VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

ODHAD 3D POZICE VOZIDEL Z DOPRAVNÍCH KAMER

VEHICLE 3D POSE ESTIMATION FROM TRAFFIC CAMERAS

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE **ONDŘEJ POSPÍŠIL**
AUTHOR

VEDOUCÍ PRÁCE **Ing. MICHAL HRADIŠ, Ph.D.**
SUPERVISOR

BRNO 2020

Vysoké učení technické v Brně
Fakulta informačních technologií

Ústav počítačové grafiky a multimédií (UPGM) Akademický rok 2019/2020

Zadání bakalářské práce

Student: **Pospíšil Ondřej**
Program: Informační technologie
Název: **Odhad 3D pozice vozidel z dopravních kamer**
Vehicle 3D Pose Estimation form Traffic Cameras
Kategorie: Zpracování obrazu

Zadání:

1. Prostudujte základy konvolučních sítí a detekce objektů.
2. Vytvořte si přehled o současných metodách detekce vozidel v záznamech z dopravních kamer se zaměřením na určování jejich pozice ve světových souřadnicích.
3. Vyberte nebo navrhněte metodu aplikovatelnou na detekci a sledování vozidel například při průjezdu křižovatkou.
4. Obstarejte si databázi vhodnou pro experimenty.
5. Implementujte navrženou metodu a proveďte experimenty nad datovou sadou.
6. Porovnejte dosažené výsledky a diskutujte možnosti budoucího vývoje.
7. Vytvořte stručné video prezentující vaši práci, její cíle a výsledky.

Literatura:

- Krizhevsky, A., Sutskever, I. and Hinton, G. E.: ImageNet Classification with Deep Convolutional Neural Networks, NIPS 2012
- Mousavian, Arsalan & Anguelov, Dragomir & Flynn, John & Košecká, Jana:. 3D Bounding Box Estimation Using Deep Learning and Geometry. CVPR, 2017.

Pro udělení zápočtu za první semestr je požadováno:

- Body 1 až 3.

Podrobné závazné pokyny pro vypracování práce viz https://www.fit.vut.cz/study/theses/

Vedoucí práce: **Hradiš Michal, Ing., Ph.D.**
Vedoucí ústavu: Černocký Jan, doc. Dr. Ing.
Datum zadání: 1. listopadu 2019
Datum odevzdání: 28. května 2020
Datum schválení: 1. listopadu 2019

Abstrakt

Cílem této bakalářské práce je vytvořit metodu pro odhad 3D pozice vozidel z dopravních kamer. V práci jsou probrány existující metody pro detekci a odhad pozice vozidel. Součástí práce je i sestavení datové sady pro trénování a experimenty nad navrženou metodou pro odhad pozice vozidel. Navržená metoda používá konvoluční neuronovou síť pro regresi podstavy vozidla na obrázku. Pozice vozidla je poté promítnuta do roviny silnice pomocí homografie. Experimenty shrnují trénovaní a vyhodnocení metody pro odhad pozice a přesnosti ruční anotace pozice.

Abstract

The goal of this bachelor thesis is to create a method for the 3D pose estimation of vehicles from traffic cameras. Existing methods for the car detection and the pose estimation of vehicles are described. Part of the thesis was to build a dataset for the purpose of training and experiments on the proposed car pose estimation method. Proposed method uses a convolutional neural network for regression of the car base in the image. Car pose is then projected into the road plane using homography. Experiments summarize training and the evaluation of the car pose estimation method and accuracy of manual vehicle annotation.

Klíčová slova

odhad pozice vozidla, detekce vozidla, dopravní kamera, konvoluční neuronové sítě, homografie

Keywords

car pose estimation, car detection, traffic camera, convolutional neural networks, homography

Citace

POSPÍŠIL, Ondřej. *Odhad 3D pozice vozidel z dopravních kamer*. Brno, 2020. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Michal Hradiš, Ph.D.

Odhad 3D pozice vozidel z dopravních kamer

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Michala Hradiše, Ph.D. Uvedl jsem všechny literární prameny, publikace a další zdroje, ze kterých jsem čerpal.

.....
Ondřej Pospíšil
27. května 2020

Poděkování

Chtěl bych poděkovat vedoucímu práce Ing. Michalu Hradišovi, Ph.D. za cenné rady, věnovaný čas a ochotu při odborném vedení této práce. Dále bych chtěl poděkovat rodičům a přítelkyni za morální podporu.

Obsah

Kapitola 1

Úvod

Zpracování obrazu často nachází uplatnění v oblasti dopravy. Pokroky v oblasti strojového učení a zlepšování parametrů výpočetní techniky vede stále k většímu množství informací, které je možné získat z fotografie vozidel. Nejčastější úlohou, kterou je potřeba vyřešit v dopravě, je detekce automobilu, přičemž je často uváděná poloha automobilu na daném snímku. Potřeba odhadnutí polohy automobilu přímo na silnici, na křižovatce, na ulici nebo na parkovišti vznikla až s příchodem autonomních vozidel, které potřebují znát polohu ostatních automobilů na silnici z důvodu případných kolizí. Polohu vozidel lze však využít i v dopravních kamerách sledující křižovatky nebo parkoviště. Cílem této bakalářské práce je navrhnout metodu pro odhad pozic vozidel v reálných souřadnicích země. Metoda by měla být schopna zpracovat snímek pořízený z kamery snímající určitou dopravní situaci a určit pozici jednotlivých vozidel v rovině silnice.

Po úvodní kapitole následuje kapitola obsahující popis existujících metod používaných pro odhad pozice vozidel a to zejména pomocí klíčových bodů nebo regrese. Kapitola obsahuje i shrnutí nejpoužívanějších metod pro detekci vozidel na snímku. Zmíněny jsou zde i existující datové sady obsahující informaci o poloze vozidla.

Třetí kapitola se věnuje sestavení vlastní datové sady, která je složena ze snímků z více zdrojů. Tato datová sada je v dále v práci použita pro experimenty nad navrženou metodou. V této kapitole jsou shrnuty požadavky na datovou sadu, obsah a formát datové sady a použití ruční anotace.

Čtvrtá kapitola obsahuje popis navrhované metody. Popisuje architekturu metody a její jednotlivé části. Zabývá se konkrétním řešením detekce vozidel použitým v této práci, řešením odhadu pozice pomocí konvoluční neuronové sítě a použitím homografie.

Poslední kapitola obsahuje experimenty prováděné v této práci. Obsahuje experiment přesnosti ruční anotace pozice vozidel, který byl prováděn nad skupinou uživatelů v kontextu použití ruční anotace při tvorbě datové sady. Dále zde byl prováděn experiment nad navrženou metodou, který obsahuje trénování neuronové sítě, její vyhodnocení a porovnání s existujícími řešeními. Poslední experiment porovnává výsledky regrese podstavy vozidel s ručními anotacemi. Výsledky tohoto experimenty jsou promítnuty a porovnány v rovině silnice.

Kapitola 2

Metody pro odhad pozice automobilu

Pozici automobilu můžeme vyjadřovat různými způsoby. Nejvíce informací o poloze poskytne vyjádření pomocí 6-DoF [22]. Šest stupňů volnosti reprezentuje polohu automobilu ve formě translace a rotace. Jinou formou vyjádření může být také tzv. 3D bounding-box. 3D box se liší od obyčejného bounding-boxu v tom, že pozice není reprezentována obdélníkovým tvarem ohraničujícím objekt, ale kvádrem.

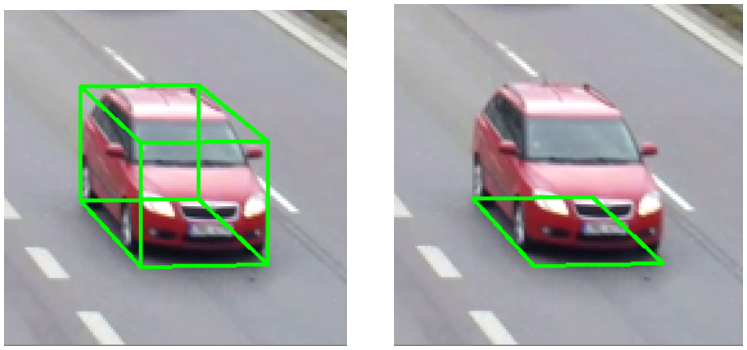
Tato práce pracuje především s částí tohoto kvádru, konkrétně s podstavou. Podstava vozidla totiž udává, kde na silnici se vozidlo nachází, souřadnice bodů určujících podstavu jsou tedy uváděny v rovině země. Podstava vozidla je znázorněna na obrázku 2.1, představit si ji lze jako ohraničující box v souřadném systému silnice, z ptačí perspektivy by se jednalo o obdélník, z pohledu kamery se nám jeví podstava jako rotovaný bounding-box.

V této kapitole jsou shrnuty nejpoužívanější metody detekce vozidel. Dále jsou zde rozebrány metody odhadu pozice vozidel pomocí detekce klíčových bodů a pomocí regrese 3D parametrů. Závěr kapitoly se poté věnuje existujícím datovým sadám. Celá tato kapitola tvoří základ informací, které byly použity pro sestavení datové sady a návrh konkrétní metody odhadu pozice.

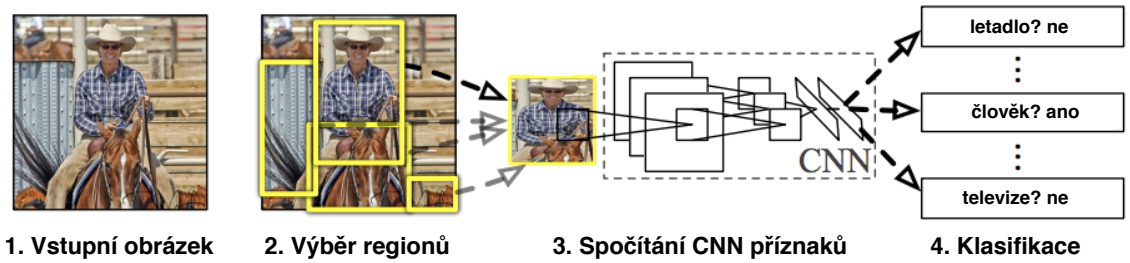
2.1 Detekce vozidel na snímku

Snímek z dopravní kamery obsahuje obvykle velké množství automobilů. Pro další zpracování snímku, je nutné určit pozici jednotlivých vozidel na snímku. K tomu se používají detektory objektů. Výstupem detektoru jsou oblasti snímků, které obsahují danou třídu objektu, v našem případě se jedná o vozidlo.

Detektor objektů se skládá ze tří částí: výběr regionů, extrakce příznaků a klasifikace. Výběr regionů slouží k lokalizaci částí obrázku, kde se daný objekt nachází. Pro vyhledání těchto regionů se používají posuvná okénka různých velikostí. Na jednotlivé regiony se aplikuje extrakce příznaků, jednotlivé příznaky poskytují robustní sémantickou reprezentaci objektu viz [43]. Často se používají Haarovy příznaky [23], HOG [7] nebo SIFT [27]. Na základě těchto příznaků určí klasifikátor třídu objektu, to je potřeba pro více-třídní detektory.



Obrázek 2.1: Reprezentace automobilu 3D bounding-boxem a z něho vycházející reprezentace podstavou na snímku z datasetu BoxCars116 [40].



Obrázek 2.2: Model metody R-CNN, převzatý z článku [15].

R-CNN

S rozvojem hlubokých neuronových sítí byla představena metoda R-CNN [15]. R-CNN se používá pro detekci objektu a využívá přitom konvoluční neuronovou síť. Postup zpracování snímků detektorem R-CNN znázorňuje obrázek 2.2. Pomocí algoritmu je vybráno 2000 regionů, neboli částí obrázku potencionálně obsahující objekt. Z těchto regionů jsou konvoluční sítí extrahovány příznaky a ty předloženy na vstup SVM [6] klasifikátoru viz [15]. Pro výběr regionů se používá algoritmus selektivního vyhledávání, nejprve je provedena sub-segmentace, která vygeneruje mnoho kandidátních regionů, podobné regiony se sloučí a jsou použity pro výběr finálních regionů [13].

Fast R-CNN

Na původní R-CNN navázal její původní autor architekturou Fast R-CNN [16]. Na rozdíl od předchozí verze nevyhodnocuje příznaky pro každý region zvlášť. Na vstup konvoluční neuronové sítě je vložen celý obrázek. Neuronová síť vygeneruje mapu příznaků, reprezentující příznaky z celého obrázku. Pomocí algoritmu selektivního vyhledávání jsou z mapy příznaků vybrány oblasti. Tyto oblasti nazýváme regiony zájmů (RoI). Regiony zájmu jsou poté předloženy na vstup neuronové sítě, která určí třídu objektu a výsledný bounding-box, viz [16]. Trénování metody a zpracování snímků metodou Fast-RCNN je násobně rychlejší než u její předchozí verze. Hlavní důvod je, že konvoluční neuronová síť nemusí u každého obrázku vyhodnocovat 2000 regionů, ale zpracuje celý obrázek naráz.

Metoda	Doba zpracování snímku (s)	Zrychlení	Přesnost (mAP)
R-CNN [15]	50	1x	66.0
Fast R-CNN [16]	2	25x	66.9
Faster R-CNN [35]	0.2	250x	66.9

Tabulka 2.1: Porovnání R-CNN architektur. Tabulka převzatá z článku [36]. Přesnost je vyjádřená jako střední průměrná přesnost (mAP) vyhodnocená z VOC 2007.

Faster R-CNN

R-CNN i Fast R-CNN využívají pro výběr regionů selektivní vyhledávání, které není moc efektivní, proto byl představen algoritmus Faster R-CNN [35]. Extrakce příznaků zůstává stejná jako u Fast R-CNN, ovšem výběr regionů má na starosti neuronová síť. Tento algoritmus je rychlejší než obě předchozí varianty. Srovnání všech variant R-CNN je uvedeno v tabulce 2.1. Rychlost Faster R-CNN umožňuje detekci objektu v reálném čase.

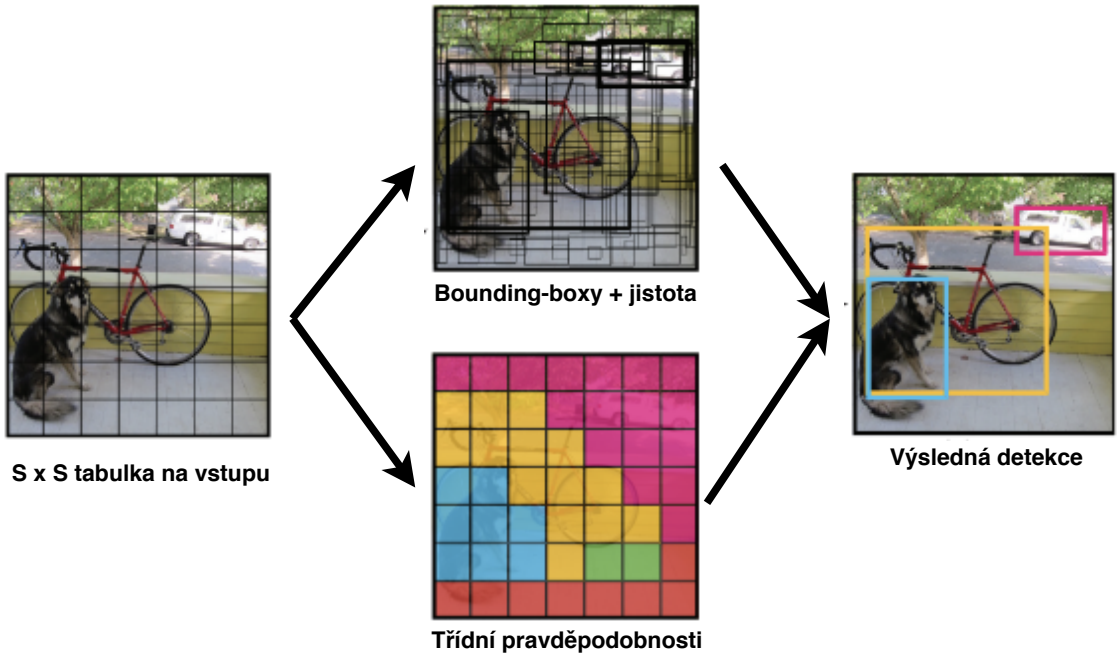
Faster R-CNN lze také zkombinovat s FPN (Feature Pyramid Network) [24]. Místo jedné úrovně příznaků se využívá hned několik úrovní rozlišení příznaků. Jednotlivé úrovně se poté sloučí do finální reprezentace příznaků. FPN nahradí část neuronové sítě použité pro výběr regionů, metoda tak zvládne lépe detekovat i menší objekty. Celkově se zvyšuje přesnost a rychlost oproti nemodifikované metodě Faster R-CNN.

YOLOv1

YOLOv1 (You Only Look Once) [33] je metoda používající konvoluční neuronovou síť. Neuronová síť predikuje bounding-boxy a třídní pravděpodobnosti, nedívá se při tom na celý obrázek, ale pouze na částí s vysokou pravděpodobností výskytu objektu. Postup zpracování snímku metodou YOLO je znázorněn na obrázku 2.3. Obrázek se rozdělí na tabulku o fixní velikosti a pro každou buňku této tabulky je provedena predikce bounding-boxů s jejich jistotou. Zároveň jsou také vyhodnoceny pravděpodobnosti tříd. Určením vhodného prahu jsou vybrány pouze boxy, obsahující objekt s vysokou jistotou. Třída objektu je zvolena na základě pravděpodobností viz [33]. YOLOv1 je rychlejší než Faster R-CNN a je tedy také vhodná pro detekci objektů v reálném čase. Nicméně má problémy s detekcí menších objektů [43].

YOLOv2

YOLOv2 [32] je další verze metody YOLO. Po každé konvoluční vrstvě v neuronové síti je použita normalizace dávek (batch normalization). Metoda také používá tzv. anchor boxy, neboli prvotní odhady bounding-boxu. Navíc je síť trénována na různých velikostech výřezů. Všechny tyto modifikace způsobují vyšší přesnost i rychlost YOLOv2 ve srovnání s předchozí variantou.



Obrázek 2.3: Detekce objektů pomocí YOLO. Obrázek je rozdělen na $S \times S$ tabulku. Pro každou buňku tabulky jsou predikovány bounding-boxy, jistota a třídní pravděpodobnost. Obrázek je převzat z článku [33].

YOLOv3

YOLOv3 [34] oproti předchozí verzi obsahuje hlubší neuronovou síť (106 vrstev) a provádí detekci objektů ve třech různých velikostech. Velikost závisí na tom, z jaké konvoluční vrstvy je použit výstup pro detekci. To pomáhá zlepšit schopnost detekce menších objektů. YOLOv3 používá 9 anchor boxů, 3 pro každou velikost, zvětšuje se tedy počet predikovaných bounding boxů.

SSD

SSD (Single Shot Detector) [26] je jedním z dalších detektorů objektů. Základ architektury tvoří neuronová síť zajišťující extrakci příznaků, často se například používá VGG-16 [38]. Na tuto síť jsou navázány další konvoluční vrstvy zakončené konvolučním filtrem, který poskytuje predikce tříd. Výstupem jsou tedy bounding-boxy spolu s pravděpodobnostmi jednotlivých tříd. Jelikož zpracování snímku konvolučními vrstvami snižuje rozlišení, fungovala pouze detekce větších objektů. Detekce se tedy provádí průběžně při zpracování jednotlivými vrstvami, nejenom na základě výstupu poslední vrstvy. Výstupy jednotlivých konvolučních vrstev jsou také zpracovány filtrem. Tento postup umožňuje detekci menších objektů [18].

Obrázek 2.4: Ukázka výstupu detektoru klíčových bodů a umístění 3D modelu na detekované klíčové body ze článku [21].

2.2 Odhad pozice vozidla na základě detekce klíčových bodů

Problém odhadu pózy osoby se řeší pomocí detekce klíčových bodů. Stejný přístup lze však uplatnit i pro automobily. Samotné klíčové body vozidla poskytují dostatečnou informaci o pozici. Pomocí těchto bodů je možné umístit na snímek 3D model vozidla, to je ukázáno na obrázku 2.4. Model se nasadí na vozidla v místě klíčových bodů. Podmínkou je dostatečné množství detekovaných klíčových bodů. Většina metod je schopna detekovat pouze viditelné body [41], proto může nastat, že k určení pozice bude k dispozici pouze malé množství bodů a výsledná pozice bude nepřesná.

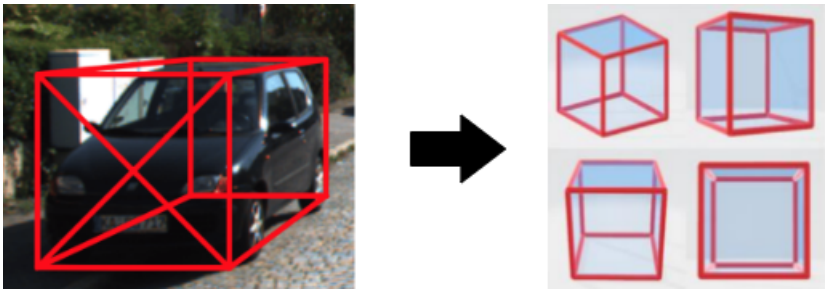
Convolutional Pose Machine

Body lze získat pomocí CPM, neboli Convolutional Pose Machine [42], ty pracují v několika krocích. V prvním kroku je použita konvoluční neuronové síť k vytvoření tzv. heatmap. Heatmapa reprezentuje pravděpodobnost klíčového bodu pro každý pixel. Tato mapa je vytvořena pro každý klíčový bod. V dalších krocích je vylepšována přesnost dané mapy pomocí dalších sítí. Tyto sítě mají opět na vstupu obrázek, který je doplněný o mapu z předchozího neuronové sítě. Tyto kroky zlepšují predikci klíčových bodů viz [2][42].

Datové sady pro detekci klíčových bodů vozidel

Datové sady pro trénování detekce klíčových bodů jsou často tvořeny syntetickými daty nebo ručně ohodnocenými snímky vozidel. Syntetické data mají výhodu snadného určení klíčových bodů, které probíhá určením 2D souřadnic 3D modelu umístěného do snímku. Taková data musí být dostatečně podobná reálným fotografiím vozidel. To je docíleno dostatečně kvalitním 3D modelem, okolní scénou a konečnou úpravou. Takový přístup byl použit v této práci [21]. I přes veškeré syntetické data je nutná ruční anotace klíčových bodů pro doplnění datové sady. Reální data se používají při validaci, ale i při samotném trénování, za účelem zlepšení generalizace. Ruční anotace dat je ovšem časově náročná, některé práce ji však využívají pro tvorbu celé sady viz [41].

Obrázek 2.5: Korespondence 3D boxu a 2D bounding-boxu, obrázek převzat z článku [29].



Obrázek 2.6: Ilustrace klasifikace zorného úhlu, pro ukázku jsou zde zobrazeny pouze čtyři druhy. Obrázek převzatý z článku [12].

2.3 Odhad pozice vozidla pomocí regrese 3D parametrů

Další metodou pro určení pozice vozidla je odhad 3D parametrů pomocí regrese [29]. Metoda používá dvě konvoluční neuronové sítě. První síť je použita pro odhad orientace a druhá pro odhad velikosti 3D bounding-boxu. Kombinací orientace a velikost boxu s dostupným 2D bounding-boxem je určena 3D pozice vozidla viz. obrázek 2.5. Spoléhá se přitom na to, že detektor objektů je natrénován tak, aby určoval bounding-box korespondující s 3D pozicí. To znamená, že 3D pozice lze zobrazit do bounding-boxu určeného detekcí.

MultiBin architektura

Metoda používá tzv. MultiBin architekturu pro odhad orientace boxu. Úhel orientace se rozdělí do více různých překrývajících se kategorií a konvoluční neuronová síť určí pravděpodobnost pro každou kategorii, zda-li v ní leží výsledný úhel. Spolu s pravděpodobností určí rotaci v dané kategorii, pomocí které se určí výsledný úhel orientace.

Přidání dalších 3D parametrů

Na tuto metodu navázala práce [12]. Oproti původní metodě přidává navíc regresi dvou dalších parametrů. Prvním přidaným parametrem je zadní stěna 3D bounding-boxu, konkrétně její středová projekce. Ta je použita pro určení počáteční polohy 3D boxu. Druhým parametrem je zorný úhel (viewpoint), který slouží ke zpřesnění odhadu polohy. Neprovádí se regrese konkrétního úhlu, ale klasifikace do celkem 16 druhů zorného úhlu viz. obrázek 2.6. Metoda pomocí těchto přidaných parametrů dosahuje lepší přesnosti 3D detekce a odhadu orientace. Dokáže zpracovat i snímky obsahující pouze částí vozidla.

2.4 Existující datové sady

Nově rozvíjející oblast autonomních vozidel způsobila potřebu pro datové sady obsahující 3D informaci. Ta bývá často udávána ve formě 3D bounding-boxu. Velké množství datových sad automobilů je proto cílené právě na odvětví autonomního řízení vozidel. Vozidla jsou snímána z výšky jedoucího vozidla, takže poskytují pohledy z nízkých úhlů, pro snímky pořízené z dopravních kamer se tedy nehodí. Mezi takové datové sady patří například Boxy [3], nyc3dcars [28] nebo KiTTi [14]. Datová sada BoxCars116k [40] poskytuje, oproti předchozím datovým sadám, snímky vozidel z vyšších úhlů dopravních kamer. Ukázky snímků z jednotlivých datových sad lze vidět na obrázku 2.7.

Boxy

Datová sada Boxy [3] obsahuje obrovské množství snímků ve vysokých rozlišení pořízených z dálnice. Sada obsahuje snímky různých stupňů hustoty provozu za různého počasí. Vozidla jsou ale snímány z jedoucího vozidla a to pouze ze zadní strany (ve směru jízdy dálnice).

nyc3dcars

Další datovou sadou je nyc3dcars [28]. Sada obsahuje snímky z ulic New Yorku. Oproti zde zmíněným datovým sadám jsou zde anotace vozidel uvedeny formou 6-DoF. Data obsahují snímky ulic města a křižovatek za hustého provozu. Fotografie jsou opět pořizovány z nízké výšky.

KiTTi

V neposlední řadě, zde musí být zmíněna datová sada KiTTi [14]. Jedná se o sadu určenou pro autonomní vozidla, data jsou pořízena ze střechy automobilu projíždějícího městem. Sada tak obsahuje různé úhly pohledů vozidel. Anotace vozidel jsou reprezentovány 3D bounding-boxem. Hlavním problémem je opět nízká výška snímaných vozidel.

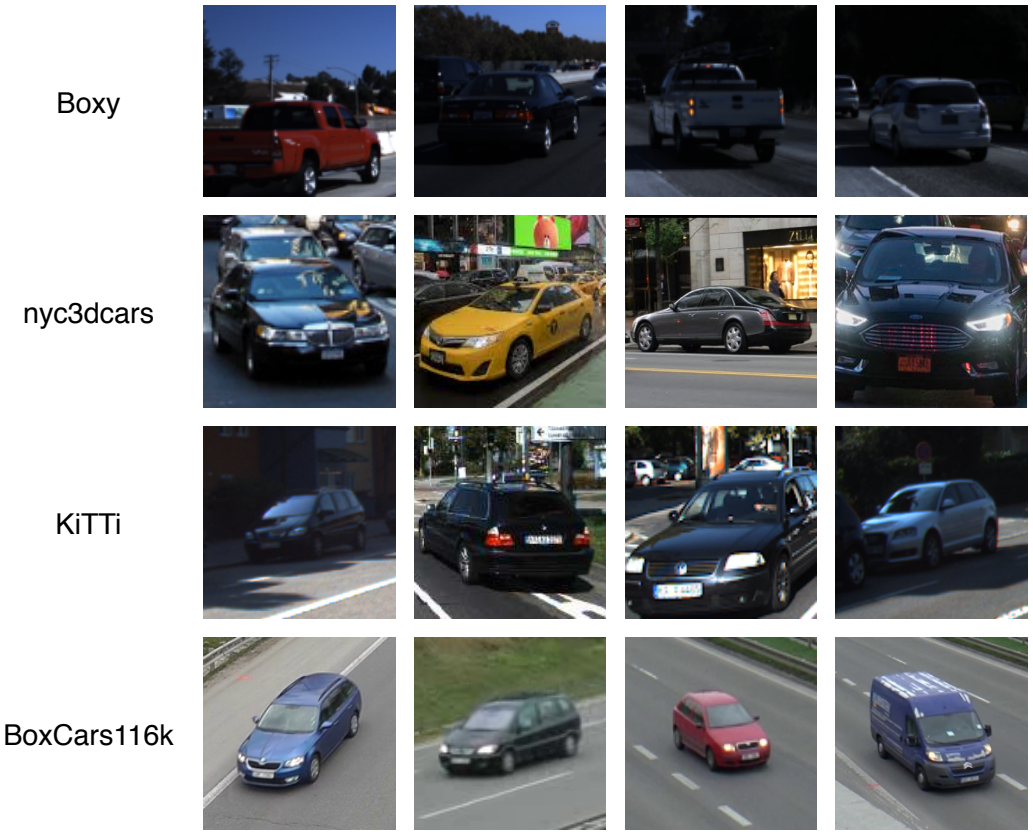
BoxCars116k

Datová sada BoxCars116k [40] obsahuje velké množství vozidel, které jsou nasnímány z různých uhlů. Výška umístění kamer, kterými byla pořízena data je dostatečně podobná výšce umístění dopravních kamer. Datová sada se skládá z předchozí verze BoxCars21k [39] doplněné o další data. Celkově sada obsahuje přes 116 tisíc snímků. Ačkoli je sada určená zejména pro klasifikaci vozidel, obsahuje pro každé vozidlo anotaci 3D bounding-boxu. Srovnání uvedených datových sad je vyjádřeno v tabulce 2.2.

3D bounding-boxy v datové sadě BoxCars116k byly vyrobeny pomocí metody automatické kalibrace dopravní kamery [11], která z několika minutového záznamu určí tři úběžné body definující směr jízdy vozidel pomocí Houghovy tranformace viz [10]. Samotný box je určen extrakcí siluety a určením tečen, které vedou z úběžných bodů k siluetě. Pomocí jejich průsečíků je konstruován 3D box, viz [11].

	Počet vozidel	Pohledy	Vyvýšení kamery
Boxy [3]	1 990 000	zadní, boční	žádné (pohled vozidla)
nyc3dcars [28]	3 787	všechny (boční méně)	mírné
KiTTi [14]	180 000	všechny	mírné (střecha vozidla)
BoxCars116k [40]	116 286	všechny	vysoké (dopravní kamery)

Tabulka 2.2: Porovnání jednotlivých datových sad podle velikosti sady, pohledů natočení vozidel a snímané výšky.



Obrázek 2.7: Ukázka snímků vozidel z datových sad Boxy [3], nyc3dcars [28], KiTTi [14] a BoxCars116k [40].

Kapitola 3

Vlastní datová sada

Existující datové sady obsahující vozidla s anotací polohy nejsou dostačující. Hlavní důvodem je malý rozsah pohledů a použitých kamer. Z tohoto důvodu byla sestavena datová sada, která je tvořena snímky z různých zdrojů. Snímky jsou pořízeny z různých kamer a zaznamenávají různé dopravní situace. Datová sada obsahuje asi 2 700 snímků. Z existujících datových sad je jako nejvhodnější dataset BoxCars116k, jeho část tvoří většinu ze sestavované sady. Sada je navíc doplněna o ručně ohodnocené snímky ze třech dalších zdrojů. Ručně zpracovány jsou snímky vozidel, které neobsahovaly informaci o poloze. Anotace je uváděna jako čtveřice bodů popisující podstavu automobilu na snímku.

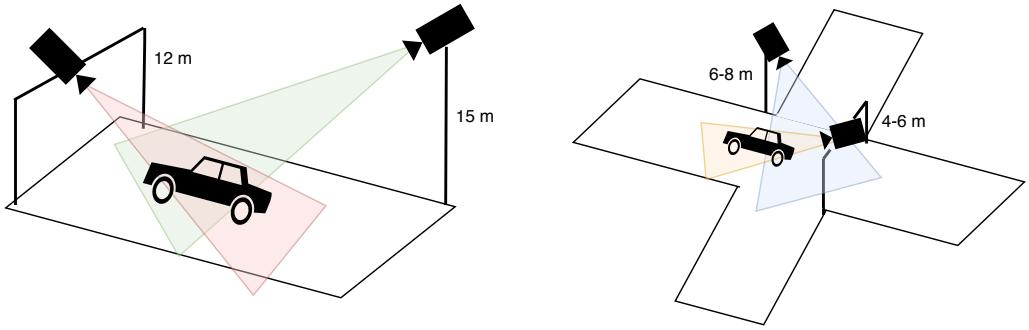
3.1 Požadavky na datovou sadu

Hlavním požadavkem na datovou sadu je výška, ze které byly pořízeny snímky vozidel. Dopravní kamery bývají umísťovány do výšky 4 – 15 metrů v závislosti, jestli je kamera umístěna nad silnicí, na okraji silnici nebo zda-li kamera sleduje dálnici, křižovatku [20]. Rozdíl mezi výškami umístění kamery je zobrazen na obrázku 3.1. Kamera, ze které jsou snímky pořízeny, musí snímat vozidla z vyšších úhlů.

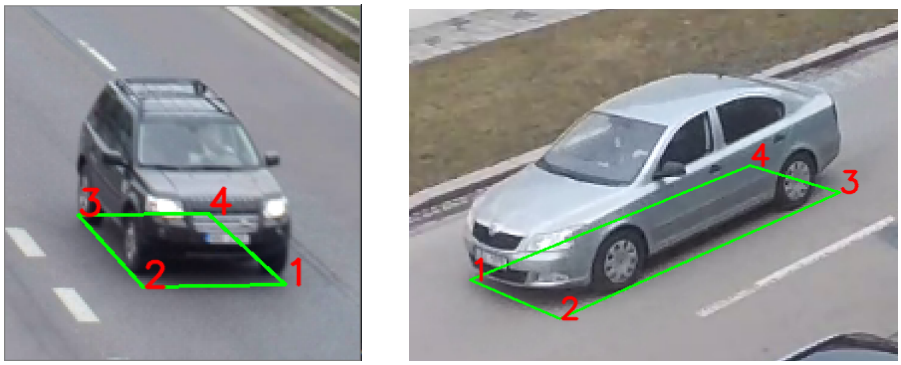
Dalším požadavkem je dostatečná generalizace. Data musí pocházet z více kamer z různých umístění. Cílem metody je totiž odhad pozice z jakékoli dopravní kamery, nezávisle na umístění. Lepší přesnosti metody by se poté dosáhlo přidáním dat z konkrétní kamery a dotrénování metody. Datová sada musí obsahovat informaci o pozici vozidla. Anotace je udávána ve formě 3D bounding-boxu nebo podstavy.

Formát dat

Obrázky obsahují výřez vozidla doplněný o jeho okolí. Okolí tvoří 20 % původní velikosti výřezu z detekce. Důvod pro zahrnutí okolí výřezu vozidla vychází z experimentu 5.2. Anotace jsou ukládány ve formě čtyř bodů reprezentující podstavu vozidla. Body se nacházejí v rovině snímku a jejich poloha je uváděna vzhledem ke středu obrázku. Pořadí bodů vychází ze způsobu anotace v datové sadě BoxCars116k, kde pořadí závisí na orientaci vozidla. Pořadí jednotlivých bodů je znázorněno na obrázku 3.2. Pokud je snímek pořízen z úhlu, kde je vidět levá strana vozidla, tak jsou body seřazeny v po směru hodinových ručiček začínající z bodu na pravé dolní straně. Při pohledu z pravé strany vozidla je anotace seřazena proti směru hodin, kde je jako první bod uveden levý dolní.



Obrázek 3.1: Rozdíly ve výšce umístění dopravních kamer na dálnici (nalevo) a křižovatce (vpravo), konkrétní výšky jsou uvedeny jako doporučené [20].



Obrázek 3.2: Pořadí bodů anotace v datové sadě BoxCars116k [40], které je použito napříč celou vlastní datovou sadou.

3.2 Obsah datové sady

Datová sada se skládá z obrázků vozidel ze čtyř různých zdrojů. Jak již bylo zmíněno většinu tvoří část datové sady BoxCars116k, přítomna jsou potom data z vlastních záznamů, data z AI City Challenge 2018 [30] a data stažená z Google Images. Velikosti jednotlivých částí jsou reprezentovány tabulkou 3.1.

Snímky z BoxCars116k tvoří většinu datové sady. Byla použita pouze malá část této sady, aby poměr dat z různých zdrojů nebyl příliš nevyvážený, a metoda tak byla schopna generalizace. Z důvodu nepřesného odhadu podstavy u některých snímků, byly data ručně rozřazeny na přesné a nepřesné. U snímků s přesnější anotací byla použita anotace z původní datové sady, tedy podstava z 3D bounding-boxu. Ostatní vozidla byla ručně ohodnocena a také zařazena do datové sady.

V průběhu přípravy práce byly pořízeny dva videozáznamy jedné křižovatky, každý z jiného úhlu. Tyto záznamy byly použity pro experiment, viz 5.1, ve kterém uživatelé prováděli ruční anotaci vozidel, tedy zadávali body podstavy do snímku. Do datové sady byly zařazeny i tyto anotované snímky.

Podobným způsobem byly zpracovány i data z datové sady použité v AI City Challenge. Sada obsahuje kamerové záznamy křižovatek z USA. Z této datové sady byla vybrána data

Zdroj	Počet snímků	Část sady (%)
BoxCars116k [40]	2400	85.9
Vlastní záznamy	153	5.5
AI City challenge [30]	191	6.8
Google Images	50	1.8

Tabulka 3.1: Poměr jednotlivých zdrojů v datové sadě.

Obrázek 3.3: Reprezentace ruční anotace ve formě zadání čtveřice bodů.

z celkově osmi kamer, které monitorují jednu křižovatku. Z každého záznamu kamery byla vybrána část snímků vozidel k ruční anotaci.

Na závěr se datová sada doplnila o malé množství snímků nalezených pomocí Google Images. Tyto obrázky vozidel pocházejí z různých zdrojů. Obrázky vozidel byly zpracovány, ručně anotovány a přidány do datové sady.

Ruční anotace automobilů

Problém anotace v oblasti odhadu 3D pozice vozidel byl již zmíněn. Mezi nejčastější řešení patří použití syntetických dat nebo použití ruční anotace. Při sestavování datové sady byla použita právě ruční anotace. Uživatel pomocí anotačního programu určil na snímku čtyři body identifikující podstavu vozidla, viz obrázek 3.3. Uživatel byl obeznámen o správném pořadí bodů na základě orientace vozidla. I přesto bylo pořadí kontrolováno a opraveno anotačním programem. Body byly ukládány jako souřadnice v souřadném systému obrázku, které byly posunuty vůči středu. Přesnost ruční anotace byla ověřena pomocí experimentu 5.1, kde se porovnávala anotace vozidla z dvou různých úhlů.

Kapitola 4

Navrhovaná metoda pro odhad pozice vozidla

Odhad pozice z dopravní kamery musí být prováděn rychle, aby ho bylo možné provést v reálném čase, proto jsem se zaměřil na to, aby metoda bylo co nejjednodušší a dostatečně rychlá. Metoda, kterou se zabývám v této práci vychází z odhadu pozice pomocí regrese. Pomocí regrese není zjištěna přímo 3D pozice vozidla, ale pozice podstavy na snímku. Tento proces je znázorněn na obrázku 4.1. Reálná pozice je určena z bodů podstavy.

Nejprve je potřeba provést detekci vozidel na snímku z dopravní kamery. Výstupem detektoru jsou boxy ohraničující vozidla a každé vozidlo je poté zpracováno zvlášť. K výřezu z detektoru se přidá okolí vozidla. Výsledný výřez vozidla s jeho okolím je předán konvoluční neuronové síti.

Konvoluční síť provede regresi čtveřice bodů reprezentujících podstavu automobilu. Neprovádí se tedy regrese 3D parametrů (orientace, velikost), jako je tomu například ve článku [29]. Výstupem neuronové sítě je přímo podstava vozidla. Podstava je udána v rovině snímku, a proto je nutné ji převést do souřadnic země.

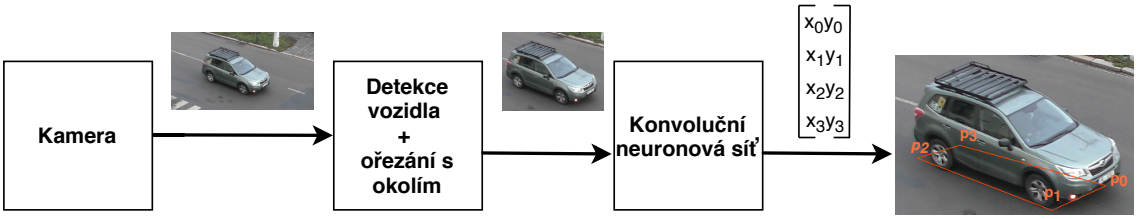
Projekce souřadnic na snímku do roviny silnici je realizována pomocí matice homografie. Vynásobením vektoru souřadnic touto maticí určí pozici bodů v souřadném systému země. K tomu jsou zapotřebí dodatečné informace o scéně. Výstupem metody je pozice vozidel v reálných souřadnicích v metrické soustavě.

4.1 Detekce automobilů ve snímku

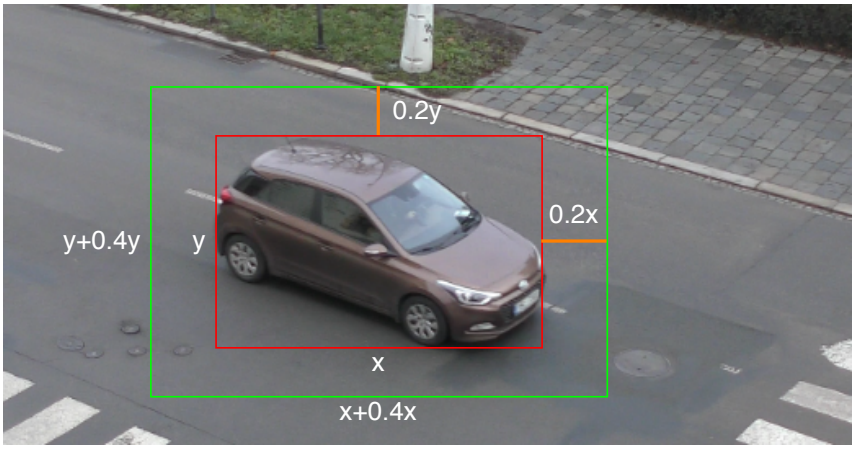
Ze snímku z dopravní kamery je nejdříve potřeba oddělit jednotlivá vozidla pomocí detektoru objektů. Pro detekci objektů je použita metoda Faster R-CNN. Konkrétně je použitá Faster R-CNN s ResNet-50 [17] používající FPN. Kombinace Faster R-CNN a FPN již byla zmíněna v kapitole 2.1. ResNet je reziduální neuronová síť umožnující přeskakovat jednotlivé vrstvy, to zrychluje učení a zlepšuje přesnost hlubších neuronových sítí.

Implementace detektoru vozidel

Implementace detektoru objektů používá předtrénovaný model této metody z knihovny PyTorch [31]. Rychlost zpracování jednoho snímků, uvedená v dokumentaci je $0,059$ sekund. Model je předtrénovaný na datové sadě COCO train2017[25]. Datová sada obsahuje přes 118 tisíc snímků celkově 91 tříd objektů. V této práci jsou použity pouze kategorie reprezentující automobil a nákladní auto. Detekční práh (threshold) je vhodně určen na hodnotu $0,5$.



Obrázek 4.1: Diagram znázorňující proces zpracování snímku. Na snímku se detekuje vozidlo, z výřezu poté konvoluční neuronová síť určí čtveřici bodů.



Obrázek 4.2: Způsob ořezání oblasti vozidla s přidaným okolím (20 %).

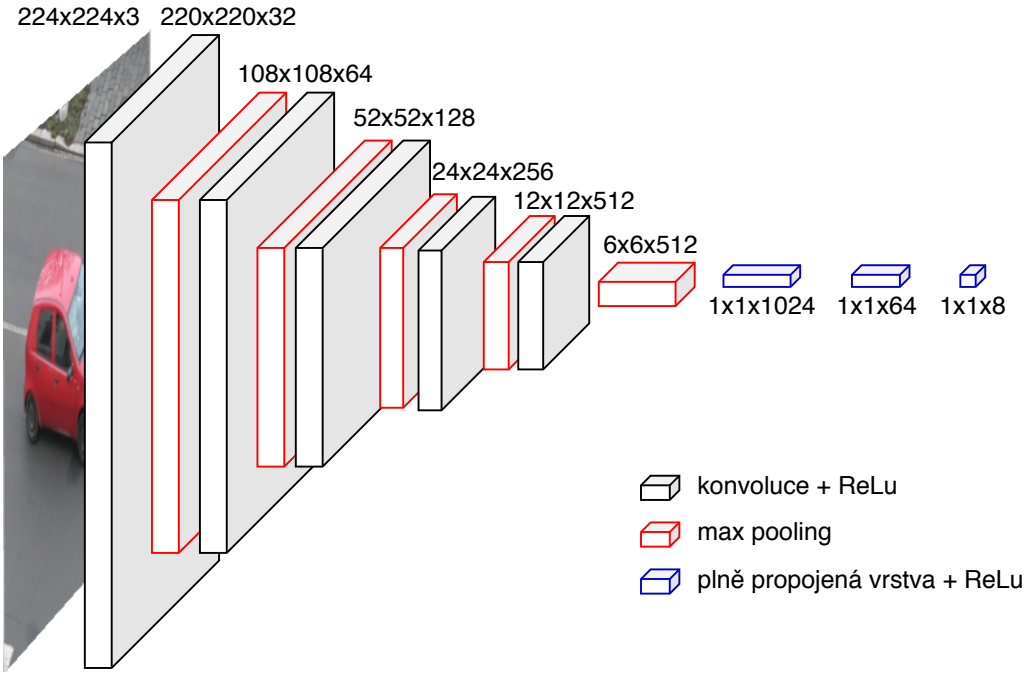
Detektor s touto hodnotou prahu vykazoval při vyhodnocení dobré výsledky. Tato hodnota se také používá ve PASCAL VOC[1]. Výstupem je 2D bounding-box reprezentace polohy vozidla na snímku.

Ořezání

Výstupem detektoru objektu jsou bounding-boxy a třídy objektů. Aby bylo možné oblast s vozidlem zpracovat dále, je nutné provést ořezání. Pro přesnější odhad pozice vozidla je zapotřebí i informace o okolí vozidla. Toto rozhodnutí bylo provedeno na základě experimentu 5.2. Velikost okolí byla nastavena na 20 % velikosti oblasti vozidla, viz obrázek 4.2.

Další možností by bylo nastavit velikost okolí fixně, jak tomu bylo například v datové sadě BoxCars, kde je k vozidlu přidaná hodnota okolí 30 pixelů z každé strany detekovaného bounding-boxu. Výsledný výřez je připraven pro zpracování neuronovou sítí pro regresi podstavy.

[1] http://host.robots.ox.ac.uk/pascal/VOC/



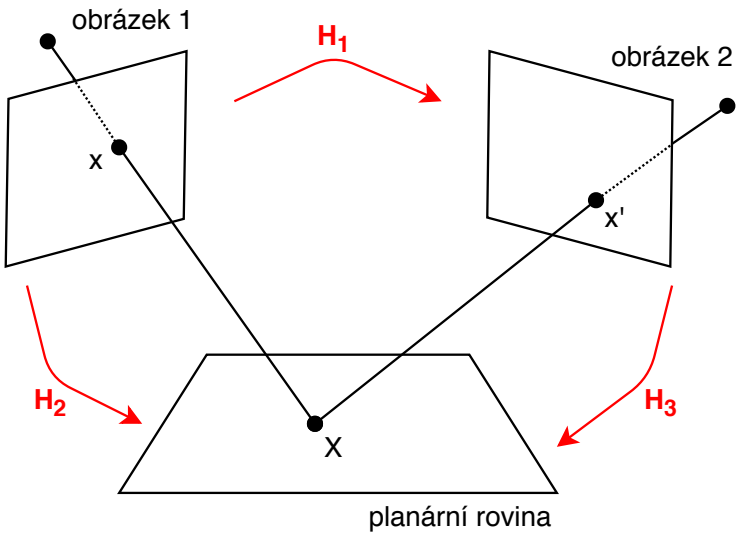
Obrázek 4.3: Model konvoluční neuronové sítě použité pro regresi bodů podstavy.

4.2 Odhad podstavy vozidla pomocí regrese

Pro odhad podstavy je použita jednoduchá konvoluční síť. Vstupem je obrázek o velikosti 224×224 pixelů. Obrázek je síti předán ve formě tenzoru, hodnoty pixelů jsou normalizovány do rozsahu $[0, 1]$. Postupným zpracováním jednotlivými konvolučnimi vrstvami, se obrázek převede na vektoru příznaků. Z tohoto vektoru je poté provedena regrese na konečný vektor o osmi souřadnicích reprezentující čtveřici bodů podstavy.

Model neuronové sítě pro regresi podstavy je uveden na obrázku 4.3. Obsahuje pět úrovní konvolučních vrstev. Úkolem těchto vrstev je získat lokální informace z rastrového obrázku. Pomocí konvolučních filtrů se provádí konvoluce nad celým obrázkem, vzniká tak velké množství rovin příznaků [4]. Pro omezení velikosti výstupů se používá tzv. max-pooling vrstva, která snižuje rozlišení výstupu z předchozí konvoluční vrstvy [37]. Konkrétně je zde použita pooling vrstva s podoblastmi 2×2. Vrstva vybere z každé podoblasti pixel s největší hodnotou, tím se rozlišení zmenší na polovinu. Pooling vrstva je použita za každou úrovní konvoluční vrstvy.

Výstupem z konvolučních vrstev je velké množství příznaků o malém rozlišení, tyto příznaky jsou transformovány do vektoru a poslány ke zpracování plně propojeným vrstvám. Výsledkem plně propojené vrstvy je regrese na vektor menší velikosti. Tři úrovně vrstev zajišťují regresi na konečný vektor o osmi hodnotách, ze kterého lze poté určit čtveřice bodů v souřadném systému snímku, která reprezentuje podstavu vozidla.



Obrázek 4.4: Homografie mezi dvěma obrázky. Pomocí matice H_1 lze provést projekci z jedné roviny obrázku do roviny obrázku druhého. Lze také provést projekci přímo do planární roviny, pomocí matice H_2 nebo H_3.

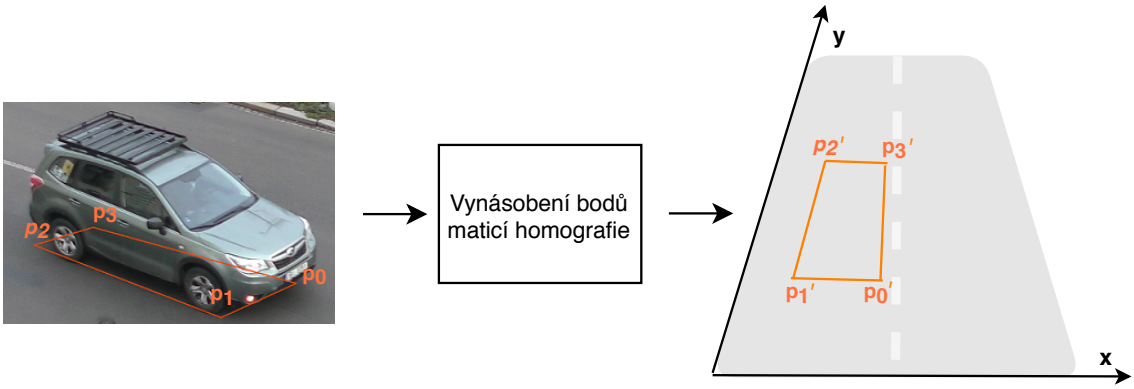
4.3 Projekce bodů do roviny země

Podstava vozidla v souřadném systému snímku nevyjadřuje 3D pozici vozidla, proto je potřeba body podstavy promítnout do roviny silnice. Tento proces vyžaduje dodatečné informace o snímané scéně. Pokud uživatel poskytne tyto informace, metoda je schopna provést projekci. Projekce je prováděna pomocí homografie.

Homografie

Homografie [9] je v geometrii pojem reprezentující izomorfismus mezi projektivními prostory. Homografii popisuje rovinice 4.1. Jedná se o způsob promítání bodů z jednoho prostoru do druhého pomocí matice homografie H. Matice homografie má velikost 3×3, hodnota h_{33} bývá normalizovaná na 1. Pro projekci je tedy zapotřebí určit osm parametrů. Matice je určena pomocí sady bodů ze zdrojového prostoru a sady korespondujících bodů z cílového prostoru. Jakmile známe matici H, můžeme provádět libovolně projekce z jednoho prostoru do druhého, nebo naopak pomocí matice inverzní. Promítání mezi dvěma snímky a snímanou rovinou je znázorněno na obrázku 4.4. Je nutné zmínit, že projekce pomocí homografie funguje jen mezi rovinami, nelze jej tedy uplatit na 3D prostor. Obvykle je homografie v počítačovém vidění použita v oblasti rozšířené reality, tvorbě panoramat nebo pro změnu perspektivy.

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = H \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{4.1}$$



Obrázek 4.5: Zobrazení procesu projekce podstavy automobilu do souřadného systému silnice určeného uživatelem.

Využití homografie k projekci podstavy

Pro využití homografie k projekci bodů do roviny země budou taky zapotřebí dvě sady korespondujících bodů, v rovině snímku a v rovině silnice. Tato metoda počítá s tím, že je snímaná silnice rovná, protože, jak již bylo zmíněno, projekce probíhá mezi dvěma rovinami, nelze tedy brát v úvahu sklon silnice ani její deformaci. Proces získání sad bodů pro určení matice je pro účel této práce nazýván kalibrace. Je nutné, aby si uživatel určil souřadný systém snímané dopravní situace. Tento systém může jako jednotky používat jednotky metrické soustavy. Střed souřadného systému je libovolný, je ale důležité, aby zvolené body na silnici byly dobře viditelné i z dopravní kamery, která zvolenou silnici snímá. Pro zvolené body na silnici poté stačí určit jejich souřadnice na snímku. Tyto dvě sady bodů jsou poté použity k vypočítání matice homografie, v této práci je použita funkce *findHomography* z knihovny OpenCV[2].

Jakmile je k dispozici matice homografie mezi obrazem z kamery a snímanou plochou, je možné libovolné body převádět přímo do roviny země. Body podstavy získané neuronovou sítí se vynásobí maticí homografie a vznikne projekce podstavy do roviny silnice. Tento proces je znázorněn na obrázku 4.5. Při správné volbě souřadného systému, aby jeho souřadné osy byly kolmé (kartézský systém), lze poté na zobrazení nahlížet jako na ptačí perspektivu.

[2] https://docs.opencv.org/2.4/modules/calib3d/doc/camera_calibration_and_3d_reconstruction.html

Kapitola 5

Experimenty

Aby mohla být metoda vyhodnocena, musí být provedeny experimenty. Tato kapitola obsahuje experiment, který zkoumá přesnost ruční anotace, ve kterém se nad dvěma pořízenými záznamy stejné dopravní situace provedla ruční anotace projíždějících vozidel. Výsledky anotace byly promítnuty do roviny silnice pomocí homografie a zde porovnány. Pro účel homografie je zde také popsána kalibrace scény. Kalibrace je, jak bylo popsáno v podkapitole 4.3, potřebná pro projekci pomocí matice homografie.

Další experimenty jsou prováděny přímo nad konvoluční neuronovou sítí, je zde shrnuto trénování neuronové sítě včetně volby trénovacích parametrů a předzpracování snímků, vyhodnocení sítě na datové sadě probíráné v kapitole 3. Výsledky jsou porovnány s výsledky modelu neuronové sítě VGG-11 [38] a také s vyhodnocením pozice pomocí detektoru klíčových bodů.

Poslední experiment obsahuje porovnání ručně anotovaných podstav s podstavami odhadnutými pomocí konvoluční neuronové sítě. Použita jsou vozidla z pořízených záznamů z validační části datové sady. Výsledky jsou porovnány v rovině silnice.

5.1 Experiment přesnosti ruční anotace

Ruční anotace již byla zmíněna v kapitole 3, kde byla probrána spolu se sestavením datové sady. Datová sada obsahuje anotace ve formě podstavy vozidla v souřadnicích snímku, jelikož část datové sady neobsahuje tyto anotace, je potřeba je zadat ručně.

Tento experiment se zaobírá přesností ručně zadané podstavy vozidla. Byly pořízeny dva záznamy jedné křižovatky, každý z jiného pohledu. Uživatel zadá podstavy vozidla pro oba tyto pohledy. Zadané pozice automobilů jsou promítnuté do souřadného systému země, pomocí homografie a zde porovnány. Tento proces je proveden pro každé vozidlo ze záznamu několikrát, pokaždé jiným uživatelem. Kromě záznamů bylo potřeba určit body na reálné křižovatce a jejich reprezentaci v rovině snímku z obou kamer.

Obrázek 5.1: Umístění kamer a společně snímaná plocha křižovatky pro účely experimentu. Křižovatku snímaly dvě kamery, na mostě mezi budovami (vlevo) a kamera ve třetím patře fakultní budovy L (vpravo).

Pořízení záznamů a informací o scéně

Pro účel experimentu byly pořízeny dva záznamy křižovatky nacházející se u Fakulty informačních technologií VUT na ulici Božetěchova. Umístění kamer a jejich společná snímaná plocha je znázorněna na obrázku 5.1. Kamery byly záměrně umístěny tak, aby byly na vyvýšeném místě splňující kritérium umístění dopravních kamer, a tak, aby sdílely co největší část snímaného prostoru. Záznamy byly pořízeny na zapůjčené kamery Panasonic HC-VX980. Záznamy byly časově synchronizovány. Ze synchronizovaných záznamů bylo ručně vybráno 71 snímků obsahující různé automobily projíždějící touto křižovatkou.

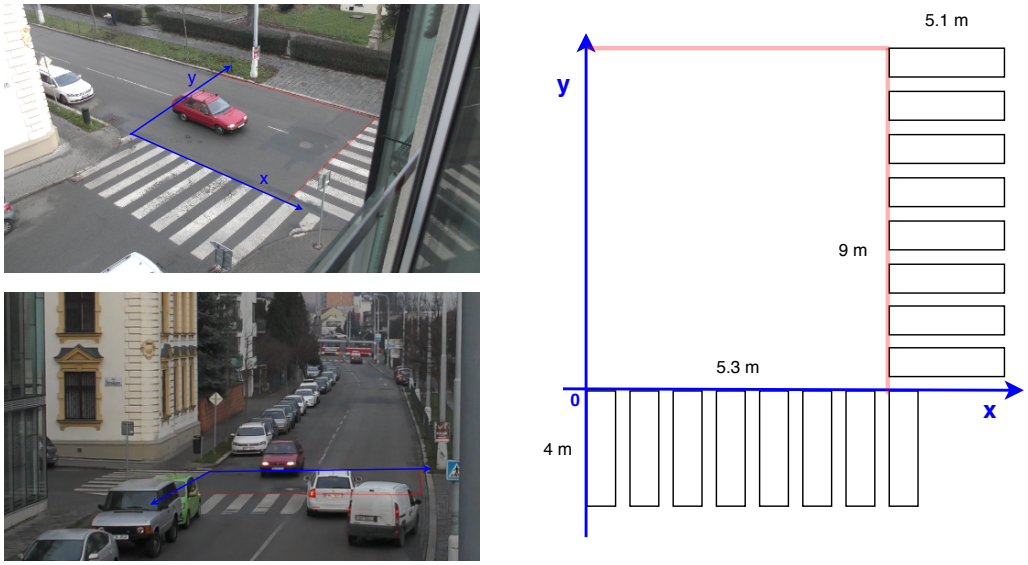
Projekce anotované podstavy do roviny křižovatky

K provedení projekce pomocí homografie je zapotřebí kalibrace scény, tedy správné zvolení souřadného systému a sady bodů. Volbu souřadného systému ukazuje obrázek 5.2. Jako základ byly použity dva přechody pro chodce, u každého přechodu byla změřena délka a šířka. Tyto rozměry byly použity pro určení čtveřice bodů v souřadném systému. Následně byla ručně určena pozice těchto bodů na obou kamerách. Tyto tři sady bodů byly poté použity k odhadu dvojce matic homografie, kde každá sloužila pro projekci bodů z jedné kamery do roviny určeného souřadného systému křižovatky.

Vyhodnocení experimentu

Celkem 71 vozidel bylo ohodnoceno skupinou pěti uživatelů. Uživatel zadal pro vozidlo dvě podstavy, každou z jiného snímaného úhlu. Pro zadávání využívali uživatelé anotační nástroj, na pořadí zadaných bodů nezáleželo. Podstavy se pomocí příslušné matice homografie promítly do roviny silnice.

V rovině země se poté porovnávaly obě podstavy podle tří parametrů. Prvním parametrem pro porovnání je průnik přes sjednocení, neboli IoU, tedy do jaké míry se podstavy



Obrázek 5.2: Využití přechodů pro chodce jako základ pro souřadný systém.

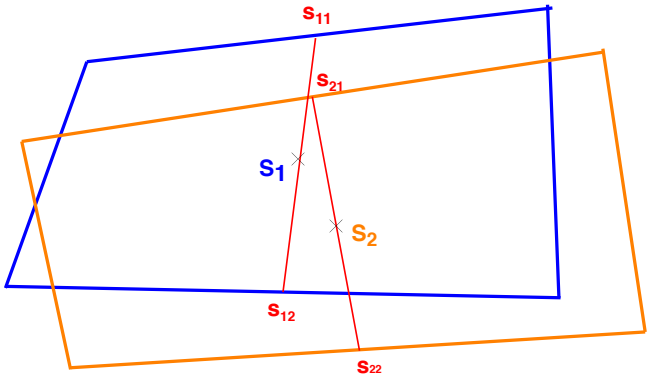
Uživatel	IoU (%)	Vzdálenost (m)	Úhel (deg)
1	59.10	0.66	5.28
2	65.92	0.51	5.96
3	62.97	0.59	10.40
4	54.36	0.81	12.90
5	70.48	0.53	8.70
průměr	62.57	0.62	8.65

Tabulka 5.1: Výsledky experimentu představující průměrné výsledky pro každého uživatele.

překrývají. Další dva parametry znázorňuje obrázek 5.3. Jedním z nich je vzdálenost středů podstav, udávána v metrech. Posledním parametrem je rozdíl úhlů. Jelikož zadané podstavy nejsou pravidelnými obdélníky, probíhá porovnání úhlů nad úhly, které reprezentují natočení úsečky vedené ze středů delších stran podstav.

Výsledky experimentu

Výsledky jsou uvedeny v tabulce 5.1. Pro každého uživatele je uváděny průměrné výsledky nad sadou 71 vozidel. Z těchto výsledků můžeme vyvodit, že ruční anotace je natolik přesná, aby se dala použít pro anotaci vozidel při tvorbě datové sady.



Obrázek 5.3: Reprezentace parametrů experimentu. Vzdálenost středů podstav $|S_1S_2|$ a rozdíl natočení přímek $s_{11}s_{12}$ a $s_{21}s_{22}$. Tyto přímky jsou vedeny ze středů delších stran podstavy, tedy ze středů bočních stran vozidla.

5.2 Vyhodnocení regrese podstavy vozidla

Druhý experiment se zaměřuje na konvoluční neuronovou síť pro regresi podstavu vozidla. Cílem tohoto experimentu je shrnout proces trénovaní neuronové sítě a zaměřit se na její vyhodnocení na datové sadě.

Datová sada, popsána v kapitole 3, je složena ze snímků ze čtyř různých zdrojů. Tato sada je rozdělena na trénovací a validační část. Trénovací část datové sady je tvořena z 80 % snímků z každého zdroje, validační ze zbylých 20 %. To zajistí aby v každé části bylo rovnoměrné zastoupení snímků ze všech čtyř zdrojů. Na vstup neuronové sítě přicházejí výřezy vozidel z detektoru objektů, tyto výřezy jsou transformovány na fixní velikost 224×224. Data určená pro trénování mají navíc 50% pravděpodobnost na horizontální otočení. Výřezy je nutné převést na tenzory, aby s nimi mohla neuronová síť pracovat.

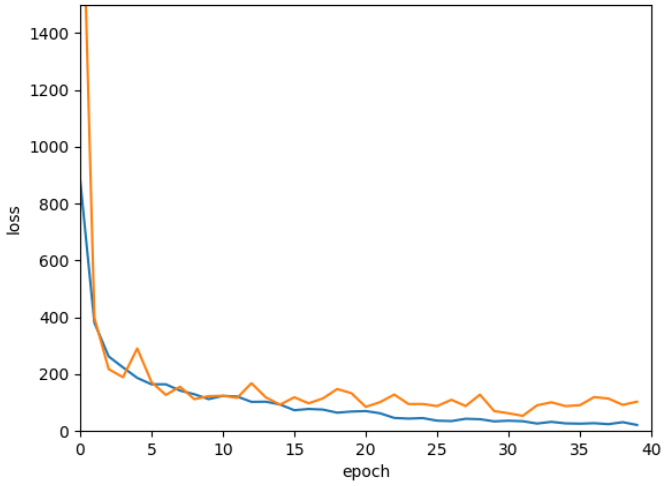
Trénovací parametry

Jako funkce pro vyhodnocení chyby sítě (loss function) je vybrána metoda nejmenších čtverců MSE. Funkce je často používaná funkce v regresních problémech, viz [5]. Vzorec funkce je popsán rovnicí 5.1. Počítá se jako průměr kvadratických rozdílů mezi predikovanou a skutečnou hodnotou, chyba je tedy vždy kladná, nezávisle na znaménku hodnot.

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(Y_i - \hat{Y}_i\right)^2 \tag{5.1}$$

Při učení neuronové sítě je cílem minimalizovat loss funkci, proto se používá tzv. optimizer. Optimizer je algoritmus, který aktualizuje parametry sítě v závislosti na hodnotách chybové funkce [1]. Byl použit algoritmus Adam [19]. Adam je rozšíření stochastického sestupu gradientu, na rozdíl od něho však nemá fixní míru učení. Počáteční míra učení, neboli velikost kroku algoritmu, byla nastavena na hodnotu $0,001$.

Velikost dávky (batch size), neboli počet snímků, které síť zpracuje před aktualizací parametrů byla zvolena na velikost čtyři. Větší velikost nebyla možná z důvodu nedostatečné paměti na grafické kartě zařízení.



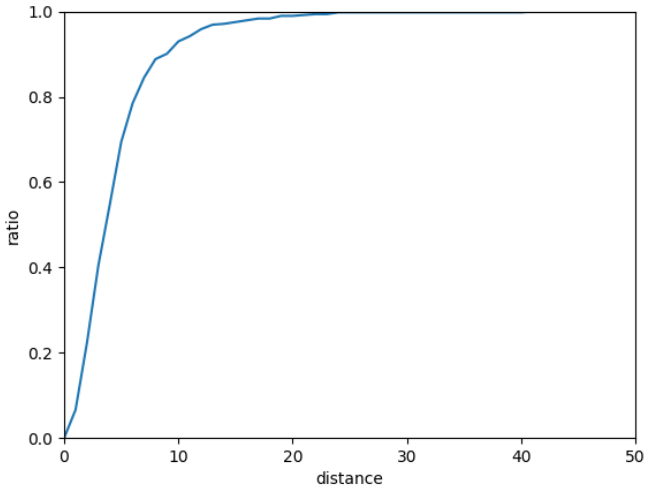
Obrázek 5.4: Graf reprezentující trénování neuronové sítě, zobrazena je trénovací chyba (modrá) a validační chyba (oranžová).

Obrázek 5.5: Ukázka výstupu konvoluční neuronové sítě pro regresi podstavy automobilu.

Výsledky regrese podstavy vozidla

Neuronová síť byla trénovaná na grafické kartě NVIDIA GeForce GTX 1660ti po dobu 35 minut. Celkový proces trénování je znázorněn na grafu 5.4. Jak jde vidět na grafu nejlepších výsledků dosáhla síť kolem 30. epochy. Konkrétně neuronová síť dosáhla hodnoty chybové funkce $18,833$ nad validační části datové sady.

Úspěsnost sítě jde vidět i na grafu 5.6. Graf zobrazuje poměr počtu bodů predikovaných pod určitou vzdálenost od jejich vzorů. Z grafu vyplývá, že téměř většina z predikovaných bodů se nacházela pod vzdálenost 20 pixelů. Ukázku výstupu sítě zobrazuje obrázek 5.5, kde je zobrazena jedná dávka (batch) vozidel.



Obrázek 5.6: Graf popisující úspěšnost regrese na základě poměru počtu bodů detekovaných pod hranici vzdálenosti od jejich vzoru.

5.2.1 Porovnání výsledků experimentu

Výsledky shrnuje tabulka 5.2. Obsahuje trénovací a validační chybu jednoduché neuronové sítě, která se používá v této metodě. Obsahuje různé modifikace sítě a jejich dopad na výslednou chybu. Dále obsahuje výsledky modelu VGG-11 a výsledky regrese podstavy podle klíčových bodů dostupných z detektoru klíčových bodů vozidel.

Jednoduchá konvoluční neuronová síť

Část tabulky popisující výsledky jednoduché konvoluční neuronové sítě zdůvodňuje rozhodnutí, které byly vykonány při trénování této sítě. Při trénování sítě pouze nad datovou sadou BoxCars116k a následné vyhodnocení nad validační částí datové sady, tak je zřejmé, že síť má problémy s generalizací a hůř zvládá případy vozidel pořízených z ostatních zdrojů.

Dále je zde vidět důvod pro přidání okolí vozidel k jejich výřezům z detektoru objektů. Dodatečná informace o okolí značně vylepšila výsledky regrese podstavy.

V poslední řadě je zde znázorněn vliv náhodného horizontálního otočení, každý snímek vozidla má 50% šanci na transformaci. To způsobuje lepší různorodost trénovacích dat při každé epoše a tedy menší přetrénování sítě.

Model	Popis	Chyba (train)	Chyba (val)
Jednoduchá CNN	Horizontální otočení	17.13	18.83
	Bez augmentace	20.37	34.75
	Bez okolí (pouze 2D boxy)	45.23	60.87
	Pouze BoxCars [40], h. otočení	37.26	86.54
	Pouze BoxCars	26.96	91.81
VGG-11 [38]	Nepředtrénované lin. vrstvy	15.21	50.11
	Pouze poslední lin. vrstva	20.46	77.10
Keypoint detektor [41]	Podstava ze 3D modelu	x	890.48

Tabulka 5.2: Tabulka reprezentující výsledky experimentu. Obsahuje údaje o trénovacích a validačních chybách jednoduché konvoluční neuronové sítě, která je popsaná v kapitole 4.2 a je používaná pro regresi podstavy. Dále jsou uvedeny pro porovnání výsledky z předtrénovaného modelu VGG-11 a výsledky z dostupného detektoru klíčových bodů.

Model	Doba zpracování výřezu (ms)	Velikost modelu (MB)
Jednoduchá CNN	7.979	74.2
VGG-11 [38]	16.954	491.0

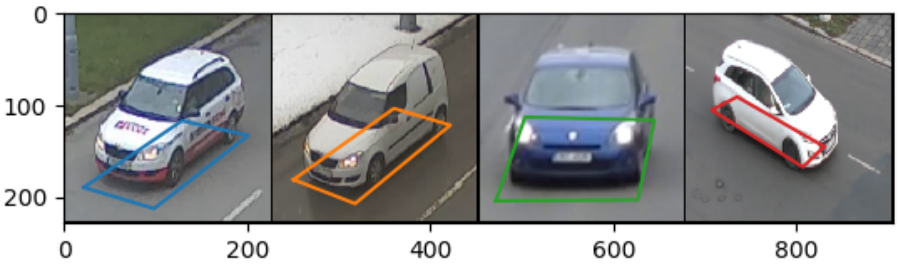
Tabulka 5.3: Porovnání navržené neuronové sítě s modelem VGG-11 podle doby zpracování jednoho vozidla (výřezu) a podle velikosti modelu na disku.

VGG-11 pro regresi podstavy vozidla

Pro porovnání se použila konvoluční neuronová síť VGG-11 [38]. Architektura VGG-11 je velice podobná architektuře zvolené neuronové sítě v této práci, hlavním rozdílem je, že VGG obsahuje na každé úrovni dvě konvoluční vrstvy před max-poolingem namísto pouze jedné. Použita byla již předtrénovaná varianta modelu z knihovny PyTorch natrénovaná na datové sadě ImageNet [8]. Model byl natrénovaný pro klasifikaci, pro aplikaci na regresi podstavy vozidla bylo nutné znovu natrénovat lineární vrstvy tohoto modelu.

Nejprve se vyzkoušela varianta, kde se doplnila pouze poslední lineární vrstva s regresí na vektor osmi čísel, tedy čtyř bodů podstavy vozidla. Tento způsob nebyl tak efektivní, jelikož zbylé lineární vrstvy byly natrénované na obrovské datové sadě, tudíž se v ní nedostatečně projevila trénovací část mé sestavené datové sady. Z toho důvodu jsem model dotrénoval bez použití předtrénovaných parametrů lineárních plně propojených vrstev. To zapříčinilo lepší výsledky než u předchozího způsobu trénování VGG, ale lepších výsledků stále dosahovala navržená konvoluční neuronová síť. Ukázka výstupu regrese podstavy z VGG lze vidět na obrázku 5.7.

Velikost modelu VGG-11 na disku dosahuje 491 MB, to je přibližně šestkrát více než velikost modelu navržené sítě. Srovnání velikosti lze vidět v tabulce 5.3. Tabulka zobrazuje i srovnání rychlosti obou modelů, zpracování jednoho výřezu vozidle je až dvakrát pomalejší u VGG. Nasazení modelu je tedy obtížnější a dotrénování pro specifické použití je tedy pomalejší.



Obrázek 5.7: Ukázka výstupu VGG-11 pro regresi podstav automobilů.

Použití detektoru klíčových bodů k odhadu podstavy vozidla

Poslední metodou pro porovnání byl detektor klíčových bodů. Byl mi poskytnut již natrénovaný model detektoru klíčových bodů vozidel navržený v práci [41]. Detektor byl natrénovaný na detekci celkem 20 klíčových bodů s tím, že detekoval pouze viditelné části vozidla.

K vyhodnocení této metody byl použit vzorový model 3D automobilu ve formě seznamu klíčových bodů a jejich 3D souřadnic, dostupný z této práce [21]. Ten byl poté upraven, aby odpovídal detekovaným klíčovým bodům. Z detekovaných klíčových bodů a jejich 3D reprezentace se poté určila póza a to pomocí funkce *solvePnP* z knihovny OpenCV, ale pouze za předpokladu, že metoda detekovala alespoň čtyři klíčové body. Výsledkem byl vektor translace a rotace použití při projekci bodů z 3D na snímek. Pomocí projekce lze tedy určili zbylé nedetekované klíčové body a také podstavu automobilu.

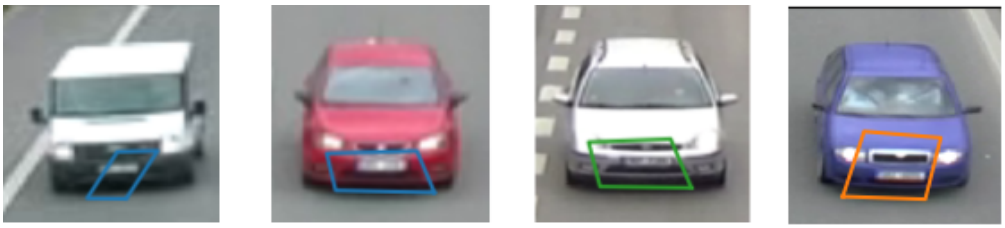
V tabulce 5.2 lze vidět , že metoda dosáhla velmi špatného výsledku v porovnáním s ostatními. To je především špatnou detekcí klíčových bodů, v některých případech detektor prohodil přední a zadní stranu automobilu a to znamenalo velice nepřesnou projekci podstavy, často se také stávalo, že množství detekovaných bodů bylo příliš malé pro správné určení vektorů transformace. Špatná detekce mohla být zapříčiněna nedotrénováním metody na trénovací části datové sady, k tomu ale nebyly dostupné anotace klíčových bodů.

5.2.2 Problémové případy

Konvoluční neuronová síť je ve většině případů schopna přesné regrese podstavy vozidla do takové míry, aby se dala tato podstava použít pro určení 3D pozice pomocí homografie. Během experimentu byly odhaleny problémové případy, u kterých je podstava vozidla zřetelně špatně odhadnuta.

Nepřesná regrese se projevuje u případů, kde je na výřezu pouze část vozidla. Příklady takových snímků lze vidět na obrázku 5.9. Tato situace nastává, když je vozidlo detekováno na okraji snímané plochy kamery, ale může nastat také při chybné detekci, kde detektor neoznačí celou část vozidla. Tento problém by mohl být vyřešen přidáním více snímků obsahujících pouze části vozidel nebo augmentací pomocí ořezání již stávajících vozidel z datové sady.

Obrázek 5.8: Ukázka chybné regrese při zpracování částí vozidel.



Obrázek 5.9: Ukázka chybné regrese při zpracování vozidel nasnímaných z přední strany.

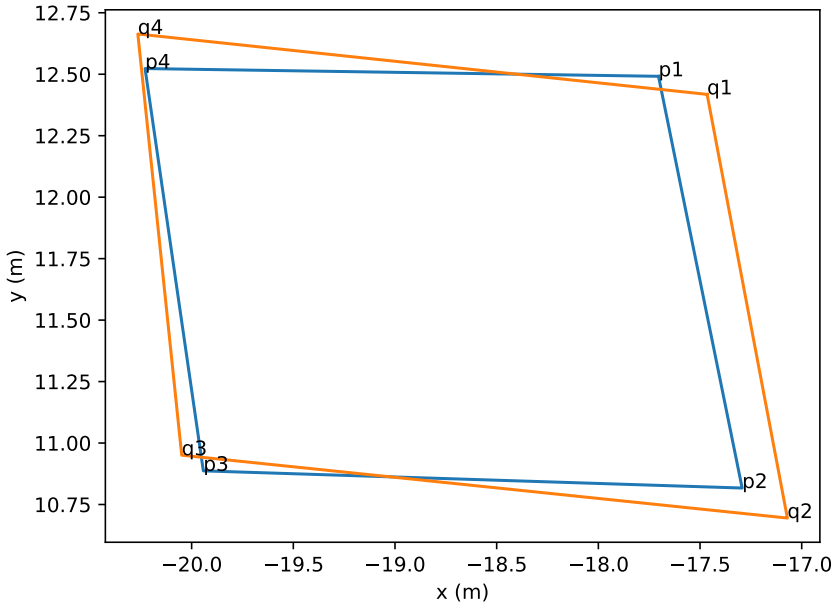
Další případ nepřesné regrese nastává, když je pohled na vozidlo příliš přímý, tedy když je na vozidlo nahlíženo zepředu, zezadu nebo přímo z boční části. Příklady snímků předních stran automobilů lze vidět na obrázku 5.8. Tato je způsobená obtížnou ruční anotací těchto případů. Při nasnímání vozidla z přímého úhlu se špatně odhadují body podstavy na opačné straně vozidla. Problémy také může působit nekonzistence formátů ukládání bodů podstavy. Pořadí uložení bodů záleží na orientaci vozidla, jak již bylo zmíněno v kapitole 3. Vozidlo v tomto případě lze však interpretovat pomocí obou způsobů uložení bodů, jelikož je zde hraniční orientace, nelze tedy rozhodnout, zda-li je vozidlo snímané z levé nebo pravé strany.

5.3 Vyhodnocení regrese v rovině silnice

Tento experiment porovnává ručně určenou pozici vozidla s pozicí odhadnutou pomocí konvoluční neuronové sítě. Experiment byl prováděn nad vozidly z pořízených záznamů, které byly použity pro experiment 5.1. Data z těchto záznamů jsou součástí datové sady, na které byla trénovaná neuronová síť. Pro účel tohoto experimentu byla použita pouze validační část těchto dat, jelikož obsahuje snímky vozidel, které nebyly viděny neuronovou sítí. Celkově se jedná o dvacet sedm vozidel.

Příprava dat

K vyhodnocení experimentu byly potřeba ručně anotované podstavy vozidel, vozidla z pořízených záznamů byla již anotována při tvorbě datové sady. Dále byla provedena regrese podstavy nad skupinou vozidel pomocí konvoluční neuronové sítě, která byla natrénovaná v rámci experimentu 5.2. Výsledné podstavy byly promítnuty do roviny silnice pomocí homografie. Použity k tomu byly matice homografie získané při kalibraci scény z předchozího experimentu 5.1. Příklad promítnutných podstav v souřadném systému silnice jsou vidět na grafu 5.10.



Obrázek 5.10: Projekce podstav do souřadného systému silnice. Graf obsahuje ručně anotovanou podstavu p (oranžová) a podstavu odhadnutou neuronovou sítí q (modrá).

Umístění kamery	IoU (%)	Vzdálenost (m)	Úhel (deg)
Most	64.61	0.23	0.06
Kancelář	72.24	0.26	0.15
průměr	68.51	0.24	0.10

Tabulka 5.4: Výsledky experimentu vyhodnocení regrese podstavy v rovině silnice. Uvedeno je zde porovnání podstav ze záznamů z kamery umístěné na mostě a v kanceláři, viz 5.1.

Výsledky experimentu

Porovnání výsledků bylo prováděno pomocí tří parametrů. Použity byly stejné parametry jako při experimentu 5.1, tedy průnik přes sjednocení (IoU), vzdálenost od středů podstav a rozdíl úhlů natočení podstav.

Výsledky experimentu jsou uvedeny v tabulce 5.4. Průměrná odchylka ve vzdálenosti podstav je $0,24$ metrů. Tato vzdálenost je v porovnání s velikosti vozidla opravdu velmi malá. Minimálně je i úhel natočení obou podstav. Průnik nad sjednocením dosahuje hodnoty 68.51%.

Kapitola 6

Závěr

Cílem této práce bylo sestavit metodu pro odhad pozice vozidel ze snímků pořízených dopravními kamerami. Nejprve byly shrnuty existující metody pro detekci a odhad pozice vozidel a probrány byly i existující datové sady. Při zkoumání datových sad vznikla potřeba si sestavit vlastní datovou sadu vhodnou pro experimenty. Datová sada byla sestavena ze snímků automobilů z různých zdrojů. Při tvorbě datové sady byla použita i ruční anotace pozice. Byla navržena metoda pro odhad pozice vozidla. Metoda se skládá z počáteční detekce automobilu na snímku z dopravní kamery, výřez automobilu je poté zpracován konvoluční neuronovou sítí, která provede regresi podstavy vozidla. Tato podstava je poté promítnuta do roviny silnice pomocí homografie. V rámci experimentů bylo popsáno trénovaní konvoluční neuronové sítě, její porovnání s ostatními možnostmi pro regresi podstavy vozidla a shrnuty zde byly i problémové případy. Proveden byl i experiment týkající se ruční anotace pozice vozidel, ve kterém byla zkoumaná přesnost zadávaní podstavy automobilů skupinou uživatelů.

Další vývoj práce by mohl obsahovat zvýšení přesnosti metody pro regresi podstavy, zejména se zaměřením na okrajové případy. Další zlepšení se může nacházet u projekce bodů podstavy pomocí homografie. Homografie vyžaduje ruční kalibraci scény. Bylo by vhodné se zaměřit na možnost provádět tuto kalibraci automaticky.

Literatura

[1] Algorithmia. *Introduction to optimizers* [online]. Algorithmia, květen 2018 [cit. 2020-05-16]. Dostupné z: https://algorithmia.com/blog/introduction-to-optimizers.

[2] Avinash. *Understanding Convolutional Pose Machines* [online]. Medium, červen 2019 [cit. 2020-04-15]. Dostupné z: https://medium.com/@avinashselvam/understanding-convolutional-pose-machines-472604578d26.

[3] Behrendt, K. Boxy Vehicle Detection in Large Images. In: *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*. 2019, s. 840–846. ISSN 2473-9944.

[4] Brownlee, J. *How Do Convolutional Layers Work in Deep Learning Neural Networks?* [online]. Machine Learning Mastery, duben 2019 [cit. 2020-05-22]. Dostupné z: https://machinelearningmastery.com/convolutional-layers-for-deep-learning-neural-networks/.

[5] Brownlee, J. *How to Choose Loss Functions When Training Deep Learning Neural Networks* [online]. Machine Learning Mastery, leden 2019 [cit. 2020-05-16]. Dostupné z: https://machinelearningmastery.com/how-to-choose-loss-functions-when-training-deep-learning-neural-networks/.

[6] Cortes, C. a Vapnik, V. Support-vector networks. *Chem. Biol. Drug Des.* Leden 2009, sv. 297, s. 273–297. ISSN 1747-0277.

[7] Dalal, N. a Triggs, B. Histograms of oriented gradients for human detection. In: *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. 2005, sv. 1, s. 886–893 vol. 1. ISSN 1063-6919.

[8] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K. et al. *ImageNet: A Large-Scale Hierarchical Image Database*. IEEE, 2009.

[9] Dubrofsky, E. *Homography Estimation*. Vancouver, 2009. [cit. 2020-05-13]. Diplomová práce. The University of British Columbia.

[10] Dubska, M. a Herout, A. Real Projective Plane Mapping for Detection of Orthogonal Vanishing Points. In:. Leden 2013, s. 90.1–90.11. ISBN 1-901725-49-9.

[11] Dubska, M., Sochor, J. a Herout, A. Automatic Camera Calibration for Traffic Understanding. *BMVC 2014 - Proceedings of the British Machine Vision Conference 2014*. Leden 2014. ISSN 0000-2014.

[12] Fang, J., Zhou, L. a Liu, G. 3D Bounding Box Estimation for Autonomous Vehicles by Cascaded Geometric Constraints and Depurated 2D Detections Using 3D Results. In:. Září 2019, abs/1909.01867. ISSN 2331-8422.

[13] Gandhi, R. *R-CNN, Fast R-CNN, Faster R-CNN, YOLO — Object Detection Algorithms* [online]. Towards data science, červenec 2018 [cit. 2020-04-11]. Dostupné z: https://towardsdatascience.com/r-cnn-fast-r-cnn-faster-r-cnn-yolo-object-detection-algorithms-36d53571365e.

[14] Geiger, A., Lenz, P. a Urtasun, R. *Are we ready for autonomous driving? The KITTI vision benchmark suite*. 2012. ISSN 1063-6919.

[15] Girshick, R., Donahue, J., Darrell, T. a Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In: *2014 IEEE Conference on Computer Vision and Pattern Recognition*. 2014, s. 580–587. ISSN 1063-6919.

[16] Girshick, R. Fast R-CNN. In: *2015 IEEE International Conference on Computer Vision (ICCV)*. 2015, s. 1440–1448. ISSN 2380-7504.

[17] He, K., Zhang, X., Ren, S. a Sun, J. Deep Residual Learning for Image Recognition. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE Computer Society, červen 2016, s. 770–778. ISSN 1063-6919.

[18] Hui, J. *What do we learn from single shot object detectors(SSD, YOLOv3)* [online]. Medium, březen 2018 [cit. 2020-04-20]. Dostupné z: https://medium.com/@jonathan_hui/what-do-we-learn-from-single-shot-object-detectors-ssd-yolo-fpn-focal-loss-3888677c5f4d.

[19] Kingma, D. a Ba, J. Adam: A Method for Stochastic Optimization. *International Conference on Learning Representations*. Prosinec 2014.

[20] Klein, L. A., Mills, M. K. a Gibson, D. R. *Traffic Detector Handbook:Third Edition—Volume II*. Turner-Fairbank Highway Research Center, 2006.

[21] Laan, C. *Real-time 3D car pose estimation trained on synthetic data* [online]. Laan Labs, březen 2019 [cit. 2020-04-15]. Dostupné z: https://labs.laan.com/blog/real-time-3d-car-pose-estimation-trained-on-synthetic-data.html.

[22] Lang, B. *An Introduction to Positional Tracking and Degrees of Freedom (DOF)* [online]. Road to VR, únor 2013 [cit. 2020-04-16]. Dostupné z: https://www.roadtovr.com/introduction-positional-tracking-degrees-freedom-dof/.

[23] Lienhart, R. a Maydt, J. An extended set of Haar-like features for rapid object detection. In: *Proceedings. International Conference on Image Processing*. 2002, sv. 1, s. I–I. ISSN 0000-2003.

[24] Lin, T., Dollár, P., Girshick, R., He, K., Hariharan, B. et al. Feature Pyramid Networks for Object Detection. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, s. 936–944. ISSN 1063-6919.

[25] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P. et al. Microsoft COCO: Common Objects in Context. *CoRR*. Květen 2014, abs/1405.0312. ISSN 0302-9743.

[26] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S. E. et al. SSD: Single Shot MultiBox Detector. *CoRR*. Prosinec 2015, abs/1512.02325. ISSN 0302-9743.

[27] Lowe, D. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*. Listopad 2004, sv. 60, s. 91–. ISSN 0920-5691.

[28] Matzen, K. a Snavely, N. NYC3DCars: A Dataset of 3D Vehicles in Geographic Context. In: *2013 IEEE International Conference on Computer Vision*. 2013, s. 761–768. ISSN 2380-7504.

[29] Mousavian, A., Anguelov, D., Flynn, J. a Košecká, J. 3D Bounding Box Estimation Using Deep Learning and Geometry. In:. červenec 2017, s. 5632–5640. ISSN 1063-6919.

[30] Naphade, M., Chang, M.-C., Sharma, A., Anastasiu, D. C., Jagarlamudi, V. et al. *The 2018 NVIDIA AI City Challenge*. 2018. 53–60 s.

[31] Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E. et al. Automatic differentiation in PyTorch. In: *NIPS-W*. 2017.

[32] Redmon, J. a Farhadi, A. YOLO9000: Better, Faster, Stronger. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, s. 6517–6525. ISSN 1063-6919.

[33] Redmon, J., Divvala, S., Girshick, R. a Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, s. 779–788. ISSN 1063-6919.

[34] Redmon, J. a Farhadi, A. YOLOv3: An Incremental Improvement. *ArXiv*. Duben 2018, abs/1804.02767, s. 1–6. ISSN 2331-8422.

[35] Ren, S., He, K., Girshick, R. a Sun, J. *Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks*. 2017. ISSN 1939-3539.

[36] Sachan, A. *Guide to Object Detection using Deep Learning: Faster R-CNN,YOLO,SSD* [online]. CV tricks, září 2018 [cit. 2020-04-14]. Dostupné z: https://cv-tricks.com/object-detection/faster-r-cnn-yolo-ssd/.

[37] Saha, S. *A Comprehensive Guide to Convolutional Neural Networks* [online]. Towards Data Science, prosinec 2018 [cit. 2020-05-22]. Dostupné z: https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53.

[38] Simonyan, K. a Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *ArXiv 1409.1556*. Září 2014. ISSN 2331-8422.

[39] Sochor, J., Herout, A. a Havel, J. BoxCars: 3D Boxes as CNN Input for Improved Fine-Grained Vehicle Recognition. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, s. 3006–3015. ISSN 1063-6919.

[40] Sochor, J., Špaňhel, J. a Herout, A. BoxCars: Improving Fine-Grained Recognition of Vehicles Using 3-D Bounding Boxes in Traffic Surveillance. *IEEE Transactions on Intelligent Transportation Systems*. 2019, sv. 20, č. 1, s. 97–108. ISSN 1558-0016.

[41] Wang, Z., Tang, L., Liu, X., Yao, Z., Yi, S. et al. Orientation Invariant Feature Embedding and Spatial Temporal Regularization for Vehicle Re-identification. In: *2017 IEEE International Conference on Computer Vision (ICCV)*. 2017, s. 379–387. ISSN 2380-7504.

[42] Wei, S.-E., Ramakrishna, V., Kanade, T. a Sheikh, Y. Convolutional Pose Machines. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, s. 4724–4732. ISSN 1063-6919.

[43] Zhao, Z.-Q., Xu, S. tao a Wu, X. Object Detection with Deep Learning: A Review. *IEEE Transactions on Neural Networks and Learning Systems*. Leden 2019, PP, s. 1–21. ISSN 2162-2388.

Příloha A

Obsah přiloženého paměťového média

Adresářová struktura přiloženého paměťového média:

- **source/** - adresář se zdrojovými kódy
- **data/** - adresář s datovou sadu
- **model/** - adresář s natrénovaným modelem neuronové sítě
- **video/** - adresář s videem
- **text/** - adresář s textovou částí této práce
- **README.md** - detailnější popis obsahu média