

České vysoké učení technické v Praze
Fakulta dopravní
Ústav aplikované matematiky

Automatické stanovení směrových vztahů na křižovatce analýzou obrazu



Bakalářská práce

20. srpna 2007

Pavel Kolín

Čestné prohlášení

Prohlašuji, že jsem svou bakalářskou práci vypracoval samostatně a použil jsem pouze podklady uvedené v přiloženém seznamu.

Nemám závažný důvod proti použití tohoto školního díla ve smyslu §60 Zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon).

V Praze dne 20.srpna 2007

Pavel Kolín
podpis

Poděkování

Chtěl bych poděkovat všem, jejichž rady a nápady mi pomohly při tvorbě této bakalářské práce. Předně pak vedoucímu bakalářské práce, panu Dr. Ing. Janu Přikrylovi, který mi poskytoval cenné rady a ochotnou pomoc během celé doby tvorby této bakalářské práce. Poděkovat bych chtěl také všem kolegům z projektu „Inteligentní vozidlo“.

KOLÍN, Pavel. *Automatické stanovení směrových vztahů na křižovatce analýzou obrazu*.
Bakalářská práce. České vysoké učení technické v Praze, Fakulta dopravní, 2007.

Klíčová slova: videodetekce, virtuální smyčky, poměr odbočení, detekce pohybu, pohybová maska.

Abstrakt:

Tato práce má za cíl nalézt vhodnou metodu pro automatickou analýzu směrových vztahů na křižovatce s použitím videodetekce. K vyhodnocení směru odbočení jsou použity dvojice virtuálních smyček. Je zde srovnání několika metod tvorby pohybové masky a modelování pozadí. Práce se také zabývá vhodným barevným prostorem pro řešení dané úlohy.

KOLÍN, Pavel. *Automatické stanovení směrových vztahů na křižovatce analýzou obrazu*. Bachelor's thesis. Faculty of Transportation sciences, Czech Technical University in Prague, 2007.

Keywords: video-detection, virtual detectors, turning rate, motion detection, motion mask.

Abstract:

This thesis studies the process of automatic video-based analysis of turning movements at an intersection. In order to detect the turning cars, a pair of virtual detectors is placed over each direction considered. Several methods of movement mask construction and background modelling are studied. Also, the problem of selecting an appropriate color space for this task is studied.

Obsah

1	Poměr odbočení	12
1.1	Význam v dopravním plánování	12
1.2	Způsoby měření poměrů odbočení	12
1.2.1	Manuální metody	12
1.2.2	Automatizované metody	13
2	Virtuální smyčky	14
2.1	Proč používat smyčky	14
2.1.1	Princip detekce směrových vztahů za použití virtuálních smyček	15
2.2	Základní princip detekce	15
2.2.1	Vzhled použité smyčky	16
2.2.2	Vyhodnocení směru jízdy	17
3	Pohybová maska a modelování pozadí	20
3.1	Pohybová maska a její tvorba	20
3.1.1	Prostý rozdíl dvou snímků	20
3.1.2	Rozdíl snímků a pozadí	21
3.2	Úpravy masky	22
3.2.1	Prahování	22
3.2.2	Morfologické transformace	22
3.3	Pozadí	25
3.3.1	Co je to pozadí	25
3.3.2	Adaptivní modelování pozadí	25
3.4	Předzpracování vstupního snímku	29
3.4.1	Redukce šumu	30
4	Barevný prostor	33
4.1	Výhoda použití barevného obrazu	33
4.2	Transformace do jiného barevného prostoru	34
4.3	Stíny za vozidly	37
4.4	Odstranění není jednoduché	37
5	Software	39
5.1	Vstupní a výstupní data	39

5.2	Vzhled a ovládání programu	39
5.2.1	Hlavní program	39
5.2.2	Editor	40
5.2.3	Popis chodu programu	41
6	Sběr dat a zhodnocení funkce schopnosti aplikace	43
6.0.4	Použité vybavení	43
6.0.5	Lokalita	43
6.0.6	Účinnost automatického sčítání	44
6.0.7	Test č. 1 – Těšnovský tunel	45
6.0.8	Test č.2 – Čechův most	46
6.0.9	Test č. 3 – Letenský tunel na nábreží	47
6.0.10	Souhrnné zhodnocení	48
7	Závěr	49
A	Obsah přiloženého DVD	52

Seznam obrázků

2.1	Jednotlivé dráhy vozů při průjezdu křižovatkou.	15
2.2	Příklad umístění smyček na vozovce.	16
2.3	Geometrický tvar dvousmyčky a celé detekční oblasti.	16
2.4	Obsazení smyčky.	17
2.5	Čtyři možné směry natočení virtuální smyčky.	19
3.1	Maska tvořená prostým rozdílem dvou sousedních snímků.	21
3.2	Maska tvořená rozdílem aktuálního snímku a referenčního.	21
3.3	Ukázka vzhledu pohybových masek.	22
3.4	Grafické znázornění prahování.	23
3.5	Strukturální element $B \ 2 \times 2$	24
3.6	Zacelení masky pomocí morfologické operace uzavření.	24
3.7	Ukázka špatného použití morfologické operace uzavření, kde dochází ke splynutí více objektů vlivem příliš silného uzavření.	25
3.8	Znázornění vzhledu pozadí v porovnání s klasickou scénou.	26
3.9	Výchozí data pro ilustraci chybného modelování pozadí metodou akumu- lace snímků.	27
3.10	Znázornění chyby zaokrouhlování při použití metody akumulace snímků.	27
3.11	Vývoj hodnoty proměnné alfa v průběhu několika set snímků u modelování pozadí.	29
3.12	Porovnání dvou hladin prahové hodnoty s výsledkem redukce šumu při použití vysoké hladiny prahu.	30
3.13	Porovnání pohybové masky vzniklé pouhým oprahováním a masky vzniklé z rozostřených snímků s výsledkem redukce šumu v obraze.	32
3.14	Detail porovnání výsledků dvou metod odstranění šumu:(a) s vysokou hodnotou prahu a (b) s nízkou hodnotou prahu.	32
4.1	Porovnání množství detailů u pohybové masky vzniklé z černobílého a barevného snímku.	34
4.2	Znázornění významu jednotlivých kanálů HSV (převzato z [8]).	35
4.3	Ukázka vztahu barevného prostoru RGB a HSV (převzato z [8]).	35
4.4	Jednotlivé složky barevného prostoru HSV.	36
4.5	Transformace RGB na normalizované rgb.	37

4.6	Barevné složky podílející se na tvorbě stínu za vozidlem: RGB1 - Hlavní zdroj světla (slunce), RGB2 - světlo rozptýlené v atmosféře s barevným posunem do modré + jiné zdroje osvitu, RGB3 - vlastní sledovaný objekt (vozidlo), jehož povrch odráží světlo zbarvené RGB3, RGB4 - pozadí (vozovka), RGB5 - stín za vozidlem, který má ovšem jiné zbarvení než RGB4.	38
5.1	Ukázka výstupního formátu, zde pro interval záznamu 15 sekund.	40
5.2	Vývojový diagram zachycující důležité procesy v průběhu programu. . . .	42
6.1	Umístění kamery a rozmístění detekčních smyček na křižovatce Těšnovský tunel – Holbova při testu číslo 1.	45
6.2	Umístění kamery a rozmístění detekčních smyček na křižovatce Čechův most – nábrž.Edvarda Beneše při testu číslo 2.	46
6.3	Umístění kamery a rozmístění detekčních smyček na křižovatce Štefánikův most – nábrž.Edvarda Beneše – Letenský tunel při testu číslo 3.	47

Seznam tabulek

6.1	Naměřené a skutečné průjezdy v testu č. 1.	45
6.2	Naměřené a skutečné průjezdy v testu č. 2.	46
6.3	Naměřené a skutečné průjezdy v testu č. 3.	48
6.4	Souhrné zhodnocení provedených testů.	48

Seznam použitých zkratek a symbolů

A	obrázek aktuálního snímku
B	obrázek snímku
h	konvoluční maska
M	obrázek masky
M_p	obrázek oprahované masky s hladinou prahu p
P	obrázek pozadí
p	hodnota prahu
k	časový index
Y	luminance
α	proměnná zapisovaná do pozadí v průběhu modelování pozadí
β	váha nově zapisovaného snímku do pozadí
$\oplus$	dilatace
$\ominus$	eroze
$\circ$	operace uzavření
θ	okolí reprezentativního bodu
RGB	barevný prostor RGB
HSV	barevný prostor HSV
FPS	snímků za vteřinu
RZ	registrační značka

1

Poměr odbočení

1.1 Význam v dopravním plánování

Poměr odbočení (anglicky *Turning Rate*) je jedním z důležitých parametrů pro správné nastavení světelné signalizace na křižovatkách. Pomáhá rozhodnout o vybudování, či nevybudování samostatného odbočovacího pruhu v případě vytížené křižovatky. Může také posloužit jako podklad pro vybudování okružní křižovatky, neboť jednou z podmínek pro vybudování okružní křižovatky je, že levé odbočení je vyšší než 20 % (dle [5]).

1.2 Způsoby měření poměrů odbočení

Existuje několik metod, jak měřit počet vozidel projíždějících v daném směru. Obecně je můžeme rozdělit na metody manuální a automatizované.

1.2.1 Manuální metody

Doposud nepoužívanější metoda je *manuální sčítání*, kdy sčítač (osoba provádějící sčítání) stojí na vjezdu do křižovatky a zapisuje, kolik vozidel v jakém směru projelo. Podle počtu ramen křižovatky se určí i počet sčítačů, který odpovídá počtu ramen. Záznam je možné provádět:

- *psanou formou* do formuláře, který obsahuje kolonky různých směrů, do kterých se čárkami označují jednotlivé průjezdy vozů, nebo
- *elektronicky* s využitím jednoduché sčítačky, kde každému směru odpovídá jedno tlačítko, jehož zmáčknutím se průjezd započítá.

Nevýhoda této metody spočívá v nutnosti výhledu na celou křižovátku, lze ji tedy použít pouze na menších křižovátkách.

Tento handicap částečně odstraňovalo použití metody rozsvícených světél. Tato metoda je dnes již nepoužitelná z legislativních důvodů, kdy je povinné celodenní svícení. Uvádím ji však alespoň pro příklad.

Metoda rozsvícených světél spočívala v tom, že řidič byl před vjezdem do studované oblasti vyzván pomocí návěstních tabulí k rozsvícení světél. Na každém výjezdu sčítači zaznamenávali do formulářů vozidla s rozsvícenými světly. Po opuštění sledované oblasti byli opět řidiči vyzváni k zhasnutí a bylo jim poděkováno za spolupráci.

Nevýhodou této metody je hlavně fakt, že se dala použít v jednom časovém úseku jen pro jeden vjezd. Dala se také použít jen přes den a za dobrých světelných podmínek. V neposlední řadě je také nevýhodou to, že ne každý řidič byl ochoten spolupracovat [2].

Nejsofistikovanější, ale také nejnáročnější metodou je *zápis registrační značky vozidla* (RZ). Sčítači zapisují na každém rameni sledované oblasti RZ vstupujících a vystupujících vozů. Tyto údaje se musí později vyhodnotit (spárovat). Tato metoda je obzvláště vhodná pro velké okružní křižovatky, či dokonce pro celé oblasti.

Manuální metody jsou velmi náročné na sčítače, neboť to jsou poměrně nudné a monotónní činnosti, do kterých může lidský faktor vnést řadu chyb. Jsou také časově omezené a prakticky nelze jednu oblast sledovat nepřetržitě po několik dní.

1.2.2 Automatizované metody

Automatizací se odstraní chyby lidského faktoru, zjednoduší se celé měření a dostáváme také možnost monitorovat celou studovanou oblast nonstop po několik dnů. S prudkým rozvojem výpočetní techniky se začíná hojně využívat videodetekce, která nám dává možnost sledovat poměry odbočení pomocí automatické identifikace RZ.

Metoda automatického rozpoznávání RZ je velice výkonná a vhodná pro rozlehlé křižovatky, ale pro menší a přehlednější je příliš robusní. Vyžaduje totiž instalaci kamerového systému na každý vjezd a výjezd do (z) křižovatky. Tento systém je také poměrně náchylný na umístění – několik metrů nad vozovkou, vyžaduje tak vlastní konstrukci na které jsou kamery zavěšeny.

2

Virtuální smyčky

Virtuální smyčka je výřez v obraze sledované scény, nad kterým je sledována obsazenost vozidly. Nejčastějším tvarem je klasický obdélníkový tvar. Tvary se mohou lišit od místa umístění a typu sledovaného objektu. Ačkoliv pracují na zcela jiném fyzikálním principu než indukční smyčky, dávají stejné informace. Obsazení, respektive uvolnění virtuální smyčky, je vyhodnocováno na základě změn jasu jednotlivých pixelů v obraze nad sledovanou oblastí.

Smyčky, jak virtuální, tak i indukční, rozlišují pouze dva stavy: *smyčka obsazena* a *neobsazena*. Z těchto informací je možné určovat hodinovou intenzitu dopravního proudu [$\text{voz} \cdot \text{h}^{-1}$] a obsazenost v [%], danou poměrem obsazenosti smyčky ku periodě vzorkování. Pro měření rychlosti se používá nejčastěji dvojice smyček definované vzdálenosti, kde je rychlost vypočtena z časového odstupu náběžných hran obou smyček [5].

2.1 Proč používat smyčky

Virtuální smyčky mají několik předností. Tou nejdůležitější předností je právě jejich virtuálnost. Na rozdíl od klasických indukčních smyček není třeba jakkoliv zasahovat do vozovky. Smyčky je možné umístit kamkoliv, v různých tvarech, velikostech, množstvích a je možné je libovolně natáčet či upravovat v případě, kdy daná konfigurace z nějakého důvodu nevyhovuje.

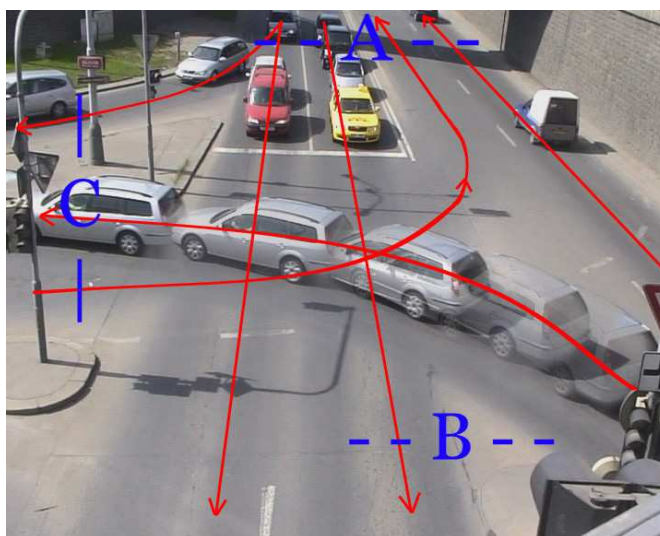
Virtuální smyčka dává stejné informace jako klasická indukční smyčka a má také pouze dva stavy – obsazená, volná. Z těchto dvou stavů se dá následně určit intenzita dopravy, rychlost (při použití alespoň dvou smyček za sebou) a při vhodném umístění dvojic smyček lze určit i směr jízdy.

Na sledování toku vozidel na křižovatce pak stačí jedna jediná kamera, která má

výhled na všechny vjezdy. V náročných podmínkách je možné použití dvou kamer, které snímají scénu z dvou různých míst, popřípadě úhlů. Ideální poloha kamery by byla kolmo nad středem křižovatky a ve výšce, tak bychom dostali nejpřehlednější video celé scény.

2.1.1 Princip detekce směrových vztahů za použití virtuálních smyček

Celá metoda se zakládá na jednoduchém faktu: Každému směru odbočení připadá jedna cesta, kterou vozidlo při tomto úkonu vykoná.



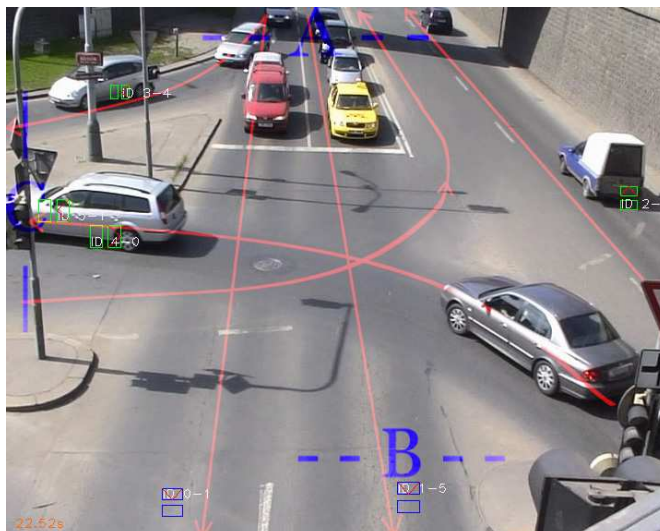
Obrázek 2.1: Jednotlivé dráhy vozů při průjezdu křižovatkou.

Stačí tedy na tyto cesty vhodně rozmístit virtuální smyčky a počítat jejich obsazení. Ve většině případů je poloha vjezdu a výjezdu vozidel do a z křižovatky v jednom směru identická, tedy přes jednu smyčku přejíždějí vozy ve dvou opačných směrech. Abychom tedy zjistili správný směr, je třeba používat dvojité smyčky, které jsou vhodně rozmístěné.

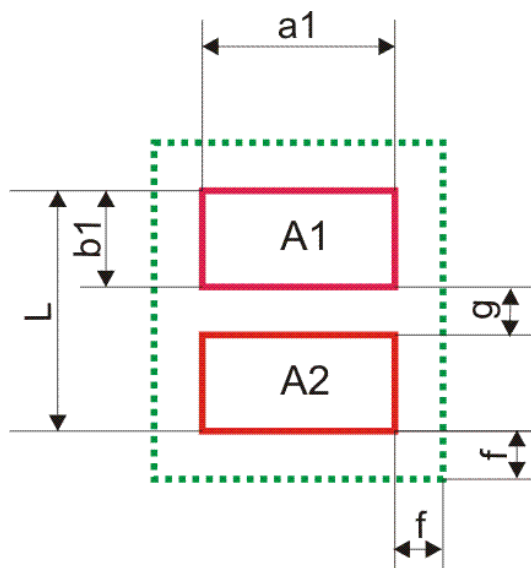
Vhodnost rozmístění závisí především na typu křižovatky, hustotě provozu a také na poloze kamery, která danou scénu snímá.

2.2 Základní princip detekce

Virtuální Smyčka je pravoúhlá oblast umístěná na vozovce, na které se testuje přítomnost vozidel. K tomu se využívá tzv. pohybová maska, což je binární obrázek snímané scény, ve kterém je vše pohybující znázorněno pixely bílé barvy (logická 1) a ostatní pixely jsou černé (logická 0). Nad každou smyčkou se počítá kolik pixelů je ve stavu log.1 (tedy bílých). Pokud počet aktivních pixelů ve smyčce překročí stanovenou hranici (tzv. prahovou hodnotu), dojde k obsazení smyčky. Následně pak při snížení počtu aktivních pixelů pod kritickou mez, dojde k opětovnému uvolnění smyčky.



Obrázek 2.2: Příklad umístění smyček na vozovce.



Obrázek 2.3: Geometrický tvar dvousmyčky a celé detekční oblasti.

2.2.1 Vzhled použité smyčky

Pro účely určení směru jízdy jsou zapotřebí minimálně dvě smyčky, které jsou umístěné v těsné blízkosti za sebou. Celá detekční oblast se tedy skládá ze dvou subsmyček, u kterých můžeme měnit jejich velikost a tím i detekční plochu.

Smyčky A1 a A2 jsou identické a poměr jejich stran a_1/b_1 a a_2/b_2 je konstantní. Tento poměr byl experimentálně odvozen a odzkoušen od rozměrů vozidel. Také velikost mezery g je odvozená od obou rozměrů smyčky a snaží se být co možná nejmenší. Důležitým rozměrem je totiž délka celé detekční oblasti, která je

$$L = 2 \cdot b_1 + g. \quad (2.1)$$



Obrázek 2.4: Obsazení smyčky.

Tato délka by neměla přesáhnout velikost mezery mezi jednotlivými vozy, které projíždějí přes detekční oblast. Pokud tomu tak nebude, mohlo by docházet ke špatnému vyhodnocení vlivem překrývání po sobě jedoucích vozidel.

Jednotlivé rozměry smyčky je třeba volit uvážene, neboť nelze jednoznačně říci jaký poměr stran a mezery je neoptimálnější. Každá kombinace přináší jisté výhody, ale i nevýhody.

Velikost mezery g . Menší rozměr této mezery, která odděluje obě subsmyčky $A1$ a $A2$, přináší výhodu především v tom, že zkracuje celkovou délku detekční oblasti. Tím s klesajícím rozměrem g klesá i možnost překrytí dvou vozidel vlivem celkově kratší délky L .

Nevýhodou ale je, že s klesající velikostí g roste pravděpodobnost projetí rychlého vozidla přes smyčky bez detekování. Tento efekt je způsoben nízkou frekvencí snímání kamery, kdy posunutí jedoucího vozidla na scéně na dvou po sobě jdoucích snímcích je natolik velké, že na prvním snímku vůz neobsadí ani jednu smyčku a na druhém snímku dojde k obsazení obou subsmyček najednou. Takováto sekvence je vyhodnocována jako chybná a nedojde tak k započítání průjezdu, neboť nelze určit směr jízdy.

Při rychlosti snímání 25 snímků za sekundu, což je standardní snímkovací rychlost běžných kamer, urazí vozidlo jedoucí rychlostí 50km/h dráhu 0,55m a pro rychlost 80km/h dráhu 0,88m. Tyto vzdálenosti platí pro případ, kdy křižovatku snímáme kolmo k vozovce. V praxi je možné snímat celou scénu pod úhly menšími než 90° , a tedy dráha vozu mezi jednotlivými snímky je podstatně menší. Teoreticky je tedy možné definovat minimální vzdálenost mezi hranicemi, kde dochází k obsazení smyček, a následně určit minimální velikost mezery mezi smyčkami. V praxi ovšem hrají roli i mnohé další faktory, jako například tvar auta nebo směr, ve kterém smyčku přejíždí.

2.2.2 Vyhodnocení směru jízdy

Vyhodnocení správného směru je zařízeno dvojicí spřažených smyček, které dohromady tvoří jednu detekční oblast. Smyčky jsou definované vždy pro jeden směr jízdy a pouze

pro definovaný směr započítávají průjezd danou oblastí. Správný směr je tedy určen podle následující sekvenční tabulky. Jedná se v podstatě o stavový automat, který hlídá předem definovanou posloupnost událostí.

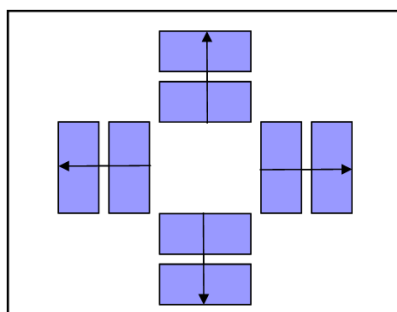
n	A1	A2	Průjezd
1	0	0	0
2	1	0	0
3	1	1	1
4	0	1	0
5	0	0	0

Sledované je pouze čelo jedoucího vozu a k započítání průjezdu dojde v okamžiku kroku 3, tedy když vůz obsadí i druhou smyčku v požadovaném pořadí. Před dalším průjezdem musí být opět celá dvojsmyčka uvolněna, jinak se průjezd nezapočítá. Celkové vynulování obou smyček je důležité, neboť u velkých objektů, jako jsou například kamiony a autobusy, by mohlo dojít vlivem nedokonalosti pohybové masky k dvojitmu započítání. Například kamion s návěsem, který má mezeru mezi kabinou a návěsem. U takového by pravděpodobně došlo k následující sekvenci:

n	A1	A2	Průjezd	Popis
1	0	0	0	volná smyčka
2	1	0	0	najíždí čelo kamionu
3	1	1	1	Obsazení obou smyček - započítání průjezdu
4	0	1	0	Kabina opouští A1 (mezera za kabinou)
5	1	1	0	Návěs obsazuje A1 zatímco A2 je stále obsazená kabinou
6	1	0	0	Kabina uvolnila A2
7	1	1	0	A1 i A2 jsou obsazeny návěsem
8	0	1	0	Návěs opouští smyčky
9	0	0	0	volná smyčka

Pokud bychom tedy neměli podmínku, že před dalším průjezdem musí být smyčka zcela volná, dostali bychom nesprávný výsledek v podobě dvou spočítaných průjezdů. Na druhou stranu podmínka úplného uvolnění obou subsmyček, komplikuje situaci v hustém provozu, kde mezera mezi vozy je nedostatečná a obsazují obě smyčky současně. Tomuto se dá ovšem poměrně dobře vyhnout tím, že smyčky umístíme pokud možno co nejdál od stopčáry a použijeme co nejmenší velikost. Pokud máme opravdu kvalitní pohybovou masku, tím je myšleno bez falešných děr a šumu – tedy celistvou, je možné použít opravdu malé smyčky v porovnání s velikostí vozidel.

Vzhledem k tomu, že každá dvojsmyčka detekuje pouze jeden jediný směr, je nutné smyčky na vozovce správně natočit, podle námi požadovaného sledovaného směru. Natočení smyček se neprovádí plynule v celých 360°, ale po krocích 90°. Tento krok byl zvolen z důvodu zjednodušení a zrychlení výpočetního procesu, neboť je pro tyto účely výhodné, když jsou strany smyček rovnoběžné s osami souřadného systému snímaného obrázku. V praxi se tento krok osvědčil a ukázal se jako dostačující pro všechny směry jízdy.



Obrázek 2.5: Čtyři možné směry natočení virtuální smyčky.

3

Pohybová maska a modelování pozadí

Pohybová maska a model pozadí je základem celé této aplikace. Na jejich kvalitě závisí spolehlivost a přesnost celého detekčního aparátu. Proto je nezbytné, se této problematice věnovat detailně.

Pohybová maska, neboli maska pohybu, je binární obrázek, který má v místě pohybu pixely bílé a v místech, kde k pohybu nedochází, pixely černé. Obrázkem v tomto pojetí je třeba chápat jako matici pixelů o rozměrech $m \times n$, kde každý pixel nese atributy intenzity jasu a barvy. Stejně jako matice, můžeme i obrázky sčítat, odčítat, popřípadě násobit.

3.1 Pohybová maska a její tvorba

Pro vytvoření pohybové masky existují dva základní principy. Jedním je odečítání dvou sousedních snímků a druhým způsobem je odečítání aktuálního snímku od pozadí.

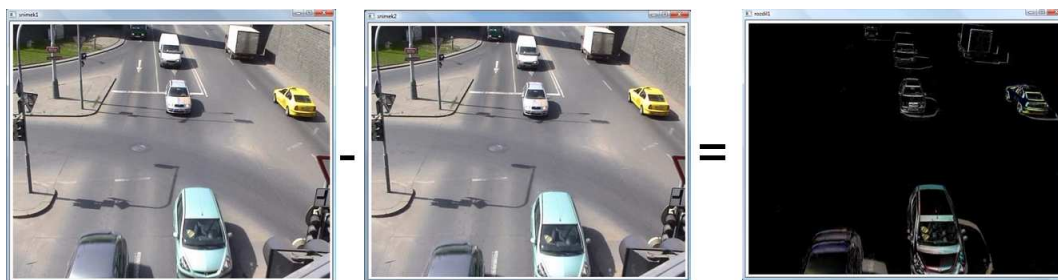
3.1.1 Prostý rozdíl dvou snímků

Jedná se o nejjednodušší a nejrychlejší metodu, jak získat pohybovou masku. Maska se získá jako absolutní hodnota rozdílů dvou po sobě jdoucích snímků,

$$M(x, y) = |A(x, y) - B(x, y)| \quad (3.1)$$

kde M je výsledná matice masky a A a B jsou matice sousedících snímků.

Kromě zmiňované výhody rychlosti a jednoduchosti nese tato metoda i řadu nevýhod. Například není schopna detekovat stojící vozidla, dále dává poměrně nedokonalou masku. Jak je vidět na Obrázku 3.1 maska není celistvá. Je to způsobeno tím, že na velkých plochách, jako jsou střechy a kapoty aut, nedochází k dostatečné změně kontrastu mezi



Obrázek 3.1: Maska tvořená prostým rozdílem dvou sousedních snímků.

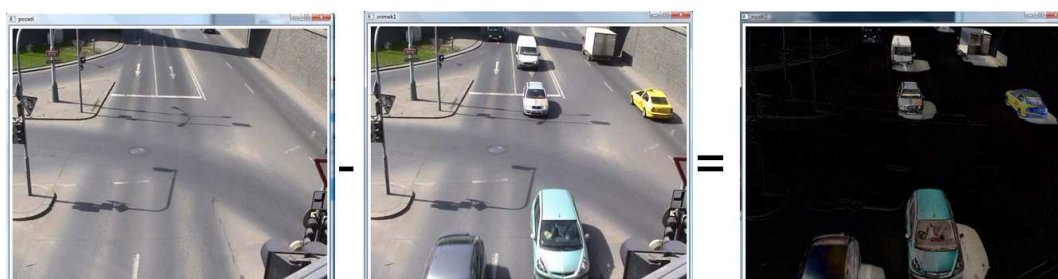
dvěma sousedními snímky. Jsou proto vidět dobře jen hranice objektů, což není příliš dostačující.

3.1.2 Rozdíl snímků a pozadí

Obě nevýhody, které prostý rozdíl obnáší, řeší metoda rozdílu aktuálního snímku od referenčního snímku pozadí. Od předchozí metody se liší jen v tom, že od aktuálních (sledovaných) snímků odečítáme statické pozadí, tedy stejnou scénu, ale bez vozidel,

$$M(x, y) = |A(x, y) - P(x, y)| \quad (3.2)$$

Zde je M opět výsledná matice masky, A je matice aktuálního snímku a P je matice referenčního snímku.



Obrázek 3.2: Maska tvořená rozdílem aktuálního snímku a referenčního.

Tato metoda je co do rychlosti a jednoduchosti stejně náročná jako prostý rozdíl, dává také velmi kvalitní masku. Jen v místech čelních skel je maska poměrně nedokonalá. To je způsobeno nízkým jasovým rozdílem mezi vozovkou a čelními skly, ve kterých se odráží obloha. Tyto odrazy je možné redukovat pomocí polarizačního filtru, který se nainstaluje na optiku snímací kamery. Tato úprava je jen teoretická, v praxi nebyla otestována, neboť použitá kamera takovouto úpravu neumožňovala.

U této metody je zapotřebí ještě získat model pozadí, tímto procesem se co do rychlosti tedy řadí na druhé místo, ale co do dokonalosti na místo první. Získání pozadí není jednoduchá záležitost, blíže se této problematice věnuje kapitola 3.3. Nejjednodušší ilustrativní metodou je výběr takového snímku, na kterém se nenacházejí žádná vozidla a

ten použit jako model pozadí. Jako model je nazýván z důvodu neustálých úprav a aktualizací v průběhu času. Scéna pozadí je sice statická co se pohybu týče, ale ve své optické jasové hodnotě se neustále obměňuje a je třeba tedy model pozadí neustále upravovat.

3.2 Úpravy masky

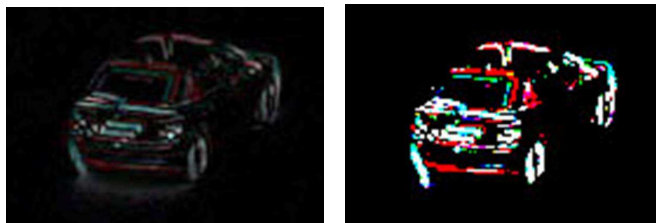
Doposud získané masky, nejsou maskami v pravém slova smyslu. Je třeba provést několik operací k jejich zdokonalení.

3.2.1 Prahování

Maska je binární snímek, neboli matice, kde její prvky nabírají pouze hodnot 0, 1. Zatím ale máme jen nedokonalou a nejasnou masku v plné paletě odstínů. Je tedy třeba provést tzv. prahování. Pro prahování je rozhodující hranice prahu, což je hodnota pomyslné linie (hranice), kde vše co se nachází pod ní je odstraněno a vše nad ní je propuštěno a zvýšeno na maximální hodnotu. Výsledkem tedy je skutečně binární snímek, kde každý pixel má jen dva stavy.

$$M_p(x, y) = \begin{cases} \text{MAX} & \text{pokud } M(x, y) \geq p \\ \text{MIN} = 0 & \text{pokud } M(x, y) < p \end{cases} \quad (3.3)$$

$M(x, y)$ je neoprahovaná maska, $M_p(x, y)$ je výsledná oprahovaná maska a p je hodnota prahu. Na Obrázku 3.3 je vidět názorný rozdíl mezi oprahovanou a neoprahovanou



(a) Neoprahovaná maska. (b) Oprahovaná maska.

Obrázek 3.3: Ukázka vzhledu pohybových masek.

maskou. Pomocí prahování se tak odstraní většina šumu, ovšem na úkor detailů na vozidle. Hladina prahové hodnoty se tak musí volit uvážene a pokud možno co nejnižší, abychom nepřišli o cenné detaily.

3.2.2 Morfologické transformace

Morfologická transformace je daná relace mezi obrázkem (bodová množina X) s jinou, typicky menší bodovou množinou B , které se říká strukturální element. Zatím je maska poměrně „děravá“, je tedy třeba ji upravit. K tomuto účelu jsou nejvhodnější morfologické operace eroze a dilatace.



Obrázek 3.4: Grafické znázornění prahování.

Dilatace $\oplus$ skládá body dvou množin pomocí vektorového součtu, např. $(a, b) + (c, d) = (a + c, b + d)$. Dilatace $X \oplus B$ je bodovou množinou všech možných vektorových součtů pro dvojice pixelů, vždy pro jeden z množiny X a jeden z množiny B

$$X \oplus B = \{p \in \epsilon^2 : p = x + b, x \in X, b \in B\} \quad (3.4)$$

Při použití na obrázek má tato operace za následek rozšíření okrajů prvků v obrázku. Jednotlivé prvky tedy jakoby „tloustnou“ [4].

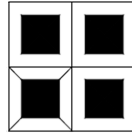
Eroze $\ominus$ je duální operací k dilataci. Eroze $\ominus$ skládá dvě množiny podle předpisu

$$X \ominus B = \{p \in \epsilon^2 : p + b \in X \forall b \in B\} \quad (3.5)$$

Použití eroze na úpravu snímku má za následek zúžení obrysů prvků v obrázku, tedy prvky „hubnou“ [4].

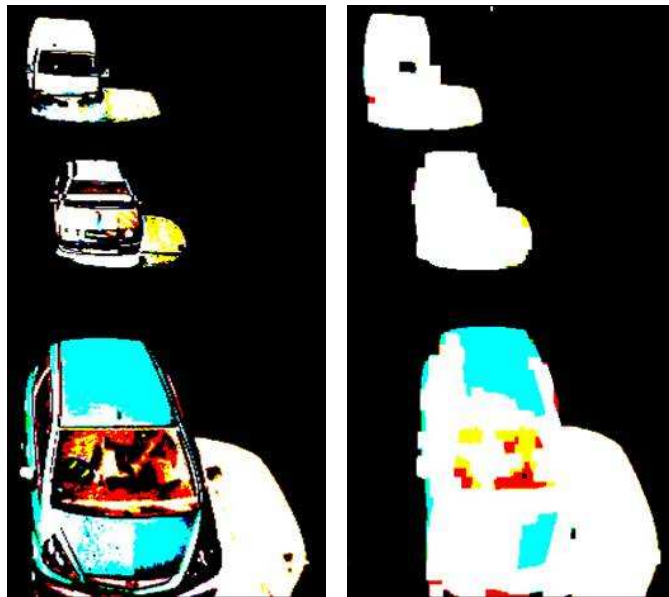
Uzavření je naše požadovaná funkce k zacelení masky pohybu. Jedná se o posloupnost použití nejprve dilatace a poté aplikací eroze.

$$X \circ B = (X \ominus B) \oplus B \quad (3.6)$$



Obrázek 3.5: Strukturální element $B \ 2 \times 2$.

V této konkrétní aplikaci používám komplikovanější a vícenásobné kombinace. V prvním kroku se provádí eroze se strukturálním elementem $B \ 2 \times 2$ pixely. Tato eroze má za úkol odstranit osamocené pixely v obraze, které vznikají šumem. V dalších krocích se provede několikanásobná dilatace, která má za cíl zaplnit drobné i větší otvory. Následně je zařazena několikanásobná eroze, neboť vlivem dilatace se obrázky poměrně zvětšily a je třeba je opět zmenšit. Vzhledem k tomu, že dilatace i eroze nejsou invertovatelné, získáme tak zacelený obraz stejné šíře jako byl původní.



(a) Výchozí maska.

(b) Uzavřená maska.

Obrázek 3.6: Zacelení masky pomocí morfologické operace uzavření.

Obrázek 3.6 ukazuje použití několikanásobné operace uzavření se strukturálním elementem $B \ 2 \times 2$. Výsledkem je poměrně kvalitní maska. Na Obrázku 3.6(b) je vidět jak uzavření postupuje od krajů do volné plochy. Jelikož čelní sklo posledního vozu bylo málo kontrastní, zůstala na jeho místě velká díra v masce. Operace uzavření ji velice dobře zredukovala na malý otvor.

Nebezpečí příliš silného uzavření (mnohonásobné uzavření, či uzavření za použití příliš velkých strukturálních elementů) spočívá v tom, že vozy mohou být „slepeny“ do jednoho celku, pokud je jejich vzdálenost příliš malá. Tato situace je znázorněna na Obrázku 3.7.



(a) Scéna, kde vozidla jsou příliš blízko.

(b) Oprahovaná a uzavřená maska, ve které došlo k propojení jednotlivých vozidel.

Obrázek 3.7: Ukázka špatného použití morfologické operace uzavření, kde dochází ke splnutí více objektů vlivem příliš silného uzavření.

3.3 Pozadí

Celá metoda detekce projíždějících vozidel je založena na pohybové masce, která musí být pokud možno co nejdokonalejší. Masku pohybu získáváme rozdílem aktuálního snímku s referenčním snímkem, neboli pozadím. Proto kvalita pohybové masky záleží především na kvalitě pozadí.

3.3.1 Co je to pozadí

Pozadí v tomto slova smyslu je statický obrázek, zobrazující nehybné objekty scény. Těmito objekty je například vozovka, sloupy veřejného osvětlení atd. Použitý výraz „statický obrázek“ není úplně pravdivý, neboť v pozadí dochází k postupným a pomalým změnám. Pozadí se musí neustále aktualizovat, neboť požadavky na detekci pohybu jsou ve vzájemném rozporu. Na jedné straně je třeba detekovat jedoucí vozidla, ale na straně druhé je nežádoucí detekovat pohyby stínů za statickými objekty. Nežádoucí jsou samozřejmě i stíny za jedoucími vozy, ale o tomto problému bude samostatně pojednáno v Kapitole 4.

3.3.2 Adaptivní modelování pozadí

Jelikož se pozadí vlivem pomalých změn, jako jsou přechody stínů či přechody mraků, mění, je třeba pozadí neustále upravovat, modelovat. Modelování lze provádět několika



(a) Scéna s vozidly

(b) Statické pozadí snímané scény.

Obrázek 3.8: Znázornění vzhledu pozadí v porovnání s klasickou scénou.

metodami, každá pak přináší jisté výhody i nevýhody. Vždy záleží na konkrétní aplikaci, v které bude model pozadí použit.

Akumulace snímků z video sekvence je nejjednodušší a nejrychlejší metoda tvorby a obnovování modelu pozadí. Základní myšlenka této metody spočívá v neustálém zápisu (akumulaci) aktuálních snímků, s jistou vahou, do pozadí. V jazyce matematiky to vypadá následovně:

$$P_k(x, y) = \beta \cdot A(x, y) + (1 - \beta) \cdot P_{k-1}(x, y) \quad (3.7)$$

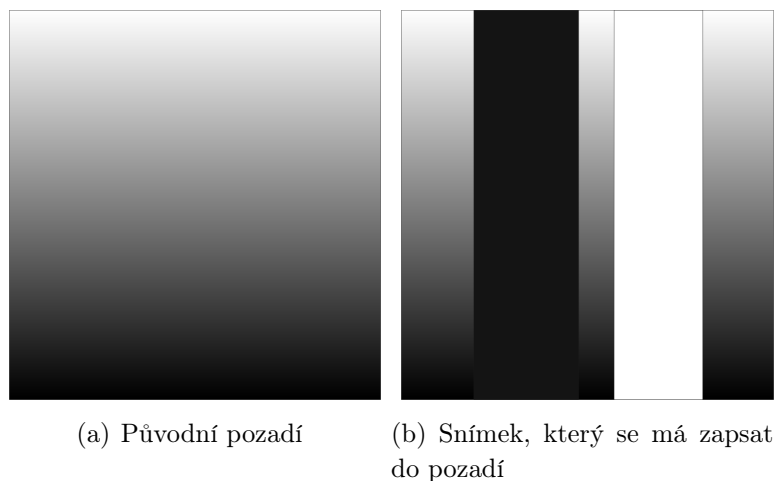
kde $P_k(x, y)$ je nové aktualizované pozadí, β představuje váhu zápisu nebo rychlost zapomínání, $(1 - \beta)$ je doplněk k hodnotě β a udává doplňkovou váhu starého pozadí, které je označeno jako $P_{k-1}(x, y)$. Váhová proměnná β tedy udává, jak rychle bude proveden přepis celého pozadí. $\beta \cdot 100$ je procentuálním vyjádřením váhy zápisu. Při hodnotě 100% bude pozadí kopírovat aktuální snímky a při hodnotě 0% zůstane pozadí neměnné.

U takto modelovaného pozadí vznikají za jedoucimi vozy „šmouhy“, které jsou způsobeny postupným zápisem do pozadí. Pomalé změny jsou tímto způsobem spolehlivě odfiltrovány.

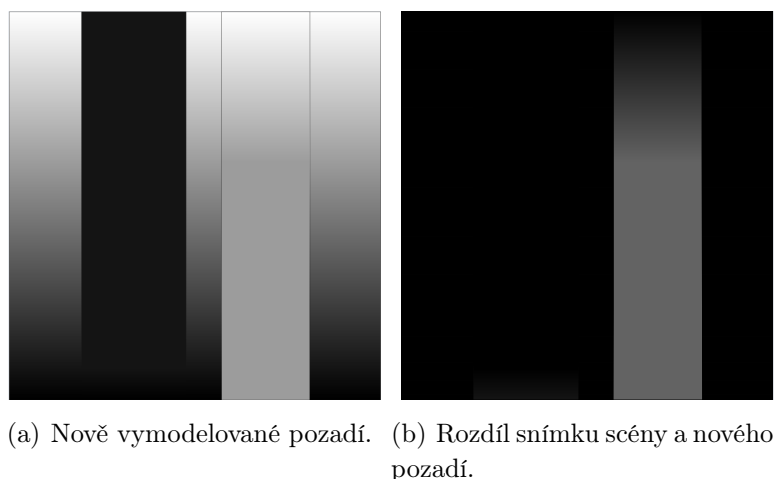
Dalším problémem této metody je, že vzhledem k zaokrouhlování v programu při každém cyklu, kdy se provádí modelování pozadí, dochází k posunu horní hranice jasu směrem k jasu starého pozadí. Pokud tedy z důvodů potřeby zvolíme hodnotu β příliš nízkou, nedosáhne nový model pozadí skutečných hodnot jasu. Tato chyba je vidět při zápisu bílé barvy.

Obrázek 3.10 ukazuje vzniklou chybu modelování pozadí. Velikost $\beta = 0.01$, tedy 1% váha zápisu. Je tedy vidět, že bílý pruh nebyl do pozadí zapsán v plné jasové hodnotě, ale jen jeho jistá část.

Selektivní akumulace je vylepšenou verzí klasické metody akumulace do pozadí, kterou jsme vymysleli spolu s kolegou Faturou na společném projektu Inteligentní Vozidlo. V této metodě k tvorbě pozadí je třeba vytvářet i pohybovou masku, pomocí které se obrázek rozdělí na tři části, kde každá část bude zapsána do pozadí s jinou va-



Obrázek 3.9: Výchozí data pro ilustraci chybného modelování pozadí metodou akumulace snímků.



Obrázek 3.10: Znázornění chyby zaokrouhlování při použití metody akumulace snímků.

hou. K rozdělení snímku na tři části se využívá neoprahaná pohybová maska, která se oprahuje pomocí dvou hladin prahu p_1 a p_2 . Tyto dvě hladiny rozdělí masku na tři části:

1. část s nejvyšší hladinou prahu. Vše nad touto hranicí je považováno za pohyb a nebude zapsáno do pozadí vůbec, nebo jen s velmi malou vahou. Varianta zápisu s velmi malou vahou slouží k ochraně před "zamrznutím" nějakého nežádoucího objektu v pozadí.
2. část, tedy oblast mezi prahy p_1 a p_2 , je považována za významný pomalý pohyb, který je třeba zapsat do pozadí.
3. část, tedy oblast pod nejnižší prahovou hladinou představuje šum v obraze a opět nebude zapsána do modelu pozadí vůbec, nebo jen s velmi malou vahou.

Vzhledem k tomu, že se do pozadí nezapisují rychle pohybující objekty, nevznikají tak nežádoucí „šmouhy“ za vozidly. Také vozidlo, které zastaví na křižovatce, se do pozadí zapíše až po výrazně delší době, než tomu bylo u klasické akumulace.

Tato metoda vyžaduje mít k dispozici prvotní model pozadí ihned po startu. Jelikož tato metoda využívá pro své účely pohybovou masku, která je tvořena rozdílem referenčního a aktuálního snímku, nastává problém, jak získat prvopočáteční model pozadí, který se bude dále již jen aktualizovat.

Jednou z možností je ručně vybrat snímek, na kterém se nenacházejí žádná vozidla, a předat ho algoritmu jako pozadí. Toto řešení je v několika ohledech silně nepraktické. Za prvé samotné ruční vyhledávání je nepohodlné a za druhé v silně saturovaných oblastech může nastat situace, že nedojde k úplnému uvolnění křižovatky nikdy. Pro účely detekce pohybu pouze nad virtuálními smyčkami ovšem stačí, pokud se žádné vozidlo nevyskytuje nad virtuální smyčkou. V tomto případě je ruční výběr daleko jednodušší.

Další možnost je použití jiné metody tvorby masky pohybu. Například metoda rozdílů dvou sousedních snímků je pro tyto účely vhodná. V prvních krocích chodu programu bude tedy pomocí metody rozdílů dvou po sobě jdoucích snímků vytvořen počáteční model pozadí, který se předá metodě rozdílů aktuálního snímku a referenčního. Stále zůstává ale problémem, kdy říci, že je pozadí již vytvořené. Také váhy zápisu do pozadí je třeba v průběhu prvopočátečního modelování korigovat dle potřeb. Nejjednodušší a nejspolehlivější je korekce pomocí operátora, který program v počátku ručně řídí.

K řízení je zapotřebí mít možnost měnit váhu zápisu do pozadí. Operátor sleduje pohybovou masku a podle výsledků ručně zvyšuje či snižuje váhu zápisu až do okamžiku, kdy je maska kvalitní a v modelu pozadí není zapsáno nic, co do něj nepatří, což může nejlépe ohodnotit zas jen člověk.

Zápis konstantní hodnoty do pozadí řeší problém chybného zaokrouhlování u metody akumulace pozadí. Do pozadí se zapisuje stále stejná hodnota, která je odvozena od maximální absolutní hodnoty jasu pixelu. Pro osmibitovou hloubku se tedy jedná o podíl z 255 a pro šestnáctibitovou hloubku o podíl 65535.

Tento podíl, který nazývám α , se do pozadí zapisuje následovně:

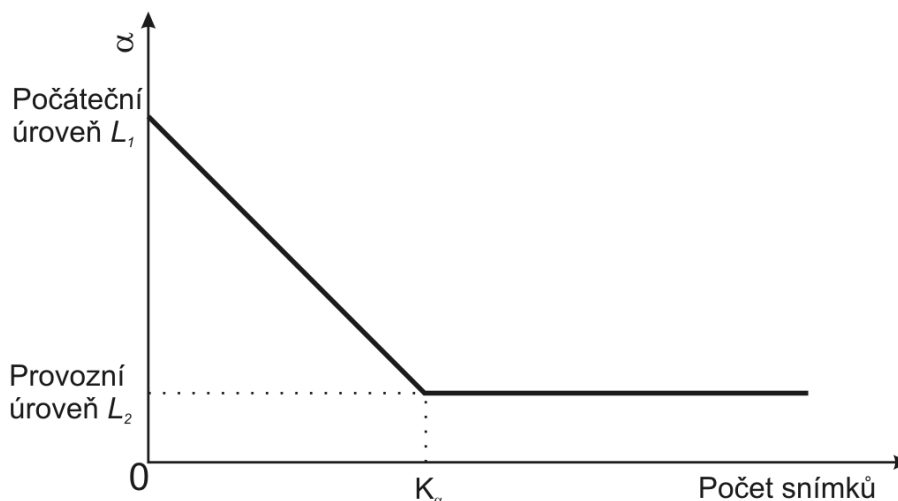
$$P_k(x, y) = \begin{cases} P_{k-1}(x, y) + \alpha & \text{pokud } P_k(x, y) < A(x, y) \\ P_{k-1}(x, y) - \alpha & \text{pokud } P_k(x, y) > A(x, y) \end{cases} \quad (3.8)$$

Kde $P_k(x, y)$ představuje nové pozadí, $A(x, y)$ představuje aktuální snímek a α představuje proměnnou, která je zapisována do pozadí.

Tento princip se snaží kopírovat režim dlouhé expozice, kde pro vytvoření obrazu je potřeba toku slabého světla po dostatečně dlouhou dobu. V důsledku dlouhého expozičního času se pohybující objekty nezachytí. Přičítání i odčítání konstanty α má schopnost korekce jasu v obou směrech.

Čím vyšší bude hodnota této konstanty, tím rychleji dojde k vytvoření počátečního pozadí. Vyšší hodnota ale znamená nižší rozlišovací schopnost, a tedy vyšší hrubost modelu pozadí. V praxi se toto projevuje probíháním na hraničních hodnotách jasu. Je

tedy velmi žádoucí, po natrénování prvotního modelu pozadí hodnotu α snížit. Snížením se navíc docílí i pomalejšího zápisu stojících vozidel do pozadí.



Obrázek 3.11: Vývoj hodnoty proměnné alfa v průběhu několika set snímků u modelování pozadí.

Konkrétní aplikace softwaru pro vyhodnocení směrových vztahů na křižovatce používá kombinovanou metodu selektivního zápisu konstantní hodnoty. Průběh hodnoty α se vyvíjí podle grafu na Obrázku 3.11, kde je proměnná α lineárně snižovaná od počáteční hodnoty $\alpha = L_1$ (počáteční úroveň) až do hodnoty $\alpha = L_2$ (provozní úroveň). Pro dosažení úrovně L_2 je třeba projít K_α snímků od počátku. Obě hodnoty, tedy počáteční úroveň a provozní úroveň, jsou zvoleny experimentálně. L_1 se řádově pohybuje v rozmezí 200-400 a L_2 v rozsahu 5-25. V programu využívám šestnáctibitovou hloubku, tedy 65535 odstínů jasu.

Selektivita zápisu do pozadí je řízena již vytvořenou maskou, tedy v místech, kde je detekován pohyb, se do pozadí zapisuje proměnná $\alpha = 5$. Tato hodnota zajistí velmi pomalý zápis pohybujících se vozidel. Vzhledem ke stále přetrvávajícímu šumu nelze použít hodnotu $\alpha = 0$, neboť by se časem celé pozadí přestalo obnovovat a tím by se maska znehodnotila.

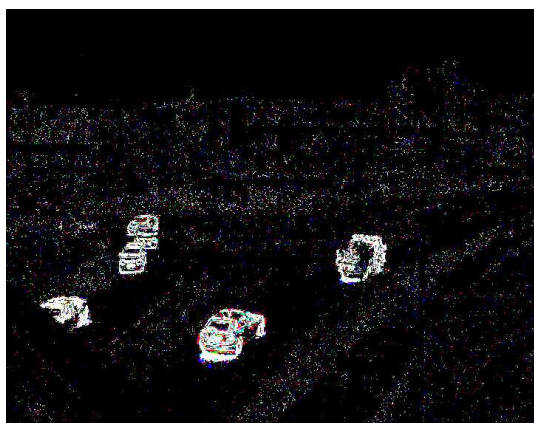
3.4 Předzpracování vstupního snímku

Kvalita vstupního videa, které se má analyzovat, je pro přesnost vyhodnocení velmi důležitá. Největšími problémy jsou šum v obraze a pohyby kamery, která by měla být statická.

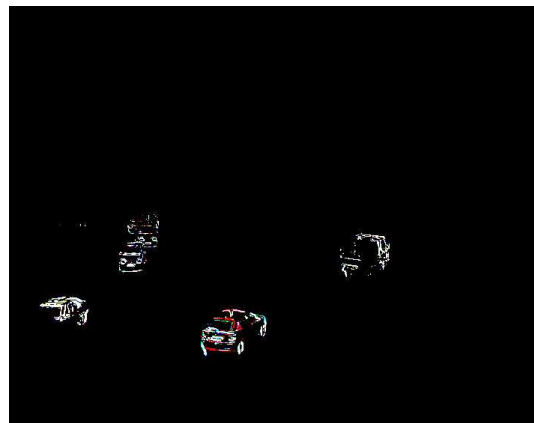
3.4.1 Redukce šumu

Šum je všudypřítomná záležitost, která se v obraze vyskytuje. Příčiny jeho vzniku mohou být různé. Jedním z nejčastějších šumů je tzv. tepelný (Johnsonův) šum. Je způsoben náhodným pohybem elektronů v krystalové mřížce polovodiče, je závislý pouze na teplotě, nezávisí na napětí, proudu ani frekvenci [6]. CCD snímače bezesporu teplé jsou, a tak se i na nich tento šum výrazně podepisuje. Hladina tohoto šumu roste s rostoucí citlivostí snímacího čipu, neboť s klesajícím osvětlením a rostoucím zesílením se šum dostává do popředí. Ve výsledku se šum projevuje v obraze jako různobarevné zrnění. Podobně se ovšem chová i déšť či sníh. Odstraněním šumu ze scény odstraníme i rušivé elementy jakými sněžení či déšť jsou.

Redukce šumu za použití prahování Šum v obraze je závažný problém, jeho úroveň může do značné míry ovlivnit kvalitu tvorby pohybové masky. Nejjednodušším způsobem jak se vypořádat s malou hladinou šumu je použití prahování. Velikost prahu je nutné volit tak, aby odstranil nežádoucí šum. S rostoucí hladinou prahu se ale spolu s šumem ztrácí i cenné informace které požadujeme. Na Obrázku 3.12 je názorná ukázka odstranění



(a) Nízká hladina prahu - šum v obraze.



(b) Vysoká hladina prahu - šum odstraněn.

Obrázek 3.12: Porovnání dvou hladin prahové hodnoty s výsledkem redukce šumu při použití vysoké hladiny prahu.

šumu pomocí oprahování. Stejně jako je vidět markantní úbytek šumu v Obrázku 3.12(b) při vysoké hladině prahu, tak je i vidět značný úbytek detailů na jednotlivých vozidlech.

Redukce šumu rozmazáním Každý pixel je zatížen aditivním náhodným šumem, který je:

- statisticky nezávislý,
- s nulovou střední hodnotou μ ,
- a nějakou směrodatnou odchylkou σ .

Vzhledem k náhodnému charakteru šumu a jeho statistickým vlastnostem, je tedy možné šum odstranit zprůměrováním několika identických snímků stejné scény. Těto metodě

se říká: Statistický princip filtrace šumu [7]. Tuto metodu ale není možno uplatnit v této konkrétní aplikaci, neboť každý následující snímek je jiný. Jistou možností by bylo snímání jedné scény za pomoci dvou či více kamer. Takovýto přístup by byl značně neekonomický a nepraktický (drahé a složité zařízení). Nezbyvá, než se spolehnout na nadbytečnost informací v obraze a provést zprůměrování několika okolních pixelů v obraze. Sousední pixely mají převážně stejnou barvu nebo podobnou hodnotu jasu, lišit se budou především vlivem šumu. Problematickou partií jsou přechody a hrany, na kterých se projeví snížení kontrastu.

Nová hodnota každého pixelu se odhaduje z malého okolí Θ kolem reprezentativního bodu. Toto okolí bývá často malým obdélníčkem, či čtverečkem. Celý obraz se prochází postupně (po řádcích, pixel po pixelu). Často se používá okolí, které má lichý počet řádků a sloupců, a tak může reprezentativní pixel ležet uprostřed.

Při váženém průměrování se využívá konvoluční masky, která přiřazuje váhu jednotlivým pixelům uvnitř okolí Θ . Příspěvky jednotlivých pixelů v okolí Θ jsou váženy v lineární kombinaci koeficienty h . Výsledný obraz je pak:

$$A_{x,y} = \sum \sum_{m,n \in \Theta(x,y)} h(x-m, y-n) f(m,n), \quad (3.9)$$

kde h představuje konvoluční masku. Tato maska může nabývat nejrůznějších hodnot, například:

$\bullet \quad h = \frac{1}{9} \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$	Tato matice nezvýhodňuje střed.
$\bullet \quad h = \frac{1}{10} \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{pmatrix}$	Zvýhodnění středového – reprezentativního bodu.
$\bullet \quad h = \frac{1}{16} \begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix}$	Středové zvýhodnění.

Další informace o předzpracování obrazu lze najít v [7] a [4]. Účinnost této metody je nejlépe vidět na výsledné oprahované pohybové masce. Pro srovnání předkládám na Obrázku 3.13 porovnání dvou pohybových masek se stejnou hladinou prahu.

Pro porovnání obou metod předkládám detail vozidla vyříznutého ze scény. V obou scénách byl odstraněn šum. V Obrázku 3.14(a) byl odstraněn za použití vyšší hladiny prahování a u Obrázku 3.14(b) pomocí rozmazání a oprahování s nízkou hladinou prahu.

Jak je vidět v Obrázku 3.14, metoda rozostření zachovává vlivem možnosti použít nižší hladinu prahování více detailů. Tato metoda je také využita v programu pro automatickou analýzu směrových vztahů v křižovatce.



(a) Pouze oprahovaná pohybová maska.



(b) Maska vytvořená z rozostřeného snímku při stejné hladině prahu jako u 3.13(a).

Obrázek 3.13: Porovnání pohybové masky vzniklé pouhým oprahováním a masky vzniklé z rozostřených snímků s výsledkem redukce šumu v obraze.



(a) Odstranění šumu vysokou hladinou prahu – detail.



(b) Odstranění šumu pomocí rozmazání a nízkou hladinou prahu.

Obrázek 3.14: Detail porovnání výsledků dvou metod odstranění šumu: (a) s vysokou hodnotou prahu a (b) s nízkou hodnotou prahu.

4

Barevný prostor

4.1 Výhoda použití barevného obrazu

Při zpracování snímků je třeba si uvědomit, zda použití barevného schématu je přínosem, či nikoliv. Barevný prostor přináší totiž velké množství redundantní informace. Oproti černobílému formátu je také u barevné scény třeba zpracovat $3\times$ více obrazových pixelů. Toto navýšení objemu zpracovávaných dat přináší zpomalení celé aplikace.

Řešení této otázky je v kontrastu jednotlivých kanálů. Budu-li uvažovat osmibitovou hloubku, dostávám tak 256 možných odstínů na každý kanál. Mějme standardní barvu vozovky, tedy šedou, která odpovídá přibližně 50% zastoupení od každého barevného kanálu, tedy $R = 128$, $G = 128$, $B = 128$, kde R,G,B jsou jednotlivé základní barvy (červená, zelená a modrá). V černobílém zobrazení je tato šedá znázorněna opět hodnotou 128 v jasové stupnici od 0 do 256. Jelikož při tvorbě pohybové masky je počítána vzdálenost od této šedé oblasti, je maximální možná vzdálenost 128 na obě strany (u černobílého modelu). V barevném podání je to již 3×128 na každou stranu. Projíždějící vozidla jsou šedá jen vzácně, tedy poměr jednotlivých kanálů nebude $1 : 1 : 1$. Co je ale podstatnější je fakt, že pokud jeden barevný kanál dominuje před ostatními (červené, modré, zelené a jejich kombinace – krom bílé), převodem do černobílé se na měřítku kontrastu posouvá blíže šedé, tedy zaniká jeho dominance a zbytečně se ztrácí velké množství dat. Pohybová maska z barevného snímku je podstatně dokonalejší, je to způsobeno především pestrostí barev jednotlivých vozidel. Pro černá, bílá a barevně nevýrazná auta by se vyplatilo používat černobílé snímky, ale takových vozů na našich silnicích mnoho nejezdí.

Barevné zpracování sice přináší mírné zpomalení běhu celé aplikace, ale vzhledem k výkonnosti dnešní výpočetní techniky, se nejedná o nijak markantní rozdíl.



(a) Maska získaná použitím černobílého snímku.



(b) Maska získaná s použitím barevného snímku.

Obrázek 4.1: Porovnání množství detailů u pohybové masky vzniklé z černobílého a barevného snímku.

4.2 Transformace do jiného barevného prostoru

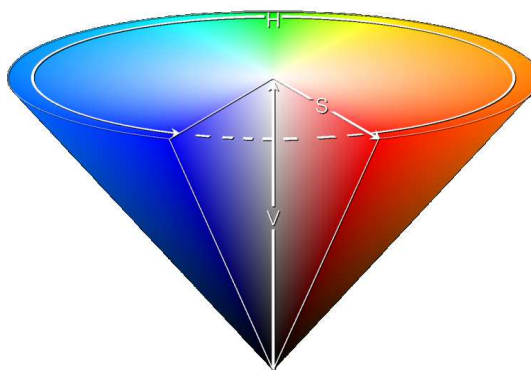
Motivací pro převod snímané scény do jiného barevného prostoru byla snaha o odstranění nebo alespoň o redukci stínů za projíždějícími vozy. Prostor RGB není totiž proporcionálně lineární a odstranění stínů v něm je téměř nemožné. Změna jasu u barevného obrázku v RGB má za následek i posun jednotlivých barevných kanálů, neboť pro převod barevného snímku z prostoru RGB do stupňů šedi, reprezentovaných luminancí Y platí

$$Y \cong 0,3 \cdot R + 0,59 \cdot G + 0,11 \cdot B. \quad (4.1)$$

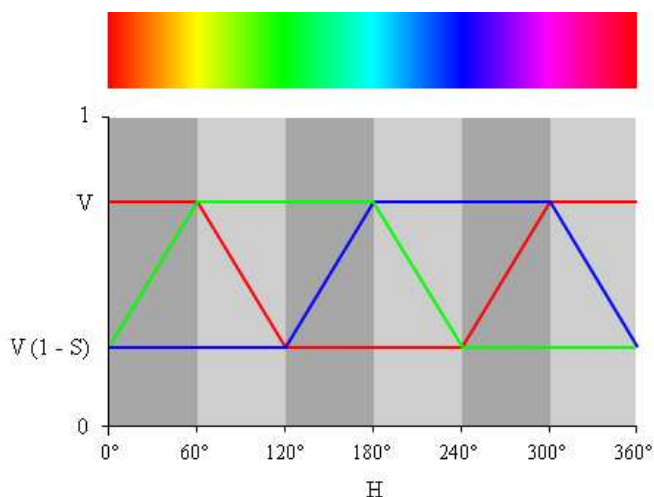
Zajímají mne tedy například barevné prostory HSV, či lineární barevný prostor RGB, tzv. normalizované RGB.

Odstranění stínů se zakládá na teorii, že osvětlená scenerie a tatáž scenerie ve stínu má stejný odstín barvy, která se liší jen v jasové složce. Formát HSV se zdál být pro naši věc nejideálnější, protože obsahuje tři složky: Hue, Saturation, Value neboli Barva, Saturace-sytost, Jas. Hue H (barva) je jediným nositelem barevné informace, udává barvu a je z rozsahu 0° - 360° , Saturation S (sytost) je z rozsahu 0% - 100% a udává sytost dané barvy, tedy jak hodně je daná barva „barevná“ (0% -bezbarvá 100% -syté barvy). Value V (jas), nabývá rozsahu 0% - 100% (0% -černá 100% -bílá).

Barevná hodnota je tedy dána úhlem H , který jako jediný nese informaci o samotné barvě pixelu. Z Obrázku 4.2 by mělo být patrné, že s klesající hodnotou V (jas) se barevný kužel zužuje a klesá tak rozdíl mezi jednotlivými barvami. Při nízkém osvětlení jsou tedy vzájemné rozdíly menší a hůře rozlišitelné, což odpovídá realitě – při nedostatečném osvětlení vidí hůře lidské oko i kamera.



Obrázek 4.2: Znázornění významu jednotlivých kanálů HSV (převzato z [8]).



Obrázek 4.3: Ukázka vztahu barevného prostoru RGB a HSV (převzato z [8]).

Převod z RGB do HSV

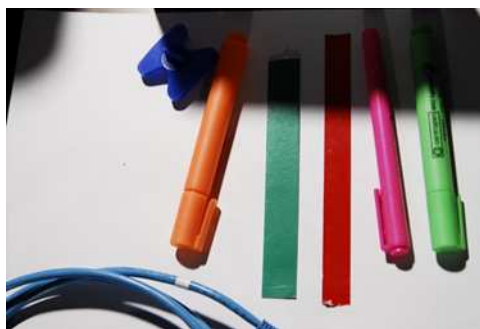
$$H = \begin{cases} \text{nedefinováno} & \text{pokud } C_{\max} = C_{\min} \\ 60 \cdot \frac{G - B}{C_{\max} - C_{\min}} & \text{pokud } C_{\max} = R \wedge G \geq B \\ 60 \cdot \frac{G - B}{C_{\max} - C_{\min}} + 360 & \text{pokud } C_{\max} = R \wedge G < B \\ 60 \cdot \frac{B - R}{C_{\max} - C_{\min}} + 120 & \text{pokud } C_{\max} = G \\ 60 \cdot \frac{R - G}{C_{\max} - C_{\min}} + 240 & \text{pokud } C_{\max} = B \end{cases} \quad (4.2)$$

$$S = \begin{cases} 0 & \text{pokud } C_{\max} = 0 \\ 1 - \frac{C_{\min}}{C_{\max}} & \text{jinak} \end{cases} \quad (4.3)$$

$$V = \{ C_{\max} \quad (4.4)$$

kde $C_{\min}$ a $C_{\max}$ představují minimální a maximální hodnotu ze složek R, G a B [8].

Převodem z barevného prostoru RGB do HSV vzniknou tři obrázky, které znázorňují tři složky prostoru HSV.



(a) Původní obrázek před transformací a rozebráním.



(b) Jasová složka, odpovídá černobílému obrázku (VALUE).



(c) Složka nesoucí informaci o sytosti barev (SATURATION).



(d) Složka nesoucí barevnou informaci (HUE).

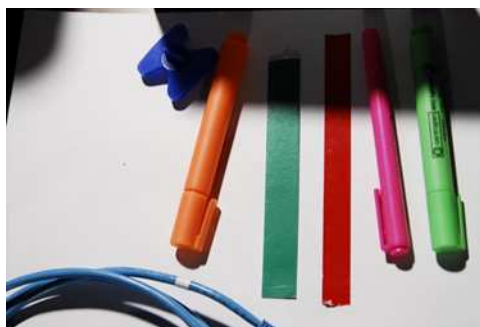
Obrázek 4.4: Jednotlivé složky barevného prostoru HSV.

Snímek VALUE na Obrázku 4.4(b) znázorňuje jasové poměry v obraze a představuje tak černobílý snímek originálu. Nás zajímal nejvíce snímek HUE, na kterém by neměly být vidět stíny za vozidly. Na Obrázku 4.3(b) je vidět, že stín se příliš v tomto kanále neprojevuje, je pouze v blízkosti samotných předmětů. Stíny se projevují jako rozšíření plochy kolem objektu, který je vrhá. Je to z toho důvodu, že barva stínu je dána odrazem (nebo průchodem) světla od objektu (skrz objekt).

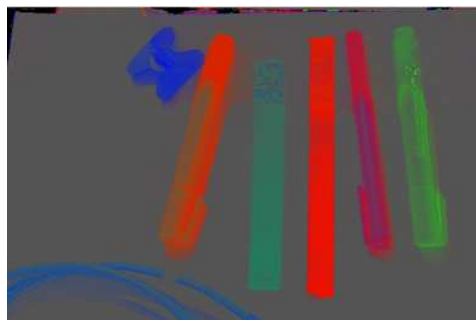
Formát normalizovaného RGB má za cíl linearizovat barevné podíly pro různé hodnoty jasu. Prostor RGB totiž není lineární, tedy změna jasu zapříčiní i změnu poměru mezi jednotlivými složkami R:G:B a tedy dojde i k barevnému posunu.

$$r = \frac{R}{R + G + B} \quad g = \frac{G}{R + G + B} \quad b = \frac{B}{R + G + B} \quad r + g + b = 1 \quad (4.5)$$

Na Obrázku 4.5. je jasně patrné zmizení stínu a zvýraznění barvy stínu u jednotlivých předmětů. Pokud by se od předmětů do zóny stínu neodráželo žádné světlo a pokud by



(a) Původní obrázek před transformací.



(b) Obrázek v normalizovaném rgb.

Obrázek 4.5: Transformace RGB na normalizované rgb.

předměty neprosvítaly, byly by převodem do normalizovaného RGB odstraněny vrhané stíny.

4.3 Stíny za vozidly

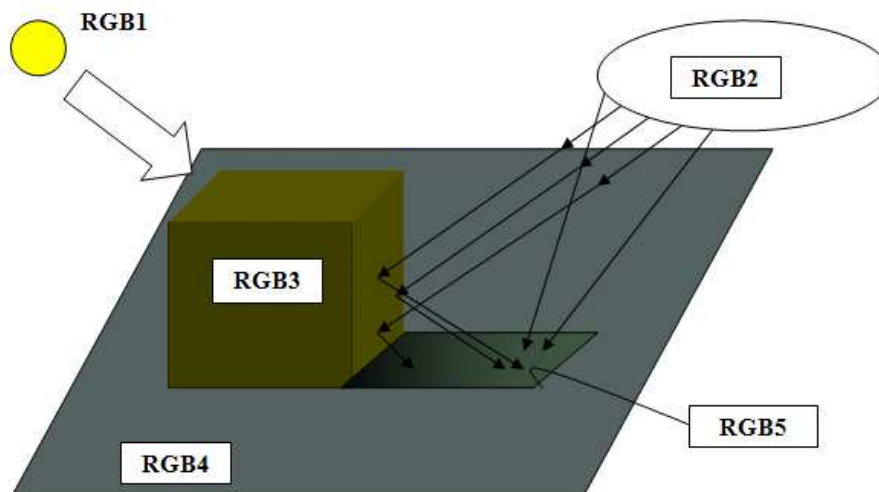
Stíny za vozidly představují jistý problém, který nám může zkomplikovat detekci pohybujících se vozů. Stín rozšiřuje detekovanou hranici a tím z malého auta dělá auto větší. V kritických okamžicích (kdy jiné vozidlo pojedje ve stínu prvního) může dojít ke splnutí několika pohybujících se vozidel dohromady.

Pokoušíme se tedy stíny alespoň částečně (v okrajových částech) odstranit. K tomu nám mohl pomoci prostý převod z prostoru RGB (který není lineární a při změně intenzity bílého světla dochází také k barevnému posunu) do normalizovaného RGB, který již lineární je, nebo do barevného prostoru HSV. Bohužel ani jedna z prostých metod nefunguje dokonale (či spíš vůbec) protože předpoklad, že stín má barvu vozovky (RGB4) a jen poklesne jasová složka, je mylná. Celou situaci se snaží vystihnout náčrt na Obrázku 4.6.

Barva stínu je ovlivněna mnoha faktory, jak znázorňuje Obrázek 4.6. Stín je ve většině případů posunutý směrem k modré. Je to způsobeno bočním osvitom ze zdroje RGB2, což je světlo lámané v atmosféře. Stín bude také nejtmavší pod vozidlem a v jeho bezprostřední blízkosti, zde nám tedy nepomůže žádný převod (černá je černá). Hlavní neznámá barva, která bude vždy jiná, je odraz světla od karosérie vozu. Účinek tohoto obarvení bude klesat se vzdáleností od objektu. Tedy dle všeho bychom měli být schopni když ne odstranit, tak alespoň zredukovat velikost stínu, a tím snížit riziko splnutí vozidel.

4.4 Odstranění není jednoduché

Problematika odstranění stínů je velice komplikovaná a neexistuje jednoznačné řešení této úlohy. Aplikace pro analýzu směrových vztahů na křižovatkách je vytvořena bez jakéhokoliv potlačování stínů. Stíny nejsou potlačovány z důvodů následného zpomalení



Obrázek 4.6: Barevné složky podílející se na tvorbě stínu za vozidlem: RGB1 - Hlavní zdroj světla (slunce), RGB2 - světlo rozptýlené v atmosféře s barevným posunem do modré + jiné zdroje osvitů, RGB3 - vlastní sledovaný objekt (vozidlo), jehož povrch odráží světlo zbarvené RGB3, RGB4 - pozadí (vozovka), RGB5 - stín za vozidlem, který má ovšem jiné zbarvení než RGB4.

celé aplikace, vlivem dalších složitých úprav obrazu. Vzhledem k metodě analýzy je možné virtuální smyčky rozmístit takovým způsobem, aby se vliv stínů minimalizoval.

Do budoucna a v případě dalšího vývoje této aplikace, bude ale třeba tuto problematiku vyřešit, neboť je jednou z nejzávažnějších.

5

Software

5.1 Vstupní a výstupní data

Software byl navržen pro potřeby automatické analýzy poměrů odbočení na křižovatce. Jako vstupní datový proud je použit videozáznam ze sledované křižovatky. Vzhledem ke schopnosti programu pracovat v reálném čase je možné po malých úpravách použít i přímý datový proud z kamery.

S rostoucím rozlišením analyzovaného videa klesá rychlost vyhodnocování, tedy tzv. FPS (počet snímků za sekundu, z anglického *Frames Per Second*). Používané a testované byly videozáznamy s rozlišením 720×576 a snímkovací frekvencí 25 snímků za sekundu.

Poloha, velikost a natočení jednotlivých smyček je uloženo v konfiguračním souboru s příponou `cfg`. Název tohoto souboru je odvozen od názvu analyzovaného videa. Pokud k videu neexistuje žádný soubor s příponou `cfg`, je automaticky po startu programu spuštěn jednoduchý editor.

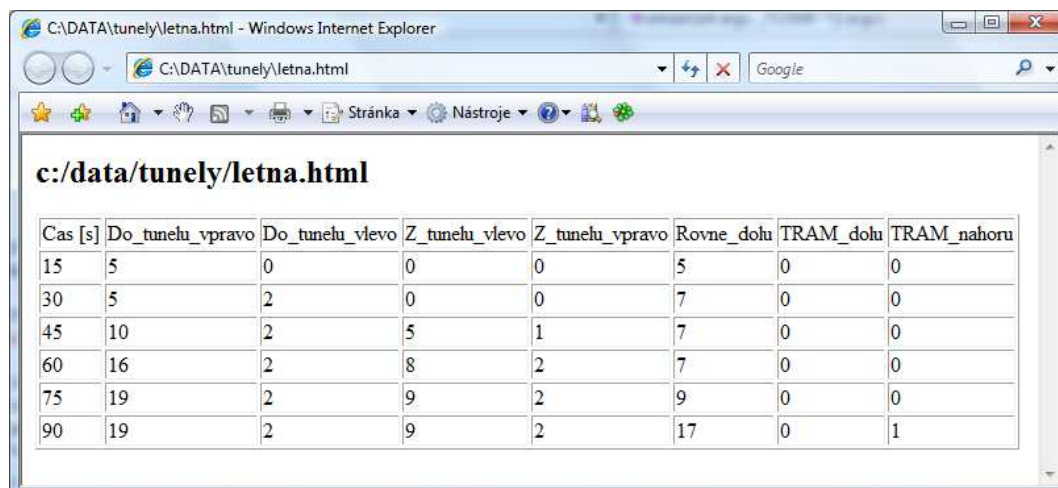
Výstupem celého programu je tabulka vytvořená v jazyce HTML a zobrazitelná je tak na kterémkoliv internetovém prohlížeči.

V tabulce jsou ve sloupcích označeny jednotlivé směry a každý řádek představuje jeden měřený interval. Interval v jakém se mají data do tabulky zapisovat se zadává jako parametr programu.

5.2 Vzhled a ovládání programu

5.2.1 Hlavní program

Program je nutné spustit se dvěma parametry. Prvním parametrem je název videa, které má být analyzováno. Název musí být zapsán bez přípony. Druhým parametrem je inter-



Cas [s]	Do_tunelu_vpravo	Do_tunelu_vlevo	Z_tunelu_vlevo	Z_tunelu_vpravo	Rovne_dolu	TRAM_dolu	TRAM_nahoru
15	5	0	0	0	5	0	0
30	5	2	0	0	7	0	0
45	10	2	5	1	7	0	0
60	16	2	8	2	7	0	0
75	19	2	9	2	9	0	0
90	19	2	9	2	17	0	1

Obrázek 5.1: Ukázka výstupního formátu, zde pro interval záznamu 15 sekund.

val ukládání naměřených dat do HTML souboru. Předpokládejme, že video soubory se nacházejí v adresáři `video`, chceme spustit video `letna.avi` a zapisovat měřené hodnoty požadujeme po 15s. Program tedy spustíme následovně:

„`krizovatka.exe video/letna 15`“. Bezprostředně po spuštění se otevřou uživateli dvě okna. V jednom je zobrazováno video a v druhém je zobrazován stav jednotlivých počítačů směrů. Po spuštění se také začne přehrávat a analyzovat video, ke kterému jsme předali cestu pomocí parametru z příkazové řádky. Pokud pro zadaný video soubor již existuje konfigurační soubor, spustí se modelování pozadí, které trvá několik desítek sekund. Proces modelování je signalizován textem v okně videa. Neexistuje-li konfigurační soubor nebo je-li stisknuta klávesa E, je spuštěn editor, ve kterém je možné rozvržení smyček vytvořit či upravit.

Po ukončení modelování pozadí se video vrátí na první snímek a čeká na spuštění od uživatele. Spuštění se provádí klávesou R a pozastavení klávesou P. Analýza probíhá až do skončení video souboru nebo do stisknutí klávesy K, která program ukončí.

5.2.2 Editor

Do režimu editoru se uživatel dostane stisknutím klávesy E v hlavním okně nebo automaticky, pokud neexistuje konfigurační soubor. V režimu editace je možné umístit a natočit virtuální smyčky nebo je jen poupravit. Scéna, do které se smyčky vkládají, je ta, která byla na obrazovce v hlavním programu při stisknutí klávesy E. Editor se ukončuje klávesou B, při jejím stisku se polohy všech smyček automaticky uloží a hlavní program bude pokračovat.

Smyčky se vkládají klávesou V (vložit). Po jejím stisku je nutné zadat označení – popis směru, který bude daná smyčka sledovat. Tento popis se zadává v DOS okně hlavního programu. ID je generováno automaticky a odpovídá pořadí vkládání smyček. Po zadání popisku následuje umístění smyčky na scénu pomocí myši. Držením levého tlačítka lze smyčkou pohybovat. Natáčení je možné pomocí klávesy R nebo pomocí posuvné lišty v

okně nastavení. Velikost smyček se nastavuje pouze pomocí posuvné lišty.

Výběr již vytvořené smyčky se provede kliknutím myši do prostoru mezi jednotlivými subsmyčkami. Smazání nevhodné smyčky se provede klávesou D.

Umístění smyček na sledovanou scénu je velmi důležité. Vhodným umístěním lze například potlačit překrývání vozidel jedoucích za sebou. Před samotným začátkem umísťování je třeba sledovat provoz a vypožorovat pokud možno co nejpresněji dráhy jednotlivých směrů. Důležité je také sledovat občasné abnormality v provozu, jako například jízda vozidla mezi dvěma jízdními pruhy, průjezd autobusu (popřípadě jeho stínu) detekční oblastí pro jiný směr. Pokud jsou takovéto abnormality vysledovány, je možné podle nich upravit polohy detekčních smyček a tím minimalizovat falešné detekce.

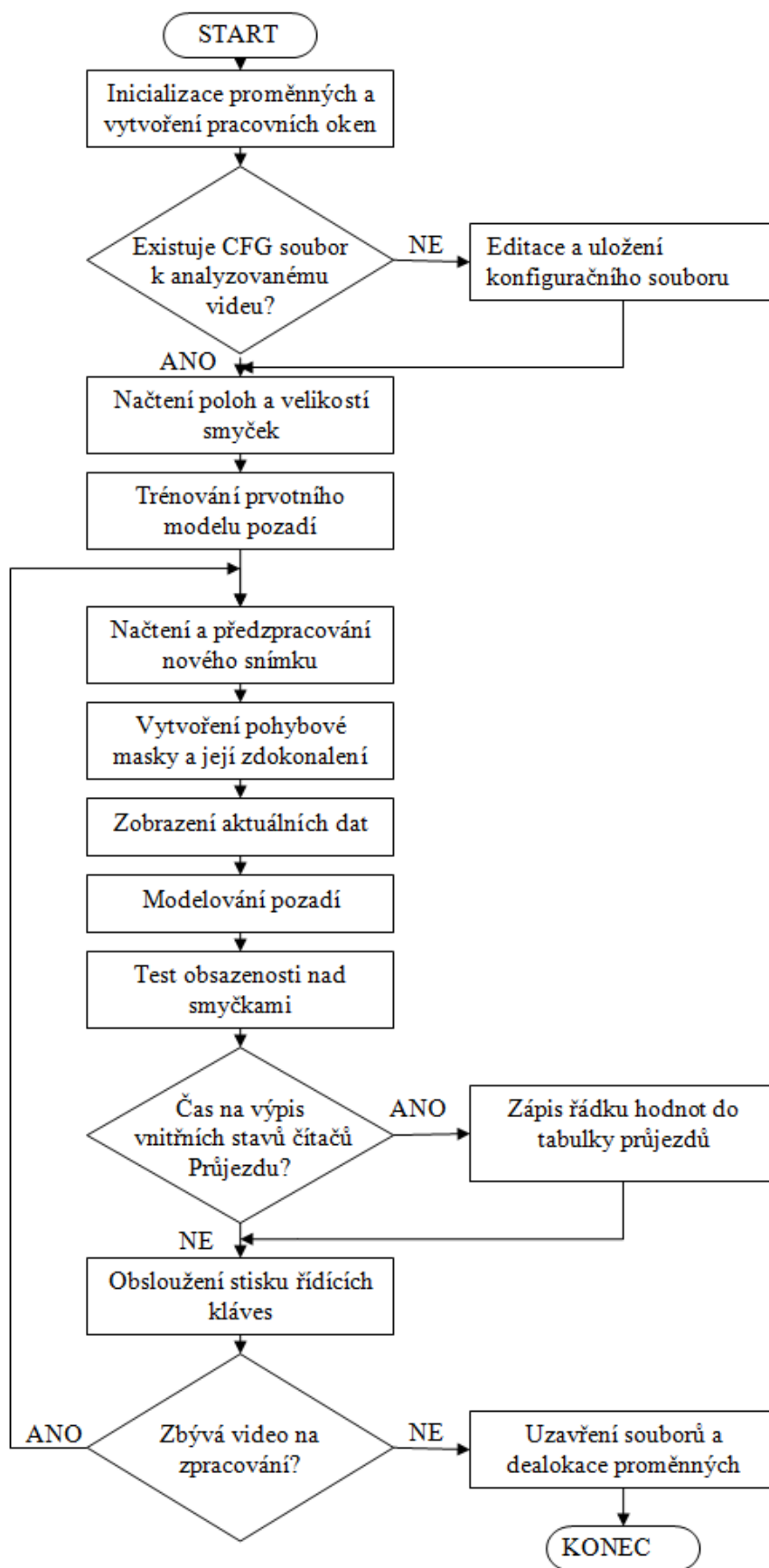
Rozměr smyček je žádoucí volit co možná nejmenší, neboť pohybová maska je poměrně kvalitní a „děr“ se v ní vyskytuje minimum. Větší velikost smyčky přináší větší objem zpracovávaných dat a tím i zpomalování programu. Hlavní důvod používání malých smyček jsou poměrně malé vzdálenosti mezi vozidly. Ty jsou způsobeny především úhlem, pod kterým kamera snímá scénu.

5.2.3 Popis chodu programu

Program byl vytvořen pro platformu Microsoft Windows za pomoci Microsoft Visual Studio 2005, pro svou činnost využívá knihoven Intel OpenCV [9], což jsou knihovny navržené speciálně pro práci s obrazem a videi.

Program jako takový je strukturován do jednotlivých funkcí a využívá objektového programování. Jako hlavní jednotkou je objekt SMYČKA, který má atributy jako jsou poloha, velikost, natočení, obsazení, počet průjezdů a jiné.

Vývojový diagram na Obrázku 5.2 popisuje nejdůležitější části celé aplikace. Prakticky celý zdrojový kód se skládá z cyklů v kterých jsou pro jednotlivé operace použity instrukce z knihovny OpenCV [9], které provádějí nejdůležitější a ucelené operace.



Obrázek 5.2: Vývojový diagram zachycující důležité procesy v průběhu programu.

6

Sběr dat a zhodnocení funkce schopnosti aplikace

6.0.4 Použité vybavení

Záznamové zařízení. Veškeré scény křižovatek byly zaznamenávány pomocí digitální videokamery SONY HandyCam DCR-PC330E. Snímkovácí rychlost 25 snímků za sekundu, expozice automatická i manuální.

Konfigurace testovací počítačové sestavy. Procesor Intel Core 2 duo 6600, Paměť 4GB RAM 1066MHz, Grafická karta GeForce 7900GTX, OS Microsoft Windows Vista 64bit

6.0.5 Lokalita

Záznamy byly prováděny na pražských křižovatkách, převážně v době začínající dopravní špičky. Výběr míst, vzhledem k použitému vybavení, byl značně omezen. K dispozici byla pouze kamera a klasický stativ, nebylo tak možné natáčet jakoukoliv křižovatku z důvodu špatného výhledu. Místa byla vybírána podle možnosti umístit kameru dostatečně vysoko a přitom blízko křižovatky. Vzhledem k letošní mírné zimě se nepodařilo udělat záběry za sněžení, ani se zasněženou krajinou.

Zaznamenané a analyzované křižovatky jsou:

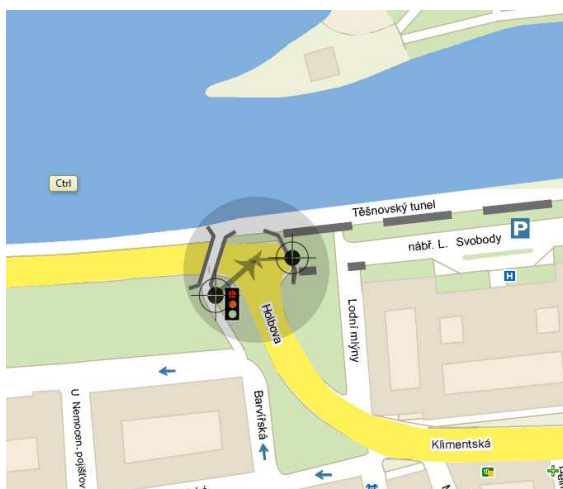
- Těšnovský tunel – Holbova
- Štefánikův most – nábr. Edvarda Beneše – Letenský tunel
- Čechův most – nábr. Edvarda Beneše

6.0.6 Účinnost automatického sčítání

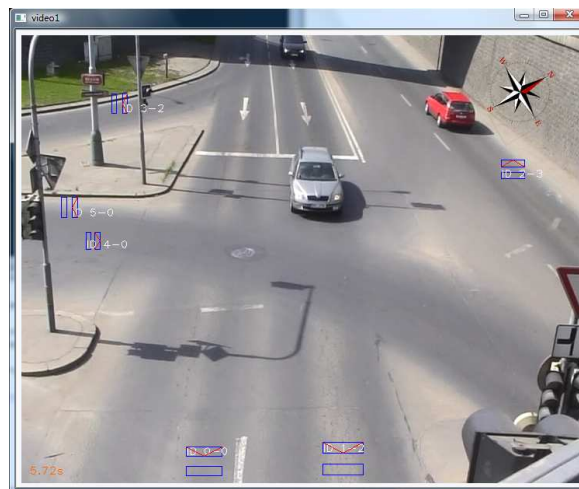
Kontrola účinnosti byla provedena porovnáním dat z programu a dat získaných ručním sčítáním jednotlivých směrů. Umístění detekčních virtuálních smyček velkou měrou ovlivňuje výsledky celkové účinnosti. Vhodné umístění je třeba vysledovat a je třeba mu věnovat značnou pozornost, neboť je to jediný faktor, který velice ovlivňuje celkové výsledky.

Testování bylo provedeno na již zmiňovaných křižovatkách a z celkového záznamu 30 minut na jednu křižovatku bylo k prověření spolehlivosti použito 15 minut, tedy 900s záznamu.

6.0.7 Test č. 1 – Těšnovský tunel



(a) Těšnovský tunel – Holbova.



(b) Rozložení smyček na křižovatce.

Obrázek 6.1: Umístění kamery a rozmístění detekčních smyček na křižovatce Těšnovský tunel – Holbova při testu číslo 1.

Směr	ID	Naměřené průjezdy	Skutečné průjezdy	Rozdíl	Účinnost
Do tunelu1	0	37	39	-2	94.8
Do tunelu2	1	103	104	-1	99
Z tunelu rovně	2	254	282	-28	90.1
Z tunelu vlevo	5	100	99	+1	99
Odbočení vpravo	3	118	118	0	100
Odbočení vlevo	4	36	38	-2	94.7
Σ		648	680	-32	95.3

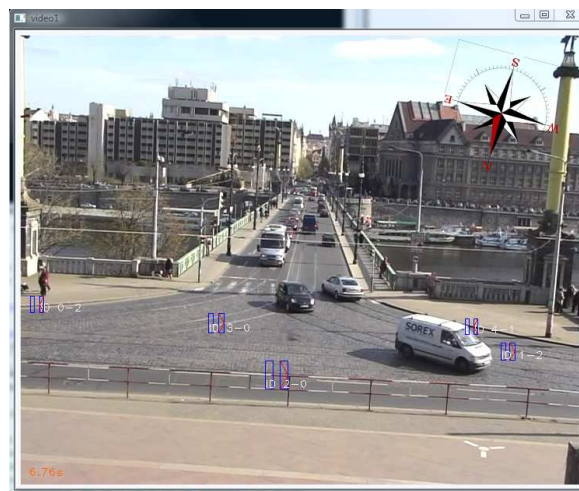
Tabulka 6.1: Naměřené a skutečné průjezdy v testu č. 1.

Z výsledků testu uvedených v Tabulce 6.1 vidíme, že nejvyšší chybovost vykazuje přímý směr z tunelu. Chyby vznikaly z důvodu předjíždění na křižovatce, kdy některá vozidla objížděla detekční smyčku ID2 zleva. Dalším faktorem je prakticky nejméně vhodný úhel nájezdu na detekční oblast, který činil přibližně 45° . Nastával tak častěji efekt, kdy vůz vlivem pomalého snímkování obsadil obě smyčky naráz a nebyl tak vyhodnocen průjezd.

6.0.8 Test č.2 – Čechův most



(a) Čechův most – nábr. Edvarda Beneše.



(b) Rozložení smyček na křižovatce.

Obrázek 6.2: Umístění kamery a rozmístění detekčních smyček na křižovatce Čechův most – nábr. Edvarda Beneše při testu číslo 2.

Směr	ID	Naměřené průjezdy	Skutečné průjezdy	Rozdíl	Účinnost
Z mostu doprava	0	82	87	-5	94.3
Z mostu doleva	1	172	182	-10	94.8
Hlavní rovně	2	161	167	-6	96.4
Hlavní vlevo na most	3	59	69	-10	85.5
Z hlavní na most vpravo	4	85	86	-1	98.8
Σ		559	591	-32	94.6

Tabulka 6.2: Naměřené a skutečné průjezdy v testu č. 2.

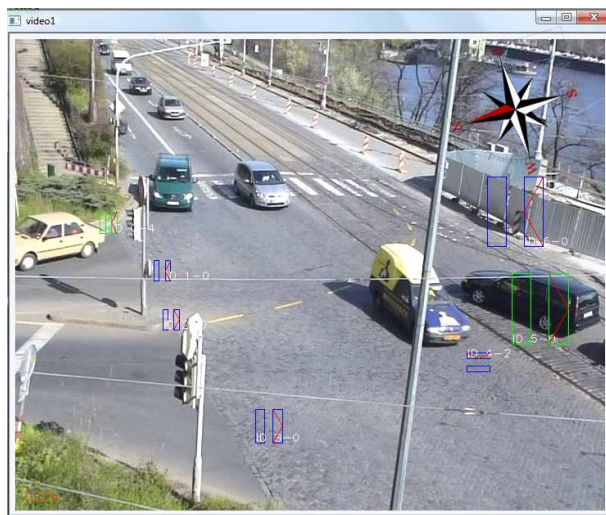
Jak je patrné z dat uvedených v Tabulce 6.2, test na Čechově mostě dopadl nejhůř. Nižší úspěšnost zde byla zapříčiněna především malým úhlem snímání a tím způsobeným překrýváním vozidel v různých trasách. Neukázněnost řidičů zde hrála také významnou roli a to především při odbočení z mostu vlevo, kdy se vlivem vznikající kolony řidiči snažili najet do křižovatky za každou cenu, někdy najížděli až daleko před detekční zónou ID3, čímž se dokonale vyhnuli měřicí smyčce ID1.

Na této situaci je patrné, že umístění kamery hraje významnou roli v úspěšnosti detekce. Jako odhadová metoda je ovšem i tato konfigurace dostačující.

6.0.9 Test č. 3 – Letenský tunel na nábreží



(a) Štefánikův most – nábrž.Edvarda Beneše – Letenský tunel.



(b) Rozložení smyček na křižovatce.

Obrázek 6.3: Umístění kamery a rozmístění detekčních smyček na křižovatce Štefánikův most – nábrž.Edvarda Beneše – Letenský tunel při testu číslo 3.

Nejvyšší chybovostí se ukazuje být směr rovně dolů po nábreží. Bylo to způsobeno především neukázněností řidičů, kteří předjížděli stojící vozidla v přímém směru a smyčku ID4 tak míjely zprava. Z důvodu častého podjíždění je smyčka ID4 umístěna téměř až na samém konci přímého směru.

Na této křižovatce se kromě závadných stínů za vozidla projevila další parazitní stav a to odrazy na lesklých dlažebních kostkách. Tyto odlesky měly za následek, stejně jako stíny, větší plochu pohybové masky, a tím častější falešné obsazení jiných smyček či splnutí více vozidel do jednoho.

Směr	ID	Naměřené průjezdy	Skutečné průjezdy	Rozdíl	Účinnost
Do tunelu vpravo	0	107	107	0	100
Do tunelu vlevo	1	21	21	0	100
Z tunelu vlevo	2	112	121	-9	92.6
Z tunelu vpravo	3	21	22	-1	95.5
Rovně dolů	4	174	186	-12	93.6
Σ		435	457	-22	95.2

Tabulka 6.3: Naměřené a skutečné průjezdy v testu č. 3.

Odrazy by bylo možné řešit kvalitnější optikou snímací kamery s použitím polarizačních filtrů.

Na scéně se vyskytují dvě velké detekční smyčky, které byly určeny pro záznam průjezdu tramvají. Zde se ukazuje, že pevný poměr stran jednotlivých subsmyček $a : b$ (viz obr. 2.3.) není vhodný pro každý tvar. Detektor tramvají vyžaduje tvar podstatně protáhlejší (delší a užší). Při tomto stávajícím tvaru docházelo k falešnému obsazení nákladními vozy či dodávkami (v případě malé smyčky), nebo k neobsazení tramvají (v případě větší smyčky).

6.0.10 Souhrnné zhodnocení

Test č.	Naměřené průjezdy	Skutečné průjezdy	Rozdíl	Účinnost
1	648	680	-32	95.3
2	559	591	-32	94.6
3	435	457	-22	95.2
Σ	1642	1728	-86	95.02

Tabulka 6.4: Souhrnné zhodnocení provedených testů.

Souhrnné zhodnocení provedených testů uvádí Tabulka 6.4. Po zhodnocení třech testů dostávám výsledek účinnosti automatického sledování poměrů odbočení 95,02%. Tento výsledek se mi zdá poměrně uspokojivý, vzhledem k ne příliš velké složitosti algoritmů.

7

Závěr

Tato bakalářská práce měla za cíl najít a aplikovat vhodnou metodu pro automatické sledování poměrů odbočení na křižovatkách. S ohledem na současné trendy a výkon výpočetní techniky, byla použita metoda videodetekce s použitím virtuálních smyček. Virtuální smyčky jsou rozmístěny na vozovce v párech pro možnost určení směru jízdy. Tyto dvojsmyčky leží v drahách jízd jednotlivých směrů, a tedy umožňují sčítání různých kombinací odbočení.

Pro samotnou detekci pohybu, což je stěžejní část této práce, je využita metoda rozdílu snímané scény od referenčního snímku pozadí. Pozadí je modelováno za pomoci pomalého selektivního zápisu konstantní hodnoty do modelu pozadí, tím se model pozadí udržuje neustále aktuální a jsou tak odstraněny pomalé změny, jako jsou pohybující se stíny vlivem rotace Země.

Výstupní hodnoty, naměřené pomocí této aplikace jsou předávány uživateli formou HTML tabulky, ve které jsou uloženy počty jednotlivých průjezdů ve sledovaných směrech v předem definovaných intervalech.

Celá aplikace splnila mé očekávání. Neobsahuje žádné extrémně složité algoritmy, a tak není příliš náročná na hardware, a přesto dosahuje poměrně značné spolehlivosti. Vzhledem k výsledkům testů je vidět, že nejčastější chybou bylo nezapočítání průjezdu, oproti tomu falešné započítání bylo poměrně vzácné. Pozměněním vyhodnocovacího sekvenčního automatu (viz. kapitola 2.2.2) tak, aby detekoval i za sebou blízko najíždějící vozidla, by se pravděpodobně dosáhlo lepších výsledků. Také úhel natočení smyčky by přinesl další malé zlepšení co se spolehlivosti detekce týče.

Hlavním nedostatkem aplikace je neschopnost odstranit stíny a odlesky od vozidel. Vyřešení tohoto problému by mělo za následek radikální zvýšení účinnosti, neboť nejčastěji byla chyba detekce způsobena projetím stínu vozidla cizí detekční smyčkou, popřípadě splynutí více vozidel do jednoho celku.

Posledním zdrojem chyb je, že občas řidiči nedodržují jízdu ve vymezeném jízdním pruhu, a tak projíždějí mimo detekční smyčku, popřípadě projíždějí detekční smyčku cizí. Tento zdroj chyb je představenou metodou neodstranitelný.

Přes všechny zmiňované nedostatky se aplikace projevila jako uspokojivá. Průměrná účinnost automatického sčítání je dle výsledků třech testů 95%.

Možné vylepšení do budoucna je vytvoření mobilní měřicí jednotky, kterou bude možno umístit v blízkosti křižovatek a pomocí výsuvného ramena sledovat scénu křižovatky z velkého úhlu pohledu. Dokonalejší metoda směrových vztahů by byla možná pomocí sledování vozidla projíždějícího křižovatkou po celou dobu jízdy. Touto metodou by bylo možné sledovat i okružní křižovatky, které jsou metodou virtuálních smyček nesledovatelné.

Literatura

- [1] JIRAVA, Petr – SLABÝ, Petr. *Dopravní inženýrství*. Ediční středisko ČVUT, Praha, 1990.
- [2] SLABÝ, Petr – UHLÍK, Petr. *Dopravní inženýrství I*. Česká technika – nakladatelství ČVUT, Praha, 2006.
- [3] PŘIBYL, Pavel. *Inteligentní dopravní systémy a dopravní telematika*. Vydavatelství ČVUT, Praha, 2005.
- [4] HLAVÁČ, Václav – ŠONKA, Milan. *Počítačové vidění*. Grada, Praha, 1992
- [5] PŘIBYL, Pavel – MACH, Radim. *Řídicí systémy silniční dopravy*. Vydavatelství ČVUT, Praha, 2003.
- [6] Obecné informace na serveru WIKIPEDIA. <<http://cs.wikipedia.org/>>.
- [7] HLAVÁČ, Václav – KYBIC, Jan. *Předzpracování v prostoru obrazů* [online]. [cit. 10. srpna 2007]. Dostupné na World Wide Web: <<http://cmp.felk.cvut.cz/cmp/courses/33DZ0zima2005/slidy/lokalniPredzpracovani.pdf>>.
- [8] *Wikipedia, barevné prostory* [online]. Wikipedia, 2000 [cit. 10. srpna 2007]. Dostupné na World Wide Web: <http://en.wikipedia.org/wiki/HSV_color_space>.
- [9] *Intel OpenCV* [online]. Intel Corp., 2000 [cit. 10. srpna 2007]. Dostupné na World Wide Web: <<http://www.intel.com/technology/computing/opencv/>>.
- [10] *Background subtraction for detection of moving objects* [online]. [cit. 10. srpna 2007]. Dostupné na World Wide Web: <<http://www.llnl.gov/CASC/sapphire/background/background.html>>.
- [11] CHEUNG, S.-C. – KAMATH, C. *Robust techniques for background subtraction in urban traffic video* [online]. SPIE Electronic Imaging, San Jose, January 2004, [cit. 10. srpna 2007]. Dostupné na World Wide Web: <<http://www.llnl.gov/CASC/sapphire/pubs/UCRL-CONF-200706.pdf>>.



Obsah přiloženého DVD

Adresáře:

- videa
- ukázky
- bakalářská práce

Obsah kořenového adresáře:

- `křižovatka.exe` – vlastní aplikace
- `*.dll` – knihovny nezbytné pro vlastní chod programu

Obsah adresáře videa:

- `*.avi` – záznamy testovacích videí
- `*.cfg` – soubory s uloženými konfiguracemi pro video soubory

Obsah adresáře ukázky:

- `prujezd pres smycku.avi` – Ukázka chodu programu bez potřeby jeho zpuštění
- `tvorba pozadí.avi` – Ukázka modelování pozadí.

Obsah adresáře bakalářská práce:

- bakalářská práce ve formátu pdf a tex

Upozornění: Program zapisuje výstupní soubor do stejného adresáře, na kterém se nachází video soubory. Při spuštění přímo z DVD nebude tento zápis možný a program se ukončí.