VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ
FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

DETEKCE KOLIZÍ VOZIDEL

VEHICLE COLLISION DETECTION

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE — MARTIN KRUTÁK
AUTHOR

VEDOUCÍ PRÁCE — Ing. JAKUB ŠPAŇHEL
SUPERVISOR

BRNO 2016

Vysoké učení technické v Brně - Fakulta informačních technologií

Ústav počítačové grafiky a multimédií Akademický rok 2015/2016

Zadání bakalářské práce

Řešitel: **Kruták Martin**
Obor: Informační technologie
Téma: **Detekce kolizí vozidel**
Vehicle Collision Detection
Kategorie: Zpracování obrazu

Pokyny:

1. Prostudujte dostupné materiály na téma detekce kolizí vozidel pro záběry ze stacionární kamery.
2. Vyberte vhodnou metodu a navrhněte způsob řešení dané problematiky.
3. Experimentujte s vaší implementací a případně navrhněte vlastní modifikace metod.
4. Porovnejte dosažené výsledky a diskutujte možnosti budoucího vývoje. Zvažte další pokračování v rámci diplomové práce.
5. Vytvořte stručný plakát nebo video prezentující vaši bakalářskou práci, její cíle a výsledky.

Literatura:

- Dle pokynů vedoucího.

Pro udělení zápočtu za první semestr je požadováno:

- Splnění prvních tří bodů zadání.

Podrobné závazné pokyny pro vypracování bakalářské práce naleznete na adrese http://www.fit.vutbr.cz/info/szz/

Technická zpráva bakalářské práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap (20 až 30% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVD-R, apod.), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Špaňhel Jakub, Ing.**, UPGM FIT VUT
Datum zadání: 1. listopadu 2015
Datum odevzdání: 18. května 2016

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav počítačové grafiky a multimédií
612 66 Brno, Božetěchova 2

doc. Dr. Ing. Jan Černocký
vedoucí ústavu

Abstrakt

Tato bakalářská práce popisuje systém pro detekci a vícenásobné sledování vozidel z dohledových kamer s detekcí kolizí. Práce je zaměřena na detekci a predikci kolize vozidel na dálnicích v jednom směru jízdy. Systém není plně automatický, tudíž vyžaduje počáteční nastavení (např. čáry vymezující pruhy na silnici) pro svou správnou funkčnost. Problém je řešen tak, že je nejdříve získána přesná kontura vozidla. Poté je vypočteno těžiště této kontury. Na základě tohoto těžiště je každému vozidlu přidělen jízdní pruh, ve kterém se pohybuje a každé vozidlo je poté sledováno individuálně. Práce dále představuje metodu predikce a detekce kolize. Pro tuto část je vytvořen obdélník kolem přízemní části vozidla. Tento obdélník je poté zvětšen a vyhodnocován proti obdélníkům jiných vozidel. Objekty, jejichž obdélníky se překrývají jsou předmětem dalšího sledování možné kolize. Experimentální výsledky ukazují 72% přesnost konstrukce přízemního obdélníku jakožto základního stavebního kamene v navrženém systému pro detekci kolizí. Výhody systému jsou v jeho možnostech využití v kamerových systémech, které monitorují provoz na dálnicích.

Abstract

This bachelor thesis decribes a system for detection and tracking of multiple vehicles from a surveillance camera with a collision detection. The focus is on detection and prediction of collisions of vehicles in one direction – towards the camera. System is not fully automatic, meaning that some initial settings are needed (e.g. lines on the road) to quarantee a good functionality of the system. Accurate vehicles' contour is obtained in the detection phase, and object centroids are calculated. Each detected vehicle is assigned to the specific lane and tracked separately. This thesis then describes the method of prediction and detection of a collision. A rectangle is created around the ground part of every vehicle. This rectangle of each of the vehicles is enlarged and checked for the overlaps. Those rectangles that overlaps are then subject to further analysis for the collision detection. Experimental results show a success rate of 72 % for the accurate rectangle construction being a crucial part for the collision detection. The advantage of the proposed system is its possible usage in surveillance cameras monitoring the traffic flow on highways.

Klíčová slova

vícenásobné sledování vozidel, predikce kolize, detekce vozidel, detekce kolize

Keywords

multiple vehicle tracking, collision prediction, collision detection, vehicle detection

Citace

KRUTÁK, Martin. *Detekce kolizí vozidel.* Brno, 2016. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Špaňhel Jakub.

Detekce kolizí vozidel

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Jakuba Špaňhela. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

. . . . . .
Martin Kruták
11. května 2016

Poděkování

Rád bych poděkoval svému vedoucímu Ing. Jakubu Špaňhelovi za cenné rady při tvorbě této práce a rodině za podporu po dobu studia.

© Martin Kruták, 2016.
Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

1 Úvod **2**

2 Techniky pro sledování vozidel, detekce a predikce kolizí **3**
2.1 Metody detekce pozadí založené na rozdílu snímků 3
2.2 Operace pro zlepšení čistoty masky 6
2.3 Techniky pro hledání kontur v obraze a jejich aproximaci 7
2.4 Metody pro popis tvarů objektů 8
2.5 Metody pro sledování objektů 8
2.6 Problém predikce kolize 11
2.7 Způsoby detekování kolizí 13

3 Návrh systému pro detekci kolizí **14**
3.1 Detekce vozidel ve videu 14
3.2 Získání a úprava kontury 15
3.3 Sledování vozidel pomocí kontury 16
3.4 Predikce vzájemných kolizí 18
3.5 Detekce kolizí 20

4 Vyhodnocení navrženého systému **22**
4.1 Přesnost navrženého systému 22
4.2 Výpočetní rychlost systému 23

5 Závěr **24**

Literatura **25**

Kapitola 1

Úvod

Inteligentní systémy pro kontrolu provozu podporují základní systémy pro správu, jako navigace automobilů, řízení semaforů, rozpoznávání státní poznávací značky nebo měření rychlosti vozidel.

V této práci je představen způsob sledování více vozidel z dohledové kamery s predikcí a detekcí vzájemných kolizí.

Práce se zaměřuje na dva typy kolizí, které mohou nastat na dálnici v jednom směru jízdy. První případ je vzájemná kolize vozidel kdy jedno narazí zezadu do druhého. Druhým případem je kolize vozidel v průběhu předjíždění.

V první kapitole jsou popsány techniky pro sledování vozidel. Zmíněny jsou metody, které se využívají pro odečet pozadí, kde zaměření je na mix Gaussových rozložení. Dále jsou v této kapitole popsány metody pro zlepšení výstupní masky detektoru, které se zabývají vylepšením masky tak, aby byla co nejlepší při vstupu do další části jejího zpracování. V závěru kapitoly jsou popsány přístupy ke sledování objektů ve videu. Zmíněny jsou zde i metody pro predikci a detekci kolizí.

V druhé kapitole je popsán samotný návrh systému. Jednotlivé podkapitoly jsou seřazeny tak, aby odpovídaly postupu při práci. Kapitola tedy začíná zmíněním aplikace zvolené metody pro odečet pozadí a detekci popředí, pokračuje přes systém predikcí kolizí po popis detekce kolizí.

Ve třetí kapitole je vyhodnocena úspěšnost systému v konstrukci přízemního trojúhelníku, jakožto elementárního prvku v konečné predikci a detekci kolizí.

V záběru jsou zhodnoceny výsledky práce, objektivně vyhodnoceny její silné a slabé stránky a možnosti dalšího vývoje.

Kapitola 2

Techniky pro sledování vozidel, detekce a predikce kolizí

Aby bylo možné sledovat jednotlivá vozidla, je nutné nejdříve detekovat pohybující se objekty ve video sekvenci. Jeden ze způsobů, jak toho docílit, je detekováním pohybujících se objektů pomocí rozdílů modelu pozadí a aktuálního snímku. Pozadí je neměnné prostředí, které se v čase obnovuje, aby se tak dalo lépe reagovat na změny prostředí, jako je měnící se povětrnostní a světelné podmínky. Pomocí modelu pozadí je možné detekovat pohybující se vozidla. Detekovaná vozidla je poté možné klasifikovat jako individuální objekty a zkoumat pro případné kolize.

2.1 Metody detekce pozadí založené na rozdílu snímků

Metoda detekce pozadí (Background subtraction) je prvním krokem k detekci pohybujících se objektů ve videu. Typickým přístupem je provedení rozdílu aktuálního snímku a referenčního snímku. Referenční snímek je model pozadí nebo snímek pozadí [4, 13]. Model pozadí je reprezentace prostředí bez pohybujících se objektů, který se v čase obnovuje, aby odolával změnám v osvětlení a geometrii. Některé metody si tento model vytváří samy na základě prvních několika snímků a pak samy také spravují. Způsoby detekce pozadí jsou zmíněny níže. V binárním snímku, získaném touto cestou, reprezentují černé pixely pozadí a bílé pixely pohybující se objekty, viz obrázek 2.1.

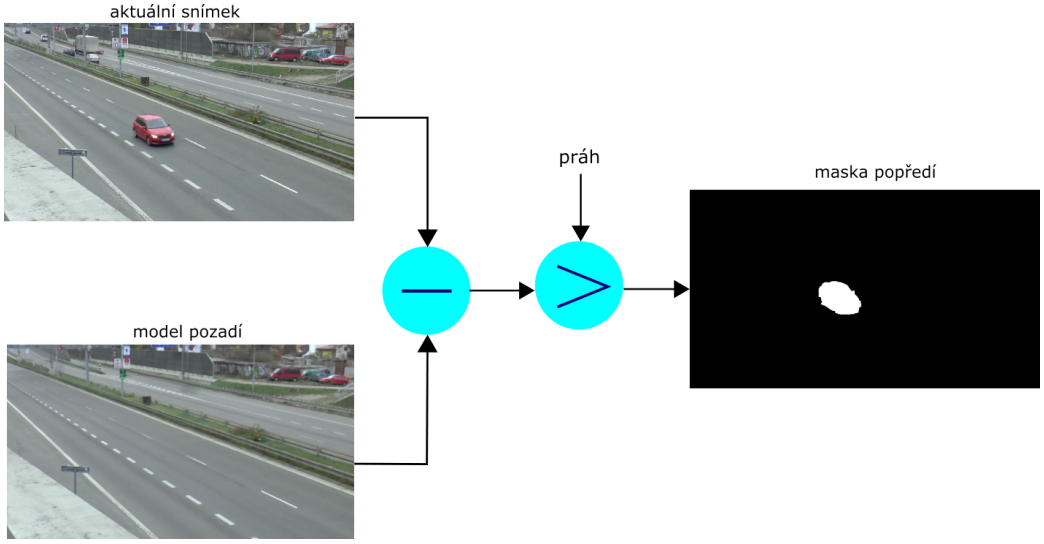
2.1.1 Klouzavý průměr Gaussových rozložení

Tato metoda je nejjednodušší z metod používaných pro detekci pozadí [20]. Metoda spočívá v tom, že model pozadí je vypočítáván nezávisle v každém bodu pixelu. Model stojí na ideálním modelování Gaussovy funkce hustoty (Gfh) z posledních n hodnot pixelu. Aby se předešlo modelování Gfh v každém snímku v čase t vždy od nuly, klouzavý průměr je vypočten následovně:

$$\mu_t = \alpha I_t + (1 - \alpha)\mu_{t-1} \tag{2.1}$$

kde I_t je aktuální hodnota pixelu, μ_t je předchozí hodnota průměru a α je empirická hmotnost.

Model navržený Wren a kol. [20] pracuje s intenzitou obrazu, ale je možné tuto metodu rozšířit o (RGB) barevný prostor. Navíc, v případě nutnosti dosažení vyšších rychlostí za cenu nižší spolehlivosti, lze snížit frekvenci aktualizace modelu.



Obrázek 2.1: Základ metody detekce pozadí [1].

2.1.2 Časový medián filtr

Tato metoda je podobná s metodou výše zmíněnou. Používá hodnotu mediánu posledních n snímků. Podle Cucchiara a kol. [6] je dostačující, když se pro detekci nepoužívá každý snímek, ale jen třeba každý desátý. Kvalita získaného pozadí je závislá na počtu snímků, ze kterých je medián vypočten. Pokud je jejich počet příliš nízký, může pozadí obsahovat „duchy“ (části zachycených pohybů objektů ve snímku).

Nevýhodou této metody je její paměťová náročnost, protože se musí uchovávat informace o každém pixelu za posledních n snímků. Kromě toho, mediánový filtr nelze přizpůsobit pro rigorózní statistický popis.

2.1.3 Mix Gaussových rozložení

Je pravděpodobné, že objekty v pozadí se v čase objevují periodicky v jednom místě. Příkladem může být větev stromu pohybující se ve větru. Na pozici pixelu *p(x,y)* se tedy bude periodicky zobrazovat listí, větev a případné objekty za touto větví. Další případy mohou být změny počasí – sněžení či déšť. Všechny doposud zmíněné metody by okamžitě obnovily svůj model pozadí, což ve výše zmíněných dočasných změnách pozadí není adekvátní.

Stauffer a Grimson [16] popisují pravděpodobnost zpozorování hodnoty pixelu x v čase t prostřednictvím mixu Gaussových rozložení:

$$P(x_t) = \sum_{i=1}^{K} \omega_{i,t} \eta(x_i - \mu_{i,t}, \Sigma_{i,t}) \tag{2.2}$$

Každé z K Gaussových distribucí popisuje jeden pixel z pozadí nebo popředí s objekty. V praxi se K pohybuje od 3 do 5.

Aby 2.2 mohla sama o sobě popisovat daný model, je nutné zvolit kritérium pro rozlišení mezi popředím a pozadím. Všechna rozložení jsou tedy seřazena podle poměru nejvyšší amplitudy ω_i a směrodatné odchylky σ_i. Předpokladem je, že čím vyšší a kompaktnější je rozložení, tím více je pravděpodobné, že patří pozadí. Správné rozložení je poté vybráno

podle následující podmínky:

$$\sum_{i=1}^{B} \omega_i > T \tag{2.3}$$

Kde T je přiřazený práh. Rozložení, která vyhovují podmínce jsou považována za pozadí.

V každém snímku v čase t musí navíc být vyřešeny dva problémy. Jedním je přiřazení nové hodnoty x_t k nejlepšímu shodující se rozložení a tím druhým je odhad nových parametrů modelu. Oba problémy se řeší najednou. Shoda se určuje podmínkou:

$$\frac{x_i - \mu_{i,t}}{\sigma_{i,t}} > 2,5 \tag{2.4}$$

První z dříve seřazených rozložení, které vyhovuje této podmínce, je přijato jako shoda pro x_t.

2.1.4 Odhad hustoty pomocí jádra

Pozadí může být pomocí Gfh[1] aproximováno jako histogram nejnovějších hodnot klasifikovaných jako pozadí. Elgammal a kol. [7] ve své práci navrhují modelovat distribuční funkci pozadí pomocí neparametrizovaného modelu odhadu hustoty pomocí jádra. Metoda je aplikována na posledních n hodnot pozadí uložených v bufferu.

Pozadí Gfh je dáno jako součet jader Gaussových rozložení vystředěných v nejnovějších n hodnotách pozadí x_i:

$$P(x_t) = \frac{1}{n} \sum_{i=1}^{n} \eta(x_t - x_i, \Sigma_t) \tag{2.5}$$

Pokud jsou hodnoty pozadí neznámé, mohou být použita neklasifikovaná data. Počáteční nepřesnost se v budoucnu s obnovou modelu obnoví. V závislosti na 2.5, může x_t být ihned klasifikováno jako popředí, pokud $P(x_t) > T$.

Obnova modelu je prováděna s obnovením bufferu hodnot pozadí ve FIFO pořadí. Odhad hustoty pomocí jádra, tak jak je navrhnut autory Elgammal a kol. [7], využívá navíc, na rozdíl od metod zmíněných v sekci 2.1.1 - 2.1.3, vyhodnocení hodnoty $P(x_t)$ také s modely okolních pixelů a nejvyšší hodnotu porovnává proti T.

2.1.5 Souběžný výskyt variací obrazu

Hlavní myšlenkou Seki a kol. [15] je, že bloky pixelů, které patří jednomu pozadí by měly mít podobné variace. Tento předpoklad je pravdivý, ale jak lze vidět v jejich práci, není funkční v blocích na pozadí vzdálených objektů. Metodu takto navrženou lze shrnout následovně:

- Metoda místo jednotlivých pixelů pracuje s bloky N×N pixelů. Každý blok reprezentuje N^2 rozměrný vektor. Je tak dosaženo vyšší rychlosti algoritmu výměnnou za nižší rozlišení obrazu.
- Pro každý blok je získán určitý počet vzorků; první je vypočten dočasný průměr; rozdíly mezi vzorky a průměrem se poté nazývají *variace obrazu.*
- S ohledem na vypočtený průměr je vypočtena $N^2 \times N^2$ kovariantní matice a po aplikaci Eigen vektorové transformace, je rozměr variací obrazu snížen z N^2 na K.

[1] Gaussova funkce hustoty

- Sousední blok u je vzat v úvahu s jeho aktuální vstupní hodnotou a je vypočtena odpovídající Eigen variace obrazu, z_u.
- Jsou nalezeny L-nejbližší sousedé z_u v Eigen prostoru $z_{(u,i)}$ a z_u je vyjádřeno jako jejich lineární interpolace.
- Tytéž interpolační koeficienty jsou aplikovány na hodnoty aktuálního bloku b, které se objevily ve stejném čase jako $z_{(u,i)}$; toto poskytuje odhad z_b^* pro jeho aktuální Eigen variaci obrazu z_b.
- Myšlenkou tohoto přístupu je, že z_b a z_b^* by měly být blízko sebe, pokud b je blok z pozadí.

2.2 Operace pro zlepšení čistoty masky

Morfologické operace se používají na segmentaci obrazu. Tyto operace jsou čtyři: eroze, dilatace a jejich základní kombinace, uzavření a otevření [19]. Operace lze vidět na obrázku 2.2. Cílem těchto operací je:

- Odstranění šumu
- Izolace a spojení rozličných objektů v obrazu
- Hledání výkyvů intenzity a děr v obraze

2.2.1 Eroze

Eroze je operace, která provádí sjednocení obrazu (X) a strukturního elementu (B), který je systematicky posunován po obraze. Tento strukturní element může být jakékoli velikosti a jakéhokoli tvaru. Eroze zmenšuje objekty, vyhlazuje hrany, odstraňuje izolované malé objekty, rozděluje nežádoucí propojení. Erozi lze vyjádřit vztahem:

$$X \ominus B = \bigcap_{b \in B} X_{-b} \tag{2.6}$$

2.2.2 Dilatace

Dilatace pracuje podobně jako eroze. Dilatace, na rozdíl od eroze, provádí průnik obrazu (X) a strukturálního elementu (B). Dilatace zvětšuje objekty, vyplňuje zálivy (vyhlazování hran), propojuje objekty a vyplňuje otvory. Lze vyjádřit následovně.

$$X \oplus B = \bigcup_{b \in B} X_b \tag{2.7}$$

2.2.3 Otevření

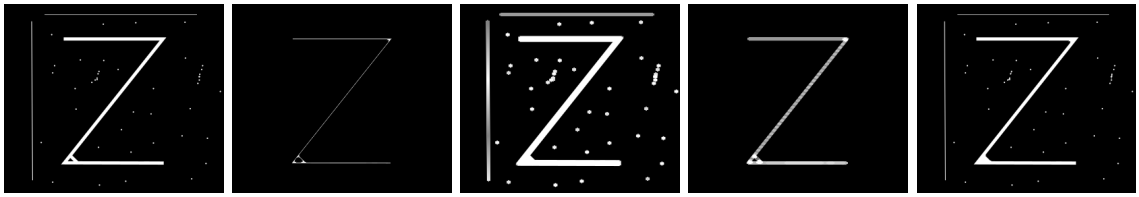
Otevření je operace, která postupně aplikuje na obraz (X) erozi a dilataci přesně v tomto pořadí, za pomoci strukturního elementu (B). Operace otevření je vhodná pro odstraňování malých objektů v obraze.

$$X \circ B = (X \ominus B) \oplus B \tag{2.8}$$

2.2.4 Uzavření

Uzavření je operace, která postupně aplikuje na obraz (X) v přesném pořadí po sobě dilataci a poté erozi, za pomoci strukturního elementu (B). Operace uzavření je vhodná pro vyplnění malých děr v obraze.

$$X \bullet B = (X \oplus B) \ominus B \tag{2.9}$$



Obrázek 2.2: Zleva: vstupní obrazec, obrazec po použití eroze, dilatace, otevření a uzavření.

2.3 Techniky pro hledání kontur v obraze a jejich aproximaci

Kontury lze detekovat několika způsoby. Obvyklým vstupem takového detektoru je výstup jakéhokoli hranového detektoru (např. Canny [5]) nebo výsledek prahování. Nejpoužívanějším algoritmem pro detekci kontur je technika sledování okraje [17].

Tento algoritmus autoři rozdělili na dva algoritmy. Hlavní algoritmus detekuje vnější okraje a díry a upravená varianta algoritmu jen vnější okraj.

V kontextu těchto dvou rozdělení definují bod okraje, okolí sloučených komponent, vnější okraj a díru okraje. V následujícím odstavci se myslí 1-pixel jako pixel s komponentou 1 a 0-pixel s komponentou 0.

- Bod okraje – pokud mezi 4-(8-) spojení bodů má 1-pixel *(i,j)* ve svém 4-(8-) okolí 0-pixel *(p,q)*, je tento bod nazýván bodem okraje.
- Okolí sloučených komponent – máme-li dvě sloučené komponenty S_1 a S_2 v binárním obraze, říkáme že S_2 obklopuje S_1, pokud existuje takový pixel patřící S_2 pro jakoukoli 4-cestu z pixelu S_1 do pixelu na snímku. Pokud S_2 obklopuje S_1 a nachází se mezi nimi bod okraje, poté S_2 přímo obklopuje S_1.
- Vnější okraj a díra okraje – vnější okraj je definován jako set okrajových bodů mezi komponentami 1 a komponentami 0, které je přímo obklopují. Podobně díra okraje je označována ta část, kde jsou body okraje mezi dírou a komponentami 1, které ji obklopují přímo.

Algoritmus navržený Suzuki a kol. [17] pracuje tedy tak, že prochází obraz bod po bodu, dokud nenarazí na pixel *(i,j)*, který uspokojí počáteční podmínky pro sledování okraje, buďto vnějšího okraje nebo díry. Poté determinuje jeho rodičovský okraj řídící se přitom podmínkami definovanými výše. Detekovanou konturu lze vidět na obrázku 2.3.

Máme-li takto detekovanou konturu, lze ji aproximovat konvexní křivkou. Graham a kol. [8] předpokládají, že polygon, který touto konvexní křivkou aproximují sám sebe nepřekřižuje a jeho vrcholy jsou orientovány po směru hodinových ručiček. Všechny vrcholy jsou poté procházeny jeden po druhém. Algoritmus vždy načte dva po sobě jdoucí vrcholy, které

Obrázek 2.3: Imperfektní kontura (vlevo), kontura aproximovaná konvexní křivkou (vpravo)

poté porovnává proti tětivě $L[q_0, q_1]$, která dělí prostor mezi dvěma vrcholy na dva prostory. Algoritmus poté vybere první vrchol napravo od tětivy a hledá první bod, který vystoupí ze současné konvexní křivky. Výstup viz obrázek 2.3.

2.4 Metody pro popis tvarů objektů

Momenty popisují tvar objektu [19]. Nižší řády momentů popisují vlastnosti objektu – obsah, těžiště nebo tvar materiálu. Takové momenty jsou vypočítány přímo z objektu nebo z jeho hranic. Momenty vyšších řádu se pak používají na popis projekcí objektu jako šikmost či špičatost. Pro momenty kontury se poté používá Greenův teorém [9]. Počáteční moment n-tého řádu je definován jako:

$$\mu_r = E(X^r) = \int_{-\infty}^{\infty} x^r f(x)\,\mathrm{d}x. \tag{2.10}$$

Centrální momenty jsou poté spočítány jako rozdíl původních hodnot a jejich průměru

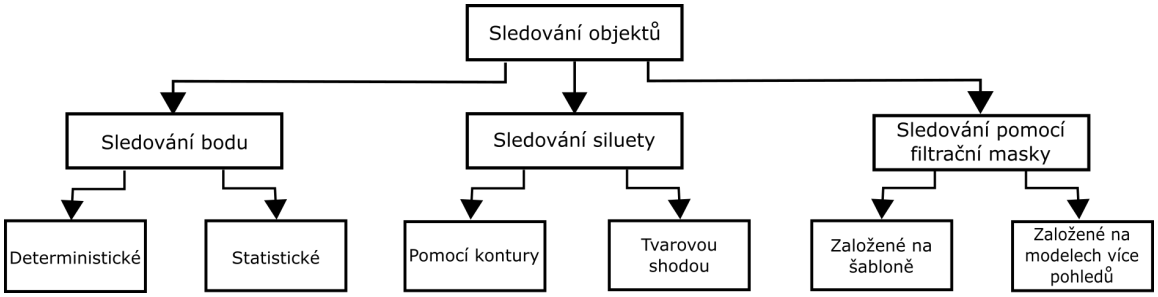
$$\mu'_r = E((X - E(X))^r) = \int_{-\infty}^{\infty} (x - E(X))^r f(x)\,\mathrm{d}x. \tag{2.11}$$

2.5 Metody pro sledování objektů

Sledování objektů je důležitým prvkem v počítačovém vidění. Tento proces napomáhá k individuálnímu zkoumání dílčích objektů ve snímku. Proces obsahuje tři klíčové prvky [21]: detekce pohybujících se objektů, sledování těchto objektů od snímku ke snímku a analýza pohybu objektu k rozpoznání jeho chování. Takové sledování lze poté použít v mnoha oblastech:

- Sledování pohybu – rozpoznání lidí, objektů
- Dohled – monitorování scény, detekování podezřelého chování
- Indexování videa – automatická anotace, získávání videí z multimediální databáze
- Interakce s člověkem – rozpoznávání gest, pohybu očí
- Navigace aut – plánování trasy, vyhýbání se objektům
- Monitorování dopravy – získávání statistik o dopravě pro řízení proudu dopravy

Metody pro sledování objektů se dají dle Yilmaz a kol. [21] rozdělit do tří hlavních kategorií a šesti podkategorií, viz obrázek 2.4.



Obrázek 2.4: Základní hierarchie sledovacích metod

2.5.1 Sledování bodu

Objekty, které jsou detekovány ve snímku jsou reprezentovány body skrze sérii po sobě jdoucích snímků. Problém sledování pomocí bodu je při překrytí dvou objektů, špatné detekci, nebo v případech, kdy sledovaný objekt opustí scénu.

Přístupy k implementaci této metody mohou být dvojí. Deterministická metoda pro korespondenci bodů a statistická metoda pro korespondenci bodů.

Deterministická metoda pro korespondencí bodů, určuje cenu asociace každého objektu ve snímku v čase $t-1$ s objekty ve snímku v čase t za použití souboru pohybových omezení. Řešení, které se skládá z „jedna k jedné“ korespondencí mezi ostatními korespondencemi, může být získáno jednou z asociačních metod. Jedna z metod asociace může být Maďarská metoda [12], nebo hladový algoritmus vyhledávání. Cena korespondence je obvykle určována pomocí následujících omezení:

- *Blízkost* – očekává, že pozice objektu se snímek od snímku příliš neliší.
- *Maximální rychlost* – definuje horní hranici rychlosti objektu.
- *Malá změna rychlosti* – očekává, že směr a rychlost objektu se drasticky nemění.
- *Obyčejnost pohybu* – rychlost objektu v okolí musí být podobná, vhodné pro objekty definované více body.
- *Tuhost objektu* – očekává, že objekty ve 3D prostoru jsou tuhé, tedy vzdálenost dvou bodů na objektu je neměnná.
- *Proximitní jednotnost* – kombinace proximity a malé změny v rychlosti.

Statistická metoda, na rozdíl od deterministické metody, počítá s náhlými změnami trajektorie objektu, ke kterým může dojít v důsledku jeho manévrování. Statistická metoda korespondence bodů používá přístup k objektu pomocí stavového prostoru. Metoda přitom využívá vlastností objektu jako zrychlení, rychlost a pozice.

Pro sledování vícero objektů lze použít Kalmanův filtr [3]. Kalmanův filtr lépe odolává šumu, ale pro jeho plné využití při sledování více objektů je potřeba i funkce, která by vybrala nejpravděpodobnější míru pro daný objekt ke stavu daného objektu. Tedy k sobě korespondující body v po sobě jdoucích snímcích. K tomuto účelu lze využít již zmíněnou Maďarskou metodu.

2.5.2 Sledování pomocí filtrační masky

U této metody bude zmíněno jen její využití pří sledování více objektů. Tato metoda se také potýká s problémy při sledování objektů, které se překrývají. Tao a kol. [18] navrhují sledovací metodu, která skládá celý snímek z několika vrstev. Jednu vrstvu pro pozadí a samostatnou vrstvu pro každý objekt. Každá vrstva se skládá z elips reprezentující objekt, modelu pohybu a vzhledu vrstvy (modelován pomocí Gaussova rozložení). Vrstvení je poté provedeno tak, že se nejprve kompenzuje pohyb pozadí modelovaného projektivním pohybem tak, že pohyb objektu může být odhadnut z kompenzovaného obrazu užitím 2D parametrického pohybu. Poté je vypočtena pravděpodobnost každého pixelu ve vztahu k náležitosti k jedné z vrstev, na základě pohybu objektu v minulosti a charakteristik tvaru. Každému pixelu, který je daleko od vrstvy je přiřazena pravděpodobnost pozadí. Tento postup je ovšem velice náročný, proto autoři individuálně vyhodnocují jeden set a přitom opravují sety ostatní. Neznámé pro každý objekt jsou získány iterativně dokud pravděpodobnost náležitosti dané vrstvě není maximální.

Sledování založeném na modelech více pohledů je založeno na tom, že objekt může vypadat z různých pohledů jinak a pokud se pohled objektu dramaticky změní, model vzhledu objektu nemusí být nadále validní. Aby se předešlo těmto problémům, různé pohledy objektu mohou být získány offline a poté použity pro sledování.

Hlavním cílem této sledovací metody je odhad pohybu objektu. S reprezentací objektu založeném na jeho oblasti, vypočtený pohyb implicitně definuje oblast objektu i jeho orientaci v dalším snímku, jelikož pro každý bod v aktuálním snímku může jeho pozice ve snímku dalším být stanovena pomocí odhadovaného modelu pohybu.

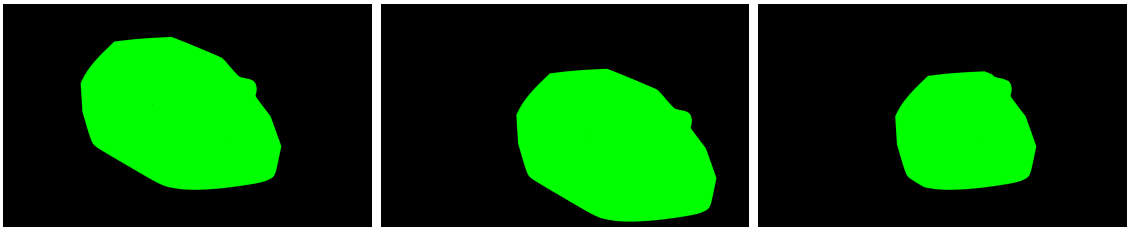
2.5.3 Sledování siluety

Metody založené na sledování siluety jsou metody, které se dají dobře využít pro sledování objektů, které nelze popsat jednoduchými tvary jako je elipsa či čtverec. Cílem této metody je najít oblast objektu v každém snímku na základě modelu objektu generovaného z předešlých snímků. Takový model může být stanoven na základě barevného histogramu objektu, detekovaných hran objektu, nebo kontury objektu. Metody lze dále dělit na dvě další, a to na metodu sledování na základě tvaru a nebo na základě kontury.

V případě **sledování** na základě **tvaru** je jedním z možných přístupů hledání odpovídajících siluet v po sobě jdoucích snímcích. Hledání shody těchto siluet je podobné sledování bodu (viz 2.5.1). Hlavní rozdíl je v reprezentaci objektu a použitého modelu objektu. Navíc, při sledování na základě tvaru lze také sledovat specifické vzhledové rysy objektu, kdežto při hledání shody bodové, lze využít jen rysy pohybu a pozice. Pro model objektu lze využít i histogram barvy a hran [11]. Histogram je generován v soustředných kruzích s různým rádiusem vycentrovaných na sadě bodů v referenčním kruhu. Referenční kruh je zvolen ten, který nejvíce přiléhá k siluetě objektu.

Sledování objektu pomocí **kontury** se liší od sledování na základě tvaru v tom, že kontury iterativně vyvíjí svoji základní konturu v předešlém snímku do nové pozice v aktuálním snímku. Takový vývoj vyžaduje, aby se části kontury z předešlého snímku překrývaly s konturou v aktuálním snímku. Stále ovšem zůstává neschopnost sledovat objekty, jejichž kontury se částečně překrývají. Tento problém lze vyřešit, pokud by kontury byly inicializované individuálně pro každý objekt.

Na obrázku 2.5 lze vidět, jak takové sledování pomocí kontury probíhá. Pokud se kontura v čase t překrývá s konturou získanou v čase $t-1$, můžeme tento objekt považovat za tentýž.



Obrázek 2.5: Kontura v čase $t-1$ (vlevo), kontura v čase t (uprostřed), průnik kontur (vpravo).

2.6 Problém predikce kolize

Jakmile proběhla detekce pohybujících se objektů a probíhá individuální sledování těchto objektů, jsme schopni zkoumat dílčí vlastnosti objektů či jejich chování, případné interakce mezi objekty. Jedním z parametrů, které lze zkoumat v dopravě je predikce kolizí vozidel.

Tato oblast je stále ještě mladá, prací v oboru není příliš. Není tedy ani práce, která by se zabývala porovnáním různých metod a jejich vyhodnocením. Každá metoda má své specifické využití, nelze tedy určit jednu metodu, která by byla v tom či onom případě ta nejlepší. Z literatury, která se zabývá touto problematikou vyvstaly některé přístupy, které jsou zmíněny níže.

2.6.1 Predikce na základě modelu trajektorií

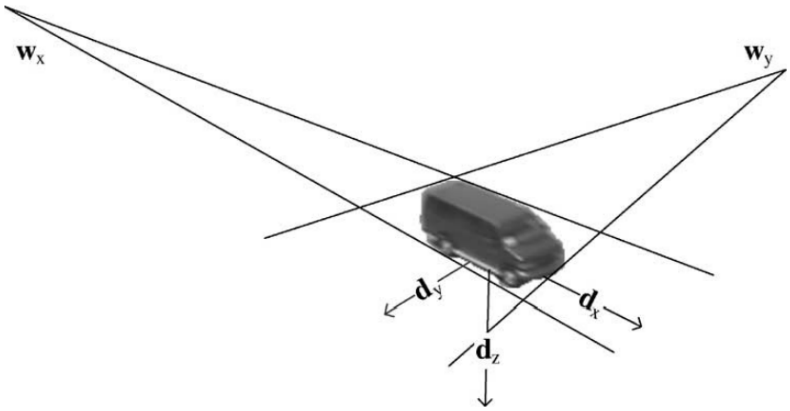
Tato metoda pracuje na základě modelování trajektorií vozidel. Počítá s tím, že řidiči jezdí po podobných trasách, opisujíc tak stále stejné (podobné) trajektorie. V práci Saunier a kol. [14] je každé vozidlo sledováno na základě jeho rysů. Těmito rysy mohou být například dobře rozeznatelné body nebo čáry. Jelikož takových rysů může být na jednom vozidle několik, je potřeba je shlukovat do jednoho objektu, ke kterému patří. Navržený systém poté vytvoří dvě databáze: databázi trajektorií, kde jsou uloženy výsledky modulu pro zpracování videa a databázi interakcí, kde jsou uváženy všechny interakce mezi uživateli v dané vzdálenosti a pro kterou jsou vypočteny pravděpodobnosti kolize a jiné vážné indikátory. Pro databázi interakcí je vypočtena předpověď času možné kolize. Prvně je definována pravděpodobnost kolize v čase t_0 pro účastníky silničního provozu (dále jen řidiči) A_1 a A_2 s příslušnými trajektoriemi $Q_1, t \leq t_0$ a $Q_2, t \leq t_0$, s ohledem na jednu extrapolovanou hypotézu každého řidiče H_i a H_j. Na základě hypotéz H_i a H_j je učiněna předpověď pozice. Předpovězený čas kolize $t_{i,j}$ je první instance, kdy by řidiči mohli být v kontaktu. Čím větší $\Delta_{i,j} = t_{i,j} - t_0$ tím je pravděpodobnější, že řidiči stihnout zareagovat a vyhnou se kolizi. Pravděpodobnost kolize dvou řidičů na základě hypotéz H_i a H_j je vypočtena následovně:

$$P(Kolize(A_1, A_2)|H_i, H_j) = e^{-\frac{\Delta_{i,j}^2}{2\sigma^2}} \tag{2.12}$$

kde σ je normalizovaná konstanta. Hu a kol. odhadují [10], že pravděpodobnost by se měla změnit, pokud je uplynulý čas $\Delta_{i,j}$ blízko reakčnímu času řidiče. Proto σ je zvoleno rovno průměru reakční doby. Podle Hu a kol. je pravděpodobnost kolize dvou řidičů A_1 a A_2 v čase t_0:

$$P(Kolize(A_1, A_2)|Q_1, t \leq t_0, Q_2, t \leq t_0) = \sum_{i,j} P(H_i|Q_1, t \leq t_0) P(H_j|Q_2, t \leq t_0) e^{-\frac{\Delta_{i,j}^2}{2\sigma^2}} \tag{2.13}$$

Obrázek 2.6: Zleva: referenční snímek, databáze trajektorií, databáze interakcí, model pravděpodobností kolize [14].



Obrázek 2.7: Konstrukce bounding boxu s tangentami vedoucími k úběžníkům w_x a w_y a příslušnými směry d_x, d_y a d_z [2].

kde $P(H_i|Q_1, t \leq t_0)$ je pravděpodobnost řidiče A_1, že se pohne na základě hypotézy extrapolace H_i. Podobně pro řidiče A_2.

2.6.2 Predikce na základě průsečíků bounding boxů

Atev a kol. [2] ve své práci navrhují metodu založenou na průsečících bounding boxů. Tyto bounding boxy jsou sestrojeny kolem vozidla na základě úběžníků. Způsob, jakým konstruují tyto úběžníky lze vidět na obrázku 2.7.

Po zkonstruování bounding boxu se mohou podle tohoto predikovat kolize. Podle Atev a kol., lze přistupovat k predikování kolizí tak, že se zkonstruované obdélníky zvětší o $\delta/2$, když δ je maximální vzdálenost mezi vozidly. Poté se tyto zvětšené obdélníky porovnají s obdélníky zkonstruovanými u okolních objektů. Autoři navíc navrhují způsob výpočtu průniků těchto obdélníků. Pokud se v obraze v daném čase objevuje méně jak 10 objektů, stačí na výpočty přistupovat metodou hrubé síly. Tato metoda využívá algoritmus pro stříhání polygonů, kdy dva obdélníky jsou použity pro obstřihnutí třetího obdélníku. Hrany obstřihnutého obdélníku jsou orientované, dle předpokladu, po směru hodinových ručiček. Každá hrana ve směru obstřihovaného obdélníku rozpůlí tento obdélník na dvě části, dále se ponechá jen pravá část. Tato část se stříhá dále, dokud všechny hrany nejsou použity. Ke kolizi došlo, pokud na konci tohoto procesu zůstane obdélník s nenulovou oblastí. V případě, že je v obraze v daném čase více jak 10 objektů, je použit čtyřstrom s kořenem reprezentující čistou obdélníkovou oblast, která bude obsahovat jakýkoli další obdélník – tedy vozidlo. Každý list stromu bude mít maximálně p vozidel. Kromě údajů, které jsou potřeba pro reprezentaci stromu, bude navíc každý uzel tohoto stromu držet obdélníkový kvadrant *R(v)*, který reprezentuje. Každý list čtyřstromu bude držet ukazatele na listy obdélníků, které

obsahuje. Poté je aplikován algoritmus pro zjištění překryvů částí úseček.

2.7 Způsoby detekování kolizí

Detekce kolizí je spíše doménou počítačové grafiky a animací. Při použití v dopravě je třeba brát v potaz další faktory jako pozice kamery, úhel pohledu, situace (křižovatka, nadjezd, most, ..) či přesnost detekce. Navíc, na rozdíl od animací, je třeba pracovat se spousty proměnnými, které je potřeba zjistit – pozice vozidla ve scéně, chování vozidla v čase, apod. Při tom je třeba velice bedlivě sledovat výstupy z detektoru, aby se nestalo, že v důsledku zvýšené iluminace, větrných podmínek či změny počasí, nedojde k detekci vozidla v obraze a v důsledku toho se systém stane nespolehlivým. V neposlední řadě, většina již navržených systému pro detekci kolizí není řešena tak, aby pracovala jako systém v dohledových kamerách, ale jako systém pracující přímo na/ve vozidle.

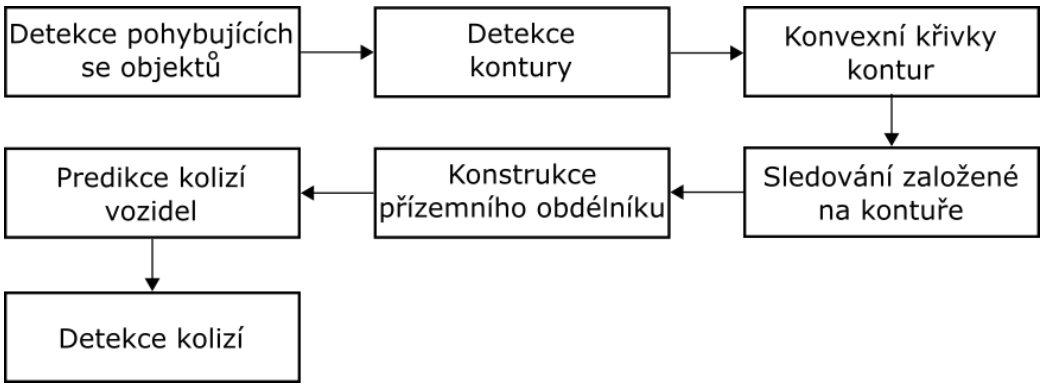
Kapitola 3

Návrh systému pro detekci kolizí

Prvním krokem v práci s pohybujícími se objekty ve scéně je detekce těchto objektů. Pro detekci pozadí je použit algoritmus mixu Gaussových rozložení, pomocí kterého je modelováno pozadí a jehož výstupem je maska popředí. Sledování takto detekovaných vozidel je poté prováděno na základě jejich kontury, viz sekce 2.5.3. V dalším kroku je zkonstruován obdélník kolem přízemní části vozidla, na základě něhož jsou predikovány a poté detekovány kolize. Hrubý postup lze vidět na obrázku 3.1.

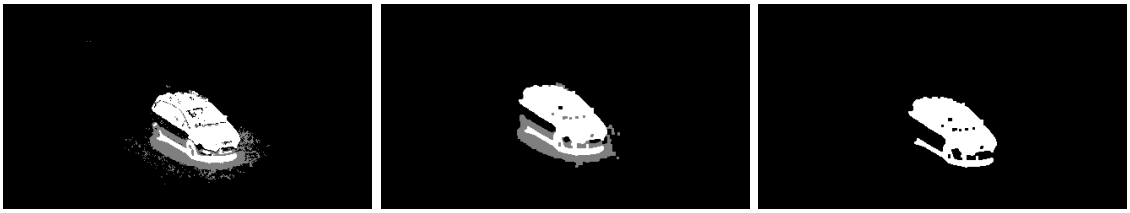
3.1 Detekce vozidel ve videu

Pro vyšší rychlost zpracování obrazu, je nejprve sníženo rozlišení obrazu na úkor kvality získané masky. Při fullHD rozlišení (1920px × 1080px) je hranice tohoto faktoru nastavena na hodnotu 0,3. To znamená, že obraz je tedy 3x menší (při fullHD vstupu tedy 576px × 324px). Při nižším rozlišení dochází k velké deformaci výstupní masky. Pro odstranění počátečního šumu (vyhlazení) je taktéž každý snímek rozostřen pomocí mediánového filtru. Medián je velikosti 3 × 3. Výsledek lze vidět na obrázku 3.2. Pro detekci vozidel byl zvolen algoritmus mixu Gaussových rozložení. Jeho vstupem je vyhlazený snímek a výstupem maska popředí. Hlavním problémem této masky je fakt, že není dokonalá. Proto jsou na tuto masku aplikovány morfologické operace – eroze a dilatace. Respektive operace otevření a uzavření. První se aplikuje otevření. Poté je provedeno uzavření. V konečném kroku jsou odstraněny stíny. Stíny jsou označeny šedou barvou s hodnotou prahu 127. Po prahování obrazu dostáváme



Obrázek 3.1: Návrhu postupu řešení.

Obrázek 3.2: Snímek se sníženým rozlišením (vlevo), vyhlazený snímek (vpravo).



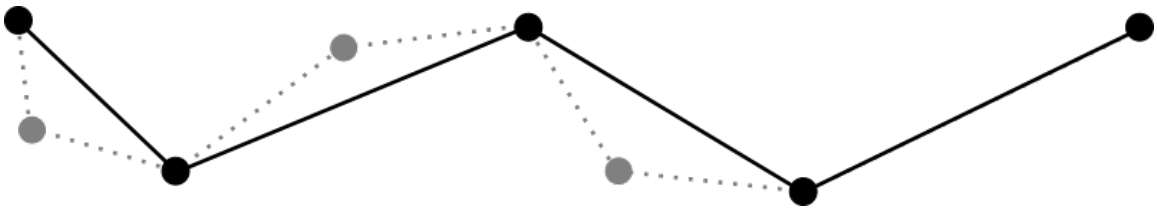
Obrázek 3.3: Výstupní maska detektoru (vlevo), maska po aplikaci morfologických operací (střed) a výsledná maska po odstranění stínů (vpravo).

výslednou masku, viz obrázek 3.3. Maska obsahuje jen pohybující se objekty (vozidla). Nejednotnost masky je způsobena změnami v osvětlení a nepřesnou detekcí stínů. Tyto problémy jsou řešeny v sekci 3.2.

3.2 Získání a úprava kontury

Po odstranění stínů v obraze je na řadě získání kontur vozidel. V úvahu jsou brány jen vnější kontury objektu, čímž se vyhneme časovým ztrátám v průběhu hledání kontur uvnitř objektu. Kontura, která je získaná metodou sledování okrajů (Border-Following). Tato metoda však vrací příliš mnoho bodů, které nejsou potřeba. Proto, za využití Ramer-Douglas-Peucker algoritmu, viz obrázek 3.4, je snížen počet bodů pro rychlejší práci s konturou.

Kontura takto získaná je stále imperfektní, viz obrázek 3.5, pro další použití, proto je ještě výsledná kontura aproximována konvexní křivkou, která vytvoří výslednou konturu, se kterou se dá lépe pracovat. Specifické problémy, které vznikají při rozpadu kontury, jsou řešeny v sekci 3.3.2.



Obrázek 3.4: Podstata Ramer-Douglas-Peucker algoritmu.

Obrázek 3.5: Imperfektní kontura (vlevo), názorná kontura po dílčích úpravách (vpravo).

3.3 Sledování vozidel pomocí kontury

Při sledování jednotlivých vozidel je využito sledování pomocí kontury, viz obrázek 2.5. Aby bylo možné sledovat vozidla podle kontury, je potřeba vozidlo přiřadit do pruhu, a poté se snažit konturu co nejvíce zrekonstruovat. To znamená, že pokud dojde k roztříštění kontury, spojit ji. V případě, že by došlo ke spojení kontur – překrytí vozidel při předjíždění – snažit se zabránit tomuto spojení.

3.3.1 Přiřazení vozidla k pruhu vozovky

V prvním kroku je každé kontuře vozidla vypočítán její moment. Tyto momenty jsou důležité pro získání těžiště každé kontury. Středové momenty jsou spočítány následovně:

$$m_{ij} = \sum_{x,y} (pole(x,y) \cdot (x - \overline{x})^j \cdot (y - \overline{y})^i) \tag{3.1}$$

kde $(\overline{x}, \overline{y})$ je těžiště:

$$\overline{x} = \frac{m_{10}}{m_{00}}, \overline{y} = \frac{m_{01}}{m_{00}} \tag{3.2}$$

Těžiště kontur jsou nápomocny při identifikaci vozidla, ve kterém pruhu se pohybuje. Pruhy na vozovce jsou vstupem systému, tedy musejí být získány apriori a systém si je nezískává sám. Těžiště představuje bod, který bude-li posunut na dolní hranici kontury, bude zcela určitě ležet v pruhu, ve kterém se pohybuje vozidlo.

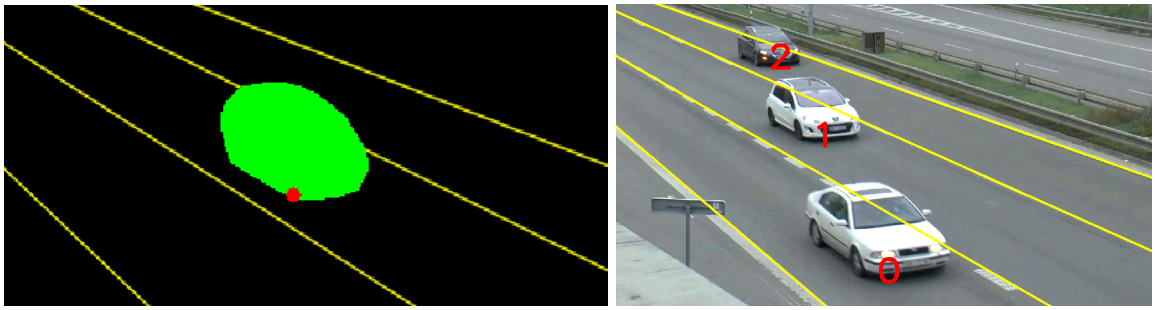
Těžiště představuje bod $P[x, y]$ a rozměry snímku jsou (x', y') s počátkem $A[0, 0]$ vlevo nahoře snímku, můžeme bod P posunout pomocí metody půlení intervalu, na dolní hranici kontury. Interval pro metodu půlení intervalu je v tomto případě $\langle y, y' \rangle$. Metoda půlení intervalu je velmi efektivní, neboť vypočtení nové pozice bodu tak, aby byl na hranici kontury trvá nejvíce devět iterací.

Nyní bod $P[x, y]$ leží přímo v pruhu vozovky. Jeho pozici určíme dle vzdálenosti od čar na vozovce. Z dvou bodů, $A[x, y]$ a $B[k, l]$, každé čáry vytvoříme obecnou rovnici ve tvaru $p : ax + by + c = 0$. Její koeficienty a, c vypočteme následovně (koeficient b je vždy roven 1):

$$a = \frac{l - y}{k - x}, c = a + \frac{x \cdot y - x \cdot l}{l - x} \tag{3.3}$$

Poté se spočítá vzdálenost bodu $C[x, y]$ od přímky p:

$$v(C, p) = \frac{|a \cdot x + y + c|}{\sqrt{a^2 + 1^2}} \tag{3.4}$$



Obrázek 3.6: Bod posunutý na hranici kontury (vlevo), přiřazená vozidla do pruhů (vlevo).

Dvě nejkratší vzdálenosti jsou uchovány a podle těchto je vozidlo přiděleno do pruhu. Příkladně, pokud dvě nejkratší vzdálenosti jsou od čar s indexem 0 a 1, vozidlo jede v levém pruhu. Výsledek lze vidět na obrázku 3.6. V této fázi, pro další zrychlení systému, jsou odstraněny vozidla jedoucí v protějším směru. Každý objekt, jehož vzdálenost k čáře úplně vpravo (z pohledu od kamery) je nižší, než vzdálenost k čáře, která je vlevo od ní, je objekt v opačném směru. Této podmínce vyhovují i objekty, které jsou v dálce, proto je zavedena i tolerance, která zajistí, že tyto objekty nebudou odstraněny.

3.3.2 Algoritmus sledování více vozidel

Sledování vozidel probíhá pomocí kontury, viz obrázek 2.5. S každou konturou vozidla je zacházeno jako s individuálním objektem. O tomto objektu jsou uloženy podstatné informace jako je: těžiště, historie pohybu, pozice v pruhu a jeho podstava, viz sekce 3.4.1.

Identita objektu ve snímku v čase t se hodnotí na základě částečného překryvu s objektem ve snímku v čase $t-1$. Pokud se tedy tyto dva objekty překrývají, resp. jejich kontury, je objekt v čase t považován za tentýž jako objekt v čase $t-1$.

V době porovnávání objekt k objektu, může nastat několik situací (porovnávají se objekty v aktuálním snímku s objekty již detekovanými ve snímcích předešlých):

1. Nedošlo k žádnému překryvu – to znamená, že byl detekován nový objekt, nebo objekt zmizel z obrazovky. Toto je vyřešeno tzv. životností objektu. Pokud se daný objekt neobjeví v dalších n snímcích, je považován za zmizelý.

2. Došlo k jednomu překryvu – to znamená, že byl nalezen objekt z předešlého snímku v aktuálním snímku. Jsou aktualizovány relevantní vlastnosti objektu.

3. Došlo ke dvěma překryvům – to znamená, že dvě kontury se slily dohromady a přistupuje se k nim dvěma způsoby:

   (a) Každá kontura je v jiném pruhu, viz obrázek 3.7 – v tomto případě nejsou objekty aktualizovány, ale jsou posunuty na základě historie pohybu. Pokud známe vzdálenost a směr, kterým objekt cestoval v minulosti, je očekáváno, že tento objekt bude pokračovat po této dráze a urazí podobnou vzdálenost v budoucnu.

   $$vzdalenost = pozice_t - pozice_{t-1} \tag{3.5}$$

   Objekt je poté posunut a nedojde ke splynutí kontur. Pokud historie pohybu není známá, třeba ve chvíli kdy objekt je detekován a hned další snímek by chtěl splynout, převezme historii pohybu od nejbližšího objektu. Pokud žádný takový objekt ve snímku není, je mu historie přiřazena.

Obrázek 3.7: Někdy má kontura tendenci se spojit dohromady (vlevo), nebo roztříštit se na menší části (vpravo).

(b) Obě kontury jsou v jednom pruhu – v tomto případě je spočítána plocha kontur a pokud jejich poměr je od 0,7 do 1,3, pravděpodobně se jedná o dva podobné objekty co do velikosti, a je to první indikátor na cestě k detekci kolize. Pokud je tento poměr mimo zadaný rozsah, došlo ke spojení jinak roztříštěné kontury, staré objekty jsou odstraněny a nahrazeny novým objekt.

4. Došlo k více jak dvěma překryvům – nejprve se zkontroluje, zda všechny objekty jsou v jednom pruhu. Pokud ne, sleduje se dále poměr k oblastem mimo pruh. Pokud je počet objektů, který je mimo pruh menší než polovina objektů v tom samém pruhu, je tento objekt považován za jeden a všechny staré objekty jsou odstraněny a nahrazeny nově detekovaným pospolitým objektem.

3.4 Predikce vzájemných kolizí

Po důsledné detekci kontury vozidla lze provádět predikce kolizí. Tyto kolize jsou predikovány na základě sestrojení obdélníku kolem přízemní části vozidla, jakožto části, kde dochází k dotyku při kolizi. Pomocí těchto obdélníků jsou později predikovány kolize.

3.4.1 Konstrukce obdélníku kolem přízemní části vozidla

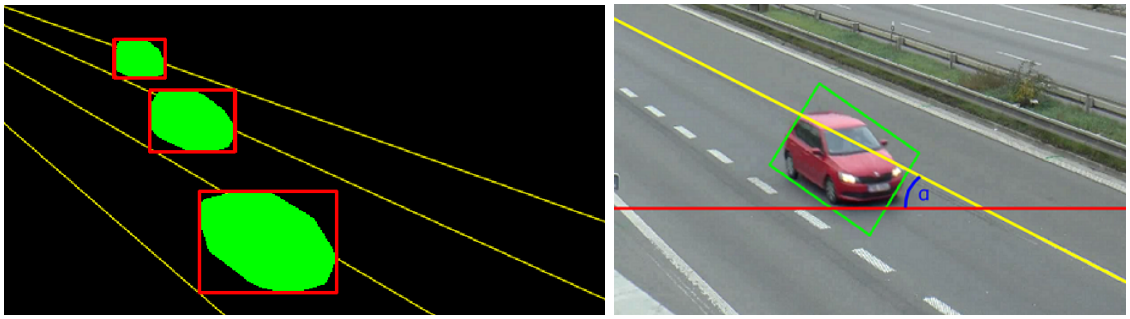
Konstrukce obdélníku kolem přízemní části vozidla se provádí v několika krocích:

1. Konstrukce bounding boxu (obdélníku) kolem objektu/kontury – v prvním kroku je třeba jako základ vytvořit nejmenší možný obdélník, který obepíná konturu, viz obrázek 3.8.

2. Otočení obdélníku ve směru jízdy – pro otočení obdélníku ve směru jízdy je potřeba vypočítat úhel natočení. Dle rovnic 3.3 se vytvoří obecné rovnice přímek – pruhu silnice a horizontální přímky, procházející bodem na hranici kontury vozidla. Z přímek jsou poté vytvořeny vektory $\vec{\mathbf{u}} = (u_1, u_2)$ a $\vec{\mathbf{v}} = (v_1, v_2)$. Cosinus úhlu α sevřený těmito vektory se poté vypočítá následovně:

$$\cos\alpha = \frac{\vec{\mathbf{u}} \cdot \vec{\mathbf{v}}}{|\vec{\mathbf{u}}| \cdot |\vec{\mathbf{v}}|} = \frac{u_1 v_1 + u_2 v_2}{\sqrt{u_1^2 + u_2^2} \cdot \sqrt{v_1^2 + v_2^2}} \tag{3.6}$$

Původní obdélník je poté natočen o úhel α.



Obrázek 3.8: Sestrojení nejmenšího možného obdélníku kolem kontury (vlevo), vypočtení úhlu α natočení obdélníku, žlutá – čára na okraji pruhu, červená – horizontální přímka, zelený – již natočený obdélník (vpravo).

3. Sestrojení přední části přízemního obdélníku – ve 2D pohledu se velice těžce kreslí úhel, který je pravý v 3D pohledu. Po několika testováních a pokusech byla tato přední část sestrojena tak, jak lze vidět na obrázku 3.9. Body A a B tvoří vektor $\vec{\mathbf{u}} = (u_1, u_2)$. Přední část obdélníku je poté jen normálový vektor $\vec{\mathbf{v}} = (v_1, v_2)$ vektoru $\vec{\mathbf{u}}$, který je kolmý na vektor $\vec{\mathbf{u}}$. Pomocí tohoto vektoru je zase vytvořena obecná rovnice přímky, která jej reprezentuje. Tato přímka je poté metodou půlení intervalu posunuta tak, aby se těsně dotýkala kontury vozidla. Průnik této přímky se spodní částí otočeného obdélníku je druhým bodem obdélníku přízemního. Prvním bodem je bod C, resp. N, viz obrázek 3.9.

4. Konstrukce zadní části přízemního obdélníku – zadní část přízemního obdélníku je rovnoběžná s tou přední. Zadní část tedy má vektor s počátkem v bodě C, viz obrázek 3.9. Další bod obdélníku je zkonstruován jako průsečík zadní přímky s čárou na vozovce. Pokud máme obě přímky p,q ve tvaru $y = ax + c$, resp. $y = bx + d$. Vypočteme jejich průsečík $A[x, y]$ následovně:

$$x = \frac{d - c}{a - b}, y = \frac{a \cdot d - b \cdot c}{a - b} \tag{3.7}$$

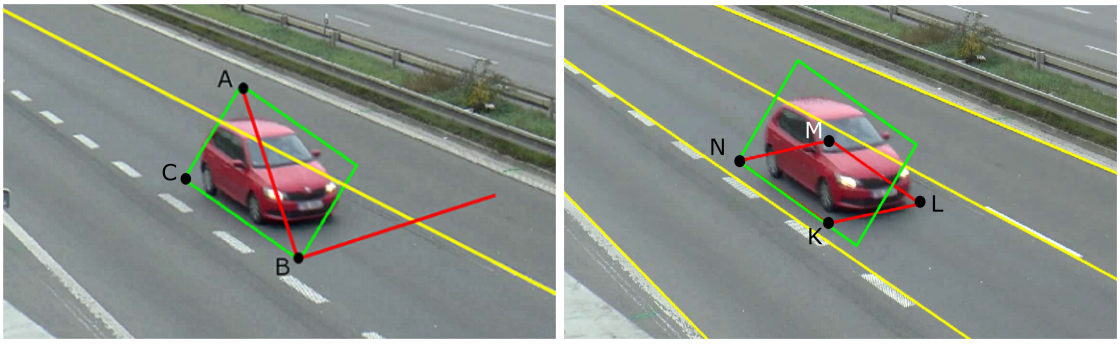
5. Vypočtení čtvrtého bodu obdélníku – čtvrtý bod obdélníku (viz bod L, obrázek 3.9), je vypočítán tak, že se vytvoří vektor úsečky NK, posune se do bodu M, vytvoří se její obecná rovnice a vypočte se průnik s přední částí obdélníku, jež začíná v bodě K a končí na pruhu silnice. Tento průsečík je bodem L.

6. Úprava bodu M – bod M ve výsledku neleží na čáře pruhu, jeho pozice je upravena tak, aby délky vektorů KL a NM byly shodné.

Z popsaného algoritmu a konstrukce vychází jeden závěr, a to že navržený algoritmus funguje jen pro boční pohledy kamery (levý a pravý pohled), nikoli však pro čelní pohled. Při pohledu zepředu nelze úplně perfektně sestrojit přízemní obdélník, jelikož jeho zadní část by obepínala vrchní část střechy auta a algoritmus v této situaci, tak jak je navržen, není úplně funkční.

3.4.2 Způsob predikování kolize

Po zkonstruování přízemního obdélníku, máme jistou představu o pozici vozidla (přesněji jeho podstavy), viz obrázek 3.10. Poté navržený algoritmus pro predikci kolizí je takový, že



Obrázek 3.9: Způsob sestrojení přední části přízemního obdélníku s body A a B reprezentující úhlopříčku a bodem C jako výchozí bod pro zadní část obdélníku (vlevo), finální červený přízemní obdélník $KLMN$ s výchozím zeleným obdélníkem (vpravo).

zvětšením tohoto obdélníku definujeme jakési ochranné pásmo kolem vozidla kde, pokud se vyskytuje další vozidlo, ny mohlo eventuálně dojít ke kolizi. Faktor zvětšení je zvolen 1,5. Predikce kolizí se poté provádí pomocí kontroly překryvů zvětšeného obdélníku a ostatních obdélníků zkonstruovaných v danou chvíli. K problému je přistupováno metodou průniku úseček.

Algoritmus vždy každou stranu zvětšeného obdélníku testuje ke každé straně všech vyskytujících se obdélníků, které jsou sestrojeny kolem přízemní části každého vozidla. Testování probíhá ve dvou krocích:

1. Zjištění průsečíku dvou přímek – nejprve je zjištěn průsečík přímek pomocí rovnic 3.7.

2. Kontrola, zda průsečík leží na úsečkách – nalezený průsečík je testován, zda-li leží na obou úsečkách.

Průsečík dvou úseček je nalezen, vyhoví-li v obou krocích. Tento čistě matematický přístup má oproti přístupu, kdy lze počítat průniky úseček vykreslením do dvou obrázků a po provedení operace logický and zjistit průniky, výhodu tu, že je více jak 10krát rychlejší. Další urychlení algoritmu je způsobeno tím, že každé dva objekty jsou testovány na průniky pouze jednou.

Pokud při vzájemném porovnání dojde k nejméně dvěma průnikům, jsou objekty označeny a poté testovány na případnou kolizi, viz sekce 3.5.

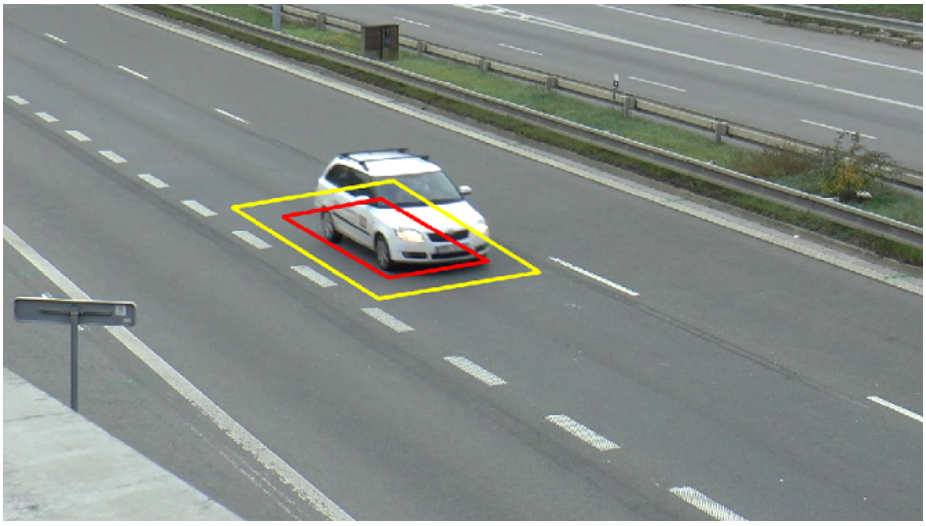
3.5 Detekce kolizí

Za předpokladu, že jsme předešli spojení kontur v sekci 3.3.2, můžeme definovat podmínky pro detekci kolizí vozidel. V sekci 3.3.2 je již částečně naznačen postup detekce kolize. V sekci 3.4.2 byla vozidla, která by mohla kolidovat, označena. Tento fakt je vstupní podmínkou pro další části algoritmu. Podmínky jsou poté definovány následovně:

1. Jsou-li na stejné pozici ve snímku v čase $t-1$ detekovány dva bloby a ve snímku v čase t detekován jen jeden.

2. A zároveň jsou bloby v čase $t-1$ přibližně stejné plochy (poměr od 0,7 do 1,3).

3. A zároveň bloby v čase $t-1$ a t jsou v jednom pruhu nebo v pruzích, které jsou bezprostředně vedle sebe.



Obrázek 3.10: Zvětšený obdélník pro predikci kolizí (žlutý).

Za splnění výše zmíněných podmínek, společně s predikovanou kolizí vozidel, můžeme tvrdit, že došlo ke kolizi. V tomto případě ke kolizi, kdy jedno vozidlo narazilo do druhého zezadu. Druhá část podmínky v bodě 3. pokrývá i kolizi v případě předjíždění. Podmínkám by vyhovovalo i vozidlo, které by případně přeletělo přes svodidla z protějšího pruhu. Došlo by tak k čelnímu nárazu, ale z hlediska systému ke stejnému typu kolize.

Poměr zvolený v podmínce v bodě 2. je důležitý. Neboť v případě spojení dříve rozpadlé kontury vozidla, kdy dojde ke spojení např. prahu vozidla s jeho zbylým celkem, by poměr ploch byl např. 2:7 (popř. 7:2), tedy 0,3 (resp. 3,5). Problém, který nastává při takto zvolených poměrech je v případě, kdy dojde ke kolizi většího vozidla (např. kamión, nákladní vozidlo) s osobním automobilem. V takové situaci bude poměr ploch zainteresovaných vozidel mimo zvolený rozsah pro detekci kolizí.

Kapitola 4

Vyhodnocení navrženého systému

Vzhledem k tomu, že se nepodařilo sehnat testovací sadu přímo pro otestování správnosti detekce kolizí, je systém vyhodnocen z hlediska přesnosti a relevantnosti zrekonstruovaného obdélníku kolem přízemní části vozidla. Tento obdélník je totiž základní částí celého systému pro detekci kolizí. Jeho konstrukce závisí na míře správně detekovaných objektů a sestrojených kontur kolem nich. Systém je vyhodnocen pomocí dvou ukazatelů Detection Rate a Recall. Taktéž je vyhodnocena rychlost celkového systému. Ukázku správně a špatně zkonstruovaného obdélníku lze vidět na obrázku 4.1.

4.1 Přesnost navrženého systému

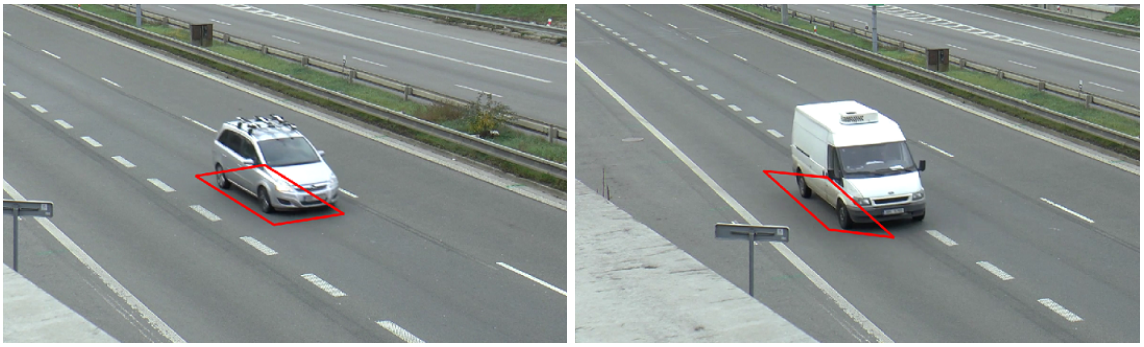
Detection Rate – je poměr správně zkonstruovaných obdélníků ke všem zkonstruovaným obdélníkům. Je definována následovně:

$$DetectionRate = \frac{TruePositive}{TruePositive + FalsePositive} \tag{4.1}$$

Recall – je poměr správně zkonstruovaných obdélníků ke všem vozidlům, které měly mít kolem sebe zkonstruované obdélníky. Je definován následovně:

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative} \tag{4.2}$$

Z tabulky 4.1, lze vidět, že úspěšnost systému je 72 %.



Obrázek 4.1: Příklad správně zkonstruovaného obdélníků (vlevo) a špatně zkonstruovaný obdélník (vpravo).

Tabulka 4.1: Výkon systému

	Detection Rate	Recall
Implementovaný systém	0.722	0.726

4.2 Výpočetní rychlost systému

Rychlost C++ implementace byla vyhodnocena na 25 minutách videí (za použití sestavy Intel i5-4200 2.5 GHz procesor a 8 GB DDR3 RAM). Při fullHD rozlišení (1920px $\times$ 1080px), kdy jsou započítávány všechny úpravy snímku včetně snížení rozlišení, vyhlazení apod., bylo dosaženo rychlosti zpracování 29 FPS při frekvenci průjezdu 15 vozidel za minutu.

Kapitola 5

Závěr

V této bakalářské práci jsme navrhl a implementoval systém pro detekci kolizí vozidel. Hlavní výhodou systému je přístup k predikci kolizí za pomocí konstruovaného přízemního obdélníku. Důležité je, že v tomto procesu bylo dosaženo 72% úspěšnosti v přesné konstrukci přízemního obdélníku.

Kdybych na začátku své práce věděl všechny nevýhody při použití kontury jako prostředku pro sledování chování objektů v obraze, zvolil bych jinou metodu. Volba jiné metody (např. sledování na základě barvy vozidla) by napomohla při řešení částečného překrývání kontur a kvalitnější analýzy chování vozidel.

Velkou výhodou, co se týče detekcí kolizí by bylo do budoucna využití Kalmanova filtru, který by se periodicky aktualizoval individuálně pro každé vozidlo a pomocí něhož by se daly kvalitněji předpovídat kolize. Vycházejíc z předpokladu, že vozidla po kolizi zastaví, mohl by systém taktéž být vylepšen o detekci zastavení po detekci kolize vozidel. Snížil by se tím počet potenciálně špatně detekovaných kolizí. Velikým vylepšením by bylo použití dvou a více kamer pro sledování vozidel. Sledování objektů z více kamer má tu výhodu, že lze odhadnout i hloubku obrazu, tedy objekty lze rozlišit na základě jiných vzdáleností i když se jejich kontury překrývají. Obecnějšími vylepšeními systému by byla jeho modularita. Implementace modulů pro sběr statistických dat (počet průjezdů vozidel), délka, typ, rychlost vozidla aj.

Co se týče vylepšení z hlediska rychlosti systému. Zrychlení by přineslo přenesení výpočtů na grafické jádro, stejně tak jako využití multi-vláknového zpracování, kdy jeden snímek by byl analyzován, zatímco následující snímek by byl mezitím předzpracován.

I přesto, že poslední část systému je implementována, ale nebylo ji z důvodů nedostatku testovacích dat možné reálně otestovat, byly všechny body zadání splněny v plném rozsahu.

Literatura

[1] How to Use Background Subtraction Methods. 2015.
URL http://docs.opencv.org/3.1.0/d1/dc5/tutorial_background_subtraction.html

[2] Atev, S.; Arumugam, H.; Masoud, O.; aj.: A vision-based approach to collision prediction at traffic intersections. *Intelligent Transportation Systems, IEEE Transactions on*, ročník 6, č. 4, 2005: s. 416–423.

[3] Beymer, D.; Konolige, K.: Real-time tracking of multiple people using continuous detection. In *IEEE Frame Rate Workshop*, Citeseer, 1999.

[4] Bouwmans, T.: Traditional and recent approaches in background modeling for foreground detection: An overview. *Computer Science Review*, ročník 11, 2014: s. 31–66.

[5] Canny, J.: A computational approach to edge detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, , č. 6, 1986: s. 679–698.

[6] Cucchiara, R.; Grana, C.; Piccardi, M.; aj.: Detecting moving objects, ghosts, and shadows in video streams. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, ročník 25, č. 10, 2003: s. 1337–1342.

[7] Elgammal, A.; Harwood, D.; Davis, L.: Non-parametric model for background subtraction. In *Computer Vision—ECCV 2000*, Springer, 2000, s. 751–767.

[8] Graham, R. L.; Yao, F. F.: Finding the convex hull of a simple polygon. *Journal of Algorithms*, ročník 4, č. 4, 1983: s. 324–331.

[9] Green, G.: *An essay on the application of mathematical analysis to the theories of electricity and magnetism*. author, 1828.

[10] Hu, W.; Xiao, X.; Xie, D.; aj.: Traffic accident prediction using 3-D model-based vehicle tracking. *Vehicular Technology, IEEE Transactions on*, ročník 53, č. 3, 2004: s. 677–694.

[11] Kang, J.; Cohen, I.; Medioni, G.: Continuous tracking within and across camera streams. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, ročník 1, IEEE, 2003, s. I–267.

[12] Kuhn, H. W.: The Hungarian method for the assignment problem. *Naval research logistics quarterly*, ročník 2, č. 1-2, 1955: s. 83–97.

[13] Piccardi, M.: Background subtraction techniques: a review. In *Systems, man and cybernetics, 2004 IEEE international conference on*, ročník 4, IEEE, 2004, s. 3099–3104.

[14] Saunier, N.; Sayed, T.; Lim, C.: Probabilistic collision prediction for vision-based automated road safety analysis. In *Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE*, IEEE, 2007, s. 872–878.

[15] Seki, M.; Wada, T.; Fujiwara, H.; aj.: Background subtraction based on cooccurrence of image variations. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, ročník 2, IEEE, 2003, s. II–65.

[16] Stauffer, C.; Grimson, W. E. L.: Adaptive background mixture models for real-time tracking. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, ročník 2, IEEE, 1999.

[17] Suzuki, S.; aj.: Topological structural analysis of digitized binary images by border following. *Computer Vision, Graphics, and Image Processing*, ročník 30, č. 1, 1985: s. 32–46.

[18] Tao, H.; Sawhney, H. S.; Kumar, R.: Object tracking with bayesian estimation of dynamic layer representations. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, ročník 24, č. 1, 2002: s. 75–89.

[19] Šikudová, E.; Černeková, Z.; Benešová, W.; aj.: *Počítačové videnie Detekcia a rozpoznávanie objektov*. Praha: Wikina,Praha, 2014, ISBN 978-80-87925-06-5, 378 s.

[20] Wren, C. R.; Azarbayejani, A.; Darrell, T.; aj.: Pfinder: Real-time tracking of the human body. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, ročník 19, č. 7, 1997: s. 780–785.

[21] Yilmaz, A.; Javed, O.; Shah, M.: Object tracking: A survey. *Acm computing surveys (CSUR)*, ročník 38, č. 4, 2006: str. 13.