

# Analýza záznamu palubní kamery automobilu

Jan Kadeřábek\*



## Abstrakt

Práce se zabývá zpracováním záznamu palubní kamery vozidla. Při zpracování probíhá detekce dopravního značení včetně klasifikace konkrétního typu a případně i zjištění specifické hodnoty. K této účelům je využíván kaskádový klasifikátor, který umožňuje rychlou detekci dopravního značení. Následně je aplikována metoda k-Nearest Neighbour 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 spolehlivosti 92 %, což je následně ještě vylepšováno díky sledování pozice značky.

Zpracované informace o dopravním značení je možné snadno vizualizovat na mapě a je možné je zveřejňovat k dalším účelům. To se může velmi hodit pro mapové podklady nebo jiné plánování tras.

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

**Připojené materiály:** Demonstrační video — [GitHub](#)

\*[xkader13@fit.vutbr.cz](mailto:xkader13@fit.vutbr.cz), Faculty of Information Technology, Brno University of Technology

## 1. Úvod

Podél veřejných komunikací se nachází velký počet dopravního značení. Nikde však nejsou dostupné 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 vozidla, kdy se snaží tyto informace z pořízeného záznamu získat.

Tato práce si klade za cíl detektovat a rozpoznávat dopravní značení. Pro řešení vymezených cílů se práce zabývá detekcí v obraze pomocí kaskádového klasifikátoru a rozlišení konkrétních dopravních značek s využitím klasifikace metodou k-Nearest Neighbour. U některých typů dopravních značek následuje klasifi-

kace hodnot v oblasti značky, například při výskytu rychlostního omezení.

Základním cílem je takto detektovat a analyzovat umístění zákazových dopravních značek. Ty mají největší dopad na možnost zvolení trasy a zároveň mají specifickou charakteristiku a díky tomu je možné je snadno detektovat.

V průběhu práce je navrženo řešení, které je postupně implementováno a jsou prezentovány výsledky implementovaných postupů.

V úvodní kapitole se nachází analýza existujících řešení a porovnání jejich silných a slabých stránek. V třetí kapitole je popsán celý navržený proces zpracování. Později je popsána konkrétní implementace a vyhodnocení implementovaných metod.

## 2. Existující řešení

Možností jak detektovat a klasifikovat objekty v obraze je několik. Některé jsou jednodušší, ale můžou dosahovat nechtěných vlastností, nicméně je možné je využít pro podporu pokročilejších řešení.

### 2.1 Metody detekce dopravního značení

Základní používanou metodou je detekce v barevném modelu. Využívá se toho, že značky mají specifické barvy, které je možné v obraze vyfiltrovat. To se děje na základě toho, že se pro každou barvu stanoví specifická spodní a horní mez, kdy odstín považujeme za barvu, která nás zajímá [1]. Tyto metody často experimentují jaký barevný model použít jestli základní RGB a nebo například vhodnější pro tuto problematiku a to HSV, který umožnuje lépe vymezit žádaný rozsah. Poté se v binární mapě vyhledávají požadované tvary pomocí Houghovy transformace [2].

Tyto popsané principy už nejsou příliš moderní, proto se více používá například kaskádový klasifikátor [3]. Ten pracuje na principu posloupnosti několika klasifikátorů za sebou s využitím algoritmu AdaBoost. Jeho výhodou je relativně snadný proces trénování a přesná a rychlá detekce.

Nejpokročilejším způsobem jsou konvoluční neuronové sítě, jejichž aplikace se velmi rozšiřuje [4].

### 2.2 Metody klasifikace dopravního značení

Jednou z možností jak klasifikovat dopravní značení je metoda k-Nearest Neighbour. Ta umožňuje dobře fungující detekci s využitím i menšího množství vzorků [5].

Další možností je využít klasifikátory do více tříd (SVM). Jejich využití pro klasifikaci dopravního značení je velmi časté. Základním principem je zde klasifikace příznaků do dvou skupin.

Pro tyto účely je opět možné využít neuronové sítě.

## 3. Analýza záznamu

Proces analýzy záznamu jsem se rozhodl rozdělit na několik částí popsaných v následujících kapitolách. Při návrhu je dbáno na co největší dekompozici problému a proces analýzy záznamu se tak sestavá z průchodu obrazu několika komponentami.

### 3.1 Detekce značky

Prvním bodem analýzy záznamu je detekce samotného dopravního značení. Pro tento účel je zvolen kaskádový klasifikátor. Jeho hlavní výhodou je rychlosť detekce, což je důležitý aspekt vzhledem k tomu, že se očekává zpracovávání většího množství dat.

Hlavní úkolem v této části je procházení jednotlivých snímků, ve kterých se bude detektor snažit nalézt všechny dopravní značky, které se ve snímku nachází. Jedním z požadavků je aby detektor byl schopen nalézt více značek nikoliv jen jednu a to jak stejného významu, které jsou umístěny na obou stranách silnice, tak i v případě přítomnosti více druhů značek na jednom snímku.

### 3.2 Rozlišení typu dopravní značky

Detekované značky je vhodné klasifikovat podle jejich typu, který vyjadřuje informaci o jakou dopravní značku se jedná. To bude má na starosti klasifikace pomocí metody k-Nearest Neighbour. Díky tomu, že detektor označí ohrazenou pozici značky bude klasifikace v rámci výřezu poměrně rychlá a spolehlivá.

### 3.3 Hodnota dopravní značky

U některých dopravních značek, zejména u rychlostního omezení, je zapotřebí dále klasifikovat jejich hodnotu. Tedy v případě značky omezující rychlosť získat hodnotu maximální povolené rychlosti v daném místě.

Pro tuto akci je navrhnutu, že se nejprve provede segmentace jednotlivých symbolů, které se nachází v ohrazeném prostoru značky. Symboly pak jsou odděleně klasifikovány a následně se sestaví výsledné hodnoty.

Pro popsané účely bude zapotřebí nejprve zmenšit výřez oblasti a následně upravit histogram výřezu, tak aby bylo možné prahováním získat klíčové části obrazu identifikující konkrétní značku. Po provedení prahování je možné v binární reprezentaci nalézt vnější kontury, které odpovídají jednotlivým symbolům na dopravní značce.

Binární reprezentace symbolu slouží jako příznaky pro klasifikaci pomocí metody k-Nearest Neighbour [6].

### 3.4 Mapování dopravního značení

Výsledkem procesu analýzy je soubor, který obsahuje všechny informace o detekovaných značkách jako je jejich typ, volitelně hodnota, GPS souřadnice a výřez detekované značky.

Za tímto účelem jsou předpokladem ke zpracování záznamy pořízené pomocí mobilní aplikace, která ke každému videozáznamu přiloží soubor obsahující GPS souřadnice v konkrétním čase.

## 4. Implementace

Vytvořené řešení využívá metod popsaných v předchozí kapitole. Aby bylo možné využívat jednotlivé metody, je nutné připravit datovou sadu pro natrénování použitých algoritmů.



**Obrázek 1.** Ukázka vstupního datasetu pro trénování kaskádového klasifikátoru

Všechny skripty jsou implementovány v jazyce Python 3 s využitím knihovny OpenCV 3.

Za vytvoření datové sady, která je předpokladem pro natrénování klasifikátoru probíhalo sbírání videí z různých typu pozemních komunikací. Ty byly pořizovány mobilním telefonem s rozlišením záznamu  $1280x720px$ . Videá pochází z osobního automobilu, natáčené za běžného provozu a to na různých typech pozemních komunikacích včetně dálnic a silnic první až třetí třídy. Záznamy tedy zahrnují záběry jak z centra měst (zejména Brno), tak i z okresních silnic. Natáčení probíhalo z větší části během zimního období, kdy mobilní telefon nebyl schopen pracovat příliš dobře s osvětlením, to však může být považováno za výhodu, protože všechny navrhnuté postupy počítají i s nepříliš kontrastním obrazem.

#### 4.1 Trénování kaskádového klasifikátoru

Základem je vytvořit sadu pozitivních a negativních snímků pro natrénování kaskádového klasifikátoru. K tomuto účelu je využit vytvořený skript, který umožňuje detekci dopravních značek v barevném HSV modelu a následném vyhledávání tvarů pomocí Houghovy transformace (v nalezených konturách).

V případě nalezení objektu splňující definované požadavky je detekovaný objekt prohlášen za možnou značku a specifický výrez se uloží do složky pro pozitivní snímky.

Naopak snímek, kde žádný takový objekt není detekován je uložen jako negativní vzorek.

Celkově bylo připraveno 970 pozitivních snímků zákazových značek a 1000 negativních vytvořených zmiňovaným postupem. Jak vypadaly pozitivní snímky lze vidět na obrázku č. 1. Tyto snímky byly použity pro vytvoření datového souboru pozitivních vzorků.



**Obrázek 2.** Vyprahovaná binární reprezentace dopravní značky pro klasifikaci druhu

Ten byl vytvářen aplikací *opencv\_createsamples*<sup>1</sup>, která vytvářela vzorky o velikosti  $50x50px$ , vzhledem k tomu, že vstupní snímky obsahovaly pouze požadovanou oblast, nebylo příliš nutné zpracovávat soubor s anotacemi, ten obsahoval pouze seznam souborů s velikostí obrázků.

K trénování kaskádového klasifikátoru je použita utilita *opencv\_traincascade*<sup>2</sup>. 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 je možné zpracovávat videozáznamy a získávat soubory s detekovanými značkami. Tyto snímky jsou následně rozděleny podle jednotlivých tříd a to zvlášť pro větší sadu určenou k trénování a menší určenou k testování klasifikace.

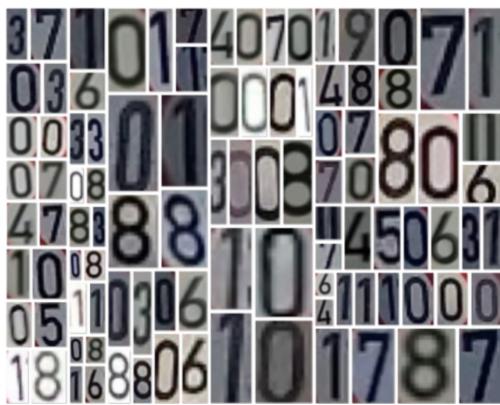
Metoda k-Nearest Neighbour funguje na principu hledání nejbližších sousedů v zadefinovaných příznacích. Ty je nutné si nejprve připravit. Jejich sestavení v tomto případě probíhá vytvořením binární reprezentace dopravní značky, což ilustruje obrázek č. 2. Takováto binární reprezentace je převedena na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

Díky implementaci klasifikace typu dopravní značky je možné snadno získat všechny značky z kategorie rychlostního omezení. Nad touto sadou je spuštěn algoritmus pro segmentaci znaků. Vyextrahované značky je nutné ručně roztrídit podle jejich hodnoty, tak aby bylo možné spustit automatické trénování pro klasifikaci metodou k-Nearest Neighbour, které opět využívá-

<sup>1</sup>[http://docs.opencv.org/3.0-alpha/doc/user\\_guide/ug\\_traincascade.html#training-data-preparation](http://docs.opencv.org/3.0-alpha/doc/user_guide/ug_traincascade.html#training-data-preparation)

<sup>2</sup>[http://docs.opencv.org/3.0-alpha/doc/user\\_guide/ug\\_traincascade.html#cascade-training](http://docs.opencv.org/3.0-alpha/doc/user_guide/ug_traincascade.html#cascade-training)



**Obrázek 3.** Dataset pro trénování klasifikace symbolů

la binární reprezentaci znaku. Jednotlivé extrahované znaky lze vidět na obrázku č. 3.

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

#### 4.4 Sledování pozice dopravní značky

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

Pokud na aktuálním snímku není nalezena žádná značka, dochází 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.

### 5. Experimenty

Pro vyhodnocení implementace uvedeného návrhu je vhodné provést experimenty, které vyjádří míru úspěšnosti řešení.

#### 5.1 Úspěšnost detekce dopravních značek

Vyhodnocení úspěšnosti detekce bere v potaz dva základní parametry a to *True Positive* vypovídající o správné detekci a *False Positive* vyjadřující chybou detekci, kdy na snímku žádná značka není, ale klasifikátor označí oblast jako dopravní značku.

Pro vyhodnocení True Positive je využito připravené testovací video, ze kterého jsou vybrány všechny snímkы s dopravní značkou, celkově takových snímků je 520. Nad těmito snímkem je spuštěn kaskádový klasifikátor, který je schopen dopravní značku detektovat celkově na 480 snímcích, procentuální úspěšnost je



**Obrázek 4.** Zakrytí dopravních značek pro testování False Positive detekce

pak vyjádřena v tabulce č. 1. Výsledná úspěšnost je ovlivněna hlavně testovacími snímkami, kdy ve většině případů značka není detekována kvůli nepříliš kvalitnímu záznamu.

V případě parametru *False Positive* jsou z testovacího videa vyextrahovány všechny snímkы o celkovém počtu 1762. Na snímcích, kde se nachází dopravní značky jsou tyto oblasti zakryty, jak je vidět na obrázku č. 4. Kaskádový klasifikátor je opět spuštěn nad těmito snímkami a výsledkem je velmi nízké číslo nechtěných detekcí s hodnotou 0.45% viz. tabulka č. 1. Toto číslo lze navíc snadno ještě snížit, díky tomu, že nad detekovanými oblastmi probíhá další klasifikace.

**Tabulka 1.** Úspěšnost detekce

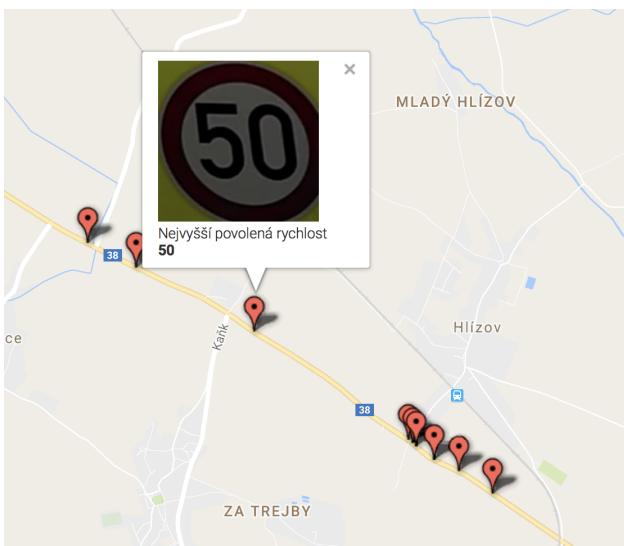
True Positive	92.3%
False Positive	0.45%

#### 5.2 Úspěšnost klasifikace druhu

Testování rozlišení druhu dopravní značky probíhá na datové skupině zahrnující celkem 184 dopravní značek různých skupin. Sada pro trénování obsahovala 1256 vzorků. Dosažená úspěšnost dosahuje 88.04 %. Jako nejproblematický se ukazuje rozlišit skupinu zákazu stání a zastavení. To je způsobeno tmavou výplní, kvůli které při prahování nevyniknou hlavní rysy těchto typů dopravních značek.

#### 5.3 Úspěšnost klasifikace hodnoty

Pro vyhodnocení úspěšnosti zjištění hodnoty rychlostního limitu je využita testovací sada, která zahrnuje celkově 105 snímků dopravních značek rychlostního omezení. Ty jsou ručně oanotovány podle jejich hodnot. Následně dojde kde spuštění vytvořeného skriptu, který nad snímkem provede segmentaci znaků a jejich klasifikaci. Výsledkem je klasifikace s úspěšností 91.43 %.



**Obrázek 5.** Ukázka vizualizace jednoho zpracovaného videozáznamu

#### 5.4 Zpracování jednoho záznamu

Tento experiment ukazuje proces zpracování jednoho záznamu. Záznam pořízený běžným mobilním telefonem s operačním systémem Android pomocí aplikace AutoBoy Dash Cam - BlackBox [7]. Pořízený záznam má rozlišení  $1280 \times 720 \text{ px}$  s frekvencí 30 snímků za sekundu. Celková délka záznamu činí přesně pět minut a velikost videosouboru je 451 MB. Aplikace ještě k videosouboru vygeneruje 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.

### 6. Závěr

V práci je 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 různých klasifikačních metod. 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 v repozitáři služby GitHub<sup>3</sup>. To dává možnost komukoliv na ně nahlédnout a využít je jako inspiraci pro vlastní práci, případně poskytuje možnost navrhnout možná vylepšení současné práce.

Jedním z dalších cílů je implementovat zjištění pozice v rámci prostoru a určit například pro jaký pruh má detekované dopravní značení platnost. Jiný budoucí cíl práce je pokrýt všechny dopravní značky a získat tak větší množství dat pro další agregaci. S tím souvisí nabídnutí možnosti veřejnosti jak pomoci se shromažďováním dat. Ideálním řešením se jeví webová aplikace, která by byla schopna zaznamenat zpracovaná data ze záznamů.

Následně by bylo možné realizovat různé analýzy nad získanými daty a nabídnout například přehled dopravního značení v širším měřítku.

### Poděkování

Za veškerou pomoc během bakalářské práce, v rámci které vznikl i tento článek, bych chtěl poděkovat svému vedoucímu Ing. Jakubovi Špaňhelovi.

### Literatura

- [1] E. Martin-Gorostiza M.A. Garcia-Garrido, M.A. Sotelo. Fast traffic sign detection and recognition under changing lighting conditions, 2006. <http://ieeexplore.ieee.org/abstract/document/649946/>.
- [2] E. Martin-Gorostiza M.A. Garcia-Garrido, M.A. Sotelo. Fast traffic sign detection and recognition under changing lighting conditions, 2006. <http://ieeexplore.ieee.org/abstract/document/1706843/>.
- [3] Michael Jones Paul Viola. Rapid object detection using a boosted cascade of simple features, 2001. <https://www.cs.cmu.edu/~Eefros/courses/LBMV07/Papers/viola-cvpr-01.pdf>.
- [4] Jianmin Li Huaping Liu Xiaolin Hu Yihui Wu, Yulong Liu. Traffic sign detection based on convolutional neural networks, 2013. <http://ieeexplore.ieee.org/abstract/document/6706811/>.

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

- [5] Alexander Chigorin Anton Konushin Boris Moiseev, Artem Konev. Evaluation of traffic sign recognition methods trained on synthetically generated data, 2013. [http://link.springer.com/chapter/10.1007/978-3-319-02895-8\\_52](http://link.springer.com/chapter/10.1007/978-3-319-02895-8_52).
- [6] Autoboy dash cam - blackbox. <https://play.google.com/store/apps/details?id=com.happyconz.blackbox&hl=cs>.
- [7] Alexander Mordvintsev. Understanding k-nearest neighbour. blogpost (english), March 2013. [http://opencv-python-tutorials.readthedocs.io/en/latest/py\\_tutorials/py\\_ml/py\\_knn/py\\_knn\\_understanding/py\\_knn\\_understanding.html/](http://opencv-python-tutorials.readthedocs.io/en/latest/py_tutorials/py_ml/py_knn/py_knn_understanding/py_knn_understanding.html/).