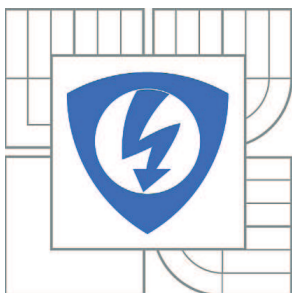


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH  
TECHNOLOGIÍ

ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION  
DEPARTMENT OF CONTROL AND INSTRUMENTATION

## DETEKCE A POČÍTÁNÍ AUTOMOBILŮ V OBRAZE (VIDEODETEKCE)

VIDEODETECTION - TRAFFIC MONITORING

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

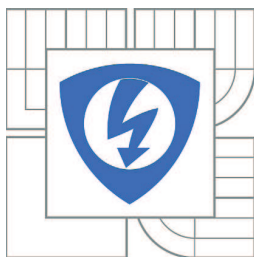
Bc. LUBOMÍR KOZINA

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. PETER HONEC

BRNO 2010



VYSOKÉ UČENÍ  
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

Ústav automatizace a měřicí techniky

# Diplomová práce

magisterský navazující studijní obor  
**Kybernetika, automatizace a měření**

**Student:** Bc. Lubomír Kozina

**ID:** 78141

**Ročník:** 2

**Akademický rok:** 2009/2010

## NÁZEV TÉMATU:

**Detekce a počítání automobilů v obraze (videodetekce)**

## POKYNY PRO VYPRACOVÁNÍ:

Cílem je detekovat a počítat automobily v dynamické scéně (video), reálná data z dopravních kamer. Rozšířte algoritmy o noční snímky a navrhnete metody pro klasifikaci vozidel do kategorií, určení průměrné rychlosti a obsazenosti jízdního pruhu.

## DOPORUČENÁ LITERATURA:

**Termín zadání:** 8.2.2010

**Termín odevzdání:** 24.5.2010

**Vedoucí práce:** Ing. Peter Honec

**prof. Ing. Pavel Jura, CSc.**

*Předseda oborové rady*

## UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

## Detekce a počítání automobilů v obraze (videodetekce)

Diplomová práce

Studijní obor: Kybernetika, automatizace a měření  
Autor práce: Bc. Lubomír Kozina  
Vedoucí práce: Ing. Peter Honec, Ph.D.

### Anotace

V diplomové práci na téma *Detekce a počítání automobilů v obraze (videodetekce)* jsem se zabýval vyhledáváním pohybujících se objektů v sekvenci na sebe navazujících snímků z dopravní kamery. V práci jsou popsány různé metody výpočtu modelu prostředí a následného označení pohybujících se vozidel, počítání pohybujících se vozidel nebo určení přibližné rychlosti pohybu vozidel.

Pro pohodlnější vyhodnocení snímané scény bylo vytvořeno uživatelské rozhraní v programu MATLAB.

### Klíčová slova

počítačové vidění, zpracování obrazu, detekce pohybu, detekce vozidel, model prostředí, detekce hran, detekce rohů, optický tok

## Videodetection – traffic monitoring

Master's thesis

Specialization:                   Cybernetics, Control and Measurements

Author:                             Bc. Lubomír Kozina

Supervisor:                     Ing. Peter Honec, Ph.D.

### Annotation

In this master's thesis on the topic *Videodetection - traffic monitoring* I was engaged in searching moving objects in traffic images sequence. There are described various methods background model computation and moving vehicles marking, counting or velocity calculating in the thesis.

It was created a graphical user interface for traffic scene evaluation in MATLAB.

### Key words

computer vision, image processing, motion detection, vehicles detection, background model, edges detection, corners detection, optical flow

### **Bibliografická citace**

KOZINA, L. *Detekce a počítání automobilů v obraze (videodetekce)*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2010. 60 s. Vedoucí diplomové práce Ing. Peter Honec, Ph.D.

## Prohlášení

„Prohlašuji, že svou diplomovou práci na téma Detekce a počítání automobilů v obraze (videodetekce) jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.“

V Brně dne: **24. května 2010**

.....  
podpis autora

## **Poděkování**

Děkuji vedoucímu diplomové práce Ing. Peteru Honcovi, Ph.D. za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé diplomové práce.

V Brně dne: **24. května 2010**

.....  
podpis autora

## **OBSAH**

<b>1. ÚVOD .....</b>	<b>11</b>
<b>2. DETEKCE AUTOMOBILŮ .....</b>	<b>12</b>
2.1 Snímky .....	12
2.2 Zpracování snímku.....	13
2.3 Vytvoření modelu prostředí .....	16
2.3.1 Model průměrováním.....	16
2.3.2 Model klouzavým průměrem.....	17
2.3.3 Model mediánem .....	18
2.3.4 Shrnutí metod pro výpočet modelu prostředí.....	19
2.4 Hledání objektů .....	20
2.4.1 Hledání objektů pomocí rozdílu.....	20
2.4.2 Hledání objektů pomocí hran.....	24
2.4.3 Hledání objektů pomocí kombinace předchozích dvou metod.....	27
2.4.4 Hledání objektů pouze v jízdnicích pruzích .....	29
2.4.5 Hledání objektů pomocí detekce rohů .....	31
2.4.6 Hledání objektů metodou optického toku .....	34
2.4.7 Shrnutí metod pro hledání objektů.....	39
2.5 Označení objektů .....	40
2.6 Noční snímky .....	44
2.7 Počítání vozidel.....	49
2.8 Počítání průměrné rychlosti .....	50
2.9 Počítání obsazenosti jízdničního pruhu .....	52
2.10 Klasifikace vozidel do kategorií .....	52
2.11 Uživatelské rozhraní .....	53
<b>3. ZÁVĚR.....</b>	<b>55</b>
<b>4. LITERATURA .....</b>	<b>58</b>
<b>5. SEZNAM ZKRATEK A SYMBOLŮ.....</b>	<b>59</b>
<b>6. SEZNAM PŘÍLOH .....</b>	<b>60</b>



## SEZNAM OBRÁZKŮ

Obrázek 1: Snímek z dopravní kamery .....	12
Obrázek 2: Lichý pulsnímek .....	14
Obrázek 3: Sudý pulsnímek .....	14
Obrázek 4: Šedotónový snímek .....	15
Obrázek 5: Model získaný průměrováním z 25 snímků .....	16
Obrázek 6: Model získaný průměrováním ze 100 snímků.....	16
Obrázek 7: Model získaný klouzavým průměrem z 25 snímků.....	17
Obrázek 8: Model získaný klouzavým průměrem ze 100 snímků.....	17
Obrázek 9: Model získaný mediánem z 25 snímků .....	18
Obrázek 10: Model získaný mediánem ze 100 snímků .....	18
Obrázek 11: Aktuální snímek.....	20
Obrázek 12: Model získaný mediánem z 25 snímků .....	21
Obrázek 13: Rozdíl mezi aktuálním snímkem a modelem .....	21
Obrázek 14: Rozdíl mezi aktuálním snímkem a modelem binárně .....	22
Obrázek 15: Obrys nalezených objektů .....	22
Obrázek 16: Výsledek – objekty označené pomocí rozdílu v akt. bar. snímku.....	23
Obrázek 17: Masky Prewittova hranového detektoru [4] .....	24
Obrázek 18: Hrany nalezené v aktuálním snímku .....	24
Obrázek 19: Hrany nalezené v modelu prostředí.....	25
Obrázek 20: Rozdíl mezi obrázkem 18 a obrázkem 19 .....	25
Obrázek 21: Obrys nalezených objektů pomocí detekce hran.....	26
Obrázek 22: Výsledek – objekty označené detekcí hran v akt. bar. snímku.....	26
Obrázek 23: Rozdíl mezi aktuálním snímkem a modelem binárně – vysoký práh....	27
Obrázek 24: Obrys nalezených objektů pomocí kombinace obou metod.....	28
Obrázek 25: Výsledek – objekty označené kombinací obou metod .....	28
Obrázek 26: Binární mapa snímané scény .....	29
Obrázek 27: Obrys nalezených objektů v jízdnicích pruzích .....	30
Obrázek 28: Výsledek – objekty označené pouze v jízdnicích pruzích.....	30
Obrázek 29: Výstup z Harrisova detektoru významných bodů .....	32

Obrázek 30: Významné body nalezené v aktuálním snímku binárně .....	33
Obrázek 31: Výsledek – označená významná místa ve snímku .....	33
Obrázek 32: Rozdělení druhů pohybu [4] .....	34
Obrázek 33: Ohnisko expanze. a) čas $t_1$ , b) čas $t_2$ , c) optický tok [4] .....	35
Obrázek 34: Výsledek - optický tok    Obrázek 35: Výsledek – optický tok.....	38
Obrázek 36: Hrany pohybujících se objektů pomocí detekce hran.....	40
Obrázek 37: Dilatace hran nalezených objektů.....	41
Obrázek 38: Dilatace a následná eroze nalezených objektů .....	42
Obrázek 39: Obdélníky kolem objektů .....	42
Obrázek 40: Označené objekty v aktuálním snímku.....	43
Obrázek 41: Noční snímek z dopravní kamery .....	44
Obrázek 42: Model snímané noční scény získaný mediánem .....	45
Obrázek 43: Hrany objektů v nočním snímku .....	46
Obrázek 44: Označené objekty v aktuálním nočním snímku.....	46
Obrázek 45: Model snímané noční scény po ekvalizaci histogramu .....	47
Obrázek 46: Označené objekty – ekvalizace histogramu .....	47
Obrázek 47: Jednostopé vozidlo v nočním snímku.....	48
Obrázek 48: Oblasti pro počítání vozidel v jízdních pruzích.....	49
Obrázek 49: Délka úsečky $L$ .....	51
Obrázek 50: Uživatelské rozhraní.....	53

## 1. ÚVOD

Základní funkcí videodetekce je detekce kolon a stojících vozidel (dopravní kongesce), které způsobují potenciální nebezpečí pro ostatní účastníky silničního provozu. Včasné rozpoznání dopravních kongescí má podle [2] význam z hlediska bezpečnosti a plynulosti silničního provozu. Dopravní kongesce způsobují nejenom ekonomické ztráty dané nerovnoměrnou jízdou, ale také ztráty na lidských životech. Na řidiče působí stres vyvolaný takovým způsobem jízdy. Z těchto poznatků lze usoudit, že plynulost dopravy má pozitivní vliv na ekologii a na psychickou pohodu řidiče. Největší problémy s kongescemi se vytvářejí ve městech a na jejich přístupových komunikacích, v prostorech placení mýtného, v místech kde je redukována rychlost nebo počet jízdnicích pruhů. Dopravní kongesce je nutné včas rozpoznat a mít možnost informovat řidiče o neprůjezdnosti komunikace např. pomocí informačních tabulí s několika řádky alfanumerického textu v dostatečném předstihu. Řidič má potom možnost volit alternativní trasu dle svého uvážení nebo za pomoci různých navigačních systémů.

Rozvoj výpočetní techniky a prostředků digitálního zpracování obrazu umožnil, aby různé kamerové systémy významným způsobem přispěly ke zvýšení dostupnosti relevantních dat o dopravní situaci [3]. Tyto systémy zaznamenávají nejen dopravní data týkající se dopravní situace v dané lokalitě jako celku, ale i údaje o jednotlivých vozidlech. Jedná se například o intenzitu dopravy, obsazenost jízdnicích pruhů, detekci stojícího vozidla, detekci kolony apod.

Videodetekční systémy jsou základem pro úspěšné monitorování dopravy a jsou jednou z podmínek úspěšného použití dopravní telematiky (ITS). Videodetekce může sloužit ke sledování porušení pravidel silničního provozu, např. průjezd křižovatkou na červenou, překročení povolené rychlosti apod. Použití videodetekce v mnoha případech neznamena další investiční náklady, protože je možné využít stávajících dopravních kamer. Použitím těchto moderních telematických prostředků můžeme zvýšit propustnost komunikací a významně snížit počet dopravních nehod.

## 2. DETEKCE AUTOMOBILŮ

### 2.1 SNÍMKY

Detekci automobilů provádím v dynamické scéně. Mám k dispozici reálné na sebe navazující snímky z dopravní kamery a v nich se pomocí různých algoritmů snažím najít a označit pohybující se automobily. Snímky zpracovávám v prostředí programu MATLAB v.7.7.0.471 (R2008b). Na obrázku 1 je ukázka snímku z dopravní kamery.



Obrázek 1: Snímek z dopravní kamery

## 2.2 ZPRACOVÁNÍ SNÍMKU

Snímky jsou zpracovány kamerou s prokládaným snímačem. Ten nejprve zpracuje liché řádky a potom zpracuje sudé řádky [1]. Potom je obraz elektronicky složen. Problém vzniká ve chvíli, kdy se snímaná scéna rychle mění. Mezi zpracováním lichých a sudých řádků dojde ke změně snímané scény a po složení obrazu jsou pohybující se objekty na snímku rozmazané, jak je vidět na obrázku 1.

Snímek z kamery se dá rozdělit na dva půlsnímky, z nichž jeden je vytvořen z lichých řádků původního snímku a druhý ze sudých řádků původního snímku. Na těchto dvou půlsnímcích je vidět, že pohybující se vozidla na sudém půlsnímku jsou ve směru jízdy o kousek dál než na lichém půlsnímku, který byl zpracován jako první. Na obrázku 2 je půlsnímek vytvořený z lichých řádků, na obrázku 3 je půlsnímek vytvořený ze sudých řádků. Tento posuv působí drobné komplikace v dalším zpracování snímků, proto jsem se rozhodl dále zpracovávat pouze jeden půlsnímek složený z lichých řádků původního snímku. Tím dojde zároveň ke snížení rozlišení snímku, což má pozitivní vliv na rychlost zpracování snímků a běh celé aplikace.



**Obrázek 2: Lichý půlsnímek**



**Obrázek 3: Sudý půlsnímek**

Dalším krokem ve zpracování snímku je převod barevného (truecolor) snímku na šedotónový snímek, který je vidět na obrázku 4.



**Obrázek 4: Šedotónový snímek**

## 2.3 VYTVOŘENÍ MODELU PROSTŘEDÍ

Cílem vytvoření modelu prostředí je získat obrázek snímané scény bez pohybujících se objektů. Model prostředí musí být v průběhu chodu aplikace aktualizován, aby zachytil změny klimatických podmínek.

### 2.3.1 Model průměrováním

Toto je jednoduchý způsob získání modelu prostředí. Model je vytvořen průměrem ze 25/100 snímků. Pro každý pixel modelu je vypočítána průměrná hodnota jasové úrovně daného pixelu ze 25/100 snímků použitých k výpočtu modelu prostředí. Model získaný průměrováním z 25 snímků je vidět na obrázku 5. Model získaný průměrováním ze 100 snímků je vidět na obrázku 6.



**Obrázek 5: Model získaný průměrováním z 25 snímků**



**Obrázek 6: Model získaný průměrováním ze 100 snímků**



### 2.3.2 Model klouzavým průměrem

Dalším způsobem získání modelu prostředí je výpočet klouzavým průměrem. Stejně jako v předchozím případě je model vytvořen klouzavým průměrem ze 25/100 snímků. Nejprve je načten první snímek do modelu prostředí. Hodnota jasové úrovně každého pixelu modelu prostředí je porovnána s hodnotou jasové úrovně následujícího snímku. Pokud je hodnota jasové úrovně daného pixelu modelu prostředí nižší, je zvýšena. Naopak pokud je hodnota jasové úrovně daného pixelu modelu prostředí vyšší, je snížena. Model získaný klouzavým průměrem z 25 snímků je vidět na obrázku 7. Model získaný klouzavým průměrem ze 100 snímků je vidět na obrázku 8.



**Obrázek 7: Model získaný klouzavým průměrem z 25 snímků**



**Obrázek 8: Model získaný klouzavým průměrem ze 100 snímků**

### 2.3.3 Model mediánem

Výpočet modelu prostředí pomocí mediánu je výpočetně nejnáročnější ze všech tří testovaných metod. Stejně jako v předchozích případech je model vytvořen mediánem ze 25/100 snímků. Nejprve je načteno a uloženo 25/100 snímků do paměti. Hodnota jasové úrovně každého pixelu modelu prostředí je získána mediánem hodnot jasových úrovní daného pixelu všech 25/100 uložených snímků. Model získaný mediánem z 25 snímků je vidět na obrázku 9. Model získaný mediánem ze 100 snímků je vidět na obrázku 10.



**Obrázek 9: Model získaný mediánem z 25 snímků**



**Obrázek 10: Model získaný mediánem ze 100 snímků**

### 2.3.4 Shrnutí metod pro výpočet modelu prostředí

Nejjednodušším způsobem výpočtu modelu prostředí je výpočet průměrováním. Tento způsob je také výpočetně nejméně náročný. Model získaný průměrem ze 100 snímků docela dobře odpovídá realitě, v místech jízdních pruhů je však ovlivněn projíždějícími vozidly. Největší nevýhodou takto získaného modelu je jeho snadná ovlivnitelnost extrémně tmavými nebo světlými snímky, které se mohou vyskytnout vlivem špatně určené clony kamery. Aktualizace modelu získaného průměrováním je poměrně jednoduchá, aktuální snímek se s určitou váhou započítá do modelu.

Výpočet modelu prostředí klouzavým průměrem je oproti metodě průměrování výpočetně náročnější. Model je bohužel velmi ovlivněn prvním snímkem, který je zpočátku považován za model. Model získaný klouzavým průměrem z 25 snímků ještě stále vykazuje známky prvního snímku. Model získaný klouzavým průměrem ze 100 snímků je o poznání kvalitnější, ale stále nedosahuje kvality modelu získaného průměrováním ze stejného počtu snímků. Model získaný klouzavým průměrem z více než 200 snímků je však kvalitní a je i poměrně odolný proti extrémně světlým nebo tmavým snímkům. Aktualizace modelu klouzavým průměrem se provádí stejným způsobem jako jeho vytvoření, je však náročnější než aktualizace modelu získaného průměrováním.

Výpočet modelu prostředí mediánem je několikanásobně složitější než předchozí dvě metody a je výpočetně velice náročný. Model je však velice kvalitní již při výpočtu z 25 snímků, v místech jízdních pruhů není ovlivněn projíždějícími vozidly a je velice odolný proti extrémně tmavým nebo světlým snímkům. Aktualizace modelu získaného mediánem je poměrně snadná, stačí v určitých časových intervalech postupně nahrazovat uložené snímky, ze kterých je model prostředí počítán.

## 2.4 HLEDÁNÍ OBJEKTŮ

Další částí práce je vyhledávání a označení pohybujících se objektů ve snímané scéně. U některých metod bude k vyhledávání pohybujících se objektů využít v předchozí kapitole popsany model prostředí (získaný mediánem z 25 snímků). Bude zde popsáno několik metod pro hledání objektů. Vyhledávání pohybujících se objektů pomocí rozdílu mezi aktuálním snímkem a modelem prostředí, vyhledávání objektů hledáním hran v aktuálním snímku a v modelu prostředí, vyhledávání objektů pomocí detekce rohů a vyhledávání objektů metodou optického toku.

### 2.4.1 Hledání objektů pomocí rozdílu

Nejjednodušší metodou vyhledávání pohybujících se objektů ve snímané scéně je vytvoření rozdílu mezi aktuálním snímkem, který je zobrazen na obrázku 11 a modelem prostředí zobrazeném na obrázku 12.



**Obrázek 11: Aktuální snímek**



**Obrázek 12: Model získaný mediánem z 25 snímků**

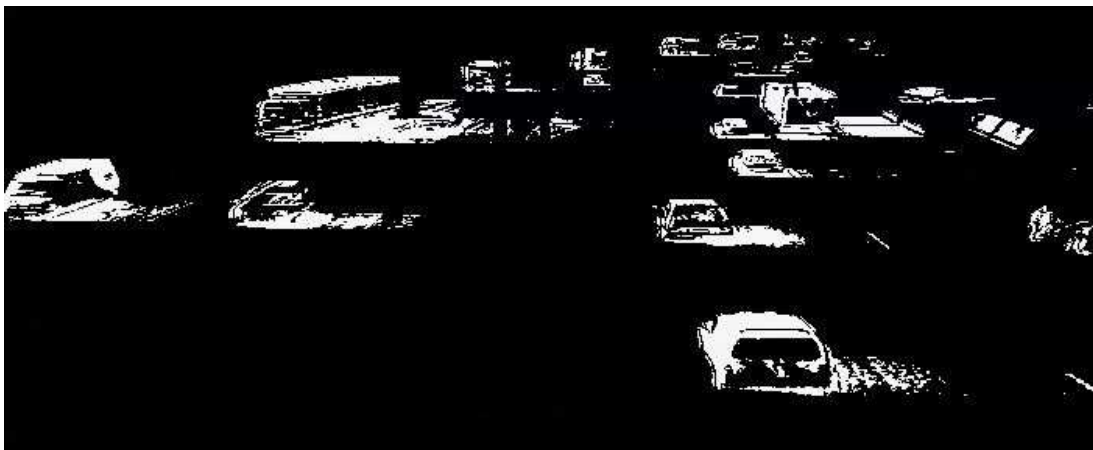
Provedeme rozdíl mezi aktuálním snímkem a modelem prostředí, poté rozdíl mezi modelem prostředí a aktuálním snímkem, a tyto dva rozdílové snímky sečteme. Výsledný rozdíl aktuálního snímku a modelu je ukázán na obrázku 13. Zachycuje pohybující se objekty, které se vyskytují v aktuálním snímku. Jak je vidět na obrázku 13, velkou roli zde hrají stíny pohybujících se objektů, které lze jen obtížně eliminovat.



**Obrázek 13: Rozdíl mezi aktuálním snímkem a modelem**

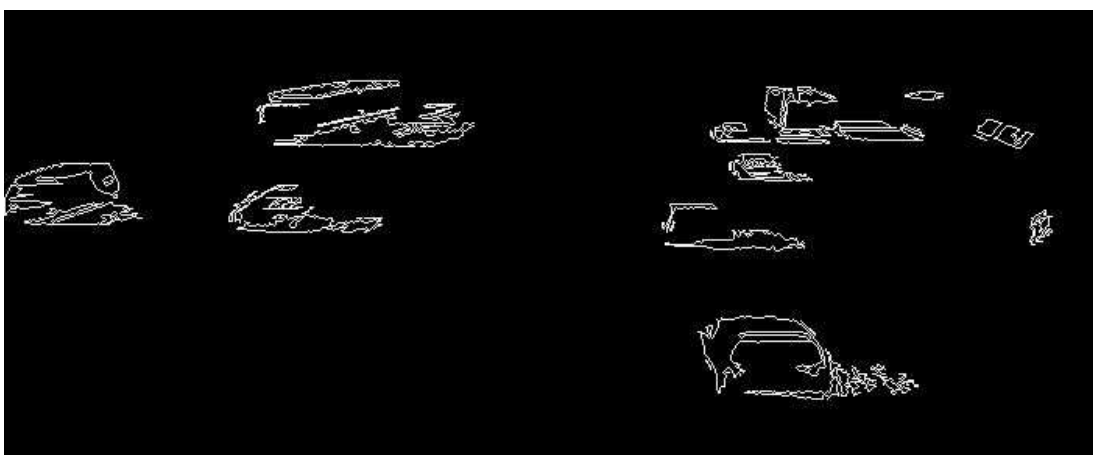
Rozdíl mezi aktuálním snímkem a modelem ve stupních šedi je pak převeden na binární obraz. U této operace se dá měnit práh. Ten určuje, které odstíny ještě budou považovány za bílou a které už budou považovány za černou. Jenže ani pomocí prahu se nedají stíny úplně eliminovat, protože jsou mnohdy výraznější než

samotné objekty. Rozdíl mezi aktuálním snímkem a modelem reprezentovaný binárně je vidět na obrázku 14.



**Obrázek 14: Rozdíl mezi aktuálním snímkem a modelem binárně**

Snímek na obrázku číslo 14 je ještě dále upravován odstraněním šumu (malých bílých oblastí) a poté je vybrán pouze obrys velkých objektů. Výsledek těchto operací je vidět na obrázku 15.



**Obrázek 15: Obrys nalezených objektů**



**Obrázek 16: Výsledek – objekty označené pomocí rozdílů v akt. bar. snímku**

Na obrázku 16 je výsledek – označené objekty v aktuálním barevném snímku. Je také vidět již zmíněný problém se stíny objektů. Následkem prahu zvýšeného kvůli alespoň částečné eliminaci stínů potom nejsou označeny některé světlé části objektů. Dalším zvyšováním prahu by se sice dosáhlo eliminace stínů, ovšem nebylo by označeno ještě více světlých objektů.

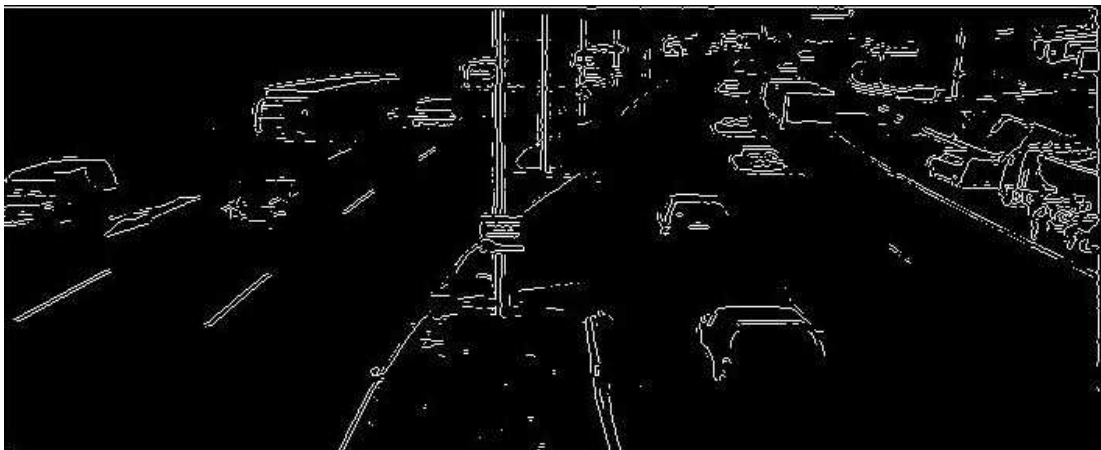
### 2.4.2 Hledání objektů pomocí hran

Další metodou vyhledávání pohybujících se objektů ve snímané scéně je detekce hran. Vyhledají se hrany v aktuálním snímku a v modelu prostředí. Rozdílem nalezených hran v aktuálním snímku a v modelu prostředí najdeme pohybující se objekty. K vysvětlení je použit stejný aktuální snímek (obrázek 11) i stejný model prostředí (obrázek 12) jako v předchozím případě. K detekci hran jsem využil Prewittova filtru, hrany jsou hledány v horizontálním i vertikálním směru. Prewittův hranový detektor využívá konvoluční masky na obrázku 17.

$$\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$

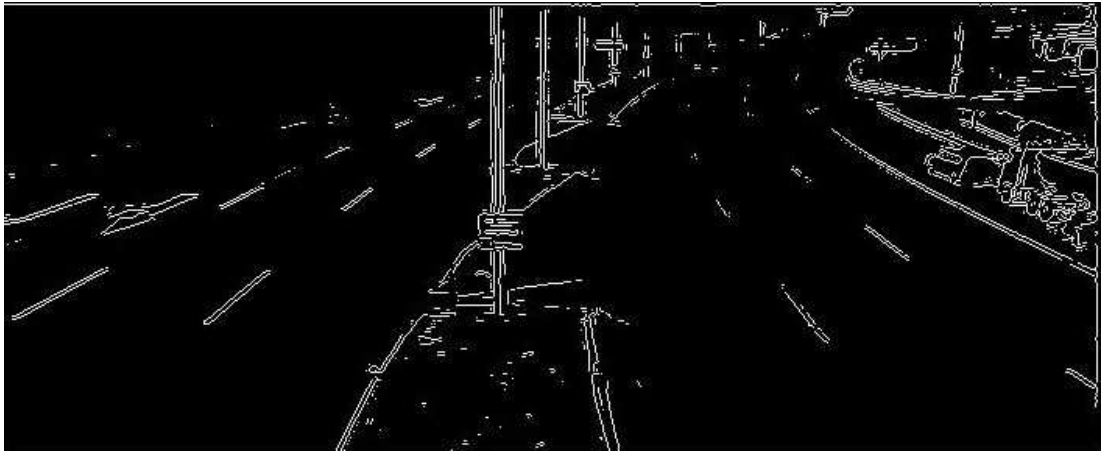
**Obrázek 17: Masky Prewittova hranového detektoru [4]**

Na obrázku 18 je vidět aktuální snímek po aplikaci Prewittova hranového detektoru. Na obrázku 19 je vidět model prostředí po aplikaci Prewittova hranového detektoru.



**Obrázek 18: Hrany nalezené v aktuálním snímku**





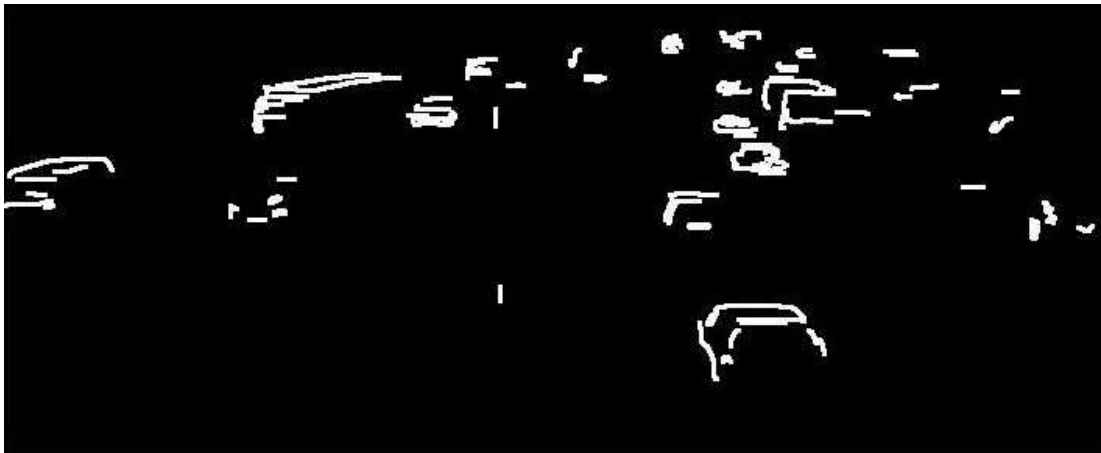
**Obrázek 19: Hraný nalezené v modelu prostředí**

Nyní provedeme rozdíl mezi hranami nalezenými v aktuálním snímku a hranami nalezenými v modelu prostředí. Výsledek této operace je na obrázku 20. Nesmí se provést absolutní hodnota rozdílu, protože pak by vznikaly problémy s vodorovnými dopravními značeními ve chvíli, kdy je zakrývají vozidla.



**Obrázek 20: Rozdíl mezi obrázkem 18 a obrázkem 19**

Snímek na obrázku 20 je ještě dále upravován odstraněním šumu (malých bílých oblastí, zbytků vodorovného dopravního značení) a poté jsou zvýrazněny (dilatace) obrysy vozidel. Výsledek těchto operací je vidět na obrázku 21.



**Obrázek 21: Obrys nalezených objektů pomocí detekce hran**

Na obrázku 22 je vidět výsledek – označené objekty v aktuálním barevném snímku.



**Obrázek 22: Výsledek – objekty označené detekcí hran v akt. bar. snímku**

### 2.4.3 Hledání objektů pomocí kombinace předchozích dvou metod

Ještě lepšího výsledku můžeme dosáhnout, pokud použijeme obě metody pro hledání objektů. Provedeme hledání objektů pomocí detekce hran (obrázek 21). Zároveň provedeme hledání objektů pomocí rozdílu mezi aktuálním snímkem a modelem prostředí, ale rozdílový snímek ve stupních šedi budeme na binární obraz převádět s vyšším prahem. Prah musí být tak vysoký, aby eliminoval stíny (eliminuje i světlé objekty), ale tmavé objekty ještě zůstaly nad tímto prahem a tedy zůstanou označeny. Výsledek převodu rozdílového snímku ve stupních šedi na binární obraz s vysokým prahem je vidět na obrázku 23.



**Obrázek 23: Rozdíl mezi aktuálním snímkem a modelem binárně s vysokým prahem**

Pokud zkombinujeme nalezené objekty pomocí detekce hran (obrázek 21) a objekty nalezené pomocí rozdílu mezi aktuálním snímkem a modelem (obrázek 23) dostaneme se k výsledku, který je vidět na obrázku 24.



**Obrázek 24: Obrys nalezených objektů pomocí kombinace obou metod**

U snímku na obrázku 2, který jsem zvolil do této dokumentace jako exemplární není výsledek získaný kombinací obou metod pro hledání objektů nijak oslnivý, ale u některých dalších snímků, které jsem měl k dispozici je touto kombinovanou metodou dosaženo opravdu nejlepších výsledků. Ještě lepších výsledků lze pomocí této metody dosáhnout u snímků, na kterých jsou vlivem jiných klimatických podmínek stíny objektů méně výrazné. Označené objekty pomocí kombinace obou metod jsou vidět na obrázku 25.



**Obrázek 25: Výsledek – objekty označené kombinací obou metod v akt. bar. snímku**

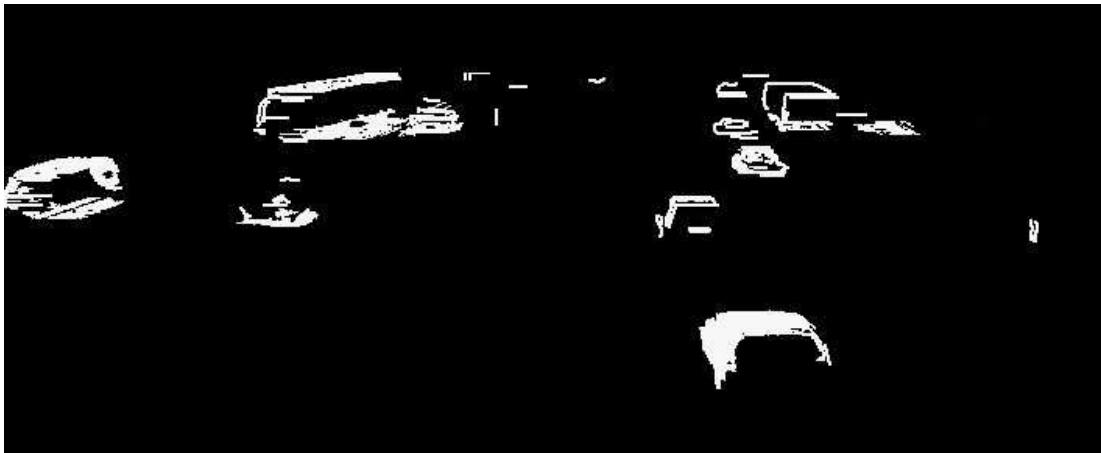
#### 2.4.4 Hledání objektů pouze v jízdních pruzích

Dalším způsobem, jak zlepšit výsledky, je omezit oblasti, ve kterých se budou objekty vyhledávat. To provedeme pomocí ručně vytvořené binární mapy snímané scény, která je vidět na obrázku 26. Touto binární mapou vynásobíme binární obrys nalezených objektů (obrázek 24). Objekty se potom budou označovat pouze v oblastech, které jsou v této mapě bílé. Objekty, nacházející se v oblastech, které jsou na této mapě černé, nebudou označeny.



Obrázek 26: Binární mapa snímané scény

Na obrázku 27 je především v horní části vidět rozdíl oproti obrázku 24. Samozřejmě by se změnou binární mapy snímané scény (obrázek 26) dalo eliminovat označování objektů v odstavných, odbočovacích či připojovacích pruzích apod.



**Obrázek 27: Obrys nalezených objektů v jízdnicích**

Označené objekty pomocí kombinace obou metod pouze v jízdnicích jsou vidět na obrázku 28. Je zde opět vidět rozdíl oproti obrázku 25.



**Obrázek 28: Výsledek – objekty označené pouze v jízdnicích**

### 2.4.5 Hledání objektů pomocí detekce rohů

Jiný přístup pro hledání objektů ve snímané scéně je detekce rohů. Tato metoda spočívá ve vyhledávání významných bodů pohybujících se objektů. Snažíme se tedy najít významná místa v obraze – musí být co nejméně podobná svému okolí, představují hranice či vrcholy objektů [7]. Významná místa v obraze můžeme nalézt pomocí lokálního detektoru. Do detektoru vstupuje šedotónový snímek (obrázek 4) a vystupuje z něj obraz, ve kterém jsou hodnoty pixelů úměrné pravděpodobnosti, že daný pixel je roh [4]. Významné body jsou získány prahováním obrazu získaného detektorem. Roh v obraze může představovat pixel, v jehož blízkém okolí jsou dvě různé hrany – takové detektory jsou také nazývány detektory rohů a jsou hodně používané. Detektory rohů obvykle nejsou moc robustní, většinou je totiž v obraze nalezeno mnohem více rohů, než je potřeba k další práci s obrazem.

Nejjednodušší detektor rohů je Moravcův detektor (1977), který má maximální hodnoty v pixelech s vysokým kontrastem. Takové body jsou v rozích a ostrých hranách. O něco lepší výsledky jsou dosaženy výpočetně mnohem náročnějšími detektory rohů, např. Zuniga-Haralick (1983), Haralick and Shapiro (1992).

Harrisův detektor rohů (1988) je vylepšený Moravcův detektor. Harrisův detektor uvažuje diferenci skóre rohu (suma kvadrátu diferencí). Algoritmus Harrisova detektoru [4]:

1. Filtrace obrazu Gaussovým filtrem.
2. Odhad intenzity gradientu ve dvou vzájemně kolmých směrech pro každý pixel  $\frac{\partial f(x, y)}{\partial x}, \frac{\partial f(x, y)}{\partial y}$ .
3. Pro každý pixel a jeho okolí:  
Vypočet lokální struktury matice A.  
Vyhodnocení odezvy funkce R(A).
4. Výběr nejlepších kandidátů na rohy v obraze.

Výhodou Harrisova detektoru je necitlivost na rotaci, na malé výchylky osvětlení, na malou změnu úhlu pohledu a malá výpočetní náročnost. Na druhou stranu není invariantní proti větší změně měřítko, proti větší změně úhlu pohledu a proti výrazným změnám kontrastu. Harrisův detektor je velmi populární metodou pro hledání významných bodů v obraze a našel si uplatnění v mnoha algoritmech počítačového vidění [5].

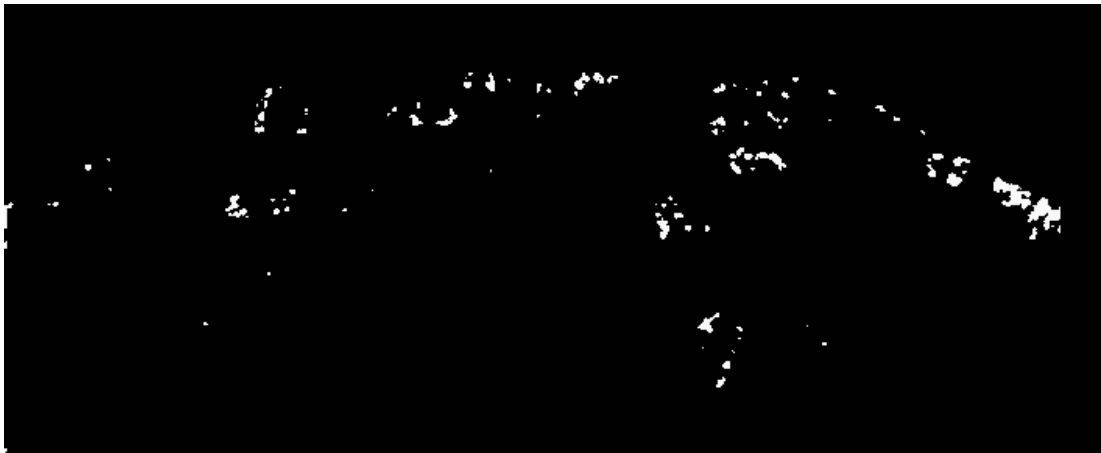
Pro výpočet významných bodů v obraze používám funkci `harris.m` [6]. Vstupem funkce `harris.m` je šedotónový obraz (obrázek 4) a výstupem je šedotónový obraz (obrázek 29), ve kterém jsou významné pixely vstupního obrazu zvýrazněny vysokou hodnotou jasu.



**Obrázek 29: Výstup z Harrisova detektoru významných bodů**

Výstup z Harrisova detektoru je převeden na binární obraz s ručně nastaveným prahem. Binární obraz je vidět na obrázku 30.





**Obrázek 30: Významné body nalezené v aktuálním snímku binárně**

Po přenesení významných bodů do barevného snímku (obrázek 31) vidíme, že většina významných bodů jsou pohybující se objekty. Ostatní významné body v aktuálním snímku (obrázek 30) bychom mohli eliminovat například vyhledáním významných bodů v modelu prostředí (obrázek 12) a následným odečtením. Významné body nalezené v aktuálním snímku (obrázek 30) můžeme vynásobit binární mapou snímané scény (obrázek 26), čímž odstraníme významné body, které nejsou v jízdnicích pruzích.



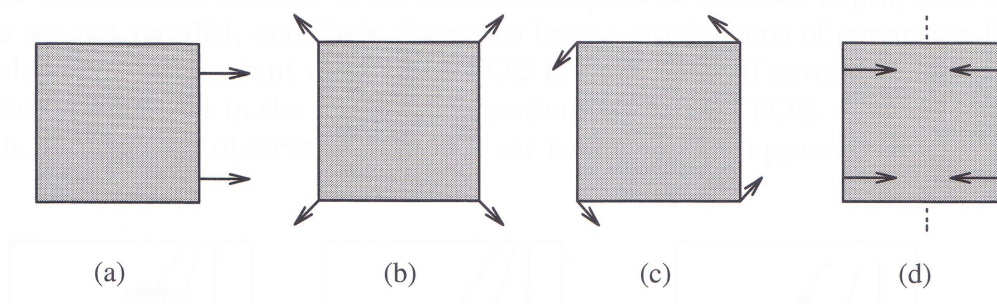
**Obrázek 31: Výsledek – označená významná místa ve snímku**

### 2.4.6 Hledání objektů metodou optického toku

Teoretické informace a vztahy týkající se optického toku jsou převzaty z [4].

Změny v obraze způsobené pohybem můžeme zjišťovat pomocí optického toku. Podobným způsobem provádějí analýzu dynamických obrazů biologické vizuální systémy. Optický tok ukazuje všechny změny v obraze za časový interval  $dt$ . Díky metodě optického toku můžeme pohyb objektu v trojrozměrné scéně přenést do dvojrozměrného obrazu. Každému bodu obrazu odpovídá dvojrozměrný vektor rychlosti vypovídající o směru a velikosti rychlosti pohybu v daném místě obrazu. Optický tok můžeme využít v situaci, kdy je scéna statická a pozorovatel se pohybuje, scéna se pohybuje a pozorovatel je statický, i v případě současného pohybu scény i pozorovatele. Máme čtyři základní druhy pohybu, které se vyskytují v dynamických obrazech:

- translační pohyb v konstantní vzdálenosti
- translační pohyb do dálky
- rotační pohyb v konstantní vzdálenosti (rotace kolem osy pohledu)
- rotační pohyb kolmý na osu pohledu



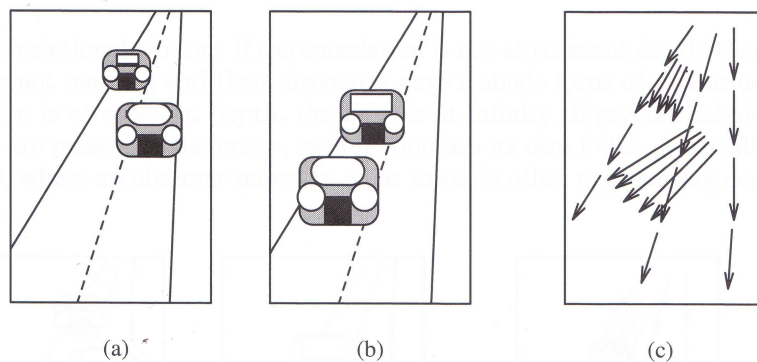
**Obrázek 32: Rozdělení druhů pohybu. a) translační pohyb v konstantní vzdálenosti, b) translační pohyb do dálky, c) rotační pohyb v konstantní vzdálenosti, d) rotační pohyb kolmý na osu pohledu [4]**

Všechny tyto základní druhy pohybu od sebe můžeme odlišit aplikací jednoduchých diferenciálních operátorů na optický tok. Využíváme několika základních vlastností – translační pohyb v konstantní vzdálenosti se v optickém toku

projeví soustavou rovnoběžných vektorů toku, translační pohyb do dálky způsobí vytvoření skupiny vektorů toku vycházejících z jednoho bodu (to je případ naší úlohy a budeme se tím dále zabývat), rotace kolem osy pohledu se projeví soustředným uspořádáním vektorů optického toku, rotace kolmá na osu pohledu vytvoří vektory toku vycházející z přímky. Pohyby v dynamickém obrazu můžeme dále podrobněji dělit, což ovšem přináší výrazné zvýšení náročnosti analýzy.

#### 2.4.6.1 Translační pohyb

V případě, že translační pohyb neprobíhá v konstantní vzdálenosti (obr. 32 a), ale probíhá do dálky (obrázek 32 b), jsou vektory optického toku různoběžné a jejich směry vycházejí z jednoho bodu v obrazu (OE – ohnisko expanze). Pokud se jedná o translační pohyb v konstantní vzdálenosti, pak jsou vektory optického toku rovnoběžné a OE leží v nekonečnu. Pokud máme v dynamickém obrazu více pohybujících se objektů, pak každý má své vlastní OE – to je znázorněno na obrázku 33, kde se pozorovatel pohybuje směrem k vozidlům, z nichž jedno vozidlo objíždí druhé.



Obrázek 33: Ohnisko expanze. a) čas  $t_1$ , b) čas  $t_2$ , c) optický tok [4]

#### 2.4.6.2 Vzájemná rychlost

Vzájemnou rychlost  $c$  pozorovatele a objektu reprezentovaného jedním bodem v obrazu můžeme zjistit z optického toku. Vzájemné rychlosti ve směrech

$x, y, z$  jsou  $c_x = u, c_y = v, c_z = w$ , kde  $z$  udává informaci o vzdálenosti (dálce). Abychom odlišili souřadnice odečítané ve dvojrozměrném obraze od souřadnic reálného světa, označíme v tomto odstavci souřadnice obrazu  $x', y'$ . Z vlastností perspektivy, pokud poloha bodu v čase  $t_0$  je  $x_0, y_0, z_0$ , pak polohu tohoto bodu v čase  $t$ , s uvažováním jednotkové ohniskové vzdálenosti snímacího zařízení a konstantní rychlostí, můžeme vyjádřit takto:

$$(x', y') = \left( \frac{x_0 + ut}{z_0 + wt}, \frac{y_0 + vt}{z_0 + wt} \right). \quad \text{Rovnice 1}$$

#### 2.4.6.3 Ohnisko expanze

Z rovnice 1 můžeme určit polohu ohniska expanze (OE) ve dvojrozměrném obraze. Předpokládejme pohyb směrem k pozorovateli, pro čas  $t \rightarrow -\infty$ , pohyb můžeme sledovat zpět k bodu vzniku:

$$x'_{OE} = \left( \frac{u}{w}, \frac{v}{w} \right). \quad \text{Rovnice 2}$$

Stejná rovnice může být použita i pro čas  $t \rightarrow \infty$  a pohyb v obráceném směru. Jednoduše můžeme říct, že jakákoliv změna směru pohybu způsobí změny rychlostí  $u, v, w$  a OE změní svoji polohu v obraze.

#### 2.4.6.4 Určení vzdálenosti (dálky)

Díky přítomnosti souřadnice  $z$  v rovnici 1, může být optický tok využit ke stanovení aktuální vzdálenosti pohybujícího se objektu od pozorovatele. Informace o této vzdálenosti je v rovnici 1 nepřímo obsažena, k jejímu přesnému určení musíme znát alespoň jednu hodnotu vzdálenosti přesně. Pokud  $D(t)$  je vzdálenost bodu od OE (měřeno v dvojrozměrném obraze) a  $V(t)$  je jeho rychlost  $dD/dt$ , pak mezi těmito veličinami a parametry optického toku platí vztah:

$$\frac{D(t)}{V(t)} = \frac{z(t)}{w(t)}. \quad \text{Rovnice 3}$$

Za předpokladu pohybu objektu směrem k pozorovateli udává poměr  $z/w$  čas, za který objekt pohybující se konstantní rychlostí  $w$  protne rovinu obrazu. Pokud známe vzdálenost kteréhokoliv bodu v obraze pohybujícího se rychlostí  $w$  ve směru osy  $z$ , můžeme vypočítat vzdálenosti všech ostatních bodů v obraze pohybujících se stejnou rychlostí  $w$  podle vztahu:

$$z_2 = \frac{z_1(t)V_1(t)D_2(t)}{D_1(t)V_2(t)}, \quad \text{Rovnice 4}$$

kde  $z_1(t)$  je známá vzdálenost a  $z_2(t)$  je neznámá vzdálenost. Pomocí těchto vztahů můžeme určit polohu bodů reálného světa v souřadnicích  $x, y$  na základě souřadnic  $x', y'$  odečítaných v obraze – relativně vůči poloze a rychlosti pohybu pozorovatele:

$$x(t) = \frac{x'(t)w(t)D(t)}{V(t)} \quad y(t) = \frac{y'(t)w(t)D(t)}{V(t)} \quad z(t) = \frac{w(t)D(t)}{V(t)} \quad \text{Rovnice 5}$$

#### 2.4.6.5 Výsledky hledání objektů metodou optického toku

Pro výpočet optického toku v dynamickém obraze používám funkci `optical_flow.m` [8]. Vstupem funkce `optical_flow.m` je několik šedotónových obrazů (v tomto případě pět snímků dopravní scény pořízených po sobě v intervalu cca 0,2 s) a výstupem jsou dvě matice s vypočítaným optickým tokem pro každý pixel. Pro reprezentaci výsledných vektorů optického toku používám funkci `vis_flow.m` [8]. Na základě vstupu (dvě matice s vypočítaným optickým tokem) funkce `vis_flow.m` vykreslí vektory optického toku do prostředního z pěti snímků, které na počátku vstupovaly do funkce `optical_flow.m`. Snímek s naznačenými vektory optického toku můžeme vidět na obrázcích 34 a 35.



**Obrázek 34: Výsledek - optický tok**



**Obrázek 35: Výsledek – optický tok**

Jak je vidět z obrázků 34 a 35, optický tok dokáže zachytit pohyb v dynamické scéně. Z vektorů optického toku můžeme poměrně snadno poznat, jakým směrem se každý objekt pohybuje a z délky vektorů je vidět, že některé objekty se pohybují rychleji, některé pomaleji. Nevýhodou je poměrně vysoká výpočetní náročnost. Další nevýhodou je možnost aplikovat metodu optického toku jen na sekvence snímků s velmi krátkými časovými intervaly mezi snímky. Tento problém je vidět na obrázku 34. I když se vozidlo v reálném prostředí pohybuje konstantní rychlostí, v našich dvojrozměrných snímcích se pohybuje ve spodní části snímku mnohem rychleji, než v horní části snímku. A právě s rychlým pohybem vozidel ve spodní části snímků má metoda optického toku problémy a nedokáže tento rychlý pohyb zachytit.

### 2.4.7 Shrnutí metod pro hledání objektů

Nejjednodušší metodou vyhledávání pohybujících se objektů ve snímané scéně je vytvoření rozdílu mezi aktuálním snímkem (součet rozdílu mezi aktuálním snímkem a modelem prostředí s rozdílem mezi modelem prostředí a aktuálním snímkem). Zachycuje pohybující se objekty, které se vyskytují v aktuálním snímku. Velkou roli zde hrají stíny pohybujících se objektů, které lze jen částečně eliminovat převodem tohoto šedotónového rozdílu na binární obraz. Výslední obrys velkých objektů v aktuálním snímku získaný rozdílovou metodou je vidět na obrázku 15.

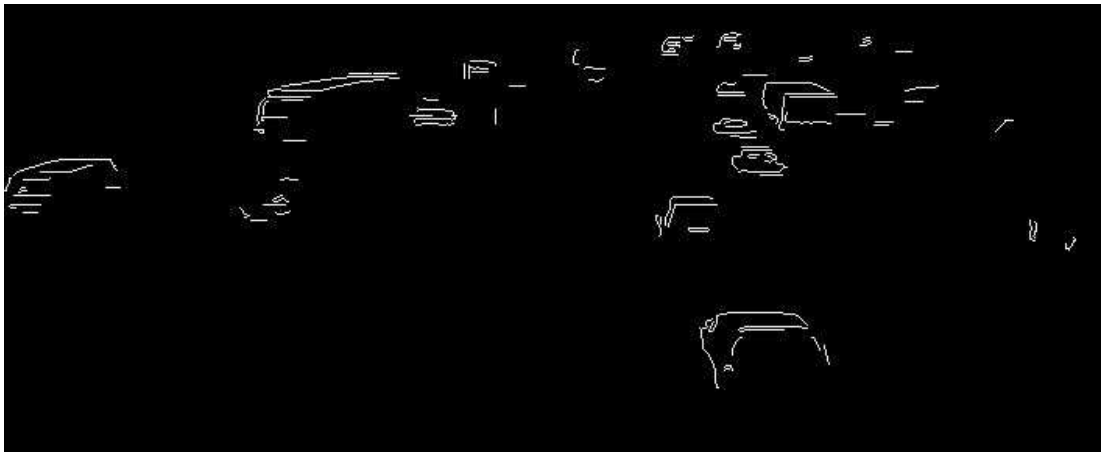
Další metodou vyhledávání pohybujících se objektů ve snímané scéně je detekce hran (rozdíl hran vyhledaných v aktuálním snímku a v modelu prostředí). Obrisy objektů označených pomocí detekce hran můžeme vidět na obrázku 21. Ještě lepších výsledků můžeme dosáhnout, pokud tuto metodu zkombinujeme s hledáním objektů rozdílovou metodou (obrázek 24).

Jiný přístup pro hledání objektů ve snímané scéně je detekce rohů (vyhledávání významných bodů v obraze, které obvykle představují hranice či vrcholy objektů). Když porovnáme nalezené významné body v obraze (obrázek 30) s aktuálním snímkem, vidíme, že většina nalezených významných bodů jsou právě pohybující se objekty.

Poslední vyzkoušená metoda pro hledání pohybujících se objektů je optický tok (hledáme dvojrozměrné vektory rychlosti vypovídající o směru a velikosti rychlosti pohybu v daném místě obrazu). Z pěti šedotónových snímků dopravní scény pořízených po sobě jsou vypočítány matice s optickým tokem pro každý pixel. Poté jsou získané vektory optického toku vykresleny do prostředního z pěti snímků, viz obrázky 34 a 35. Z vektorů optického toku můžeme poznat, jakým směrem se každý objekt pohybuje a z délky vektorů lze odhadnout vzájemnou rychlost objektů. Nevýhodou je poměrně vysoká výpočetní náročnost. Další nevýhodou je možnost aplikovat metodu optického toku jen na sekvence snímků s velmi krátkými časovými intervaly mezi snímky.

## 2.5 OZNAČENÍ OBJEKTŮ

Z vyzkoušených metod pro hledání pohybujících se objektů jsem pro tento účel vybral metodu detekce hran. Na této metodě ukážu, jak jsem postupoval při výsledném označení pohybujících se objektů. Po provedení rozdílu mezi hranami nalezenými v aktuálním snímku a v modelu prostředí získáme hrany pohybujících se objektů, viz obrázek 36.



**Obrázek 36: Hrany pohybujících se objektů pomocí detekce hran**

Obrys každého objektu však není zvýrazněn jednou dlouhou nepřerušenou linkou, ale několika kratšími od sebe oddělenými linkami. Abych mohl jednotlivé objekty označit, musím vždy všechny linky, patřící k jednomu objektu, sloučit dohromady a vytvořit jednu ucelenou oblast. Sloučení jsem se snažil dosáhnout pomocí matematické morfologie.

Nejprve jsem na počáteční snímek s hranami aplikoval morfologickou transformaci dilatace. Výsledek dilatace závisí na vstupním obraze a strukturním elementu. Ke každému bílému bodu v binárním vstupním obraze je přičten strukturní element. Tím dochází k rozšiřování bílých oblastí ve vstupním obraze. Velikost a způsob rozšíření bílých oblastí závisí na velikosti a tvaru strukturního elementu. Při použití dostatečně velkého strukturního elementu dochází ke slévání bílých oblastí ve vstupním obraze a právě toho potřebuji dosáhnout. Na horní polovinu vstupního



obrazu (obrázek 36) jsem aplikoval strukturní element ve tvaru čtverce o délce strany 4 pixely. Na spodní polovinu vstupního obrazu jsem zvolil element ve tvaru čtverce s délkou strany 20 pixelů. Výsledek po aplikaci dilatace na vstupní obraz je vidět na obrázku 37.



**Obrázek 37: Dilatace hran nalezených objektů**

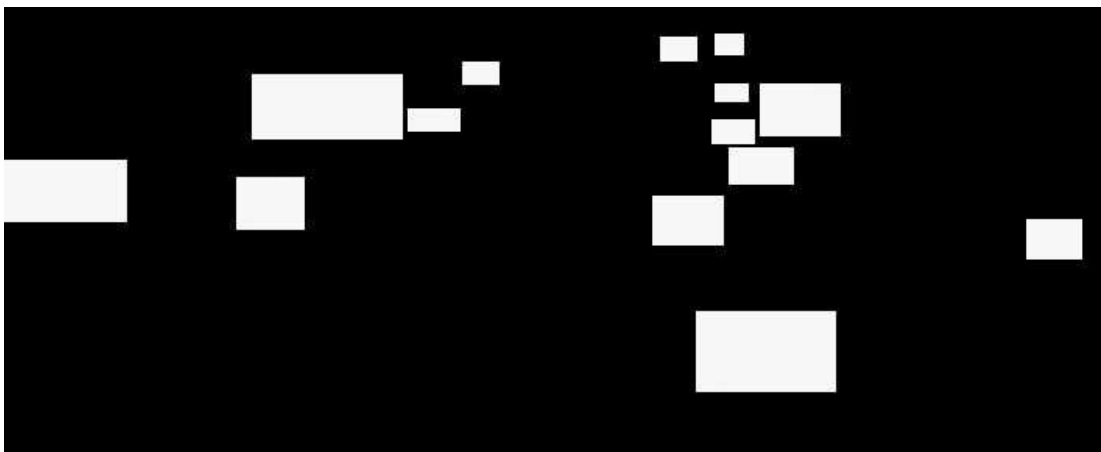
Kdybych na celý vstupní snímek použil dilataci se stejně velkým strukturním elementem, došlo by v horní části snímku, kde jsou objekty blízko u sebe, ke sloučení několika objektů v jeden útvar. Proto jsem musel v horní části použít menší strukturní element. Dilatací jsem dokázal sloučit jednotlivé linky daného objektu v ucelenou oblast. Zároveň také došlo ke zvětšení a roztažení označených objektů, což je nežádoucí důsledek dilatace, který ovšem lze částečně eliminovat další morfologickou transformací.

Tato morfologická transformace se nazývá eroze. Eroze, stejně jako dilatace, závisí na vstupním obraze a strukturním elementu. Erozí tedy dokážu zmenšit označené objekty. Na horní část vstupního obrazu (obrázek 37) jsem aplikoval erozi se strukturním elementem ve tvaru čtverce o délce strany 2 pixely. Na spodní část vstupního obrazu jsem použil strukturní element ve tvaru čtverce s délkou strany 15 pixelů. Výsledek po aplikaci eroze na vstupní obraz je vidět na obrázku 38.



**Obrázek 38: Dilatace a následná eroze nalezených objektů**

Opět jsem musel použít různé strukturní elementy na horní a dolní část vstupního snímku, protože jinak by došlo k nežádoucím transformacím označených objektů v obraze. Erozí jsme tedy zmenšili označené objekty, dále odstraníme malé objekty, tj. hrany, které se nespojily s většími objekty. A kolem všech zbylých objektů vytvoříme obdélníky. Výsledek těchto operací můžeme vidět na obrázku 39.



**Obrázek 39: Obdélníky kolem objektů**

Z obdélníků kolem objektů na obrázku 39 vybereme pouze jejich obvodové hrany, poté aplikujeme morfologickou transformaci dilataci (strukturní element ve tvaru čtverce o délce strany 2 pixely) kvůli lepší viditelnosti označených objektů v aktuálním snímku. Takto upravený snímek, na kterém jsou pouze výrazné obvodové hrany obdélníků použijeme na zvýraznění objektů v aktuálním snímku. Výsledek můžeme vidět na obrázku 40.



**Obrázek 40: Označené objekty v aktuálním snímku**

Abychom eliminovali označování objektů v nežádoucích oblastech, můžeme do předchozího postupu zpracovat ručně vytvořenou binární mapu snímané scény, která je vidět na obrázku 26. Touto binární mapou vynásobíme nalezené objekty na obrázku 38 a tím odstraníme označení objektů mimo jízdní pruhy.

## 2.6 NOČNÍ SNÍMKY

Další úloha, kterou jsem se zabýval, je hledání pohybujících se objektů v nočních snímcích. Opět se jedná o sekvenci na sebe navazujících snímků pořízených dopravní kamerou v noci za umělého osvětlení. Ukázka takového nočního snímku je vidět na obrázku 41.



**Obrázek 41: Noční snímek z dopravní kamery**

Jak je z obrázku 41 na první pohled vidět, nejvýraznějšími body na pohybujících se vozidlech jsou jejich světlomety. Světlo vycházející z předních světlometů vozidel vytváří silné odlesky na vozovce. Tyto odlesky způsobují problémy při detekci pohybujících se objektů, podobně jako tomu bylo se stíny objektů na snímcích za denního osvětlení. Pohybující se objekty v nočních snímcích budeme detekovat podobnými metodami, jako objekty v denních snímcích.

K těmto metodám budeme opět potřebovat model snímané noční scény. Každý barevný noční snímek z dopravní kamery převedeme na šedotónový snímek a ten budeme dále zpracovávat. Model snímané noční scény vytvoříme mediánem z 25 šedotónových snímků. Takto získaný model můžeme vidět na obrázku 42.



**Obrázek 42: Model snímané noční scény získaný mediánem**

Při pokusu o vyhledávání objektů rozdílou metodou jsem narazil na problém s odlesky ve vozovce, které jsou vytvořeny předními světly vozidel. Tyto odlesky mají mnohdy vyšší hodnotu jasu než samotné automobily a kvůli tomu jsou chybně vyhodnocovány jako objekty ve snímané scéně. Rozdílová metoda se tedy ukázala jako velmi nevhodná. Stejným způsobem pracovala i metoda vyhledávání významných bodů v obraze, protože odlesky ve vozovce byly často vyhodnoceny jako významné body a to opět způsobovalo chyby při označování objektů.

Nejlepších výsledků dosahovala metoda vyhledávání objektů detekcí hran. Po provedení rozdílu mezi hranami nalezenými v aktuálním snímku a v modelu prostředí získáme hrany pohybujících se objektů, viz obrázek 43. Hranový detektor nalezne hrany především v okolí světlometů, protože ty jsou mnohem výraznější než siluety pohybujících se objektů. Ale světlomety jsou součástí pohybujících se objektů, které chceme detekovat, takže této skutečnosti můžeme využít. Detektor občas označí i odlesky ve vozovce, ale oproti ostatním vyzkoušeným metodám se to neděje tak často. Navíc lze dalším zpracováním tyto detekované odlesky částečně eliminovat.



**Obrázek 43: Hrany objektů v nočním snímku**

Dalšími úpravami tohoto binárního snímku (viz kapitola 2.5) se dostaneme až k výslednému označení objektů v aktuálním nočním snímku, který můžeme vidět na obrázku 44.



**Obrázek 44: Označené objekty v aktuálním nočním snímku**

Na označeném vozidle vlevo nahoře je vidět problém s odlesky ve vozovce, které jsou spolu s vozidlem vyhodnoceny jako jeden objekt. Tuto chybu se můžeme pokusit eliminovat pomocí ekvalizace histogramu šedotónového aktuálního snímku a modelu prostředí. Ekvalizací histogramu daného snímku dosáhneme jiného rozložení jasových hodnot histogramu ve snímku, čímž by se měl zvýšit kontrast tohoto

snímku. Model snímané noční scény z obrázku 42 po ekvalizaci histogramu můžeme vidět na obrázku 45.



**Obrázek 45: Model snímané noční scény po ekvalizaci histogramu**

Totéž provedeme s aktuálním šedotónovým snímkem, poté vypočítáme rozdíl vyhledaných hran v tomto snímku a v modelu prostředí (obrázek 45). Tím získáme hrany pohybujících se objektů, provedeme další úpravy (viz kapitola 2.5) a získáme opět označené objekty v aktuálním barevném snímku na obrázku 46. Zde můžeme vidět rozdíl (proti obrázku 44) u označeného vozidla vlevo nahoře, které je označeno již bez odlesku ve vozovce.



**Obrázek 46: Označené objekty – ekvalizace histogramu**

V sekvenci nočních snímků se objevilo také jednostopé vozidlo. Je detekováno stejným algoritmem (s ekvalizací histogramu), který byl popsán výše. Výsledné označení jednostopého vozidla v nočním snímku je vidět na obrázku 47.



**Obrázek 47: Jednostopé vozidlo v nočním snímku**



## 2.7 POČÍTÁNÍ VOZIDEL

Jakmile máme vyzkoušené a vybrané metody pro vyhledávání vozidel, můžeme vozidla ve snímané scéně začít počítat. Nejprve si ve snímané scéně určíme, v kterých místech budeme vozidla počítat. V každém jízdním pruhu vybereme oblast, ve které budou vozidla počítána. Zvolené oblasti jsou znázorněny bíle v modelu snímané scény na obrázku 48.



**Obrázek 48: Oblasti pro počítání vozidel v jízdních pruzích**

Jakmile se v kterékoliv oblasti objeví detekované vozidlo, je započítáno. Pro tuto snímanou scénu jsem zvolil dvě počítadla. Jedno z nich počítá příjezdějící vozidla (vidíme jejich přední část) a druhé počítá odjíždějící vozidla (vidíme jejich zadní část).

## 2.8 POČÍTÁNÍ PRŮMĚRNÉ RYCHLOSTI

Dalším úkolem bylo počítat průměrnou rychlost jízdního pruhu. Pro zjištění průměrné rychlosti jízdního pruhu budeme muset znát rychlost každého vozidla v daném pruhu. Rychlost vozidla spočítáme podle vzdálenosti, kterou vozidlo urazilo mezi dvěma po sobě jdoucími snímky. Dopravní kamera, jejíž snímky mám k dispozici, zvládá vytvořit cca 5 snímků/s. V názvu každého snímku je vložena informace o datu a času pořízení snímku. Odtud lze zjistit, že každý snímek je pořízen o 0,2 s později než předchozí snímek. Časový interval mezi snímky však není vždy stejný, ve skutečnosti se pohybuje od 180 ms do 220 ms. Ve výpočtech průměrné rychlosti jízdního pruhu však bude počítáno vždy s časovým intervalem 0,2 s.

Druhou částí tohoto úkolu je určení vzdálenosti, kterou vozidlo urazí mezi dvěma snímky po sobě. Musíme tedy najít souvislost mezi vzdáleností ve snímku v pixelech a vzdáleností v reálném prostředí v metrech. Potřebujeme tedy transformovat body z kamery do reálného 3D prostoru. Když kamera převádí 3D prostor do 2D snímku, dochází ke ztrátě jedné souřadnice. Zpětný převod, který potřebujeme provést, tedy není jednoznačný, museli bychom znát některé parametry (ohnisková vzdálenost čočky kamery, poloha kamery). Proto jsem přistoupil k jiné metodě určení vzdálenosti, kterou vozidlo urazí. K ní musím znát alespoň určitou vzdálenost ve snímku v metrech. Ta je uprostřed obrázku 49 naznačena úsečkou L. K určení této vzdálenosti jsem využil znalosti délky vodorovného dopravního značení „podélná čára přerušovaná“ (délka čar 6 m, délka mezer 12 m) [9]. Pokud tedy známe délku úsečky L, můžeme pomocí mocninné funkce převádět vzdálenost ve snímku v pixelech na vzdálenost v reálném prostředí v metrech.



**Obrázek 49: Délka úsečky L**

Pro zjištění vzdálenosti, kterou vozidlo urazilo mezi dvěma snímky po sobě, jsem postupoval následujícím způsobem. V prvním snímku vyhledávám v určité oblasti daného jízdního pruhu (podobně jako na obrázku 48) vozidla. Pokud se zde vozidlo vyskytuje, uložím souřadnice těžiště tohoto vozidla. V následujícím snímku vyhledám opět souřadnice těžiště tohoto vozidla, přičemž využívám faktu, že vozidlo se může pohybovat pouze dopředu daným směrem. Z těchto dvou souřadnic vypočítám odpovídající vzdálenosti v 3D reálné scéně (v metrech) a jejím odečtením vypočítám vzdálenost (v metrech), kterou vozidlo urazilo. Jak jsem již dříve uvedl, časový interval mezi snímky budeme uvažovat 0,2 s, tím získáme rychlost vozidla v m/s, kterou přepočítáme na km/h. Z několika vypočítaných rychlostí vozidel již snadno získáme průměrnou rychlost daného jízdního pruhu. Ta se v testované sekvenci snímků pohybovala okolo 80 km/h v levém jízdním pruhu (zde by měla být průměrná rychlost nejvyšší ze všech jízdních pruhů v daném směru jízdy). Jedná se však o přibližný údaj, protože výpočet rychlosti ovlivňuje mnoho chyb, nepřesností a zaokrouhlování.

## 2.9 POČÍTÁNÍ OBSAZENOSTI JÍZDNÍHO PRUHU

Obsazenost jízdního pruhu udává, jak často se v daném jízdním pruhu vyskytuje vozidlo. Proto v určité oblasti jízdního pruhu (podobně jako na obrázku 48) provádíme kontrolu, jestli se zde nachází detekované vozidlo. Toto provádíme u každého následujícího snímku. Obsazenost jízdního pruhu potom zjistíme podílem detekovaných vozidel ku celkovému počtu snímků, které jsme prověřili. V testované sekvenci snímků byla obsazenost prostředního jízdního pruhu (ze tří jízdních pruhů v daném směru) okolo 15 %.

## 2.10 KLASIFIKACE VOZIDEL DO KATEGORIÍ

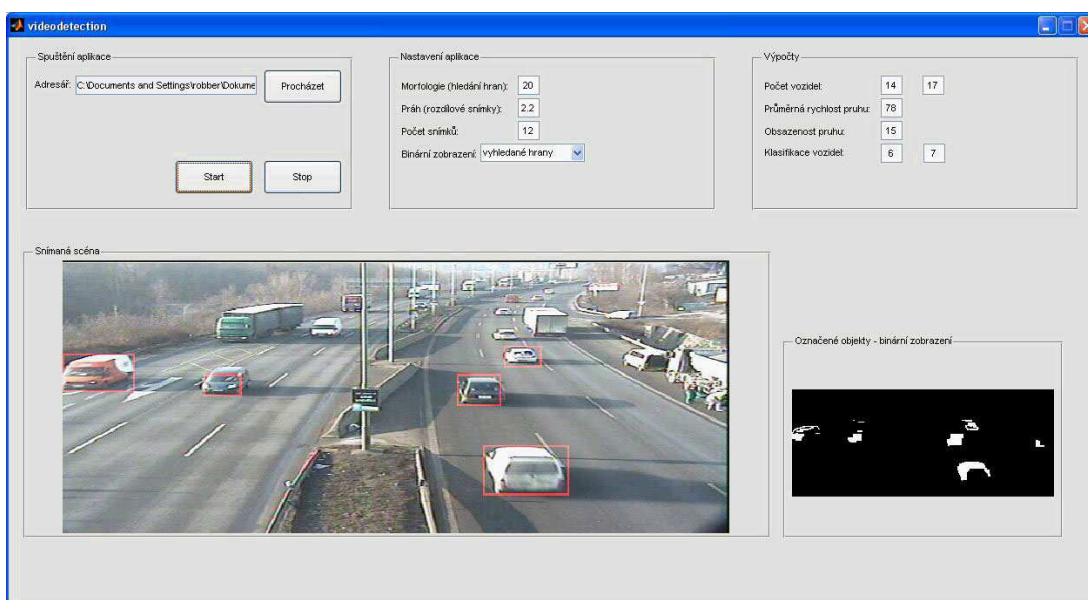
Vozidla jsou klasifikována do dvou kategorií (osobní, nákladní) podle jejich velikosti. U detekovaného vozidla v daném jízdním pruhu je proveden součet jeho rozměrů (šířka v ose x a výška v ose y) a podle tohoto součtu se rozhoduje, do které ze dvou kategorií bude toto vozidlo zařazeno.

## 2.11 UŽIVATELSKÉ ROZHRAŇÍ

Pro pohodlnější ovládání a vyhodnocení snímané scény na jedné obrazovce bylo vytvořeno uživatelské rozhraní GUI v programu MATLAB v.7.7.0.471 (R2008b). Ve zdrojových kódech jsou použity některé speciální funkce a příkazy, týkající se zpracování obrazu a pro jejich funkčnost je potřeba mít v MATLABu nainstalovanou knihovnu Image Processing Toolbox<sup>TM</sup>. Pro otevření, editaci a následné spuštění uživatelského prostředí lze postupovat následovně:

- v hlavním okně MATLABu zvolíme File -> New -> GUI
- v okně GUIDE Quick Start přejdeme do záložky Open Existing GUI, zde klikneme na Browse a otevřeme soubor videodetection.fig
- pro spuštění aplikace v okně videodetection.fig zvolíme Tools -> Run

Uživatelské rozhraní, které je vidět na obrázku 50, je rozděleno na několik částí. Horní polovina prostředí slouží k nastavení aplikace, spodní polovina slouží k zobrazení zpracovaných snímků.



Obrázek 50: Uživatelské rozhraní

V části „Spuštění aplikace“ uživatel vybere složku se sekvencí na sebe navazujících snímků z dopravní kamery, která bude dále zpracována. Složku lze vybrat kliknutím na tlačítko „Procházet“ nebo lze cestu k adresáři vepsat přímo do textového pole.

Sekce „Nastavení aplikace“ slouží k editaci důležitých parametrů, které mají vliv na způsob chodu aplikace. Např. parametry ovlivňující morfologické transformace s binárním obrazem hran, nebo výběr binárního obrazu, který bude zobrazen v pravé dolní části hlavního okna.

V části „Výpočty“ jsou zobrazeny údaje o snímané scéně a vozidlech, která se zde vyskytují. Můžeme zde zjistit, kolik vozidel projelo od spuštění aplikace nebo rychlost či obsazenost jízdního pruhu.

V části „Snímaná scéna“ je zobrazen výsledný aktuální snímek, na kterém jsou označena vozidla pohybující se ve snímané scéně.

Sekce „Označené objekty – binární zobrazení“ zobrazuje částečně zpracovaný binární snímek. Typ snímku, který má být zobrazen, může uživatel vybrat v sekci „Nastavení aplikace“.

V aplikaci lze nastavovat jen několik důležitých parametrů, které mají vliv na výslednou detekci vozidel a s ní spojené výpočty. Pro každý parametr aplikace je připravena stručná nápověda, která se zobrazí po přesunutí kurzoru na daný prvek.

### 3. ZÁVĚR

Úkolem této diplomové práce bylo seznámit se s problematikou vyhledávání pohybujících se objektů v sekvenci na sebe navazujících snímků z dopravní kamery. Práce je rozdělena na čtyři hlavní části. V první z nich jsou popsány způsoby vytvoření modelu snímané scény. Ve druhé části jsou popsány některé metody vyhledávání pohybujících se objektů ve snímané scéně. Třetí část se zabývá výpočty s detekovanými objekty. Čtvrtá část popisuje vytvořené uživatelské rozhraní.

Pro výpočet modelu prostředí byly vyzkoušeny tři různé metody. První popsaný způsob výpočtu modelu prostředí je výpočet průměrováním. Další metodou je výpočet modelu prostředí klouzavým průměrem. Poslední vyzkoušená metoda je výpočet modelu mediánem. Výpočet modelu prostředí mediánem je mnohem složitější než předchozí dvě metody a je výpočetně velice náročný. Model je však velice kvalitní již při výpočtu z 25 snímků (obrázek 9), v místech jízdních pruhů není ovlivněn projíždějícími vozidly a je velice odolný proti extrémně tmavým nebo světlým snímkům. Takto získaný model prostředí je použit ve všech metodách pro hledání objektů, které model prostředí vyžadují.

První popsanou metodou vyhledávání pohybujících se objektů ve snímané scéně je vytvoření rozdílu mezi aktuálním snímkem a modelem prostředí (obrázek 16). Negativní úlohu zde hrají stíny pohybujících se objektů, které se nedají moc účinně eliminovat.

Další metodou vyhledávání pohybujících se objektů ve snímané scéně je detekce hran. Vyhledají se hrany v aktuálním snímku a v modelu prostředí. Rozdílem nalezených hran najdeme pohybující se objekty (obrázek 22). K detekci hran jsem využil Prewittova filtru, hrany jsou hledány v horizontálním i vertikálním směru. Pokud použijeme obě tyto metody pro hledání objektů současně, můžeme dosáhnout ještě lepšího výsledku (obrázek 25). Tato metoda je účinná hlavně u snímků, na kterých jsou vlivem jiných klimatických podmínek stíny objektů méně výrazné.

Jiný přístup pro hledání objektů ve snímané scéně je detekce rohů. Tato metoda spočívá ve vyhledávání významných bodů pohybujících se objektů. Pro hledání významných bodů jsem použil Harrisův detektor. Po přenesení nalezených

významných bodů do barevného snímku (obrázek 31) vidíme, že většina významných bodů jsou pohybující se objekty.

Zcela odlišný způsob vyhledávání pohybujících se objektů v sekvenci snímků je vyhledávání pomocí optického toku. Podobným způsobem provádějí analýzu dynamických obrazů biologické vizuální systémy. Každému bodu obrazu odpovídá dvojrozměrný vektor rychlosti vypovídající o směru a velikosti rychlosti pohybu v daném místě obrazu. Pokud máme v dynamickém obrazu více pohybujících se objektů, pak každý má své vlastní ohnisko expanze – to je znázorněno na obrázku 33. Výsledný snímek s naznačenými vektory optického toku můžeme vidět na obrázcích 34 a 35. Z vektorů optického toku můžeme poměrně snadno poznat, jakým směrem se každý objekt pohybuje a z délky vektorů je vidět, že některé objekty se pohybují rychleji, některé pomaleji. Nevýhodou této metody je poměrně vysoká výpočetní náročnost. Další nevýhodou je možnost aplikovat metodu optického toku jen na sekvence snímků s velmi krátkými časovými intervaly mezi snímky.

Další úloha, kterou jsem se zabýval, je hledání pohybujících se objektů v nočních snímcích. Nejlepších výsledků dosahovala metoda vyhledávání objektů detekcí hran. Detektor občas označí i odlesky ve vozovce, ale oproti ostatním vyzkoušeným metodám tuto chybu nedělá tak často. Navíc lze dalším zpracováním tyto detekované odlesky částečně eliminovat, viz obrázek 46.

Jakmile máme vyzkoušené a vybrané metody pro vyhledávání vozidel, můžeme vozidla ve snímané scéně začít počítat. Pro tuto snímanou scénu jsem zvolil dvě počítadla. Jedno z nich počítá příjíždějící vozidla a druhé počítá odjíždějící vozidla. Dalším úkolem bylo počítat průměrnou rychlost jízdního pruhu. Rychlost vozidla spočítáme podle vzdálenosti, kterou vozidlo urazilo mezi dvěma po sobě jdoucími snímky. Z několika vypočítaných rychlostí vozidel získáme průměrnou rychlost daného jízdního pruhu. Ta se v testované sekvenci snímků pohybovala okolo 80 km/h v levém jízdním pruhu. Jedná se však o přibližný údaj, protože výpočet rychlosti ovlivňuje mnoho chyb, nepřesností a zaokrouhlování. Dále můžeme počítat obsazenost jízdního pruhu, která udává, jak často se v daném jízdním pruhu vyskytuje vozidlo. Obsazenost jízdního pruhu zjistíme podílem



detekovaných vozidel ku celkovému počtu snímků, které jsme prověřili. V testované sekvenci snímků se obsazenost prostředního jízdního pruhu (ze tří jízdních pruhů v daném směru) pohybovala okolo 15 %. Detekovaná vozidla můžeme také klasifikovat do kategorií. Vozidla jsou klasifikována do dvou kategorií (osobní, nákladní) podle jejich velikosti. Podle součtu šířky a výšky vozidla se rozhoduje, do které ze dvou kategorií bude vozidlo zařazeno.

Poslední částí tohoto textu je popis uživatelského rozhraní, které bylo vytvořeno pro pohodlnější ovládání a vyhodnocení snímané scény. Uživatelské rozhraní (GUI) bylo vytvořeno v programu MATLAB. V aplikaci jsou uplatněny ty postupy, které jsou popsány v této práci a zároveň vedou k nejlepším výsledkům při detekci a počítání vozidel. Výhodou aplikace je velký rozsah hodnocených parametrů snímané dopravní scény. Největší slabinou je časová náročnost těchto výpočtů. Pro zajištění vyšší rychlosti výpočtů by bylo nutné algoritmy napsat v jiném programovacím jazyce. Na základě vyzkoušených postupů může být v budoucnu vytvořena aplikace schopná rozpoznat dopravní kongesce a včas na ně upozornit.

## 4. LITERATURA

- [1] ŽÁRA, Jiří, BENEŠ, Bedřich, SOCHOR, Jiří, FELKEL, Petr. *Moderní počítačová grafika*. 1. vyd. Brno : Computer Press, 2004. 609 s. ISBN 80-251-0454-0.
- [2] PŘIBYL, Pavel, SVÍTEK, Miroslav. *Inteligentní dopravní systémy*. 1. vyd. Praha : BEN - technická literatura, 2001. 543 s. ISBN 80-7300-029-6.
- [3] KALOVÁ, Ilona, LISZTWAN, Marek. *Kontrola, sledování a řízení silniční dopravy*. Automatizace [online]. Leden 2007, 50, 1, [cit. 2010-05-13]. Dostupný z WWW: <<http://www.automatizace.cz/article.php?a=1589>>.
- [4] SONKA, Milan, HLAVAC, Vaclav, BOYLE, Roger. *Image Processing, Analysis, and Machine Vision*. 3. vyd. Toronto : Thomson Learning, 2008. 829 s. ISBN 0-495-08252-X.
- [5] SVOBODA, Tomas, KYBIC, Jan, HLAVAC, Vaclav. *Image Processing, Analysis, and Machine Vision: A MATLAB Companion*. 1. vyd. Toronto : Thomson Learning, 2008. 255 s. ISBN 0-495-29595-7.
- [6] KOVESI, Peter. *Harris corner detector* [online]. c2002-2005 [cit. 2009-12-03]. Dostupný z WWW: <<http://www.csse.uwa.edu.au/~pk/Research/MatlabFns/Spatial/harris.m>>.
- [7] HLAVÁČ, Václav, ŠONKA, Milan. *Počítačové vidění*. 1. vyd. Praha : Grada, 1992. 252 s. ISBN 80-85424-67-3.
- [8] GAUTAMA, Temujin. *Phase-based Optical Flow* [online]. c2002-2004 [cit. 2009-12-03]. Dostupný z WWW: <<http://www.mathworks.com/matlabcentral/fileexchange/2422>>.
- [9] Centrum dopravního výzkumu. *Zásady pro vodorovné dopravní značení na pozemních komunikacích*. II. vydání. Brno : Centrum dopravního výzkumu, 2005. 70 s. Dostupné z WWW: <[http://www.ibesip.cz/files/=646/TP\\_133\\_2vydani.pdf](http://www.ibesip.cz/files/=646/TP_133_2vydani.pdf)>. ISBN 80-86502-25-2.

## 5. SEZNAM ZKRATEK A SYMBOLŮ

<b>3D</b>	3 Dimensional, tří-dimenzionální prostor
<b>2D</b>	2 Dimensional, dvou-dimenzionální prostor
<b>GUI</b>	Graphical User Interface, grafické uživatelské prostředí
<b>ITS</b>	Intelligent Transport Systems, inteligentní dopravní systémy
<b>OE</b>	ohnisko expanze – optický tok

## 6. SEZNAM PŘÍLOH

### Příloha 1: DVD

- zdrojový tvar písemné zprávy diplomové práce
- zdrojové texty programů, aplikace GUI

