

**Vyšší odborná škola, Střední průmyslová škola a
Střední odborná škola, Varnsdorf, příspěvková
organizace**



ROZPOZNÁVÁNÍ DOPRAVNÍCH ZNAČEK POMOCÍ UMĚLÉ INTELIGENCE

Marek Charvát

4. IT, Informační technologie

Varnsdorf 2023

Obsah

1	Úvod	1
2	Teoretická část	2
2.1	Limitace a náročnost	4
2.2	Strojové učení.....	5
2.3	Proč zrovna Python.....	6
3	Praktická část.....	7
3.1	Využití umělé inteligence	7
3.2	Využití Pythonu.....	8
3.3	Tvorba aplikace.....	9
3.4	Využití při komunikaci	10
3.4.1	Chatboti	10
3.4.2	Automaticky generované odpovědi	11
3.4.3	Hlasoví asistenti.....	12
3.4.4	Robot Operating System (ROS)	13
3.4.5	Predikce textu.....	14
3.4.6	Webové vyhledávače a UI	15
3.5	Ukázka kódu v jazyce Python	16
3.6	Haar kaskády	16
4	Závěr	19
4.1	Aktuálnost tématu.....	20
5	Citace	21
5.1	Čestné prohlášení.....	23
5.2	Poděkování	23

1 Úvod

V dnešní době, kdy se stále více a více vozidel pohybuje na silnicích, je bezpečnost našich silnic klíčovým tématem. Jednou z hlavních příčin dopravních nehod je nedostatečná pozornost řidičů, kteří nedodržují dopravní značení a pravidla silničního provozu.

S rozvojem umělé inteligence se naskýtá možnost využít moderní technologie pro zlepšení bezpečnosti silničního provozu. Projekt rozeznávání dopravních značek pomocí umělé inteligence má za cíl vytvořit systém, který dokáže automaticky rozpoznávat dopravní značky na silnicích.

Takový systém by mohl být velkým přínosem pro bezpečnost na silnicích a mohl by pomoci snížit počet dopravních nehod způsobených nedodržením dopravních pravidel.

I když rozeznávání dopravních značek pomocí umělé inteligence může přinést mnoho výhod v oblasti bezpečnosti silničního provozu, existují také potenciální hrozby, které by mohly ovlivnit úspěšnost tohoto projektu.

Jednou z možných hrozeb je například nesprávné rozpoznávání dopravních značek v důsledku nepříznivých klimatických podmínek nebo špatné kvality silničního značení. To by mohlo vést k nebezpečným situacím na silnicích.

Další potenciální hrozbou je kybernetický útok na systém rozeznávání dopravních značek, který by mohl způsobit narušení správné funkce systému a tím způsobit nebezpečí na silnicích.

Také je nutné brát v úvahu otázku ochrany osobních údajů, protože při rozpoznávání dopravních značek jsou sbírána data o vozidle a jeho řidiči, což může být vnímáno jako porušení soukromí.

V neposlední řadě je nutné zajistit, aby systém rozeznávání dopravních značek byl dostatečně robustní a spolehlivý, aby nedocházelo k falešným pozitivům nebo negativům, které by mohly vést k nebezpečným situacím na silnicích

2 Teoretická část

Projekt rozeznávání dopravních značek pomocí umělé inteligence spočívá v použití pokročilých technologií strojového učení a hlubokého učení k identifikaci dopravních značek na silnicích a k upozornění řidiče na příslušná dopravní pravidla.

Pro úspěšné rozpoznávání dopravních značek je nutné vytvořit velkou databázi obrázků dopravních značek, které budou použity jako trénovací data pro umělou inteligenci. Tyto obrázky musí být různorodé a obsahovat různé varianty dopravních značek, aby systém dokázal rozpoznat i méně časté typy značek.

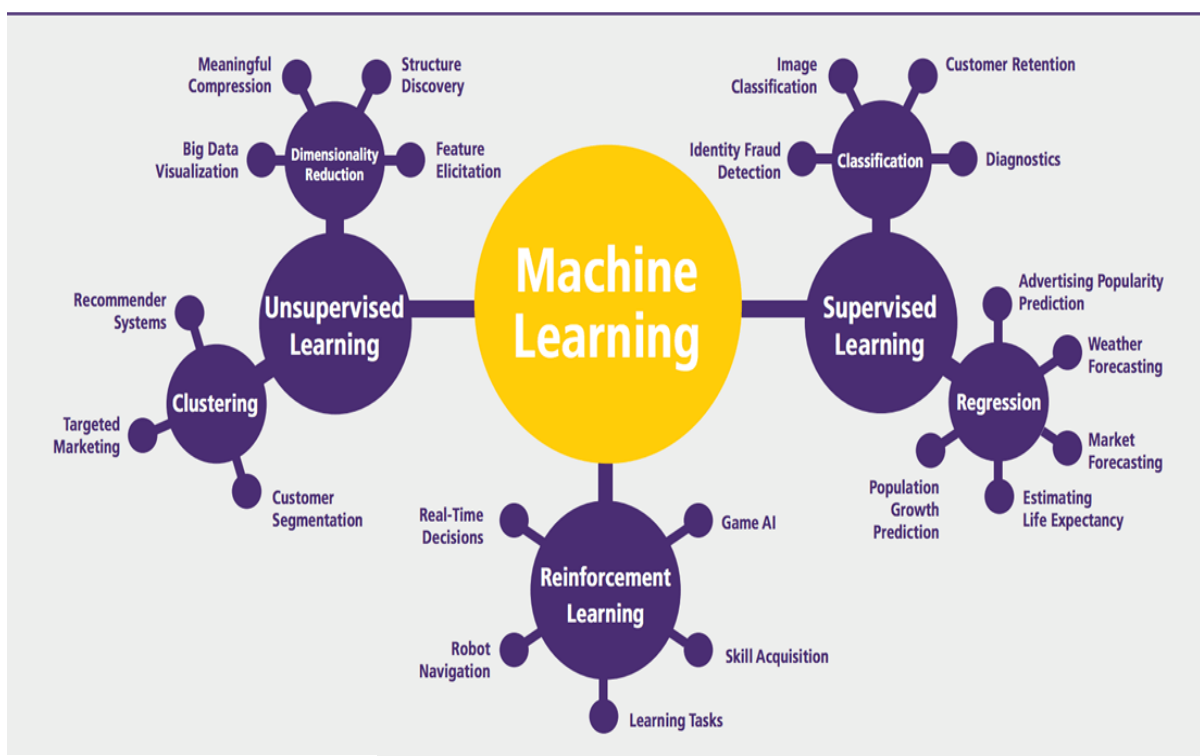
Pro rozpoznávání značek se obvykle využívá neuronových sítí, které jsou schopné učit se rozpoznávat dopravní značky na základě trénovacích dat. V průběhu trénování se síť učí rozpoznávat různé typy značek a postupně se zlepšuje ve své schopnosti rozpoznávat i složitější situace na silnicích.

Kromě rozpoznávání dopravních značek může systém také být vybaven funkcemi, jako jsou detekce rychlosti, rozpoznání jízdních pruhů nebo rozpoznání vzdálenosti mezi vozidly. Tyto funkce mohou být integrovány do systému pro lepší sledování silničního provozu a zlepšení bezpečnosti silničního provozu.

V praxi by mohl takový systém být implementován jako součást moderních vozidel, které by byly schopny rozpoznávat dopravní značky a informovat řidiče o příslušných pravidlech na silnici. Systém by také mohl být propojen s centrálním řídicím systémem silničního provozu, aby mohl poskytovat informace o dopravní situaci na daném úseku silnice a pomoci řidičům při plánování trasy.

Rozpoznávání dopravních značek pomocí umělé inteligence (AI) je aktuální a zajímavé téma v oblasti počítačového vidění a strojového učení. Tento proces spočívá v tom, že počítačový systém pomocí kamer a dalších senzorů sbírá obrazová data z okolí a následně je analyzuje a klasifikuje na základě naučených modelů AI.

Při výběru vhodného algoritmu pro rozpoznávání dopravních značek je nutné zohlednit různé faktory, jako jsou rychlost zpracování, přesnost, robustnost a efektivitu využití zdrojů. Důležitou roli hraje také výběr vhodných datových sad pro trénování modelů AI a testování výsledků.



Obrázek 1 Rozdělení strojového učení

2.1 Limitace a náročnost

Náročnost strojového učení závisí na mnoha faktorech, jako jsou velikost datové sady, složitost modelu, algoritmus učení a výpočetní prostředky k dispozici. Některé modely a algoritmy mohou být jednodušší a vyžadovat méně výpočetních prostředků, zatímco jiné mohou být velmi složité a vyžadovat vysoký výkon a množství dat pro trénování.

Některé z faktorů, které mohou ovlivnit náročnost strojového učení jsou:

Velikost datové sady:

Čím větší je datová sada, tím více času a výpočetních prostředků je potřeba pro trénování modelu.

Složitost modelu:

Složitější modely s více parametry vyžadují více výpočetních prostředků pro trénování.

Algoritmus učení:

Některé algoritmy učení jsou náročnější než jiné. Například algoritmus Gradient Descent vyžaduje více výpočetních prostředků než algoritmus k-Nearest Neighbor.

Výpočetní prostředky:

Výpočetní výkon a dostupné výpočetní prostředky mohou ovlivnit náročnost strojového učení. Pokud máme k dispozici vysoký výkon a paralelní výpočetní architekturu, můžeme trénovat složitější modely v kratším čase.

Celkově lze říci, že strojové učení může být velmi náročné, zejména pokud pracujeme s velkými datovými sadami nebo složitými modely. Nicméně, díky vývoji technologií a výpočetních prostředků, se náročnost strojového učení postupně snižuje a umožňuje nám trénovat stále složitější a výkonnější modely

2.2 Strojové učení

Existuje několik druhů strojového učení, zahrnující:

Supervizované učení:

Tento druh učení se používá, když máme k dispozici trénovací data, která obsahují správné odpovědi. Algoritmus strojového učení se učí na základě těchto dat a snaží se nalézt funkci, která dokáže předpovídat správné odpovědi pro nová data.

Nesupervizované učení:

Tento druh učení se používá, když nemáme k dispozici trénovací data s označenými správnými odpověďmi. Algoritmus strojového učení se učí na základě struktury dat a snaží se najít skryté vzory nebo struktury.

Polosupervizované učení:

Tento druh učení kombinuje prvky supervizovaného a nesupervizovaného učení. Máme k dispozici několik trénovacích dat s označenými odpověďmi a několik bez označení. Algoritmus se snaží využít všechny dostupné informace k naučení se co nejlepší funkce.

Zpětnovazební učení:

Tento druh učení se používá v dynamických prostředích, kde se algoritmus strojového učení učí na základě zpětné vazby, kterou získává z prostředí. Algoritmus se snaží maximalizovat nějakou odměnu nebo minimalizovat chybu na základě zpětné vazby.

Posílené učení:

Tento druh učení se používá v dynamických prostředích, kde se agent učí, jaké akce mají být provedeny na základě stavu prostředí a odměn, které jsou uděleny za provedení určitých akcí. Algoritmus strojového učení se snaží naučit agenta takovou strategii, která maximalizuje celkovou odměnu.

2.3 Proč zrovna Python

Python je jeden z nejpoužívanějších programovacích jazyků pro vývoj umělé inteligence (AI). Je to díky mnoha knihovnám a frameworkům, které jsou k dispozici a usnadňují vývoj AI.

Níže jsou uvedeny některé z nejznámějších knihoven a frameworků pro AI v Pythonu:

Keras: Je to vysokoúrovňová knihovna pro vývoj neuronových sítí, která umožňuje vývoj modelů v několika řádcích kódu. Keras lze použít jako nadstavbu pro TensorFlow.

scikit-learn: Je to knihovna pro strojové učení v Pythonu, která obsahuje mnoho algoritmů pro klasifikaci, regresi, shlukování a další úlohy.

Natural Language Toolkit (NLTK): Je to knihovna pro zpracování přirozeného jazyka (NLP), která obsahuje nástroje pro tokenizaci, lemmatizaci, kategorizaci a další úlohy v NLP.

Tyto knihovny a frameworky poskytují vývojářům a výzkumníkům AI mnoho nástrojů pro vytváření a nasazení modelů AI. Python také poskytuje snadnou syntaxi, která umožňuje rychlý a jednoduchý vývoj.

TensorFlow: Je to open-source framework pro vývoj strojového učení, který byl vytvořený firmou Google. TensorFlow umožňuje vývoj neuronových sítí, hladké optimalizace, a distribuované výpočty.

PyTorch: Je to další open-source framework pro vývoj strojového učení, který byl vytvořený firmou Facebook. PyTorch je snadno použitelný a umožňuje vývoj komplexních modelů s vysokou úrovní abstrakce.

3 Praktická část

3.1 Využití umělé inteligence

Bezpečnost na silnicích:

Systém rozpoznávání dopravních značek může pomoci řidičům dodržovat příslušná pravidla a zlepšit bezpečnost na silnicích. Například pokud řidič projede značku stop, systém mu může okamžitě vyslat upozornění, aby zabránil nehodě.

Snížení počtu dopravních přestupků:

Rozpoznávání dopravních značek může také snížit počet dopravních přestupků. Například když systém rozpozná, že řidič překročil rychlost, může mu být automaticky vystaveno pokutové řízení.

Podpora autonomního řízení:

Systémy rozpoznávání dopravních značek jsou také klíčovými prvky pro autonomní řízení vozidel. Vozidla vybavená touto technologií mohou být schopna samostatně reagovat na dopravní značky, což zlepší bezpečnost a plynulost provozu.

Lepší správa dopravních toků:

Rozpoznávání dopravních značek může také pomoci správcům dopravních sítí zlepšit řízení dopravních toků. Systémy rozpoznávání dopravních značek mohou být propojeny s centrálním řídicím systémem, což umožní správcům dopravních sítí získat aktuální informace o dopravní situaci a lépe řídit dopravní toky.

Zlepšení mobility pro osoby se zrakovým postižením:

Technologie rozpoznávání dopravních značek může být také využita pro zlepšení mobility pro osoby se zrakovým postižením. Například aplikace rozpoznávání dopravních značek může být integrována s hlasovým asistentem, který by poskytoval informace o dopravních značkách osobám se zrakovým postižením.

3.2 Využití Pythonu

Následující jsou některé způsoby, jakými lze v Pythonu využít AI:

Strojové učení:

Python je velmi populární pro vývoj modelů strojového učení díky knihovně scikit-learn, která poskytuje mnoho algoritmů pro klasifikaci, regresi a shlukování.

Hluboké učení:

Knihovny jako TensorFlow, Keras a PyTorch jsou běžně používány pro vývoj modelů hlubokého učení, jako jsou neuronové sítě.

Přírodní jazyk:

Pro zpracování přirozeného jazyka (NLP) lze použít knihovny jako NLTK, Spacy nebo Gensim.

Computer Vision:

Knihovna OpenCV je běžně používána pro zpracování obrazu a rozpoznávání objektů.

Robotika:

Python je také často používán pro programování robotů, jako je například Robot Operating System (ROS).

Chatboty:

Python se také používá pro vývoj chatbotů a dalších aplikací pro lidskou interakci.

Zkrátka, Python poskytuje širokou škálu nástrojů a knihoven pro vývoj různých typů umělé inteligence, což umožňuje rychlý a efektivní vývoj AI aplikací.

3.3 Tvorba aplikace

Vytvoření aplikace pro rozpoznávání dopravních značek pomocí umělé inteligence v Pythonu s využitím haarcascade vyžaduje následující kroky:

Instalace potřebných knihoven: Pro rozpoznávání dopravních značek pomocí haarcascade v Pythonu budeme potřebovat následující knihovny: OpenCV, NumPy, Matplotlib a os. Tyto knihovny lze nainstalovat pomocí nástroje pip.

Příprava trénovacích dat: K vytvoření modelu pro rozpoznávání dopravních značek budeme potřebovat trénovací data. Tyto data mohou být stáhnuté z veřejných databází, jako je například databáze German Traffic Sign Recognition Benchmark (GTSRB).

Trénování modelu: Pro trénování modelu můžeme použít knihovnu OpenCV a haarcascade klasifikátor. Tento klasifikátor se používá k detekci objektů, jako jsou například značky.

Vytvoření aplikace pro rozpoznávání značek: Pro vytvoření aplikace můžeme využít knihovny OpenCV, NumPy a Matplotlib. Aplikace by měla umožnit uživateli nahrát obrázek s dopravní značkou a následně aplikovat na něj naučený klasifikátor, aby byla značka rozpoznána.

3.4 Využití při komunikaci

3.4.1 Chatboti

Umělá inteligence může být velmi užitečná pro chatboty, protože může pomoci zlepšit schopnost chatbotu porozumět uživatelskému dotazu a poskytnout relevantní odpověď. Některé způsoby, jak umělá inteligence může být použita u chatbotů, jsou:

Natural Language Processing (NLP): NLP je obor umělé inteligence, který se zabývá porozuměním lidské řeči a její interpretací strojem. Použití NLP umožňuje chatbotům porozumět uživatelským dotazům v přirozené řeči a poskytnout odpovědi na základě toho, co bylo řečeno.

Machine Learning: Machine Learning umožňuje chatbotům učit se z předchozích interakcí s uživateli a zlepšovat své schopnosti odpovídat na dotazy. Chatbot může být naprogramován tak, aby se učil z předchozích odpovědí a mohl tak poskytovat lepší a více personalizované odpovědi.

Natural Language Generation (NLG): NLG je další oblast umělé inteligence, která umožňuje strojům generovat přirozené jazykové výstupy. Použití NLG umožňuje chatbotům generovat odpovědi na základě vstupních dat a poskytovat tak uživatelům personalizované odpovědi.

Dialog Management: Dialog Management je oblast umělé inteligence, která se zabývá řízením dialogu mezi uživatelem a strojem. Použití Dialog Management umožňuje chatbotům lépe řídit interakce s uživatelem a poskytovat lepší odpovědi na základě kontextu.

Celkově lze říci, že umělá inteligence může výrazně zlepšit schopnost chatbotu odpovídat na uživatelské dotazy a poskytnout uživatelům lepší zážitek.

3.4.2 Automaticky generované odpovědi

Použití umělé inteligence pro komunikaci může být velmi užitečné pro mnoho různých oblastí, jako jsou například zákaznická podpora, e-commerce, zdravotnictví, vzdělávání a mnoho dalších. Umělá inteligence může pomoci zlepšit efektivitu a rychlost komunikace a také poskytnout personalizované řešení pro každého uživatele.

Použití umělé inteligence pro komunikaci také zvyšuje dostupnost služeb pro uživatele, protože umožňuje interakci 24 hodin denně, 7 dní v týdnu bez ohledu na fyzickou přítomnost člověka. Nicméně, při použití umělé inteligence pro komunikaci je důležité zohlednit etické a bezpečnostní aspekty a zajistit, aby uživatelé měli kontrolu nad svými daty a soukromím.

3.4.3 Hlasoví asistenti

Hlasoví asistenti jsou jednou z nejčastějších aplikací umělé inteligence. Tyto asistenti využívají UI k tomu, aby dokázaly rozpoznat a interpretovat lidskou řeč a poskytnout uživatelům různé služby a informace. Některé z konkrétních způsobů, jak mohou hlasoví asistenti využívat UI, jsou například:

Rozpoznání hlasu: Hlasoví asistenti využívají UI k tomu, aby mohli rozpoznat hlas uživatele a porozumět tomu, co říká. UI se používá k překladu zvukového signálu na text, který může být dále zpracován.

Přirozené zpracování jazyka (NLP): Hlasoví asistenti využívají UI k tomu, aby mohli porozumět jazyku a interpretovat lidské dotazy. UI používá slovníkové databáze a algoritmy strojového učení k rozpoznání významu slov a frází a poskytnutí odpovídajících odpovědí.

Personalizace: UI se také používá k personalizaci hlasových asistentů. Hlasoví asistenti mohou uchovávat informace o uživateli, jako jsou jejich preference a historie, a využít je k poskytnutí personalizovaných odpovědí.

Integrace: UI umožňuje hlasovým asistentům integraci s dalšími aplikacemi a službami. Asistenti mohou být integrováni s různými zařízeními, jako jsou chytré reproduktory nebo chytré telefony, a poskytovat uživatelům přístup k různým funkcím a službám.

Učení se: UI se také používá k trénování hlasových asistentů, aby mohli lépe porozumět jazyku a poskytnout lepší odpovědi. Hlasoví asistenti se učí na základě interakcí s uživateli a mohou být aktualizováni a vylepšováni na základě zpětné vazby.

Celkově lze říci, že UI je klíčovým prvkem pro fungování hlasových asistentů a poskytuje jim možnosti, jak lépe porozumět lidské řeči a poskytovat užitečné informace a služby.

3.4.4 Robot Operating System (ROS)

ROS je softwarový framework, který umožňuje programování a řízení robotů. UI může být využita v ROS k řešení různých úloh, například rozpoznávání překážek, navigaci, řízení pohybu robotů, sběru dat a dalších.

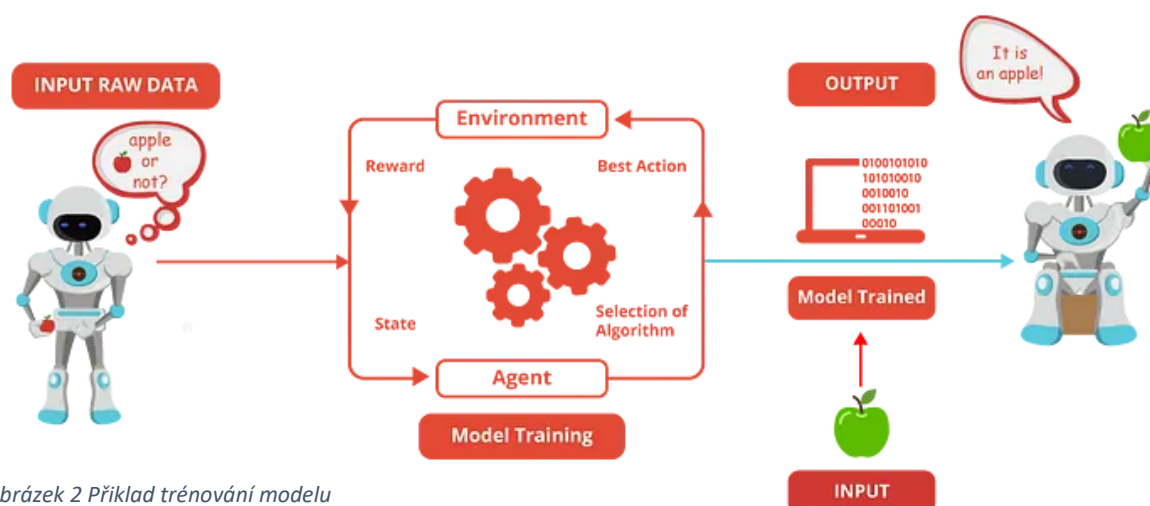
V oblasti rozpoznávání překážek může UI pomoci robotu rozpoznat překážky v jeho okolí pomocí kamer nebo senzorů a podle toho přizpůsobit svůj pohyb a směr. UI také může pomoci robotům přizpůsobit se různým podmínkám, jako jsou změny světelných podmínek nebo změny v terénu.

UI také umožňuje robotům učit se nové věci a zlepšovat své chování. Například může být použita pro rozpoznávání gest nebo pro učení se nových pohybů a chování. UI také umožňuje robotům interagovat s lidmi a přizpůsobit se jejich potřebám a přáním.

Využití UI v ROS může také pomoci zlepšit efektivitu a rychlost robota a zároveň zlepšit jeho schopnost přizpůsobit se různým prostředím a situacím. To může vést ke zlepšení výkonu robotů a zároveň k zvýšení jejich použitelnosti a přizpůsobivosti.

V oblasti robotického průmyslu se UI využívá k řešení různých úloh, jako je například kontrola kvality, sledování výrobních procesů, manipulace s materiály, balení a další. UI také umožňuje robotům spolupracovat s lidmi na výrobní lince a přizpůsobit se jejich potřebám a přáním.

Využití UI v ROS také umožňuje snadnou integraci různých senzorů a komponent, což usnadňuje programování a řízení robotů. Vývojáři tak mohou využívat širokou škálu nástrojů a technologií pro vývoj robotů, což zvyšuje jejich efektivitu a přesnost.



Obrázek 2 Příklad trénování modelu

3.4.5 Predikce textu

Automatické doplňování textu pomocí umělé inteligence je funkce, která umožňuje uživatelům snadno dokončit věty nebo fráze během psaní textu. Tato funkce funguje tak, že po každém zadaném slovu se na základě kontextu a statistického modelu předpovídají další slova, která by mohla uživatel pokračovat v psaní. Například, když uživatel napíše slovo "dobrý", automatické doplňování textu může navrhnout slova jako "večer", "den" nebo "nápad".

Tato funkce je obvykle implementována pomocí algoritmů strojového učení a hlubokého učení. Tyto algoritmy se učí na velkých datasetech textů a učí se předpovídat další slova na základě kontextu.

Automatické doplňování textu je užitečné pro různé oblasti, jako jsou například sociální sítě, e-maily, zprávy a další. Pomáhá uživatelům psát rychleji a efektivněji a může snížit počet chyb v textu.

Nicméně, automatické doplňování textu pomocí umělé inteligence může také vyvolávat obavy ohledně ochrany soukromí. Pokud jsou tyto funkce aktivovány, uživatelé mohou sdílet informace, které by jinak nebyly sdíleny. Proto je důležité, aby vývojáři dbali na zabezpečení a ochranu osobních údajů uživatelů.

3.4.6 Webové vyhledávače a UI

Umělá inteligence se v poslední době stává stále důležitější součástí webových vyhledávačů. Toto využití UI umožňuje vyhledávačům lépe porozumět tomu, co uživatelé hledají, a nabídnout jim přesnější výsledky.

UI využívá algoritmy strojového učení, které umožňují webovým vyhledávačům lépe porozumět uživatelskému záměru, tedy tomu, co uživatel přesně hledá. Například vyhledávač může použít UI k analýze uživatelských dotazů a odhadu toho, co uživatel hledá, i když používá neformální jazyk nebo nepřesné výrazy.

UI také umožňuje webovým vyhledávačům porozumět vztahům mezi jednotlivými slovy a větami, což může vést k lepším výsledkům vyhledávání. Například vyhledávač může porozumět, že slova jako "auto" a "vozidlo" jsou si blízká a zobrazit uživateli výsledky, které obsahují obě slova.

Dalším způsobem, jak UI vylepšuje webové vyhledávače, je použití algoritmů strojového učení k určení, které stránky jsou nejrelevantnější pro daný dotaz. Tyto algoritmy mohou zohlednit řadu faktorů, jako je kvalita obsahu stránky, autorita stránky a způsob, jakým se odkazuje na danou stránku z jiných stránek.

UI může být také využito k personalizaci výsledků vyhledávání pro každého uživatele. To znamená, že vyhledávač může využít algoritmy strojového učení k analýze uživatelského chování a zobrazovat uživateli výsledky, které jsou pro něj nejrelevantnější.

Využití UI v webových vyhledávačích tak výrazně zlepšuje kvalitu výsledků vyhledávání a umožňuje uživatelům snadněji a rychleji najít to, co hledají.

3.5 Ukázka kódu v jazyce Python

```
import cv2
import numpy as np
import matplotlib.pyplot as plt
import os
# Nahrání trénovacích dat
data_dir = "path/to/training/data"
training_data = []
for img in os.listdir(data_dir):
    img_array = cv2.imread(os.path.join(data_dir, img), cv2.IMREAD_GRAYSCALE)
    training_data.append(img_array)
# Trénování klasifikátoru
classifier = cv2.CascadeClassifier('path/to/haar/cascade/classifier.xml')
# Funkce pro detekci značek
def detect_signs(img):
    signs = classifier.detectMultiScale(img, scaleFactor=1.1, minNeighbors=4, minSize=(20, 20))
    return signs
# Nahrání testovacího obrázku a aplikace detekce značek
img = cv2.imread("path/to/test/image")
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
signs = detect_signs(gray)
# Vykreslení nalezených značek a zobrazení výsledku
for (x,y,w,h) in signs:
    cv2.rectangle(img, (x,y), (x+w, y+h), (0, 255, 0), 2)
plt.imshow(cv2.cvtColor(img, cv2.COLOR_BGR2RGB))
plt.show()
```

3.6 Haar kaskády

Haar kaskády jsou v informatice používány pro detekci objektů v obraze. Jedná se o algoritmus založený na vlnkových funkcích, který lze trénovat k detekci určitého objektu (např. obličeje) v obraze.

Zde je ukázka Haar kaskád použitých pro detekci obličejů:

Nejprve je vytvořena trénovací množina obrázků obsahujících obličeje a neobličeje.

Poté se trénuje Haar kaskáda na těchto obrázcích, což umožňuje algoritmu naučit se rozpoznávat, co je obličej a co není.

Po dokončení trénování se Haar kaskáda aplikuje na testovací obrázky a vyhledává obličeje v obraze.

Když Haar kaskáda identifikuje obličej v obraze, tak na něj umístí tzv. bounding box (omezení boxu), kterým označí nalezený objekt.

Nakonec se vyhodnotí výsledky, zda Haar kaskáda správně identifikovala všechny obličeje v obraze.

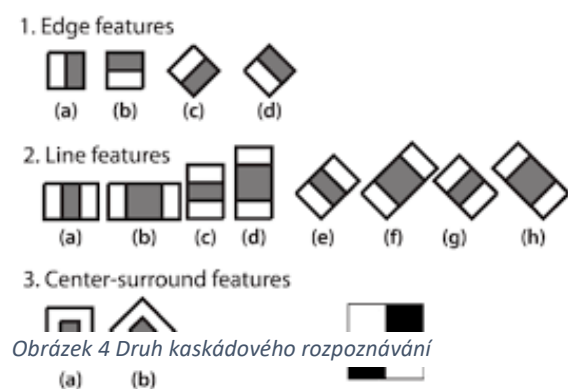
Haar kaskády jsou využívány nejen pro detekci obličejů, ale také pro detekci dalších objektů, jako jsou auta, zvířata, nebo i identifikaci zájmových oblastí v obraze.

Kaskády jsou složeny z mnoha malých kaskád, které se skládají z Haarových filtrů. Haarové filtry jsou obdélníkové masky, které se používají k detekci určitých rysů v obraze.

Každá malá kaskáda se skládá ze tří nebo více Haarových filtrů, které jsou nasměrovány na různé úhly a jsou umístěny vedle sebe v jednom sloupci. Tyto filtry se používají k identifikaci různých rysů v obraze, jako jsou například okrajové čáry, změny intenzity nebo změny textury.

Celková kaskáda je pak vytvořena spojením několika těchto malých kaskád. Všechny tyto malé kaskády jsou trénovány na různých velikostech a umístěních objektů v obraze. Tímto způsobem Haar kaskáda může detekovat objekty v různých velikostech a na různých pozicích v obraze.

Celkový vzhled Haar kaskády závisí na počtu a velikosti jednotlivých kaskád a také na konkrétní aplikaci, pro kterou je Haar kaskáda vytvořena. Nicméně obecně jsou Haar kaskády většinou velmi kompaktní a skládají se z mnoha různých malých kaskád, které jsou optimalizovány pro danou aplikaci.



4 Závěr

Tento projekt se zaměřuje na využití umělé inteligence pro rozeznávání dopravních značek z obrazových dat. K tomuto účelu se používá knihovna cv2 v jazyce Python.

V první fázi projektu byly shromážděny trénovací data v podobě fotografií dopravních značek. Tyto fotografie byly předzpracovány, aby se zajistilo, že mají správnou velikost a kvalitu. Poté byly vytvořeny a natrénovány soubory Haarcascades, které jsou schopny rozpoznat dopravní značky na fotografii.

V druhé fázi projektu byly testovací data použity k otestování výkonu. Testovací data byla sestavena z fotografií dopravních značek, které nebyly součástí trénovacích dat. Tato fáze projektu umožnila rozeznávání optimalizovat a zlepšit schopnost rozpoznávat typy dopravních značek. Momentální přesnost rozeznávání se pohybuje kolem 80%. Stále hraje velkou roli kvalita poskytnutých fotografií.

V poslední fázi projektu byly vytvořeny aplikace pro rozpoznávání dopravních značek pomocí umělé inteligence a jednoduché GUI. Tato aplikace jsou zatím schopny přijímat fotografie uložené pouze na lokálním uložišti. Aplikace pak použijí natrénované kaskádové soubory k rozpoznání dopravní značky na fotografii.

Celkově je tento projekt úspěšným příkladem využití umělé inteligence pro rozpoznávání dopravních značek. Použití knihovny CV2 a jazyku Python umožňuje rychlý a efektivní vývoj aplikací, které jsou schopny pracovat s obrazovými daty.

Momentálně CV2 neumožňuje kompilaci tak aby program byl funkční, pracuji na opětovném uvedení do funkčního stavu. Po každém updatu CV2 přestává fungovat a objevují se nové chyby, momentálně je na stránkách stackoverflow otevřený ticket který má za cíl vyřešit problém s kompilací, ticket je přibližně rok neaktivní.

4.1 Aktuálnost tématu

Rozpoznávání dopravních značek pomocí umělé inteligence je aktuální a velmi zajímavé téma v oblasti strojového učení a počítačového vidění. Tato oblast má mnoho praktických využití, například pro zlepšení bezpečnosti na silnicích, optimalizaci řízení dopravy a lepší orientaci řidičů.

V současné době se výzkumníci a inženýři snaží vylepšovat algoritmy pro rozpoznávání dopravních značek, aby dosáhli co nejvyšší přesnosti a spolehlivosti. S tím souvisí také využívání různých metod strojového učení, jako jsou například neuronové sítě, rozhodovací stromy a podobně.

V praxi se rozpoznávání dopravních značek pomocí umělé inteligence používá například u rychlostních měřičů, které dokáží automaticky zaznamenávat rychlost vozidel a porovnávat ji s omezením daným dopravními značkami. Dále se tato technologie využívá u autonomních vozidel, která musí být schopna rozpoznávat a interpretovat dopravní značky, aby dokázala přizpůsobit svou jízdu aktuální situaci na silnici.

Zkrátka, rozpoznávání dopravních značek pomocí umělé inteligence má velký potenciál pro zlepšení bezpečnosti na silnicích a pro podporu autonomního řízení vozidel. Jedná se o aktuální téma, které stále láká mnoho výzkumníků a inženýrů k dalšímu vývoji a zdokonalování.

5 Citace

OpenCV: OpenCV modules. *OpenCV documentation index* [online]. Dostupné

z: <https://docs.opencv.org/4.7.0/>

3.11.2 Documentation. *302 Found* [online]. Copyright © [cit. 02.03.2023]. Dostupné

z: <https://docs.python.org/3/>

Advanced Computer Vision with Python - Full Course - YouTube. *YouTube* [online]. Copyright © 2023

Google LLC [cit. 02.03.2023]. Dostupné z: https://www.youtube.com/watch?v=01sAkU_NvOY

Image Processing with OpenCV and Python - YouTube. *YouTube* [online]. Copyright © 2023 Google

LLC [cit. 02.03.2023]. Dostupné z: <https://www.youtube.com/watch?v=kSqxn6zGE0c>

OpenCV Python Tutorial #7 - Template Matching (Object Detection) - YouTube. *YouTube* [online].

Copyright © 2023 Google LLC [cit. 02.03.2023]. Dostupné z: <https://www.youtube.com/watch?v=T-0lZWYWE9Y>

OpenCV-Python Tutorials — OpenCV Documentation. *Site not found · GitHub Pages* [online].

Copyright © Copyright 1999 [cit. 02.03.2023]. Dostupné z: https://vovkos.github.io/doxyrest-showcase/opencv/sphinx_rtd_theme/page_tutorial_py_root.html

The Jupyter Notebook — Jupyter Notebook 6.5.2 documentation. [online]. Dostupné

z: <https://jupyter-notebook.readthedocs.io/en/stable/>

OpenCV: Cascade Classifier. OpenCV documentation index [online]. Dostupné z:

https://docs.opencv.org/3.4/db/d28/tutorial_cascade_classifier.html

Haar Cascades for Face Detection | Kaggle. Kaggle: Your Machine Learning and Data Science

Community [online]. Copyright © Original Authors [cit. 05.03.2023]. Dostupné z:

<https://www.kaggle.com/datasets/gpreda/haar-cascades-for-face-detection>

What is machine learning [online]. Copyright © Original Authors [cit. 05.03.2023] Dostupné z:

<https://becominghuman.ai/what-is-machine-learning-d292114cc6ce>

Training Haar Cascades. memememe [online]. Dostupné z:

<https://mememememememe.me/post/training-haar-cascades/>

Cascade Classifier. APMonitor Optimization Suite [online]. Dostupné z:

<https://apmonitor.com/pds/index.php/Main/CascadeClassifier>

6 Seznam ilustrací

Obrázek 1 Rozdělení stojového učení	3
Obrázek 2 Příklad trénování modelu	13
Obrázek 3 Příklad rozpoznávání obličeje	18
Obrázek 4 Druh kaskádového rozpoznávání	18

6.1 Čestné prohlášení

Prohlašuji, že jsem maturitní práci na téma „Rozpoznávání dopravních značek pomocí umělé inteligence“ vypracoval samostatně a s použitím uvedené literatury a pramenů.

V Mikulášovicích ., dne 3.3. 2023

Marek Charvát

6.2 Poděkování

V této části bych chtěl poděkovat svému kontultantovi kterým byl Mgr. Petr Kozák za veškeré konzultace, za přínosné nápady a pomoc při řešení klíčových problémů. Také chci poděkovat svému oponentovi kterým byl Tomáš Polák za konstruktivní kritiku a nápady na možné zlepšení.