

UNIVERZITA KOMENSKÉHO V BRATISLAVE  
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY

DETEKCIA DOPRAVNÝCH ZNAČIEK  
BAKALÁRSKA PRÁCA

2020  
ROBERT STERNMÜLLER



UNIVERZITA KOMENSKÉHO V BRATISLAVE  
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY

DETEKCIA DOPRAVNÝCH ZNAČIEK  
BAKALÁRSKA PRÁCA

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

Bratislava, 2020  
Robert Sternmüller





Univerzita Komenského v Bratislave  
Fakulta matematiky, fyziky a informatiky

---

### ZADANIE ZÁVEREČNEJ PRÁCE

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

**Názov:** Detekcia dopravných značiek  
*Detection traffic signs*

**Anotácia:** Z videa zosnímaného mobilom umiestneným na palubnej doske automobilu, vedieť detegovať a rozpoznať dopravné značky.

**Cieľ:** Z videa zosnímaného mobilom umiestneným na palubnej doske automobilu, vedieť detegovať a rozpoznať dopravné značky.

**Vedúci:** RNDr. Zuzana Černeková, PhD.  
**Katedra:** FMFI.KAI - Katedra aplikovanej informatiky  
**Vedúci katedry:** prof. Ing. Igor Farkaš, Dr.  
**Dátum zadania:** 27.09.2018

**Dátum schválenia:** 07.10.2019

doc. RNDr. Damas Gruska, PhD.  
garant študijného programu

.....  
študent

.....  
vedúci práce

**Čestné prehlásenie:** Čestne prehlasujem, že som túto bakalársku prácu vypracoval samostatne, pod vedením vedúceho bakalárskej práce a za použitia uvedených zdrojov.

V Bratislave dňa: .....

.....  
Robert Sternmüller

**Pod'akovanie:** Ďakujem vedúcej bakalárskej práce RNDr. Zuzane Černekovej, PhD. za pomoc, čas a užitočné rady poskytnuté pri vypracovávaní tejto práce.

## Abstrakt

Cieľom tejto bakalárskej práce je vytvorenie mobilnej aplikácie na detekciu rôznych typov dopravných značiek a oboznámenie používateľa s problematikou počítačového videnia v mobilných zariadeniach. Práca je rozdelená na úvodnú časť opisujúcu obdobné bakalárske alebo diplomové práce, ďalej sa v práci nachádzajú súčasne používané technológie pre spracovanie videa. V závere práce sa nachádza samotný popis implementácie.

**Kľúčové slová:** Dopravné značky, mobilné systémy, detekcia, obraz, Android



## Abstract

The aim of this bachelor thesis is to create a mobile application for the detection of various types of traffic signs and acquaint the user with the issue of computer vision in mobile devices. The work is divided into an introductory part describing similar bachelor's or master's theses, then there are currently used technologies for video processing. At the end of the work is the description of the implementation.

**Keywords:** Traffic signs, mobile systems, detection, image, Android



# Obsah

<b>Úvod</b>	<b>1</b>
<b>1 Prehľad problematiky</b>	<b>3</b>
1.1 Dopravné značky . . . . .	3
1.1.1 Farby a tvary dopravných značiek . . . . .	3
1.2 Podobné bakalárske alebo diplomové práce . . . . .	4
1.2.1 Rozpoznávanie dopravných značiek [3] . . . . .	4
1.2.2 Detekcia a rozpoznávanie dopravných značiek [4] . . . . .	5
1.2.3 Detekcia textových dopravných značiek [5] . . . . .	7
1.2.4 Rozpoznávanie dopravných značiek a ich použitie v mapových aplikáciách [6] . . . . .	7
1.3 Teória k práci . . . . .	8
1.3.1 Snímanie videa . . . . .	8
1.3.2 Predspracovanie videa . . . . .	9
1.3.3 Modely a priestory farieb . . . . .	11
1.3.4 Klasifikátory . . . . .	12
<b>2 Návrh postupu pri riešení problematiky</b>	<b>13</b>
2.1 Nasnímanie videa . . . . .	13
2.2 Úprava vstupného obrazu . . . . .	14
2.3 Predspracovanie obrazu . . . . .	14
2.4 Segmentácia podľa farby . . . . .	14
2.5 Získanie kandidátov a ich SIFT/SURF descriptorov kandidátov . . . . .	14
2.6 Porovnanie príznakov kandidátov so vzorovými obrazmi . . . . .	15
2.7 Výsledok porovnávania . . . . .	15
<b>3 Implementácia</b>	<b>17</b>
3.1 Použité technológie . . . . .	17
3.2 Snímanie obrazu . . . . .	17
3.3 Vzory značiek . . . . .	18
3.4 Predspracovanie videa . . . . .	18

3.5	Segmentácia podľa farby . . . . .	18
3.6	Segmentácia podľa tvaru . . . . .	19
3.7	Porovnanie príznakov kandidátov so vzorovými značkami . . . . .	20
3.8	Informovanie používateľa o detekovaní značky . . . . .	20
	<b>Záver</b>	<b>21</b>

# Zoznam obrázkov

1.1	Ukážka rozdielov známych rozlíšení . . . . .	9
1.2	Ukážka Gaussovho filtra . . . . .	10
1.3	Ukážka Mediánovho filtra . . . . .	10
1.4	Ukážka Mediánovho filtra . . . . .	11
1.5	Porovnanie príznakov . . . . .	12
3.1	Porovnanie príznakov . . . . .	19
3.2	Porovnanie príznakov . . . . .	20



# Úvod

Cestná doprava sa stala v posledných desaťročiach neodmysliteľnou súčasťou moderného spôsobu života vo väčšine vyspelých krajín. Denne sa stovky miliónov ľudí prepravuje pomocou osobných automobilov alebo verejnej hromadnej dopravy. Na Slovensku v roku 2017 bežný vodič strávil na cestách 23,33 hodín ( v Spojenom kráľovstve až 45,73). [1] Neoddeliteľnou súčasťou automobilovej dopravy sú dopravné značenia. Dopravná značka je spravidla jednoduchý obrázkový znak (piktogram) poskytujúci účastníkovi cestnej premávky informácie, upozorňujúci na nebezpečenstvo, udávajúci pokyny alebo doplňujúci inú dopravnú značku. História dopravných značiek však siaha ďaleko pred prvý automobil. Už v starovekej Rímskej ríši sa nachádzali dopravné značenia väčšinou oznamujúce vzdialenosť tabule od Ríma. Po vynájdení bicykla a neskôr automobilu a jeho rozšírením medzi obyčajných ľudí sa rozšíril aj počet nových druhov dopravných značiek na cestách kvôli čomu bolo potrebné dopravné značenia zjednotiť. V roku 1908 v Paríži sa prvý krát uskutočnilo stretnutie za účelom ich zjednotenia vďaka čomu boli predstavené prvé medzinárodne dopravné značenia. [2] V súčasnosti je hlavným cieľom vlád, aj automobilového priemyslu čo najviac zvýšiť bezpečnosť na cestách. Hľadajú sa nové spôsoby ako znížiť riziko na cestách. Jedným zo spôsobov je zobrazenie dopravných značení v automobile napr. prostredníctvom GPS, alebo kamerového systému vo vozidle. Problémom týchto metód je buď nie úplná spoľahlivosť (GPS) alebo systém je zahrnutý iba v drahšej variante vozidla (kamery).





# Kapitola 1

## Prehľad problematiky

Úvodom si povieme niečo teórií potrebnej na realizáciu bakalárskej práce s takouto témou, a o momentálne používaných metódach a technológiách na spracovanie obrazu. Ukážeme si práce či už bakalárske alebo diplomové spracované v minulosti a aké postupy a technológie boli v nich použité.

### 1.1 Dopravné značky

Dopravná značka je jednoduchý piktogram slúžiaci na riadenie a reguláciu cestnej premávky. Na základe svojho tvaru, farby, textu či obrázku poskytujú informácie alebo ustanovujú príkazy či zákazy účastníkovi cestnej premávky. Význam každej značky stanovujú pravidlá cestnej premávky, a ich znenie môže upravovať vo vyhláškach ministerstvo vnútra. Dopravné značky majú väčšinou presne vyhradený tvar (kruhový, štvorcový, trojuholíkový, osmihran) a farbu (červená, modrá, žltá, zelená, hnedá etc.) aby boli ľahko odlišiteľné od okolia. Jedným z druhov dopravných značiek su zvislé dopravné značky deliace sa na:

- Značky s výstražnou funkciou
- Značky s informačnou funkciou
- Značky s informačnou funkciou (pruhové)
- Značky s regulačnou funkciou
- Dodatočné tabuľky

#### 1.1.1 Farby a tvary dopravných značiek

Pre nás však bude zaujímavejšie delenie z pohľadu tvaru a farby. Práce spomenuté nižšie majú väčšinou rovnaký postup. Začína sa záznamom nasnímaním kamerou. Ďalej

sa záznam predspracuje použitím filtrov na vyhladenie a odstránenie šumu (Gaussov, mediánov etc.), na čo sa v takto upravenom zázname určia oblasti kde by sa dopravná značka mohla nachádzať. Takéto oblasti sa nazývajú kandidáti. Na záver sa pomocou klasifikátorov, alebo inou metódou určí či je daný kandidát dopravná značka alebo nie. Popríklad aj o akú dopravnú značku ide. Z tohto dôvodu nás budú zaujímať konkrétne kombinácie farieb a tvarov aké dopravná značka môže nadobúdať.

### **Farby a význam značiek s touto farbou:**

1. Červené - väčšinou majú zákazovú, alebo výstražnú funkciu
2. Modré - väčšinou príkazové, určujúce prikázaný smer, minimálnu rýchlosť alebo značky s informačnou funkciou
3. Zelené - všeobecné infomatívne tabuľe
4. Hnedé - upozorňujúce na historickú či kultúrnu pamiatku
5. Oranžové - premenlivé dopravné značenie, dočastne upravujúce trvalé značenia
6. Žlté - upravujúce prednosť v jazde
7. Biele - dodatkové tabuľe

### **Tvar a význam značiek s týmto tvarom:**

1. Kruhový - zákazové alebo príkazové značenia
2. Trojuholníkový - upozornenie na nebezpečenstvo
3. Štvorcový/obdĺžnikový - všeobecné infomatívne tabuľe
4. Kosoštvorcový - upravujúca prednosť v jazde
5. Osmiuholníkový - značenie "STOJ"

Na značení značiek sa používa normové písmo Tern.

## **1.2 Podobné bakalárske alebo diplomové práce**

### **1.2.1 Rozpoznávanie dopravných značiek [3]**

Túto tému spracoval Miloš Fabián vo svojej bakalárskej práci v roku 2011. Cieľom jeho práce bolo opísanie problematiky systémov vizuálneho rozpoznávania zvislých značiek a metódami spracovania obrazu s tým spojených. Pričom sa vo svojej práci zamerl na

červené zákazové značenia. Detekcia prebiehala offline zo zosnímaného videa kamerou na alebo vo vozidle. Ako vzor pre svoju prácu použil „General Traffic Sign Recognition by Feature Matching“ (Všeobecné rozpoznávanie dopravných značiek pomocou porovnávania príznakov) autorov Ren, Huang, Jiang a Klette ktorí použili vtedy bežný postup t.j. snímanie videa, predspracovanie, detekcia oblastí a nakoniec klasifikácia kandidátov. Preto aj v bakalárskej práci používa rovnaký postup.

**Riešenie:** Vstup práce je video z vozidla pohybujúcom sa po cestách. Pričom sa kladie dôraz na kvalitu videa, ktoré sa v aplikácii spracováva jednotlivo po snímkach. Prvou fázou je predspracovanie videa. Autor začal konverziou obrazku vo farebnom priestore RGB na farebný priestor HSV. Dôvod tejto konverzie je zníženie vplyvu pri zmene svetelných podmienok. V druhej fáze autor určuje oblasti kde sa potenciálne môže značka nachádzať. Na určenie tejto oblasti sa použije segmentácia na základe farby a tvaru pričom autor využíva charakteristický kruhový tvar a jasnú červenú farbu značky. Pre segmentáciu na základe farby využil autor prahovanie. Prahovanie je metóda, ktorá umožňuje určenie popredia a pozadia na základe zvolenej prahovej hodnoty, vďaka čomu je výsledkom binárny obraz. Najdôležitejším faktorom je nastavenie správnej prahovej hodnoty aby sa do popredia dostali hlavne červené dopravné značky a čo najmenej iných objektov. Po vysegmentovaní obrazu sa na vybrané oblasti aplikuje Houghova transformácia na vyhľadávanie kruhových tvarov. Po segmentácii na základe farby a tvaru sa na snímke nachádzajú oblasti s kandidátom. Oblasti sa ešte normalizujú škálovaním na menšiu veľkosť vďaka čomu bude výpočet pri klasifikovaní nižší a zvýši sa úspešnosť rozponania. Poslednou fázou je klasifikovanie kandidáta. Autor využil metódu porovnávania príznakov. Pri tejto metóde je nutné získať príznaky nájdeného kandidáta v snímku a porovnať ich s príznakmi vzorových obrázkov značiek v nejakej vytvorenej databáze. Na získanie týchto príznakov autor použil dvojicu známych algoritmov SIFT a SURF ktoré určujú kľúčové body pomocou extrémov. Výsledkom práce bola aplikácia ktorá vedela na 87,5 percenta pri použití algoritmu SIFT detekovať značku a pri SURF na 51,79 percenta. Aplikácia bola vytvorená v jazyku C++, a na prácu s videom využitá knižnica OpenCV vo verzii 2.4. Nevýhodou aplikácie bola funkčnosť iba offline a za ideálnych svetelných podmienok.

### 1.2.2 Detekcia a rozpoznávanie dopravných značiek [4]

Ďalšiu prácu venujúca sa rozpoznávaniu a detekciám dopravných značiek napísala Anikó Szabóová roku 2017. Jej práca sa zamerala na detekciu modrých dopravných značiek. Postup zvolila podobný ako Miloš Fabián v roku 2011. Po nasnímaní videa prišla fáza predspracovania, potom segmentácia oblastí na základe farby a tvaru, a na záver rozpoznanie kandidáta pomocou klasifikátorov.

**Riešenie:** Autorka začala prácu nasnímaním videí v rôznom rozlíšení, kvalite a pri rôznych poveternostých podmienkach. Snažiac sa nájsť čo najlepšie rozlíšenie aby sa dobre hľadali úseky s dopravnými značkami, ale na druhej strane aby rozlíšenie nebolo natoľko veľké že by zbytočne spomaľovalo proces. Vo fáze predspracovania snímky autorka porovnávala jedny s najpoužívanejších filtrov na vyhladenie a odstránenie šumu, a to Gaussov a mediánov filter. Pričom vo svojej práci zistila že Gaussov filter nielen dával presnejšie výsledky pri segmentácií, ale taktiež bol podstatne rýchlejší ako medianov filter, ktorý spomaľoval beh programu. Po vyhladení sa snímka prahovaním pretransformovala na binárny obraz, kde biele pixely predstavovali modrú farbu, ktorú mala dopravná značka nadobúdať v snímke a čierne pixely boli ostatné nevyhovujúce farby. Autorka porovнала viacero farebných modelov (RGB, HSV, CIE L\*a\*b). RGB model mal dobré výsledky pri dobrých svetelných podmienkach, keďže ale RGB model je citlivý na osvetlenie, pri zmene svetelných podmienok vyrad'oval aj oblasti kde sa značky nachádzali. Najlepšie výsledky dosiahol model HSV keďže zmena osvetlenia neovplyvňovala všetky tri zložky modelu. Po prahovaní sa snímka ešte upravila morfológickými operáciami, konkrétne použila morfológické otvorenie vďaka čomu sa odstránil šum a príliš malé objekty, ktoré značkami nemohli byť. Následne aplikáciou morfológického uzatvorenia sa zjednotili oblasti ktoré sa mohli oddeliť pri segmentácií a vyplnili sa diery v objektoch. Po určení oblastí kde sa nachádza vhodná farba bolo ešte potrebné určiť tvar objektu. Výpočtom na kruhovitost' sa zistil tvar objektu, pričom kvôli uhlu natočenia bolo potrebné intervaly poupraviť. Z toho dôvodu autorka pridala ešte jednu kontrolu pomocou Houghovej transformácie na potvrdenie kruhu. V poslednej fáze prišiel na rad klasifikovanie kandidátov. Pomocou algoritmov na získanie príznakov, sa zistili významné body kandidáta, a porovnávali sa s významnými bodmi vzorových značiek z predpripravenej databázy. Pričom sa vybrala vzorová značka s najviac spoločnými významovými bodmi. Autorka porovnávala viacero bežne používaných algoritmov (SIFT, SURF, MSER, Harrisov rohový detektor, FAST, BRISK). Výsledky tohto porovnania dopadli najlepšie pre algoritmus SIFT ktorý zdetegoval značky s 78% úspešnosťou, horšie dopadol SURF len s 56%. MSER A Harrisov rohový detektor dopadli horšie, ak našli zhodu so značkou väčšinou nesprávne. Najhoršie dopadli FAST a BRISK ktoré našli takmer minimum správnych značiek. Autorka pridala taktiež metódu na porovnávanie pixelov na rovnakých pozíciách, pričom vďaka kombinácii s algoritmom SIFT získala 93% úspešnosť. Práca bola realizovaná v jazyku MATLAB a na spracovanie obrazu bol použitý Image Processing Toolbox poskytujúci komplexnú sadu algoritmov na prácu s obrazom.

### 1.2.3 Detekcia textových dopravných značiek [5]

Ďalšou prácou zaoberajúcou sa detekciou dopravných značiek vypracoval Michal Zrubec v roku 2018. Cieľom jeho práce bola úspešná detekcia textových dopravných značiek. Pričom postupnosť krokov algoritmu je takmer totožný s predchádzajúcimi dvoma prácami.

**Riešenie:** Prvou fázou práce bolo nasnímanie videa na analýzu. Autor vybral rôzne typy záznamových zariadení s rôznou kvalitou videa. Taktiež sa zamerával na rôzne typy svetelných podmienok (slnečno, šero, počas dažďa..). Prvým krokom v predspracovaní videa bol downsampling, keďže videá boli natáčané vo vysokom rozlíšení kvôli čomu by mala väčšina algoritmov vysokú časovú náročnosť. Autor pevne určil rozlíšenie videa na 480x270 pixelov vďaka čomu znížil časovú náročnosť ale neznížil kvalitu až tak veľmi aby sa stratila niektorá dôležitá informácia z videa. Ďalším krokom bola konverzia z farebného modelu BGR, keďže autor pracoval s knižnicou OpenCV, ktorá ukladá snímky v tomto poradí farebných kanálov, do modelu HSV. Ako posledný krok autor vyhladil snímku pomocou mediánovho filtra. Pre porovnanie testoval aj bilaterálny filter, vďaka ktorému síce získal vyššiu úspešnosť ale za cenu väčšej časovej náročnosti. Po predspracovaní videa autor segmentoval snímku podľa farby. Pomocou prahovania snímku zmenil na binárnu kde biele oblasti boli tie ktoré spĺňali nastavené intervaly pre modrú, zelenú a hnedú farbu. Následne na takýto binárny obraz aplikoval morfológickú rekonštrukciu eróziou a v nájdených bielych oblastiach ešte vyplnil diery aby vznikla jednotná biela oblasť pre celý objekt. Po nájdení vhodných kandidátov autor začal v týchto oblastiach hľadať text. V tejto fáze pracoval s pôvodnou vstupnou snímkou pričom oblasť kandidáta bola jasne určená predchádzajúcimi fázami. Keďže značka môže obsahovať text bielej ale aj čiernej farby autor rozdelil kroky algoritmu pre každú farbu zvlášť. Takto získané vlastnosti potom porovnal pričom bral do úvahy či tvar spĺňa tvar obdĺžniku, pomer obsahu kontúry k opísanému obdĺžniku, uhol skolnu alebo či je šírka väčšia ako výška obdĺžnika. Ak kandidát spĺňal tieto vlastnosti bol vyhodnotený ako dopravná značka. Autor prácu implementoval v jazyku Python, pričom použil na spracovanie obrazu knižnicu OpenCV vo verzii 3.4. Celkovú úspešnosť mali najviac zelené značky s 95,5% úspešnosťou, ďalej nasledovali modré značky s 85% a najhoršie dopadli hnedé dopravné značky s 73,7% úspešnosťou.

### 1.2.4 Rozpoznávanie dopravných značiek a ich použitie v mapových aplikáciách [6]

Táto práca je zaujímavá z hľadiska klasifikátora, kde autor Štefan Tóth nepoužil bežné algoritmy porovnávania príznakov, ale na klasifikovanie vytvoril neurónovú sieť. Vďaka tejto práci vznikla knižnica .NET umožňujúca rozpoznávať a sledovať dopravné značky

rôznych druhov s možnosťou učením a tým ďalej zvyšovať počet nových druhov schopných aplikácia rozpoznávať.

**Riešenie:** Autor vybral fixnú veľkosť videa na 640x360 pixelov ktorá bola primeraná na kvalitu aj rýchlosť aplikácie. Na predspracovanie snímku využil Gaussovú filtráciu. Následne snímku prahovaním rozdelil na tri snímky každú podľa farby (červenú, modrú, žltú). Na takto získané binárne obrazy aplikoval morfológickú operáciu otvorenia vďaka čomu odstránil príliš malé objekty a väčšie sa zvýraznili. Aby odstránil objekty nevyhovujúce tvaru využil algoritmus na hľadanie kontúr a takto upravený obrazok následne klasifikovali natrénované neurónové siete rozdelené podľa farby. Na vstupe neurónovej siete je kandidát rozdelený podľa tvaru a pokračuje ďalej cez ďalšiu neurónovú sieť. Keďže takýmto postupom je možné značky detegovať iba na statických obrázkoch bolo potrebné ešte dopravnú značku sledovať vo videu. Autor si vybral Lucas-Kanadeho algoritmus pracujúci na princípe porovnania dvoch po sebe idúcich obrázkov ktorými sa prepočítavajú nové pozície určitých sledovaných bodov. Prácu autor implementoval v jazyku C# pričom úspešnosť detekcie a rozpoznania dopravnej značky je 70%, a úspešnosť neurónových sietí je okolo 85%.

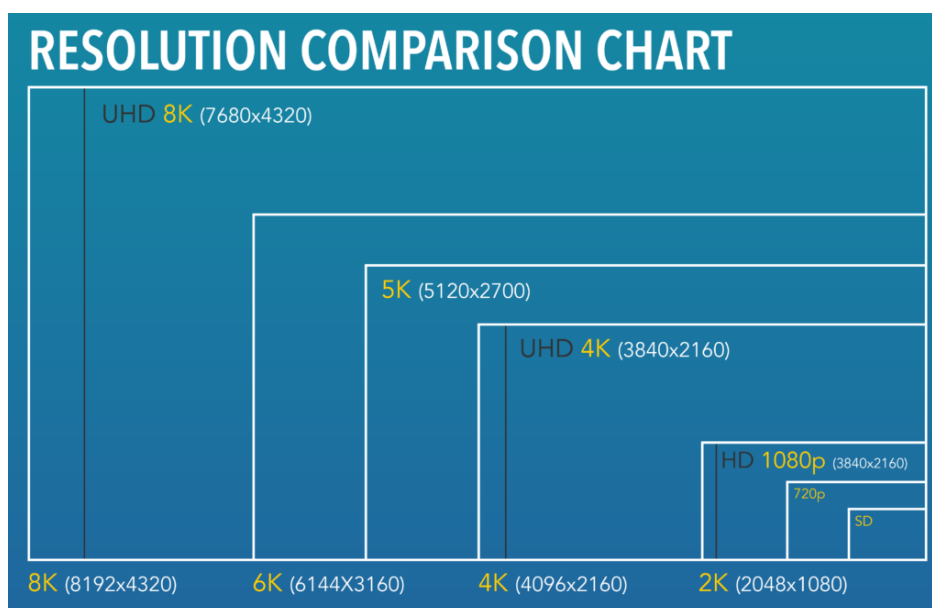
## 1.3 Teória k práci

V tejto časti sa zameriame na teóriu potrebnú k vypracovaniu témy a načrtneme problematiku počítačového videnia.

### 1.3.1 Snímanie videa

Video je získavané použitým mobilného zariadenia upevneného v aute tak aby záznam zachytával vozovku pred automobilom a dopravné značky popri ceste. Z hľadiska výpočtov a časovej zložitosti nás zaujíma rozlíšenie vstupného videa. Rozlíšenie videa sa značí ako počet pixelov (obrazový prvok) nachádzajúcich sa na jednej snímke. Rozlíšenie sa bežne značí počet pixelov na šírku krát počet pixelov na výšku. Medzi známe rozlíšenia pri pomere strán 16:9 patrí:

- HD (1280x720)
- Full HD (1920x1080)
- Quad HD (2560x1440)
- 4K (3840x2160)
- 8K (7680x4320)



Obr. 1.1: Ukážka rozdielov známych rozlíšení [7]

Ďalším faktorom je počet snímok za sekundu (fps), ktoré vie smartfón pri nahrávaní videa zachytiť. Čím vyššia kvalita videa tým sa počet snímok znižuje no dnes už telefóny nemajú problém zachytiť FullHD video pri 30fps (frames per second).

### 1.3.2 Predspracovanie videa

Po zachytení videa je zväčša ešte potrebné upraviť video pred ďalším spracovaním. Predovšetkým odstrániť nežiadúce javy ako je napríklad šum, alebo ak je potrebné vyhladiť snímku či pre lepšie spracovanie s obrázkom konvertovať do iného farebného modelu. Spracovanie obrazu môže prebiehať vo frekvenčnej oblasti alebo priestorovej oblasti. Medzi najznámejšie vyhladzovacie filtre v priestorovej oblasti patria:

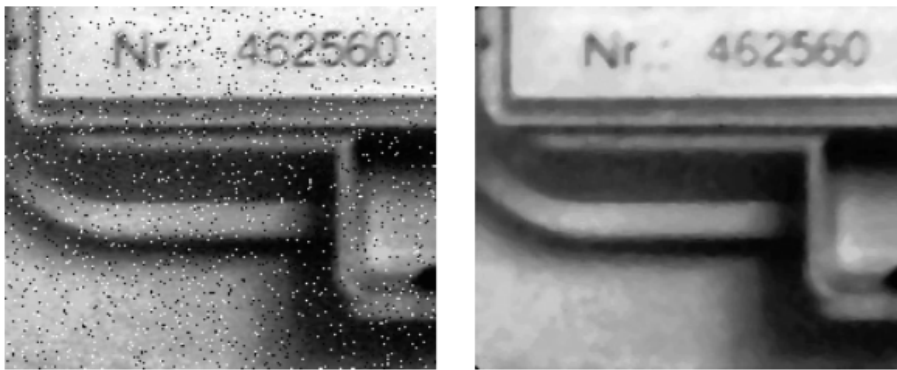
- Priemerovací filter - vypočíta hodnotu pixelu na základe aritmetického priemeru susediacich bodov okolia.
- Gaussov filter - často používaný filter, na rozdiel od priemerovacieho filtra pri vypočítaní hodnoty pixela používa masku vypočítanú Gaussovou funkciou, vďaka čomu sú výsledky presnejšie

Spríevodným javom pri vyhladzovaní je rozmazanie hrán. Pre hľadanie útvarov vo videu budú dôležité hranové detektory slúžiace na vypočítanie hrán a segmentáciu podľa tvaru. Medzi používané hranové detektory patria napríklad:

- Sobelov filter
- Cannyho hranový detektor



Obr. 1.2: Gaussov filter[8]



Obr. 1.3: Mediánov filter[9]

- Houghova transformácia

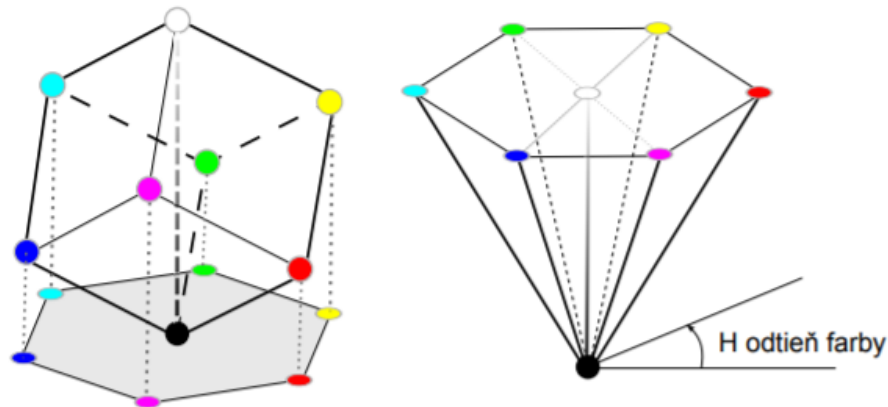
Často používaným filtrom je taktiež mediánový filter využívajúci nelineárnu filtráciu. Pričom zo zoradenej postupnosti danej zoradovacím filtrom vyberá prvok na strednej pozícii. Veľmi užitočný na odstránenie šumu typu soľ a korenie t.j. čierne alebo biele pixely rozmiestnené náhodne v obraze. Medzi ďalšie nástroje predspracovania obrazu patria morfológické operácie. Tieto operácie využívajú informáciu o susedných pixeloch v okolí spracovaného pixela. Základné morfológické operácie patria:

- dilatácia - pre vyplnenie dier v objektoch, alebo pri potrebe spojenia menších objektov do väčšieho celku
- erózia - odstránenie objektov menších ako vstupný objekt

a ich základné kombinácie sú:

- morfológické otvorenie - operácia erózie nasledovaná operáciou dilatácie
- morfológické uzavretie - operácia dilatácie nasledovaná operáciou erózie





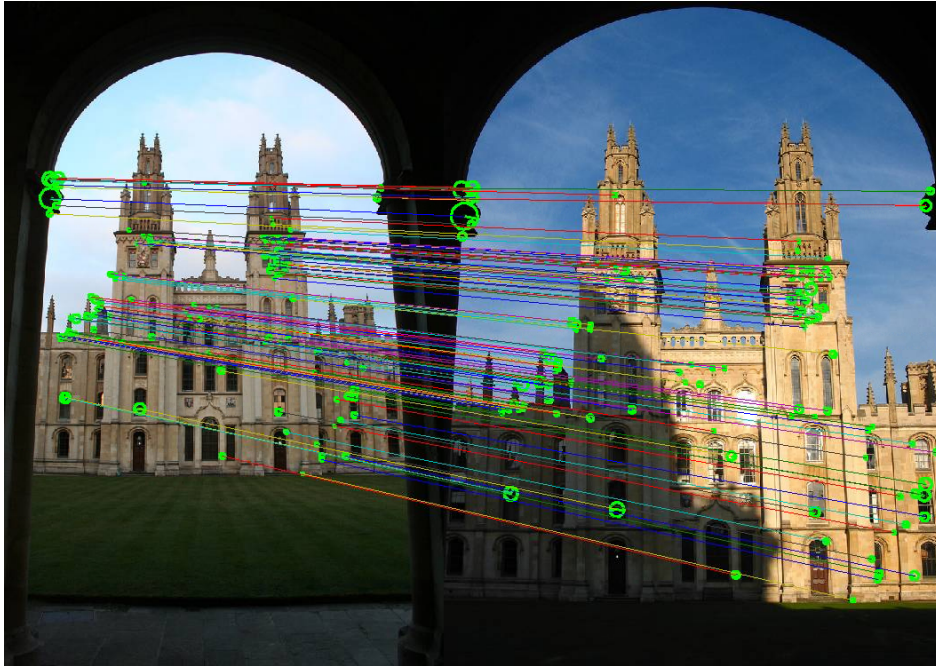
Obr. 1.4: RGB kocka a jej transformácia do HSV modelu[9]

### 1.3.3 Modely a priestory farieb

Farebný model je matematické vyjadrenie konkrétnej farby väčšinou  $n$ -ticou čísiel. Medzi najzákladnejší farebný model patrí RGB (Red, Green, Blue) kde primárne farby modelu sú červená, zelená, modrá. Ostatné farby sa vyjadria aditívnym skladaním týchto troch farieb. Model je reprezentovaný ako jednotková kocka a jednotlivé osi súradnicového systému reprezentujú množstvo príslušnej farebnej zložky v intervale  $\langle 0,1 \rangle$ . Model je v porovnaní s ostatnými modelmi najviac technicky orientovaný no pri práci s reálnymi obrazmi nie je veľmi efektívny. V hardvérovo orientovaných modeloch býva ťažké vyjadriť niektoré farby a preto vznikli používateľsky orientované modely ktoré sú navrhnuté pre intuitívnejšie vnímanie farieb. Základnými parametrami sú:

- farebný odtieň (hue),
- sýtosť farby (saturation),
- jasová zložka (value, lightness, brightness, luminance).

Medzi predstaviteľov takýchto modelov patrí HSV a HSL. Pre priestorové zobrazenie HSV modelu sa používa tvar obráteného šesťbokého ihlana kde vrch leží na začiatku súradnicovej sústavy v bode  $[0,0,0]$  reprezentujúci čiernu farbu. Tento tvar vzniká deformáciou RGB kocky kde diagonála vedúca z čierneho bodu do bieleho reprezentuje jasovú os. Sýtosť farby sa určuje ako vzdialenosť od stredovej osi po danú farbu. Farebný odtieň je uhlová súradnica z intervalu  $\langle 0,360 \rangle$ . Pri segmentácii na základe farby sa najčastejšie používa prahovanie. Táto metóda je časovo málo náročná, kde vstupný obraz transformuje na binárny rozdelením na popredie a pozadie na základe zadanej prahovej hodnoty.



Obr. 1.5: Porovnanie príznakov[10]

### 1.3.4 Klasifikátory

Na záver po vysegmentovaní oblastí kde by sa mohli dopravné značenia nachádzať treba určiť či kandidát je dopravná značka. Pri rozpoznávaní sa najčastejšie využívajú dva typy algoritmov. Prvým z nich je strojové učenie používané riadené klasifikátory (neurónové siete, rozhodovacie stromy, SVM..) alebo porovnávanie na základe príznakov. Medzi najznámejšie takéto algoritmy patria SIFT a SURF.

**SIFT** je škálovo a rotačne invariantný detektor a deskriptor zaujímavých bodov. A teda dokáže detekovať objekt aj keď prešiel zmenami ako sú škálovanie, rotácia, iný uhol etc.. Algoritmus deteguje zaujímavé body objektu a porovnáva so vzorovými obrázkami z databázy. Metóda SURF pracuje na rovnakom princípe no používa inú postupnosť krokov, vďaka čomu je rýchlejší ako SIFT.

## Kapitola 2

# Návrh postupu pri riešení problematiky

Hlavným cieľom pri vytváraní práce bolo vytvoriť mobilnú aplikáciu, ktorá by vedela úspešne detekovať z videa nasnímaného mobilným telefónom dopravné značenia rôznych druhov požívaných na území Slovenskej republiky. Postup riešenia vieme rozdeliť do jednoznačných krokov:

- Nasnímanie videa
- Úprava vstupného obrazu
- Predspracovanie obrazu
- Segmentácia podľa farby a tvaru
- Získanie kandidátov a ich SIFT/SURF descriptorov kandidátov
- Porovnanie príznakov kandidátov so vzorovými značkami
- Výsledok porovnávania

### 2.1 Nasnímanie videa

Video je snímané mobilným telefónom upevnenom na palubnej doske auta, pričom kamera bude snímať cestnú komunikáciu pred autom a dopravné značenia popri a nad cestou. Kamera telefónu posieľa do aplikácie snímky s ktorými následne aplikácia pracuje.

## 2.2 Úprava vstupného obrazu

Aby sa so snímkom čo najlepšie pracovalo je potrebné snímku ešte pred ďalším spracovaním upraviť na vhodnú veľkosť tak aby sa výpočtová zložitosť algoritmov znížila, no nemôže sa kvalita znížiť natoľko aby sa z obrazu stratila pre nás dôležitá informácia.

## 2.3 Predspracovanie obrazu

V tomto kroku sa odstraňujú chyby, ktoré môžu vzniknúť pri snímaní obrazu(šum, mŕtve pixely..), alebo znižujú vplyvy iných faktorov(počasie, osvetlenie..). Vhodný nástroj na vyhladenie obrazu je Gaussov filter, ktorý odstraňuje šum a malé detaily. Následne je obraz potrebné konvertovať z RGB modelu do vhodnejšieho farebného modelu pre lepšiu prácu s farbami obrazu. Vhodný model na túto prácu je farebný model HSV, ktorý pracuje s farbami spôsobom lepšie chápaním ľudským okom.

## 2.4 Segmentácia podľa farby

Obraz sa rozdelí na významné oblasti kde by sa mohla dopravná značka nachádzať. Prvou časťou je extrahovanie hľadanej farby (modrej, červenej, zelenej, hnedej) od ostatných pre nás bezvýznamných farieb. Pomocou prahovania sa obraz zmení na binárny, pričom do popredia sa dostanú nami hľadané farby. Pri hľadaní prahu je potrebné počítať aj so zmenou počasia a svetelných podmienok. Následne sa na takýto binárny obraz aplikujú hranové detektory na hľadanie tvarov v obraze. Vďaka špecifickým tvarom môžeme tvary redukovať na kruhové a štvorcové, pričom je potrebné dávať pozor na uhol z akého je objekt snímaný a podľa toho upraviť hodnoty aj pre takto deformované objekty.

## 2.5 Získanie kandidátov a ich SIFT/SURF descriptorov kandidátov

Po vyselektovaní oblastí kde by sa potenciálni kandidáti mohli nachádzať sa tieto oblasti vyberú pre zníženie časovej náročnosti, a pomocou SIFT a SURF descriptorov sa zistia ich významné body (descriptor).

## **2.6 Porovnanie príznakov kandidátov so vzorovými obrazmi**

Následne sa descriptory kandidátov porovnajú so získanými descriptormi zo vzorovej množiny značiek. Párovaním sa hľadajú rovnaké kľúčové body. A ako výsledok sa určí značka s najvyšším počtom spoločných bodov. Ak kandidát neprekročí minimálny počet vzájomných rovnakých kľúčových bodov so žiadnou dopravnou značkou, kandidát nie je klasifikovaný ako dopravná značka.

## **2.7 Výsledok porovnávanía**

Informovanie používateľa o výsledku porovnávanía. Či sa na obraze nachádza dopravná značka alebo nie.



# Kapitola 3

## Implementácia

Pri implementácii sme postupovali podľa návrhu z kapitoly 2. Implementácia je tvorená tromi hlavnými súbormi:

- activity\_main.xml
- AndroidManifest.xml
- MainActivity.java

Súbor activity\_main.xml obsahuje rozdelenie prvkou a grafickú časť aplikácie. V súbore AndroidManifest.xml sa nachádzajú povolenia potrebné pre aplikáciu. Posledný súbor MainActivity.java obsahuje funkčný kód aplikácie.

### 3.1 Použité technológie

Pre prácu s mobilným Android systémom použijeme vyšší programovací jazyk Java. Ide o objektovo-orientovaný programovací jazyk, ktorý sa začal vyvíjať už v roku 1991 a dnes patrí medzi najrozšírenejšie programovacie jazyky na svete, a momentálne najpoužívanejším na platforme Android. [11] Na prácu sme si vybrali vývojové prostredie Android Studio. Keďže väčšina práce je spojená so spracovaním obrazu a prácou s obrazom, budeme používať knižnicu OpenCV. Ide o open-source knižnicu obsahujúcu veľa funkcií na prácu s obrazom a jeho spracovaním. No vzhľadom na to, že posledné verzie neobsahujú algoritmy SIFT a SURF, ktoré neskôr používame. Budeme používať upravenú verziu knižnice OpenCV obsahujúcu niektoré neotestované alebo patentované funkcie, no však povolené na akademické a výskumné účely.[12]

### 3.2 Snímanie obrazu

Video je snímané mobilným telefónom položeným na palubnej doske aute. Na prácu sa používa mobilný telefón Motorola Moto G5 Plus s Androidom 8.1. Na snímanie tele-

fón využíva prednú duálnu kameru s objektívom veľkosti 12 megapixelov. Minimálne požiadavky na zariadenie je 4.4 (Kitkat) Android a fotoaparátom minimálne 4 Megapixelov. Aplikácia bola skúšaná za dobrých svetelných podmienok. Pri iných podmienkach (v noci, za dažďa..) nemusí aplikácia pracovať správne. Na začiatku aplikácie sa zavolá prvá metóda `onCreate`, ktorá spustí kameru. Keďže kamera zachytáva snímky vo veľmi veľkom rozlíšení kamera má obmedzenie rozlíšenia na maximálne 800x480 pixelov. Kamera ďalej posiela obrazy do metódy `onCameraFrame` kde sa ďalej tieto metódy spracovávajú.

### 3.3 Vzory značiek

Metóda `onCreate` taktiež volá metódy `redSignsTemplate`, `blueSignsTemplate`, ktoré slúžia na vytvorenie SURF descriptorov pre vzorové značky. Tieto descriptorov sa nemenia a preto ich stačí vygenerovať na začiatku aplikácie a neskôr ich už len porovnávať s najdenými kandidátmi.

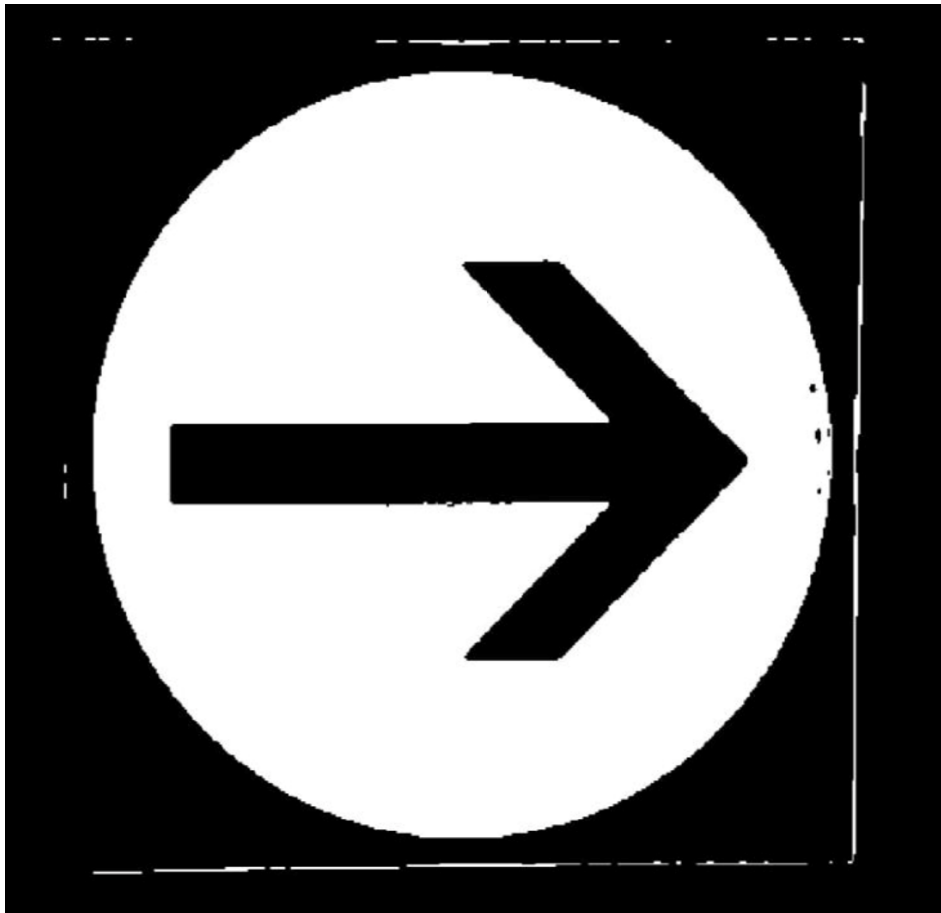
### 3.4 Predspracovanie videa

Kamera zachytáva snímky v RGBA modely, preto pre ďalšie spracovanie konvertujeme obraz do HSV modelu. Na odstránenie nedostatkov kamery, a nechcených javov zniknutých pri snímaní videa, vyhladíme snímku Gaussovým filtrom. Následne sa takto upravená snímka posiela na prahovanie do metód `blueSigns` a `redSigns`.

### 3.5 Segmentácia podľa farby

Nasleduje segmentácia hľadaných farieb. Metódy `blueSigns` a `redSigns` na začiatku vyprahujú červenú a modrú farbu pomocou funkcie `inRange`, pričom ako hodnoty treba zadať horný aj dolný prah. Keďže obraz je vo farebnom modely HSV prah pozostáva z troch hodnôt Hue(odtieň), Saturation(sýtosť) a Value(jas). Na vysegmentovanie červenej farby bolo potrebné sčítanie dvoch binárnych obrazov, pričom prvý mal prahové hodnoty  $\langle 150, 60, 0 \rangle - \langle 180, 256, 256 \rangle$ . A hodnoty pre druhý obraz sčítania obsahovali červenú farbu z rozsahu  $\langle 0, 60, 0 \rangle - \langle 10, 256, 256 \rangle$ . Tieto hodnoty sme získali z práce Miloša Fabiána z roku 2011.[3] Pre modrú farbu bolo segmentovanie jednoduchšie. Tu stačil iba jeden vysegmentovaný obraz. Modrú farbu sme získali prahovými hodnotami  $\langle 98, 102, 51 \rangle - \langle 10, 256, 256 \rangle$  z práce Anikoó Szaboóvej. [4] Prahovaním sme získali binárny obraz kde biela farba zastupuje nami hľadanú farbu a čierna zastupuje všetky ostatné farby. Na oba obrazy aplikujeme morfológickú operáciu otvorenia na odstránenie príliš malých objektov a takto upravené snímky ďalej pošleme na





Obr. 3.1: Segmentácia modrej farby

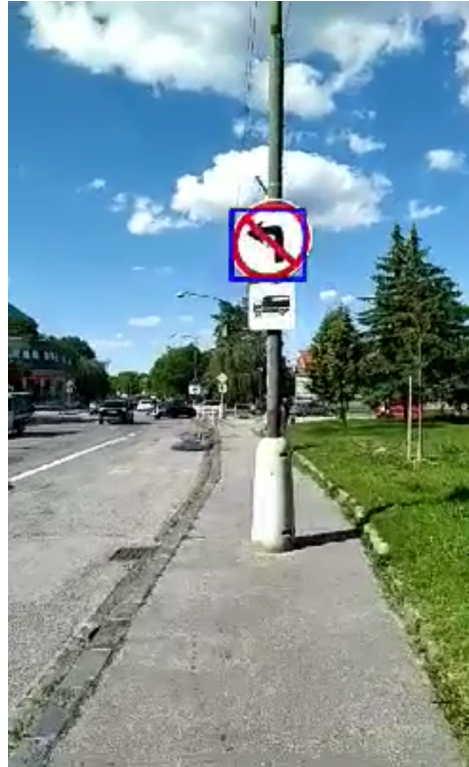
segmentáciu podľa farby a tvaru.

### 3.6 Segmentácia podľa tvaru

Na takto upravené binárne obrazy teraz môžeme zavolať hranové detektory na hľadanie tvarov v obraze. Pre nájdenie kruhu v snímke sme použili funkciu `HoughCircles` z knižnice `OpenCV`. Funkcia pomocou Houghovej transformácie hľadá kruhy v obraze. Na nájdenie vhodných kruhov potrebujeme zadať vstupné hodnoty:

- minimálnu vzdialenosť medzi dvoma stredmi (25 pixelov)
- minimálny polomer na detekciu (12 pixelov)
- maximálny polomer na detekciu (75 pixelov)

Tieto hodnoty sme opäť získali z práce Miloša Fabiána. [3] Funkcia `HoughCircles` nám vracia zoznam nájdených kruhov. Pre každý kruh vráti x-ovú a y-ovú súradnicu stredu a taktiež polomer kruhu. Vďaka týmto trom údajom vieme ďalej metóda `getROI` vy-



Obr. 3.2: Klasifikácia dopravnej značky

tiahnuť danú časť obrazu. Táto časť je náš kandidát a posielame ju ďalej na porovnanie so vzorovými značkami.

### 3.7 Porovnanie príznakov kandidátov so vzorovými značkami

Získaním kandidátom sa najprv zistia pomocou metódy `makeDescriptor` významové body. Metóda používa na získanie bodov algoritmus SURF kde pomocou metód OpenCV `detect` a `compute` sa vypočíta descriptor obsahujúci významové body. Takýto descriptor sa ďalej porovnáva s množinou descriptorov vzorových značiek v metóde `surf`. Kandidát sa porovnáva so všetkými vzorovými značkami svojej farby pričom sa hľadá najvyššia zhoda bodov. Ak je táto zhoda väčšia ako tri tak sa kandidát prehlasuje za dopravnú značku.

### 3.8 Informovanie používateľa o detekovaní značky

Pri nájdení kandidáta sa na snímke obrazovky tento kandidát ohraničí červeným štvorcikom pokiaľ sa kandidát nedetekuje ako dopravná značka. V tej chvíli sa už kandidát nachádza v modrom štvorci informujúc používateľa o nájdení dopravného značenia.

# Záver

Výsledok práce je mobilná aplikácia fungujúca na platforme Android detekujúca modré a červené dopravné značenia. Na začiatku sme si ukázali práce, ktoré s touto témou už vo svete pracovali. Ďalej sme si ukázali problematiku, ktorú téma počítačového videnia skrýva a jej základné princípy a pojmy. Ďalej sme ukázali návrh riešenia problematiky, pričom sme tento návrh rozdelili na základné kroky medzi, ktoré patrilo snímanie obrazu, priespracovanie, segmentácia podľa farby a tvaru, klasifikácia kandidátov a na záver vyhodnotenie porovnávania. Na záver sme sa pozreli na samotnú implementáciu aplikácie a postupy a metódy aké sme pri tejto aplikácii použili.



# Literatúra

- [1] Hours spent in road congestion annually, [online] Dostupné na internete: [https://ec.europa.eu/transport/facts-fundings/scoreboard/compare/energy-unioninnovation/roadcongestion\\_en](https://ec.europa.eu/transport/facts-fundings/scoreboard/compare/energy-unioninnovation/roadcongestion_en) zdroj: JRC, TomTom 2017.
- [2] Traffic sign, [online] Dostupné na internete: [https://en.wikipedia.org/wiki/Traffic\\_sign](https://en.wikipedia.org/wiki/Traffic_sign), last edited: 19 May 2020.
- [3] Miloš Fabian. Rozpoznávanie dopravných značiek. Bakalárska práca, 2012. Univerzita Komenského v Bratislave, Fakulta Matematiky, Fytiky a Informatiky.
- [4] Anikó Szabóová. Detekcia a rozpoznávanie dopravných značiek. Bakalárska práca, 2017. Univerzita Komenského v Bratislave, Fakulta Matematiky, Fytiky a Informatiky.
- [5] Michal Zrubec. Detekcia textových dopravných značiek. Bakalárska práca, 2017. Univerzita Komenského v Bratislave, Fakulta Matematiky, Fytiky a Informatiky.
- [6] Štefan Toth. Rozpoznávanie dopravných značiek a ich použitie v mapových aplikáciách. Bakalárska práca, 2011. Žilinská univerzita v Žiline.
- [7] Display Resolution, <https://medium.com/@yashwate/explained1display-resolution11a67a6096c>, Yash Wate 2017.
- [8] `imgaussfilt`, <https://www.mathworks.com/help/images/ref/imgaussfilt.html>, MathWorks.
- [9] Elena Šikudová, Zuzana Černeková, Wanda Benešová, Zuzana Haladová, and Júlia Kučerová. Počítačové videnie Detekcia a rozpoznávanie objektov. Vydavateľstvo Wikina, 2011.
- [10] Extracting invariant features from images using SIFT for Key-Point Matching, <https://blog.ekbana.com/extracting-invariantfeaturesfromimagesusingsift-forkeypointmatching675f818ce199>, Ashuta 2019
- [11] Java (programovací jazyk), [online] Dostupné na internete: [https://en.wikipedia.org/wiki/Java\\_programming\\_language](https://en.wikipedia.org/wiki/Java_programming_language), 2019.

- [12] Knižnica OpenCV Contrib, [online] Dostupné na internete: <https://github.com/chaoyangnz/opencv3androidsdkwithcontrib> , 2019.