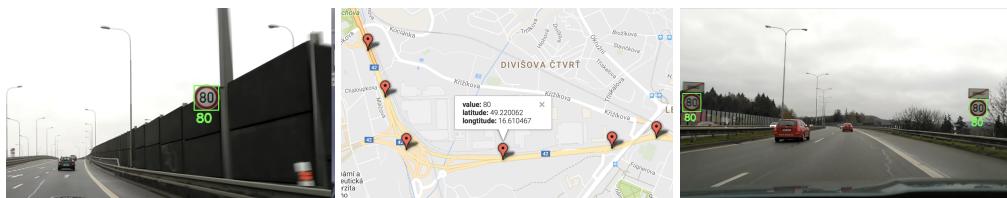


Analýza záznamu palubní kamery automobilu

Jan Kadeřábek*



Abstrakt

Podél veřejných komunikací se nachází velký počet dopravního značení. Nikde však nejsou dostupné souhrnné informace o jeho rozmístění. Vědět, kde je jaká značka je důležité například pro navigační systémy, ale může to být zajímavá informace pro veřejnou správu nebo i běžné občany. Proto se tato práce zabývá analýzou záznamu z palubní kamery, kdy se snaží všechny tyto informace z pořízeného záznamu získat.

Pro popsané účely je využíván kaskádový klasifikátor, který umožňuje rychlou detekci dopravního značení. Následně je využívána klasifikace pomocí k-Nearest Neighbour, která slouží pro klasifikaci typu dopravního značení a případného získání hodnoty na nalezené značce. Práce je implementována v jazyce Python s využitím knihovny OpenCV pro podporu počítačového vidění. Výsledkem práce je program, který umožňuje zpracovat videozáznam a získat z něj uvedené informace, které lze dále využívat. Detekce dopravního značení dosahuje vysoké spolehlivosti a je tak možné pracovat s reálnými výsledky.

Zpracované informace o dopravním značení je možné snadno vizualizovat na mapě a je možné je zveřejňovat k dalším účelům.

Klíčová slova: Kaskádový klasifikátor — Mapování dopravního značení — Palubní kamera

Přiložené materiály: [Demonstrační video](#) — [GitHub](#)

*xkader13@fit.vutbr.cz, Faculty of Information Technology, Brno University of Technology

1. Úvod

Dopravní značení podél veřejných komunikací je velice rozsáhlé a jeho zmapování je vhodné z několika důvodů. Jedná se o užitečné podklady pro navigační systémy a může sloužit také jako informační prostředek pro občany nebo pro městskou správu.

Tato práce si klade za cíl detektovat a rozpoznávat dopravní značení. Za tímto účelem se zabývá detekcí pomocí kaskádového klasifikátoru a rozlišením konkrétních dopravních značek pomocí klasifikace metodou k-Nearest Neighbour. U některých typů dopravních značek následuje klasifikace hodnot v oblasti značky, například při výskytu rychlostního omezení.

V průběhu práce je navrženo řešení, které je schopné implementovat všechny uvedené požadavky.

Během práce vzniklo několik skriptů v jazyce Python, které postupně zpracovávají jednotlivé kroky celého průběhu analýzy záznamu z palubní kamery.

V úvodní kapitole se nachází analýza existujících řešení a porovnání jejich silných a slabých stránek. V třetí kapitole je popsán celý navržený proces zpracování videozáznamu. Následující kapitola se věnuje postupu implementace jednotlivých částí. V páté kapitole se nachází souhrn provedených experimentů. Na závěr je čtenář informován o vyhodnocené celé práce.

26	2. Existující řešení	
27	2.1 Metody detekce dopravního značení	
28	Základní používanou metodou je detekce v barevném	74
29	modelu, kdy dochází k extrakci míst do binární reprezen-	75
30	tace na základě tolerance pro určitou barvu [1]. Poté se	76
31	v binární mapě vyhledávají požadované tvary pomocí	
32	Houghovy transformace [2].	
33	Tyto popsané principy už nejsou příliš moderní,	
34	proto se více používá například kaskádový klasifikátor	
35	[3]. Ten pracuje na principu posloupnosti několika	
36	klasifikátorů za sebou s využitím algoritmu AdaBoost.	
37	Jeho výhodou je relativně snadný proces trénování a	
38	přesná a rychlá detekce.	
39	Nejpokročilejším způsobem jsou konvoluční neu-	
40	ronové sítě, jejichž aplikace se velmi rozšiřuje [4].	
41	2.2 Metody klasifikace dopravního značení	
42	Jednou z možností jak klasifikovat dopravní značení je	
43	metoda k-Nearest Neighbour. Ta umožňuje dobře fun-	
44	gující detekci s využitím i menšího množství vzorků	
45	[5].	
46	Další možností je využít klasifikátory do více tříd	
47	(SVM). Jejich využití pro klasifikaci dopravního značení	
48	je velmi časté. Základním principem je zde klasifikace	
49	příznaků do dvou skupin.	
50	Pro tyto účely je opět možné využít neuronové	
51	sítě.	
52	3. Analýza záznamu	
53	Proces analýzy záznamu byl rozdělen na několik částí	
54	popsaných v následujících kapitolách. Při návrh byl	
55	brán potaz na co největší dekompozici problému. Celý	
56	proces tedy představuje posloupnost uvedených akcí.	
57	3.1 Detekce značky	
58	Prvním bodem analýzy záznamu je detekce samotného	
59	dopravního značení. Pro tento účel byl zvolen kaská-	
60	dový klasifikátor jehož princip je popsán v předchozí	
61	kapitole. Jeho hlavní výhodou je rychlosť detekce, což	
62	je důležitý aspekt vzhledem k tomu, že se očekává	
63	zpracovávání většího množství dat.	
64	Hlavní skript bude procházet jednotlivé snímky, ve	
65	kterých bude hlavním úkolem detektoru nalézt všechny	
66	dopravní značky, které se ve snímku nachází. Jedním	
67	z požadavků je aby detektor byl schopen nalézt více	
68	značek nikoliv jen jednu.	
69	3.2 Rozlišení typu dopravní značky	
70	Detekované značky je vhodné klasifikovat podle je-	
71	jich typu, který vyjadřuje informaci o jakou dopravní	
72	značku se jedná. To bude mít na starosti klasifikace	
73	pomocí metody k-Nearest Neighbour. Díky tomu, že	
58	pomocí detektoru získáme ohraničenou pozici značky	
59	bude klasifikace v rámci výřezu poměrně rychlá a	
60	spolehlivá.	
77	3.3 Hodnota dopravní značky	
78	U některých dopravních značek, zejména u rychlost-	
79	ního omezení, bude zapotřebí dále klasifikovat jejich	
80	hodnotu. Tedy v případě značky omezující rychlost	
81	získat hodnotu maximální povolené rychlosti v daném	
82	místě.	
83	Pro tuto akci bylo navrhнуто, že se nejprve provede	
84	segmentace jednotlivých symbolů, které se nachází v	
85	ohraničeném prostoru značky. Symboly pak budou	
86	odděleně klasifikovány a následně se sestaví výsledné	
87	hodnota.	
88	Klasifikace bude v tomto případě opět probíhat s	
89	využitím metody k-Nearest Neighbour [6].	
90	3.4 Mapování dopravního značení	
91	Výsledkem procesu analýzy by měl být datový soubor,	
92	který bude obsahovat všechny detekované značky a	
93	informace o nich jako je typ, případně hodnota a GPS	
94	souřadnice s jejich polohou.	
95	Za tímto účelem jsou pořizovány záznamy pomocí	
96	mobilní aplikace, které ke každému videozáznamu	
97	přiloží soubor obsahující GPS souřadnice v konkrétním	
98	čase.	
99	4. Implementace	
100	Vytvořené řešení využívá metod popsaných v předchozí	
101	kapitole. Aby bylo možné využívat jednotlivé metody,	
102	bylo nutné připravit datovou sadu pro natrénování	
103	použitých algoritmů.	
104	Všechny skripty jsou implementovány v jazyce	
105	Python 3 s využitím knihovny pro počítacové vidění	
106	OpenCV 3.	
107	4.1 Trénování kaskádového klasifikátoru	
108	Základem bylo vytvořit sadu pozitivních a negativních	
109	snímků pro natrénování kaskádového klasifikátoru. K	
110	tomuto účelu byl využit skript, který umožňuje de-	
111	tekci dopravních značek v barevném HSV modelu a	
112	následné vyhledávání tvarů pomocí Houghovy trans-	
113	formace (v nalezených konturách).	
114	V případě nalezení objektu splňující definované	
115	požadavky je detekovaný objekt prohlášen za možnou	
116	značku a specifický výřez se uloží do složky pro pozi-	
117	tivní snímky.	
118	Naopak snímek, kde žádný takový objekt nebyl	
119	detekován je uložen do složky pro negativní vzorky.	
120	Celkově bylo připraveno 400 pozitivních snímků	
121	zákazových značek a 1000 negativních z vytvořených	



Obrázek 1. Ukázka vstupního datasetu pro trénování kaskádového klasifikátoru

zmiňovaným postupem. Jak vypadaly pozitivní smínky lze vidět na obrázku č. 1. Tyto snímky byly použity pro vytvoření datového souboru pozitivních vzorků. Ten byl vytvářen aplikací `opencv_createsamples`¹, která vytvářela vzorky o velikosti 50x50px, vzhledem k tomu že vstupní snímky obsahovaly pouze požadovanou oblast, tak nebylo příliš nutné zpracovávat soubor s anotacemi, ten obsahoval pouze seznam souborů s velikostí obrázků.

K trénování kaskádového klasifikátoru byla použita utilita `opencv_traincascade`². Trénování bylo spuštěno s příznaky LBP, s počtem 10 iterací a parametrem `maxFalseAlarmRate` s hodnotou 0.3.

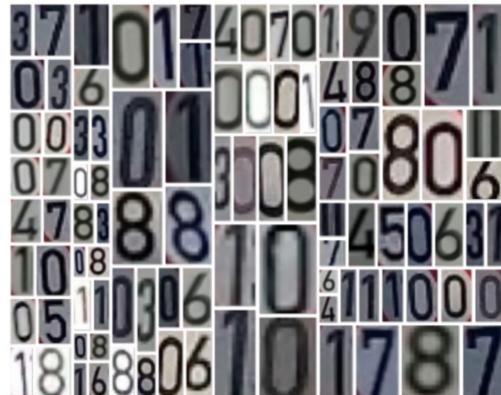
4.2 Klasifikace druhů značek

S využitím funkčního kaskádového klasifikátoru bylo možné zpracovávat video záznamy a získávat soubory s detekovanými značkami. Tyto snímky následně byly rozděleny podle jednotlivých druhů a to zvlášť pro větší sadu určenou k trénování a menší určenou k testování klasifikace.

Metoda k-Nearest Neighbour funguje na principu hledání nejbližších sousedů v zadefinovaných příznacích. Ty je nutné si nejprve připravit. Jejich sestavení v tomto případě probíhá vytvořením binární reprezentace dopravní značky, což ilustruje obrázek č. 2. Takováto binární reprezentace je poté převedena na jednodimensionální pole, které je serializováno do souboru a oanotováno příslušným významem. Tento soubor si později načte klasifikátor pro kNN, který je



Obrázek 2. Vyprahovaná binární reprezentace dopravní značky pro klasifikaci druhu



Obrázek 3. Dataset pro trénování klasifikace symbolů

dostupný v rámci knihovny OpenCV. Při klasifikaci je nutné klasifikovanou oblast upravit stejným způsobem jako při přípravě trénovacích dat, poté už stačí nechat určit klasifikátor do jaké třídy vzorek zařadí.

4.3 Klasifikace rychlostního omezení

Ze skupiny rychlostního omezení byly vybrány všechny snímky nad nimiž proběhla segmentace znaků a opět byly rozděleny do jednotlivých skupin podle významu jednotlivý znaků, tedy symbolů 0-9. Jednotlivé extra-hované znaky lze vidět na obrázku č. 3. Následně bylo spuštěno automatické natrénování klasifikátoru z připravené datové sady.

Klasifikace hodnoty nebyla vždy úplně přesná a proto bylo přistoupeno k porovnávání hodnot v rámci předešlých snímků a následné odfiltrování pravděpodobně nerelevantních výsledků.

4.4 Trackování značky

Vzhledem k tomu, že detekce ani klasifikace nebyly vždy stoprocentní, bylo nutné tyto nedostatky odstínit. V případě detekce k výpadku docházelo zejména u rozmazených snímků. Ideálním způsobem se ukázalo sledování pozice dopravní značky a porovnávání klasifikovaných hodnot v předchozích snímcích.

Pokud na aktuálním snímku nebyla nalezena žádná značka, dojde ke kontrole přítomnosti značky na předchozím snímku. V případě, že je zde přítomné minimálně jedno dopravní značení, přistoupí se k vypočítání pravděpodobného umístění na aktuálním snímku, kde nedošlo k úspěšné detekci. To probíhá ze dvou předchozích známých pozic v rámci snímku.

¹http://docs.opencv.org/3.0-alpha/doc/user_guide/ug_traincascade.html#training-data-preparation

²http://docs.opencv.org/3.0-alpha/doc/user_guide/ug_traincascade.html#cascade-training



Obrázek 4. Zábery z kamerového systému ukazující zakrytí dopravních značek pro testování False Positive detekce



Obrázek 5. Ukázka vizualizace jednoho zpracovaného videozáznamu

AutoBoy Dash Cam - BlackBox [7]. Pořízený záznam má rozlišení 1280x720 px s frekvencí 30 snímků za sekundu. Celková délka záznamu činí přesně pět minut a velikost videosouboru je 451 MB. K záznamu byl vygenerován standardní SRT soubor. Ten obsahuje bloky informací, kde vždy jeden záznam reprezentuje čas platnosti, aktuální rychlosť, GPS souřadnice a adresu místa pořízení. Tyto informace jsou vždy po jedné sekundě, což dostačuje k určení pozice.

Skript byl spuštěn se zadáním parametrů pořízeného videa a přiloženého SRT souboru. Celý proces zpracování trval 199 s, rychlosť zpracování tedy odpovídá přibližně 45 snímků za sekundu.

Výsledkem je CSV soubor, který obsahuje celkově sedm řádků, kdy se na každém řádku nachází hodnota rychlostního omezení, a zeměpisná šířka a délka. Úspěšnost detekce a klasifikace je v tomto případě 100%. Vytvořený soubor byl nahrán do webové služby Google Fusion Tables. Ta umožňuje z importované tabulky vykreslit interaktivní mapu jejíž náhled lze vidět obrázku č. 5.

Tabulka 1. Úspěšnost detekce

True Positive	92.3%
False Positive	0.45%

6. Shrnutí

V práci byl prezentován postup analýzy záznamu automobilu vedoucí ke zmapování dopravního značení. Výsledkem je navrhnutý a implementovaný proces celé analýzy.

Práce ukazuje jak lze využít kaskádový klasifikátor pro detekci dopravních značek a jak je možné vytvořit datovou sadu pro trénování. Dále je přednesena možnost jak realizovat klasifikaci dopravního značení metodou kNN a to nejen pro klasifikaci typu, ale i klasifikaci konkrétní hodnoty.

Všechny zdrojové kódy jsou zveřejněny ve veřejném repozitáři služby GitHub³. To dává možnost komukoli na ně nahlédnout a využít je jako inspiraci pro

³<https://github.com/jankaderabek/dashcam-analyzer>

5.2 Úspěšnost klasifikace druhu

Testování rozlišení klasifikace druhu dopravní značky probíhalo na datové skupině zahrnující pět druhů značek po pěti vzorcích. Sada pro trénování obsahovala vzorku 50. Dosažená úspěšnost nad testovacími daty dosahovala 87 %.

5.3 Zpracování jednoho záznamu

Tento experiment ukazuje proces zpracování jednoho záznamu. Záznam byl pořízen běžným mobilním telefonem s operačním systémem Android pomocí aplikace

181 **5. Experimenty**

182 **5.1 Úspěšnost detekce dopravních značek**

183 Pro vyhodnocení úspěšnosti detekce byly brány v potaz
184 dva základní parametry a to *True Positive* vypovídající
185 o správné detekci a *False Positive* vyjadřující chybnou
186 detekci, kdy na snímku žádná značka není, ale klasifi-
187 kátor označí nějakou oblast jako dopravní značku.

188 Pro vyhodnocení *True Positive* bylo využito při-
189 pravené testovacího video, ze kterého byly vybrány vše-
190 chny snímky s dopravní značkou, celkově takových
191 snímků bylo 520. Nad těmito snímky byl spuštěn
192 kaskádový klasifikátor, který byl schopen dopravní
193 značku detektovat celkově na 480 snímcích, procentuální
194 úspěšnost je pak vyjádřena v tabulce č. 1.

195 V případě parametru *False Positive* byly z testo-
196 vacího vyextrahovány všechny snímky o celkovém
197 počtu 1762. Na snímcích, kde se nacházely dopravní
198 značky došlo k zakrytí téhoto oblasti jak je vidět na
199 obrázku č. 4. Kaskádový klasifikátor byl opět spuštěn
200 nad těmito snímky a výsledkem je velmi nízké číslo
201 nechteňých detekcí s hodnotou 0.45% viz. tabulka č.
202 1. Toto číslo lze navíc snadno ještě snížit, díky tomu, že
203 nad detekovanými oblastmi probíhá další klasifikace.

235

236 V práci byl prezentován postup analýzy záznamu au-
237 tomobilu vedoucí ke zmapování dopravního značení.
238 Výsledkem je navrhnutý a implementovaný proces celé
239 analýzy.

240 Práce ukazuje jak lze využít kaskádový klasifikátor
241 pro detekci dopravních značek a jak je možné vytvořit
242 datovou sadu pro trénování. Dále je přednesena možnost
243 jak realizovat klasifikaci dopravního značení metodou
244 kNN a to nejen pro klasifikaci typu, ale i klasifikaci
245 konkrétní hodnoty.

246 Všechny zdrojové kódy jsou zveřejněny ve veřejném
247 repozitáři služby GitHub³. To dává možnost komuko-
248 li v na ně nahlédnout a využít je jako inspiraci pro

249	vlastní práci, případně poskytuje možnost navrhnout 250 možná vylepšení současné práce.	http://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_ml/py_knn/py_knn_understanding/py_knn_understanding.html .	297
251	Hlavním cílem budoucí práce je pokrýt všechny 252 dopravní značky a získat tak větší množství dat pro 253 další agregaci.		298
254	S tím souvisí nabídnutí možnosti veřejnosti jak 255 pomocí se shromažďováním dat. Ideálním řešením se 256 jeví webová aplikace, která by byla schopna zaznamenat 257 zpracovaná data ze záznamů.		299
258	Následně by bylo možné realizovat různé analýzy 259 nad získanými daty a nabídnout například přehled do- 260 pravního značení v širším měřítku.		300
261	Poděkování		301
262	Za veškerou pomoc během bakalářské práce, v rámci 263 které vznikl i tento článek, bych chtěl poděkovat svému 264 vedoucímu Ing. Jakubovi Špaňhelovi.		
265	Literatura		
266	[1] E. Martin-Gorostiza M.A. Garcia-Garrido, 267 M.A. Sotelo. Fast traffic sign detection and 268 recognition under changing lighting conditions, 269 2006. http://ieeexplore.ieee.org/abstract/document/649946/ .		
271	[2] E. Martin-Gorostiza M.A. Garcia-Garrido, 272 M.A. Sotelo. Fast traffic sign detection and 273 recognition under changing lighting conditions, 274 2006. http://ieeexplore.ieee.org/abstract/document/1706843/ .		
276	[3] Michael Jones Paul Viola. Rapid object 277 detection using a boosted cascade of simple fea- 278 tures, 2001. https://www.cs.cmu.edu/%7Eefros/courses/LBMV07/Papers/viola-cvpr-01.pdf .		
281	[4] Jianmin Li Huaping Liu Xiaolin Hu Yi- 282 hui Wu, Yulong Liu. Traffic sign detec- 283 tion based on convolutional neural networks, 284 2013. http://ieeexplore.ieee.org/abstract/document/6706811/ .		
286	[5] Alexander Chigorin Anton Konushin 287 Boris Moiseev, Artem Konev. Evalu- 288 ation of traffic sign recognition methods 289 trained on synthetically generated data, 2013. 290 http://link.springer.com/chapter/10.1007/978-3-319-02895-8_52 .		
292	[6] Autoboy dash cam - blackbox. https://play.google.com/store/apps/details?id=com.happyconz.blackbox&hl=cs .		
295	[7] Alexander Mordvintsev. Understanding k-nearest 296 neighbour. blogpost (english), March 2013.		