

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
FAKULTA DOPRAVNÍ, ÚSTAV APLIKOVANÉ MATEMATIKY



Karel Fatura

Odhad délky fronty stojících vozidel

Bakalářská práce, Srpen 2007

Čestné prohlášení

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

Nemám závažný důvod proti použití tohoto 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).

Abstrakt

Klíčová slova: Videodetekce, detekce pohybu, řízení dopravního proudu, kolona vozidel, odhad délky kolony.

Tato práce si klade za cíl shrnout používané metody odhadu délky fronty stojících vozidel. Dále za pomoci videodetekce a algoritmů pro zpracování obrazu vytvořit program, který by délku kolony odhadoval pomocí přehledových kamer v reálném čase.

Obsah

1	Motivace	6
1.1	Význam pro řízení dopravního uzlu (křižovatky)	7
1.2	Negativní vlivy kolon	9
1.3	Metody odhadu a jejich vlastnosti	9
1.3.1	Lidský potenciál	9
1.3.2	Indukční smyčky	10
1.3.3	Videodetekce	11
1.4	Navrhované řešení	11
2	Detekce pohybu	13
2.1	Snímek s pozadím	15
2.2	Modus, průměr	15
2.3	Gausovský model pozadí	15
2.4	Optický tok	17
2.5	Vážený součet, dlouhá expozice	18
3	Aplikace Odhad délky kolony	19
3.1	Vstupy a výstupy programu	19
3.1.1	Formát XML	20
3.2	Vlastnosti modelu	20
3.3	Třídy aplikace	21
3.4	Popis funkce programu	22
3.5	Přesnost odhadu	23
4	Sběr dat	25
4.1	Požadavky na scénu	25
4.2	Výběr lokality	26
4.3	Okolní vlivy, expozice	28
5	Experimenty	30
5.1	Odstranění stínů	30
5.1.1	Princip	30
5.1.2	Postup	31
5.1.3	Nastavení mezí intervalů	32
5.1.4	Výpočet mezí	32
5.1.5	Hodnocení experimentu	33
5.2	Silné a slabé stránky aplikace pro odhad délky kolon	34
5.2.1	Metodika testování	34

5.2.2	Výsledky	35
5.2.3	Hodnocení experimentu	37
6	Závěr	38
6.1	Odstranění slabin	38
6.2	Další vývoj	39
A	Návod k obsluze programu pro odhad délky kolon	41
A.1	Požadavky pro běh programu	41
A.2	Spuštění programu a ovládání	41
A.2.1	Spuštění programu z příkazového řádku	41
A.2.2	Nastavení parametrů zástupci (jiná možnost spuštění)	43
A.2.3	Hlavní menu a ovládací klávesy	44
A.3	Konfigurace	44
A.3.1	Kolonová smyčka	45
A.3.2	Měřicí smyčka	48
A.4	Zpracování výsledků	49
B	CD příloha – aplikace pro odhad délky kolon	50

Seznam obrázků

1.1	Datové toky v aplikaci	7
1.2	Možnosti rozmístění prvků pro odhad délky kolny	10
1.3	Kamera pro použití v dopravě instalovaná na stožáru	12
2.1	Aktuální snímek, pozadí, binární maska a výsledný snímek	14
2.2	Světle zvýrazněný výřez bude v příštím snímku nahrazen tmavým.	14
2.3	Výsledky při různých metodách tvorby pozadí	16
2.4	Směs Gaussových křivek	16
3.1	Výstupní XML soubor vystavený na webu.	21
3.2	Znázornění trigonometrie pro výpočet délky kolonové smyčky	22
3.3	Červeně zvýrazněné délky smyček nejsou započítány	24
4.1	Stíny nebezpečné pro odhad kolon v pruhu neobsazeném kolonovou smyčkou jsou zvýrazněné.	26
4.2	Možnosti umístění kamery.	27
4.3	Automatická expozice způsobuje potíže při příjezdu světlého vozu.	28
5.1	Krychle znázorňující barevný prostor RGB.	31
5.2	Zašpiněný bok vozidla se jeví jako stín.	31
5.3	Tmavá část stínu je díky nízkému jasů chybně barevně interpretována.	32
5.4	Barva světelného zdroje.	33
5.5	Automatické nastavení intervalů nepodává přesvědčivý výsledek	34
5.6	Rozvržení detekčních prvků při testech.	37
6.1	Výsledné okno aplikace.	39
A.1	Spuštění příkazového řádku.	42
A.2	Zadání příkazu s argumenty.	42
A.3	Okno aplikace.	42
A.4	Vytvoření zástupce.	43
A.5	Vlastnosti zástupce programu.	43
A.6	Seznam ovládacích kláves.	44
A.7	Zobrazení informační řádek.	45
A.8	Konfigurační okno.	45
A.9	První bočnice.	46
A.10	Druhá bočnice.	46
A.11	Potvrzení konfigurace smyčky.	47
A.12	Pomocné konfigurační okno.	47

A.13 Zadání parametrů do okna konzole.	48
A.14 Konstrukce virtuální smyčky.	48
A.15 Zadání parametrů měřicí smyčky do konzole.	48

Seznam tabulek

5.1	Výsledek pozorování pixelů.	33
5.2	Délky vozidel pro odhad délky kolony.	34
5.3	Výsledky prvního testu	35
5.4	Výsledky druhého testu	36
5.5	Výsledky třetího testu	36

1

Motivace, používané metody pro odhad délky kolon

Délkou kolony rozumíme vzdálenost mezi předním nárazníkem prvního a zadním nárazníkem posledního vozidla stojícího v řadě. Kolony zpravidla vznikají v místech řízení dopravního proudu zastavováním, mohou se však vyskytovat i jinde (kongesce). Délka kolon je kvalitativní ukazatel. Naší snahou je celkovou délku kolon minimalizovat a dosáhnout tak maximální možné kvality dopravního procesu.

V dopravním inženýrství má odhadnutá délka kolon veliký potenciál pro:

- **Řízení provozu na křižovatce** Pro efektivní přepínání jednotlivých fází, nastavení doby rozsvícení signálu volno při dynamickém řízení.
- **Zastavení v zakázané oblasti** Pro upozornění na vozidla zastavující např. v tunelu, příp. na jiném místě, kde toto není dovoleno.
- **Identifikaci saturovaných oblastí** V oblasti, kde se tvoří kolony ve všech směrech zřejmě dochází k její přesycení. Tato informace je důležitá pro řízení dopravního proudu informováním.

Přestože je tento hodnotný dopravní ukazatel vhodný pro řadu dopravních aplikací, není využíván. Je to způsobeno jeho obtížným odhadováním pomocí současných detektorů, instalovaných na dopravní síti. V drtivé většině jsou to indukční smyčky, o jejichž vlastnostech bude pojednáno dále v textu. Tato práce si klade za cíl popsat a realizovat jednoduchou a univerzální aplikaci, která by délku kolon stojících vozidel dokázala odhadovat v reálném čase pomocí dopravních přehledových kamer a algoritmů zpracování obrazu.



Obrázek 1.1: Datové toky v aplikaci

1.1 Význam pro řízení dopravního uzlu (křižovatky)

Velké uplatnění by ukazatel délka kolony našel při řízení světelných křižovatek. Díky vlastnostem ukazatele by za jistých podmínek mohlo dojít ke zkvalitnění řízení oproti stávajícím metodám. Světelná křižovatka je řízena z řadiče, který přepíná jednotlivé signály na návěstidlech (semaforech) [9]. Při realizaci *statického modelu řízení* je program řadiče přepínán na základě historického dopravního modelu pouze v závislosti na čase [?]. Tento model se používá již od 50. let 20. století a je velmi dobře zpracován. Jeho podstatnou nevýhodou je skutečnost, že řízení je dáno neměnným modelem a na momentální či trvalé změny v dopravě nelze reagovat jinak než změnou celých programů řadiče.

Od poloviny 80. let jsou používány *dynamické systémy*, které počítají parametry signálního plánu (délku cyklu, dobu fází zelených nebo celkovou skladbu všech fází) v závislosti na měřených dopravních datech. V případě dynamického řízení křižovatky jsou dopravní řadiče vybaveny detektory pro monitorování přítomnosti vozidel nebo chodců, přičemž akčními členy jsou světelná návěstidla, která předávají informace řidičům a chodcům. Řadiče mohou pracovat izolovaně bez vazeb na jiné řadiče či ústřednu, mohou být uspořádány v koordinované linii nebo být řízeny dopravní ústřednou.

Na základě detektorů (nejčastěji indukčních smyček) sleduje dopravní řadič provoz a poté mu přizpůsobuje délku signálu volno (při *dopravně závislém řízení*). Sledování probíhá monitorováním prodlužovacího detektoru, který bývá umístěn ve vzdálenosti 30 až 50 metrů před stop čarou. Zde je měřen odstup mezi vozidly. V případě, že je menší než stanovená hodnota (obvykle 3 až 5 sekund [?]), je prodlužována doba fáze zelené až do předem stanoveného maxima. To je nutné vymežit, aby nedošlo k dlouhodobému zastavení kolizního směru. K takovému řízení lze využít i parametr délka kolony. Má oproti intenzitě (počet vozidel, která daným úsekem projedou za určitý časový interval), tu zpravidla měří indukční smyčka, jisté výhody.

- *Délka signálu volno je známá již před jeho rozsvícením.* Již při přepínání fází je znám přibližný počet čekajících vozidel, z něž lze stanovit dobu potřebnou pro jejich projetí křižovatkou.

- *Lze identifikovat nestandardní situaci.* Nehoda v jízdním pruhu trvale zablokuje indukční smyčku. Při měření délky kolony zůstane část měřicí oblasti nezasažená nehodou i nadále funkční.
- *Lze snadno získat informaci o stupni nasycení křižovatky.* Dlouhodobé kolony ve všech směrech signalizují stav nasycení, na který může navázat dispečerské rozhodnutí na oblastní úrovni. Střední kolony zasahující za prodlužovací smyčku se od stavu úplného přesycení (dlouhodobé kolony ve všech směrech) pomocí smyček těžko rozeznávají.

Zmíněné výhody mohou napomoci ke kvalitnějšímu řízení provozu nejen z pohledu první dopravní úrovně (křižovatka) ale také z oblastní (druhé) úrovně. Zde jsou jednotlivé uzly propojeny s ústřednou, která na úrovni oblasti koordinuje řídí jejich činnost. Např. zjistí-li software v oblastní ústředně, že několik sousedících křižovatek uvnitř sledovaného prostoru je přesycených, může vést dopravní proud směřující přes kritickou oblast po objízdě trase. S podobnými aplikacemi se již na mnoha místech setkáme avšak jsou zpravidla realizovány jinak než na základě sledování parametru délka kolony. Tato realizace bývá často složitější a nákladnější nejen při pořizování ale i při údržbě. Pro oblastní řízení se používají různé metody [?].

Časově závislé řízení (off-line): dopravní stavy oblasti jsou získány statistickou analýzou historických hodnot dopravních dat (intenzity dopravy) získaných v rozhodujících místech dopravní sítě. Na jejich základě jsou vypočítány sestavy signálních plánů dopravních řadičů. Ty jsou pak implementovány do řadičů v závislosti na denním čase nebo na dni v roce. Při výpočtech vycházejících z historických dat se optimalizuje délka fází zelených, doby cyklu a ofset (časový posuv zelené fáze jednoho směru na následující křižovatce).

Dopravně závislé řízení (on-line): pro různé stavy dopravní sítě jsou předem vypočítány sestavy signálních plánů (souborů programů s charakteristickými konstantami řízení), které jsou uloženy buď v řadičích nebo v dopravní ústředně. Jedná se tedy o řízení v režimu on-line bez optimalizace. Zároveň jsou v oblasti vybrány strategické detektory a jsou sestaveny logické podmínky, které popisují různé kombinace stavů nad všemi nebo vybranými detektory. V závislosti na momentální dopravní situaci je prostřednictvím vhodné podmínky vybraný program, který nejlépe vyhovuje dané situaci.

Dopravně závislé řízení (on-line) s optimalizací, centralizované systémy: inteligence systému je soustředěna do dopravní ústředny, kam jsou přenášeny dopravní parametry a v sekundovém rastru jsou prováděny optimalizační výpočty, jejichž výsledkem jsou hodnoty délky zelených, doby cyklu a ofsetu pro všechny uzly. Jednotlivá světelná návěstidla jsou tedy přímo řízena dopravní ústřednou.

Adaptivní metody řízení: vycházejí z výše uvedených metod a vylepšují je tím, že řadiči ponechávají značnou autonomnost (lokální inteligenci), neboť řadič pracuje v sekundovém rastru v režimu dopravně závislého řízení. Přitom však dopravní ústředna optimalizuje v delším časovém rastru (přibližně 15 až 30 min.), jejich parametry, obvykle maximální délku fáze zelených, dobu cyklu, ofset, a to pro řadiče v celé oblasti. Tím se trvale přizpůsobují parametry řadičů dopravním poměrům v oblasti (síti komunikací).

Expertní a heuristické řízení: v některých případech jsou tak komplikované poměry v dopravní síti, že je nutné používat expertní řízení. To pracuje s databází předem definovaných a stále modifikovaných podmínek rozhodování, že v jistých situacích simuluje chování experta.

1.2 Negativní vlivy kolon

Při zastavování vozidla v koloně dochází k značným energetickým ztrátám. přebytečná kinetická energie je pomocí třecích brzd přeměněna na odpadní teplo, aby mohla být v zápětí (přepnutí na signál volno) znovu získána z chemické energie vázané v palivu. Tato skutečnost má neblahý vliv na životní prostředí, kdy je do ovzduší vypouštěno větší množství spalin, než je nezbytně nutné. Dále s sebou nese negativní ekonomické důsledky v podobě zvýšených nákladů na provoz vozidel. Do těch spadá nejen větší množství spotřebovaného paliva, ale také vyšší opotřebení brzdové soustavy a pneumatik. Důsledkem je i sekundární působení, projevující se v počtu nehod a zvýšeném stresu řidičů i cestujících.

Popsané negativní vlivy se projevují u všech vozidel. Nutno podotknout, že u hybridních vozidel s mnohem menší intenzitou. Hybridní vozidla dovedou přebytečnou energii při brždění využívat pro dobíjení akumulátoru. I zde však dochází ke ztrátám, kdy energie získaná snížením rychlosti z v_2 na v_1 je menší než energie dodaná při akceleraci z v_1 na v_2 . Dále v praxi nelze rekuperovat energii při nízkých rychlostech, kdy brzdový výkon klesne pod požadovanou mez. A v neposlední řadě platí, že do plně nabitě baterie již nelze uložit žádnou další energii. Připustíme-li, že cena hybridních vozidel v budoucnu poklesne na hodnotu, která bude konkurence schopná cenám vozidel se spalovacími motory, je zřejmé, že ani tehdy (kdy bude dopravní proud složen výhradně z hybridních vozidel) nebude problém zastavování v před křižovatkou vyřešen.

1.3 Metody odhadu a jejich vlastnosti

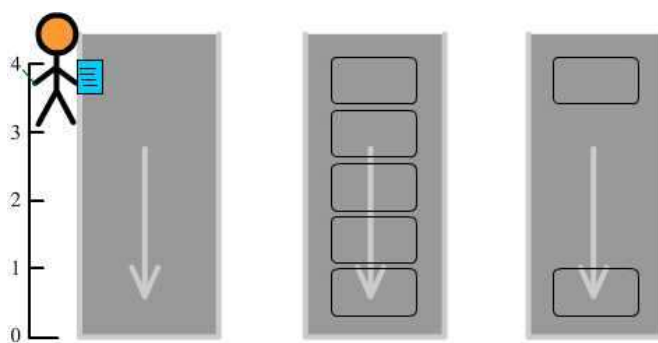
Pro měření délky kolon se používají různé metody. nejdůležitější z nich jsou popsány v následujících odstavcích.

1.3.1 Lidský potenciál

Vedle jízdního pruhu je pomocí měřicího pásma vyznačena vzdálenost od stopčáry. Pozorovatel zaujme vhodnou pozici a vždy po rozsvícení signálu volno pro daný jízdní pruh zapisuje pozorovanou hodnotu délky kolony. Tento postup je použitelný při dopravním průzkumu, nikoli však pro navazující dopravní aplikace (dynamické řízení atp.). Ty potřebují automatizované měření, na jehož výstupu je strojově snadno zpracovatelná hodnota bezprostředně po provedení měření. V principu k tomu lze využít jakýkoli bodový detektor (pneumatický, piezo-detektor, indukční smyčku) resp. soustavu detektorů, rozmístěných v definovaných vzdálenostech od stopčáry. Na takovéto měření je možné navázat dopravní aplikaci, avšak pro vysokou nákladnost se toto neprovádí.

1.3.2 Indukční smyčky

Pomocí indukčních smyček lze odhadovat délku kolony. Jednou z možností je realizace již zmíněné soustavy smyček v definovaných vzdálenostech před stopčárou. Vzhledem k pracnosti a finanční náročnosti (při pořizování i údržbě) se s tímto řešením nesetkáme. Jiným, které lze na silnicích pozorovat, avšak není obvyklé, je několik metrů dlouhá smyčka přes celou šířku jízdního pruhu. Ta se, jak je postupně zaplňována vozidly, projevuje různým rozladěním vstupního harmonického signálu. Z rozladění je možné délku kolony s poměrně velkou přesností odhadnout. Další možností je soustava dvou smyček. Indukční smyčky mají pro rozmístění před křižovatkou jistá pravidla [9]. Rozmísťují se za konkrétním účelem (např. prodlužovací se umísťuje cca 50 m před stopčárou a slouží k prodloužení signálu volno při dynamickém řízení). Pozorováním těchto smyček lze za jistých okolností usoudit, zda se před křižovatkou vytvořila kolona či nikoli. Tento odhad má však zpravidla binární charakter (je kolona, není kolona) a je použitelný pouze jako orientační informace. Např. upozornění pro dispečera, zda daná oblast není zahlcena. Ten poté na základě pozorování situaci vyhodnotí a podnikne příslušná opatření.



Obrázek 1.2: Možnosti rozmístění prvků pro odhad délky kolny

Pro odhad délky kolon pomocí dvou smyček lze použít i sofistikovanější metody [?]. Ve výstupu projektu Galileo v roce 2003 byl proveden úvod do problematiky identifikace délky kolon. V principu jsou používány dvě metody. Do první skupiny patří model vyhodnocující délku kolony na základě informací o obsazenosti detektoru a do druhé skupiny model využívající analogie hydrodynamiky — hydrodynamický model dopravy.

První metoda využívá faktu, že obsazenost detektoru je vyšší pro delší fronty vozidel než pro krátké, kdy se konec kolony pohybuje v blízkosti detektoru. Hustota vozidel na konci kolony je nižší než v jejích ostatních částech. Uprostřed je kolona homogenní, zatímco několik metrů před koncem hustota vozidel kolísá. Z větší obsazenosti detektoru lze také usuzovat na snížení rychlosti nad detektorem. Metoda je omezená rozsahem odhadu délky kolony. Minimální měřitelná délka kolony bývá asi 20 až 40 m, maximální odhadnutelná délka kolony je přibližně o 50 až 80 m větší, než je umístění detektoru před stop čarou. Poté se již výstupní hodnoty dostávají do saturace a pohybují se v oblasti maximální odhadnutelné délky kolony, v tomto případě tedy asi 200 m.

Druhá metoda je založena na analogii fyzikálního zákona o zachování hmoty. V dopravní oblasti jej lze zjednodušeně formulovat pravidlem: „Co na komunikaci vjede, to z ní buď odjede, nebo na ní zůstává ve formě kolony.“ Použití této metody je zvláště výhodné na dálničních komunikacích, kde se nenachází žádné neměřené přístupy na hlavní tah mezi jednotlivými detekčními prvky. V případě městské sítě mohou vznikat problémy s vedlejšími vjezdy a výjezdy. Možné nepřesnosti je nutné odstraňovat kalibrací pomocí plovoucích vozidel.

V rámci projektu *Optimalizace propustnosti městské dopravní sítě využívající satelitní systém Galileo* byl vyvinut hybridní model kombinující oba výše uvedené modely. Pro dosažení vyšší přesnosti se jedná navíc o parametrický model, neboť jeho parametry jsou dynamicky kalibrovány pravidlově orientovaným expertním systémem. Tento model již byl testován v Praze 5 na vybraném úseku komunikace Zborovská-Svornosti pro reálné dopravní podmínky. Shoda mezi reálnými a odhadovanými dobami jízdy je velmi dobrá [?].

Obecně platí, že instalace indukčních smyček je nákladná a obtěžující vzhledem k dopravnímu provozu (vlastnost většiny intruzivních detektorů), kdy je nutné na několik hodin uzavřít jízdní pruh. To stejné platí i o opravách. Poznamenejme, že indukční smyčky umístěné v místech, kde vozidla zastavují, se často přetrhají. Umísťují se totiž do obrusné vrstvy, která se při brzdění těžkých vozidel smýká podobně jako špatně přilepený koberec. Toto mechanické namáhání značně zkracuje jejich životnost. Zmíněná nevýhoda ale není omezující vzhledem k tomu, že velká část křižovatek je smyčkami již osazena.

1.3.3 Videodetekce

Videodetekce se v posledních letech velmi rozšiřuje. Je to způsobeno klesající cenou elektroniky (nízká pořizovací cena kamery) a také rostoucím výkonem výpočetní techniky. To dovoluje provádět i složitější operace nad obrázkem s dostatečným rozlišením v reálném čase. Obecně lze říci, že videodetekci lze, při vhodném umístění kamery, získat jakýkoli ze základních dopravních ukazatelů. Lidský mozek přijímá asi 80% všech informací právě pomocí zraku, což nasvědčuje tomu, že videodetekce má veliký potenciál při vhodném následném zpracování signálu z kamery.

Videodetekce má všechny výhody neintruzivních detektorů. K tomu nízká pořizovací cena a provozní náklady. Dále jistá variabilita, plynoucí z možnosti zpracovávat signál kamery libovolným způsobem. Detektor tedy může sloužit k několika účelům současně (měření několika ukazatelů). Nutno podotknout, že videodetekce má i jisté nevýhody jako reakce na stín vozidla. Neúplná informace při zákrytu vozidel atp. Tyto nevýhody lze ve většině případů kompenzovat jednak umístěním kamery jednak předzpracováním obrazu, příp. použitím několika kamer.

1.4 Navrhované řešení

Předmětem zkoumání této práce je metoda odhadu délky kolony pomocí videodetekce. Tu jsem si vybral kvůli nízkým nárokům na provoz a instalaci detektorů (kamery). V metodě používám segmentů, jejichž vlastnosti jsou velice podobné vlastnostem in-



Obrázek 1.3: Kamera pro použití v dopravě instalovaná na stožáru

dukčních smyček. Takové uspořádání jsem zvolil jednak z důvodu dobrého zpracování problematiky všeobecně rozšířených indukčních smyček, jednak kvůli názornosti při obsluze programu a jednoduchosti řešení.

2

Detekce pohybu

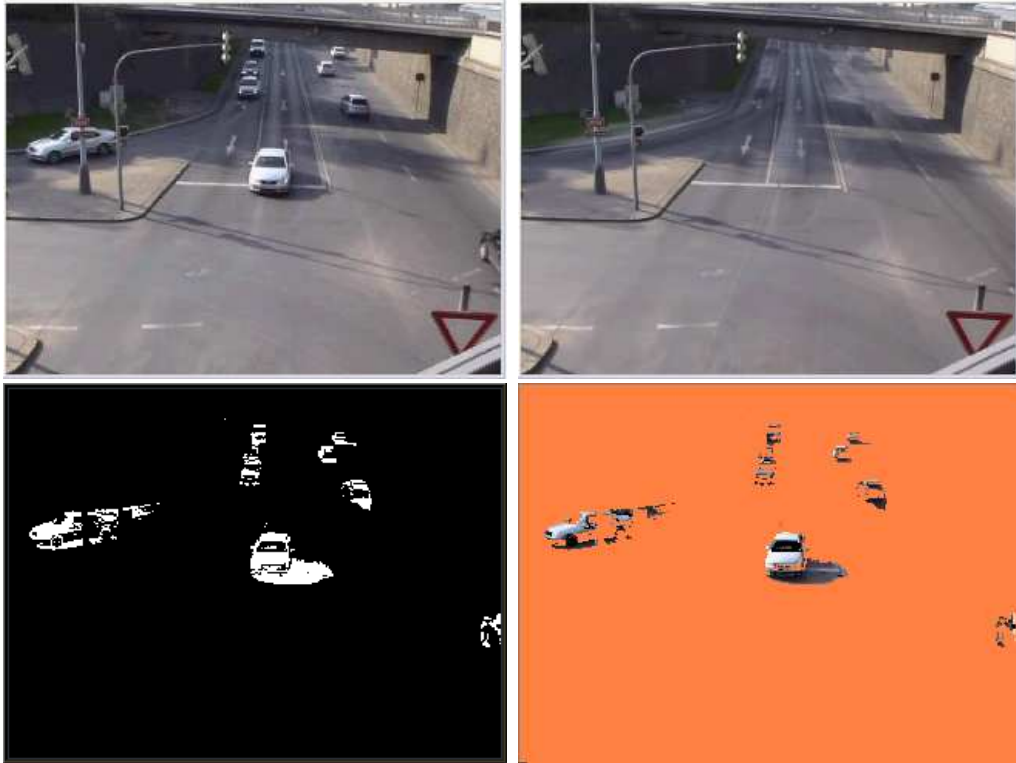
Detekcí pohybu rozumíme postup, kterým se ze sekvence následných snímků (z videa), identifikují pohybující se části na scéně. Tyto části nazýváme *popředím*, ostatní plochu snímku označujeme jako *pozadí*. Výsledkem detekce je binární snímek označovaný jako maska (angl. *mask*). Binární znamená že jeho pixely mohou nabývat pouze dvou hodnot (ano a ne, resp. + a -). Po vytvoření lze masku znovu aplikovat na aktuální snímek (poslední z videosekvence) takovým způsobem, že pozitivní pixely masky zkopírujeme, negativní ponecháme rovny nule. Takovému postupu se obecně říká *maskování* (odtud název obrázku) a jeho výsledkem je snímek pouze s pochybujícími se objekty. Masku lze po jejím vytvoření zpracovat různými způsoby. Obecně však všechny činnosti navazující na detekci pohybu pracují s maskou kvůli její jednoduchosti (nízké nároky na paměť, rychlé procházení po jednotlivých pixelech), plynoucí s použité datové struktury.

Aplikace pro odhad délky kolon pracuje se standardními nástroji pro detekci pohybu, což je rozdíl dvou po sobě následujících snímků a jeho různé variace s podpůrnými algoritmy pro speciální účely (např. odstranění stínů)

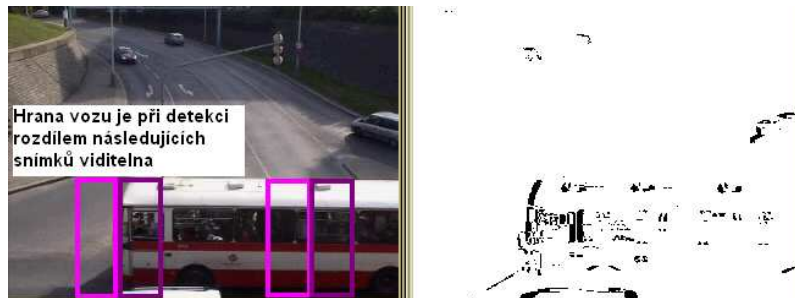
$$|\vec{A}_{xy}(t) - \vec{A}_{xy}(t+1)| = R_{xy}(t+1) \quad (2.1)$$

kde $\vec{A}_{xy}(t)$ značí pixel aktuálního snímku o souřadnicích x, y a složkách RGB v čase t a $R_{xy}(t)$ je rozdílový pixel.

Asi největším nešvarem při takovéto detekci pozadí, který se intenzivně projevuje právě při dopravních aplikacích, je detekce obdelníkových struktur, pohybujících se v ose své strany. Např. návěs nákladního automobilu se mezi dvěma snímky posune, avšak průmět jeho středu na čip kamery zůstane nezměněn. Detekovatelné jsou potom pouze boční hrany návěsu (viz obrázek 2.2). Nejsnáze tento problém vyřešíme zavedením snímku s pozadím scény, tedy takovým, kde nebudou zachyceny žádné objekty popředí. Rozdílem aktuálního a snímku z pozadím dostaneme právě objekty popředí.



Obrázek 2.1: Aktuální snímek, pozadí, binární maska a výsledný snímek



Obrázek 2.2: Světle zvýrazněný výřez bude v příštím snímku nahrazen tmavým.

Rozdílový snímek je dále oprahován a převeden na binární podobu podle vztahu:

$$if \quad (|\vec{A}_{xy}(t) - \vec{P}_{xy}(t)| \geq \vec{P}h) \quad \vec{M}_{xy}(t) = 1 \quad else \quad \vec{M}_{xy}(t) = 0 \quad (2.2)$$

kde $\vec{A}_{xy}(t)$ je pixel snímku s pozadím, $\vec{P}h$ je prahová hodnota a $\vec{M}_{xy}(t)$ je pixel masky. Jako prahová hodnota je obvykle použit vektor o stejné velikých složkách ale v principu mohou být jednotlivé složky rozdílné, čímž docílíme jiné citlivosti pro každý barevný kanál.

Výsledkem je maska pohybu. Nad ní se odehrává další činnost aplikace. Tato maska má proto klíčový význam ve výsledné kvalitě a přesnosti odhadu.

2.1 Snímek s pozadím

Na otázku jak vytvořit snímek s pozadím není jednoduchá odpověď. Nejjednodušší by bylo snímek vytvořit (např. retuší v grafickém editoru) někým, kdo dokáže scénu sémanticky chápat a rozlišit tak popředí od pozadí, ten následně poslat aplikaci jako argument při startu. Toto řešení není realizovatelné v automatických systémech. K jistým změnám totiž dochází i na snímku s pozadím, např. intenzita osvětlení nebo postup stínů. To by se po jisté době začalo jevit jako pohyb v popředí. Dále by toto kolidovalo s požadavkem na univerzálnost aplikace a stěžovalo by to její spuštění (kolize s požadavkem na jednoduchost). Tím se úloha zkomplikovala na rozpoznání pomalých (pozadí) a rychlých (popředí) pohybů. Protože kvalita snímku s pozadím přímo masku o jejíž důležitosti bylo již dříve pojednáno. Byla tvorbě modelu pozadí věnována velká pozornost. Použité metody budou předloženy v následujících odstavcích.

2.2 Modus, průměr

Jednou z možností jak pozadí modelovat je využít k tomu historii snímků a z ní vyvozovat závěry na základě statistických metod. Aby měly vypovídající hodnotu je třeba použít velký počet vzorků. V programovacím jazyku C++ lze toto realizovat pomocí kruhové fronty, tu však musíme vždy po příchodu nového snímku celou projít, což negativně ovlivní rychlost aplikace. V praxi byla realizována fronta o 100 snímcích, přičemž nový vzorek byl zapisován každý 10. snímek. S jistým zjednodušením lze říci, že fronta uchovávala historii 1000 snímků.

Modus byl vybírán jako hodnota, jejíž četnost výskytu je stejná nebo větší než četnost výskytu kterékoli jiné hodnoty v souboru [3]. Výsledek byl uspokojivý svou kvalitou. Nevyhovoval však pomalou dobou reakce na změny v pozadí např. při odjezdu kolony vozidel.

Průměr počítaný pro každý pixel P_i přes celou kruhovou frontu jako:

$$\bar{x} = \frac{1}{n} \sum_{i=0}^n P_i \quad (2.3)$$

vykazoval podobné výsledky. Sporný pixel (ten, jehož četnost ve frontě překročila 50%) se však měnil plynule, což kopírovalo jeho postupnou dominanci ve frontě. Oběma metodám je nutno vytknout nedostatečnou rychlost rehabilitace při náhlé změně, dále nefunkčnost při použití v místech, kde objekty popředí zaujímají nadpoloviční dobu oproti objektům pozadí.

2.3 Gausovský model pozadí

Hodnota pixelu byla doposud popisována jako trojice skalárů (každému barevnému kanálu odpovídal jeden skalár). Využijeme-li k popisu střední hodnotu a rozptyl, získáme kromě informace o nejpravděpodobnějším kandidátovi také informaci o šířce intervalu,

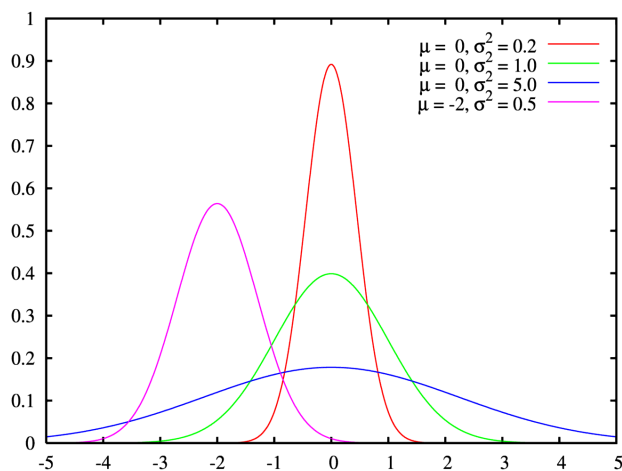


Obrázek 2.3: Výsledky při různých metodách tvorby pozadí

ve kterém se pravděpodobné hodnoty nacházejí. Čím bude interval užší, tím je jistější, že příslušná střední hodnota patří na dané místo (pixel). Grafické znázornění odpovídá Gaussově křivce.

Gausovský model, anglicky Gaussian Mixture Model (GMM), je technika modelování pozadí, kdy pro každý pixel existuje popis jeho hodnoty pomocí několika Gaussových křivek. Nejčastěji se používá 3 až 5 křivek [4]. Ty popisují s jakou pravděpodobností, se na daném místě (pixelu) vyskytne právě jistá hodnota, viz obrázek 2.4.

Tento model je implementován v knihovně openCV, která byla použita při tvorbě ap-



Obrázek 2.4: Směs Gaussových křivek

likace pro odhad délky kolon. Výsledek detekce pohybu byl vynikající, časová náročnost výpočtu střední. V praxi se však ukázalo, že dění za stop-čarou (vozidla zastavující na signál stůj) model nedokáže věrně zachytit. V těchto místech docházelo k detekci falešného popředí po odjezdu vozidel. Nastavením parametrů se nepodařilo tuto nevýhodu odstranit.

2.4 Optický tok

V sekvenci snímků, které zachycují pohybující se objekty, nebo pohyb samotné kamery, dochází ke skutečnosti, že obrazy těchto objektů se ve výsledném videu přesouvají z jednoho místa na jiné. Takovýto přesun nazýváme optickým tokem. V oboru počítačového vidění lze jeho pomocí získat užitečné informace [10] jako:

- Kolik pohybujících se objektů je na scéně
- Jakým směrem se pohybují
- Zda se pohybují rotačně nebo translačně
- Jak rychle se pohybují

K rovnici optického toku se můžeme dopracovat takto [10]:

Barevná intenzita pixelu v daném $I(x, y, t)$ je funkcí rychlostí x a y a času t . Intenzitu pixelu za malý okamžik v malé vzdálenosti od původního určíme jako

$$I(x + dx, y + dy, t + dt) = I(x, y, t) + \frac{\partial I}{\partial x}dx + \frac{\partial I}{\partial y}dy + \frac{\partial I}{\partial t}dt \quad (2.4)$$

nyní předpokládejme, že v místě (x, y) a čase t je pixel objektu, který se za dt posune do pozice (x, y) , dále předpokládejme, že intenzita pixelu, který zobrazuje pixel nyní je přesně stejná jako ta v místě o okamžik dříve. Tedy

$$I(x + dx, y + dy, t + dt) = I(x, y, t) \quad (2.5)$$

pro parciální derivace tedy platí

$$\frac{\partial I}{\partial x}dx + \frac{\partial I}{\partial y}dy + \frac{\partial I}{\partial t}dt = 0 \quad (2.6)$$

označíme-li rychlosti $\frac{\partial I}{\partial x} = u$ a $\frac{\partial I}{\partial y} = v$ dostaneme po formální úpravě 2.5 rovnici optického toku:

$$-\frac{\partial I}{\partial t} = \frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v \quad (2.7)$$

Tuto rovnici je nutné řešit pro každý pixel. V praxi se takto náročná úloha zjednodušuje tak, že výpočet je prováděn ve dvou krocích. V prvním se definují shluky pixelů (cluster). Ty tvoří pixely, které jsou si nějakým způsobem podobné (mají podobný jas, barvu a jsou v bezprostřední blízkosti). K tomu je nutná jejich identifikace na aktuálním a předchozím snímku. Dalším krokem je výpočet posunutí clusterů vzhledem k předchozímu

snímku. Po zjištění posunů jsou sestaveny matice vektorů rychlosti pro směr x a pro směr y . I přesto, že výpočet probíhá pouze pro několik shluků, je časově velice náročný.

To, že optický tok podává informaci o rychlosti pohybujících se objektů, by pro rozpoznání stavu křižovatky (kongesce, volno atp.) mohlo být velmi užitečné. Praktická zkušenost však ukázala, že vzhledem k veliké výpočetní náročnosti není tento model pro potřeby aplikace pro odhad délky kolon vhodný. Dále bylo nutné nastavit jako parametr přibližnou velikost hledaných shluků. Aby byla aplikace univerzální, nelze předem tento parametr zadat a samotná tato skutečnost vylučuje použití optického toku pro účely aplikace.

2.5 Vážený součet, dlouhá expozice

Dlouhou expozici si lze v diskretním světě představit jako sumu určitého počtu snímků, z nichž každý má jistou váhu. Rovnice pro každý pixel má tvar:

$$\sum_{n=1}^u \vec{A}_{xy}(n) \cdot \frac{1}{u} \quad (2.8)$$

kde u je celkový počet snímků v sumě a n je index konkrétního snímku. Ve spojitém světě by se rovnice transformovala na odpovídající integrál. Vážený součet je obdobou dlouhé expozice, zobrazené na nekonečný počet příchozích snímků. Sumace zde probíhá vždy jen dva snímky (pozadí a aktuální), z nichž každý má jinou váhu. Např.

$$\vec{V}_{xy}(t) = \alpha \cdot \vec{P}_{xy}(t) + (1 - \alpha) \cdot \vec{A}_{xy}(t) \quad (2.9)$$

kde $\vec{V}_{xy}(t)$ je pixel výsledku, $\vec{A}_{xy}(t)$ pixel aktuálního snímku, $\vec{P}_{xy}(t)$ pixel pozadí a α je koeficient paměti pozadí.

Tyto metody mají velikou výhodu v rychlosti jednak výpočetní jednak rehabilitace po zápisu objektu popředí do pozadí, dále vykazují konvergentní charakter. Vysoká rychlost rehabilitace však znamená i vysokou rychlost zápisu cizího objektu do pozadí. Velká změna (např. příjezd bílého vozidla na černou vozovku) probíhá po větších krocích než změna malá. To plyne z rovnice 2.9 a znalosti reprezentace obrázku ve světě výpočetní techniky (černá barva odpovídá hodnotě 0, bílá maximum, tj. 2^8 , 2^{16} atp.). V dopravních aplikacích znamená velká jasová změna zpravidla nežádoucí jev a je proto vhodné nezapisovat ji do snímku s pozadím. Nebo aspoň nezapisovat ji příliš rychle. Pro dosažení konstantní rychlosti změny snímku s pozadím byla rovnice 2.9 upravena na:

$$V_{xy}(t) = V_{xy}(t) + \gamma \quad (2.10)$$

Pro případ, že rozdíl hodnot příslušných barevných kanálů pixelů aktuálního a pozadí je větší než definovaná mez a

$$V_{xy}(t) = V_{xy}(t) - \gamma \quad (2.11)$$

pro případ, že rozdíl je menší. γ je konstanta, ovlivňující rychlost zápisu do snímku s pozadím. Tento způsob tvorby snímku s pozadím se jeví jako optimální, aby byl použit pro potřeby aplikace pro odhad délky kolon. Jistý potenciál skýtá práce v jiných barevných prostorech než RGB, avšak naše zkušenost (viz kapitola 5.1) s použitou kamerou nedovolila plně využít jejich vlastnosti.

3

Aplikace Odhad délky kolony

Požadavky kladené na program:

- Jednoduché *user-friendly* ovládání, snadná instalace a konfigurace
- Univerzálnost – aplikace funguje kdekoli a kdykoli (za splnění podmínek provozu kamery)
- Snadné další zpracování výstupu
- Schopnost pracovat v režimu *real-time*
- Dostatečná (resp. maximální možná) přesnost odhadu

3.1 Vstupy a výstupy programu

Jako vstup je aplikaci přiváděn aktuální snímek z videa. Díky dynamickému řešení uvnitř aplikace může být velikost vstupujícího snímku libovolná. Platí, že při práci s větším snímek (větší rozlišení) lze dosáhnout přesnějších výsledků avšak na úkor výpočetní náročnosti, potažmo rychlosti aplikace. Vstupní snímek má tři barevné kanály.

Vzhledem k tomu, že aplikace pracuje on-line, je vhodné, kromě maximální, průměrné a okamžité délky kolony uchovávat i historii těchto hodnot pro případnou hlubší analýzu. Pro uchování historie využívá program zápis do souboru. Jeho formát byl zvolen na základě následujících požadavků na výstup:

- Srozumitelnost, přehlednost
- Použitelnost při dalším zpracování
- Nezávislost na platformě
- Okamžitá použitelnost

Zmíněné požadavky definují úlohu o společném rozhraní [6] rozšířenou o skutečnost, že není znám způsob, jakým bude výstup aplikace zpracován resp. existuje několik různých možností pozdějšího zpracování a cílem je vyhovět všem. Proto je třeba výstupní formát dat volit pokud možno co nejvíce univerzální, avšak současně neztratit informaci o logických souvislostech mezi naměřenými daty. Rozeberme tedy možnosti, které se nabízejí.

Při použití **vlastního formátu** specifického pro danou aplikaci nelze splnit požadavek na snadné následné zpracování výstupu. Zpracovatel výstupního souboru by potřeboval speciální software pro jeho prohlížení, případně transformaci na požadovaný formát. Další možností je použití již **existujícího formátu**, připraveného pro další zpracování např. *.xls. Takové řešení vyhoví většině kladených požadavků, včetně nezávislosti na platformě. Formát je ale pro použití v aplikaci pro odhad délky kolon (uchování několika časových vektorů) příliš komplexní. Dále je těžko uchopitelný (není open-source), tedy v podstatě nelze získat jeho kompletní popis. To platí obecně pro většinu komerčních formátů. Jako optimální řešení se proto jeví **formát *.xml**.

3.1.1 Formát XML

XML (eXtensible Markup Language) je standard XML vytvořený konsorciem W3, aby poskytl standardní formát pro zpracování dat a splňoval následující kritéria (mimo jiné) [1]:

- je „human readable“ - je ho tedy možno číst a upravovat v libovolném textovém editoru
- je rozšiřitelný - umožňuje vytvářet vlastní formáty
- má jednoduchou syntaxi
- implementace parserů a dalších nástrojů na jeho zpracování je snadná (alespoň ve srovnání se SGML)

Syntaxe je velice podobná formátu HTML. XML dokument se skládá z pojmenovaných elementů [2]. Každý element přitom může obsahovat text a další vnořené elementy. XML dokument si tak lze představit jako hierarchickou stromovou strukturu. Do ní lze uložit v podstatě libovolné informace. Každý element může mít ještě několik atributů, což jsou uspořádané dvojice obsahující jméno atributu a jeho hodnotu. Je navržen dostatečně univerzálně na to aby jím bylo možno efektivně popsat jakkoli složitá (jednoduchá) data. Jeho možnosti rozšíření (např. implementace souboru *.css s kaskádovými styly) zajistí snadnou a okamžitou (bezprostředně po ukončení měření) prezentaci např. na internetové stránce (obrázek 3.1). Obecně je podporován komerčním softwarem, proto jej lze přímo a bez dalších komplikací zpracovávat.

3.2 Vlastnosti modelu

Modelem rozumíme reprezentaci určitého objektu nebo systému, pojatou z určitého úhlu pohledu. Je sestaven na základě nashromážděných doposud známých informací a měl by ověřit správnost doposud známých faktů, provádět předpovědi, umožnit verifikaci

Cas [ms]	Delka [m] levy_pruh	Delka [m] pravy_pruh	Pocet [vozidel] odboceni do tunelu	Pocet [vozidel]
63000	0.0	0.0	33	4
63800	0.0	2.9	34	4
64600	0.0	6.7	34	4
65400	0.0	6.7	34	4
66200	0.0	6.7	35	4

Obrázek 3.1: Výstupní XML soubor vystavený na webu.

předpovědí [5]. Může se více či méně podobat své předloze.

V modelu, se kterým pracuje aplikace, jsou tři objekty. Ty s dostatečnou přesností zachycují hlavní rysy modelované skutečnosti a současně zajišťují jednoduchost modelu, čímž nedochází ke zpomalování běhu aplikace. Jako hlavní objekty byly zvoleny *TVirtualniSmycka*, *TKolonovaSmycka* a *TScena*. Vnitřní proměnné a funkce byly navrženy s respektováním hlavních principů objektově orientovaného programování. Každý objekt uchovává pro něj charakteristické informace. Dále je nositelem charakteristického chování, které programátor vyvolá přístupem k příslušné funkci.

3.3 Třídy aplikace

TVirtualniSmycka Virtuální smyčka je schopná počítat projíždějící vozidla a tuto hodnotu na základě dotazu sdělit příp. vynulovat. Dále dokáže určit zda je obsazena či nikoli. Kromě těchto základních vlastností ještě umožňuje vykreslení do obrázku, který je poslán jako parametr. Vnitřní proměnná uchovává délku smyčky v metrech vypočtenou na základě trigonometrie při její konstrukci.

TKolonovaSmycka Kolonová smyčka je složena z několika (zadáno jako parametr uživatelem při konstrukci) virtuálních smyček. Ze stavu jejich obsazenosti vypočítá délku kolony. Při tom nevyužívá pouze prostou kumulaci dílčích délek obsazených virtuálních smyček, ale používá i vyšší pravidla vycházející ze znalostí situace (vstup programátora) viz kapitola 3.4. Jako příklad uveďme započítání vynechané (neobsazené) smyčky mezi dvěma obsazenými, k tomu může dojít nevydařenou detekcí vozidla, jehož jas i barva jsou pro detekční modul programu „neviditelné“, tedy pod prahovou hodnotou.

TScena Objekt scény má dvě základní funkce. Jednak se stará o rozmístění detekčních prvků, k čemuž slouží konfigurační soubor zadaný uživatelem při spuštění aplikace. Lze uchovávat několik konfiguračních souborů pro specifické situace (ranní špička, noční provoz apod.) mezi nimiž uživatel jednoduše vybere při spouštění bez nutnosti rekonfigurace.

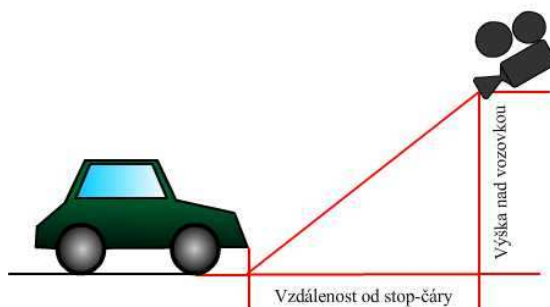
Druhou vlastností je modul detekce pohybu. Ten generuje snímek s pozadím, přičemž pracuje pouze s těmi částmi obrázku, kde jsou rozmístěny detekční prvky.

Každý z nich má přitom zajištěn specifický přístup. Tedy pozadí odpovídající virtuální smyčce je obnovováno jiným způsobem (rozdílná hodnota parametrů) než pozadí pod kolonovou smyčkou. Také uzamykání kolonových smyček probíhá pro každou nezávisle viz kapitola 3.4.

3.4 Popis funkce programu

Jak již bylo uvedeno, program pracuje s konfiguračními soubory. Ty nesou informaci o rozmístění a vlastnostech smyček na scéně. V případě absence se program po spuštění přepne do módu konfigurace. Zde uživatel rozmístí smyčky, přičemž je tázán na doplňující informace potřebné k výpočtu délek jednotlivých fragmentů kolonové smyčky. Výpočtem se zjistí přímá vzdálenost kamery od stopčáry a dále na základě znalosti lineární perspektivy délka sledovaného úseku. Ta je rozpočítána na jednotlivé fragmenty, které si ji pamatují jako vnitřní parametr. Tento výpočet proběhne pouze při konfiguraci, nebrzdí tedy program při jeho běhu.

O update pozadí se stará třída *TScena*. Probíhá dle pravidel (2.10) a (2.11). Pro zvýšení



Obrázek 3.2: Znázornění trigonometrie pro výpočet délky kolonové smyčky

výpočetního výkonu se pozadí modeluje pouze v místech obsazených smyčkou. Jelikož před stop-čarou dochází k jevům, které pro běžnou detekci pohybu nejsou standardní, pracuje update ve třech režimech (rychlý, pomalý a zamčené pozadí). Režim se nastavuje nezávisle pro každou kolonovou smyčku dle pravidel sestavených na základě experimentu. Z pozadí a aktuálního snímku je vytvořena maska. Pro omezení šumu byl vytvořen snímek *cistaMaska*, který pracuje na principu filtrování snímku *maska* tak, že pixely, které jsou bílé kratší než definovanou dobu nejsou použity (ponechají se černé). Snímek *cistaMaska* je dále vyhodnocen cíleným dotazováním jednotlivých fragmentů kolonových smyček na jejich obsazenost. Délky obsazených fragmentů se sumují, čímž se získá výsledná délka kolony. Při sumaci byla zavedena následující pravidla jako reakce na praktické zkušenosti.

Obsaditelnost od jedné třetiny Ne všechna vozidla zastaví přesně na stopčáře. Smyčka má proto vlastnost, že může být obsazena až od fragmentu, jehož pořadové číslo odpovídá celkovému počtu fragmentů děleného třemi.

Vynechání neobsazeného fragmentu Čelní okna vozidel jsou díky odleskům a jasů, který se často příliš neliší od jasu vozovky, poněkud obtížněji detekovatelná než karoserie vozidel. Může dojít k situaci, že jeden z fragmentů kolonové smyčky se jeví jako neobsazený, protože rozdíl jasu čelního skla a vozovky je pod prahovou hodnotou. Podobný jev nastane v případě, že mezi vozidly vznikne mezera (nezařadí v bezprostřední blízkosti za sebou). Pro tyto situace je program vybaven schopností potlačit neaktivitu fragmentu (resp. považovat jej za obsazený).

Po vyhodnocení a vykreslení všech rozmístěných detekčních prvků je v hlavním okně zobrazen výsledný snímek a následuje znovu celý cyklus až dokud nepřichází nové snímky, nebo do ukončení programu. V průběhu odhadu je možné vyvolat menu (stiskem klávesy M). Více o ovládání programu nalezne čtenář v příloze A.

3.5 Přesnost odhadu

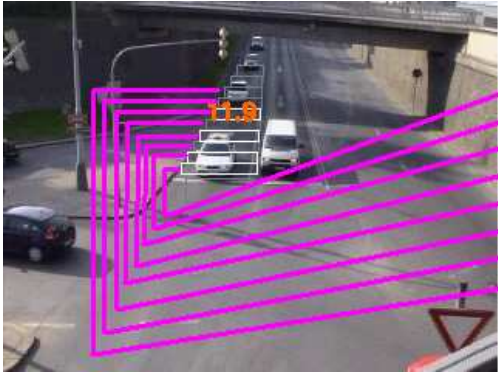
Přesností budeme v následujícím textu označovat takovou vlastnost programu, že zobrazovaná hodnota odhadu je totožná se skutečnou délkou kolony v daném okamžiku. Přesnost je ovlivněna několika faktory.

- Skladba dopravního proudu
- Povětrnostní podmínky
- Přesnost konfiguračních vstupů

Praktická zkušenost ukazuje, že při odhadu nastává chyba menší než 7 m (viz kapitola 5.2). To odpovídá asi jednomu osobnímu vozidlu. Tento výsledek je dostačující pro většinu aplikací, k nimž byl program navržen. Asi nejchoulostivější z nich je dynamické řízení křižovatky. I zde však stačí určit několik škál obsazenosti (např. prázdná, nízké, střední, vysoké obsazení a plná), k čemuž lze program bez potíží využít.

Vyšší přesnosti odhadu lze dosáhnout zkvalitněním konfiguračních vstupů např. tak, že kameru umístěnou nad scénou vybavíme laserovým dálkoměrem, který při konfiguraci dodá kvalitativně hodnotnější informace než uživatel. Co se týká povětrnostních podmínek a skladby dopravního proudu, tyto faktory nelze ovlivnit a chyby jimi způsobené se na odhadu vždy projeví. To je dáno vlastnostmi optické detekce.

Za povšimnutí stojí kolísání odhadu v situaci po rozsvícení signálu volno, kdy se vozidla rozjíždějí. Mohlo by se zdát, že odhad selhává avšak položme si otázku jak velká je v danou chvíli skutečná délka kolony. Dochází zde k přechodovému jevu (rozjíždějící se kolona vozidel se po určitou dobu nejeví ani jako kolona ani jako projíždějící vozidla), jehož eliminace je možná avšak není pro navrhované použití nutná.



Smyčka	Délka [m]	Obsazení	Suma
1.	1.2	ano	+
2.	1.4	ano	+
3.	1.8	ne	+
4.	2.2	ano	+
5.	2.5	ne	+
6.	2.8	ano	+
7.	3.2	ne	0
8.	3.7	ne	0

Obrázek 3.3: Červeně zvýrazněné délky smyček nejsou započítány

4

Sběr dat

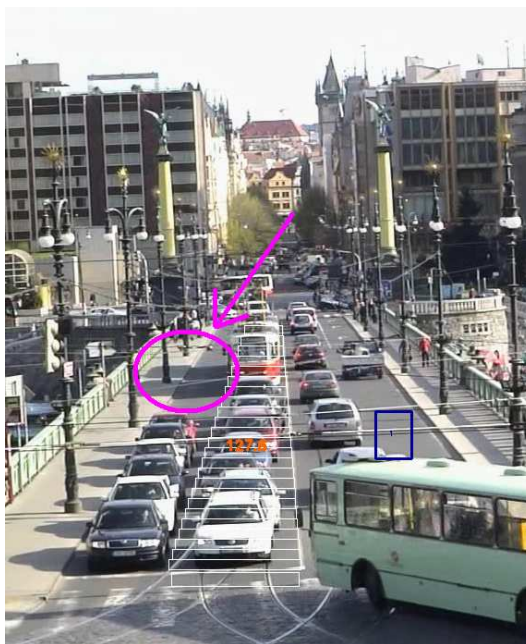
Pro sběr dat byla použita kompaktní kamera SONY. Ta disponuje nízkou hmotností a malými rozměry, tedy velkou mobilitou. Nastavení expozice a dalších parametrů kamery se budeme věnovat dále v kapitole 4.3. Pro uchycení kamery byl použit masivní železný stativ. Při ručním snímání by mohlo dojít k roztřesení obrazu, což by při detekci značně znehodnotilo snímek s pozadím. Posun kamery by se jevil jako pohyb objektů pozadí. Tím by se algoritmu jevily jako popředí a došlo by ke zkreslení výsledného odhadu.

Všechna prezentovaná data byla pořízena v rozmezí listopad 2006 - květen 2007. Letošní zima nebyla příliš štědrá, co se týče sněhové nadílky, a proto se nepodařilo natočit žádné video na sněhu.

4.1 Požadavky na scénu

Aby program bezchybně fungoval, klademe na scénu následující požadavky. Prvním je, aby se zde neobjevovaly stíny. Stín se pohybuje současně s vozidlem a vzhledem k jasů povrchu vozovky a jasů stínu se tento jeví jako popředí. Algoritmus tedy vyhodnotí pohyb stínu jako pohyb vozidla. Lze se tomu bránit eliminací stínů (viz kapitola 5.1), nebo vhodným umístěním kamery. Pro odhad délky kolony jsou nebezpečné boční stíny, které zasahují do sousední kolonové smyčky viz obrázek 4.1. Naopak stíny za vozidly jsou žádoucí, protože ucelují kolonu tím, že zakryjí viditelné pruhy vozovky mezi stojícími vozidly. Pokud však jejich délka zasahuje příliš daleko za poslední vozidlo v koloně, mohou se podílet na zvýšení nepřesnosti odhadu.

Druhý požadavek je kladen na umístění kamery. Ta musí scénu snímat z patřičné výšky, aby docházelo pokud možno k co nejmenšímu překrývání vozidel. V idealizovaném případě by kamera byla přímo nad kolonou, přičemž její optická osa by směřovala svisle dolů. Dále platí, že čím je kamera k sledované oblasti blíže, tím přesnějšího odhadu lze dosáhnout. Jak již bylo řečeno program načítá výslednou délku kolony v krocích (sumace délek dílčích smyček).



Obrázek 4.1: Stíny nebezpečné pro odhad kolon v pruhu neobsazeném kolonovou smyčkou jsou zvýrazněné.

Třetí požadavek podmiňuje správnou funkci programu výskytem vozidel, jejichž barva se dostatečně liší od barvy povrchu vozovky. Tento požadavek nelze ovlivnit, proto je třeba hledat alternativní řešení. Při obnově pozadí jsou stanovena dvě pravidla, která zmíněný problém do jisté míry řeší. Jedním je detekce bílé barvy. Při příjezdu bílého vozidla se toto nezapisuje do pozadí protože jas jeho karosérie překračuje maximální možný jas vozovky. Ten byl stanoven experimentálně. Druhým pravidlem je hlídání poměru jednotlivých barevných složek. Barva vozovky se pohybuje od černé (nová vozovka) až po šedou. To jsou barvy, které obsahují přibližně stejné jasové hodnoty pro jednotlivé kanály RGB. Barevná vozidla mají poměr jasu v kanálech RGB různý. Takové pixely se do pozadí rovněž nezapiší.

4.2 Výběr lokality

Je nutné, aby měřená kolona byla umístěna v čelním, příp. zadním pohledu vzhledem ke kameře. Boční pohled není vhodný, protože z parametrů, které při konstrukci smyčky zadává uživatel, nelze dopočítat perspektivu (viz kapitola 3.4). Jako vhodné se jeví umístit kameru na výložník návěstidla. Zde by byl dosažen i požadovaný nadhled (vhodná výška kamery nad scénou). Naším úsilím bylo získat záznam z monitorovacích kamer umístěných na pražských křižovatkách. Ty jsou vzhledem k pozorované dopravě umístěny mnohem vhodněji než lze z pozice chodce umístit námi použitou kameru. Narazili jsme však na následující problém. Většina pro nás zajímavých kamer byla v rámci modernizace vybavena servo motorem pro natáčení. K jeho ovládání má současně přístup několik dispečerů aniž by byly nastaveny priority. Tedy neexistuje nástroj, který by zajistil, že v průběhu natáčení nedojde k pohybu kamery.



a) Dopravní značení ve výhledu.



b) Hranice křižovatky mimo záběr.



c) Kolona umístěná vodorovně.



d) Nebezpečné stíny, špatně umístěná kamera z hlediska zásad správné expozice.



e) Správné umístění kamery.

Obrázek 4.2: Možnosti umístění kamery.

4.3 Okolní vlivy, expozice

Jedním z důležitých faktorů, ovlivňujících výsledek odhadu je režim expozice kamery. V zásadě lze použít dva principy. Prvním je automatická expozice. Její funkce spočívá v tom, že kamera pomocí čidla měří světelnost scény a na jejím základě upraví své parametry (doba expozice, clona, citlivost snímače) tak, aby výsledný obrázek nebyl přexponovaný nebo podexponovaný. Tato funkce může způsobovat problémy při detekci kolon např. při příjezdu velkého světlého objektu (bílý autobus). Ten vnese do scény velikou světlou plochu, tím zvýší jas a reakcí kamery bude zvýšení clonového čísla. To způsobí, snížení jasu objektů pozadí. Stejný objekt snímáný s větší clonou se ve výsledku jeví jako tmavší. Pokud je tento rozdíl nad prahovou hodnotou, jeví se celá scéna jako popředí a informace o podobě pozadí se začne obnovovat zvýšenou rychlostí, protože algoritmus tuto situaci nedovede odlišit od např. zastínění přímého slunečního svitu oblakem, kdy je rychlé přizpůsobení na nové světelné podmínky žádoucí.

Jako řešení se nabízí buďto detekce velkých světlých objektů, které mohou expozici přestavit. To by ovšem znamenalo zpomalení celého programu. Dalším úskalím je mlhavé vymezení vlastností hledaného objektu. Jiným možným řešením, se kterým je spojeno zpomalení programu, je automatické vyrovnaní histogramu pro každý snímek. Nejelegantnějším a nejjednodušším se v tuto chvíli jeví nastavení stálé expozice kamery za cenu toho, že v jistých okamžicích dojde k přexponování. To si lze představit jako ořezání jasu maximální hodnotou (bílá barva) což se na výsledku detekce neprojeví negativně.



Obrázek 4.3: Automatická expozice způsobuje potíže při příjezdu světlého vozu.

Z okolních vlivů nás nejvíce zajímají ty, které které razantně zasáhnou do jasu scény. Jako jsou náhle nasněžení, déšť (dochází k odleskům na vozovce) a setmění. Pokud chceme algoritmem dynamicky řídit křižovatku, je nutné zajistit plnou funkčnost ve všech zmíněných podmínkách. Automatickou expozici jsme již dříve vyloučili. Nabízí se tedy použití více kamer (pro noční a denní snímání). Jiným řešením je návrat k automatické expozici za předpokladu že tato může probíhat až po předem stanovené době, která bude delší než doba jednoho cyklu křižovatky. Tím se zajistí reakce na změny dlouhodobé (delší než jeden cyklus křižovatky) a současně odbourá nežádoucí reakce na změny krátkodobého rázu.

Předchozí úvahy nejsou v předloženém algoritmu implemetovány, protože jeho primárním

zaměřením je práce s videem, tedy parametry vstupu programu (výstup kamery) nelze ovlivnit.

5

Experimenty

Kromě experimentů s jednotlivými modely snímku pozadí (Gausův model, optický tok atp.), které již byly stručně komentovány v kapitole 2. Byl podniknut poměrně úspěšný pokus o odstranění stínů. Ve výsledné práci není zahrnut vzhledem k výpočetní náročnosti.

5.1 Odstranění stínů

Jak již bylo řečeno, objekty pohybující se na sluncem osvětlené scéně doprovází jejich stín. Tato skutečnost znepříjemňuje detekci pohybu, protože jasová změna stín–osvětlená vozovka je natolik velká, že je při použití metody odečítání snímku od statického pozadí je detekována jako pohyb. Velikost pohybujícího se objektu je potom nadhodnocena v závislosti na velikosti (délce) stínu, tu ovlivňuje úhel který svírá světelný zdroj s vodorovnou rovinou. Naší snahou je stíny odstranit.

5.1.1 Princip

Pro eliminaci stínů využijeme poznatku, že barva vozovky se v místě zastínění nemění, změní se pouze její jas. Ten je s jednotlivými barevnými kanály provázán tzv. kolorimetrickou rovnicí:

$$Y_{xy} = 0.299 \cdot R_{xy} + 0.587 \cdot G_{xy} + 0.114 \cdot B_{xy} \quad (5.1)$$

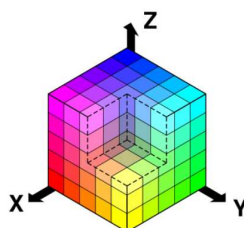
Kde Y_{xy} je jasová složka pixelu o souřadnicích x, y a R_{xy} , G_{xy} a B_{xy} jsou jednotlivé barevné kanály. Z předchozího vyplývá, že změna jasů o ΔY v zastíněné části obrazu vyvolá změnu o $0.299 \Delta Y$ v kanále R resp. $0.587 \Delta Y$ v kanále G resp. $0.114 \Delta Y$ v kanále B , tedy jinou odezvu v každém barevném kanále. Této vlastnosti barevného prostoru RGB říkáme nelinearita. Dalšími faktory které ovlivní do jaké míry se změní odezvy v jednotlivých kanálech jsou barva světelného zdroje a kvalita snímacího zařízení, na tu jsou kladeny velké nároky hlavně v tmavých místech obrazu (viz obrázek 5.3).

Kromě prostoru RGB existují další, jejichž definice mohou více respektovat způsoby

jakými vnímá barvy lidský mozek, např. HSV. Zde je každý pixel reprezentován třemi složkami. Barvou, nasycením a jasnem. Tedy pixel stínu bude mít stejnou barvu a nasycení jako pixel vozovky, bude se lišit pouze jasnem. Tím bude snadno rozlišitelný od pixelu karoserie vozidla, který se od vozovky bude lišit i barvou. Tato jednoduchost se v praxi nepotvrdila. Proto jsem pro experiment zvolil prostor RGB.

5.1.2 Postup

Vytvořený algoritmus připouští určitou barevnou změnu. Jeho činnost spočívá v určení velikosti změn jednotlivých složek pixelu pro každý barevný kanál. Jsou tedy definovány tři intervaly pro R , G a B , které společně umožní konkrétní barevnou mutaci. Ta by na obrázku 5.1 ležela asi uprostřed stěny YZ . Měla by tvar krychle o velikosti hrany asi jedna pětina celého RGB prostoru.



Obrázek 5.1: Krychle znázorňující barevný prostor RGB.

Výsledek zachycují následující obrázky. Stíny jsou eliminovány pouze u pohybujících se objektů. Stíny statických objektů problémy při detekci nezpůsobují, jejich pohyb je natolik pomalý, že je stačí kompenzovat algoritmus pro dynamickou změnu pozadí. V experimentu obnovují pozadí váženým součtem každého pátého snímku s váhou $\alpha = 0,9$ (dle rovnice 2.9).

Jak je vidět na obrázku 5.2 nečistoty na boční straně automobilu odpovídají barevné



Obrázek 5.2: Zašpiněný bok vozidla se jeví jako stín.

změně definované pro rozpoznání stínu a ta se stává „průhledná“. Tento nedostatek lze minimalizovat následným voláním algoritmů pro seskupování nad maskou pohybu.

Další zajímavostí dobře pozorovatelnou na obrázku 5.3 je příliš tmavá část stínu v blízkosti vozidla. Nízká intenzita osvětlení je kamerou interpretována jako jiná barva než ve skutečnosti a barevná změna potom nespadá do definovaného intervalu.



Obrázek 5.3: Tmavá část stínu je díky nízkému jasů chybně barevně interpretována.

5.1.3 Nastavení mezí intervalů

Výše pořízené snímky byly zpracovány při nahrazování změn spadajících současně do následujících intervalů. Ty byly stanovili experimentálně tak, že za běhu programu jsem hodnoty jednotlivých mezí měnil tak dlouho, až výsledek odpovídal mým představám.

- pro kanál R : 0 – 73
- pro kanál G : 41 – 114
- pro kanál B : 44 – 107

Uvedené hodnoty platí pro snímek s barevnou hloubkou 256 bitů pro každý barevný kanál.

Pro automatizovaný běh je nutné, aby hranice intervalů definovala sama aplikace. K tomu byla využita následující úvaha. Při známé velikosti jasové změny stín/osvětlená scéna, napoví kolorimetrická rovnice kam umístit středy hledaných intervalů pro jednotlivé kanály R , G a B . V případě, že budou k dispozici potřebná data, lze určit i jejich odchylku od průměrné hodnoty, tedy šířku hledaných intervalů.

5.1.4 Výpočet mezí

Sledováním několika pixelů, které jsou střídavě zastíněné a osvětlené bylo získáno 1299 údajů. Z nich spočtené základní statistické ukazatele zachycuje tabulka 5.1

Ty poslouží k základnímu nastavení algoritmu. O tom, zda je pixel tmavý nebo světlý (resp. ve stínu nebo osvětlený) je rozhodnuto na základě celkového průměru jasů všech pixelů tak, ty s nadprůměrným jasem spadají do množiny světlých, ostatní do množiny zastíněných. Se zvětšujícím počtem údajů klesá směrodatná odchylka. Ta určuje šířku

	Světlo	Stín
Průměr	131,3245	59,8049
Směrodatná odchylka	4,5494	22,4041
Počet údajů	1017	282

Tabulka 5.1: Výsledek pozorování pixelů.

hledaných intervalů. Proto bude nutné omezit počet vyhodnocovaných dat, případně provést jiné kroky, jimiž bude docíleno použitelné šířky intervalů, definovaných následujícími rovnicemi:

$$\Delta = \frac{1}{n} \sum_{i=0}^n |x_{n-Svetlo} - \bar{x}_{n-Svetlo}| + \frac{1}{n} \sum_{i=0}^n |x_{n-Stin} - \bar{x}_{n-Stin}| \quad (5.2)$$

$$B \in \langle (\bar{x}_{Svetlo} - \bar{x}_{Stin}) \cdot 0, 114 - \Delta, (\bar{x}_{Svetlo} - \bar{x}_{Stin}) \cdot 0, 114 + \Delta \rangle \quad (5.3)$$

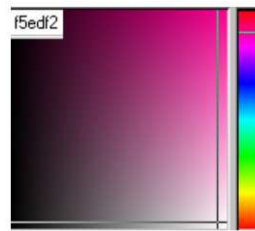
$$R \in \langle (\bar{x}_{Svetlo} - \bar{x}_{Stin}) \cdot 0, 299 - \Delta, (\bar{x}_{Svetlo} - \bar{x}_{Stin}) \cdot 0, 299 + \Delta \rangle \quad (5.4)$$

$$G \in \langle (\bar{x}_{Svetlo} - \bar{x}_{Stin}) \cdot 0, 587 - \Delta, (\bar{x}_{Svetlo} - \bar{x}_{Stin}) \cdot 0, 587 + \Delta \rangle \quad (5.5)$$

5.1.5 Hodnocení experimentu

Ukazuje se, že se odchylka Δ (z rovnice 5.2) automaticky nastaví na malou hodnotu. Tuto skutečnost lze kompenzovat buďto robustnějším modelem statistiky (místo několika jednotlivých pixelů bude předmětem zájmu ploška – několik sousedících pixelů) nebo umělým zvětšením odchylky o konstantu. Tu lze určit z rozdílů barev jednotlivých pixelů reprezentujících povrch vozovky (vozovka není jednobarevná plocha).

Ani s tímto opatřením není výsledek ideální. Je to zřejmě důsledkem zabarvení světelného zdroje, které je třeba kompenzovat. Barvu zdroje lze určit buďto experimentálním nastavením jednotlivých barevných složek nebo automaticky nalezením bílého pixelu a následným vyhodnocením odchylek v barevných kanálech od jasu celého pixelu. Naše scéna jeví nejlepší výsledek při kompenzaci $R = 10$, $G = 20$ a $B = 13$, což odpovídá barvě na obrázku 5.4.



Obrázek 5.4: Barva světelného zdroje.

Univerzálnost algoritmu je vyvážena kvalitou detekce. Ta je zde o poznání horší než v případě ručního nastavení. Odpadá zde však nutnost neustálého přizpůsobování intervalů pro určení barevné mutace měnícím se podmínkám. K lepšímu výsledku by mohl napomoci kvalitnější model pozadí.



Obrázek 5.5: Automatické nastavení intervalů nepodává přesvědčivý výsledek .

Vozidlo	Odhadovaná délka [m]
Osobní automobil	6
Nákladní automobil	10
Autobus, n. auto s návěsem	18
N. automobil s přívěsem	22

Tabulka 5.2: Délky vozidel pro odhad délky kolony.

5.2 Silné a slabé stránky aplikace pro odhad délky kolon

K silným stránkám programu patří schopnost pozorovat scénu i za složitých dopravních situací (kongesce apod.). Další silnou vlastností je jeho nenáročnost jednak výpočetní, program dokáže sledovat dvě kolony na barevném snímku 240 x 320 pixelů rychlostí cca 49 sn/sekundu (tento údaj byl pořízen na stroji s procesorem Celeron M420), jednak ekonomickou. Pro běh programu stačí dobře umístěná kamera nad křižovatkou a počítač. Není nutná složitá instalace intruzivních detektorů.

Za slabou stránku považuji nutnost příznivých podmínek na scéně při startu programu. Při spuštění nad trvale obsazenou křižovatkou nebude správnost odhadu jistá, protože program nedokáže sestavit dostatečně přesný model pozadí.

5.2.1 Metodika testování

Pro testování výsledků programu jsem zvolil následující metodiku. Test začíná ve chvíli, kdy se programu podaří sestavit přesný model pozadí (kolonová smyčka je aspoň krátkodobě neobsazena). Uživatel pozoruje situaci a vždy v okamžiku rozjezdu vozidel zapíše čas od spuštění programu a vlastní odhad délky kolony. Ten vychází z předpokladu, že osobní automobil je dlouhý cca 6 m, nákladní 10 m (viz tabulka 5.2). Údaje jsou mírně nadhodnoceny z důvodu opomíjení mezer, které mezi vozidly vznikají při jejich řazení v koloně. Test probíhá na třech trénovacích záznamech vždy v délce asi 10 minut.

5.2.2 Výsledky

Pozorované výsledky zachycují následující tabulky 5.3, 5.4 a 5.5.

číslo měření	naměřeno [m]	skutečná d. [m]	rozdíl [m]	naměřeno [m]	skutečná d. [m]	rozdíl [m]
1	29,9	22	7,9	19,1	12	7,1
2	33,5	30	3,5	9	6	3
3	4,7	6	1,3	19,1	18	1,1
4	29,9	36	6,1	23,1	24	0,9
5	35,7	36	0,3	27,6	30	2,4
6	35,7	34	1,7	27,6	38	10,4
7	35,7	38	2,3	19,1	38	18,9
8	2,1	6	3,9	2,7	18	15,3
9	35,7	34	1,7	32	32	0

Tabulka 5.3: Výsledky prvního testu

Průměrná hodnota rozdílu mezi naměřenou a skutečnou délkou kolony 3,2 *m* pro smyčku v pravém pruhu a 6,6 *m* pro smyčku v levém pruhu. Algoritmus sestavil model pozadí za 47 *s*. Název testu *testv2* (test je součástí přílohy B).

číslo měření	naměřeno [m]	skutečná d. [m]	rozdíl [m]	naměřeno [m]	skutečná d. [m]	rozdíl [m]
1	5	6	1	0	0	0
2	0	0	0	4,8	6	1,2
3	2,3	6	3,7	0	0	0
4	5	0	5	6,2	6	0,2
5	13,7	15	1,3	15,1	15	0,1
6	0	0	0	0	15	15
7	5	6	1	6,7	6	0,7
8	0	0	0	8,4	12	3,6
9	13,7	15	1,3	17,2	18	0,8
10	0	0	0	12,1	18	5,9
11	3,8	15	11,2	0	6	6
12	0	0	0	0	18	18
13	0	0	0	0	18	18
14	12,7	12	0,7	1,3	10	8,7
15	0	0	0	1,3	16	14,7
16	16	16	0	9,4	6	3,4

Tabulka 5.4: Výsledky druhého testu

Průměrná hodnota rozdílu mezi naměřenou a skutečnou délkou kolony 1,6 m pro smyčku v pravém pruhu a 6,0 m pro smyčku v levém pruhu. Algoritmus sestavil model pozadí za 74 s. Název testu *testv3*.

číslo měření	naměřeno [m]	skutečná d. [m]	rozdíl [m]	naměřeno [m]	skutečná d. [m]	rozdíl [m]
1	52,4	40	12,4	5	60	55
2	66,2	33	33,2	52,6	60	7,4
3	16	12	4	57	60	3
4	66,2	22	44,2	57	60	3
5	66,2	38	28,2	57	60	3
6	66,2	30	36,2	19,4	60	40,6
7	66,2	32	34,2	57	60	3
8	44,2	0	44,2	55,7	60	4,3

Tabulka 5.5: Výsledky třetího testu

Průměrná hodnota rozdílu mezi naměřenou a skutečnou délkou kolony 29,6 m pro smyčku v pravém pruhu a 14,9 m pro smyčku v levém pruhu. Algoritmu se podařilo sestavit model pozadí pouze pro smyčku v pravém pruhu a to za 176 s. Název testu *testv4*. Nutno podotknout, že tento test nesplňoval podmínky nutné pro běh programu. Intenzita provozu byla příliš velká (v levém pruhu) což znemožnilo sestavení modelu pozadí. To se částečně podařilo pro smyčku nad pravým pruhem, avšak odhad v ní byl zkreslen stíny vozidel z levého pruhu.



a) vjezd do Těšnovského tunelu b) Křižovatka u Těšnovského tunelu



c) Hlávkův most

Obrázek 5.6: Rozvržení detekčních prvků při testech.

5.2.3 Hodnocení experimentu

První a druhý test vykazují poměrně dobré výsledky. Chyba mezi naměřenou a skutečnou (resp. odhadnutou uživatelem) hodnotou délky kolony se pohybuje v průměru okolo 4,3 m. Průměrná chyba pro žádnou smyčku nepřesahuje 7 m. Při třetím testu nebyly splněny základní předpoklady pro správný chod programu a proto by bylo nevhodné zahrnovat jeho výsledky do konečného hodnocení. Jeho přínos spočívá v potvrzení tezí o nefunkčnosti programu při nedodržení zmíněných pravidel umístění kamery a startu programu.

6

Závěr

V rámci této práce vznikla aplikace schopná odhadnout délku kolony stojících vozidel před křižovatkou. Aplikace dovede modelovat pozadí i v místech, kde se objekty popředí zastavují a z časového hlediska zaujímají větší prostor než pozadí. Odhad probíhá s jistotou chybou, kterou do aplikace vnese jednak uživatel, jednak je dána vlastnostmi optické detekce (viz kapitola ??). Pro využití navrhované v kapitole 1 je však tato přesnost postačující. Po rozjezdu stojících vozidel dochází na křižovatce k přechodovému jevu (rozjíždějící vozidla již nejsou kolonou avšak současně se neprojevují ani jako volně projíždějící vozidla), který se projevuje kolísáním odhadu.

6.1 Odstranění slabin

Nepřesnost vnesenou uživatelem lze eliminovat instalací měřícího zařízení na kameru. Nejlépe by vyhovoval laserový dálkoměr, který by s dostatečnou přesností změřil vzdálenost kamery od stop čáry. Zmíněná nepřesnost se v programu projeví několikrát. Jednak při zadání výšky kamery, jednak při zadání horizontální vzdálenosti kamery od stopčáry. Nutno podotknout, že zmíněné vzdálenosti se měří k pomyslnému bodu a proto je nelze přesně určit. Oba parametry se znovu použijí při výpočtu přímé vzdálenosti kamery a stopčáry. Zde znovu vzniká chyba, která je podpořena následným zaokrouhlením. Dálkoměr by tedy byl pro přesnost aplikace velkým přínosem.

Nepříjemné kolísání odhadu při rozjezdu vozidel by mohlo odstranit propojení s řadičem. Program by měl informaci o rozsvíceném signálu a mohl by kolonu odhadovat pouze v okamžiku, kdy by na semaforu svítila návěst stůj. V opačném případě by odhadovaná vzdálenost byla rovna nule nebo by měření vůbec neprobíhalo. To by harmonizovalo průběh odhadu a zprůhlednilo výsledek měření, kde by byly jasně rozlišitelné jednotlivé cykly křižovatky. Dále by toto napomohlo při statistickém zpracování, kde by hodnoty přechodových jevů nevnašely do výsledku další nepřesnost.

6.2 Další vývoj

Program odhaduje délku kolon bez potíží v situaci, kdy startuje za příznivých podmínek. Po spuštění neexistuje model pozadí. Pro jeho pořízení je nutné, aby obsazenost sledované vozidly byla menší než 50%. O tomto problému již byla řeč. Mé další úsilí bude směřovat k vytvoření modelu pozadí resp. startovacího režimu programu, aby tento bylo možné spustit i za složitých dopravních podmínek (nasycení oblasti), a byl při tom vytvořen bezchybný model pozadí. Dále se zaměřím na práci s kamerou. Předložený program pracuje na základě údajů z videa, to je užitečné pro off-line sledování dopravní situace, avšak aby mohl být využit při řízení provozu, je nutné, aby zpracovával snímky přímo z kamery.



Obrázek 6.1: Výsledné okno aplikace.

Literatura

- [1] B. Kořata: *XML-Úvod*, URL: <http://www.root.cz/clanky/xml-uvod/> 2001
- [2] J. Kosek: *Inteligentní podpora navigace na WWW s využitím XML*, VŠE Praha, Diplomová práce 2001
- [3] J. Novovičová: *Pravděpodobnost a matematická statistika*, ČVUT Praha, Skripta 1999
- [4] J. L. Williams: *Gaussian mixture reduction for tracking multiple maneuvering targets in clutter*, Air force institute of technology, Ohio, 2003
- [5] Kolektiv autorů: *Mixture model*, URL: <http://en.wikipedia.org/wiki> 2007
- [6] Z. Votruba, M. Kalika, J. Klečáková: *Systémová analýza*, ČVUT Praha, Skripta 2004
- [7] M. Svítek, J. Borka, M. Vlček: *Modelování systémů a procesů*, ČVUT Praha, Skripta 2001
- [8] P. Příbyl, R. Mach: *Řídící systémy silniční dopravy*, ČVUT Praha, Skripta 2003
- [9] P. Příbyl: *Optimalizace propustnosti městské dopravní sítě využívající satelitní systém Galileo*, Článek časopisu AUTOMATIZACE (12-2004) 2004
- [10] D. Marshall: *Optical Flow*, URL: http://www.cs.cf.ac.uk/Dave/Vision_lecture/node45.html 1997



Návod k obsluze programu pro odhad délky kolon

A.1 Požadavky pro běh programu

Pro bezproblémový chod programu je nutno:

- Procesor Pentium 1,5 GHz a vyšší
- Operační systém Windows (program byl testován na verzi XP)
- Knihovna OpenCV (zdarma ke ztažení na <http://sourceforge.net/projects/opencvlibrary/>)

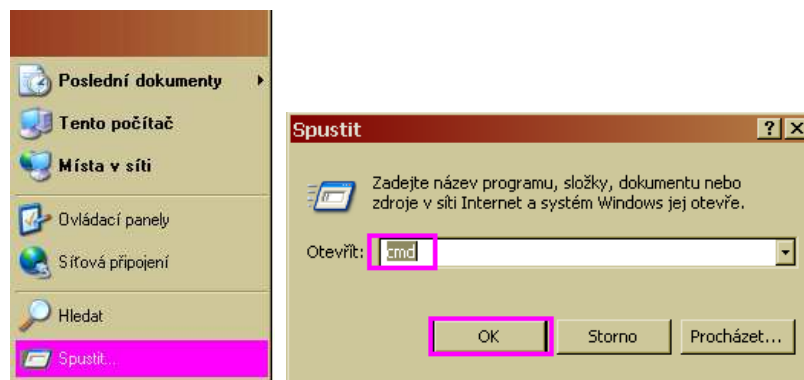
Poslednímu zmíněnému požadavku lze vyhovět nakopírováním potřebných souborů *.DLL z příloženého CD (příloha B přímo do systémového adresáře C:/WINDOWS/SYSTEM32/. Konkrétně se jedná o soubory *cvaux100.dll*, *cvcam100.dll*, *cxcore100.dll*, které jsou standardně součástí distribuce knihovny *openCV*.

A.2 Spuštění programu a ovládání

Při spouštění je nutné zadat programu dva argumenty. Jako první název zpracovávaného videosouboru, druhým argumentem je pracovní název scény. K němu se váže konfigurační soubor, který uchovává informace o rozmístění a vlastnostech detekčních smyček. Pro jedno video lze tedy uchovat současně několik konfigurací smyček, z nichž se požadovaná vybere zadáním příslušného názvu při spouštění programu. V případě, že konfigurační soubor pro zadaný název není nalezen, spustí se program v režimu *konfigurace*.

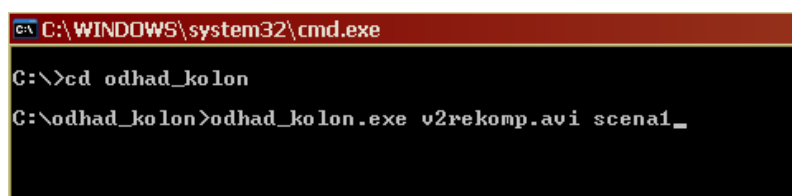
A.2.1 Spuštění programu z příkazového řádku

V systému Windows lze spustit příkazový řádek například zadáním příkazu *cmd* do okna vyvolaného z nabídky *Start* → *Spustit...*



Obrázek A.1: Spuštění příkazového řádku.

Následně je nutné zadat příkaz ke spuštění programu v tomto tvaru
cesta_k_programu/odhad_kolon.exe cesta_k_video/video.avi nazev_sceny



Obrázek A.2: Zadání příkazu s argumenty.

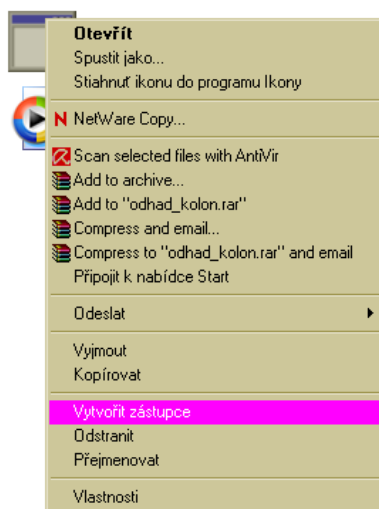
Ve výsledném okně je kolonová smyčka zobrazena bílou (obsazený segment) resp. šedou barvou (neobsazený segment).



Obrázek A.3: Okno aplikace.

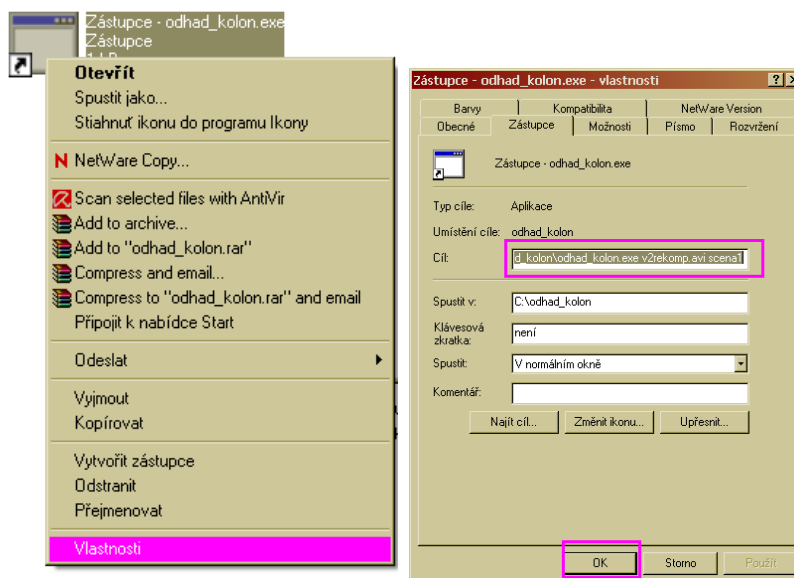
A.2.2 Nastavení parametrů zástupci (jiná možnost spuštění)

Další možností jak program spustit je dvojklikem v grafickém prostředí *Windows*. K tomu je však nutné zadat vstupní argumenty (soubor s videem a název scény). Toho lze docílit stisknutím pravého tlačítka myši nad ikonou programu a následným výběrem nabídky *Vytvořit zástupce*.



Obrázek A.4: Vytvoření zástupce.

Nad zástupcem lze obdobným způsobem vyvolat nabídku *Vlastnosti*. Zde na kartě *Zástupce* u položky *Cíl* požadované argumenty dopsat a zadání potvrdit. Nyní je možné spouštět program dvojklikem na ikonu zástupce.



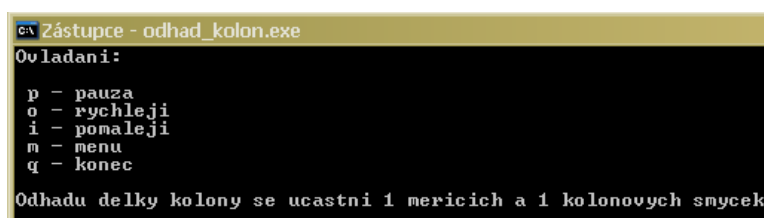
Obrázek A.5: Vlastnosti zástupce programu.

A.2.3 Hlavní menu a ovládací klávesy

Za běhu lze program ovládat pomocí následujících kláves

- *P* - pro zastavení programu, opětovné spuštění proved'te stiskem libovolné klávesy.
- *O* - pro zrychlení, standardně se program spustí maximální možnou rychlostí. Urychlení probíhá v krocích (po 100 ms).
- *I* - pro zpomalení.
- *M* - zobrazí hlavní menu programu.
- *Q* - ukončí běh programu.

Seznam ovládacích kláves je vypsan při spuštění programu do okna konzole.



Obrázek A.6: Seznam ovládacích kláves.

Pro uživatelský komfort je hlavní menu při spuštění programu skryto. Vyvolat lze klávesou *M*. Z něj je možné ovládat základní funkce programu kliknutím na zvýrazněné oblasti:

- *Zapisuj výsledky* - za běhu programu je vytvářen XML soubor s údaji o odhadu délek kolon, které přísluší daným časovým okamžikům. Při spuštění je tato volba neaktivní a soubor se automaticky nevytváří.
- *Konfigurace* - přepne program do režimu konfigurace (viz kapitola A.3).
- *Konec programu* - ukončí běh programu stejně jako klávesa *Q*.
- *Zobrazuj informace* - ve spodní části hlavního okna zobrazí informační řádek. Dále jsou zobrazeny. V informačním řádku lze dohledat čas od spuštění programu, počet zpracovaných snímků za sekundu, informaci o tom, zda je aktivován režim zápisu výsledků do souboru a dále doplňující údaje odhadu kolonových smyček (průměrná a maximální délka).

A.3 Konfigurace

Tento režim se aktivuje automaticky v případě, že při spuštění nebyl nalezen požadovaný konfigurační soubor. Nebo jej lze vyvolat pomocí hlavního menu. V režimu *konfigurace* lze rozmístit dva druhy detektorů (smyček) a zadat jejich konfigurační parametry. Úkony týkající se rozmístění provádí uživatel pomocí myši, parametry smyček zadává vstupem z klávesnice.



Obrázek A.7: Zobrazený informační řádek.



Obrázek A.8: Konfigurační okno.

Konfigurace je aktivní dokud nejsou rozmístěny všechny požadované smyčky. Poté ji lze ukončit kliknutím na oblast *Konec konfigurace*. Tím bude vytvořen resp. přepsán (pokud již existuje) konfigurační soubor a program se přepne do režimu *sledování*. Vytvoření virtuálních detektorů je možné provést dle kroků uvedených v kapitolách A.3.1 a A.3.2.

A.3.1 Kolonová smyčka

Kolonová smyčka sleduje a průběžně informuje o délce kolony. Umisťuje se do prostoru před stopčarou a to jak v čelním tak zadním pohledu. Důležité je začít s konstrukcí smyčky na stopčáře. Konstrukce probíhá pomocí dvou bočnic v několika krocích.

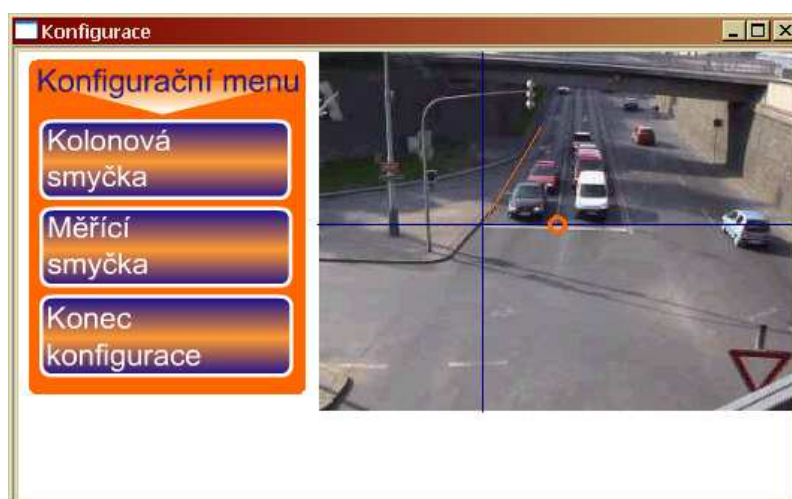
- Nejprve vyvolejte režim konstrukce kolonové smyčky kliknutím na oblast *Kolonová smyčka* v levé části konfiguračního okna. Poté umístěte první roh smyčky. Je nutné aby tento roh ležel na stop čáře (z důvodu automatického dopočítání perspektivy), nezáleží na tom, zde je vpravo nebo vlevo. Roh umístěte nastavením kurzoru na

požadovanou pozici a stisknutím levého tlačítka. To přidržte stisknuté a posuňte kurzor do druhého rohu smyčky vymezené zobrazenou bočnicí. Zde pravé tlačítko uvolněte. Oranžově zvýrazněná čára představuje první bočnici.



Obrázek A.9: První bočnice.

- Pokud došlo při konstrukci k chybě, lze ji napravit novým zadáním bočnice. v opačném případě zadání potvrďte stiskem pravého tlačítka. To způsobí přepnutí do režimu konstrukce druhé bočnice. V okně je zobrazen modrý kříž. Ten vymezuje možné počátky druhé bočnice. Horizontála definuje počátky pro přímé smyčky (pohled zepředu nebo zezadu), vertikála potom pro smyčky v bočním pohledu. Ty lze konstruovat, avšak program nepodporuje jejich funkčnost z důvodu nemožnosti dopočítání délek jednotlivých segmentů (nelze provést výpočet lineární perspektivy z důvodu nedostatku informací získaných z následně zadaných parametrů).



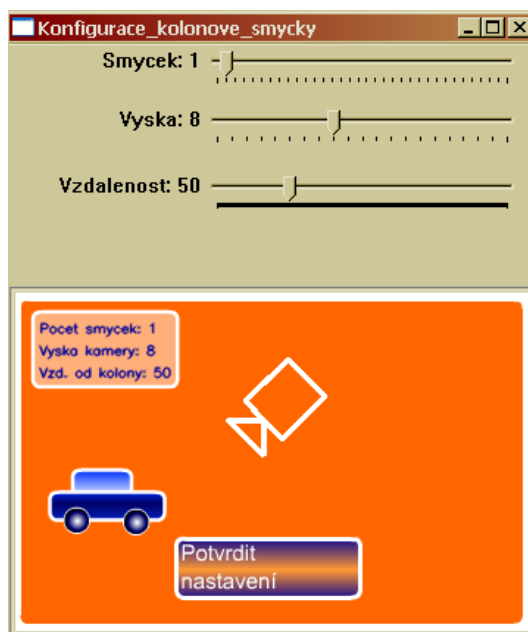
Obrázek A.10: Druhá bočnice.

- Pravým kliknutím umístíte třetí roh a za současného stisku tlačítka posuňte kurzor do poslední hraniční polohy. K nalezení její pozice napomáhá pomocná čára, která se zobrazí při přiblížení kurzoru do množiny možných cílů. Při nezdařilé konstrukci bočnice lze opět chybu napravit, obdobným způsobem jako v prvním případě. Konstrukci potvrďte stiskem pravého tlačítka.



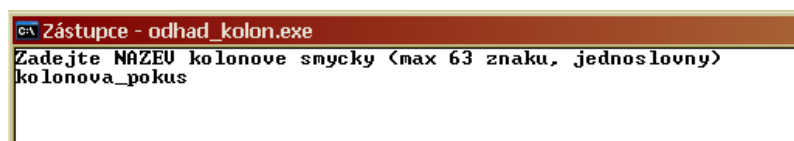
Obrázek A.11: Potvrzení konfigurace smyčky.

- Nyní se zobrazí okno *Konfigurace kolonové smyčky*. Zde pomocí posuvníků nastavte pozici kamery vzhledem ke stopčáře a dále počet segmentů, ze kterých bude kolonová smyčka složena.



Obrázek A.12: Pomocné konfigurační okno.

- Jako poslední krok zadejte název nově zkonstruované smyčky. To proved'ete přepnutím do okna konzole, které se při výzvě na reakci uživatele zbarví bíle. Zadání potvrďte klávesou ENTER. Nyní je smyčka připravena ke spuštění.



Obrázek A.13: Zadání parametrů do okna konzole.

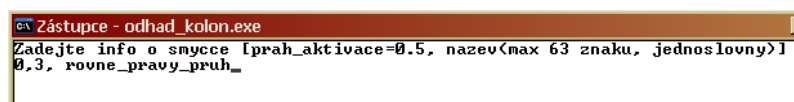
A.3.2 Měřicí smyčka

Měřicí smyčka slouží k počítání vozidel. Lze ji umístit na libovolné místo na scéně. Vzhledem k nezávislosti jednotlivých smyček se tyto mohou i překrývat. Konstrukce probíhá definováním dvou rohů a diagonální vzdálenosti. Kliknutím do oblasti *Měřicí smyčka* přepnete program ro režimu pro vytvoření požadovaného detektoru. Nastavte kurzor myši na místo, kam chcete umístit roh smyčky. Při stisknutí levém tlačítku definujte posunem myši diagonální vzdálenost a tlačítko uvolněte.



Obrázek A.14: Konstrukce virtuální smyčky.

Nyní je smyčka zobrazena ve své budoucí poloze, pokud nevyhovuje vašim představám, můžete ji stejným postupem jako při konstrukci přesunout na jiné místo. Zadání potvrďte stiskem pravého tlačítka myši a v okně konzole (které se zbarví bíle) zadejte požadované parametry. Potvrďte klávesou ENTER.



Obrázek A.15: Zadání parametrů měřicí smyčky do konzole.

A.4 Zpracování výsledků

Výsledný XML soubor lze zpracovat například v aplikaci EXCEL. Při otevření se jednotlivé údaje nakopírují do oddělených buněk, odkud je lze snadno statisticky analyzovat díky funkcím zahrnutým v aplikaci EXCEL.

Jiným způsobem, na který je program připraven, je vystavení naměřených informací na webu. K tomu slouží soubor s kaskádovými styly, vytvořený v rámci bakalářské práce. Ten je součástí přílohy ???. Nutno poznamenat, že pro korektní zobrazení výsledku je nutné použít prohlížeč Mozilla Firefox (obrázek 3.1).

B

CD příloha – aplikace pro odhad délky kolon