ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

Fakulta elektrotechnická

Katedra radioelektroniky

Automatická detekce a identifikace registračních značek vozidel

BAKALÁŘSKÁ PRÁCE

Vedoucí práce: Ing. Stanislav Vítek, Ph.D.

Autor: Petr Melničuk, KME, Multimediální technika

květen 2015

České vysoké učení technické v Praze
Fakulta elektrotechnická

katedra radioelektroniky

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student: **Petr Melničuk**

Studijní program: Komunikace, multimédia a elektronika
Obor: Multimediální technika

Název tématu: **Automatická detekce a identifikace registračních značek vozidel**

Pokyny pro vypracování:

1. Prostudujte problematiku rozpoznávání registračních značek vozidel.
2. Na základě studie navrhněte komplexní kamerový systém pro detekci a identifikaci registrační značky vozidla.
3. Navržený systém implementujte.
4. Diskutujte vliv světelných podmínek na účinnost použitých algoritmů.

Seznam odborné literatury:

[1] Shan D., Ibrahim M., Shehata M., Badawy W., Automatic License Plate Recognition (ALPR): A State-of-the-Art Review, IEEE Transactions on Circuits and Systems for Video Technology, Vol. 23, No.2, pp.311-325, Feb. 2013
[2] Bradski G., Kaehler A., Learning OpenCV: Computer Vision with the OpenCV Library, O'Reilly Media, 2008, ISBN 978-0596516130

Vedoucí: Ing. Stanislav Vítek, Ph.D.

Platnost zadání: do konce letního semestru 2015/2016

L.S.

doc. Mgr. Petr Páta, Ph.D.
vedoucí katedry

prof. Ing. Pavel Ripka, CSc.
děkan

V Praze dne 10. 2. 2015

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

Dne 20. května 2015 v Praze

.....

Petr Melničuk

Poděkování

Děkuji panu Ing. Stanislavovi Vítkovi, Ph.D. za odborné vedení, vstřícnost a rady při vypracovávání této bakalářské práce.

Obsah

1. Úvod 8

2. Využití identifikace registračních značek 10

2.1 Dopravní systémy 10

2.2 Systémy spojené s vynucováním práva 14

2.3 Ochrana osobních údajů 15

3. Aspekty návrhu systémů 16

3.1 Vzhled registrační značky 17

3.2 Defekty registračních značek 19

3.3 Získávání obrazu 20

3.4 Reprezentace a interpretace obrazu 24

4. Teorie zpracování obrazu 26

4.1 Základní operace s maticemi 27

4.2 Prahování 28

4.3 Geometrické transformace 30

4.4 Morfologické operace 32

4.5 OpenCV contours 34

4.6 Jednoduchá klasifikace obrazových dat 36

5. Přístupy k řešení problematiky 39

6. Praktická část 42

6.1 Návrh systému 43

6.2 Detekce registrační značky 46

6.3 Identifikace registrační značky 49

6.4 Spolehlivost implementovaného systému 52

6.5 Diskuze vlivu světelných podmínek 56

7. Závěr 58

Seznam obrázků a tabulek 59

Seznam použité literatury a zdrojů 61

Příloha 64

Abstrakt

Tato bakalářská práce se zabývá problematikou rozpoznávání registračních značek vozidel v obraze. Jedná se o transformaci obrazových dat, na jejímž výstupu se obvykle vyskytuje text registrační značky jednoznačně identifikující vozidlo. Cílem práce je návrh a implementace komplexního kamerového systému s touto schopností. Algoritmus, který byl implementován v programovacím jazyku Python s využitím knihovny OpenCV, transformuje obrazová data zejména pomocí morfologických operací a klasifikátoru nejbližšího souseda. Hlavním výstupem praktické části práce je systém, schopný v reálném čase za denních podmínek identifikovat české registrační značky ze záběru stacionární průmyslové kamery s úspěšností vyšší než 90 %.

Klíčová slova

počítačové vidění, zpracování obrazu, detekce registračních značek, identifikace registračních značek, knihovna OpenCV, optické rozpoznávání znaků, dopravní systémy, vynucování práva

Abstract

This bachelor thesis deals with problem of automatic license plate recognition (ALPR). ALPR is the transformation of an image based data into the text of license plate which is unique for every vehicle. The goal of this thesis is to design and to implement a complex camera system with the stated ability. The developed algorithm involves morphological operations and a nearest neighbour classifier. The programming language Python along with the OpenCV library was used for the purpose of implementation. The main result of practical part of this work is a stationary camera system capable of reading license plates in real-time with an over 90% success rate in daytime conditions.

Key Words

computer vision, image processing, license plate detection, license plate identification, OpenCV library, OCR, APNR, ALPR, traffic surveillance, law enforcement

Seznam zkratek

RZ	registrační značka
SPZ	státní poznávací značka
ALPR	automatic license plate recognition[US]
APNR	automatic number plate recognition[EN]
OCR	optical character recognition
GPS	global positioning system
CCD	charge-coupled device
CMOS	complementary metal-oxide-semiconductor
RGB	red, green, blue
ROI	region of interest
KNN	k-nearest neighbours
NN	nearest neighbour
HD	high definition
RAM	random-access memory

1. Úvod

Automatická detekce a identifikace registračních značek vozidel je souhrnným názvem pro procesy, při kterých dochází k převodu obrazových dat na informaci o registračních značkách v obraze. Registrační značka[1] (dále RZ) je ze zákona[2] povinným prvkem motorových vozidel, který obsahuje jednoznačné alfanumerické označení, unikátní pro každé vozidlo. Systémy automaticky extrahující toto označení z obrazu jsou v současnosti s rostoucím počtem osobních automobilů ve světě stále důležitější a užitečnější, zvláště ve sférách spojených s dopravou a vynucováním práva [1, 2].

S přístupem do vhodných databází je možné z označení RZ zjišťovat např. technický popis vozidla či informace o jeho majiteli. Jedním z hlavních subjektů s přístupem k takovým databázím bývá zejména policie. Názvy systémů, se kterými je možné se nejčastěji setkat v souvislosti s identifikací vozidel, jsou např.:

- detekce a záznam dopravních přestupků;
- měření úsekové rychlosti;
- analýza dopravy;
- pátrání po odcizených vozidlech;
- pátrání po pachateli trestného činu;
- závorové systémy.

Motivačním příkladem všeobecného přínosu by mohly být například velmi efektivní mobilní policejní systémy, jež se těší velké oblibě zejména v USA. Jeden z tamních systémů, dodávaný nadnárodní společností 3M, je umísťován na policejní hlídkové vozy a nepřetržitě automaticky skenuje RZ v okolí vozu. Přečtené značky okamžitě vyhledává v gigantické databázi a zároveň je ukládá do své paměti spolu s GPS souřadnicemi a časovým razítkem. Díky tomu mohou operátoři policejního vozu například velmi rychle reagovat při náhodném střetnutí s podezřelým vozidlem. Data z minulosti potom nalézají uplatnění hlavně při vyšetřovaní, umožňují například zpětně zjistit polohu pachatele před tím, než došlo ke spáchání zločinu [3].

Cílem této práce je zejména prostudování problematiky rozpoznávání RZ vozidel v obraze a implementace navrženého systému pro demonstraci osvojených znalostí.

[1] Starším a možná známějším názvem účelově stejného prvku v ČR je státní poznávací značka (SPZ).

[2] Zákon o podmínkách provozu vozidel na pozemních komunikacích. *56/2001 Sb*. 2001.

Implementovaný algoritmus je zapsán v programovacím jazyku Python s využitím OpenCV.

OpenCV (*Open Source Computer Vision Library*) je rychle se rozvíjející programová knihovna obsahující stovky optimalizovaných funkcí, usnadňujících práci s obrazem. Knihovna je vyvíjena v programovacím jazyku C a C++. Její využití je možné v několika programovacích jazycích na mnoha operačních systémech [4]. Další knihovny usnadňující v Pythonu manipulaci s obrazem, které stojí za zmínku, jsou například SciPy[3] nebo PIL[4].

Praktická část práce se zabývá hlavně systémy pracujícími bez databáze. Tedy systémy, které transformují vstupní obrazová data na textový kód odpovídající RZ v obraze bez možnosti systematického zaznamenávání či pozdějšího analyzování výstupních dat. Transformace obrazových dat je obvykle docíleno v několika úrovních [5]. V základním případě se jedná o detekci RZ, její separaci, separaci jednotlivých znaků a jejich rozpoznání. Z algoritmického hlediska se výstupů dosahuje pomocí pokročilých metod zpracování obrazu a funkcí z oblasti tzv. strojového učení.

Vstupními daty navrženého systému mohou být fotografie, videozáznamy nebo i aktuální záběry ze stacionárních průmyslových kamer. Spolehlivost implementovaného algoritmu v různých podmínkách je diskutována v jedné ze závěrečných kapitol.

Jako páteřní pramen literatury pro tuto práci figuruje především přehledový článek [5], shrnující stav řešení problematiky k roku 2013 a kniha *Learning OpenCV* [4], popisující především teoretickou část zpracování obrazu.

Zbytek dokumentu je členěn následovně. Ve 2. kapitole jsou popsány vybrané systémy využívající rozpoznávání RZ bez probírání principů samotného rozpoznávání. Kapitola č. 3 se zabývá především aspekty návrhu libovolných podobných diskutovaných systémů. Ve 4. kapitole je popsáno několik operací s obrazovými daty, které nejvíce souvisí s dále implementovanými algoritmy a zkoumanou oblastí. V 5. kapitole je stručně uvedeno několik možných přístupů k detekci a rozpoznání RZ tak, jak o nich informují vědecké články. V rámci 6. kapitoly je probrána praktická část práce, tedy návrh, implementace a testování vytvořeného systému včetně debaty o jeho nedokonalostech. Kapitola č. 7 obsahuje shrnutí celého dokumentu.

---

[3] *SciPy.org* [online]. [cit. 2015-04-10]. Dostupné z: http://www.scipy.org/

[4] *Python Imaging Library (PIL)* [online]. [cit. 2015-04-10]. Dostupné z: http://www.pythonware.com/products/pil/

2. Využití identifikace registračních značek

Tato kapitola se zabývá letmým popisem několika vybraných systémů, které využívají schopnosti čtení RZ z obrazu. Popis jednotlivých systémů je na úvod předložen bez probírání principů samotného rozpoznávání.

Na začátek postačí uvést, že tato tématika spadá do oboru tzv. počítačového vidění. Počítačové vidění je rychle rostoucí odvětví z oblasti výpočetní techniky a snímaní digitálního obrazu. Jde o souhrnný název pro získávání, zpracovávání, analyzování a rozpoznávání obrazových dat. Obecně se dá říci, že počítačové vidění je transformace dat digitálního obrazu na formu dat, která odpovídají požadované aplikaci [4]. Pro případ zde probíraných systémů je požadovanou výstupní formou dat text RZ (Obr. 1).

Obr. 1: Zjednodušený pohled na vstup a výstup probíraných systémů

V úvodu práce byly nastíněny některé aplikace z oblasti dopravy a z oblasti související s vymáháním práva. Předem je nutno podotknout, že určité aplikace (např. měření úsekové rychlosti) se nedají příliš dobře zařadit jen do jedné oblasti.

2.1 Dopravní systémy

V dopravě nalézá schopnost čtení RZ v obraze uplatnění zejména v dopravních analýzách. Z perspektivy dopravních měření se libovolná video technologie snímající dopravu dá zařadit mezi tzv. neintrusivní detektory. Neintrusivní detektory jsou technické prvky pro měření dopravních dat, které se instalují bez zásahu do vozovky a fungují bezdotykově. Základní detektory měří veličiny, jako je například rychlost vozidel nebo intenzita dopravy [6]. Video technologie spolu se zpracováním obrazu přinesla do dopravních měření schopnost měřit data, která momentálně není jiná

technologie schopna zajistit. Právě díky identifikaci vozu je možné například měřit úsekovou rychlost, či provádět komplexní analýzy dopravy.

Princip dopravních systémů s rozpoznáváním RZ spočívá většinou ve využívání většího počtu strategicky umístěných stacionárních kamerových jednotek podél významných silničních spojení. Z každé jednotky poté vychází tok dat složený především z přečtených RZ a času jejich výskytu. Hromadnou statistickou analýzou dat z více stanovišť je možné například efektivně zjišťovat aktuální stav dopravy [7].

V České republice je hlavním dodavatelem poskytujícím popsané dopravní systémy společnost CAMEA s. r. o.[5] Tato společnost dodává plně certifikované systémy, jež je možné spatřit na mnohých místech v Praze. Nejtypičtěji se kamery, společně s potřebným vybavením, instalují na sloupy veřejného osvětlení (Obr. 2), ze kterých se dá vyvést napájení pro celý systém. Kamery bývají umístěny zhruba do výšky 5-8 m přímo nad silniční komunikaci a snímají vozidla přibližně ze vzdálenosti 10-20 m.

Obr. 2: Typické umístění kamer systémů od společnosti CAMEA s. r. o.

Měření dojezdové doby za pomoci identifikace vozidel

Velmi podstatnou veličinou v silniční dopravě je čas, se kterým blízce souvisí pojem dojezdová doba. Dojezdová doba je v podstatě průměrný čas potřebný pro překonání určité vzdálenosti mezi dvěma místy v určitý moment [7]. Tuto dobu samozřejmě ovlivňuje mnoho faktorů (stav dopravy, nejvyšší povolená rychlost, dopravní nehody na trase, výjimečné situace a další).

Nehledě na zkoumání těchto faktorů je možné efektivně zjišťovat dojezdovou dobu s pomocí identifikace vozidel. Základní myšlenka spočívá ve sběru textů RZ spolu

[5] *CAMEA, spol. s r. o.* [online] [cit. 2015-04-15]. Dostupné z: http://www.camea.cz/cz/

s časovými razítky na dvou různých místech, na začátku ***(textZAC; tZAC)*** a na konci ***(textKON, tKON)*** vybraného silničního úseku. Délka vybraného úseku se typicky pohybuje v rozmezí jednotek až desítek kilometrů.

$$\boldsymbol{textZAC; tZAC} = (textZAC_1, textZAC_2 \ldots textZAC_n; tZAC_1, tZAC_2 \ldots tZAC_n)$$
$$\boldsymbol{textKON; tKON} = (textKON_1, textKON_2 \ldots textKON_n; tKON_1, tKON_2 \ldots tKON_n)$$

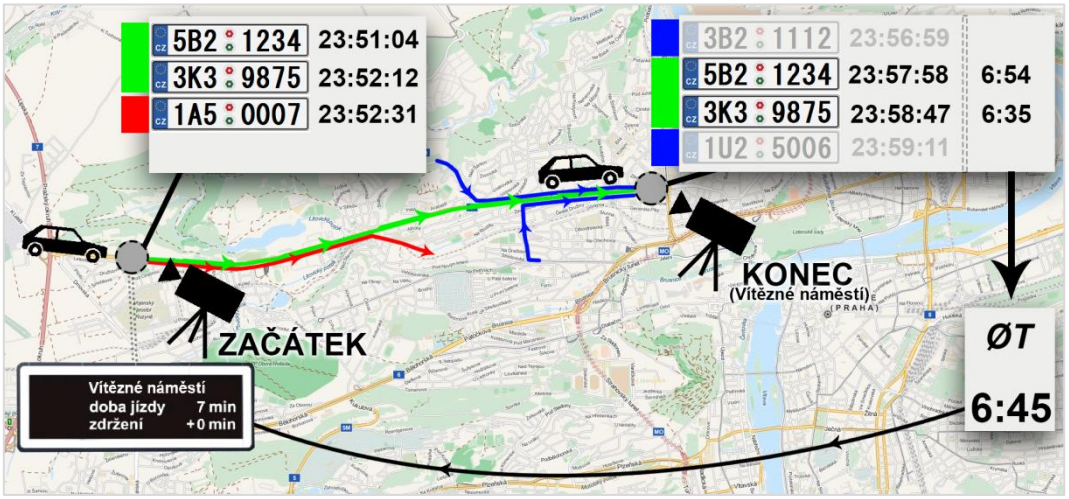
Pokud se ve výsledcích z obou lokací nalezne shoda textu RZ, je snadno vypočtena doba jízdy jednoho vozidla T mezi začátkem a koncem úseku.

např.:

$$textZAC_{82} = textKON_{125}$$
$$T = tKON_{125} - tZAC_{82} \text{ [s]}$$

Doba jízdy jednoho vozidla však není relevantním údajem popisujícím dobu dojezdu. Proto se najdou všechny shody mezi proměnnými ***textZAC*** a ***textKON*** a vypočítají se všechny doby jízd jednotlivých vozidel $\boldsymbol{T} = (T_1, T_2, \ldots T_k)$. Dalším krokem je statistická analýza proměnné ***T***. Je nutné vyloučit extrémní případy a vybrat časy odpovídající vozidlům, které jeli přímo a nenuceně nezastavovali během jízdy. Aritmetický průměr vybraných prvků z proměnné ***T*** odpovídá konečnému výsledku dojezdové doby [7].

Výsledky dojezdové doby se mohou každou chvíli aktualizovat a je možné je okamžitě zobrazovat na informační tabuli v místě začátku zkoumaného úseku. Celý proces je zjednodušeně ilustrován na Obr. 3.



Obr. 3: Ilustrace[6] systému pro měření dojezdové doby

[6] Podkladová mapa: *OpenStreetMap* [online]. [cit. 2015-04-24]. Dostupné z: https://www.openstreetmap.org/

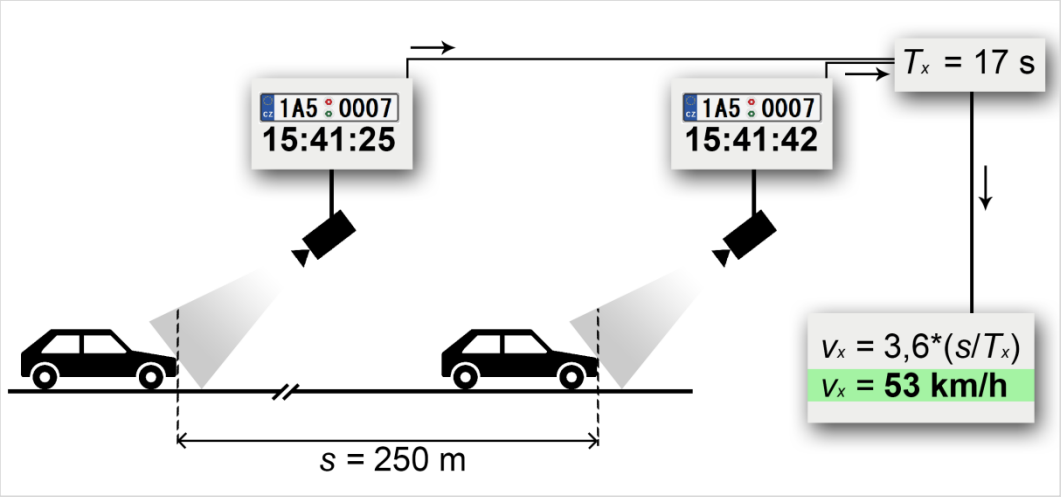
Měření úsekové rychlosti

Další důležitou a často měřenou veličinou v dopravě je rychlost vozidel. Tato veličina nezajímá pouze dopravní analytiky, ale zejména i policii. Existuje mnoho typů stacionárních i mobilních měřidel, využívajících různých prostředků (radar, infračervené záření, ultrazvuk) ke zjištění okamžité rychlosti vozidla [6]. Obecnou nevýhodou měřidel okamžité rychlosti je fakt, že měří rychlost pouze ve velmi malém výřezu komunikace a tak řidiči obvykle za místem měření beztrestně zrychlují. Naproti tomu úsekové měření rychlosti měří průměrnou rychlost v delším výřezu komunikace typicky zhruba 100-2000 m. To nutí řidiče, kteří se chtějí vyhnout postihu, dodržovat maximální povolenou rychlost po delší dobu [8].

Princip systémů měření úsekové rychlosti se neliší od systémů sloužících k měření dojezdové doby. Funkce opět spočívá ve sběru textů RZ společně s časovými razítky na dvou různých místech. Stejným způsobem jako u měření dojezdové doby jsou změřeny i doby jízd jednotlivých vozidel mezi začátkem a koncem úseku $\boldsymbol{T} = (T_1, T_2, \ldots T_k)$. Se znalostí konstantní délky úseku (s) je pak možné spočítat úsekové rychlosti ($\boldsymbol{v}$) jednotlivých vozidel.

$$\boldsymbol{v} = 3{,}6*(s/\boldsymbol{T}) \text{ [km/h]}$$
$$\boldsymbol{v} = (v_1, v_2, \ldots v_k)$$

Pokud je detekováno vozidlo, které svou průměrnou rychlostí překročilo nejvyšší povolenou rychlost, předá se text jeho RZ společně s důkazním obrázkem příslušným orgánům. Ilustrace celého procesu při průjezdu jednoho vozidla je zobrazena níže (Obr. 4).



Obr. 4: Ilustrace[7] systému pro měření úsekové rychlosti

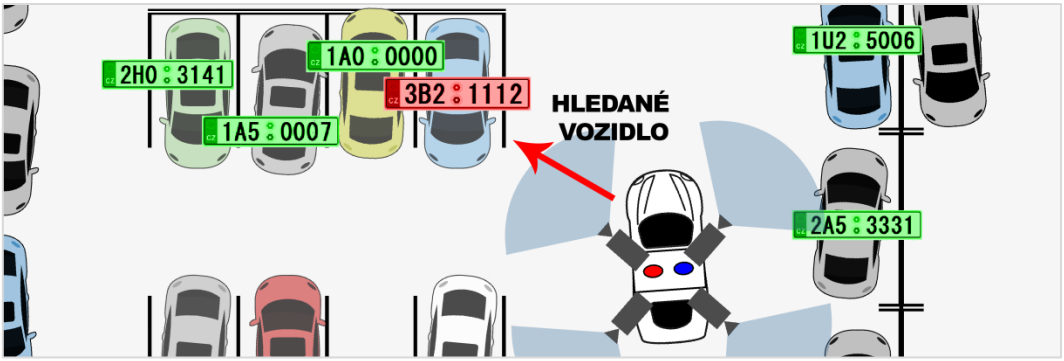
[7] Schématické značky: *Peníze.cz* [online].[cit. 2015-04-20]. Dostupné z: http://spz.penize.cz/

2.2 Systémy spojené s vynucováním práva

Mezi systémy spojené s vynucováním práva se dají zařadit aplikace, které pomáhají čtením RZ příslušným oprávněným orgánům (nejčastěji policii) prosazovat zákon. Mnoho takových aplikací se týká dopravních přestupků. V předchozí části (viz kapitola 2.1) bylo například zmíněno měření úsekové rychlosti. Dalšími měřitelnými přestupky jsou například jízda na červenou, porušení zákazu vjezdu, jízda v protisměru a další. Tyto zmíněné aplikace většinou mohou používat velmi podobný nebo i stejný hardware (Obr. 2) a software jako již zmiňované dopravní systémy.

Přeprava vozidly hraje významnou roli v životech lidí a dopravní přestupky nejsou bohužel jediné typy protiprávního jednání spojeného s vozidly. Velké množství menších i větších zločinů je spjato s přepravou osob či věcí. Čtení RZ dává vyšetřovatelům nástroj, se kterým jsou schopni řešit zločiny, které by jinak schopní řešit nebyli [3]. V principu jednoduchým systémem pro demonstraci je například systém hledání odcizených vozidel. Po nahlášení odcizení vozu je zadána jeho RZ do databáze hledaných vozů. Poté už stačí, aby se zloděj s autem objevil na místě, kde se vyskytuje systém schopný přečíst jeho RZ. Pokud dojde k přečtení hledané RZ, ohlásí se lokace a čas výskytu odcizeného vozidla příslušným osobám, které pak mohou vůz snadněji lokalizovat. Systém může využívat i stacionární kamerové jednotky, které jsou primárně k jiným účelům, například k měření dojezdové doby nebo může mít kamery vlastní [3].

Mocnou pátrací sílu představují mobilní jednotky umísťované na policejní vozy. Jeden z takových systémů byl zmíněn v úvodu práce, jeho ilustrační snímek při nalezení hledaného vozidla průjezdem parkoviště je zobrazen na Obr. 5. Vozidlo neustále skenuje své okolí, čte RZ a porovnává je s údaji v databázi. Systémy s mobilními jednotkami se užívají zejména ve Spojených státech a Velké Británii [3].



Obr. 5: Policejní vůz vybavený systémem čtení RZ[8]

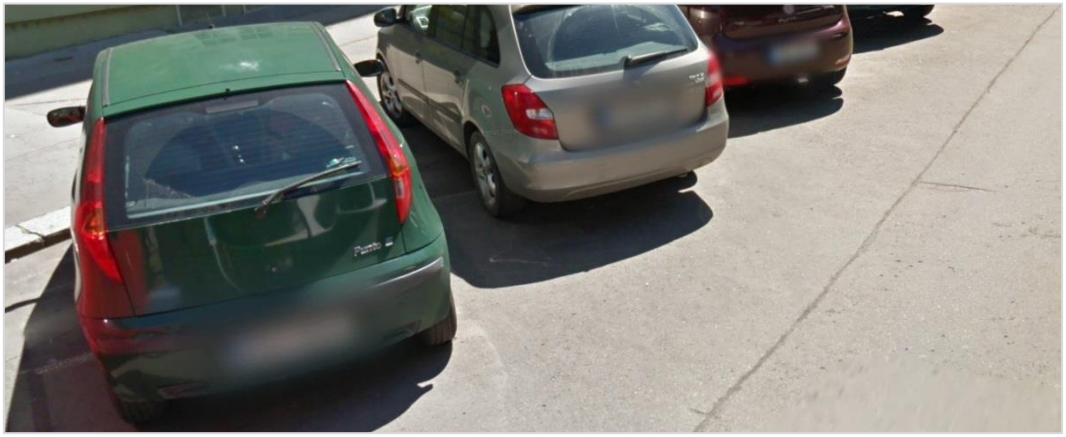
[8] Schématické značky: *Peníze.cz* [online]. [cit. 2015-04-20]. Dostupné z: http://spz.penize.cz/

2.3 Ochrana osobních údajů

V přechozích sekcích byly zmíněny aplikace spojené s dopravou a s vynucováním práva. Všechny tyto systémy čtou RZ a ukládají jejich text spolu s časovým razítkem. Podle úřadu na ochranu osobních údajů je text RZ údajem osobním, protože umožňuje nepřímo dohledat informace o majiteli vozu. Systematické uchovávání či sdílení takových informací není standardně povoleno[9].

Je zřejmé, že systémy detekce a rozpoznávání RZ s osobními údaji manipulují a musí tedy ctít jejich ochranu. Existují však i algoritmy, jejichž úkolem je naopak znemožňovat manipulaci s osobními údaji ve formě vyfocených nebo natočených RZ. Princip funkce je obvykle založen na spolehlivé detekci RZ a jejího následného rozmazání tak, aby nebyla čitelná.

V praxi se lze setkat s uplatněním těchto algoritmů nejčastěji v hromadně a automaticky sbíraných obrazových datech. Za největší kolekci takových obrazových se dá považovat aplikace Google Street View. Jde o gigantickou kolekci 360° panoramat získaných z úrovně ulic. Aby Google předcházel případným obviněním ze sdílení osobních údajů, rozmazává ve svých obrázcích citlivé údaje, jako jsou právě RZ (Obr. 6) nebo obličeje. Algoritmy plnící tuto činnost musí být kvůli množství zpracovávaných dat plně automatizované, spolehlivé a rychlé [9].



Obr. 6: Rozmazávání RZ v Google Street View[10]

Obrázky s kompletně čitelnou RZ v tomto dokumentu jsou použity se svolením majitele vozu. U fotografií značek ostatních došlo k rozmazání několika znaků. Schématické značky jsou smyšlené.

[9] K celé problematice je možné zjistit více informací na webových stránkách Úřadu pro ochranu osobních údajů: *Úřad pro ochranu osobních údajů* [online]. [cit. 2015-05-10]. Dostupné z: https://www.uoou.cz/

[10] *Mapy Google* [online]. 2014 [cit. 2015-04-25]. Dostupné z: https://www.google.cz/maps

3. Aspekty návrhu systémů

Návrh a vývoj spolehlivého systému detekce a rozpoznávání RZ není jednoznačným a jednoduchým úkolem [10]. Existuje snaha o vytvoření ideálního systému, který si lze představit jako nástroj, který dokáže číst RZ z obrázků minimálně stejně spolehlivě jako člověk. Proto je vhodné si zprvu uvědomit rozdíly mezi lidským vizuálním systémem a libovolným systémem počítačového vidění, do kterého čtení RZ z obrazu spadá.

Pro člověka je zrak jeho nejvýznamnějším smyslem. Jednou z jeho nejpozoruhodnějších funkcí je to, že umožňuje lidem díky vizuálním zkušenostem detekovat a klasifikovat objekty, které nikdy dříve neviděli. Lidé se této schopnosti učí již od nejútlejšího věku. S dospíváním organismu a nárůstem vizuálních zkušeností se přirozeným učením v mozku staví velice silná a stabilní základna zrakových vjemů, která umožňuje výše zmíněnou schopnost [11].

Výkon lidského vizuálního systému z pohledu schopnosti porozumění vizuálním vjemům zdaleka přesahuje nejmodernější systémy počítavého vidění. Lidé jsou schopni správně chápat vjemy téměř okamžitě a to až v neuvěřitelně širokém spektru vnějších podmínek [11].

Přestože systémy popisované v kapitole č. 2 dokáží mnohdy detekovat a rozpoznávat RZ rychleji než člověk, jejich spolehlivost značně závisí na splnění mnoha předpokladů. Počítačové systémy jsou totiž většinou vytvářeny a vyladěny pouze pro užší okruh aplikací při určitých podmínkách [11].

Při návrhu systému detekce a rozpoznávání RZ je nutné zohlednit skutečnosti, které mohou ovlivňovat jeho výslednou spolehlivost.

Za fundamentální předpoklad pro úspěšnou interpretaci se dá považovat dostatečná kvalita vstupních obrazových dat, což je obecně spojeno především s jejich získáváním. Existuje však mnoho dalších faktorů, které se podle [5] dají sdružit do dvou základních kategorií. První kategorie se týká zejména vzhledových variací RZ a bude zde probrána jako první. Druhá kategorie se týká variací prostředí, čímž se myslí hlavně vliv světelných podmínek a geometrie snímání.

3.1 Vzhled registrační značky

Jak již bylo uvedeno, RZ je ze zákona povinným prvkem motorových vozidel umožňujícím jejich identifikaci. Z perspektivy probíraných systémů je nejpodstatnější vlastností RZ jejich vzhled. Obvykle se jedná o obdélníkovou plechovou tabulku s kontrastním textem, tvořeným kombinací písmen a čísel. Tento text tvoří unikátní kód pro každé vozidlo.

Formou, obsahem a provedením RZ se v České republice zabývá ministerstvo dopravy. Aktuální vyhláška stanovuje několik možných typů RZ pro osobní automobily, motocykly, autobusy, či speciální vozidla [12]. Zřejmě nejčastěji se vyskytujícím typem RZ v ČR je typ 101. Jedná se o tabulku v rozměru 52 x 11 cm tvořenou sedmi černými znaky na bílém podkladu (Obr. 7).

Obr. 7: Nejčastěji se vyskytující typ RZ v ČR, typ 101 [12]

Kromě typu 101 existuje v ČR relativně velké množství regulérních vzhledů RZ. Existují značky s jiným poměrem stran, rozličnými barevnými podklady i znaky, jiným počtem znaků a mezerami mezi nimi. Při pohledu do minulosti navíc vyplyne skutečnost, že forma, obsah ani provedení RZ není rozhodně časově stabilní. Momentálně se například uvažuje o možnosti zavedení RZ na přání, taková značka by pak mohla obsahovat téměř libovolné seskupení povolených znaků (Obr. 8).

Obr. 8: Povolené znaky na RZ a SPZ v ČR [12]

Kromě českých RZ samozřejmě existují i značky cizích států, které se řídí jinými nařízeními, a proto je těžké, ne-li nemožné předložit definici standardního vzhledu RZ. Rozmanitost značek českých i zahraničních zobrazuje Obr. 9.

Obr. 9: Rozmanitost RZ[11] (poměr velikostí nedodržen)

V souvislosti s výše uvedeným vyplývá, že při vytváření robustního systému schopného rozpoznávat RZ více států je vhodné se zaměřit na jejich společné vlastnosti, nikoliv na přesný vzhled. Společnými vlastnostmi je až na malé výjimky například čtyřúhelníkový tvar nebo existence kontrastního přechodu mezi znaky a pozadím.

Významnou fyzikální vlastností moderních RZ je jejich tzv. retroreflexe. Jde o schopnost materiálů odrážet většinu světla směrem zpět k jeho zdroji. Tento jev lze dobře pozorovat ve tmě, kdy jsou značky po nasvícení výborně vidět (Obr. 10). Původně začala být tato vlastnost značkám přidávána z důvodu zvýšení viditelnosti zaparkovaných nesvítících aut. Nyní retroreflexe představuje důležitý element, na který spoléhá při snímání valná část systému detekce a rozpoznávání RZ vozidel [13]. Využívání této vlastnosti bude probráno v kapitole 3.3.

Obr. 10: Ukázka retroreflexe RZ

[11] *Licence plates of the world* [online]. [cit. 2015-04-28]. Dostupné z: http://www.worldlicenseplates.com/

3.2 Defekty registračních značek

Již bylo zmíněno, že existuje velké množství typů RZ, které se vzhledově liší. Ve skutečnosti se vzhledové odlišnosti dají pozorovat i v rámci jednoho typu značek. Vozidla se pohybují v širokém spektru prostředí a RZ na vozech nebývá nijak speciálně chráněna vůči mechanickému poškození nebo špíně. Mnohdy tak vznikají defekty, které potom způsobují méně, či více závažný vzhledový rozdíl od značek zobrazovaných schematicky (např. Obr. 7). Míra defektů pak souvisí samozřejmě s čitelností značky a ovlivňuje spolehlivost diskutovaných systémů.

K zašpinění RZ může dojít velmi jednoduše například pohybem vozidla za nepříznivých metrologických podmínek nebo častým průjezdem místem se zvýšenou koncentrací prachu. Přestože je ze zákona povinností udržovat tabulku čistou a čitelnou [12], je možné se setkat s vozy, jejichž RZ je natolik špinavá, že rozlišit jednotlivé znaky dělá problém dokonce i člověku při bližším zkoumání.

Celoplošné zašpinění RZ způsobuje především snížení kontrastu mezi znaky a pozadím, což může mít za následek neschopnost značku detekovat. Lokální, malé kousky špíny (např. tečky od bláta nebo sražený hmyz) většinou příliš problémů nezpůsobují, obecně však velmi záleží na umístění, charakteru a rozsáhlosti nečistot.

Mechanické poškození RZ vzniká nejčastěji v důsledku drobných nehod, které nutně nevyžadují výměnu celé tabulky za novou. Riziková mechanická poškození jsou například promáčkliny, škrábance či praskliny. Jako velmi negativní mechanická vada se jeví koroze, ta však již není moc obvyklá a je spíše defektem velmi starých RZ. Ukázky některých vzhledových nedokonalostí RZ jsou zobrazeny na Obr. 11.

Obr. 11: Obvyklé defekty RZ

Při návrhu systému je vhodné počítat s faktem, že ne každé vozidlo má RZ zcela v pořádku. Defekty RZ v prvních dvou sloupcích (na Obr. 11) jsou relativně časté a kvalitní algoritmus by se s nimi měl být schopen vypořádat. Značku, kterou nepřečte z obrázku člověk, nebude s největší pravděpodobností schopen přečíst ani algoritmus. Příliš nečitelná značka již není ani tak problémem systému čtení RZ, jako spíš problémem majitele vozu, který se jeho provozováním může vystavovat postihu.

3.3 Získávání obrazu

Získávání obrazu je velice důležitým krokem v diskutovaných systémech. Získávají se digitální obrazová data, která si s sebou nesou určité charakteristiky související zejména se snímanou scénou a snímacím zařízením. Jejich kombinace tvoří prvotní element diskutovaných systémů a velice významně ovlivňuje jejich úspěšnost [5].

Snímací zařízení
Nástroje pro získávání digitálního obrazu jsou nejčastěji kamery a fotoaparáty. Jde o nástroje, které byly do značné míry inspirovány lidským zrakem a skládají se z účelově podobných prvků.

Člověk je prostřednictvím očí schopen vnímat část elektromagnetického záření v rozsahu vlnových délek přibližně od 400 do 700 nm (viditelné světlo). Světlo přicházející do oka je postupně opticky lámáno a dopadá na sítnici, na které se nacházejí fotoreceptory. Při dopadu světla na fotoreceptor se v něm začne odehrávat chemická reakce, jejímž výsledkem je signál putující skrze optický nerv do mozku.

Snímací zařízení napodobují lidský vizuální systém. Optická soustava přizpůsobuje obraz na velikost snímače. Snímač je ekvivalentem sítnice u člověka. Místo fotoreceptorů se na něm nachází fotodetektory. Výstupem fotodetektorů je elektrický signál putující do dalších obvodů zařízení, kde bývá mimo jiné digitalizován. Není podstatné se v této práci zabývat principem snímacích zařízení hlouběji, důležité jsou však jejich vlastnosti a parametry stanovující jaký charakter a kvalitu budou mít obrazová data.

Vlastnosti a parametry snímacích zařízení
Výběr snímacího zařízení by měl brát v potaz záměr následného zpracování obrazu. V současnosti jsou k dispozici zařízení s kvalitními CCD nebo CMOS senzory s rozlišením několika milionů bodů a schopnostmi snímat desítky, stovky, či dokonce

tisíce snímků za sekundu. Hlavní parametry snímacích zařízení související se schopnosti rozpoznávat RZ v obraze jsou zejména:

- rozlišení snímače;
- snímková frekvence a expoziční doba;
- citlivost a dynamika snímače;
- šumové vlastnosti;
- kompresní standard;
- parametry objektivu (často samostatný, vyměnitelný prvek).

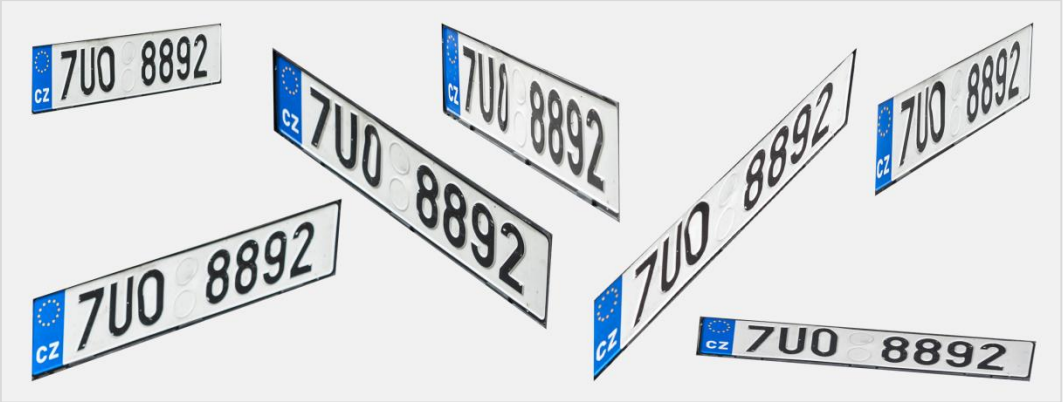
Při pořizování zařízení pro získávání obrazu jsou parametry rozlišení a snímková frekvence udávány ve specifikacích většinou na prvních místech. Vysoké rozlišení a vysoká snímková frekvence povětšinou znamenají zvýšení detailů sledované scény, ale také úměrně souvisí s objemem dat, který je nutno zpracovávat.

Zároveň je potřebné poznamenat, že kvalita a jemnost získaných obrazových dat nesouvisí pouze s rozlišením a snímkovou frekvencí. Například u některých levných zařízení bývá rozlišení interpolováno na vyšší, což ale většinou neznamená žádný markantní nárůst kvality.

Umístění a konfigurace zařízení

Výběr i vysoce kvalitního zařízení s prvotřídními parametry není postačujícím předpokladem pro maximální spolehlivost systému. V závislosti na navrhované aplikaci a předpokládaném prostředí použití je potřeba spekulovat o umístění a konfiguraci snímacího zařízení.

U stacionárních jednotek existuje snaha umísťovat snímací zařízení do pozic, ze kterých bude na RZ nejlepší pohled. Z hlediska algoritmu je optimálním úhlem snímání úhel vertikálně i horizontálně nulový vůči kolmici značky. To ovšem mnohdy nelze dodržet. Navíc v takovém úhlu může snadno docházet k zastínění značky jinými objekty. Typicky bývají stacionární kamery umísťovány nad úroveň střech vozidel, co nejméně horizontálně vychýleny od středu předpokládaného výskytu značek. Snímání značek z velkého úhlu (Obr. 12) všeobecně komplikuje jejich detekování i rozpoznávání, ale například u mobilních systémů se tomu nelze vyhnout, takové výsledné snímky si vyžadují při zpracování korekci perspektivy (viz kapitola 4.3).



Obr. 12: Pořízení snímku značky z různých úhlů

V mnoha aplikacích probíhá snímání RZ v pohybu. V zásadě existují tři možnosti vzájemného pohybu vozidla a snímacího zařízení: stacionární zařízení a pohybující se vozidlo, pohybující se zařízení a stojící vozidlo, pohybující se zařízení i vozidlo.

Vzájemný pohyb přináší zásadní požadavek na nastavení expoziční doby snímacího zařízení. Expoziční doba je čas, po který se akumuluje světlo dopadající na snímač. Dlouhá expoziční doba umožňuje snímači zachytit větší množství světla a získat tak například světlý obrázek i v noci. Při delších expozičních dobách je nutné pro zachování ostrého snímku více dbát na zajištění minimálního vzájemného pohybu scény a zařízení.

Při snímání vozidla jedoucího rychlostí 50 km/h expoziční dobou 1/6 s (167 ms) urazí vůz za dobu expozice přibližně vzdálenost 2 m. Tyto dva metry dokáží způsobit na snímku značné rozmazání celého vozu včetně jeho RZ, která se stává naprosto nečitelnou. Zkracováním expoziční doby se zmenšuje uražená vzdálenost vozu za dobu expozice a tím i rozmazání ve výsledném snímku (Obr. 13). Aby pohybové rozmazání nezpůsobovalo tragické zhoršování čitelnosti RZ, musí se použít dostatečně krátká expoziční doba odpovídající rychlosti vozidla a geometrickému uspořádání scény.

Obr. 13: Expoziční doba a vliv na pohybové rozmazání

Doba expozice není však parametrem, který by se dal libovolně měnit bez dalších důsledků na výsledný snímek. Zkrácením expoziční doby dochází ke snížení celkového množství akumulovaného světla ve snímači, výstupní signál je slabší a ve výsledku se snímek jeví tmavším.

Tmavší vzhled snímku se dá vykompenzovat dalšími dvěma parametry, ke kterým se expoziční doba váže a společně s nimi tvoří tzv. expoziční trojúhelník. Jde o označení pro trojici parametrů (expoziční doba, citlivost, clona), které se nastavují podle množství světla vstupujícího do objektivu a dalších požadavků. Kombinací těchto tří parametrů se nastavuje expozice. Za ideální expozici se v případě této problematiky dá považovat takové nastavení parametrů, které ve výsledném obrázku bude znamenat co největší dynamiku v oblasti RZ (prostřední na Obr. 14).

Obr. 14: Snímek RZ pořízen s různou expozicí

Malá dynamika v oblasti RZ značí v podstatě slabý kontrast přechodu mezi jednotlivými znaky a pozadím značky. S takovými snímky může mít řada systémů problém, protože na existenci zmíněného kontrastu spoléhá. Obecně probíhá nastavování expozice automaticky, polo-automaticky nebo plně manuálně na základě různých měřících technik. V případě diskutovaných systémů je snahou udržet luminanci RZ v oblasti s největší možnou dynamikou pro danou situaci.

Parametry expozičního trojúhelníku se mezi sebou z hlediska nastavení expozice doplňují. Pokud je například vyžadováno zkrácení expoziční doby na polovinu, je možné tuto změnu kompenzovat zvýšením citlivosti, nebo pootevřením clony. Charakter obrázku se však touto kompenzací mění. Zvyšování citlivosti představuje nárůst nežádoucího šumu a otevírání clony snižuje hloubku ostrosti. Malá hloubka ostrosti může komplikovat zaostření na snímané objekty, a proto se v extrémních případech dokáže jevit stejně nežádoucí jako šum.

Světelné podmínky prostředí

S nároky na krátkou expoziční dobu, minimální šum a maximální zaostření nelze za každých světelných podmínek přirozeně dosáhnout optimální expozice. Tento problém nastává typicky v nočních hodinách, kdy je množství světla vstupujícího do objektivu malé. Snaha o posun expozice do oblasti s maximální dynamikou pak většinou způsobí rozmazaný nebo příliš zašuměný snímek.

Od snímacího zařízení se však většinou požaduje zajišťovat dostatečně kvalitní snímky po celých 24 hodin. Za tmy se nabízí řešení s použitím zdroje světla přidaného ke snímacímu zařízení. Zdroj světla může zajistit, že snímky budou exponované podle potřeb s minimálním ohledem na vnější světelné podmínky. Po osvícení moderní RZ se navíc většina světla výhodně odráží zpět k jeho zdroji díky retroreflexi zmíněné v kapitole 3.1.

Použitím zdroje světla nesmí nedocházet k oslňování řidičů vozidel, proto pracují užívané světelné zdroje v oblasti infračerveného záření na vlnových délkách 850 nm. Výsledné snímky pořízené v infračerveném spektru s použitím infračerveného zdroje světla vykazují výborné vlastnosti pro následné zpracování. Na snímcích bývají dobře viditelné RZ a velká část zbytku snímku zůstává černou nebo velmi tmavou (podobně jako na Obr. 10) [13].

Tma ale není jediným problémem souvisejícím se světelnými podmínkami. Nastavení optimální expozice dále komplikuje například protisvětlo, různé odlesky nebo světla reflektorů vozidel. Nevyužití infračervené technologie snižuje schopnosti systému spolehlivě pracovat i za zhoršených světelných podmínek [13].

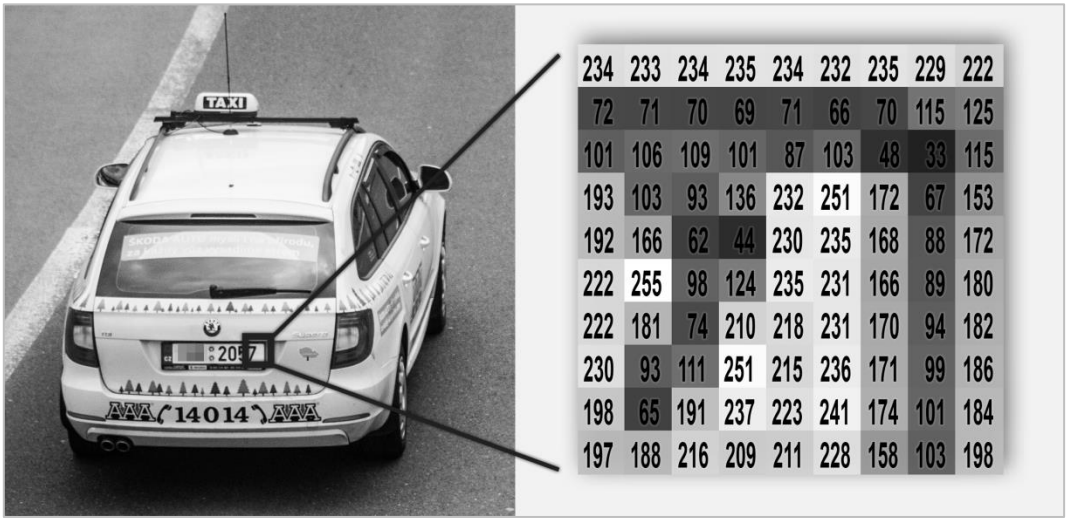
3.4 Reprezentace a interpretace obrazu

V celé této kapitole bylo nastíněno několik okolností, které ve výsledném snímku způsobují vzhledové odlišnosti RZ. Všechny tyto vzhledové odlišnosti vnímá člověk celkem snadno pozorováním a automatickou interpretací snímku. Ve způsobu reprezentace se ale vzhledově odlišná data v podstatě neliší a systémem je nelze interpretovat bez dalšího zpracovávání.

Po promítnutí obrazu skrz objektiv na jednotlivé body snímacího prvku dochází k převodu dopadajícího světla na elektrické signály, které se zesílí a digitalizují. Výstupní číselné hodnoty se ve svém základu neliší od jakýchkoliv jiných digitálních dat.

V případě barevného obrazu se obvykle jedná o 3 – rozměrnou matici obsahující čísla, která udávají intenzitu jednotlivých barevných složek příslušného bodu (pixelu) v obraze. Standardními barevnými složkami (kanály) jsou červená (R), zelená (G) a modrá (B), kterými lze z principu aditivního mísení barev vytvořit širokou škálu barevných odstínů. Standardní černobílý obraz je 2 – rozměrná matice čísel, jež udávají intenzitu jednotlivých bodů (Obr. 15). Nejčastěji bývá intenzita jednotlivých bodů

jednoho kanálu vyjadřována 8 bitovým číslem (256 možných úrovní). Celkovému množství bodů se říká rozlišení. Interpretace snímku zařízením se zpravidla dosahuje zpracováním popsaných matic [4].



Obr. 15: Reprezentace černobílého digitálního obrazu (výřez hrany RZ)

Systémy počítačového vidění se obecně dají rozdělit na snímací a interpretační část. Snímací část byla popsána v předchozí sekci (viz kapitola 3.2). Interpretační část je tvořena výpočetní jednotkou (počítač) schopnou podle sepsaného algoritmu provádět operace s obrazovými daty. Náročnost interpretace obrazových dat je dána především jejich velikostí a složitostí algoritmu vybrané aplikace.

Autor knihy [11], pojednávající o interpretaci obrazových dat obecně, uvádí, že je dobré, aby mezi jednotlivými částmi systémů existovala určitá harmonie z pohledu jejich výkonu. Například příliš slabá interpretační část neumožní dostatečně rychle zpracovávat velké množství obrazových dat dodávaných výkonným snímacím zařízením. Komponenty by měly být vůči sobě správně dimenzované.

Procesem souvisejícím s reprezentací obrazových dat je komprese (zmenšení jejich datového objemu). Komprese jsou rozdělovány na ztrátové a bezztrátové. Bezztrátové komprese odstraňují redundantní (nadbytečná) data a jde pouze o popsání dat efektivnějším způsobem. Ztrátové komprese jsou založeny na odstraňování irelevantních (nepodstatných) dat, tedy informací, které jsou pokud možno co nejméně významné pro výsledný subjektivní vjem. Ztrátová komprese může významně ovlivňovat spolehlivost algoritmů detekce a rozpoznávání RZ. Je třeba dbát na použité komprese a jejich význam pro danou aplikaci.

4. Teorie zpracování obrazu

Obrazová data jsou vícerozměrné matice obsahující čísla, která odpovídají intenzitám příslušné barevné složky (nejčastěji RGB) příslušného pixelu v obraze. Matematickými a logickými funkcemi aplikovanými na tuto množinou čísel se dá docílit mnoha obrazových operací [4].

V této kapitole bude stručně probráno několik užitečných operací s obrazovými daty, které nejvíce souvisí s dále implementovanými algoritmy a se zkoumanou oblastí. Procesy budou probrány z teoretického hlediska, s upozorněním na jejich praktické využití.

Implementaci se následně věnuje praktická část práce. Celý vytvořený algoritmus detekce a rozpoznávání RZ vozidel je poskládán především z právě zde probíraných bloků:

- základní maticové operace;
- prahování;
- geometrické transformace;
- morfologické operace;
- OpenCV contours;
- jednoduchá klasifikace obrazových dat.

Jako podkladová literatura pro celou kapitolu č. 4 slouží především kniha o OpenCV [4] a stejnojmenná online dokumentace [14]. Další odkazy, na zmíněné zdroje, nebudou proto v rámci této kapitoly uváděny.

4.1 Základní operace s maticemi

Základními operacemi s maticemi se myslí zejména elementární aritmetické operace s jednotlivými pixely a také procesy přistupující k libovolně velkému bloku bodů.

Elementární aritmetické operace

Mezi elementární aritmetické operace patří sčítání, rozdíl, násobení a dělení. Násobením určitým koeficientem a přičítáním konstanty ke všem bodům obrazu se dají například provádět globální úpravy jasu či kontrastu. Dále mají tyto elementární operace uplatnění například v systémech detekce pohybu. Sčítání pomáhá společně s dělením odhadovat tzv. snímek pozadí. Rozdílem snímku pozadí od aktuálního snímku se získají oblasti s pohybem (Obr. 16).

Obr. 16: Ilustrace systému detekce pohybu

Při modifikaci intenzity jednotlivých bodů je nutné ošetřit to, aby nebyla překročena jejich maximální či minimální možná hodnota (dáno bitovou hloubkou). V tu chvíli by docházelo k tzv. přetečení a většinou k nežádoucímu zkreslení výsledků.

ROI – oblast zájmu (*Region of interest*)

Často je žádané zpracovávat místo celého vstupního obrázku pouze jeho část (Obr. 17). Výběr menšího výřezu snímku zjednodušuje řešený problém a zvyšuje výkon celého systému. Umístění a velikost oblasti zájmu lze určovat manuálně či typicky na základě předcházejícího zpracování. Realizace obvykle spočívá ve využití indexování matic.

Obr. 17: Ilustrační případ vytvoření ROI

4.2 Prahování

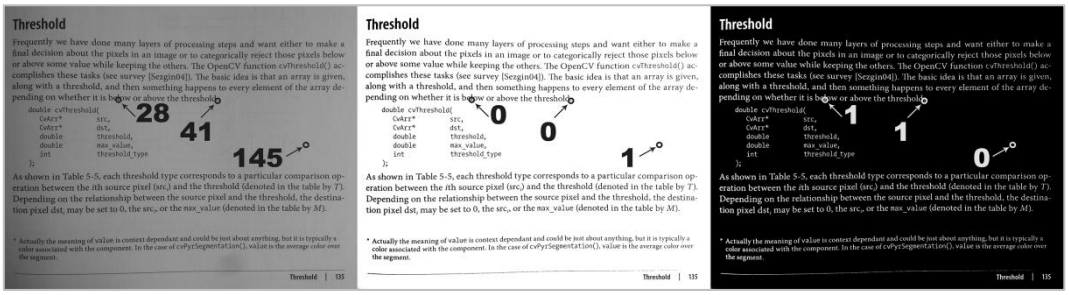
Prahování (*Threshold*) nachází uplatnění téměř v každém algoritmu počítačového vidění. Slouží k jistému zjednodušení obrazu a umožňuje například snadnější separaci objektů.

Vstupem pro funkci prahování bývá obvykle černobílý snímek tedy 2 – rozměrná matice čísel, jež udávají luminanci jednotlivých pixelů. Počet možných úrovní luminance odpovídá bitové hloubce (obvykle 8 bit – 256 úrovní). **Výstupem** prahování je stejně velká matice, jejíž prvky ale mohou nejčastěji nabývat pouze dvou stavů „0“ (černá) a „1“ (bílá). Jedná se o tzv. binární obraz.

Globální prahování

Princip globálního prahování spočívá v rozdělení všech pixelů do dvou skupin podle jejich luminance. Pixely, jejichž úroveň je vyšší než určitá mez (práh) se stanou „1“, naopak pixely s úrovní nižší se stanou „0“. V inverzním prahování je tomu naopak.

$$\textit{výstupní_snímek}_{i,j} = 1, \text{ když } \textit{vstupní_snímek}_{i,j} > \textit{práh}$$
$$\textit{výstupní_snímek}_{i,j} = 0, \text{ když } \textit{vstupní_snímek}_{i,j} < \textit{práh}$$

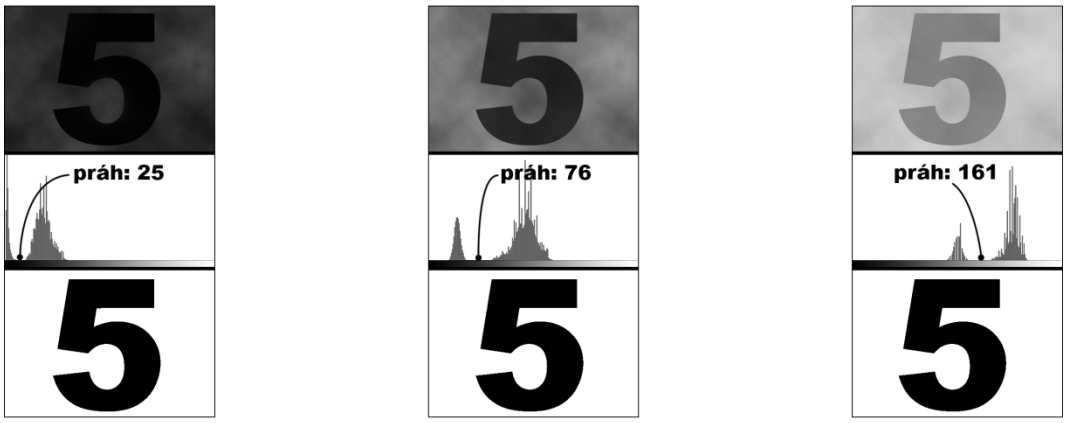


Obr. 18: Originální snímek/globální prahování/inverzní globální prahování

Typickým příkladem užití prahování může být obrázek výše (Obr. 18). Z inverzního prahování se dají snadno identifikovat jednotlivé znaky jako „bílé“ plochy o určité ploše a určitém poměru stran.

Zajímavým speciálním typem globálního prahování je tzv. **Otsu prahování**. Jde o prahování, které předpokládá, že má na svém vstupu tzv. bimodální snímek. Zjednodušeně řečeno jde o snímek, v jehož histogramu je možné pozorovat dva vrcholy. Oproti standardnímu prahování není práh vstupní konstantou funkce, ale určuje se z distribuce luminance (histogramu) vstupního obrázku.

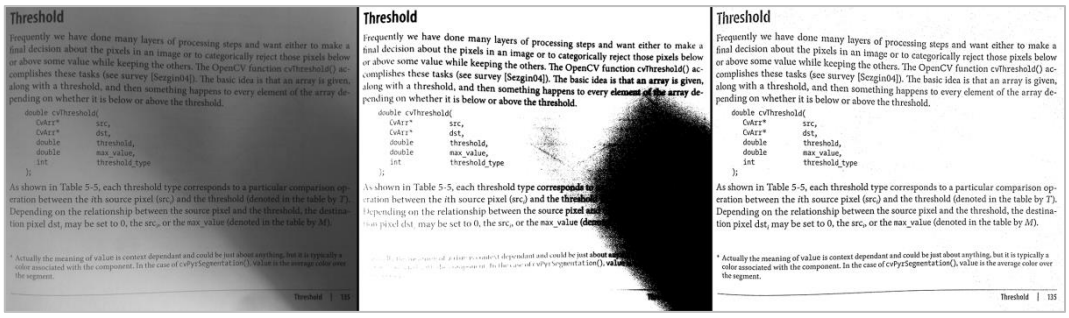
Použití Otsu prahování přináší výhody například při separaci znaků z pozadí, kdy mezi znakem a pozadím je relativně malý kontrast a celý snímek je posunut k tmavším nebo světlejším tónům (Obr. 19).



Obr. 19: Originální snímky (nahoře)/Po Otsu prahování (dole)

Adaptivní prahování

V jistých situacích nepostačuje globální prahování požadované aplikaci. Adaptivní prahování na rozdíl od globálního respektuje nerovnoměrnost rozložení světelné intenzity (Obr. 20). U globálního prahování je práh konstantou pro celý snímek. Zatímco u adaptivního prahování je práh proměnnou hodnotou prostoru závisející na nastavených parametrech a struktuře prahovaného snímku.



Obr. 20: Originální snímek/globální prahování/adaptivní prahování

Vstupními parametry funkce adaptivního prahování jsou především **velikost bloku** a odečítaná **konstanta**. Princip spočívá v určování průměrných hodnot pixelů v okolí jednotlivých bodů obrazu, které je stanovené velikostí bloku. Prostorově závislý práh se pak určí tímto váženým průměrem, od kterého se odečítá zadaná konstanta.

V OpenCV existují dvě metody adaptivního prahování (mean a gaussian). Liší se způsobem počítání průměru ve stanovených blocích (mean-všechny pixely mají stejnou váhu, gaussian-pixely jsou váhovány Gaussovou funkcí podle vzdálenosti od středového bodu).

4.3 Geometrické transformace

Jedná se operace, při kterých se v podstatě přetransformovávají souřadnice jednotlivých bodů podle specifických požadavků. S mnoha geometrickými transformacemi se také váže nějaký druh interpolace (lineární, kubická, nejbližší soused). Mezi nejzásadnější transformace patří změna rozlišení, rotace a úpravy perspektivy.

Změna rozlišení obrazu
Jde o základní geometrickou transformaci, která přibližně zachovává charakter obrazu (při zachování proporcí). Důvody ke změně rozlišení mohou být různé. Zmenšení rozlišení zřejmě nejčastěji probíhá z důvodu snížení datové velikosti a tím pádem i snížení nároků jak na paměťové, tak výpočetní části systémů.

Zvyšování rozlišení, s použitím jednoduché interpolace, nemá povětšinou příliš velký význam pro diskutované aplikace, jelikož se jím nezvyšuje kvalita obrazových dat. Existují metody tzv. *superresolution*, které fungují například na principu kombinování několika snímků. Využitím informace z několika snímků se dá docílit smysluplnější interpolace a u výsledného snímku dochází k jistému zvýšení kvality.

Na obrázku (Obr. 21) je zobrazen demonstrační případ snižování rozlišení za účelem zvýšení rychlosti výpočtu průměrných intenzit jednotlivých barevných kanálů. Tento příklad má poukazovat na skutečnost, že mnohdy je výhodnější zpracovávat snímky v nižším rozlišení.

Obr. 21: Demonstrační příklad využití změny rozlišení

Originální snímek (vlevo na Obr. 21) má rozlišení přibližně 6,3 Mpix a výpočet průměrných hodnot kanálů zabere testovacímu algoritmu přibližně 45 ms[12]. Snížení rozlišení na 0,5 Mpix zabere okolo 10 ms a výpočet průměrných hodnot kanálů ze zmenšeného snímku už jen 5 ms. Podstatné je, že požadované výsledky se liší pouze

[12] Uvedené časy jsou pro PC: jádro 3.6 GHz, RAM 8 GB. Algoritmus využívá pouze jedno jádro procesoru.

zanedbatelně. Cestou zpracování se snížením rozlišení bylo tedy v demonstračním případě docíleno úspory času 30 ms.

Rotace

Podobně jako u změny rozlišení zachovává rotace přibližně charakter obrazu. Rotace o úhel $\varphi = n \cdot 90°$, kde $n \in \mathbb{Z}$ neporušuje nijak kvalitu obrazu a není u ní potřeba interpolace. Rotace o jiný úhel musí zahrnovat nějakou interpolační techniku.

Důvody k rotaci obrazu jsou zřejmé, při snímání může být například nějaký zkoumaný objekt natočen v nežádoucím směru (Obr. 22). Pokud je správné logické natočení objektu požadavkem pro následující zpracování obrazu, musí v nějakém bodě algoritmu dojít k rotaci snímku, či jeho částí.

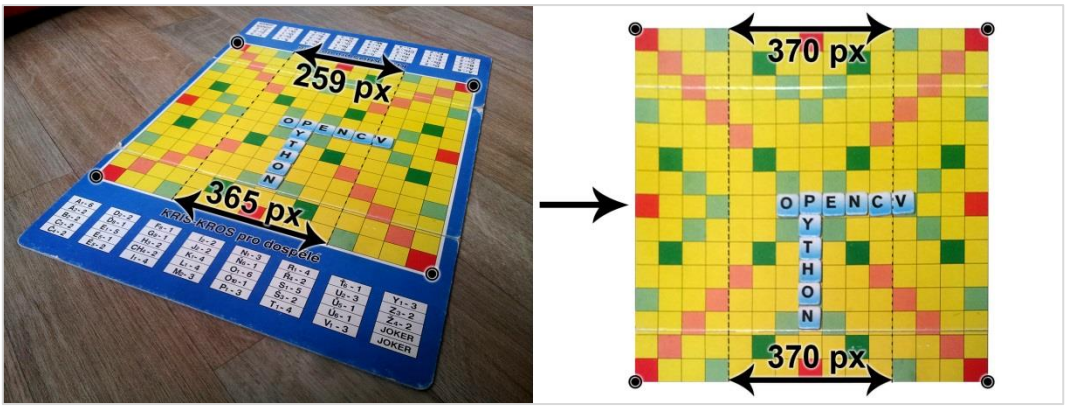
Otáčení lze provádět kolem libovolně definovaného bodu o libovolný úhel. Tyto parametry mohou být opět zadávány manuálně nebo mohou být jako obvykle výstupem nějakého předcházejícího algoritmu. Realizace rotace se dosahuje pomocí maticového násobení s využitím transformační matice, umožňující stanovit souřadnice jednotlivých pixelů ve výstupním snímku.

Obr. 22: Ilustrace rotace

Úpravy perspektivy

Na rozdíl od změny velikosti a rotace jde o geometrickou transformaci, jež může zásadněji měnit charakter snímku. Při úpravách perspektivy většinou musí docházet k interpolaci.

Použití úpravy perspektivy se zřejmě nejčastěji uplatňuje ve smyslu korekce roviny sejmutého objektu na rovinu snímače. Důvodem pro používání takové korekce bývají zejména požadavky pro zachování velikostí a proporcí objektů, jejichž části se nacházejí v různých vzdálenostech od roviny snímače snímacího zařízení (Obr. 23).



Obr. 23: Transformace perspektivy

Realizace funguje na podobném principu jako rotace obrazu. Opět se využívá transformační matice. Bez hlubšího pohledu do geometrie na pozadí těchto operací si lze podstatu transformace představit jako modifikaci souřadnic čtyř bodů (vyznačeno černými kolečky na obrázku výše). První sada souřadnic ve vstupním obrázku je transformována na sadu druhou s tím, že všechny ostatní pixely se určí příslušně podle transformace, vstupního snímku a interpolace.

Komplikovanější částí vytváření určitého algoritmu, který má sám korigovat perspektivu, může být nalezení první sady zmiňovaných souřadnic. V případě snímku (Obr. 23) by se tyto body automaticky daly hledat například zkoumáním nejkrajnějších oblastí obrázku s rozhraním tmavě modré a sytě červené nebo například jako nejkrajnější body žlutých odstínů po rozmazání snímku.

4.4 Morfologické operace

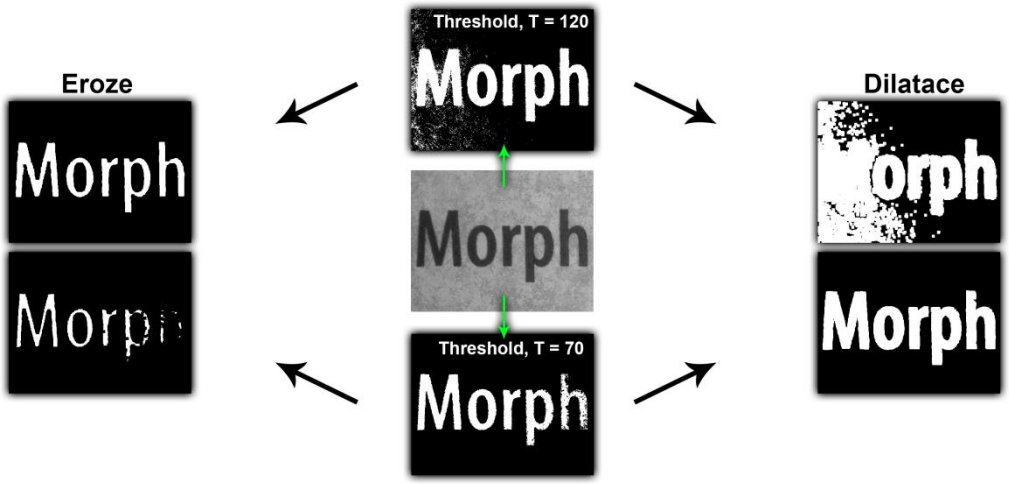
V principu se jedná jednoduché funkce, založené na „tvarech“ na pixelové úrovni [1]. Morfologické operace se zřejmě nejčastěji používají k úpravě binárního obrazu, ale mohou být využity i pro obraz intenzitní (s více než dvěma úrovněmi).

Tyto operace nacházejí uplatnění téměř v každé aplikaci počítačového vidění, podobně jako prahování, za kterým se obvykle v algoritmech nacházejí. Dalo by se říci, že tyto operace jistým způsobem obraz filtrují. Pomáhají zejména izolovat jednotlivé objekty v obraze a odstraňovat šum. Nezbytným a zásadním vstupním prvkem pro tyto operace je mimo vstupního snímku tzv. jádro, které bývá v nejobvyklejším případě malá čtvercová nebo obdélníková matice naplněná hodnotami „1“.

Eroze a dilatace

Tyto transformace jsou základními stavebními kameny pro všechny další morfologické operace. Oba procesy jsou realizovány konvolucí jádra se vstupním snímkem.

Na jádro se dá pohlížet jako na šablonu, která postupně projíždí prostorem přes celý obraz. Každý konkrétní pixel je překryt šablonou, která je větší než on sám a zkoumá se maximum (dilatace) nebo minimum (eroze) v oblasti pod šablonou. Nalezená hodnota je výsledkem operace pro daný pixel. Tento proces má zajímavý vliv na výsledný vzhled transformovaného obrazu. Dilatace představuje nárůst světlých oblastí a eroze tmavých. Funkce je demonstrována níže na Obr. 24.



Obr. 24: Eroze a dilatace

Otevírání a zavírání

Jde o morfologické operace, které jsou kombinací eroze a dilatace. Vyžívají stejného principu i stejných jader. Mají vysokou praktickou použitelnost.

Otevírání (*opening*) není nic jiného než eroze, po které následuje dilatace. Opening je vhodný například pro odstraňování malých částeček šumu, při přibližném zachování podstatného elementu obrazu. Zavírání (*closing*) je opačnou operací a jde tedy o dilataci, za kterou následuje eroze. Closing je vhodný pro „zabělení“ malých černých děr v obraze.

Obr. 25: Opening(otevírání) a Closing(zavírání)

Top Hat, Black Hat, Morfologický gradient

Jde o komplexnější morfologické operace, plnící podobné funkce jako předchozí zmíněné. Tyto operace pracují různými způsoby s kombinací vstupního snímku, jeho otevíráním, zavírání, dilatací a erozí. Všechny funkce mohou přijímat stejný typ jádra. Funkce Top Hat například izoluje lokální světlé špičky (Obr. 26), čehož se využívá v implementovaných algoritmech této práce.

Top Hat, Black Hat – rozdíl mezi vstupním snímkem a jeho otevíráním, zavíráním
Morfologický gradient – rozdíl mezi dilatací a erozí vstupního snímku

Obr. 26: Top Hat, Black Hat, Morfologický gradient

4.5 OpenCV contours

Tato sekce se zabývá speciální entitou v knihovně OpenCV. Přestože jde o konkrétní název právě pro tuto knihovnou, je možné se setkat s podobnými entitami i mimo ní. Například v programovém prostředí MATLAB je možné najít podobné objekty pod označením *blob*[13].

Contours (obrysy), by se jednoduše daly popsat jako křivky spojující body podél určitých hranic. Pojmenování „contours“ v sobě zahrnuje velké množství funkcí a operací spojených nejčastěji s detekcí, analýzou a rozpoznáváním objektů v obraze. Budou zde probrány pouze nejpodstatnější poznatky o nich.

Find & Draw contours

Contours se dají popsat jako křivky podél určitých hranic. Hranicí se myslí rozhraní bílé a černé v binárním obraze (Obr. 27). Funkce pro vyhledávání těchto křivek má na vstupu tedy binární obraz a další parametry související s hledáním obrysů.

[13] Blob analysis. *MathWorks* [online]. [cit. 2015-05-15]. Dostupné z: http://www.mathworks.com/help/vision/ref/blobanalysis.html

Obr. 27: Ilustrace Find & Draw contours

Na obrázku je naznačeno získávání pouze tzv. externích obrysů, je tedy možné si všimnout, že písmena „o“ a „p“ jsou obehnána hranicí jen okolo celého písmena, nikoliv ve svých vnitřních částech.

Není příliš podstatné se dopodrobna zabývat způsobem nacházení obrysů. Podstatné je, že umožňují určitým způsobem ohraničit separované objekty a uchovat souřadnice v proměnné. Z pohledu analýzy a rozpoznávání objektů je zajímavější a důležitější se zabývat vlastnostmi, které obrysy (tedy často určité objekty v obraze) mají.

Vlastnosti contours

Primární vlastností obrysů jsou tzv. obrazové momenty. Zjednodušeně řečeno, je moment jakási charakteristika, která se dá spočítat sumací přes všechny pixely daného obrysu. Hodnoty pixelů se různě váhují a tím se získávají momenty různých řádů. Výpočtem momentů je možné určovat mnoho geometrických vlastností nalezených obrysů či mezi sebou obrysy porovnávat. Níže jsou uvedeny vybrané základní vlastnosti obrysů.

Plocha obrysů

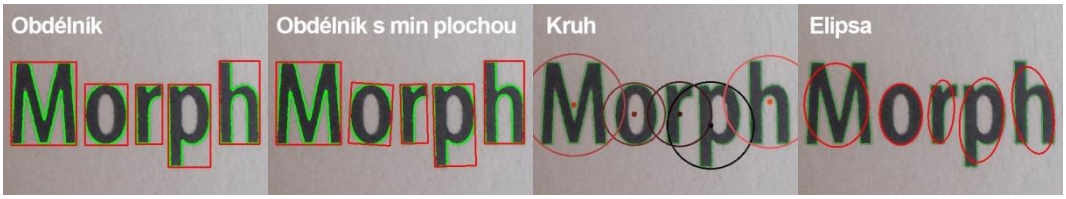
Plocha je základní geometrickou vlastností 2D objektu v obraze a jde o prostý počet pixelů uvnitř zkoumaného obrysu (Obr. 28).

Obr. 28: Určení plochy obrysů

Ohraničující objekty

Často je žádané zjišťovat proporce různých objektů v obraze. Pro jejich zjištění je možné užít možnosti ohraničení nalezených obrysů určitým útvarem. Mezi nejčastěji užívané ohraničující útvary patří obdélník, obdélník minimální plochy, kruh a elipsa (Obr. 29).

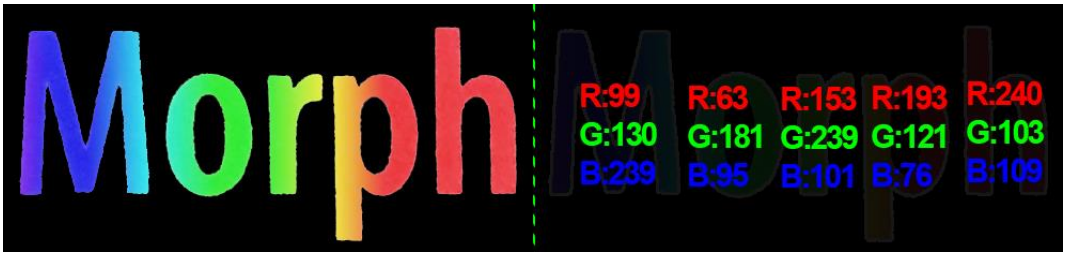
Dalším zkoumáním těchto objektů se dají určovat parametry jako například: poměr stran, natočení, průměr, umístění v obraze apod.



Obr. 29: Ohraničení obrysů různými útvary

Průměrné hodnoty jednotlivých kanálů

Tento příklad je pouze ilustrační ukázkou jedné z mnohých vlastností, které se dají za pomoci obrysů získávat a dále zpracovávat (Obr. 30).



Obr. 30: Průměrné hodnoty jednotlivých kanálů

4.6 Jednoduchá klasifikace obrazových dat

Do teorie zpracování obrazu také částečně patří jeho klasifikace a rozpoznávání. S pojmem rozpoznávání se pojí obor tzv. strojového učení[14], jehož úkolem je na základě nějakého modelu přeměňovat neznámá vstupní data na užitečnou informaci.

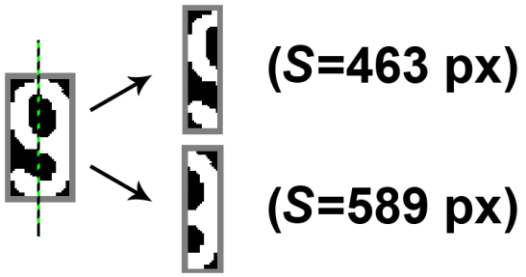
Ve strojovém učení se data dají obecně rozdělit na data trénovací a testovací. Trénovací data souvisí s učením klasifikátoru a jsou to informace o prvcích a třídách, do kterých prvky spadají. Data testovací jsou potom neznámým vstupem pro klasifikátor, který má za úkol testovaný vstup přiřadit do určité třídy.

Nejjednodušším možným případem klasifikátoru dat je metoda k-Nearest neighbours (KNN). Klasifikátor KNN je typu „učení s učitelem“, což znamená, že pro své naučení potřebuje označená vstupní data [11]. V této sekci je pro demonstraci uveden ilustrační případ využití klasifikátoru KNN na obrazových datech. Tento demonstrační případ je zde uveden zejména z důvodu využití podobného klasifikátoru v praktické části práce, kde by nebyl prostor na jeho ozřejmění.

[14] Strojové učení je velmi širokým tématem a tato práce se ho dotýká jen velmi okrajově. Pro pokročilé metody interpretace obrazu mohu doporučit především knihu [11].

Ilustrační případ algoritmu KNN

Vstupními daty demonstračního algoritmu využívajícího klasifikátor KNN budou obrázky znaků „6“ a „9“ v rozlišení 30x60 px. Pro možnost ilustrace případu ve 2D, budou z každého obrázku znaku extrahovány dvě vlastnosti, které o něm něco vypovídají. Jako tyto dvě základní vlastnosti byly vybrány plochy znaku v pravé a levé části snímku (Obr. 31).



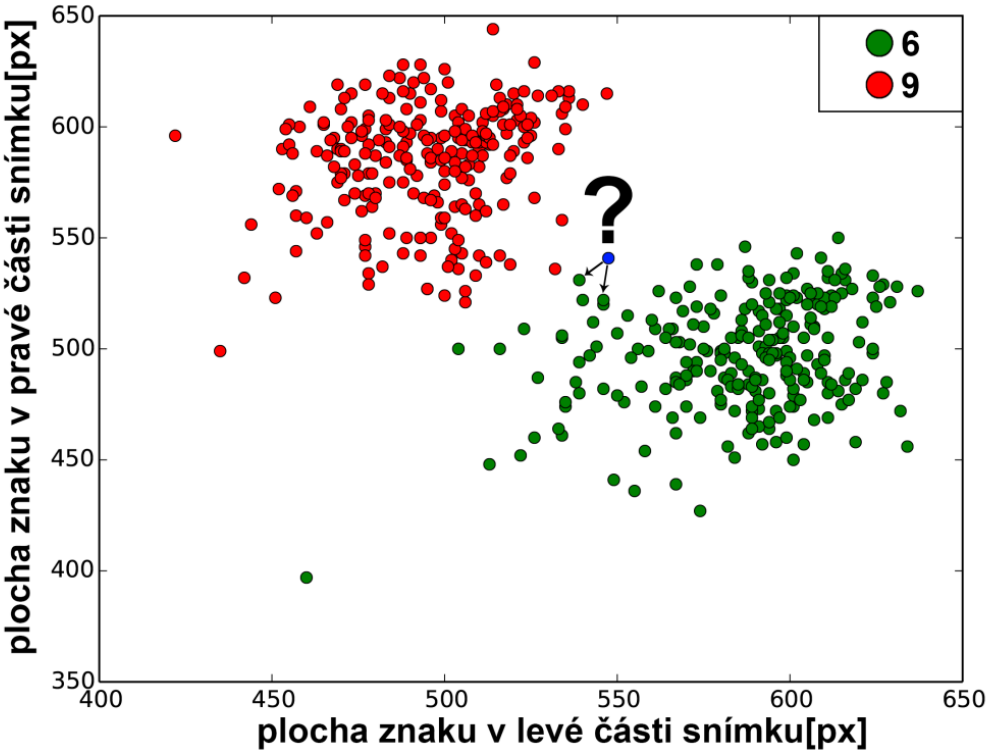
Obr. 31: Extrakce parametrů ze snímku jednoho znaku

Výpočtem zmíněných vlastností pro určitou sadu snímků znaků „6“ a „9“ (Obr. 32), lze vypozorovat jednoznačné rozdíly ve výsledcích. Dvojice získávaných parametrů se dá také v obecném případě nazvat vektorem příznaků (*feature vector*), což jsou data popisující jeden vzor, čímž je v tomto případě snímek znaku „6“, nebo „9“. Po zanesení zjištěných vektorů příznaků do grafu (Obr. 33) je vidět jisté shlukování vzorů stejné třídy. Systém se natrénuje poskytnutím vektorů příznaků a jejich označení.

Obr. 32: Část sady snímků znaků „6“ a „9“

Snímek č.		1	2	3	…	n-2	n-1	n
Označení		„9“	„9“	„9“	…	„6“	„6“	„6“
Vektory příznaků	*S* vlevo [px]	463	506	510	…	616	627	573
	S vpravo [px]	589	595	582	…	531	529	494

Tab. 1: Data pro ilustrační případ KNN



Obr. 33: Vynesení hodnot z **Tab. 1** do grafu

Jakmile je systém natrénovaný, může probíhat klasifikace, která je v případě KNN prostá a spočívá v měření vzdáleností mezi vektory příznaků, na které je systém natrénován a vektorem příznaků neznámého testovaného objektu (modrý bod na Obr. 33). Pro měření vzdáleností existují různé metriky (hammingova, euklidovská, atd.). Podle zadaného parametru k se hledá různý počet nejbližších vektorů příznaků. Majoritní označení nalezených příznaků je výsledkem klasifikace (pro $k = 2$ bude modrý bod klasifikován jako znak „6“).

5. Přístupy k řešení problematiky

Jak již bylo uvedeno, automatická detekce a rozpoznávání RZ je transformací obrazových dat na text RZ, která má velké uplatnění v mnoha praktických aplikacích. Transformace obrazových dat se obvykle dociluje v několika krocích za pomoci pokročilých metod zpracování obrazu (viz kapitola č. 4).

K řešení problematiky této transformace existuje nesčetné množství přístupů, jejichž komplexní shrnutí je možné nalézt především v článku [5]. Většina prací, na které článek odkazuje, se snaží docílit co největší robustnosti a uvádí algoritmy, jejichž spolehlivost je invariantní vůči mnoha variacím prostředí i vzhledů RZ (viz kapitola č. 3).

V nejjednodušším zúžení lze problematiku rozdělit na dvě části. První částí je detekce RZ a druhou je její rozpoznání. Obě části jsou velmi důležité a jedna bez druhé tvoří systémy identifikace RZ nefunkčními. V této kapitole budou velmi stručně nastíněny vybrané přístupy k oběma částem. Nemá příliš cenu zabíhat do podrobností a opakovat tak informace, které už jsou výborně shrnuty v [5]. Podstatné je, že všechny přístupy se většinou zaměřují na pochopení rysů obrazových dat v oblasti RZ. Se zaměřením na zjištěné rysy jsou následně vytvářeny systémy, jež se ke kýženému výsledku dopracovávají různými kroky.

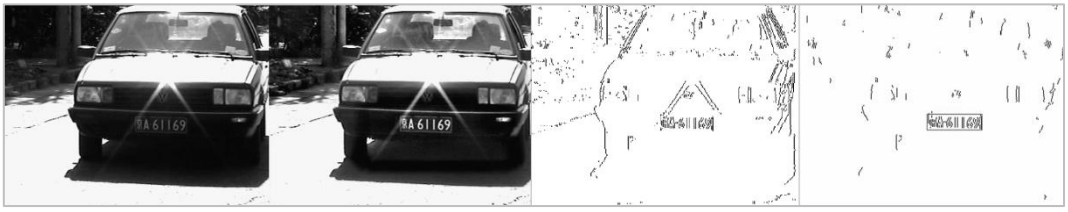
Možné přístupy k detekci RZ
Metody detekce lze dle článku [5] obecně rozdělovat do několika kategorií, a to podle typů vlastností, na jejichž základě se snaží RZ z obrazu extrahovat. Nejrozšířenějším způsobem jsou zřejmě metody, které detekují RZ díky hranovým informacím.

Například v článku [10] autoři uvádějí, že RZ je objektem, ve kterém se obvykle vyskytuje velké množství hran (prudké změny luminance na malém prostoru). Navíc je většina těchto hran vertikálních a jsou relativně rovnoměrně rozmístěny v oblasti celé RZ. Na základě těchto poznatků konstruují systém, který detekuje RZ jako oblasti statisticky odpovídající datům, na které byl systém natrénován. V této zmiňované práci využívají k detekci především klasifikátor spojený tzv. *Harr-like features*, což značí název pro extrakci určitých jednoduchých parametrů z obrazu. *Harr-like feartures* jsou známé především ze systémů detekce obličejů.

Na skutečnost velkého množství vertikálních hran v oblasti RZ se rovněž zaměřuje článek [1]. V prvním kroku jsou podobným způsobem jako v [10] získány vertikální

hrany. Dále jsou oblasti s velkou hustou hran stanoveny jako kandidáti na RZ. Speciálním obrazovým filtrem jsou potom vyloučeni ti kandidáti, kteří neodpovídají předem stanovenému modelu. Nakonec je finální hledaná oblast RZ získána pomocí morfologických operací aplikovaných na zbylé kandidáty.

Efektivní metodu detekce navrhuje článek [15]. Opět pracuje s vertikálními hranami, podobně jako dvě předchozí metody. Na rozdíl od nich zde ale zprvu dochází k jakémusi vylepšení snímku. Ilustrace zásadních kroků systému navrhovaného v [15] je naznačena níže (Obr. 34).

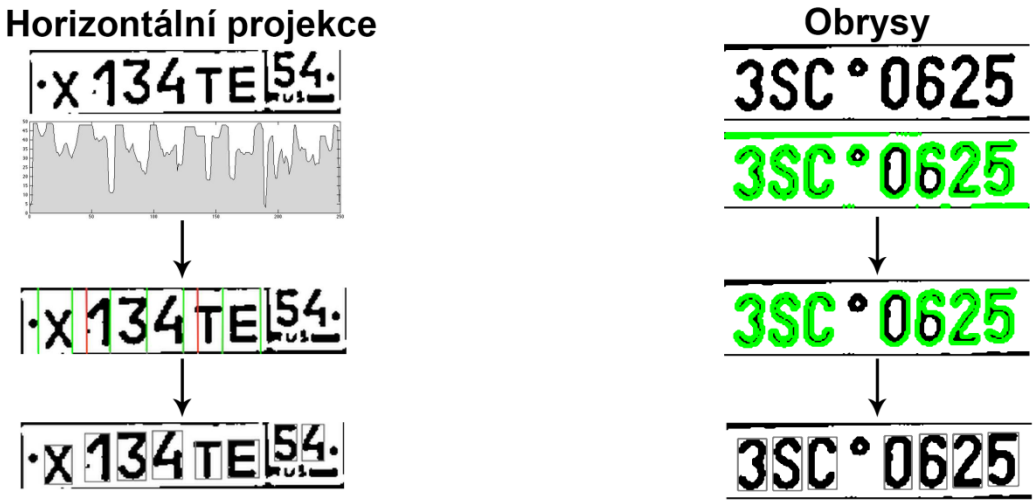


Obr. 34: Zásadní kroky metody pro detekci RZ navrhované v [15]

Přístupy k rozpoznání RZ

Rozpoznávání RZ probíhá obvykle po její detekci, jejímž výstupem bývá separovaný výřez RZ. Jak udává článek [5], tato separovaná RZ může podléhat různým nedokonalostem. Značka může být například nerovnoměrně osvícena nebo natočena v nežádoucím směru, což bez předzpracování komplikuje separaci jednotlivých znaků. Takové předzpracování může obecně zahrnovat různé geometrické transformace, odstraňování šumu, ekvalizaci histogramu či vylepšení kontrastu mezi znaky a pozadím.

Po předzpracování už následuje separace jednotlivých znaků a jejich rozpoznávání. Pro separaci znaků opět existuje mnoho přístupů. Většina z nich začíná s využitím různých typů prahování (viz kapitola 4.2) tzn. převedení snímku RZ na binární obraz. Dalším krokem jsou už různé metody zjišťující polohy znaků. Nejčastěji je zřejmě možné setkat se s metodami projekčními (*segmentation using projection profiles*) [16, 17] nebo už méně často s metodami založenými na extrakci znaků za pomocí obrysů (viz kapitola 4.5). Obě metody jsou ilustračně naznačeny na Obr. 35.



Obr. 35: Vybrané způsoby separace znaků (obrázek vlevo [18])

Posledním z praktikovaných kroků přímo souvisejícím z rozpoznáním RZ je klasifikace jednotlivých znaků neboli OCR (*optical character recognition*). Klasifikace spadá do oboru strojového učení (viz kapitola 4.6). Podle [5] lze rozpoznávání znaků RZ rozdělit do dvou skupin. První skupinou je rozpoznávání založené na používání surových dat, tzv. *template matching* (viz kapitola 6.3). Druhou skupinou je rozpoznávání na základě extrakce různých parametrů (viz kapitola 4.6, demonstrační případ). Algoritmy z druhé skupiny jsou náročnější na implementaci, protože vyžadují transformaci dat na různé rysy, ale obvykle jsou rychlejší a dobře navržené přináší lepší výsledky klasifikace. Pro OCR je možné využívat mnoha různých klasifikátorů, nejznámějším jsou zřejmě neuronové sítě.

6. Praktická část

Obsahem praktické části práce je především návrh a implementace komplexního kamerového systému pro detekci a identifikaci RZ vozidel. Vytvořený systém transformuje vstupní obrazová data na textový kód odpovídající RZ v obraze bez možnosti dalšího hlubšího analyzování výstupních dat. Vstupními daty navržených algoritmů mohou být fotografie, videozáznamy nebo i aktuální záběry ze stacionárních průmyslových kamer. Implementace je provedena v programovacím jazyku Python spolu s několika externími knihovnami, zejména s OpenCV.

Mým hlavním záměrem bylo vytvořit systém, který bude schopen číst RZ v reálném čase ze statického záběru HD kamery. Taková schopnost s sebou přináší zásadní požadavek na rychlost algoritmu. Budu-li za pojem „reálný čas“ považovat udržení snímkovací frekvence 25 sn. /s, nesmí kompletní doba zpracování jednoho snímku přesáhnout 40 ms. Zpočátku této práce jsem se tedy zaměřoval na výběr operací, které umožňují dostatečně rychlé zpracování obrazu. Finální algoritmus dokáže zpracovávat jeden snímek v HD rozlišení do 30 ms[15] a splňuje tedy podmínku pro zpracovávání v reálném čase. Při výběru oblasti ROI (viz kapitola 4.1) o rozlišení zhruba 4x menším než je HD, jsem docílil dokonce rychlostí pod 10 ms na snímek.

Implementovaný algoritmus pro zpracování snímků v reálném čase má však nedostatek z pohledu variability vstupních obrazových dat, zejména z pohledu velikosti RZ a rozlišení obrazu. Pro rozšíření schopností algoritmu jsem vytvořil program s grafickým rozhraním (Obr. 36) umožňující rozpoznávat RZ ze statických snímků téměř libovolného rozlišení a s téměř libovolně velkou RZ. Rozšíření variability funguje na principu několikanásobných změn rozlišení vstupního snímku, kvůli nimž dochází k značnému nárůstu času zpracování. Doba zpracování jednoho HD snímku se v programu s grafickým rozhraním pohybuje okolo 180 ms, snímek o rozlišení 3 Mpix (2352x1568) se už zpracovává přibližně 600 ms.

Na tomto místě musím rovněž upozornit, že implementovaný systém nedokáže bohužel rozpoznávat RZ s textem ve dvou řádcích, RZ většiny cizích států a RZ s neobvyklým počtem znaků. Všechny implementované algoritmy jsou silně zaměřeny na nejčastěji se vyskytující RZ a SPZ v ČR, tedy značky, jež mají 7 tmavých znaků na světlém pozadí a jsou obdélníkového tvaru s poměrem stran přibližně 5:1.

[15] Všechny časy zpracování implementovaného algoritmu jsou udávány pro PC: jádro 3.6 GHz, RAM 8 GB. Implementované algoritmy využívají k výpočtům pouze jedno jádro procesoru.

Obr. 36: Implementovaný algoritmus s grafickým rozhraním

6.1 Návrh systému

Výsledný vytvořený systém vznikl jakousi kompilací myšlenek systémů existujících uvedených především v [5, 14, 18, 19] a myšlenek vlastních, získaných experimentováním. Před tím, než začnu popisovat jednotlivé implementované algoritmy, musím (v souvislosti s kap. 3.3) uvést několik informací o tom, jak jsem získával obrazová data k testování a trénování systému.

Většinu obrazových dat v rámci této práce jsem natočil, nebo nafotil sám. Výjimku tvoří sekvence snímků RZ stažených z internetu ze stránek PlatesMania [20]. Pro získávání obrazových dat jsem používal především tři snímací zařízení. Základní informace o nich jsou uvedeny v Tab. 2. Dále budou zařízení nazývána už jenom podle názvů svých výrobců.

Název zařízení	Zařazení	Snímač	Užívaný objektiv	Fotografie	Video
Basler acA1300-30gc[16]	průmyslová kamera	CCD 18 mm^2	Pentax 12,5 – 75 mm	nevyužito	1294x964 30fps
Canon EOS 500D[17]	digitální zrcadlovka	CMOS 232 mm^2	Sigma 70-300 mm	4752 x3168 (15 Mpix)	1280x720 30fps
Samsung Galaxy S II	smartphone	CMOS 16 mm^2	originální integrovaný	3264 x 2448 (8 Mpix)	nevyužito

Tab. 2: Využívaná snímací zařízení

Pro svůj primární cíl (systém pracující v reálném čase) jsem využíval kameru Basler, ze které je možné přímo „živě“ zachytávat obrazová data pomocí funkcí knihovny OpenCV. Kameru jsem měl umístěnou ve 2. patře bytového domu a snímal projíždějící vozidla zezadu přibližně na 90m vzdálenost. Průměrnou rychlost vozidel ve snímaném úseku odhaduji na 30 km/h, což je rychlost nepřinášející příliš zvýšené nároky na expoziční dobu (viz kapitola 3.3). Typické používané geometrické uspořádání kamery a záběru je naznačeno na Obr. 37.

Obr. 37: Záběr implementovaného systému pracujícího v reálném čase

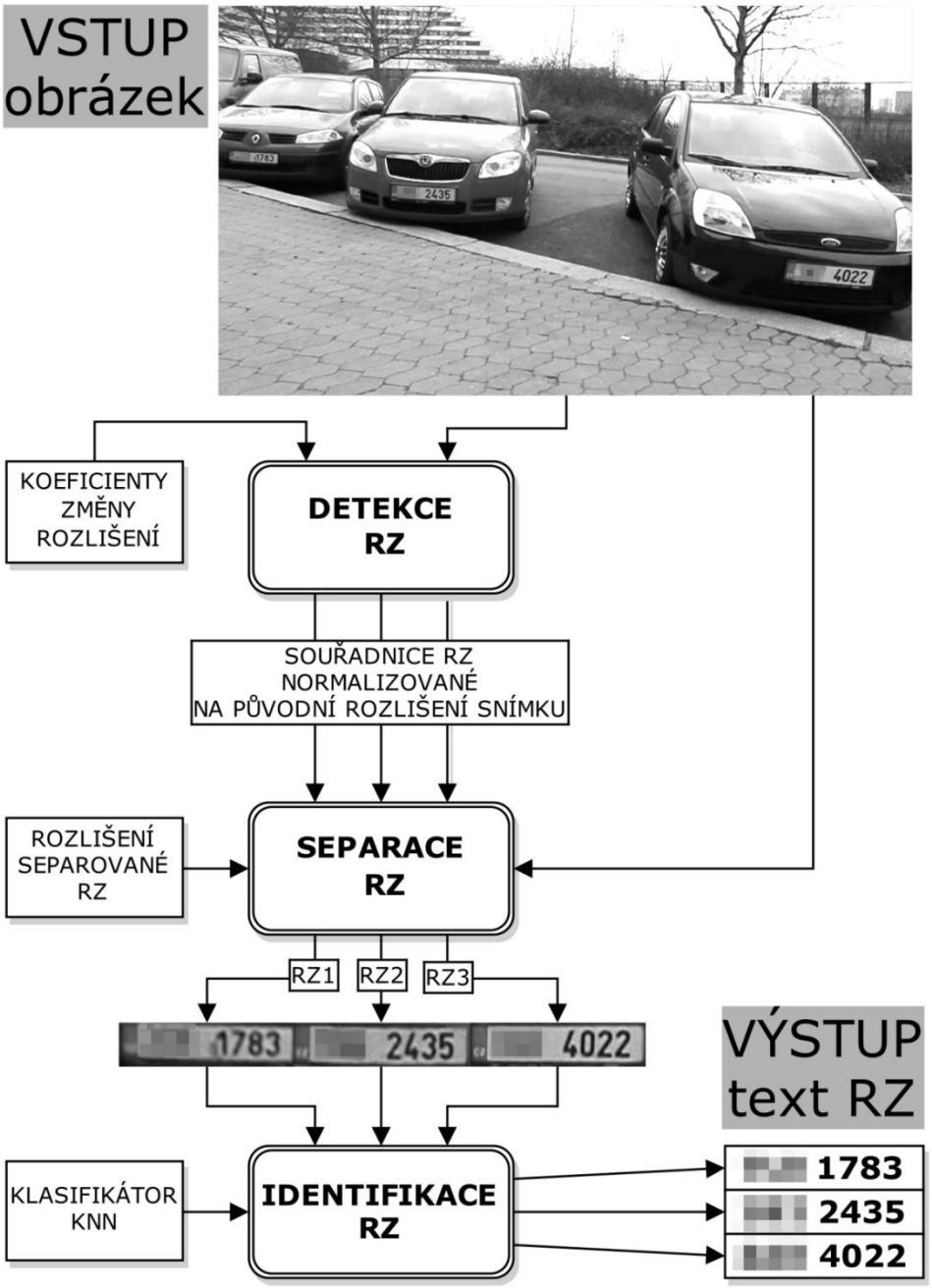
Stejný záběr jsem v různé denní doby rovněž zaznamenával fotoaparátem Canon pro získání co největšího počtu testovacích a trénovacích dat v různých světelných podmínkách. Světelné podmínky na tomto záběru jsou během dne velmi rozličné. Záběr

[16] AcA1300-30gc. *BASLER* [online]. 2015 [cit. 2015-05-17]. Dostupné z: http://www.baslerweb.com/en/products/area-scan-cameras/ace/aca1300-30gc

[17] Pro Canon 500D jsem používal vylepšující firmware: *Magic Lantern* [online]. [cit. 2015-05-17]. Dostupné z: http://www.magiclantern.fm/

je situovaný na jih, což přináší nepříjemné protisvětlo v poledních hodinách. Během hodin odpoledních zase velmi často dochází k zastínění oblasti snímaní okolními budovami. S těmito nepříjemnosti jsem musel počítat při nastavování expozice.

Souběžně se získáváním obrazových dat jsem se snažil postupně vyvíjet a vylepšovat implementované algoritmy, jejichž základní blokové schéma je naznačeno na Obr. 38. Jednotlivé bloky korespondují s konkrétními vytvořenými funkcemi a budou probrány v následujících sekcích.



Obr. 38: Základní blokové schéma implementovaných algoritmů

V implementovaných algoritmech je transformace obrazových dat docíleno v několika úrovních. Ve vstupním snímku se detekují kandidáti na RZ a získají se jejich souřadnice (*detekce_RZ*). Na základě těchto souřadnic jsou značky ze vstupního snímku separovány a normalizovány na určité rozlišení (*separace_RZ*). Posledním krokem je separace jednotlivých znaků a jejich rozpoznání na základě klasifikátoru KNN (*identifikace_RZ*)

6.2 Detekce registrační značky

Ve svém algoritmu detekce RZ jsem se zaměřil především na fakt, že české RZ jsou podle zákona tvořeny tmavými znaky na světlém pozadí. Pro jisté zvýraznění těchto oblastí jsem využil morfologickou operaci Top Hat (viz kapitola 4.4) aplikovanou téměř na samém začátku algoritmu na černobílý snímek (inspirace především z [18, 19]).

Za morfologickou operací Top Hat postupně následuje Otsu prahování (viz kapitola 4.2) a dále morfologické operace closing a opening. Tomuto pořadí rovněž odpovídají snímky na Obr. 39. Implementace je naznačena níže.

```
# První kroky při detekci RZ
img = cv2.cvtColor(imgBGR, cv2.COLOR_BGR2GRAY)
toph = cv2.morphologyEx(img, cv2.MORPH_TOPHAT, np.ones((3, 25), np.uint8))
ret, thrsh = cv2.threshold(toph, 0, 255, cv2.THRESH_BINARY+cv2.THRESH_OTSU)
clos = cv2.morphologyEx(thrsh, cv2.MORPH_CLOSE, np.ones((3, 10), np.uint8))
open = cv2.morphologyEx(clos, cv2.MORPH_OPEN, np.ones((5, 20), np.uint8))
# Jádra použitá pro morfologické operace jsou obdélníky s poměrem stran >3
```

Obr. 39: Nejpodstatnější kroky při detekci RZ v implementovaném algoritmu

V aktuálním kroku je k dispozici snímek opening (úplně vpravo na Obr. 39). V tomto snímku se naleznou OpenCV contours (viz kapitola 4.5) a postupně se v cyklu testuje jejich plocha a poměr stran. Obrysy nevyhovující podmínkám na určitý formát a velikost jsou vyloučeny z dalšího zpracování.

```
# Nalezení obrysů ve snímku opening
cntdata = cv2.findContours(open,cv2.RETR_EXTERNAL,cv2.CHAIN_APPROX_SIMPLE)
contours = cntdata[0]

# Zde dochází v cyklu k postupnému vylučovaní kandidátů na RZ
for i in range (0, len(contours)):
      # další, dále diskutované zpracování
```

Zbytek obrysů splňující podmínky se již dá považovat za kandidáty na RZ. Za pomoci parametrů obepínajícího obdélníku (viz kapitola 4.5) se z nich stanou oblasti ROI (viz kapitola 4.1). Pro demonstrační snímky se oblastmi zájmu staly opravdu pouze RZ, což znamená, že v binárních snímcích opening neexistovaly další prvky splňující podmínky na plochu a poměr stran obrysů. To však není úplným standardem a v tomto kroku se může mezi kandidáty objevovat hodně oblastí snímku, které RZ neobsahují (tzv. falešná detekce).

Obr. 40: Kandidáti na RZ, nalezení v demonstračních snímcích (oblasti ROI)

V dalších krocích, které provádím ve funkci *detekce_RZ* již spoléhám na existenci znaků uvnitř získaných oblastí zájmu (Obr. 40). Je provedeno adaptivní prahování a „jemný“ opening všech kandidátů RZ nalezených v předchozích krocích. Tyto operace mají na výstupu binární snímek s dobře viditelnými znaky (Obr. 41). Opět jsou nalezeny OpenCV contours, tentokrát s cílem najít kandidáty na znaky. Pokud je nalezen požadovaný počet kandidátů na znak, předají se souřadnice kandidátů k dalšímu zpracování.

Obr. 41: Dobře viditelné znaky v binárním snímku

Následující zpracování je ze všech částí algoritmů nejvíce konkretizované pro české RZ a SPZ určitého typu. Informace o kandidátech na znak RZ jsou předány funkci pro zpracování souřadnic znaků (*zprac_souradnice*). Tato funkce má za cíl zpřesnit detekování lokace značky a zjistit její natočení. Primárními událostmi odehrávajícími se uvnitř těla funkce jsou procesy, které na základě jednoduché statistiky vylučují prvky neodpovídající znakům RZ. Ze zbývajících znaků jsou dopočteny souřadnice (rohy) obdélníku, ohraničujícího text RZ (souřadnice RZ, Obr. 42).

Obr. 42: Souřadnice RZ vykreslené do vstupních snímků

Celý doposud popsaný algoritmus je relativně hodně vázaný na rozměr RZ, což vychází především z používání mnoha konstantních parametrů (jádra morfologických operací, podmínky při vylučování kandidátu, apod.). Experimentálně jsem otestoval, že algoritmus ve svém základním režimu umožnuje detekovat RZ pouze v rozměru přibližně 10-30 pixelů na výšku.

Zprvu jsem toto zjištění příliš neřešil, protože jsem se zaměřoval především na systém pracující se stacionární kamerou, u kterého šlo dobře zaručit přibližně konstantní velikosti RZ při nastavování záběru.

Nicméně později jsem přišel s jednoduchým zlepšením, nezasahujícím do již implementované struktury. Princip spočívá ve změnách rozlišení vstupního snímku (viz kapitola 4.3) podle zadaných koeficientů. Celá funkce probíraná v této sekci se potom několikanásobně opakuje s různě velkými snímky na vstupu. Získané výstupní souřadnice RZ se po každé iteraci normalizují na původní velikost obrázku.

Celá funkce pro detekci různě velkých RZ se volá následovně:

```
KOEF = [1, 0.7, 0.5, 0.3] # Definice koeficientů velikosti vstupního snímku
# Získání souřadnic RZ v obraze
Souradnice_RZ = multiple_size_detekce_RZ(IMAGE, KOEF)
# Souřadnice jsou tvořeny rohy obdélníků ohraničujících nalezené RZ
```

Funkce *multiple_size_detekce_RZ* umožňuje za cenu prodloužení výpočetního času rozpoznávat RZ rozdílných velikostí ve snímcích s různým rozlišením. Toho využívám v programu s grafickým rozhraním. Algoritmu pracujícímu v reálném čase se předává pouze jeden koeficient rovný 1, a tak se RZ hledají jenom v nemodifikovaném vstupním snímku. Detekce RZ je v mém algoritmu časově nejnáročnějším procesem, pro HD snímek se časy zpracování této části algoritmu pohybují okolo 15-25 ms.

6.3 Identifikace registrační značky

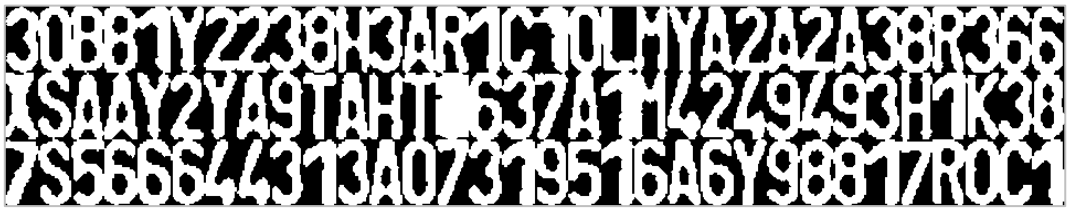
Po získání normalizovaných souřadnic značky dojde k separaci výřezu RZ ze vstupního obrázku. K tomuto účelu slouží funkce *separace_RZ*, ve které je využita operace pro perspektivní transformaci (viz kapitola 4.3), ale plní zde v podstatě účel výřezu, otočení a normalizace velikosti nalezených RZ (Obr. 43).

Obr. 43: Normalizované výřezy RZ

Výše uvedené snímky jsou pak výstupem funkce *separace_RZ* a vstupem pro funkci *identifikace_RZ*. Jde o snímky s konstantní velikostí 200x50 px získané výřezem ze vstupního snímku.

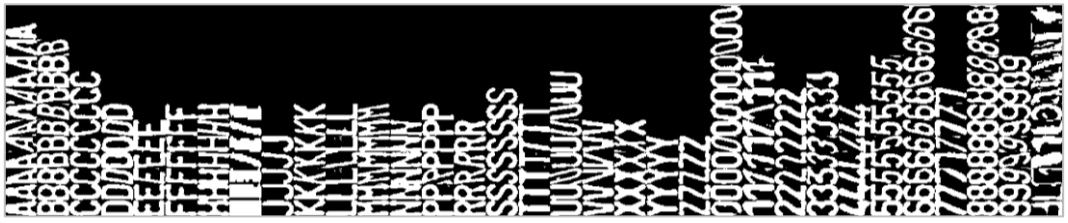
Podobným způsobem jako ve funkci *detekce_RZ* jsou z normalizovaných výřezů RZ získány souřadnice znaků. Rozdíl procesů spočívá v selektivitě při výběru kandidátů na znak, v této části algoritmu je podstatné získat všechny souřadnice znaků.

Na základě získaných souřadnic jsou jednotlivé znaky separovány a normalizovány na konstantní velikost (Obr. 44). Realizováním této operace na větší sadu snímků jsem získal velké množství dat, která jsem využil k natrénování klasifikátoru KNN (viz kapitola 4.6).



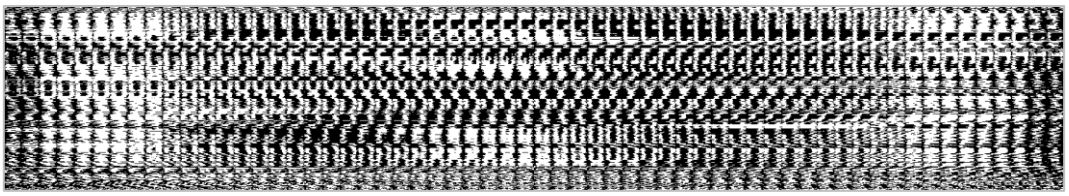
Obr. 44: Různé separované znaky RZ

Zprvu jsem musel roztřídit nalezené separované znaky do jednotlivých tříd. To jsem dělal částečně manuálně a částečně s pomocí implementovaného algoritmu fungujícího už na principu klasifikace (*add2trainimg_algoritmus*). Výsledkem třídění bylo posbírání trénovacích dat. Celá následná klasifikace probíhá pouze podle snímku na Obr. 45.



Obr. 45: Snímek trénovacích dat (otočen o 90° doleva)

Principem klasifikace jsem se značně inspiroval v návodu pro OCR ručně psaných čísel [21]. Každý jednotlivý znak v trénovacím snímku má rozměr 15x30 px. Pro účel využití klasifikátoru KNN jsou znaky rozprostřeny do jednoho řádku, do rozměru 450x1 px a jsou seřazeny pod sebe. Transformovaný snímek trénovacích dat je zobrazen níže na Obr 46.

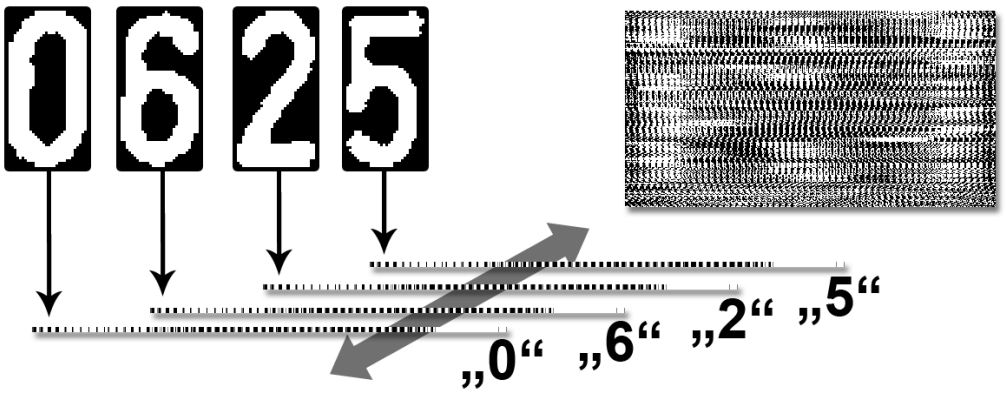


Obr. 46: Transformovaný snímek trénovacích dat

Každý řádek je popsán označením znaku ze kterého vzešel a představuje jeho vektor příznaků o rozměru (1, 450). Tento snímek je společně s označeními řádků zadán k naučení klasifikátoru.

```
knn  = cv2.KNearest() # Nadefinování klasifikátoru k-nejbližších sousedů
knn.train(train_data,train_labels) # Natrénovaní klasifikátoru
# train_data – není nic jiného než Obr. 46
# train_labels – označení popisující jaký řádek odpovídá jakému znaku
```

Nyní se vrátím zpět k demonstračním testovaným snímkům, ze kterých jsou získány jednotlivé znaky. Znaky jsou podobně jako při tvorbě trénovacích dat rozprostřeny do jednoho řádku a následně jsou porovnávány s daty v trénovacím snímku. Podoba se měří euklidovskou vzdáleností vektorů příznaků (viz kapitola 4.6). Označení nejpodobnějšího řádku je potom vráceno jako výsledek klasifikace. Jde o případ klasifikace KNN, kdy $k=1$, tuto metodu lze rovněž nazývat NN (*nearest neighbour*). Ilustrace výše popisovaného je na Obr. 47. Seřazením výsledků klasifikace podle horizontální souřadnice daného znaku v obraze získávám výsledky přečtených RZ.



Obr. 47: Ilustrace způsobu klasifikace v implementovaných algoritmech

Rychlost implementované funkce *identifikace_RZ* je silně závislá na množství trénovacích dat. Finální využívaný snímek trénovacích dat obsahuje celkem 311 znaků (průměrně zhruba 9 variací na symbol) a doba zpracování znaků jedné RZ se pohybuje okolo 4 ms. Uměle jsem vytvořil trénovací snímky odpovídající vyššímu počtu znaků. Největší testovaný snímek byl ekvivalentem přibližně 5000 znaků (průměrně 150 variací na znak) a doba zpracování znaků jedné RZ se pohybovala okolo 30 ms. Pozorovaný nárůst času zpracování v této oblasti měl přibližně lineární charakter.

Zde končí popis algoritmu pro rozpoznávání RZ z jednotlivých snímků. Mým hlavním cílem ale bylo rozpoznávat RZ z videa, případně z živého záběru stacionární průmyslové kamery. Video není ničím jiným, než sekvencí časově po sobě jdoucích snímků, takže princip již popisovaných funkcí zůstává stejný i pro algoritmy pro video. Dále se však se zpracováním videa ještě pojí jedna významná implementovaná funkce (*casovy_vyskyt*).

V této funkci se vyžívá faktu, že jedna a ta samá RZ se po dobu průjezdu vozidla dá rozpoznat několikrát a výsledky lze statisticky vyhodnotit. S tím však přichází trochu nebezpečí ve formě diskretizace výsledků (1 RZ může být čtena několikrát, několik RZ může být čteno 1x). Základní princip funkce je naznačen v Tab. 3. Přestože podle tabulky byla RZ přečtena špatně na 4 z 5 snímků, celkový výsledek je díky využití více výsledků správný.

Snímek	Rozpoznána RZ	Diskretizace výsledků z několika snímků
n	0B2 1722	3B2 1722
n + 1	3BZ 1722	
n + 2	382 1T22	
n + 3	3B2 1722	
n + 4	3B2 1721	

Tab. 3: Ilustrace principu funkce *casovy_vyskyt*

6.4 Spolehlivost implementovaného systému

V průběhu tvorby této bakalářské práce jsem v součtu získal přibližně 100 GB obrazových dat obsahově souvisejících s RZ. Z přibližně 500 rozličných snímků RZ jsem separoval znaky a využil je k natrénování klasifikátoru NN. Žádný ze snímků využitý k tréninku nebyl použit na testování spolehlivosti.

Spolehlivost vyjadřuje míru úspěšnosti detekce a rozpoznání RZ a jde o podstatnou informaci, vystihující funkčnost systému z praktického hlediska. V případě implementovaného systému se jeho spolehlivost bohužel silně váže na typ testovacích dat. Jak už jsem uvedl v úvodu této kapitoly, můj přístup k problematice umožňuje číst pouze RZ některých typů, což jeho obecnou spolehlivost značně sráží dolů. Při testování spolehlivosti implementovaného algoritmu jsem ignoroval neschopnost systému číst značky, na které nebyl koncipován.

Vytvořil jsem mnoho sekvencí na testování spolehlivosti implementovaného systému, zde ale zmíním pouze tři největší a nejdůležitější datové sady, pro které platí spolehlivost uvedená v Tab. 4. Jde o jednu videosekvenci (BaslerVideo) a dvě sady fotografií s různými vlastnostmi (SamsungFoto, PlatesmaniaFoto).

Sekvence	**BaslerVideo**		**SamsungFoto**		**PlatesmaniaFoto**	
	počet RZ [-]; z celkem [%]					
Celkem	928	100	254	100	935	100
detekováno	921	99,2	252	99,2	802	85,8
úspěšně rozpoznáno	**881**	**94,9**	**236**	**92,9**	**616**	**65,9**
nerozpoznáno	40	4,3	16	6,3	186	19,9
nedetekováno	7	0,8	2	0,8	133	14,2

Tab. 4: Úspešnost implementovaného algoritmu v testovacích sekvencích

Jednotlivé sekvence mají diametrálně rozdílný charakter a spolehlivost algoritmů s ním značně souvisí. Pro komplexnost doplním několik informací o každé ze sekvencí.

BaslerVideo
Jak už název napovídá, jedná se o videosekvenci pořízenou kamerou Basler. Pro získání této sekvence byla kamera umístěna stejným způsobem jako na Obr. 37. Výslednou spolehlivost lze tedy brát i jako spolehlivost pro systém pracující v reálném čase, a proto této sekvenci přikládám největší význam. Zároveň je tedy uspokojivé, že ze všech

tří testovaných sekvencí je implementovaný algoritmus právě v tomto případě nejspolehlivější (94, 9 % RZ a SPZ úspěšně rozpoznáno).

Záznam je dlouhý přibližně 1 h a 15 min a byl pořízen za pomoci jednoduchého detektoru pohybu (viz kapitola 4.1) během jednoho dne přibližně od 10:00 do 20:00. Naznačení světelných podmínek během dne je ilustrováno na několika snímcích níže.

Obr. 48: Světelné podmínky v sekvenci BaslerVideo

Ve skutečnosti jsem do tabulky úspěšnosti zahrnul průjezdy vozidel pouze z doby přibližně od 10:00 do 19:00. Po 19:00 začínala prudce klesat hladina osvětlení a jakékoliv další pokusy o vyrovnání expozice selhávaly. Kvůli snižujícímu se množství světla docházelo k postupnému degradování spolehlivosti systému (Tab. 5). Západ slunce byl pro daný den v 19:55 a po 20:00 jsem již nebyl schopen detekovat jedinou RZ.

Sekvence	**BaslerVideo**					
Čas snímaní	**10:00-19:00**		**19:00-19:40**		**19:40-20:00**	
	počet RZ [-]; z celkem [%]					
Celkem	928	100	31	100	15	100
detekováno	921	99,2	24	77,5	4	26,7
úspěšně rozpoznáno	**881**	**94,9**	**22**	**71,0**	**3**	**20,0**
nerozpoznáno	40	4,3	2	6,5	1	6,7
nedetekováno	7	0,8	7	22,5	11	73,3

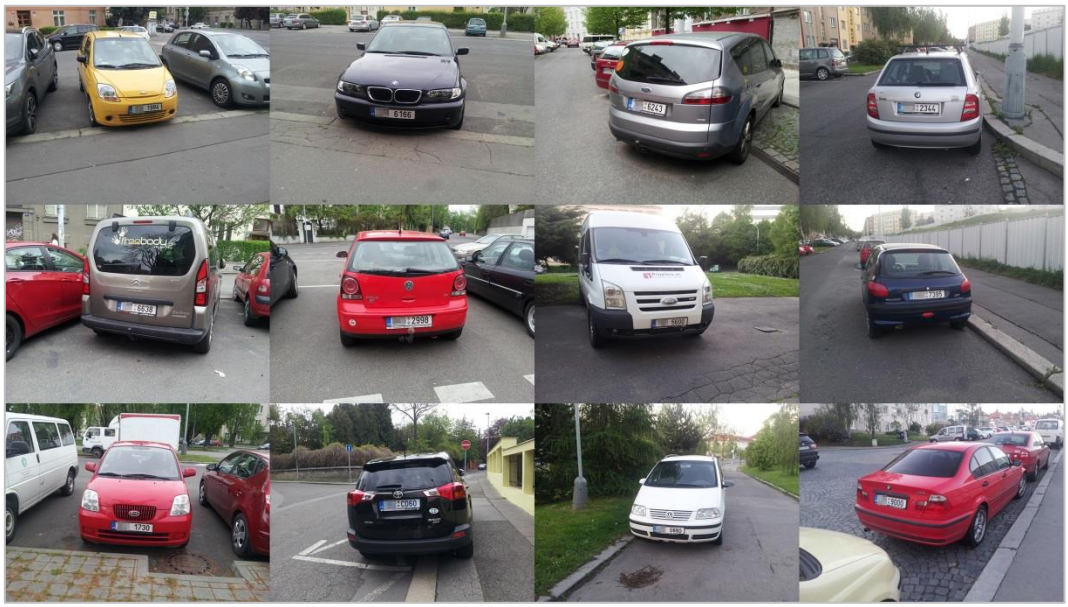
Tab. 5: Úspešnost implementovaného algoritmu během dne v sekvenci BaslerVideo

Do sekvence jsem nezapočítával RZ motorek (celkem 21), RZ cizích států (celkem 12) a RZ s jiným poměrem stran (celkem 8). V součtu jde přibližně o 4 % všech projíždějících motorových vozidel.

Ze 7 nedetekovaných RZ standardního typu v době mezi 10:00 a 19:00 byly 2 velmi špinavé, 1 měla dva znaky spojené a další 2 byly zastíněny jiným elementem vozidla. Co se týče důvodu nerozpoznání 40 detekovaných RZ, tak největší důvod přikládám nedokonalosti funkce *casovy_vyskyt*. Celkem 17 vozidel ze 40 nebylo rozpoznáno, protože selhala diskretizace výsledků z několika předchozích snímků. Chybová situace obvykle nastávala, když dvě nebo tři vozidla jela v těsném závěsu za sebou a byla považována za jedno.

SamsungFoto

Jde o sekvenci celkem 254 fotografií RZ pořízených z minimálního úhlu na telefon Samsung. Fotografie jsem testoval při sníženém rozlišení 1280x960 programem s grafickým rozhraním. Ukázky fotografií z této sekvence jsou níže na Obr 49.

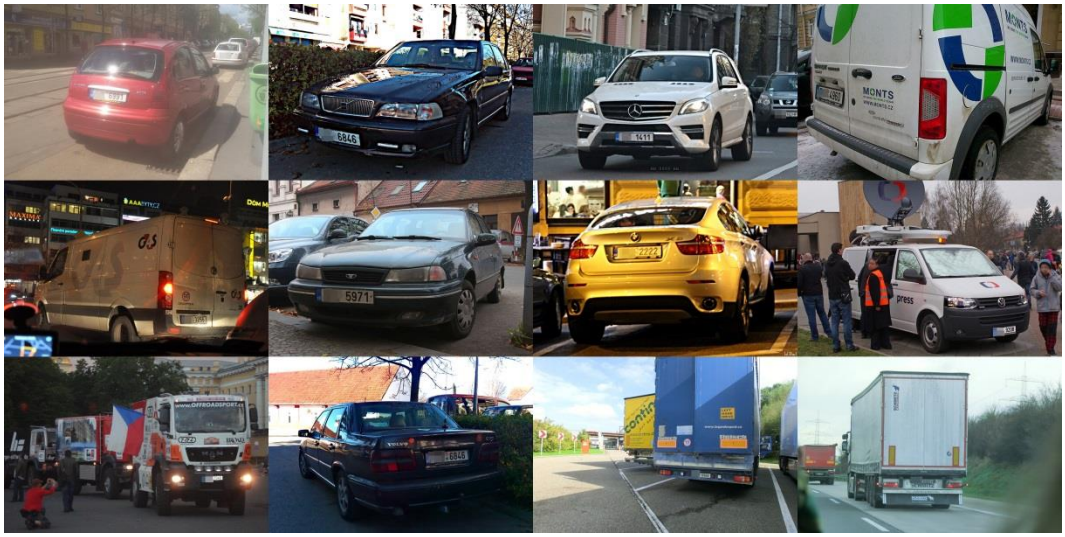


Obr. 49: Typické fotografie z testovací sekvence SamsungFoto

Účelem nafocení sekvence bylo získat kvalitní fotografie, u kterých se dá předpokládat, že by s nimi systém neměl mít problém. Na základě neúspěšných detekcí a rozpoznání v této sekvenci jsem vypozoroval určité neduhy, kterými můj algoritmus trpí (popisuji dále).

PlatesmaniaFoto

Ze všech používaných testovacích dat, jde o nejkomplexnější sbírku. Fotografie jsou staženy z webu [20], kde je sdílí fanoušci RZ. Snímky jsou rozličného rozlišení a byly získány rozdílnými zařízeními za různých podmínek, z různých úhlů i vzdáleností. Některé snímky z této sekvence jsou zobrazeny na Obr. 50.



Obr. 50: Vybrané fotografie z testovací sekvence PlatesmaniaFoto [20]

Nedokonalosti implementovaného algoritmu

Všeobecným testováním implementovaného algoritmu jsem byl schopen vypozorovat jeho určité nedokonalosti. Z pohledu rozpoznávání RZ z videa jde především o již zmiňovanou nedokonalost funkce *casovy_vyskyt*. Pro identifikaci RZ z jednotlivých obrázků je nedostatků více.

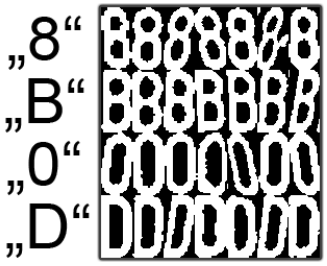
První chybu vidím už při detekci v příliš velkém vázání se na RZ konkrétních velikostí se sedmi znaky. Kvůli tomu může algoritmus vyřazovat správné kandidáty na RZ hned zpočátku a snižovat tak rapidně svou spolehlivost.

Dalším relativně často se objevujícím problémem (především při testování sekvence PlatesmaniaFoto) byla neschopnost algoritmu detekovat RZ vyfocené z větších úhlů (Obr. 12). To je dáno především zpracováváním souřadnic kandidátů na znaky ve funkci *zprac_souradnice*. Procesy funkce sice umožnují přesněji detekovat RZ, které byly vyfoceny z malého úhlu, ale vyřazuje RZ vyfocené z úhlu většího. Maximální horizontální vychýlení, při kterém je algoritmus ještě schopen správně detekovat RZ, odhaduji z experimentů přibližně na 50-60° od kolmice značky.

Při rozpoznávání již detekovaných RZ jsem se potýkal především s dvěma chybami. První chybou bylo považování hran RZ za různé znaky (především: „I“, „L“, „T“, „J“) (Obr. 51). Myslím, že s touto vadou by se šlo lehce vypořádat při zohlednění možných formátů textů RZ.

Obr. 51: Typická chyba při rozpoznávání RZ

Druhá chyba byla poněkud komplikovanější a šlo o zaměňování podobných znaků při klasifikaci. Možné řešení problému podobných znaků v diskutovaných systémech je například naznačeno v [22]. Nejčastěji jsem se při testování setkával se záměnou znaků „B“ „8“, „0“ a „D“. Jejich podobnost zároveň naznačuje snímek níže (Obr. 52).

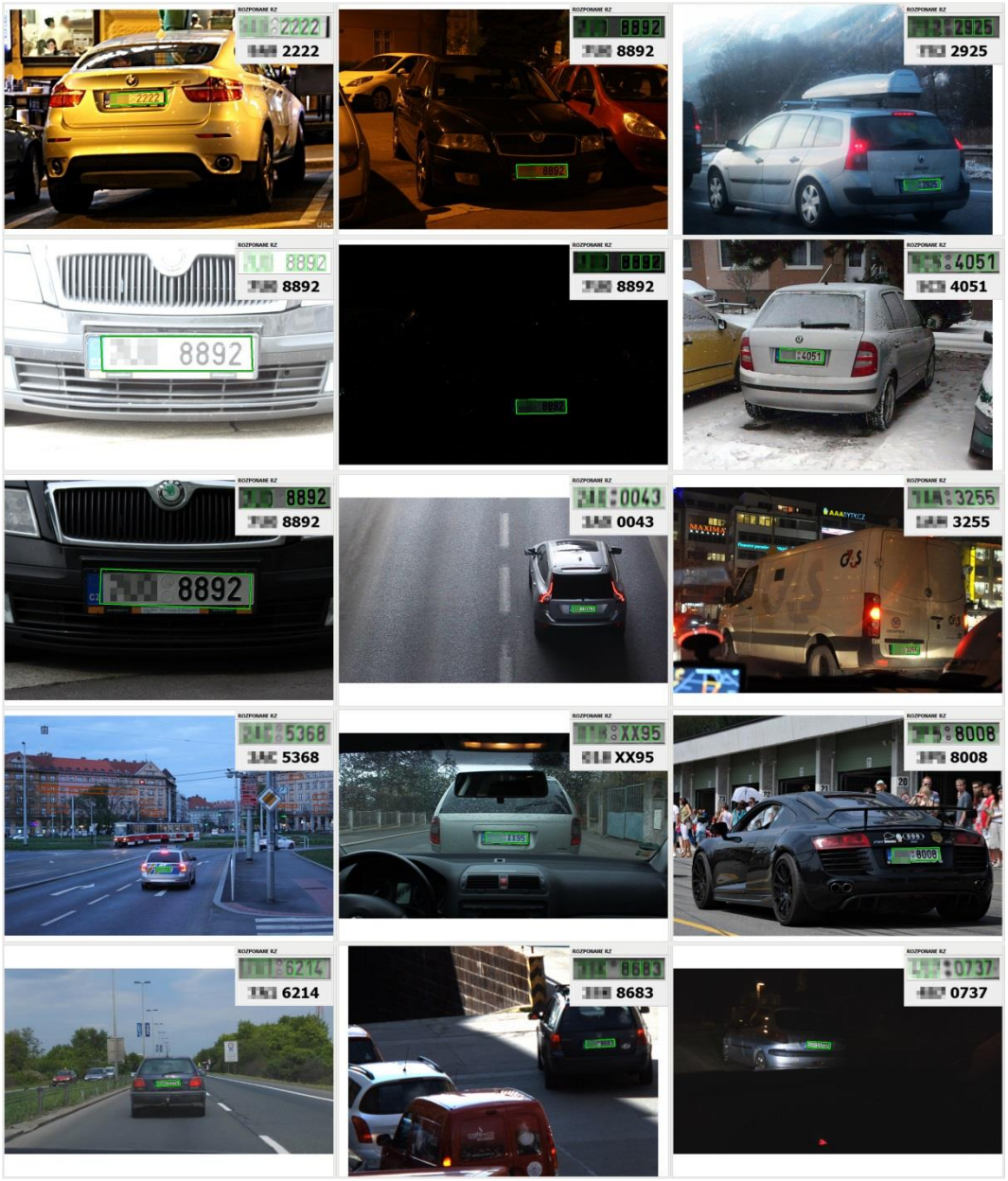


Obr. 52: Příklad snadno zaměnitelných znaků

6.5 Diskuze vlivu světelných podmínek

Součástí cíle této práce je diskutovat vliv světelných podmínek na účinnost použitých algoritmů. Musím podotknout, že tato otázka se kromě implementovaného algoritmu také výrazně týká kapitoly 3.3, ve které jsem uváděl mnoho aspektů ovlivňujících kvalitu snímku při jeho získávání a tím i účinnost algoritmu.

Implementovaný algoritmus využívá na svém počátku morfologickou operaci Top Hat, která umožňuje z obrázku izolovat lokální maxima. Díky tomu je možné detekovat značky i v případě výraznějších posunů expozice v oblasti RZ (Obr. 14). Spolu s následným využitím adaptivního a Otsu prahování při separaci jednotlivých znaků, se algoritmus stává relativně robustním vůči světelným poměrům ve snímku. Ukázky úspěšné identifikace RZ v různých světelných podmínkách jsou zobrazeny na Obr. 53.



Obr. 53: Úspěšná identifikace v různých světelných podmínkách (některé snímky [20])

Ačkoliv tvrdím, že algoritmus je robustní vůči světelným poměrům ve snímku, výsledky testování spolehlivosti pro celodenní sekvenci BaslerVideo toto tvrzení nepodporují (Tab. 5). Zkoumaná spolehlivost poklesla k nule po 20:00, kdy úroveň světla nedovolovala nastavit expozici podle minimálních požadavků na výstupní snímky (viz kapitola 3.3). Celkový implementovaný systém jako takový tedy není robustní vůči velkým změnám světelných podmínek a nedovoluje algoritmu spolehlivě fungovat za tmy. Robustnost by se mi nejspíš podařilo navýšit s využitím infračerveného zdroje světla a zařízením pracujícím v téže pásmu. Ostatně jak už bylo zmíněno, většina existujících systémů infračerveného světla využívá.

7. Závěr

Ve své bakalářské práci jsem se zabýval problémem rozpoznávání RZ v obraze. Postupným studiem problematiky jsem si uvědomil její rozsáhlost a komplexnost. Ačkoliv se mi RZ zdála zprvu velmi dobře definovaným objektem, pochopil jsem, že se její vzhled v obraze může v závislosti na mnoha skutečnostech diametrálně lišit.

Na základě studie existujících systémů jsem navrhl a implementoval skupinu algoritmů rozpoznávajících české RZ z mnoha typů vstupních obrazových dat. Implementaci jsem uskutečnil v programovacím jazyku Python spolu s knihovnou OpenCV.

Transformace obrazových dat na text RZ dociluji v několika úrovních pomocí pokročilých metod zpracování obrazu a jednoduchého klasifikátoru NN. Za hlavní výstup praktické části práce považuji realizaci systému, schopného číst RZ v reálném čase ze záběru stacionární průmyslové HD kamery. Za denních podmínek tento systém prokázal úspěšnost čtení RZ vyšší než 90 %, což může být v případě některých aplikací dostačující hodnota.

Systém má však také mnoho nedokonalostí a dokáže pracovat pouze v určitých vymezených podmínkách. Jednou z jeho hlavních vad je neschopnost číst RZ méně obvyklejších typů, jako jsou třeba značky s textem ve dvou řádcích či značky s nestandardním počtem znaků. Další výrazné problémy pak nastávají například při pokusech o čtení RZ nasnímaných z příliš velkého úhlu či při snímaní za malého množství světla. Algoritmy nejsou navíc postaveny jako finální aplikace, které by umožňovaly různé uživatelské nastavení nebo například možnost analýzy výstupních dat.

Věřím, že k napravení mnohých nedokonalostí algoritmů by mi pomohlo hlubší studium oboru strojového učení a komplexnějších klasifikátoru než je KNN. Z pohledu hardwarové části systému by zajisté pomohlo využití snímacích zařízení pracujících v blízkém infračerveném pásmu spolu s ekvivalentním zdrojem světla.

Přestože je přínos této práce především osobního charakteru myslím si, že tento dokument může vhodně posloužit jako úvodní materiál studentům, seznamujícím se s danou problematikou. Zároveň dávám k dispozici relativně velké množství testovacích obrazových dat s českými RZ, které mohou být využity pro testování jiných systémů. Praktické využití implementovaných algoritmů si s malými modifikacemi dokáži představit především v oblasti dopravních průzkumů.

Seznam obrázků a tabulek

Obrázky, u nichž není uveden zdroj, jsou dílem autora práce. Mohou se však na nich vyskytovat části, které byly získány z internetu vyhledáváním s filtrem: „Povoleno opětovné použití s úpravami“. Zdroje také nejsou uváděny u snímků, které jsou výstupem implementovaných algoritmů.

Obr. 1: Zjednodušený pohled na vstup a výstup probíraných systémů
Obr. 2: Typické umístění kamer systému od společnosti CAMEA s. r. o.
Obr. 3: Ilustrace systému pro měření dojezdové doby
Obr. 4: Ilustrace systému pro měření úsekové rychlosti
Obr. 5: Policejní vůz vybavený systémem čtení RZ
Obr. 6: Rozmazávání citlivých údajů v Google Street View
Obr. 7: Nejčastěji se vyskytující typ RZ v ČR, typ 101 [12]
Obr. 8: Povolené znaky na RZ a SPZ v ČR [12]
Obr. 9: Rozmanitost RZ (poměr velikostí nedodržen)
Obr. 10: Ukázka retroreflexe RZ
Obr. 11: Obvyklé defekty RZ
Obr. 12: Pořízení snímku značky z různých úhlů
Obr. 13: Expoziční doba a vliv na pohybové rozmazání
Obr. 14: Snímek RZ pořízen s různou expozicí
Obr. 15: Reprezentace černobílého digitálního obrazu (výřez hrany RZ)
Obr. 16: Ilustrace systému detekce pohybu
Obr. 17: Ilustrační případ vytvoření ROI
Obr. 18: Originální snímek/globální prahování/inverzní globální prahování
Obr. 19: Originální snímky (nahoře)/Po Otsu prahování (dole)
Obr. 20: Originální snímek/globální prahování/adaptivní prahování
Obr. 21: Demonstrační příklad využití změny rozlišení
Obr. 22: Ilustrace rotace
Obr. 23: Transformace perspektivy
Obr. 24: Eroze a dilatace
Obr. 25: Opening(otevírání) a Closing(zavírání)
Obr. 26: Top Hat, Black Hat, Morfologický gradient
Obr. 27: Ilustrace Find & Draw contours
Obr. 28: Určení plochy obrysů
Obr. 29: Ohraničení obrysů různými útvary
Obr. 30: Průměrné hodnoty jednotlivých kanálů
Obr. 31: Extrakce parametrů ze snímku jednoho znaku

Obr. 32: Část sady snímků znaků „6“ a „9“
Obr. 33: Vynesení hodnot z Tab. 1 do grafu
Obr. 34: Zásadní kroky metody pro detekci RZ navrhované v [15]
Obr. 35: Vybrané způsoby separace znaků (obrázek vlevo [18])
Obr. 36: Implementovaný algoritmus s grafickým rozhraním
Obr. 37: Záběr implementovaného systému pracujícího v reálném čase
Obr. 38: Základní blokové schéma implementovaných algoritmů
Obr. 39: Nejpodstatnější kroky při detekci RZ v implementovaném algoritmu
Obr. 40: Kandidáti na RZ, nalezení v demonstračních snímcích (oblasti ROI)
Obr. 41: Dobře viditelné znaky v binárním snímku
Obr. 42: Souřadnice RZ vykreslené do vstupních snímků
Obr. 43: Normalizované výřezy RZ
Obr. 44: Různé separované znaky RZ
Obr. 45: Snímek trénovacích dat (otočen o 90° doleva)
Obr. 46: Transformovaný snímek trénovacích dat
Obr. 47: Ilustrace způsobu klasifikace v implementovaných algoritmech
Obr. 48: Světelné podmínky v sekvenci BaslerVideo
Obr. 49: Typické fotografie sekvence SamsungFoto
Obr. 50: Vybrané fotografie z testovací sekvence PlatesmaniaFoto [20]
Obr. 51: Typická chyba při rozpoznávání RZ
Obr. 52: Příklad snadno zaměnitelných znaků
Obr. 53: Úspěšná identifikace v různých světelných podmínkách

Tab. 1: Data pro ilustrační případ KNN
Tab. 2: Využívaná snímací zařízení
Tab. 3: Ilustrace principu funkce *casovy_vyskyt*
Tab. 4: Úspešnost implementovaného algoritmu v testovacích sekvencích
Tab. 5: Úspešnost implementovaného algoritmu během dne v sekvenci BaslerVideo

Seznam použité literatury a zdrojů

[1] ABOLGHASEMI, Vahid a Alireza AHMADYFARD. 2007. A Fast Algorithm for License Plate Detection. (EDS)., Guoping Qiu .. [et al.]). *Advances in visual information systems 9th international conference, VISUAL 2007, Shanghai, China, June 28-29, 2007: revised selected papers*. Berlin: Springer, s. 468-477. ISBN 9783540764144.

[2] DONOSER, Michael, Clemens ARTH a Horst BISCHOF. 2007. Detecting, Tracking and Recognizing License Plates. (EDS)., Yasushi Yagi .. [et al.]). *Computer vision ACCV 2007: Asian Conference on Computer Vision, Tokyo, Japan, November 18-22, 2007 ; proceedings*. Berlin: Springer, s. 447-456. ISBN 9783540763901.

[3] 3M Motor Vehicle Systems & Services: Public Safety Solutions. 2015. *3M* [online]. [cit. 2015-05-16]. Dostupné z: http://solutions.3m.com/wps/portal/3M/en_US/NA_Motor_Vehicle_Services_Systems/Motor_Vehicle_Industry_Solutions/parking-tolling-dmv-solutions/law-enforcement-solutions/

[4] BRADSKI, Gary a Adrian KAEHLER. 2008. *Learning OpenCV: Computer Vision with the OpenCV Library*. Sebastopol: O'Reilly, xvii, 555 s. ISBN 978-0-596-51613-0.

[5] DU, Shan, Mahmoud IBRAHIM, Mohamed SHEHATA a Wael BADAWY. 2013. Automatic License Plate Recognition (ALPR): A State-of-the-Art Review. In: *IEEE Transactions on Circuits and Systems for Video Technology*. Calgary, Canada: IEEE, roč. 23, s. 311-325. DOI: 10.1109/TCSVT.2012.2203741. ISSN 1051-8215. Dostupné také z: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6213519

[6] PŘIBYL, Ondřej. 2015. *Měření a zpracování dat (MDS): Neintrusivní dopravní detektory* [Přednáška]. [cit. 2015-04-10].

[7] LIU, Guixiang, Zhongyou MA, Zhongguo DU a Can WEN. 2011. The Calculation Method of Road Travel Time Based on License Plate Recognition Technology. HONGHUA TAN, Mark Zhou (eds. *Advances in information technology and education International Conference, CSE 2011, Qingdao, China, July 9-10, 2011, Proceedings*. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg, s. 385-389. DOI: 10.1007/978-3-642-22418-8_54. ISBN 10.1007/978-3-642-22418-8_54. Dostupné také z: http://link.springer.com/10.1007/978-3-642-22418-8_54

[8] Měření úsekové rychlosti: Camea UnicamVELOCITY [katalogový list]. 2008. [cit. 2015-05-16]. Dostupné z: http://camea.cz/underwood/download/files/UnicamVelocity_Web.pdf

[9] FROME, Andrea, German CHEUNG, Ahmad ABDULKADER, Marco ZENNARO, Bo WU, Alessandro BISSACCO, Hartwig ADAM, Hartmut NEVEN, Luc VINCENT, et al. 2009. Large-scale privacy protection in Google Street View. *2009 IEEE 12th International Conference on Computer Vision*. DOI: 10.1037/e502102013-034.

[10] ZHANG, Huaifeng, Wenjing JIA, Xiangjian HE a Qiang WU. 2006. Real-Time License Plate Detection Under Various Conditions. (EDS)., Jianhua Ma .. [et al.]). *Ubiquitous intelligence and computing third international conference, UIC 2006, Wuhan, China, September 3-6, 2006: proceedings*. Berlin: Springer, s. 192-199. DOI: 10.1007/11833529_20. ISBN 9783540380924. Dostupné také z: http://link.springer.com/10.1007/11833529_20

[11] BEHNKE, Sven. 2003. *Hierarchical neural networks for image interpretation*. New York: Springer, xii, 224 p. ISBN 35-404-0722-7.

[12] ČESKÁ REPUBLIKA. Nové registrační značky. 2006. In: *243/2001 Sb*. Dostupné také z: http://www.mdcr.cz/cs/Silnicni_doprava/Dovoz_registrace_a_schvalovani_vozidel/Registrace+vozidel/Nove+registracni+znacky/nove_registracni_znacky.htm

[13] GURNEY, R., M. RHEAD, S. RAMALINGAM, W.E. MARTIN a N. COHEN. 2013. The Effect of Retro-reflectivity and Reflectance of UK Number Plates on ANPR Performance. *5th International Conference on Imaging for Crime Detection and Prevention (ICDP 2013)*. DOI: 10.1049/ic.2013.0264.

[14] *OpenCV dokumentace* [online]. 2011. [cit. 2015-05-16]. Dostupné z: http://docs.opencv.org/

[15] ZHENG, Danian, Yannan ZHAO a Jiaxin WANG. 2005. An efficient method of license plate location. *Pattern Recognition Letters*. Beijing, China: Elsevier, roč. 26, s. 2431-2438. DOI: 10.1016/j.patrec.2005.04.014. ISBN 10.1016/j.patrec.2005.04.014. ISSN 01678655. Dostupné také z: http://linkinghub.elsevier.com/retrieve/pii/S0167865505001406

[16] SANYUAN, Zhang, Zhang MINGLI a Ye XIUZI. 2004. Car plate character extraction under complicated environment. *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No.04CH37583)*. The Hague: IEEE, s. 4722-4726. DOI: 10.1109/ICSMC.2004.1401277. ISBN 0-7803-8567-5. Dostupné také z: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1401277

[17] Building an Automatic Vehicle License-Plate Recognition System. 2005. DUAN, T. D., T. L. H DU, T. V. PHUOC a N. V. HOANG. *Building an Automatic Vehicle License-Plate Recognition System*. Can Tho, Vietnam, s. 59-63.

[18] Real-Time Video Processing Algorithm for Instant License Plate Recognition in iOS Apps. 2014. *AZOFT* [online]. [cit. 2015-05-16]. Dostupné z: http://rnd.azoft.com/instant-license-plate-recognition-in-ios-apps/

[19] IWANOVWSKI, Marcin. Automatic car number plate detection using morphological image processing. 2005. *Przegląd elektrotechniczny*. Warszawa, 81(3), s. 58-61,v. ISSN 033-2097

[20] *PlatesMania: License plates of the Czech Republic* [online]. [cit. 2015-05-16]. Dostupné z: http://platesmania.com/cz/

[21] OCR of Hand-written Data using kNN. *OpenCV-Python Tutorials* [online]. [cit. 2015-05-17]. Dostupné z: https://opencv-python-tutroals.readthedocs.org/en/latest/py_tutorials/py_ml/py_knn/py_knn_opencv/py_knn_opencv.html#knn-opencv

[22] PULORIA, Keerti a Mr. Sachin MAHAJAN. 2013. A Review on Automatic Number Plate Recognition System. *International journal of software & hardware research in engineering*.

Příloha

DVD

Obsah DVD

Součástí přílohy je datový disk, který obsahuje především zdrojové kódy implementovaných algoritmů, spustitelný soubor programu s grafickým rozhraním pro rozpoznávání RZ ze statických snímků a testovací data. Hlubší informace o uložených datech a instrukce k jejich použití je možné najít v textu *ctete.pdf*.

Adresářová struktura DVD a významné soubory:

```
\Algoritmy
      \ALPR_FUNKCE.py
      \ALPR_OBRAZKY.py
      \ALPR_OBRAZKY_GUI.py
      \ALPR_OBRAZKY_GUI_archiv.exe
      \ALPR_OBRAZKY_GUI_LAYOUT.py
      \ALPR_VIDEO
      \add2trainIMG_algoritmus
            \add2train.py
      \trainIMG.png

\Algoritmy_SCREENSHOTS
      \screen_add2train.png
      \screen_ALPR_OBRAZKY.jpg
      \screen_ALPR_OBRAZKY_GUI.jpg
      \screen_ALPR_OBRAZKY_GUI_LAYOUT.png
      \screen_ALPR_VIDEO.jpg
      \video_ALPR_VIDEO.mp4

\testDATA[18]
      \BaslerVideo
      \PlatesmaniaFoto
      \SamsungFoto
      \Dalsi

\testDATA_SAMPLE
      \Add2trainIMG
      \ALPR_OBRAZKY
      \ALPR_VIDEO

\ctete.pdf
\BP_Melnicuk_Dokument.pdf
```

[18] Obsah adresáře testDATA není z důvodu svého objemu součástí elektronicky odevzdávané přílohy a je pouze na DVD. V případě zájmu o tyto data kontaktujte vedoucího práce.