplárov toho istého obrazu, šum odstránime priemerovaním. Výsledkom toho bude, že každý výsledný pixel bude priemerom alebo mediánom všetkých pixelov z tej pozície. Avšak obvykle máme k dispozícii len jediný obrázok. Vtedy sa šum odstraňuje na základe charakteristiky okolia každého pixelu. Najčastejšie sa používajú filtre ktoré šum rozmažú.

Najjednoduchšia filtrovacia metóda počíta hodnotu pixelu ako priemer

z jeho okolia. Táto metóda sa nazýva priemerovanie. Ďalšou možnosťou je mediálny filter, kde hodnotu pixelu nahradíme hodnotou mediánu z okolitých pixelov. Gaussov filter vyhladzuje obrázok podobne ako pri priemerovaní, ale využíva sa pri tom korelácia na základe masky, ktorá sa označuje ako konvolučné jadro. Toto jadro je reprezentované maticou, ktorá nadobúda hodnoty gaussovej krivky, vďaka čomu vieme získať presnejší a lepší výsledok ako pri priemerovaní [13].

2.3 Hranové detektory

Pri hľadaní nejakého objektu v obraze sa často využívajú hranové detektory. Pomocou nich vieme vytvoriť obrysy útvarov, ktoré sa nachádzajú na obraze. Algoritmy na detekciu hrán fungujú na predpoklade, že hrany sa nachádzajú tam, kde je veľký rozdiel v intenzite jasú.

Najpoužívanejšie hranové detektory sú Cannyho a Sobelov algoritmus [20].

2.4 Morfologické operácie

Pri hľadaní objektov v obraze je potrebné upraviť obraz tak, aby sme odstránili čo najviac nepotrebných informácií a zároveň uviedli do pozornosti tie dôležité. Na to slúžia aj morfologické operácie ako erózia (*erosion*) a dilatácia (*dilation*).

Dilatácia sa väčšinou používa na binárne obrazy, avšak v niektorých prípadoch funguje aj na šedotónové obrazy. Základným cieľom tohto operátora je postupne rozšíriť hranice pixelov a tak zväčšiť objekt a zároveň zaplniť menšie diery a úzke zálivy.

Erózia sa taktiež aplikuje hlavne na binárne obrazy. Na rozdiel od dilatácie, erózia znižuje veľkosť objektu tým, že odoberie hraničné pixely objektu. Vďaka tejto operácii dostaneme objekt so zvýraznenými (väčšími) dierami a odstránime malé objekty.

Otvorenie (*opening*) a uzavretie (*closing*) sú zložené operácie z erózie a dilatácie. Otvorenie získame ak najprv aplikujeme na objekt eróziu a následne dilatáciu. Uzavretie dosiahneme ak použijeme tie isté morfológické operácie, len v opačnom poradí [7].

2.5 Identifikácia tvaru

2.5.1 Kruhovitost'

Na identifikáciu tvaru objektov sa používa výpočet na kruhovitosť. Na základe výsledku sa tak vieme rozhodnúť, či je aktuálny objekt v tvare kruhu, štvorca alebo trojuholníka. Hodnotu kruhovitosti získame z pomeru obsahu objektu a z obsahu kruhu s rovnakým obvodom.

$$C = \frac{4\pi A}{O^2}$$

kde C je kruhovitosť, A je obsah objektu a O je obvod kruhu [12].

Kruh je najviac kruhovitý objekt, keďže s daným obvodom uzatvára čo najväčšiu plochu. Preto pre kruh platí $C = 1$; pre 1x2 obdĺžnik $C = 0,698$; pre rovnostranný trojuholník $C = 0,605$; pre štvorec $C = 0,785$ a pre šesťuholník $C = 0,907$.

2.5.2 Houghova transformácia

Houghova transformácia pôvodne slúžila na hľadanie priamok a úsečiek v obraze. Neskôr bola funkcionálna rozšírená aby vedela identifikovať aj jednoduché tvary. Najčastejšie sa používa na detekciu kruhov a elipsy [23].

2.6 Template matching

Porovnávanie šablón (*Template matching*) je bežný problém v počítačovom videní.

Existuje mnoho algoritmov, ktoré riešia túto problematiku. Hlavným zámerom algoritmov je nájsť významné body v obraze, ktoré sa porovnávajú s významnými bodmi zo šablóny. Ak je dostatočne vysoká zhoda významných bodov z oboch vstupov, tak sa na obrázku pravdepodobne nachádza objekt zo šablóny.

Detektor slúži na to, aby našiel významné body v obraze a deskriptor opíše zaujímavý bod a jeho okolie príznakovým vektorom. V prípade, ak vstupy majú podobné vlastnosti (rozsah, orientácia atď.), na hľadanie podobných objektov postačia aj rohové detektory, inak je potrebné použiť algoritmus, ktorý počíta s rôznymi vlastnosťami obrazu.

2.6.1 Harrisov rohový detektor

Harrisov rohový detektor (*Harris corner detektor*) bol prvý krát publikovaný v roku 1988 [10]. Je jeden z najčastejšie používaných rohových detektorov aj vďaka tomu, že je dostatočne rýchly.

Významné body sa určia tak, že metóda hľadá miesta v obraze, kde sa mení gradient svetla v dvoch smeroch. Výhodou tohto detektoru je to, že je rotačne invariantný. To znamená, že natočenie nemá vplyv na vyhľadávanie významných bodov. Keďže je výpočet gradientu veľmi náchylný na šum, pri tomto algoritme sa používa Gaussova funkcia na odstránenie šumu.

2.6.2 SIFT

SIFT (*Scaleable Invariant Feature Transform*) je škálovo a rotačne invariantný detektor a deskriptor zaujímavých bodov.

Hlavnou motiváciou pre vytvorenie SIFT deskriptoru bolo to, že Harrisov rohový detektor nie je škálovo invariantný. V dôsledku toho bol Lowe-ovým hlavným cieľom vytvoriť deskriptor, ktorý dokáže pracovať s typickými zmenami v zobrazovaní, ako je škálovanie, rotácia, osvetlenie, uhol pohľadu a iné.

V prvom kroku SIFT generuje niekoľko oktáv pôvodného obrazu. V každej oktáve je veľkosť obrázku polovičná v porovnaní s predchádzajúcou oktávou.

V oktáve sú obrázky postupne rozmazané Gaussovým operátorom:

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

kde hodnota σ je štandardná odchýlka. Počet oktáv a počet obrázkov v oktáve závisí od veľkosti vstupného obrázka. Tieto hodnoty sa dajú zvoliť, pričom tvorca SIFT-u navrhuje pre ideálne fungovanie algoritmu štyri oktávy s piatimi levelmi rozmazania.

V ďalšom kroku sa v každej oktáve vygeneruje $n-1$ obrázkov, ktoré predstavujú rozdiel Gausiánov (*Difference of Gaussians - DoG*). Tento rozdiel je približne ekvivalentný k Laplaciánu z Gausiánu (*Laplacian of Gaussian - LoG*). Pomocou Laplaciánu z Gausiánu sa dajú určiť kľúčové body, avšak je výpočtovo veľmi náročný proces. Gausiánov rozdiel získame odčítaním dvoch nasledujúcich obrázkov.

Potom sa určia lokálne maximá a minimá. V okolí $3 \times 3 \times 3$ sa porovnávajú hodnoty pixelov a ak aktuálny pixel má najväčšiu alebo najnižšiu hodnotu s pomedzi 26 pixelov, ktorými sa porovnáva, tak je to lokálny extrém. Okolie $3 \times 3 \times 3$ predstavuje okolie 3×3 okolo skúmaného bodu a okolie 3×3 na predošlom a nasledujúcom obrázku.

Týmto spôsobom sa získa príliš veľa bodov na to, aby sme ich mohli označiť za významné body. Preto je potrebné odstrániť tie, ktoré ležia na hranách alebo majú nízky kontrast.

Ďalším krokom je priradenie orientácie všetkým významným bodom. Orientáciu získame pomocou histogramu a malého regiónu okolo bodu. Pomocou histogramu sa identifikujú najvýznamnejšie orientácie gradientu. Okolo významných bodov vytvoríme oblasť veľkosti 16×16 pixelov. Túto oblasť rozdelíme na šestnásť 4×4 regiónov. Z každého regiónu sa vygeneruje histogram, ktorého hodnoty budú od 0 po 360, rozdelené na osem úsekov. Tieto hodnoty predstavujú orientáciu gradientu v danom bode. Takto sa získa 128 hodnôt pre jeden významný bod [15] [16].

2.6.3 SURF

Tento algoritmus je založený na rovnakom princípe ako SIFT, ale používa inú schému, vďaka čomu je rýchlejší. Podobne ako SIFT, aj SURF (*Speeded Up Robust Features*) počíta s oktávami a levelmi, ale narozdiel od neho, SURF postupne zväčšuje pôvodný obrázok. Namiesto počítania Gaussiánov pre získanie aproximálnej hodnoty LoG, SURF používa box filtre na získanie približnej hodnoty Gaussiánov druhej parciálnej derivácie. Na získanie významných bodov používa potlačenie nemaximálnych hodnôt v okolí $3 \times 3 \times 3$ a determinanty Hessianových matic. Pre významné body sa vypočíta deskriptor pomocou susedných bodov. Orientácia bodu sa určí pomocou odozvy Haarovho waveletu, ktorej sa prideli váha na základe Gaussovho kruhového okna. Okolo významného bodu sa vytvorí oblasť 4×4 , kde sa podobným spôsobom určí orientácia.

Vo výsledku deskriptor obsahuje 64 hodnôt pre jeden významný bod [1].

2.6.4 MSER

MSER (*Maximally Stable Extremal Regions*) je metóda pre detekciu zaujímavých oblastí v obraze. Algoritmus oddelí z obrazu stabilné prepojené regióny, ktoré zostávajú rovnaké pri aplikovaní rôznych prahových hodnôt.

Na začiatku sa zvolí prahová hodnota. Všetky pixely pod danou prahovou hodnotou budú biele a ostatné čierne. Postupným zvyšovaním prahovej hodnoty sa biele škvrny zväčšujú a vytvárajú prepojené oblasti až kým celý obrázok nebude biely. Prepojené stabilné regióny predstavujú miestne extrémny, čo znamená, že všetky pixely z regiónu majú buď nižšiu alebo vyššiu intenzitu ako pixely za hranicou regiónov. Vybrané regióny sa označia elipsou a sú vyhlásené za významné oblasti [17].

Kapitola 3

Postup pri riešení problematiky

Väčšina algoritmov na detekciu a rozpoznávanie dopravných značiek postupuje podľa nasledujúcich krokov:

1. snímanie obrazu
2. predspracovanie obrazu
3. segmentácia významných oblastí
4. detekcia dopravnej značky
5. rozpoznávanie konkrétnej značky

3.1 Snímanie obrazu

Algoritmy na detekciu a rozpoznávanie dopravných značiek pracujú so sadou obrázkov alebo s videom. V niektorých autách sa nachádzajú zabudované videokamery, ktoré zachytávajú obraz ak je vozidlo v pohybe. Pre niektoré značky áut sú TRS systémy už súčasťou výbavy a na palubnej doske sa objavujú jednotlivé dopravné značky, ktoré sa nachádzajú práve popri ceste. Ďalšou možnosťou je získať video z externých kamier, ktoré sa požívajú

v autách bez stavanej kamery. Okrem toho, video môžeme nasnímať aj mobilným telefónom, ktoré sú kvôli navigačným systémom často umiestené na čelnom skle, alebo nad palubnou doskou.

3.2 Predspracovanie obrazu

Predspracovanie obrazu je štandardný krok pri riešení problémov v počítačovom videní. V tejto fáze sa upravuje obraz tak, aby sa pri segmentácii určili čo najpresnejšie úseky. Aplikujú sa operácie, ktoré pomáhajú prekonať nedokonalosti snímacieho zariadenia, napríklad odstraňujú šum alebo upravujú farby. Najpoužívanejšia operácia pri algoritmoch na detekciu dopravných značiek je gaussov filter, ktorý rozostreje obrázok a odstraňuje malé detaily. Taktiež používaný, ale náročnejší na výpočet je mediánový filter, ktorý okrem odstránenia šumu aj zvýrazňuje ostré hrany [23].

3.3 Segmentácia významných oblastí

Po predspracovaní je ďalším krokom rozdelenie upraveného obrázka na významné časti (oblasti záujmu) a na pozadie (zvyšnú oblasť). Na segmentáciu sa bežne používajú metódy na základe farebnej informácie alebo na základe tvaru. Keďže dopravné značky sú štandardizované čo sa týka tvaru i farieb, obe metódy sú vhodné. Bežnejšou metódou na rozdelenie obrázka je segmentácia na základe farby, pričom treba rátať s rôznymi poveternostnými podmienkami. Preto pri zadávaní prahu treba vymedziť širšiu škálu, ako len konkrétnu farebnú hodnotu, ktorou sú definované dopravné značky. Podľa štúdie [28] najvhodnejším farebným modelom na segmentáciu je model HSV.

Pri segmentácii na základe tvaru sa hľadajú jednoduché tvary v obraze ako kruh, trojuholník alebo obdĺžnik. Na nájdenie týchto tvarov sa najbežnejšie používa Cannyho hranový detektor [22], alebo Houghova transformácia [4]. Pri tejto metóde je nutné počítať aj s deformovanými tvarmi kruhu, trojuholníka a štvorca, keďže sa značky snímajú z rôzneho uhla.

Kvôli výhodám a nevýhodám oboch spôsobov sa v tomto kroku často kombinujú obe metódy.

3.4 Detekcia dopravnej značky

V tomto kroku sa zo segmentovaného obrazu vyberú konkrétne oblasti, kde by sa mohla dopravná značka nachádzať a následne sa určí typ značky (kruhovú, trojuholníkovú alebo štvorcovú značku). Ak v predošlej časti sa zvýraznili hrany, prebehne analýza týchto hrán a na základe toho sa určujú tvary. Ďalšou metódou je Houghova transformácia, ktorá dokáže hľadať priamky a kruhy v obraze pomocou parametrického vyjadrenia týchto útvarov [11].

Na určenie tvaru je možné použiť aj výpočet na kruhovitosť. Pomocou nej vieme definovať nielen kruhové útvary, ale aj štvorcové a trojuholníkové.

3.5 Rozpoznávanie konkrétnej značky

Po rozdelení kandidátov na kategórie sa pre každý kandidát určí prislúchajúca dopravná značka. V niektorých prípadoch sa tento krok zlučuje s detekciou, v iných je na základe detekcie obmedzená množina možných výsledkov. To znamená, že pri rozdeľovaní kandidátov na rôzne tvary, rozpoznávanie prebehne len na konkrétnych piktogramoch. Často sa určujú ďalšie vlastnosti kandidátov pred klasifikáciou, aby samotná detekcia prebehla v čo najužšom kruhu. Výsledná dopravná značka sa určí pomocou porovnávania kandidáta s piktogramami. Na to existujú rôzne algoritmy, ktoré nájdu významné body a porovnávajú ich navzájom. Takéto algoritmy sú napríklad SIFT a SURF.

Kapitola 4

Návrh riešenia

Rozpoznávanie dopravných značiek je založené na analýze obrazu získaného z videokamery. Najprv sa musia odhaliť oblasti, ktoré obsahujú nejaké dopravné značenie, potom identifikovať značku a nakoniec odovzdať informáciu vodičovi.

Automatické rozpoznávanie značiek vieme rozdeliť na dve fázy. V prvej fáze sa vyberú oblasti, kde sa môžu potenciálne značky nachádzať a v druhej fáze sa určí, ktorá značka to presne je.

4.1 Získavanie obrazu

Aby sme dosiahli čo najlepšie a najpresnejšie výsledky, potrebujeme kvalitné zábery. Kvalita videa má byť dostatočná, aby sa na jednotlivých snímkach dala presne segmentovať oblasť s kandidátmi, zistiť ich tvar a rozpoznať značku. Príliš vysoká kvalita však zbytočne spomaľuje proces. Preto sme otestovali viacero kamier, rozlíšení a záberov, získaných za rôznych poveternostných podmienok.

1. Záznamová kamera do auta Sencor SCR 2100FHD

Automobilová kamera Sencor SCR 2100FHD ponúka rôzne možnosti nastavenia rozlíšenia. Zvolili sme rozlíšenie 1920 x 1080 px (16:9), čo

bola najvyššia ponúkaná hodnota. Pri týchto podmienkach kamera sníma 24 snímok za sekundu vo formáte AVI (*Audio Video Interleave*). Rozlíšenie snímača je 1,3 Mpx (*Megapixel*).

2. Sony Cyber-shot DSC-HX50V

Video pomocou tohto zariadenia je možné natáčať vo Full HD rozlíšení (1920 x 1080 px) pri 50 snímok za sekundu v AVCHD (*Advanced Video Coding High Definition*) formáte. Rozlíšenie snímača je 20,4 Mpx.

3. Apple iPhone 6

Tretím zariadením ktorým sme získavali videá bol mobilný telefón Apple iPhone 6. Videá taktiež natáča v 1920 x 1080 px rozlíšení s 59 snímkami za sekundu, pričom rozlíšenie snímača má 8 Mpx. Videá sú v MOV formáte.

4.2 Nástroje

4.2.1 MATLAB

Detekciu a rozpoznávanie dopravných značiek sme sa rozhodli riešiť v MATLAB-e. MATLAB je programovacie prostredie, ktoré sa špecializuje na vedec-kotechnické výpočty, modelovanie, návrhy algoritmov, analýzu a prezentáciu údajov atď. Používa vlastný programovací jazyk MATLAB [18].

Na spracovanie obrazu sme použili Image Processing Toolbox, ktorý poskytuje komplexnú sadu štandardných algoritmov na spracovanie obrazu, ich analýzu a vizualizáciu [19]. Okrem toho aj VLFeat, čo je multiplatformová open source kolekcia vizuálnych algoritmov so špeciálnym zameraním na vizuálne funkcie [29].

4.2.2 Wondershare Video Editor

Na úpravu nasnímaných videí použijeme softvér na editovanie videí - Wondershare Video Editor [6]. Pomocou tohto programu vytvoríme videá rôznej

kvality, rozlíšenia, formátu a počtu snímok za sekundu. Pripravíme aj špeciálne video na testovanie výsledného programu, kde sa bude nachádzať zvýšený výskyt modrých značiek.

4.3 Predspracovanie obrazu

V tejto fáze sa na snímky aplikujú vyhladzovacie filtre, aby sa pri segmentácií dosiahli presnejšie výsledky. Porovnáme dve techniky: Gaussov filter a mediánový filter. Gaussov filter odstraňuje šum pomocou rozmazania obrázka. Mediánový filter okrem odstránenia šumu aj zvýrazní hrany.

4.4 Segmentácia modrých oblastí

Po vyhladení snímky sa hľadajú oblasti v obraze, kde sa nachádzajú modré pixely, teda kde je šanca, že sa bude dopravná značka nachádzať. Segmentovať modrú farbu však nie je ľahké. Je potrebné čo najpresnejšie určiť prah aby sme vyhodili zbytočné oblasti, ako napríklad jasno modrú oblohu alebo veľké reklamné tabule popri ceste.

Segmentáciu modrých pixelov skúsime pomocou troch farebných modelov: RGB, HSV, CIEL*a*b*.

4.5 Morfologické operácie

Ďalším krokom je upraviť segmentovaný obraz. Pomocou morfológických operácií ako erózia, dilatácia, otvorenie a uzatvorenie vieme odstrániť príliš malé objekty alebo vyplniť diery v objektoch. Často sa stáva, že po segmentovaní sa objekt rozdelí na dve menšie časti. Pomocou týchto operácií ich vieme znovu spojiť a nestratíme informácie o objekte.

4.6 Detegovanie tvaru

Aby sme vedeli určiť značky, potrebujeme vedieť ich tvar. Pomocou hránových detektorov vieme vytvoriť obrisy útvarov, čo nám môže pomôcť pri určovaní tvaru objektu. Pre tento účel nám rovnako vyhovuje aj Cannyho či Sobelov algoritmus.

Pomocou Houghovej transformácie vieme detegovať kruhové tvary v obraze. Keďže sa zaoberáme aj so štvorcovými značkami, potrebujeme metódu, pomocou ktorého nájdeme aj iné útvary. Na tento účel zvolíme výpočet kruhovitosti. Na základe výslednej hodnoty daného objektu sa vieme rozhodnúť, či je útvar v tvare kruhu, štvorca, trojuholníka alebo je to pre nás nezaujímavý objekt. Keďže kamera sníma značky z rôznych uhlov, nemôžeme počítať s dokonalými tvarmi. Preto treba upraviť intervaly hodnôt tak, aby sme vedeli správne zaradiť aj elipsy či kosoštvorce.

4.7 Databáza značiek

Aby sme vedeli rozpoznať značky, potrebujeme šablóny jednotlivých značiek, s čím vieme porovnať kandidátov. Databázu značiek sme získali z Prílohy č. 1 1. diel Vyhlášky Ministerstva Vnútra Slovenskej Republiky č. 9/2009 Z. z. [30] (Obr. 4.1). Rozdelili sme ich podľa tvaru, aby sme nemuseli kontrolovať každú šablónu. Po zistení tvaru kandidáta ho porovnáme len so šablónami rovnakého tvaru.



Obr. 4.1: Ukážka použitých šablón [30]

4.8 Druhá fáza - detekcia

Po získaní segmentovaných oblastí a následne určení ich tvaru prichádza fáza zisťovania, ktorá značka sa vyskytuje na obraze. Na rozpoznávanie objektov existuje viacero algoritmov.

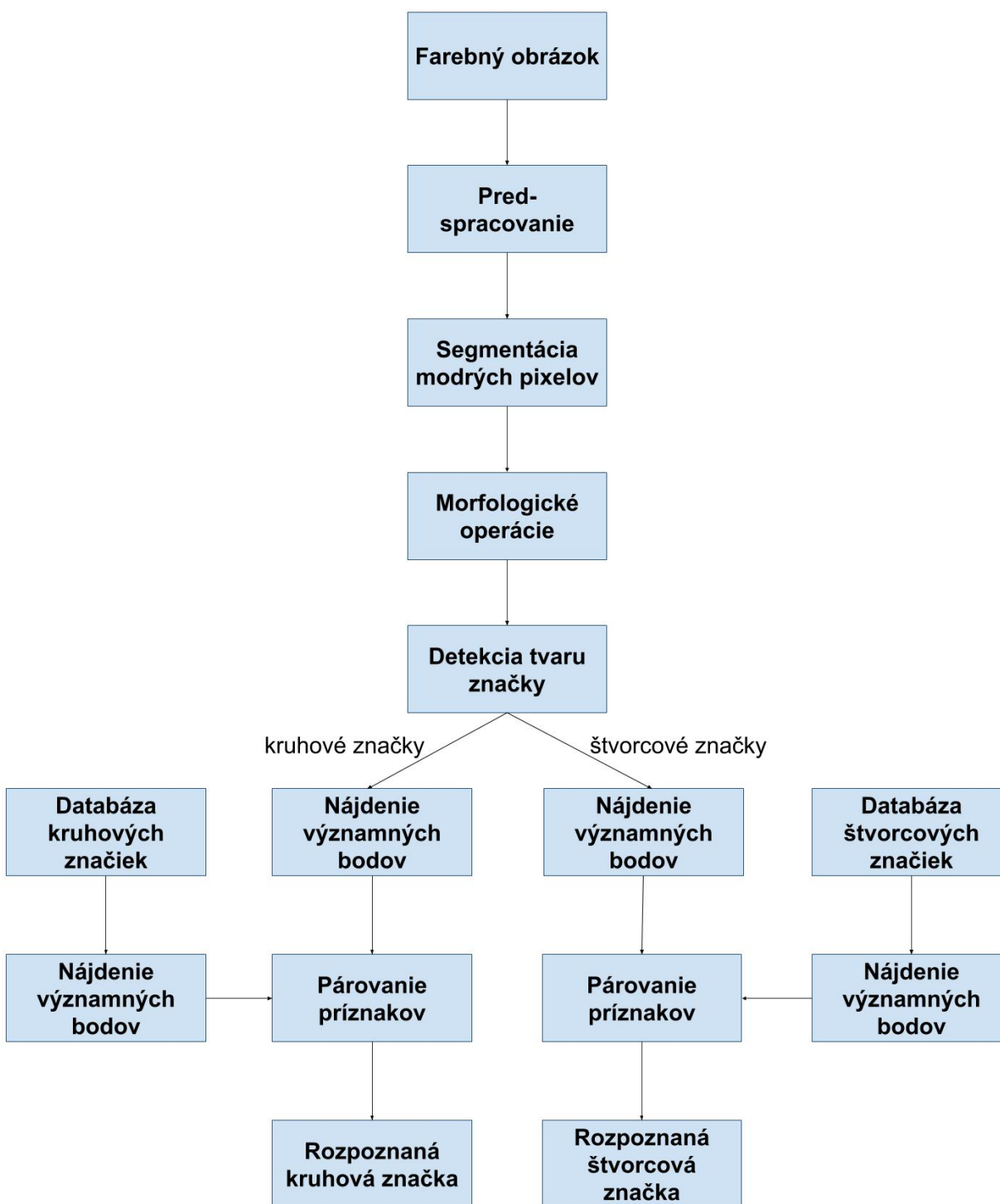
Pomocou porovnávania šablón vieme nájsť významné body v oboch obrazoch a zistiť či sa zhodujú. Tieto zaujímavé body vieme získať aj pomocou rohových detektorov. Táto metóda však často nie je postačujúca. Algoritmy SIFT, SURF alebo MSER okrem určenia významných bodov k nim vytvoria deskriptory a tým presnejšie určia okolie významného bodu.

Ďalšia možnosť ako rozpoznať ktorá značka sa nachádza na snímke, je spočítať množstvo zhodujúcich sa pixelov. V tomto prípade je potrebné mať presne segmentovaného kandidáta z okolia a upraviť jeho veľkosť na rovnakú, ako je šablóna.

4.9 Uživatelské prostredie

Naším cieľom je aj vytvoriť užívateľské prostredie, kde znázorníme detegovanú oblasť so značkou a rozpoznanú príslušnú značku. Pripravíme aj prostredie na testovanie jednotlivých techník a ďalšie na porovnanie týchto techník.

Postupnosť krokov algoritmu na detekciu dopravných značiek sme zobrazili na obrázku 4.2.



Obr. 4.2: Postup pri riešení problematiky

Kapitola 5

Implementácia

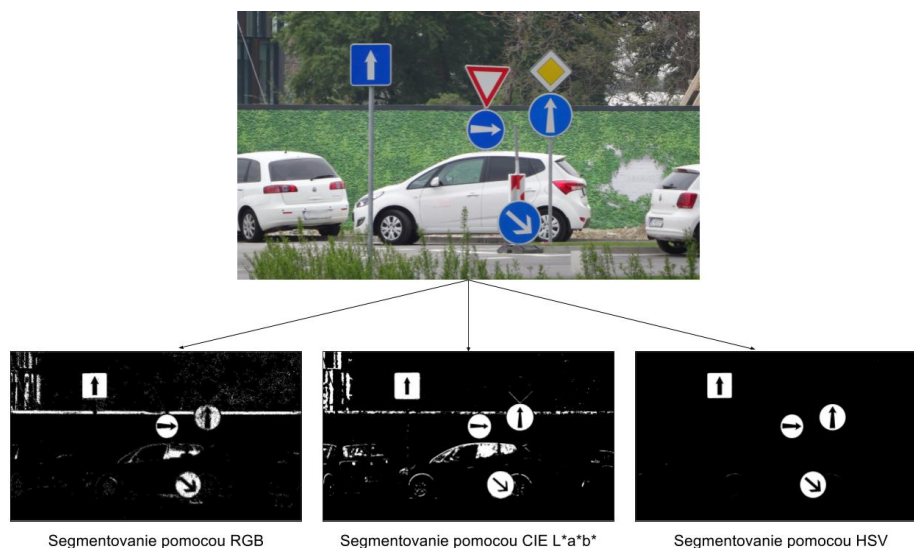
Našou úlohou bolo vytvoriť aplikáciu na detekciu a rozpoznávanie dopravných značiek. Ako sme už spomenuli v návrhu riešenia, rozhodli sme sa to implementovať v MATLAB-e.

Prvým krokom pri spustení programu je načítanie vzorových značiek, aby sme s nimi vedeli neskôr pracovať a nespomaľovali program. Potom si používateľ môže vybrať video, ktoré sa načíta a podľa voľby používateľa sa aj spustí. Postupne sa na každú snímku zavolajú funkcie, aby odhalili potenciálny výskyt značiek.

Vo fáze predspracovania obrazu sa snímka vyhladí a odstráni sa šum z obrazu. Skúšali sme použiť Gaussov a mediánový filter. Po porovnaní výsledkov z oboch filtrov sme zistili, že Gaussov filter je vhodnejší pre použitie v našej práci. Nie len preto, že sme pri segmentácií získali presnejšie určenú oblasť, ale aj kvôli tomu že mediánový filter je náročnejší na spracovanie a spomaľoval beh programu.

Po vyhladení obrázku sme vytvorili binárny obrázok, kde biele pixely reprezentovali modré oblasti z pôvodného obrazu a čierne všetky ostatné. Porovnávali sme RGB, HSV a CIE L^*a^*b farebné modely (Obr. 5.1).

Farebný model RGB vykazoval dobré výsledky za určitých podmienok.



Obr. 5.1: Segmentovanie pomocou RGB, CIE L*a*b* a HSV farebného modelu

Prah sme nastavili tak, aby vyhovovali len pixely, ktoré majú hodnotu modrej farebnej zložky viac ako 72 a zároveň menej ako 80 z červenej aj zelenej zložky. Keďže je model RGB veľmi citlivý na osvetlenie, často sa po segmentovaní vyradili oblasti, kde sa modré značky nachádzali.

Ani farebný model CIE L*a*b* sa neosvedčil ani napriek dlhému skúšaní rôznych prahových hodnôt.

Najlepšie výsledky sa podarilo dosiahnuť pomocou farebného modelu HSV. Pre jednotlivé zložky sme určili hodnoty tak, aby vedeli segmentovať modrú farbu v rôznych odtieňoch. Farebný tón (H) sme previedli na interval od 0 po 1. Na tomto intervale sa odtiene modrej nachádzajú medzi hodnotami 0,55 až 0,7. Aby sme zúžili počet kandidátov, nastavili sme obmedzenie aj pre ďalšie zložky. Jasová hodnota (V) zaujímavého pixelu musí byť medzi 0,2 až 0,8 a sýtosť farby (S) nesmie byť menšia ako 0,4. Tieto hodnoty nám zabezpečili presnejší výsledok.

Po prahovaní sme snímku upravili pomocou morfológických operácií (Obr. 5.2). Aplikovali sme otvorenie a uzatvorenie. Morfológické otvorenie odstráni



Obr. 5.2: Obrázok pred a po úprave morfológickými operáciami

príliš malé oblasti, teda oblasti ktoré sa považujú za šum. Morfológické uzatvorenie zjednotí a uzatvorí oblasti, ktoré sa pri segmentácii mohli oddeliť a zároveň vyplní diery.

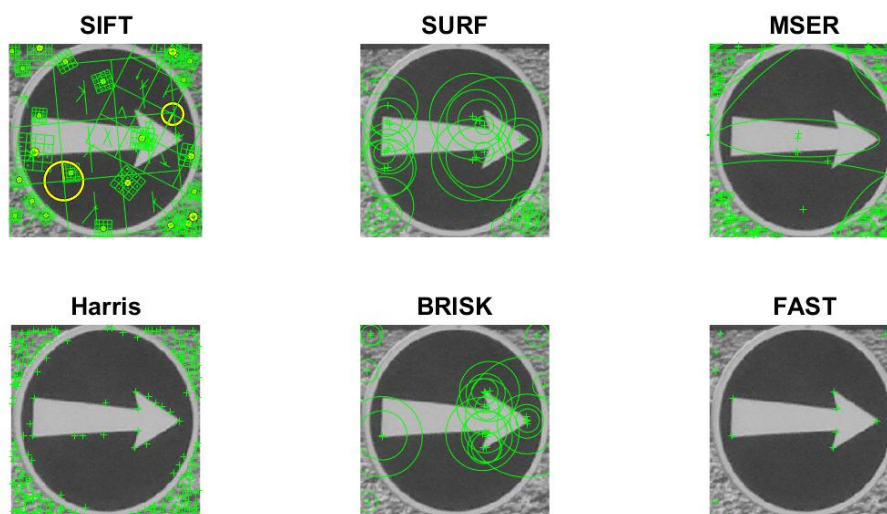
Po získaní regiónov, kde sa môžu značky nachádzať sme použili výpočet na kruhovitosť. Nie len že takto získame informáciu o tvare kandidáta, ale aj zahodíme oblasti, kde značka nie je s veľkou pravdepodobnosťou. Z dôvodu, že sa značky nachádzajú popri ceste, teda nie priamo ich sníma kamera, ich tvar sa často zdá zdeformovaný. Vzhľadom na to sme určili intervaly, podľa ktorých sme zaradili objekty na základe ich hodnoty kruhovitosti. Ak však kamera priamo nasnímala štvorcovú značku, kvôli upraveným intervalom ho zaradila medzi kruhové tvary.

Preto sme zaviedli ďalšiu kontrolu. Objekt sa vyhlási za kruh až po tom, ak sa nachádza v určenom intervale pre kruhový tvar a zároveň sa tam našiel kruh pomocou Houghovej transformácie.

Ak na základe hodnoty kruhovitosti usúdime, že kandidát nie je ani kruh, ani štvorec, tak ho za značku nepovažujeme a ďalej sním už nepracujeme.

V ďalšom kroku rozpoznávame konkrétnu značku. Implementovali sme viacero metód a porovnávali získané výsledky.

Pre testované detektory a deskriptory bol postup skoro rovnaký. Najprv sme nastavili rovnakú veľkosť pre kandidáta a šablóny. Potom sa určili významné body kandidáta a všetkých šablón z databázy značiek (Obr. 5.3). Keďže



Obr. 5.3: Významné body získané pomocou rôznych algoritmov

v tejto fáze už vieme tvar značky, pracujeme len so šablónami toho istého tvaru ako má kandidát. Po získaní významných bodov prebehne ich párovanie a spočítame počet zhôd. Šablóna, ktorá má s kandidátom najviac spoločných významných bodov sa vyhlási za správnu značku. V prípade, že kandidát nemá so žiadnou šablónou viac ako tri zhody, zahadzuje sa.

Deskriptory SIFT a SURF fungovali pomerne správne. Keďže sú rotačne invariantné, mali problém so značkami, na ktorých sa nachádzala nejaká šípka.

Algoritmus MSER a Harrisov rohový detektor fungoval už o čosi horšie. Málokedy našli zhodu s nejakou značkou a ak našli, často to nebola tá správna.

FAST [24] a BRISK [25] fungovali najhoršie. Algoritmus FAST nenašiel v skúšaných prípadoch žiadne zhodujúce sa body medzi obrázkami a BRISK ich našiel veľmi málo. Preto sme ich už pri ďalších testovaniach nebrali do úvahy.

Okrem týchto metód sme implementovali aj ďalšiu, pomocou ktorej sme taktiež zisťovali, ktorá značka sa nachádza na snímke. Najprv sme nastavili rovnakú veľkosť pre kandidáta a pre šablónu. Najlepšie výsledky sme dosahovali pri veľkosti obrazu 128 x 128 pixelov. Potom sme obrázky premenili na binárne obrazy a spočítali sme, koľko pixelov má na rovnakej pozícii rovnakú hodnotu. Za správnu značku sme vyhodnotili šablónu s najväčším počtom rovnakých pixelov s kandidátom, pričom počet zhôd musela byť väčšia než 200 000. Táto metóda fungovala bezchybne, ak sa presne určila značka pri segmentovaní. V prípade, že sa za značkou nachádzala nejaká modrá reklamná tabuľa, alebo iné, čo skreslil región značky, dostávali sme zlé výsledky.

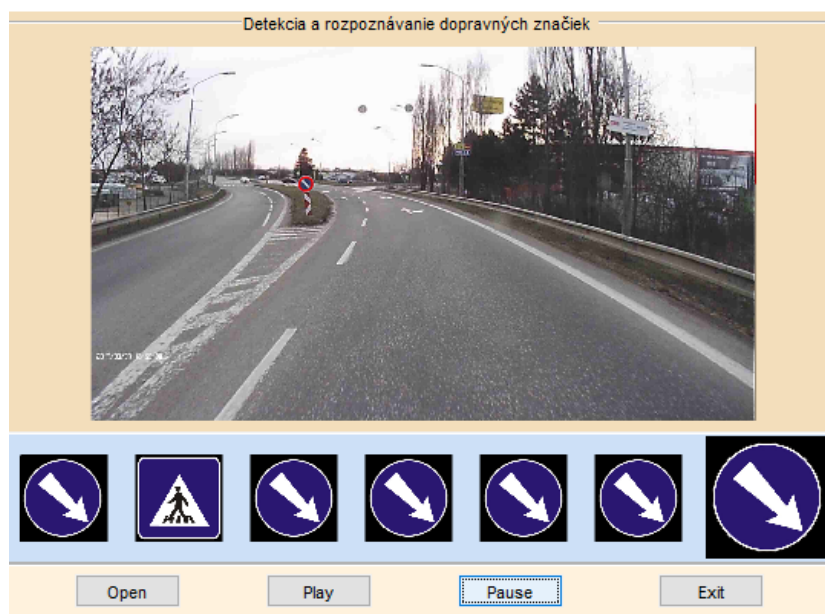
Uvedený postup aplikujeme na každú snímku, avšak nie stále sa zdeteguje správna značka, alebo vôbec nejaká. Preto sme sa rozhodli počítať výskyt značiek a zobrazíť tú, ktorá má najviac bodov. Ak sa na danej snímke zdeteguje značka označujúca prechod pre chodcov, pripočítame jeden bod k hodnote, ktorá reprezentuje túto značku. Pred určením správnej značky sa najprv pozrieme na hodnoty a zobrazíme značku s najväčším počtom bodov. Takto zvýšime pravdepodobnosť vyhodnotenia správnej značky. Na to, aby sa nejaká značka zobrazila, musí mať minimálne dva body. Ak sa na troch po sebe idúcich snímkach žiadna značka nenájde, tak to znamená, že daná značka už zmizla z obrazu a vynulujeme spočítané hodnoty.

Pripravili sme tri rôzne používateľské rozhrania. Dve rozhrania na testovacie účely a jedno finálne rozhranie. Do finálneho rozhrania sme zapracovali metódy a funkcie, ktoré sa počas testov najviac osvedčili.

Do všetkých sme implementovali tlačítko na načítanie ľubovoľného videa z priečinka. Používateľ pustí video tlačítkom *Play* a tým sa začne aj detekcia. Pomocou tlačítka *Pause* môže video zastaviť a tlačítkom *Play* znovu spustiť. V každom rozhraní je tlačítko *Exit*, ktorým sa detekcia skončí a program sa zatvorí.

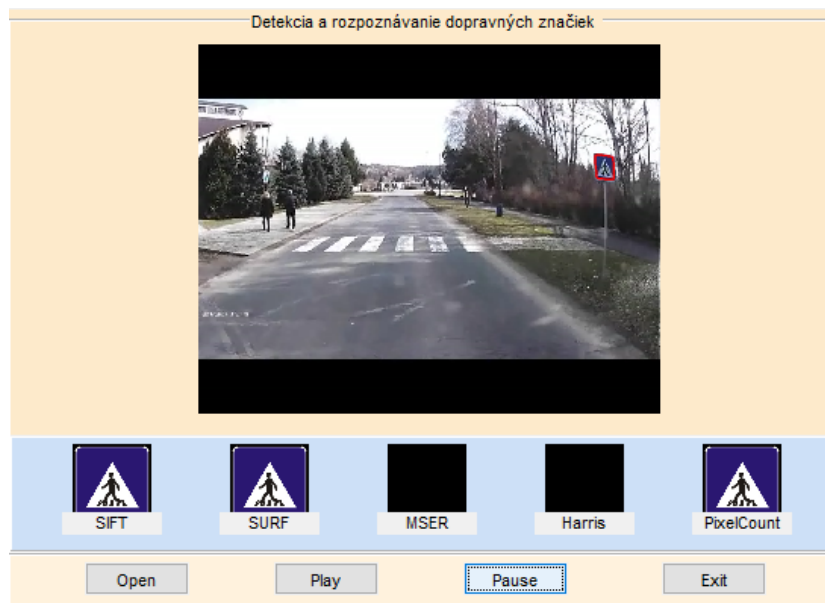
Prvé rozhranie slúži na testovanie vybranej techniky. Popri prehrávaní

video sa pod ním zobrazujú zdetegované značky za posledných šesť snímok. V najväčšom okienku sa zobrazuje značka, ktorá má najviac bodov a teda je s najväčšou pravdepodobnosťou tá správna značka (Obr. 5.4).



Obr. 5.4: Používateľské rozhranie č.1

Ďalšie rozhranie umožňuje testovanie všetkých implementovaných techník naraz. Pod videom sa zobrazí výsledok pre každú metódu (Obr. 5.5). Finálne rozhranie zobrazí zdetegovanú oblasť a vedľa nej aj príslušnú rozpoznanú značku (Obr. 5.6).



Obr. 5.5: Používateľské rozhranie č.2



Obr. 5.6: Používateľské rozhranie č.3

Kapitola 6

Výsledky a testovanie

Program sme otestovali pomocou nasnímaných videí z kamier, ktoré sme opísali v časti “Návrh riešenia”. Výsledok sme získali na základe zostaveného videa, v ktorom sa vyskytovali zábery s modrými značkami zo všetkých troch kamier. Testovacie video obsahovalo zábery natočené za rôznych poveternostných podmienok. Vyskytovali sa tam zábery získané počas slnečného dňa, zamračeného počasia ale aj zábery natočené v noci.

Najvyššiu úspešnosť zdetegovaných značiek mali videá natočené keď kolo zamračené počasia alebo za tmy. Zábery nasnímané za slnečného počasia mali najhoršiu úspešnosť. Dôvodom bola jasno modrá obloha, ktorá sa často segmentovala spolu so značkou a preto sa značka stratila v regióne. Algoritmy SIFT a SURF by síce vedeli nájsť významné body a spárovať ich s bodmi zo šablóny, ale pri detegovaní tvaru sa neurčitý tvar oblasti značky a oblohy zahodil. Ďalším problémom pri týchto záberoch bolo to, že lúče slnka spôsobili skreslenie farieb. Keďže sa snímané modré značky javili ako šedé, zle sa odsegmentovali.

Testovacie video obsahovalo 50 modrých dopravných značiek. Z toho bolo zdetegovaných 45 značiek (tab. 6.1), ktoré sme následne pomocou algoritmov rozpoznávali.

Testovali sme hlavne metódy SIFT, SURF a porovnávanie pixelov.

Počet dobravných značiek v testovacom videu	Počet zdetegovaných značiek	Počet nezdetegovaných značiek
50	45	5
	90%	10%

Tabuľka 6.1: Zdetegované dopravné značky z testovacieho videa

6.1 SIFT

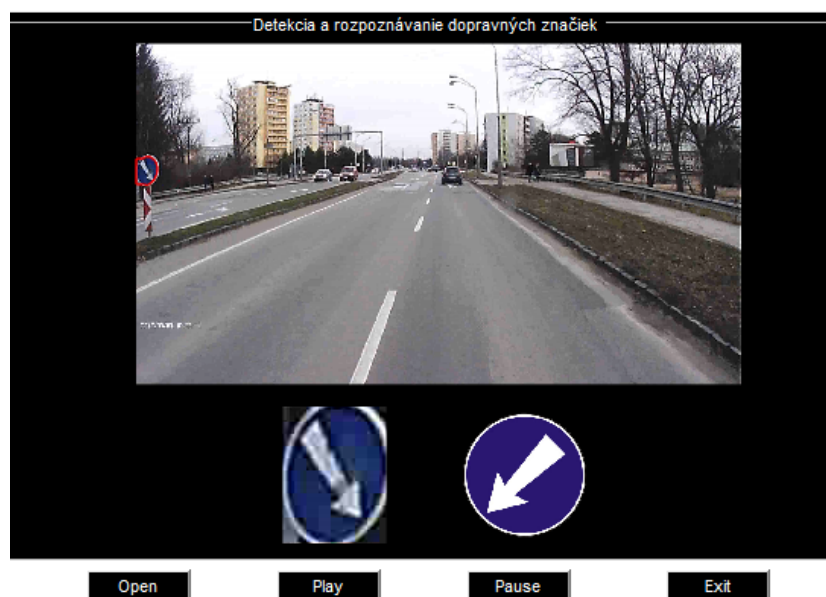
Algoritmus SIFT fungoval na 78% správne (tab. 6.2). Najväčší problém spôsobovali kruhové značky obsahujúce nejakú šípku. SIFT je rotačne invariantný, preto často zdetegoval inak otočenú šípku, ako bola na snímke (Obr. 6.1). Keby informácia o orientácii tabule nebola nevyhnutná, získali by sme 91%-nú úspešnosť.

Počet kandidátov	Správne klasifikovaní	Nesprávne klasifikovaní	Neklasifikovaní kandidáti
45	35	10	0
	78%	22%	0%

Tabuľka 6.2: Úspešnosť klasifikácie pomocou algoritmu SIFT

6.2 SURF

Pomocou algoritmu SURF sme mali len 56%-nú úspešnosť (tab. 6.3). Taktiež ako SIFT, aj SURF je rotačne invariantný. Ak by sme správnu tabuľu so zlou orientáciou vyhodnotili ako správnu detekciu, získali by sme 69% správnych výsledkov.



Obr. 6.1: Príklad zlej klasifikácie

Počet kandidátov	Správne klasifikovaní	Nesprávne klasifikovaní	Neklasifikovaní kandidáti
45	25	13	7
	56%	29%	15%

Tabuľka 6.3: Úspešnosť klasifikácie pomocou algoritmu SURF

6.3 Porovnávanie pixelov

Tretia metóda porovnávala jednotlivé pixely kandidátov a šablón, na základe rovnakých hodnôt na rovnakej pozícii. Pomocou tejto metódy sme získali 80% správne detegovaných tabúľ (tab. 6.4). Pri tejto metóde je nevyhnutné, aby sa presne segmentoval tvar značky. Ak sa dostal do regiónu, kde sa nachádzala značka nejaký iný objekt, metóda nefungovala správne. To sa stávalo najmä v prípadoch, ak sa za značkou nachádzali modré objekty, reklamné tabule, modré autá alebo jasno modrá obloha (Obr. 6.2, 6.3).



Obr. 6.2: Zle segmentovaná oblasť kvôli oblohe



Obr. 6.3: Zle segmentovaná oblasť kvôli reklamnej tabuli

Metódy SIFT a porovnávanie pixelov mali pomerne dobrú úspešnosť, avšak pokúšali sme sa získať ešte presnejšie výsledky rozpoznávania. Keďže tieto techniky mali rôzne výhody a nevýhody, rozhodli sme sa ich skombinovať. Takto sme získali správne určenú orientáciu šípky a správne určenú

Počet kandidátov	Správne klasifikovaní	Nesprávne klasifikovaní	Neklasifikovaní kandidáti
45	36	9	0
	80%	20%	0%

Tabuľka 6.4: Úspešnosť klasifikácie pomocou porovnávania pixelov

tabuľu aj pri nie úplne presnej segmentácii tabule. Kombinácia techník priniesla 93%-nú úspešnosť (tab. 6.5).

Počet kandidátov	Správne klasifikovaní	Nesprávne klasifikovaní	Neklasifikovaní kandidáti
45	42	3	0
	93%	7%	0%

Tabuľka 6.5: Úspešnosť klasifikácie pri kombinácii dvoch techník

Tabuľka 6.6 zobrazuje celkovú úspešnosť rozpoznávania modrých dopravných značiek. Najlepšie výsledky sme dosiahli pri kombinovaní dvoch techník, SIFT-u a porovnávania pixelov.

Metóda	Počet značiek	Správne klasifikované	Správne klasifikované v %
SIFT	50	35	70%
SURF	50	25	50%
Porovnávanie pixelov	50	36	72%
Kombinácia dvoch metód	50	42	84%

Tabuľka 6.6: Celková úspešnosť rozpoznávania

Záver

V tejto práci sme sa venovali detekcii a rozpoznávaniu dopravných značiek. Opísali sme základné pojmy, postup a návrh riešenia pri spracovaní tohto problému. Zamerali sme sa na modré dopravné značky, nakoľko sa predchádzajúce práce sústreďovali iba na červené. Veľkou výzvou bolo správne segmentovať modré značky bez iných rušivých objektov.

Pri detekcii dopravných značiek sme najprv porovnávali filtrovacie metódy a segmentáciu pomocou rôznych farebných modelov. Na základe výsledkov sme určili vhodnejšiu metódu, ktorú sme implementovali do finálneho programu. Na segmentované oblasti sme použili morfológické operácie otvorenie a uzatvorenie.

Na identifikáciu tvaru sme použili výpočet na kruhovitosť a Houghovu transformáciu. Vďaka tejto informácii sme vedeli upresniť typ značky.

V ďalšom kroku sme rozpoznávali konkrétnu značku. Pre tento účel sme porovnávali viacero algoritmov. Deskriptory SIFT a SURF vykazovali najlepší výsledok spomedzi detektorov a deskriptorov. Implementovali sme aj metódu, ktorá porovnávala jednotlivé pixely na oboch snímkach na rovnakej pozícii a na základe najvyššej zhody sme určili výsledok.

Z implementovaných techník sme dosiahli najväčšiu úspešnosť správne rozpoznaných značiek pri kombinovaní dvoch techník - SIFT-u a počítania pixelov.

V budúcej práci by sme sa mohli venovať rozpoznávaniu textu z informačných tabúl.

Literatúra

- [1] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. *Computer Vision and Image Understanding*, 110(3):346–359, 2008. [online]. <http://www.vision.ee.ethz.ch/~surf/eccv06.pdf>.
- [2] Karla Brkić. An overview of traffic sign detection methods, [Dátum: január 2017]. http://www.esa.fer.unizg.hr/_download/repository/Brki-cQualifyingExam.pdf.
- [3] Tim Horberry Candida Castro. *The Human Factors of Transport Signs*. CRC Press, 2004.
- [4] John Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8, Issue: 6:679 – 698, Nov. 1986.
- [5] MathWorks Dokumentation. Convert from hsv to rgb color space, [Dátum: január 2017]. [online]. <https://www.mathworks.com/help/images/convert-from-hsv-to-rgb-color-space.html>.
- [6] Filmora, [Dátum: apríl 2017]. [online]. <https://filmora.wondershare.com/video-editor/>.
- [7] R. Fisher, S. Perkins, A. Walker, and E. Wolfart. *Morphology*, 2003. The University of Edinburg, School of Informatics.

- [8] David Geronimo, Antonio M. Lopez, Angel D. Sappa, and Thorsten Graf. Survey of pedestrian detection for advanced driver assistance systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(7):1239 – 1258, July 2010.
- [9] Technical Guides. Color models, cielab, [Dátum: január 2017]. [online]. http://dba.med.sc.edu/price/irf/Adobe_tg/models/cielab.html.
- [10] Chris Harris and Mike Stephens. A combined corner and edge detector. *In Proc. of Fourth Alvey Vision Conference*, pages 147–151, 1988. [online]. <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.434.4816&rep=rep1&type=pdf>.
- [11] Peter E. Hart. How the hough transform was invented [dsp history]. *IEEE Signal Processing Magazine*, Volume: 26, Issue: 6, November 2009.
- [12] E. Šikudová, Z. Černeková, W. Benešová, Z. Haladová, and J. Kučerová. Počítačové videnie. detekcia a rozpoznávanie objektov, 2015.
- [13] P. Felkel J. Žára, B. Beneš. *Moderní počítačová grafika*. Computer Press, 1998.
- [14] Jun Ji, [Dátum: január 2017]. [online]. <http://jun.hansung.ac.kr/DI/Chapter-2.htm>.
- [15] David G. Lowe. Object recognition from local scale-invariant features. *Computer Science Department University of British Columbia*, 1999. [online]. <http://www.cs.ubc.ca/~lowe/papers/iccv99.pdf>.
- [16] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. [online]. https://www.robots.ox.ac.uk/~vgg/research/affine/det_eval_files/lowe_ijcv2004.pdf.

- [17] Matas, J., Chum, O., Urban, M., and Pajdla T. Robust wide baseline stereo from maximally stable extremal regions. In *Proc. BMVC*, pages 36.1–36.10, 2002. doi:10.5244/C.16.36.
- [18] MathWorks, [Dátum: marec 2017]. [online]. <https://www.mathworks.com/products/matlab.html>.
- [19] MathWorks, [Dátum: marec 2017]. [online]. <https://www.mathworks.com/products/image.html>.
- [20] O. Folorunso O. R. Vincent. A descriptive algorithm for sobel image edge detection. *Proceedings of Informing Science and IT Education Conference (InSITE)*, 2009.
- [21] M. Jones P. Viola. Robust real-time object detection. *International Journal of Computer Vision*, 2001.
- [22] A. Walker R. Fisher, S. Perkins and E. Wolfart. Digital filters, 2003. The University of Edinburg, School of Informatics.
- [23] A. Walker R. Fisher, S. Perkins and E. Wolfart. Hough transform, 2003. The University of Edinburg, School of Informatics.
- [24] E. Rosten and T. Drummond. Machine learning for high-speed corner detection. *European Conference on Computer Vision*, page 430–443, 2006.
- [25] M. Chli a R. Y. Siegwart S. Leutenegger. Brisk: Binary robust invariant scalable keypoints. *Computer Vision (ICCV), 2011 IEEE International Conference on. IEEE*, page 2548– 2555, 2011.
- [26] A. Scukanac S. Pasagic. Historical development of traffic signs. *Promet-Traffic- Traffico*, 10(5-6):309–313, 1998.
- [27] Inland transport comitee. Convention on road signs and signals, 1968. Economic comission for Europe.

- [28] E.A.Edirisinghe U.Zakir, A.N.J.Leonce. Road sign segmentation based on colour spaces: A comparative study. *Proceedings of the 11th IASTED International Conference*, February 17-19, 2010. Computer Graphics and Imaging (CGIM 2010).
- [29] VLFeat.org, [Dátum: marec 2017]. [online].
<http://www.vlfeat.org/about.html>.
- [30] Ministerstvo vnútra SR. Vyhláška mv sr č. 9/2009 z. z. o cestnej premávke a o zmene a doplnení niektorých zákonov v znení neskorších predpisov, 2009.

Prílohy

K tejto práci je priložené CD-médium, na ktorom sa nachádza písomná aj praktická časť bakalárskej práce.