

# **Praktické aplikace umělé inteligence pomocí cloudových služeb**

Dominik Zdráhala

---

Bakalářská práce  
2019



Univerzita Tomáše Bati ve Zlíně  
Fakulta aplikované informatiky

---

Univerzita Tomáše Bati ve Zlíně  
Fakulta aplikované informatiky  
akademický rok: 2018/2019

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Dominik Zdráhala**  
Osobní číslo: **A15677**  
Studijní program: **B3902 Inženýrská informatika**  
Studijní obor: **Informační a řídicí technologie**  
Forma studia: **prezenční**

Téma práce: **Praktické aplikace umělé inteligence pomocí cloudových služeb**  
Téma anglicky: **Practical Applications of Artificial Intelligence through Cloud Services**

Zásady pro vypracování:

1. Vypracujte literární rešerši na dané téma.
2. Analyzujte dostupné cloudové služby pro práci s umělou inteligencí, včetně jejich srovnání a popisu.
3. Vytvořte vzorové příklady (praktické ukázky) využívající vybrané cloudové služby.
4. Navrhněte a vytvořte aplikaci pro analýzu objektů na fotografii s využitím jedné z dostupných cloudových služeb.
5. Zhodnoťte Vámi vytvořenou aplikaci a navrhněte její praktické využití.

Rozsah bakalářské práce:

Rozsah příloh:

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

1. **MARÍK, Vladimír, Jiří LAŽANSKÝ a Olga ŠTĚPÁNKOVÁ. Umělá inteligence 6. Academia, 2013. ISBN 978-80-200-2276-9**
2. **LAŽANSKÝ, Jiří, Vladimír MARÍK a Olga ŠTĚPÁNKOVÁ. Umělá inteligence 5. Academia, 2007. ISBN 978-80-200-1470-2.**
3. **Amazon Rekognition Developer Guide. AWS Documentation [online]. ?2018 [cit. 2018-11-26]. Dostupné z: <https://docs.aws.amazon.com/rekognition/latest/dg/rekognition-dg.pdf>**
4. **Co je počítačové zpracování obrazu?. Docs.microsoft.com/cs-cz/azure/ [online]. [cit. 2018-11-26]. Dostupné z: <https://docs.microsoft.com/cs-cz/azure/cognitive-services/computer-vision/home>**
5. **Text detection samples. Cloud.google.com [online]. [cit. 2018-11-26]. Dostupné z: <https://cloud.google.com/vision/docs/detecting-text>**
6. **Cloud Speech-to-Text Basics. Cloud.google.com [online]. [cit. 2018-11-26]. Dostupné z: <https://cloud.google.com/speech-to-text/docs/basics>**

Vedoucí bakalářské práce: **Ing. Milan Navrátil, Ph.D.**  
Ústav elektroniky a měření

Datum zadání bakalářské práce: **21. prosince 2018**

Termín odevzdání bakalářské práce: **15. května 2019**

Ve Zlíně dne 21. prosince 2018

doc. Mgr. Milan Adámek, Ph.D.  
*děkan*



prof. Ing. Vladimír Vašek, CSc.  
*ředitel ústavu*

### Prohlašuji, že

- beru na vědomí, že odevzdáním bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – diplomovou/bakalářskou práci nebo poskytnout licenci k jejímu využití jen připouští-li tak licenční smlouva uzavřená mezi mnou a Univerzitou Tomáše Bati ve Zlíně s tím, že vyrovnání případného přiměřeného příspěvku na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše) bude rovněž předmětem této licenční smlouvy;
- beru na vědomí, že pokud bylo k vypracování bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

### Prohlašuji,

- že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně, dne 14. 5. 2019

.....DOMINIK ZPŘÁNÁL.....  
podpis diplomanta

## **ABSTRAKT**

Cílem bakalářské práce je analýza a srovnání dostupných cloudových služeb pro práci s umělou inteligencí. Práce se bude dále zabývat popisem možností práce s těmito systémy, popisem nabízených služeb, srovnáním a praktickými ukázkami. V praktické části student vytvoří vzorové příklady použití těchto služeb. Součástí praktické části bude také vytvoření aplikace k rozpoznání objektu na fotce s využitím jedné z dostupných služeb.

Klíčová slova: cloud, služba, Microsoft, Google, Amazon, API

## **ABSTRACT**

The aim of the bachelor thesis is to analyze and compare available cloud services which are working with artificial intelligence. Moreover, the paper will describe possibilities how to work with these systems, the services offered, their comparison and practical demonstration. In the practical part will be created examples of these services. This part includes creation of an application to recognize an object on a photograph using one of the available services.

Keywords: cloud, services, Microsoft, Google, Amazon, API

## Poděkování

Tímto bych chtěl srdečně poděkovat vedoucímu této bakalářské práce panu Ing. Milanu Navrátilovi, Ph.D. za čas, který mi věnoval při řešení mé práce. Dále bych rád poděkoval své rodině za podporu při studiu.

Prohlašuji, že odevzdaná verze bakalářské/diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

# OBSAH

<b>ÚVOD</b> .....	<b>10</b>
<b>I TEORETICKÁ ČÁST</b> .....	<b>11</b>
<b>1 VYUŽITÉ TECHNOLOGIE</b> .....	<b>12</b>
1.1 UMĚLÁ NEURONOVÁ SÍŤ .....	12
1.1.1 Umělý neuron .....	12
1.2 UMĚLÁ INTELIGENCE .....	14
1.3 REST API .....	15
1.4 CLOUD COMPUTING .....	16
<b>2 POPIS DOSTUPNÝCH CLOUDOVÝCH SLUŽEB</b> .....	<b>17</b>
2.1 MICROSOFT AZURE.....	17
2.1.1 Machine Learning Studio.....	18
2.1.2 Hlasové služby.....	19
2.2 GOOGLE CLOUD PLATFORM .....	20
2.2.1 Speech to Text .....	20
2.2.2 Cloud AutoML.....	21
2.2.3 DialogFlow.....	22
2.3 AMAZON WEB SERVICES.....	22
2.3.1 Amazon Forecast .....	22
2.3.2 Amazon Polly .....	22
2.3.3 Amazon Textract .....	23
<b>3 POPIS POUŽITÝCH SLUŽEB</b> .....	<b>24</b>
3.1 POČÍTAČOVÉ ZPRACOVÁNÍ OBRAZU .....	24
3.1.1 Společnost Google.....	24
3.1.1.1 Obecný popis rozhraní.....	24
3.1.1.2 Detekce log známých značek .....	26
3.1.1.3 Detekce tváře.....	27
3.1.1.4 Detekce objektů.....	28
3.1.1.5 Lokalizace obrázků .....	28
3.1.1.6 Vytvoření výstřižku .....	29
3.1.1.7 Detekce známých míst.....	29
3.1.1.8 Detekce textu.....	30
3.1.1.9 Detekce dominantní barvy .....	31
3.1.1.10 Detekce nevhodného obsahu .....	32
3.1.1.11 Nalezení obsahu na webu .....	32
3.1.1.12 Školení vlastních modelů.....	34
3.1.2 Společnost Microsoft.....	34
3.1.2.1 Obecný popis rozhraní.....	35
3.1.2.2 Popis obrázku v celých větách .....	35
3.1.2.3 Vytvoření miniatur.....	36
3.1.2.4 Označení obrázku.....	36
3.1.2.5 Obecný popis funkce Analyze image .....	37
3.1.2.6 Detekce známých míst.....	37

3.1.2.7	Detekce osobností .....	38
3.1.2.8	Určení kategorie.....	38
3.1.2.9	Detekce log známých značek .....	39
3.1.2.10	Detekce dominantní barvy.....	39
3.1.2.11	Detekce nevhodného obsahu .....	40
3.1.2.12	Detekce obličeje .....	40
3.1.2.13	Detekce typu obrázku.....	41
3.1.2.14	Detekce objektů .....	41
3.2	PŘEKLAD TEXTU .....	42
3.2.1	Společnosti Google.....	42
3.2.1.1	Obecný popis rozhraní.....	42
3.2.1.2	Překlad textu.....	43
3.2.1.3	Detekování jazyka.....	44
3.2.2	Společnosti Microsoft.....	44
3.2.2.1	Obecný popis rozhraní.....	44
3.2.2.2	Překlad textu.....	45
3.2.2.3	Převod mezi znakovými sadami .....	46
3.2.2.4	Detekce jazyka .....	46
<b>4</b>	<b>PRAKTICKÉ UKÁZKY VYBRANÝCH SLUŽEB .....</b>	<b>47</b>
4.1	POPIS PROGRAMOVÉHO VYBAVENÍ.....	47
4.2	POČÍTAČOVÉ ZPRACOVÁVÁNÍ OBRAZU .....	48
4.2.1	Popis obslužných metod.....	48
4.2.2	Detekce objektů .....	49
4.2.3	Popis obrázku v celých větách.....	51
4.2.4	Dominantní barvy na obrázku.....	53
4.2.5	Školení vlastního modelu .....	55
4.3	PŘEKLAD TEXTU .....	60
4.3.1	Analýza výsledků a srovnání služeb.....	63
<b>II</b>	<b>PRAKTICKÁ ČÁST .....</b>	<b>65</b>
<b>5</b>	<b>POPIS PROGRAMU PRO DETEKCI AUTOMOBILU .....</b>	<b>67</b>
5.1	VÝBĚR VHODNÉ SLUŽBY .....	68
5.2	DETEKCE STÁTNÍ POZNÁVACÍ ZNAČKY .....	69
<b>6</b>	<b>VYHODNOCENÍ VYTVOŘENÉ APLIKACE .....</b>	<b>74</b>
6.1	DETEKCE AUTOMOBILU NA OBRÁZKU .....	74
6.2	DETEKCE STÁTNÍ POZNÁVACÍ ZNAČKY .....	76
6.3	CELKOVÉ ZHODNOCENÍ A NÁVRH REÁLNÉHO VYUŽITÍ.....	79
	<b>ZÁVĚR .....</b>	<b>81</b>
	<b>SEZNAM POUŽITÉ LITERATURY .....</b>	<b>82</b>
	<b>SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK .....</b>	<b>87</b>
	<b>SEZNAM GRAFŮ .....</b>	<b>88</b>
	<b>SEZNAM OBRÁZKŮ .....</b>	<b>89</b>
	<b>SEZNAM TABULEK.....</b>	<b>90</b>



<b>SEZNAM PŘÍLOH.....</b>	<b>91</b>
---------------------------	-----------

## ÚVOD

Tato bakalářská práce se zabývá popisem, využitím a analýzou získaných dat služeb využívající umělou inteligenci. Nejdříve práce zaobírá pojmy, jako je neuronová síť, umělá inteligence, REST API a cloud computing. Druhá část práce se zabývá obecným popisem dostupných služeb společnosti Google, Microsoft a Amazon. Je uveden jejich účel a vlastnosti. Poté jsou vybrány služby pro počítačové zpracování obrazu a překlad společnosti Google a Microsoft, které jsou podrobně analyzovány a využity pomocí nabízeného rozhraní REST API. Následně jsou u jednotlivých služeb a jejich dílčích funkcí získaná data analyzována. Na základě těchto dat jsou služby srovnány.

Praktická část této práce se nabývá popisem vytvořené aplikace pro detekci automobilů na obrázku a případné detekci státní poznávací značky a jejího skrytí. Je vybrána nejvhodnější služba na základě získaných dat. Následně je program podrobně popsán a jeho reálná funkce je v několika testech ověřena. Poslední část práce je věnována návrhu reálného využití této aplikace.

## **I. TEORETICKÁ ČÁST**

# 1 VYUŽITÉ TECHNOLOGIE

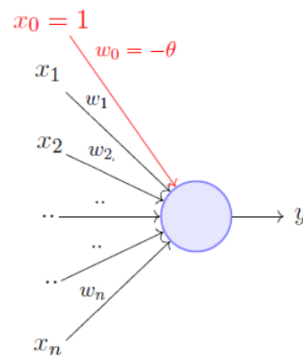
## 1.1 Umělá neuronová síť

Umělé neuronové sítě lze definovat jako soubor neuronů, jež jsou mezi sebou propojeny. Tyto spojení mezi neurony jsou ohodnoceny váhami definujícími důležitost daného spoje. Každé spojení mezi neurony je jednosměrné. Důležitou vlastností neuronových sítí je jejich schopnost učit se na poskytnutých tréninkových datech. V tomto procesu učení nalézají sítě závislosti v tréninkových datech a tyto závislosti poté reprezentují pomocí vah. Takto naučená neuronová síť dokáže pomocí nově nabitých zkušeností správně reagovat i na vstupy, jež nebyli obsaženy v tréninkové sadě. [1]

### 1.1.1 Umělý neuron

První model umělého neuronu byl vytvořen v roce 1943 McCullochem a Pittsem. Tento model je složen ze tří hlavních částí a to vstupu, výstupu a funkční části neuronu. Vstupní část je složena ze vstupů do neuronu a k nim příslušných synaptických vah. Tyto váhy jednotlivé vstupy zvýhodňují nebo potlačují, a to na základě jejich velikosti. Druhou částí je část funkční. Tato část zpracuje vstupy a podle nich vygeneruje odezvu neuronu. Třetí částí je výstup, jež přivádí výstupní informaci na vstup dalších neuronů nebo na výstup neuronové sítě. [2][3]

Výkonná jednotka umělého neuronu je tvořena jednoduchou nelineární funkcí. Neuron pracuje následujícím způsobem. Vstupní hodnoty jsou vynásobeny příslušnými hodnotami vah a následně sečteny. Na tento součet je aplikována přenosová funkce a výsledná hodnota je přivedena na výstup neuronu. Neuron dále disponuje tzv. prahem. V případě, že vážený součet vstupů do neuronu nepřesáhne tuto prahovou hodnotu, neuron není aktivován a jeho výstup zůstává nezměněn. Každý neuron, s výjimkou neuronů vstupní vrstvy sítě, má tzv. Bias. Ten síť při procesu učení nepatrně rozladí, aby učící algoritmus divergoval do globálního minima. [3]

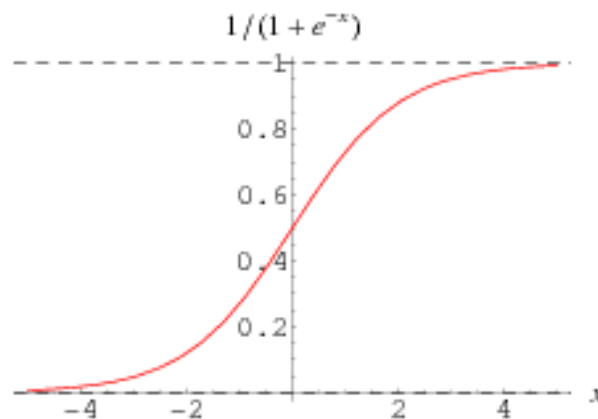


Obrázek 1 Perceptron [3]

Celkový potenciál neuronu je dán následující rovnicí:

$$\xi = \sum_{i=1}^n w_i x_i - \vartheta \quad (1)$$

Na potenciál reaguje neuron výstupním odezvou  $y = \sigma(\xi)$ , kde  $\sigma$  je definovaná nelineární funkce, z pravidla ve tvaru sigmoidy.



Obrázek 2 Sigmoidální funkce [4]

Jediný neuron není schopen vykonávat příliš složitou funkci. Síla neuronových sítí spočívá ve vzájemném propojení těchto neuronů. Neuronová síť umožňuje různě propojovat vstupy a výstupy neuronů, zvýhodnit nebo potlačit některé vstupy a minimalizovat vliv nesprávně fungujícího neuronu na celkový výsledek. U vícevrstvé struktury umělé neuronové sítě jsou neurony děleny do vrstev. Výstupy z n-té vrstvy jsou přivedeny na vstup každého neuronu ve vrstvě n+1. První vrstva se nazývá vstupní, jež přímá informace z okolí. Tyto informace zpracuje a přivede je na vstup každého neuronu následující vrstvy. Tato druhá vrstva se nazývá skrytá. Poslední vrstva je výstupní a hodnoty na jejím výstupu jsou odezvou celého

systemu na vstupní vzorky. Počet vrstev závisí na složitosti funkce, kterou má síť vykonat a také na zvoleném typu sítě. [3]

Neuronové sítě se dělí do dvou základních skupin, a to podle struktury. Jedná se o sítě s dopředným šířením signálu a sítě se zpětnou vazbou. V dnešní době se nejvíce používají neuronové sítě s dopředným šířením signálu, kde výstupy z jedné vrstvy jsou vedeny na vstup následující vrstvy. Poslední výstupní vrstva je také výstupní vrstvou celé sítě. Struktura sítí se zpětnou vazbou se dopředných sítí liší tak, že výstupy z vrstvy jsou vedeny zpět na vstup dané vrstvy. Taková struktura umožňuje provádět výpočty založené na iteračním procesu a řešit tak optimalizační úlohy. [5]

## 1.2 Umělá inteligence

Jedná se o odvětví informatiky zabývající se tvorbou strojů, které řeší komplexní úlohy, mezi něž patří např. zpracování velkého objemu dat či obrazových materiálů. Jednoznačná definice umělé inteligence neexistuje z důvodu multioborového využití. V historii bylo však vzneseno mnoho definic. Mezi nejznámější definici patří Turingův test. Tuto definici vznesl v roce 1950 Alan Turing ve svém článku *Computing machinery and intelligence*. Tento test předpokládá, že systém je možné považovat za inteligentní, pokud jeho lingvistický projev nelze rozeznat od lidského. Na tuto definici však John Searle v roce 1980 vznesl protiargument nazvaný jako argument čínské pokoje. Tento argument dokazuje že stroj, jež je schopen smysluplné odpovědi na otázku nutně nemusí být považován za inteligentní. V tomto myšlenkovém experimentu se předpokládá pokoj, ve kterém se nachází veškeré smysluplné věty v čínšském jazyce. V pokoji se nachází člověk, který však čínštinu neovládá. Při vznesení otázky v psané formě je člověk v pokoji hypoteticky schopen nalézt dostatek materiálu k tomu, aby byl schopen z kontextu otázky a vizuální podoby znaků odvodit odpověď. Vnější pozorovatel by si mohl myslet, že tato odpověď byla vytvořena porozuměním zadané otázky a vytvořením odpovědi, tedy inteligentním chováním. Avšak v tomto případě by mohl být člověk nahrazen strojem řízeným předem vytvořeným algoritmem, jež by na otázky odpovídal. [6][7]

Za další definici umělé inteligence se zasloužil Marvin Minsky. Tato definice říká, že umělá inteligence je vědní obor o vytvoření systému, který řeší zadané problémy takovým způsobem, že kdyby tento problém řešil člověk považovali bychom to za projev jeho inteligence.

S podobnou definicí přišel také Elaine Rich, podle něhož se umělá inteligence zabývá tím, jak počítačově řešit problémy, které dnes zatím lépe zvládají lidé. [8]

### 1.3 REST API

Roy Fielding v roce 2000 uvedl Representational State Transfer jinými slovy REST, ve své dizertační práci. Jedná se o architekturu orientovanou datově. Jednou z hlavních výhod této architektury je, že používá otevřené standardy a nezavádí tedy žádné specifické implementace. Je možné tedy vytvořit server pomocí např. technologie .NET Core, jež poskytuje data pomocí rozhraní REST API a tyto data distribuovat různým systémům, které dokáží generovat a zpracovat HTTP požadavky napříč programovacími jazyky. Odpovědi jsou posílány v počítačově čitelném formátu JSON (Javascript Object Notation) nebo XML (eXtensible Markup Language). Důležitým aspektem funkčnosti tohoto protokolu jsou bezstavové požadavky. Znamená to, že jednotlivé dotazy na službu spolu nesouvisí a můžou přicházet v libovolném pořadí. REST protokol implementuje čtyři základní operace s daty CRUD (CREATE, READ, UPDATE, DELETE), pomocí metod protokolu HTTP. [9]

- **POST (CREATE)**

Metoda sloužící k vytvoření nových zdrojů na definované URI. Požadavek je tvořen tělem nesoucím informace o vytvořeném zdroji. Tato metoda je využitelná i v případech, kdy je nežádoucí předávat informace pomocí parametrů URI. [9]

- **GET (READ)**

Jedná se o základní metodu HTTP protokolu. Pomocí této metody je možné získat data ze zdroje. [9]

- **PUT (UPDATE)**

Jedná se o metodu z jejíž pomocí je možné vytvořit či nahradit zdroj na příslušné URI. Podobně jako u metody POST jsou v těle tohoto požadavku umístěny informace o konkrétním zdroji. [9]

- **DELETE**

Jedná se o metodu, pomocí níž je možné smazat zdroj na určené URI. [9]

## 1.4 Cloud computing

Jedná se o systémy poskytující služby, zdroje a aplikace, které jsou uloženy na serverech, ke kterým je možno přistupovat prostřednictvím webových prohlížečů nebo pomocí dostupných rozhraní API. Nabízejí možnosti pro využití nejrůznějších služeb, aplikací, operačních systémů či datových uložišť. Popis cloud computing lze pojmově dvěma způsoby. Distribuční model jej popisuje podle toho, jaká služba je poskytována a model nasazení popisuje cloud computing z pohledu způsobu poskytnutí služeb. Cloud computing lze rozdělit do tří vrstev, které jsou Aplikace (SaaS), platforma (PaaS), infrastruktura (IaaS). [10]

- **Infrastructure as a Service (IaaS)**

V tomto případě je jako služba poskytnuta infrastruktura. Může se jednat o pronájem serveru, uložště či výpočetního času. Obrovskou výhodou je, že zákazník nemusí v tomto případě řešit správu hardwaru, případné poruchy či upgrade. Typicky se pak jedná o virtuální servery. [10]

- **Platform as a service (PaaS)**

Platforma jako služba znamená pronájem prostředků k vývoji aplikací. V tomto případě dostává zákazník od poskytovatele účinné nástroje pro správu databází, testování či specifické IDE pro vývoj. Výhodou tohoto řešení je urychlení vývoje díky integrovaným funkcím, jež platformy nabízejí. Velkým mínusem je pak fakt, že uživatelé jedné platformy ve většině případů nemohou své projekty migrovat jinam. [10]

- **SaaS (Software as a Service)**

V modelu softwaru jako platformy si zákazníci kupují přístup k určité aplikaci, nikoliv však její licenci. Zákazníci k těmto aplikacím přistupují typicky pomocí webového prohlížeče. Odpadá tedy nutnost instalace aplikací na lokálním zařízení. Další výhodou je možnost přístupu k aplikaci různých počítačů skrze internet. Zákazníci tedy nejsou limitováni pouze na jeden konkrétní počítač s nainstalovanou aplikací. Využitím tohoto řešení jsou značně sníženy pořizovací náklady a také odpadá nutnost pravidelné aktualizace ze strany zákazníka. [10]



## 2 POPIS DOSTUPNÝCH CLOUDOVÝCH SLUŽEB

V současné době je k dispozici mnoho cloudových řešení využívající umělou inteligenci. Mezi lídry v tomto odvětví patří společnosti Google, Microsoft a Amazon. Ty nabízí možnosti pro využití dobře odladěných modelů ke zpracování obrazu, práci s textem, práci se zvukem či analýze velkého množství dat. Služby se však neomezují pouze na tyto funkce, ale poskytují také nástroje pro uživatelské rozšíření nebo vylepšení určitých funkcí na základě poskytnutých dat. Komunikace s těmito službami probíhá skrze rozhraní API. Některé ze služeb však navíc nabízejí i webové uživatelské prostředí, ve kterém je možné služby využívat či spravovat poskytnutá data. Za služby u všech popisovaných společností je placeno, jen v případě jejich reálného využití. Většina služeb je také v omezené míře poskytována zdarma k vyzkoušení či odladění aplikací, jež tyto služby využívají.

### 2.1 Microsoft Azure

Jedná se o otevřenou cloudovou platformu, určenou pro rychlý vývoj, nasazení a správu aplikací v rámci datových center společnosti Microsoft. Díky této platformě je možné vytvářet aplikace pomocí široké škály programovacích jazyků, nástrojů nebo frameworků. Microsoft Azure u všech služeb využívající umělou inteligenci poskytuje měsíčně 99 % SLA což umožňuje vytvářet a provozovat vysoce dostupné aplikace. Platforma je v dnešní době dostupná ve 140 zemích světa ve 44 regionech. Jen v Evropě Microsoft disponuje sedmi datacentry, jež jsou umístěny v Německu či Francii. Díky tomu je zajištěn bezproblémový chod aplikací využívající jejich služeb. [11]

Společnost nabízí kognitivní funkce mezi něž patří analýza obrázků, analýza textu, překlad textu či detekce tváře. Společnost však zašla mnohem dál a nabízí svým zákazníkům možnosti výcviku vlastních modelů s využitím prakticky jakýchkoliv dat pomocí nástroje Machine Learning Studio či výcvik vlastních modelů k detekci a kategorizaci fotografií pomocí jednoduchého webového nástroje Custom Vision. Bezesporně největší výhodou je možnost využití těchto služeb bez žádné či základní znalosti problematiky neuronových sítí. Následující kapitoly se věnují popisu jednotlivých služeb. [11]

### 2.1.1 Machine Learning Studio

Jedná se o nástroj, z jehož pomocí je možné vytvářet, testovat a nasaďit řešení pro prediktivní analýzu dat. Machine Learning Studio nabízí interaktivní webové prostředí jehož základem je pracovní plátno. Nástroj nabízí datové sady a analytické modely, jež je možné na tomto plátně mezi sebou propojit a vytvářet tak experimenty. Datovou sadou jsou myšleny vstupní data, jak uživatelská tak ukázková, která jsou použita k vycvičení modelu pro prediktivní analýzu. Analytické modely představují sada kódů (algoritmů), které provádějí definované úkoly na základě vstupních dat do těchto modulů. Moduly jsou členěny do následujících kategorií. [12]

- **Moduly pro vstup a výstup dat**

Jedná se o moduly určené k manipulaci dat. Z jejich pomocí je možné provádět import či export dat v jednotlivých experimentech. Zdrojová data pro experiment mohou být získána z SQL databáze, XML souborů či z veřejné URL adresy. [13]

- **Moduly pro transformaci dat**

Pomocí těchto modulů je možné spojovat jednotlivé datové sady, odstraňovat chybějící řádky v těchto datech, volit sloupce tabulek pro použití v následujících blocích, měnit názvy sloupců či rozdělovat data. Do této skupiny spadají veškeré modely, jež jakýmkoliv způsobem manipulují s daty. [13]

- **Moduly s algoritmy strojového učení**

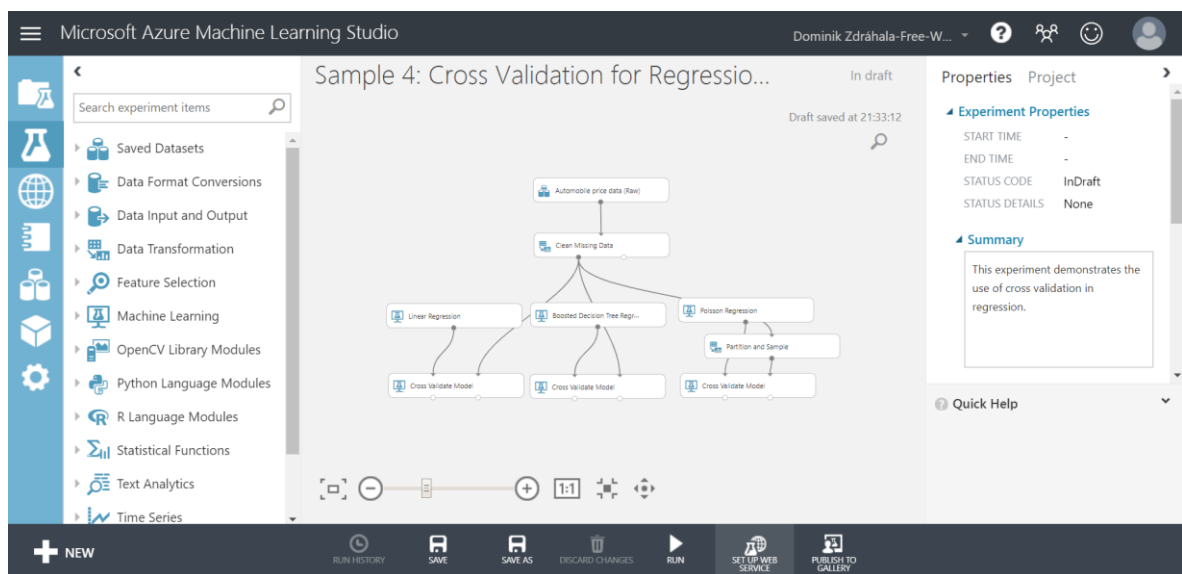
Jedná se o modely jež obsahují algoritmy strojového učení. Machine Learning Studio nabízí celou řadu těchto modelů mezi něž patří detekce anomálií, klasifikace, shlukování, regrese, analýza textu, klasifikace obrázků či analýza časových řad. Dalšími důležitými moduly jsou moduly určené pro trénink a vyhodnocení experimentu. [13]

- **Analytické moduly**

Tato skupina nabízí moduly určené k numerickým funkcím, jež souvisejí s vědou o datech. Mezi tyto funkce patří numerické operace jako zaokrouhlení či použití absolutní hodnoty, ale také statistické funkce pro výpočet korekčního a pravděpodobnostního skóre. [13]

Služba podporuje běžně používané datové typy jako integer, string, date atd. Výsledky experimentů je možné dále exportovat do CSV či XML souborů nebo experiment propojit s SQL databází k uložení výsledků. Nástroj je možné využít zdarma s následujícím omezením. Modelů na plátně jednoho experimentu nesmí být více jak 100, doba trvání experimentu

je omezena na jednu hodinu a místo v uložení je omezeno na 10 GB. V případě využití nástroje zdarma, není povoleno využití vyškolených modelů pomocí rozhraní API. V případě standardní úrovně platby je limit na využití jednoho experimentu navýšen na 7 dní, místo v uložení je neomezené a lze nástroj využít pomocí rozhraní API. Následující obrázek zobrazuje webové rozhraní tohoto nástroje. Jak je patrné jednotlivé bloky experimentu je možné vybírat z adresáře v levé části. Experimenty je možné archivovat či sdílet komunitě. Výsledky experimentů je možné zobrazit přímo ve webovém rozhraní nebo využít zmíněné metody pro uložení výsledků. [12][13]



Obrázek 3 Ukázka grafického prostředí Machine Learning Studia

### 2.1.2 Hlasové služby

Tento soubor služeb disponuje funkcemi pro převod textu na řeč, řeči na text a překladu řeči. Funkcí převodu řeči na text se rozumí schopnost služby převádět mluvené slovo do textové podoby v reálném čase. Funkce podporuje možnosti převodu řeči ze streamu nebo ze souboru ve formátu WAV, MP3, OGG a PCM nahraných se vzorkovací frekvencí 8 nebo 16 kHz. Funkce dále nabízí možnosti pro cenzuru vulgárních výrazů, zpracování interpunkcí nebo využití metadat ve kterých se nacházejí čas vyslovení jednotlivých slov. Podporuje přes dvacet jazyků jako španělštinu, angličtinu či japonštinu. [14]

Pomocí funkce převodu textu na řeč lze syntetizovat až 75 standardních hlasů ve více než 45 jazycích a 5 hlasů syntetizovaných pomocí neuronových sítí v pěti jazycích. Neuronové syntetizátory se od těch standardních liší hlavně v projevu a přirozenosti vytvořeného hlasu. Těmto hlasům lze dále jejich zabarvení (neutrální, veselé) nebo určit tempo vyslovování

jednotlivých vět. Výstupní hlas je možné získat v podobě streamu nebo jako zvukový soubor ve formátu MP3 či WAV. [14]

Překlad řeči je funkce s jejíž pomocí je možné v reálném čase provádět vícejazyčný překlad. Samotný překlad řeči je možné provádět dvěma způsoby a to pomocí statických překladů (SMT) nebo pomocí překladu s využitím neuronové sítě (NMT). Statický překlad využívá pokročilou statickou analýzu k nalezení nejlepšího možného překladu a to v kontextu několika vedlejších slov ve větě zatímco překlad pomocí neuronové sítě využívá kontext celé věty čímž vytváří přirozenější překlad. Díky tomu je v dnešní době překlad s využitím neuronových sítí populárnější. Funkce je schopna překladu řeči ze streamu nebo z WAV či PCM souborů. Maximálně je schopna přeložit nahrávku o maximální délce 90 minut. Výstup překladu může být text nebo syntetizovaný hlas. [14][15]

## 2.2 Google Cloud Platform

Platforma jež nabízí společnost Google je sada služeb cloud computing, využívající stejnou infrastrukturu, kterou společnost používá k distribuci svých produktů jako je YouTube či vyhledávač Google. Dále poskytuje celou řadu moderních cloudových služeb včetně služeb využívajících umělou inteligenci. Platforma poskytuje infrastrukturu jako službu a platformu jako službu. I u této společnosti platí, že za služby je placeno jen v případě jejich reálného využití. Společnost disponuje šestnácti datacentry, z toho v Evropě jich je umístěno pět a to v Německu či Irsku. [16]

### 2.2.1 Speech to Text

Jedná se o službu z jejich pomocí je možné převádět hlas na text. Služba podporuje až 120 jazyků včetně češtiny a slovenštiny. Nabízí také několik modelů vyvinutých pomocí neuronové sítě s pomocí kterých se převod provádí. Jsou to modely pro krátké hlasové klipy, typicky pak hlasové povely, model pro převod řeči z telefonních hovorů, model pro převod řeči z videa jež využívají YouTube pro tvorbu automatických titulek a výchozí model pro převod dlouhých textů. Služba je schopna převádět řeč ze streamu či zvukových souborů FLAC nebo WAV. Dále nabízí možnosti pro filtraci vulgárního obsahu, možnost poskytnutí alternativních prepisů a možnost poskytnout službě dodatečné informace o slovech a frázích v nahrávce. Převod textu na řeč je časově náročný proces. Obecně platí, že čas strávený převodem je polovinou času poskytnuté nahrávky. Služba dále dokáže detekovat slova v klikech

se vzorkovací frekvencí od 8 000 až 48 000 *Hz*. Hlasové nahrávky je službě možno poskytnout jako URL adresu či jako datový řetězen kódovaný v base64. V případě synchronního dávkového přepisu souboru je služba omezena na jednu minutu záznamu. Pro asynchronní dávkový přepis je služba schopna zpracovat až 480 minut záznamu. Služba nabízí prvních 60 minut přepisu zdarma. Za každých 15 sekund je pak služba zpoplatněna \$0,006. [17][43]

### 2.2.2 Cloud AutoML

Služba sloužící k vyškolení vlastních modelů za pomoci strojového učení pro porozumění přirozenému jazyku, zpracování obrázků a videa, překladu a zpracování uživatelský dat. Pro analýzu obrázků společnost Google nabízí služby AutoML Vision. Této službě lze poskytnout vlastní obrázky, tyto obrázky rozřadit do příslušných kategorií nebo na obrázcích označit objekt zájmu, vytrénovat vlastní model a použít jej k predikci vytvořených kategorií nebo nalezení definovaného objektu na obrázku. [18]

Pro vytvoření vlastních modelů pro překlad je určena služba AutoML Translate. Této službě jsou poskytnuty páry vět ve zdrojovém a cílovém jazyce. Služba pomocí těchto vět, se kterými pracuje nezávisle na sobě, vycvičí model pro překlad. Poté je možné tento model využít k překladu mnoha různých vět s podobným kontextem jako mají věty určené k výcviku. Služba AutoML Natural Language slouží ke klasifikaci obsahu do vlastních kategorií. Další možností této služby je definovat vlastní entity a následně tyto entity identifikovat v textu. Entitami je v tomto případě myšleno specifická slova, jména nebo časový údaj. V neposlední řadě tato služba nabízí možnost pro vycvičení modelu, jež dokáže predikovat citové zabarvení vět. Další možností je trénovat vlastní modely pro kategorizaci obsahu ve video podle předem definovaných štítků za pomoci dílčí funkce AutoML Intelligence Classification. Poslední dílčí funkcí AutoML Tables. Jedná se o funkci, které jsou poskytnuty data ve formě tabulky. Z této tabulky je následně vybrán sloupec určený k predikci. Na základě ostatních dat v tabulce služba vyškolí model s jehož pomocí je možné predikovat údaje ve zvoleném sloupci. [18][19]

### 2.2.3 DialogFlow

Jedná se o službu, z jejíž pomoci je možné vytvořit konverzačního bota. Tento bot je tvořen tzv. agenty. Tyto agenti jsou modely schopny porozumět přirozenému jazyku. Agentům jsou dále poskytnuty cvičné scénáře rozhovorů, na kterých jsou školeni nazývané záměry. Záměry definují uživatelskou otázku a její odpověď. Dalším dílčí částí záměru jsou entity. Pomocí entit je možné definovat data, která se mají z otázky uživatele vyseparovat. Tyto data mohou být různého typu jako například barva, jméno, datum atd. Po výškolení je tento agent schopen odpovídat otázky dle poskytnutých scénářů. Díky umění porozumět přirozenému textu však není omezen pouze na předem definované otázky, ale je schopen pochopit kontextu a zvolit odpovídající záměr. [20]

## 2.3 Amazon Web Services

Jedná se o platformu společnosti Amazon, která poskytuje cloud computing a na něj navázané služby. Společnosti disponuje 25 datacentry po celém světě z nichž je 6 umístěno v Evropě. Společnost také poskytuje možnosti pro využití umělé inteligence pomocí rozhraní API. Následující kapitoly se věnují popisu vybraných služeb této společnosti. [21]

### 2.3.1 Amazon Forecast

Jedná se o služby s jejíž pomocí je možné je možné provádět velice přesné prognózy na základě poskytnutých dat. Služba analyzuje poskytnutá data a s jejich pomocí dokáže předpovědět budoucí vývoj. Typy dat, které je službě možné poskytnout je nepřehledné množství. Ať už jsou službě poskytnuty data o počasí, aktuálních trendech v módě či ve finančnictví, služba v těchto datech dokáže nalézt spojitosti a předpovídat budoucí vývoj. Služba dále nabízí možnosti pro přeškolení vytvořených modelů s použitím nově získaných dat či vizualizaci výsledků prognóz spolu s reálnou situací. Správa vyškolených modelů a použitých dat je možná pomocí webového uživatelského prostředí. [22]

### 2.3.2 Amazon Polly

Jedná se o službu s jejíž pomocí je možné převést text na řeč. Na rozdíl od podobných služeb společností Google či Microsoft nedisponuje velkým počtem podporovaných jazyků, ale pouze základními jako angličtina, španělština či japonština. Mezi podporované jazyky se

bohužel neřadí čeština ani slovenština. Služba dále nabízí možnost pro zvolení pohlaví mluvčího. Umožňuje také získání finální hlavy v reálném čase v podobě audio streamu nebo jako soubor v běžném formátu (MP3, OGG, PCM). Také poskytuje možnost získání metadat s informacemi o tom, v jakém čase nahrávky se vyslovují jednotlivá slova a věty. Na základě těchto dat je možné v aplikacích vytvořit vizualizaci ve stylu karaoke. [23]

### 2.3.3 Amazon Textract

Jedná se o službu s jejíž pomocí je možné extrahovat texty a data s naskenovaných dokumentů. Neomezuje se pouze na přečtení prostých textů, ale nabízí možnosti extrakce textu z formulářových polí nebo tabulek. Služba dále nabízí možnosti pro rozpoznání párových záznamů. Tím jsou myšleny položky formuláře, které služba dokáže identifikovat a správně mezi sebou spárovat. Jedná se např. o formuláře z iniciály zákazníka, kde služba dokáže identifikovat řádek formuláře, třeba jméno a k tomuto klíči přiřadit vyplněnou hodnotu zákazníkem. Služba dále poskytuje informace o vrcholech rámečku, jež detekovaný text nebo tabulku ohraničuje. To usnadňuje případnou orientaci v původním dokumentu. Služba podporuje extrakci textů z obrázku formátu JPEG a PNG, které mají maximální velikost 5 MB nebo z dokumentů ve formátu PDF o maximální velikosti 500 MB. Dokument formátu PDF je dále omezen počtem stran. Maximální počet stran, jež služba dokáže analyzovat je 3 000. Text pro analýzu je v ideálním případě vodorovný, služba však dokáže detekovat i texty které jsou mírně potočené, a to maximálně o 10 % od svislé osy. [24][40]

### 3 POPIS POUŽITÝCH SLUŽEB

V této práci jsou využity služby pro počítačové zpracování obrazu a překlad společností Google a Microsoft. U jednotlivých služeb jsou popsány jejich vlastnosti, dílčí funkcionality, ceny a také rozhraní REST API, jež následně využito k testování služeb.

#### 3.1 Počítačové zpracování obrazu

Následující kapitola se zabývá podrobným popisem služeb sloužících ke zpracování obrazu, jež poskytují společnosti Microsoft a Google. U jednotlivých služeb jsou popsány jejich základní vlastnosti, možnosti komunikace skrze rozhraní REST API a jejich dílčí funkcionality.

##### 3.1.1 Společnost Google

Zkoumaný obrázek musí splňovat určitá kritéria, jako je jeho formát, rozlišení či velikost. Podporované formáty jsou (jpg, png8, png24, gif, bmp, webp, raw, ico, pdf a tiff). Jelikož formát gif umožňuje vytvořit animované snímky složené z více obrázků, služba v tomto případě využije pouze první snímek animace. Co se týče rozlišení, pro tuto službu se obecně doporučuje obrázek s rozlišením nejméně 640 x 480 pixelů. Při zvolení menšího, než doporučeného rozlišení může dojít k méně kvalitnímu vyhodnocení. Naopak pokud je obrázek větší, než je doporučeno, ne vždy je zajištěno získání relevantnějších výsledků. Doporučené rozlišení se u jednotlivých funkcí liší a jsou uvedeny u jejich podrobných popisů. Velikost obrázku je omezena na 20 MB a velikost těla dotazu pak na 10 MB. Při jednom dotazu je služba schopna zpracovat maximálně 16 fotografií. Maximální počet dotazů za minutu je 1800. Textové soubory jsou podporovány pouze ve formátu PDF. [25][26]

##### 3.1.1.1 Obecný popis rozhraní

Požadavek na tuto službu se liší podle poskytnutých dat ke zpracování. Poskytnutá data mohou být ve formě obrázku nebo ve formě textového souboru. Služba také disponuje možností pro synchronní a asynchronní zpracování těchto dat. Požadavek na tuto službu je zprostředkován pomocí http metody POST. Veškerá data a další informace na analýzu jsou umístěny v těle požadavku, které je tvořeno objektem formátu JSON. Dodatečné informace jsou umístěny v URL adrese služby. Obecně lze tělo požadavku napsat následovně. [27]



```
{
  "request": [
    [PŘÍSLUŠNÝ_OBJEKT_DLE_ANALYZOVANÝCH_DAT]
  ]
}
```

Tělo požadavku je tvořeno polem request. Toto pole může obsahovat objekt AnnotateImageRequest, jež definuje vlastnosti pro analýzu obrázku nebo objekt AnnotateFileRequest definující vlastnosti pro analýzu souborů. [28]

### Tělo objektu AnnotateImageRequest

```
{
  "image" : {
    [INFORMACE_O_OBRÁZKU]
  },
  "features" : [ {
    [INFORMACE_O_FUNKCI]
  } ],
  "ImageContext" : {
    [DODATEČNÉ_INFORMACE]
  }
}
```

- **Informace o obrázku**

Definuje, z jakého zdroje má být obrázek pro analýzu použit. Lze využít lokálně umístěný obrázek jako řetězen kódovaný v base64. V případě použití lokálního obrázku je řetězec hodnotou klíče content. Další možností je použít URL adresu. Ta je umístěna v objektu source jako hodnota klíče imageUrl. Poslední možností je využít obrázek jež je umístěn v uložišti Google Cloud Storage. Adresa je pak umístěna v objektu source jako hodnota klíče gcsImageUrl. [29]

- **Informace o funkci**

Definuje, jakou analýzu má služba na daném obrázku provést. Tato definice funkce je hodnotou klíče type. Dále maximální počet poskytnutých výsledků jako hodnotu klíče maxResults a model který má provést analýzu jako hodnotu klíče model. [30]

- **Dodatečné informace**

Poskytují službě další informace k jednotlivým funkcím. Tyto informace jsou uvedeny u popisu jednotlivých funkcí. [31]

Objekt `AnnotateFileRequest` je svou strukturou podobný objektu `AnnotateImageRequest`. Jeho tělo však navíc disponuje polem `pages`, jež je určeno k definování strany dokumentu, na kterém se má daná funkce provádět. Data pro analýzu jsou v tomto objektu předávána jako hodnota klíče `inputConfig`, jež nahrazuje klíč `image` v objektu `AnnotateImageRequest`. `InputConfig` navíc disponuje klíčem `mimeType` pomocí jehož hodnoty lze definovat typ souboru. [28][32]

V adrese na službu je dále definována využívaná verze, typ souboru a informace, zda má služba provést analýzu synchronně či asynchronně.

`https://vision.googleapis.com/[VERZE]/[TYP_SOUBORU]:[REŽIM]`

- Verze – použitá verze služby
- Typ souboru – nabývá hodnoty `files` nebo `images` podle typu analyzovaných dat.
- Režim – nabývá hodnoty `annotate` nebo `asyncBatchAnnotate` dle zvoleného režimu.

### 3.1.1.2 Detekce log známých značek

Služba disponuje funkcí pro detekci log známých značek na obrázku s doporučenou velikostí 640 x 480 px. V tomto případě je za hodnotu klíče `type` zvolen řetězen `LOGO_DETECTION`. Při provedení požadavku se služba pokusí detekovat známé obchodní značky. V případě detekce služba vrací pole `logoAnnotations` jež disponuje objekty s informacemi o název společnosti, pravděpodobnost určení, identifikátor `mid` a vrcholy rámečku, který daný text nebo logo ohraničují. Tyto vrcholy jsou udávány v pixelech. Detekovaných značek může být více na jednom poskytnutém obrázku. [33]

```
“logoAnnotations”:[
  {
    “mid”: [IDENTIFIKÁTOR],
    “description”: [NÁZEV_ZNAČKY],
    “score”: [PRAVDĚPODOBNOST_URČENÍ],
    “boundingPoly”: { [VRCHOLY_OBRAZCE] }
  }
]
```

### 3.1.1.3 Detekce tváře

Pro detekci tváře je hodnota klíče type rovna řetězci FACE\_DETECTION. Služba se pokusí detekovat tváře na poskytnutém obrázku. Při detekování obličeje služba vrací velkou škálu informací. Mezi tyto informace patří vrcholy rámečku ohraničující obličej. Tyto vrcholy služba vrací ve dvou provedeních, a to pro rámeček jež ohraničuje obličej dle lidského očekávání a pro rámeček, který je blíže detekovanému obličej sloužící k rozmazání detekovaného obličeje. Další informací v odpovědi je pole, jež nese objekty s informacemi o jednotlivých prvcích tváře jako je nos či oči. Tyto objekty obsahují typ nalezeného orientačního bodu na tváři a jeho umístění na fotografii včetně hloubky.

Služba se však neomezuje jen na obecnou detekci jednotlivých prvků, ale poskytuje detailní popis rysů tváře jako šířka nosu nebo umístění spodního rtu. Služba dále poskytuje informace o natočení obličeje, detekci emocí, pravděpodobnost detekování obličeje jako celku a pravděpodobnost detekování jednotlivých prvků obličeje. Doporučené rozlišení obrázku je pro tuto funkci 1600 x 1200 px. [27][33]

```
“boundingPoly”: {  
    [OHRANIČENÍ_OBLIČEJE]  
},  
“fdBoundingPoly”: {  
    [TĚSNÉ_OHRANIČENÍ_OBLIČEJE]  
},  
“landmarks”: [  
    {  
        type: [TYP_NALEZENÉHO_ORIENTAČNÍHO_BODU_OBLIČEJE],  
        position: {  
            “x”: [POZICE_X_BODU_UPROSTŘED_NALEZENÉHO_PRVKU],  
            “y”: [POZICE_Y_BODU_UPROSTŘED_NALEZENÉHO_PRVKU],  
            “z”: [POZICE_Z_BODU_UPROSTŘED_NALEZENÉHO_PRVKU]  
        }  
    }  
],  
“rollAngle”: [VODOROVNÝ_ÚHEL_NATOČENÍ_OBLIČEJE],  
“panAngle”: [ÚHEL_NATOČENÍ_OBLIČEJE_VŮČI_OBRÁZKU],  
“tilAngle”: [SVYSLÝ_ÚHEL_NATOČENÍ_OBLIČEJE],  
“detectionConfidence”: [PRAVDĚPODOBNOST_DETEKCE_OBLIČEJE],  
“landmarkingConfidence”: [PRAVDĚPODOBNOST_DETEKCE_PRVKŮ_TVÁŘE],  
“joyLikelihood”: [PRAVDĚPODOBNOST_ŠTASTNÉHO_VÝRAZU],  
“SorrowLikelihood”: [PRAVDĚPODOBNOST_SMUTNÉHO_VÝRAZU],  
“angerLikelihood”: [PRAVDĚPODOBNOST_VÝRAZU_HNĚVU],
```

```

“surpriseLikelihood”: [PRAVDĚPODOBNOST_VÝRAZU_PŘEKVAPENÍ],
“underExposedLikelihood”: [PRAVDĚPODOBNOST_PODEXPONOVÁNÍ],
“blurredLikelihood”: [PRAVDĚPODOBNOST_ROZMAZÁNÍ],
“headwearLikelihood”: [PRAVDĚPODOBNOST_POKRÝVKY_HLAVY]
}

```

#### 3.1.1.4 Detekce objektů

Služba dokáže detekovat objekty na poskytnutém obrázku. Pro použití této funkce je nutné hodnotu klíče type nahradit řetězcem LABEL\_DETECTION. Služba v tomto případě detekuje veškeré objekty na obrázku s doporučenou velikostí 640 x 480 px a vrací objekty s identifikátorem mid, popisem obrázku a pravděpodobností určení daného objektu. Tato funkce však nalezené objekty nijak na obrázku nelokalizuje a neposkytuje informace o jejich poloze.[33]

```

“responses”:[
  {
    “labelAnnotations”:[
      {
        “mid”: [IDENTIFIKÁTOR],
        “description”: [NÁZEV_OBJEKTU],
        “score”: [PRAVDĚPODOBNOST_URČENÍ]
      }
    ]
  }
]

```

#### 3.1.1.5 Lokalizace obrázků

Tato funkce dokáže nejen detekovat objekty na obrázku a poskytnout k nim odpovídající popis, ale také polohu daného objektu na obrázku. Využití této funkce je inicializováno pomocí hodnoty OBJECT\_LOCALIZATION klíče type. Po detekci objektů služba vrací ve své odpovědi pole localizedObjectAnnotations obsahující objekty, jež představují jednotlivé detekce. V těchto detekcích se nachází identifikátor mid, název objektu, pravděpodobnost určení tohoto objektu, a souřadnice rámečku, jež tento objekt ohraničuje. [33]

```

“localizedObjectAnnotations” : [
  {
    “mid”: [IDENTIFIKÁTOR],
    “mane”: [JMÉNO_NALEZENÉHO_OBJEKTU],
    “score”: [PRAVDĚPODOBNOST_DETEKCE],

```

```

    "boundingPoly": {
      [VRCHOLY_OHRAŇUČÍJÍCÍHO_RÁMEČKU]
    }
  }
]

```

### 3.1.1.6 Vytvoření výstřižku

Pro vytvoření výstřižků je klíč type nahrazen řetězcem CROP\_HINTS. Dále lze službě poskytnout informaci o poměru stran, jaký má mít finální výstřižek. Tento poměr je reprezentován jako výsledek dělení zvoleného poměru. Pokud je zvolený poměr např. 16/9 službě je předáno číslo 1,77. Toto číslo pak může nabývat nejvíce hodnoty 16. V případě ignorování tohoto parametru služba automaticky zvolí vhodný poměr výstřižku. Poměr se službě předává jako hodnota pole aspectRatios v dodatečných informacích. Odpověď na tento dotaz obsahuje informaci o důležitosti vyřezané oblasti vůči původnímu obrázku, pravděpodobnost určení správné oblasti a souřadnice pro vytvoření výřezu. [33]

### 3.1.1.7 Detekce známých míst

Služba nabízí možnost pro detekci geograficky či architektonicky známých míst např. Eiffelova věž. Pro provedení tohoto požadavku je nutné hodnotu klíče type nahradit řetězcem LANDMARK\_DETECTION. Pro obrázek je doporučená velikost 640 x 480 px. Pokud služba detekuje známé místo vrací odpověď, ve které se nachází identifikátor mid, název tohoto místa, pravděpodobnost detekce, geografickou polohu a souřadnice vrcholů rámečku, který označuje nalezené známé místo na obrázku. [33]

```

"landmarkAnnotations": [
{
  "mid": [IDENTIFIKÁTOR],
  "description": [NÁZEV_ZNÁMÉHO_MÍSTA],
  "score": [PRAVDĚPODOBNOST_URČENÍ],
  "boundingPoly": { [VRCHOLY_RÁMEČKU] },
  "locations": [
    {
      "latLng": {
        "latitude": [ZEMĚPISNÁ_ŠÍŘKA],
        "longitude": [ZEMĚPISNÁ_DÉLKA]
      }
    }
  ]
}
]

```

]

### 3.1.1.8 Detekce textu

Služba disponuje možností detekovat text na poskytnutém obrázku nebo souboru. Pro poskytnutý obrázek je udávána doporučená velikost 1024 x 768 px. Dokáže detekovat jak jednotlivá slova textu např. na reklamních poutacích tak i hutný text dokumentů. V neposlední řadě služba dokáže detekovat ručně psaný text. Pro detekci jednoduchých textu je hodnota klíče type nahrazena řetězcem TEXT\_DETECTION. Při detekci textu služba vrací odpověď v poli textAnnotations. Toto pole obsahuje objekty, jež poskytují informace o poloze textu jako body vrcholů rámečku, který tento text ohraničuje a změni detekovaného textu. V prvním z objektů v odpovědi služby jsou umístěny informace o celém detekovaném textu včetně automatického rozpoznání jazyka. Ostatní objekty nesou informace o jednotlivých slovech.

```
“textAnnotations”: [  
  {  
    “locale”: [AUTOMATICKY_IDENTIFIKOVANÝ_JAZYK],  
    “description”: [NALEZENÝ_TEXT],  
    “boundingPoly”: {  
      “vertices”: [SOUŘADNICE_VRCHOLŮ]  
    }  
  }  
]
```

Pro detekci hustého textu či rukopisu je hodnota klíče type nahrazena řetězcem DOCUMENT\_TEXT\_DETECTION. Odpověď je v tomto případě složena z pole textAnnotations jako v předchozím případě. Navíc je v této odpovědi přidán objekt fullTextAnnotation. Tento objekt ve svém těle nese detailnější informaci jako šířku a výšky analyzované stránky nebo informace o jednotlivých blocích nalezeného textu. Jelikož je služba v tomto případě schopna detekovat více jazyku na jedné stránce dokumenty či obrázku, obsahuje tento objekt i seznam detekovaných jazyků. [33][42]

```
“pages”: [  
  {  
    “property”: {  
      “detectionLanguages”: [  
        {  
          “languageCode”: [KOD_DETEKOVANÉHO_JAZYKA],  
          “confidence”: [PRAVDĚPODOBNOST_DETEKCE_JAZYKA]  
        }  
      ]  
    }  
  }  
]
```

```

    ],
    "detectionBreak": {
        "type": [TYP_PŘERUŠENÍ_JEDNOTLIVÝCH_JAZYKŮ],
        "isPrefix": [JE_JAZYK_PŘERUŠEN]
    }
},
"width": [ŠÍŘKA_STRÁNKY],
"height": [VÝŠKA_STRÁNKY],
"blocks": [
    {
        [OBJEKT_DETECTED_LANGUAGE],
        [OBJEKT_DETECTED_BREAK]
    }
],
"confidence": [PRAVDĚPODOBNOST_SPRÁVNÉ_DETEKCE_CELÉ_STRÁNKY]
],
"text": [DETEKOVANÝ_TEXT]

```

### 3.1.1.9 Detekce dominantní barvy

Pomocí této funkce služba dokáže detekovat dominantní barvy na obrázku. Pro využití této funkce je hodnota klíče type nahrazena řetězcem IMAGE\_PROPERTIES. Odpověď v tomto případě je tvořena objektem imagePropertiesAnnotation, který obsahuje pole colors. Toto pole je tvořeno objekty, které definují informace o jednotlivých dominantních barvách. Jedná se o barvu v podobě RGB složek, pravděpodobnost určení této barvy a poměr kolik pixelů zabírá barva na obrázku. [33]

```

"dominantColors": {
    "colors": [{
        "color": {
            "red": [MNOŽSTVÍ_ČERVENÉ],
            "green": [MNOŽSTVÍ_ZELENÉ],
            "blue": [MNOŽSTVÍ_MODRÉ],
            "alpha": [MÍRA_PRŮHLEDNOSTI]
        },
        "score": [PRAVDĚPODOBNOST_URČENÍ],
        "pixelFraction": [POMĚR_BARVY_NA_OBRÁZKU]
    }]
}

```

### 3.1.1.10 Detekce nevhodného obsahu

Pomocí této funkce služba dokáže rozlišit pět druhů nevhodného či jinak urážlivého obsahu. Jedná se o obsah pro dospělé jako je pornografie, upravený obsah za účelem satiry, obsah vyobrazující operaci či lékařský zákrok, násilný obsah a obsah pro dospělé jako např. letmá nahota. Tato funkce je využita pomocí hodnoty SAFE\_SEARCH\_DETECTION klíče type. Úrovně detekce jednotlivých kategorií jsou VERY\_UNLIKELY, UNLIKELY, POSSIBLE, LIKELY a VERY\_LIKELY seřazené od nejméně pravděpodobné po naprosto jisté určení kategorie. V každé odpovědi této funkce je přítomno všech pět kategorií s příslušnou mírou určení. [33]

```
“safeSearchAnnotation”: {  
    “adult”: [PORNOGRAFIE],  
    “spoof”: [OBSAH_JE_SATIRA],  
    “medical”: [LÉKAŘSKÝ_OBSAH],  
    “violence”: [NÁSILNÝ_OBSAH],  
    “racy”: [OBSAH_PRO_DOSPĚLÉ]  
}
```

### 3.1.1.11 Nalezení obsahu na webu

Služba umožňuje díky této funkci nalézt podobný obsah na internetu jako je na poskytnutém obrázku s doporučenou velikostí 640 x 480 px. Tuto funkci lze využít pomocí hodnoty WEB\_DETECTION klíče type. V odpovědi služba jsou informace jako URL adresa na obrázek, jež přesně odpovídá obrázku poskytnutému spolu s pravděpodobností určení. Dále URL adresu na obrázek, který je přibližně stejný jako analyzovaný spolu s jeho pravděpodobností určení. V neposlední řadě také anglický popis, co se na obrázku vyskytuje. [33]

Pro určení URL adresy a pravděpodobnosti detekce jednotlivých obrázku na internetu funkce využívá objekt WebImages. Url adresa je umístěna jako hodnota klíče url a pravděpodobnost pak jako hodnota klíče score. V následujícím ukázce odpovědi funkce je uveden jen obecný objekt WebImages.

```
“webEntities”: [  
    {  
        “entityId”: [IDENTIFIKÁTOR],  
        “score”: [PRAVDĚPODOBNOST_URČENÍ_ENTITY],  
        “description”: [ANGLICKÉ_POPIS_NALEZENÉHO_OBJEKTU]
```



```
    }  
  ],  
  "fullMatchingImages": [  
    [WEB_IMAGES]  
  ],  
  "partialMatchingImages": [  
    [WEB_IMAGES]  
  ],  
  "pagesWithMatchingImages": [  
    {  
      "url": [URL_ADRESA_NA_STRANKU_OBSAHUJÍCÍ_OBRÁZEK],  
      "score": [PRAVDĚPODOBNOST_OBRÁZKU_NA_STRÁNCE],  
      "pageTitle": [NÁZEV_WEBOVÉ_STRÁNKY_VČETNĚ_HTML_TAGŮ],  
      "fullMatchingImages": [  
        [OBJEKT_FULL_MATCHING_IMAGES]  
      ],  
      "partialMatchingImages": [  
        [OBJEKT_PARTIAL_MATCHING_IMAGES]  
      ]  
    }  
  ],  
  "visuallySimilarImages": [  
    [OBJEKT_WEB_IMAGES]  
  ],  
  "bestGuessLabels": [  
    [OBJEKT_WEB_IMAGES]  
  ]  
]
```

### 3.1.1.12 Školení vlastních modelů

Společnost Google nabízí možnost pro vyškolení vlastních modelů prostřednictvím služby AutoML Vision jejíž obecný popis je uveden v kapitole 2.2.2. Práci s touto službou lze rozdělit do čtyř částí. První z nich je příprava dat a zvolení příslušných kategorií. Jedním s klíčových prvků v této fázi je dostatečný počet relevantních dat, v tomto případě obrázků, pro každou ze zvolených kategorií. Společnost Google uvádí hranici 100 snímků pro každou uživatelskou kategorii, aby byl zajištěn správný výcvik a tím i schopnost predikce modelu. Při zvýšení počtu snímků schopnost určení správné kategorie stoupá. V takovém případě by však mělo být zachováno rozložení počtu snímků u jednotlivých uživatelských kategorií. V ideálním případě je u každé z kategorií zvolen stejný počet obrázků. V reálných případech je pak vhodné použít alespoň jednu desetinu počtu snímků z nejpočetnější kategorie v kategorii s nejnižším výskytem. Dalším významným prvkem ovlivňující výkon vyhodnocení je kontext zvolených snímků. Je vhodné využívat fotografie, které jsou co možná nejblíže reálním snímkům, které bude model následně vyhodnocovat. Ovlivňujícím faktorem může být např. roční období, denní doba, či kvalita snímku. Po přípravě relevantních dat potažmo obrázků nastává druhá část práce s touto službou, kterou je výcvik modelu. Zvolená data jsou službou rozdělena do tří skupin. První skupina těchto dat je využita v samotném výcviku modelu a tvoří 80 % celkové velikosti. Druhou skupinou jsou data použita pro validaci při školení a tvoří 10 % celkové velikosti. Posledních 10 % je vyhrazeno pro samotné vyhodnocení modelu. Rozdělení do příslušných skupin lze uživatelsky zvolit. Zatímco první dvě skupiny dat se přímo účastní výcvikové procesy, poslední vzorek je využit pro reálný test funkčnosti. Výcvik základního modelu trvá kolem 15 minut. Pro delší dobu výcviku je cena tohoto výcviku 20 dolarů za každou započatou hodinu. Služba nabízí možnost zdarma vycvičit 10 takovýchto modelů. V případě výcviku více modelů paralelně je možno vycvičit až 5 modelů současně. Po úspěšném vycvičení modelu přichází třetí fáze, kterou je vyhodnocení. Služba nabízí informace o pravděpodobnostech určení jednotlivých uživatelských kategorií. V případě této predikce je prvních 1000 analyzovaných obrázků zdarma, poté 3 dolary za každých 1000 predikcí. [34]

### 3.1.2 Společnost Microsoft

Analyzování obrázků musí splňovat podmínky na formát, velikost a rozlišení. Služba přímá tyto formáty obrázku (JPEG, PNG, GIF, BMP). V případě formátu GIF služba využije pouze

první snímek animace. Velikost poskytnutého obrázku nesmí přesáhnout 4 MB a jeho rozlišení musí být větší jako 50 x 50 px. [35][41]

### 3.1.2.1 *Obecný popis rozhraní*

Jednotlivé funkce služby jsou definovány jako parametr URL koncového bodu. Obecně lze tuto adresu napsat následujícím způsobem.

`https://[UMÍSTĚNÍ].api.cognitive.microsoft.com/vision/[VERZE]/[FUNKCE]?[PARAMETRY FUNKCE]`

- Umístění – definuje který ze serverů společnosti Microsoft se má použít
- Verze – definuje verzi služby, liší se dle jednotlivých funkcí
- Funkce – definuje jaká konkrétní funkce se má využít
- Parametry funkce – definuje parametry jednotlivých funkcí

Tyto obecné parametry jsou konkrétně definovány u popisu jednotlivých funkcí. Požadavky na jednotlivé funkce jsou prováděny pomocí http metod POST nebo GET. V případě požadavku POST je u všech takových funkcí tělo tohoto požadavku tvořeno objektem JSON nesoucí analyzovaný obrázek. Tento obrázek lze definovat více způsoby, a to jako URL adresu na které je daný obrázek přístupný nebo jako obrázek ve formě sekvence bytů. V případě poskytnutí obrázku ve formě URL adresy je tato adresa v těle požadavku umístěna jako hodnota klíče url. U všech požadavků se do hlavičky umísťuje API klíč jako parametr Ocp-Admin-Subscription-Key a parametr Content-Type, který definuje datový typ obrázku. [35]

### 3.1.2.2 *Popis obrázku v celých větách*

Jedná se o funkci, která dokáže popsat obrázek v lidsky čitelné formě. Jde se o popis obrázku v celých větách. Funkce v tomto případě jako odpověď vrací seznam nalezených objektů a pole vytvořených popisujících vět. Každá z vět pak disponuje pravděpodobností, ze kterou funkce tuto větu vytvořila. [35][36]

Při dotazu na tuto funkci je nutné využít metodu POST. Obecný parametr funkce adresy na službu je nahrazen hodnotou description. Tato funkce dále přímá jediný parametr maxCandidates, který definuje maximální počet popisujících vět. Tento parametr je nepovinný, přičemž v případě neuvedení tohoto parametru je výchozí počet popisujících vět roven jedné.

V tělo odpovědi na dotaz je pak uvedeno pole captions nesoucí vytvořený popis a pravděpodobnost jeho určení. Dále se v těle požadavku nachází pole nalezených objektů nazvané jako tags nesoucí názvy detekovaných objektů. [36]

```
"description": {  
  "tags": [ [NALEZENÉ_OBJEKTY] ],  
  "captions": [  
    {  
      "text": [VYTVOŘENÍ_TEXT],  
      "confidence": [PRAVDĚPODOBNOST_URČENÍ]  
    }  
  ]  
}
```

### 3.1.2.3 Vytvoření miniatur

Služba nabízí možnost vytvoření miniatury. Jedná se o funkci, jež podle předem definované výšky a šířky dokáže vytvořit miniaturu poskytnutého obrázku. Funkce dále disponuje možností určení zájmu, kdy je ve zvoleném obrázku automaticky nalezen objekt, např. tvář nebo automobil. Tento objekt je ve finální miniatuře zachován, přičemž poměr stran je definován podle zadané výšky a šířky. [36]

Při dotazu na funkci se využívá metody POST. V obecné adrese na službu je parametr funkce nahrazen řetězcem generatorThumbnail. Funkce dále přebírá jako své parametry width pro definici šířky a height pro definici výšky. Dalším nepovinným parametrem je smartCropping pro povolení nalezení oblasti zájmu. V případě, kdy není tento parametr uveden, je tato funkce ignorována. Funkce v odpovědi vrací vytvořený náhled jako bitovou kopii ve formátu jpeg. [36]

### 3.1.2.4 Označení obrázku

Jedná se o funkci, která dokáže detekovat objekty na obrázku. Tato detekce se však nezaměřuje na dominantní předmět, ale poskytuje celkový popis obrázku. Na rozdíl od detekce objektů, tato funkce poskytne pouze přehled všech nalezených objektů a k nim příslušné pravděpodobnosti určení. [36]

Při dotazu na tuto funkci je využita metoda POST. V obecné adrese služby je parametr funkce nahrazen řetězcem tag. Tato funkce nepřebírá žádné specifické parametry. Odpovědí této funkce je pole tags, ve kterém se nacházejí objekty nesoucí název detekovaného objektu a pravděpodobnost jeho určení. [35]

```
"tags": [  
  {  
    "name": [NÁZEV_OBJEKTU]  
    "confidence": [PRAVDĚPODOBNOST_URČENÍ]  
  }  
]
```

### 3.1.2.5 *Obecný popis funkce Analyze image*

Jedná se o komplexní funkci, jež zastřešuje jednotlivé dílčí funkcionality jako je detekce známých míst či určení kategorie obrázku. V obecné adrese služby je parametr funkce nutné nahradit řetězcem analyze. Funkce dále přebírá parametry visualFeatures a details. Tyto obecné parametry určují, jaká funkce se má provést. Při využití více funkcí jsou parametry odděleny čárkou. Tyto parametry jsou v této práci uvedeny u podrobných popisů jednotlivých funkcí. Funkce Analyze image dále přebírá parametr language, který definuje výstupní jazyk odpovědi funkce. Výchozím jazykem je angličtina, avšak český jazyk je pro tuto funkci také podporován. Pro tuto funkci je služba využita ve verzi 2.0 pomocí http metody POST. Obecně lze adresu na tuto funkci napsat následovně.

[https://\[UMÍSTĚNÍ\].api.cognitive.microsoft.com/vision/v2.0/analyze?\[visualFeatures\]&\[details\]&\[language\]](https://[UMÍSTĚNÍ].api.cognitive.microsoft.com/vision/v2.0/analyze?[visualFeatures]&[details]&[language])

Následující kapitoly jsou věnovány jednotlivým dílčím funkcionalitám této komplexní funkce, jejich popisu a možnostem. [35]

### 3.1.2.6 *Detekce známých míst*

Funkce sloužící k detekci geograficky či architektonicky známých míst. Využití této funkce je inicializováno nahrazením obecného parametru details hodnotou Landmarks. V případě správné detekce obsahuje odpověď pole objektů nesoucí anglický název tohoto místa a pravděpodobnost jeho určení. [36]

```
"landmarks": [  
  {  
    "name": [NÁZEV_MÍSTA],
```

```
        "confidence": [PRAVDĚPODOBNOST_URČENÍ]
    }
]
```

### 3.1.2.7 Detekce osobností

Jedná se o dílčí funkci jejíž účelem je detekce celebrit. Pro využití této funkce je nutné obecný parametr `details` nahradit hodnotou `Celebrities`. V případě správné detekce funkce vrací pole nalezených tváří celebrit s údaji rámečku, ve kterém se tvář celebrity nachází, jménem známé celebrity a pravděpodobností se kterou služba tuto celebritu detekovala. Informace o čtverci jsou pak bod uprostřed čtverce definovaný jako vzdálenost od levé a horní strany obrázku. Dalšími nezbytnými informacemi o tomto čtverci jsou jeho výška a šířka. Tyto hodnoty jsou udávány v pixelech. [36]

```
"result": {
  "celebrities": [{
    "faceRectangle": {
      "top": [VZDÁLENOST_OD_HORNÍ_HRANY],
      "left": [VZDÁLENOST_OD_LEVÉ_HRANY],
      "width": [ŠÍŘKA_ČTVERCE],
      "height": [VÝŠKA_ČTVERCE]
    },
    "name": [JMÉNO_CELEBRITY],
    "confidence": [PRAVDĚPODOBNOST_URČENÍ]
  }]
}
```

### 3.1.2.8 Určení kategorie

Nahrazením obecného parametru `visualFeatures` hodnotou `Categories` je po službě vyžádáno určení kategorie, která popisuje poskytnutý obrázek. Tato kategorizace však neznamená nalezení objektů nebo jejich umístění na obrázku. Funkce pouze určí nejvhodnější kategorii pro daný obrázek. Služba disponuje 86 hlavními kategoriemi jež jsou systematicky uspořádané podle příbuzných objektů. Od těchto nadřazených kategorií dále dědí kategorie, jež poskytují detailnější popis obrázku. Kupříkladu při poskytnutí obrázku pizzy funkce v tomto případě vrací kategorii `food_pizza`, jež dědí od své nadřazené kategorie `food`.

Služba při určení kategorie vrací pole categories, v němž se nacházejí objekty nesoucí název kategorie a pravděpodobnost určení této kategorie. [36]

```
"categories": [  
  {  
    "name": [NÁZEV_KATEGORIE],  
    "score": [PRAVDĚPODOBNOST_URČENÍ]  
  },  
]
```

### 3.1.2.9 Detekce log známých značek

Pro detekci log známých značek je obecný parametr visualFeatures nahrazen hodnotou Brands. Funkce se pokusí identifikovat značku na obrázku a v případě úspěchu vrací pole nazvané brands, jež nese objekty s informacemi o jednotlivých značkách. Mezi tyto informace patří název značky, pravděpodobnost určení a souřadnice rámečku, jež detekované logo ohraničuje. [36]

```
"brands": [  
  {  
    "name": [NÁZEV_ZNAČKY],  
    "confidence": [PRAVDĚPODOBNOST_URČENÍ],  
    "rectangle": {  
      "x": [X_SOUREADNICE_BODU],  
      "y": [Y_SOUREADNICE_BODU],  
      "w": [VYSKA_OBDELNIKU],  
      "h": [SIRKA_OBDELNIKU]  
    }  
  }  
]
```

### 3.1.2.10 Detekce dominantní barvy

Jedná se o dílčí funkci, jejíž účelem je detekovat dominantní barvu obrázku. V tomto případě je nutné obecný parametr visualFeatures nahradit hodnotou Color. Funkce analyzuje obrázek a poskytne informace o dominantní barvě popředí, pozadí a celého obrázku. V případě dominantní barvy celého obrázku může funkce těchto barev poskytnout více než jednu. Dalšími informacemi, jež služba poskytuje je logický údaj, zda je obrázek černobílý a hodnotu barvy uvedenou jako hexadecimální číslo. Toto číslo je kombinací nejjasnější barvy, saturace a dominantní barvy celého obrázku. Tyto informace jsou pak v odpovědi umístěny jako objekt color. [35][36]

```
"color": {
  "dominantColorForeground": [DOMINANTNÍ_BARVA_POPŘEDÍ],
  "dominantColorBackground": [DOMINANTNÍ_BARVA_POZADÍ],
  "dominantColors": [POLE_DOMINANTNÍCH_BAREV],
  "accentColor": [HEXADECIMÁLNÍ_ČISLO_KOMBINACE],
  "isBWImg": [JE_ČERNOBÍLÍ]
}
```

### 3.1.2.11 Detekce nevhodného obsahu

Pomocí této dílčí funkce je služba schopna detekovat sexuální či pikantní obsah. Pro využití této funkce je nutné nahradit obecný parametr `visualFeatures` hodnotou `Adult`. Při provedení detekce funkce vrácí jako svou odpověď objekt `adult`. Tato odpověď obsahuje logický parametr `isAdultContent` nesoucí informaci o tom, zda je obsah klasifikován jako sexuální. Dalším parametrem odpovědi je logická proměnná `isRacyContent` nesoucí informaci, zda se v analyzovaném obrázku nachází pikantní obsah. Parametry `adultScore` a `racyScore` pak poskytují informaci o pravděpodobnosti určení těchto dvou typů nevhodného obsahu. [35][36]

```
"adult": {
  "isAdultContent": [SEXUÁLNÍ_OBSAH],
  "isRacyContent": [NEVHODNÝ_OBSAH],
  "adultScore": [PRAVDĚPODOBNOST_URČENÍ_SEXUÁLNÍHO_OBSAHU],
  "racyScore": [PRAVDĚPODOBNOST_URČENÍ_NEVHODNÉHO_OBSAHU]
}
```

### 3.1.2.12 Detekce obličejů

Možnost detekce tváře lze provést nahrazením obecného parametru `visualFeatures` hodnotou `Faces`. V tomto případě dílčí funkce provede detekci tváře a poskytne odhad věku, pohlaví a umístění tváře na obrázku. Toto umístění obsahuje souřadnice bodu, šířku a výšku rámečku ohraničující tvař. V odpovědi jsou pak informace o detekovaných tvářích uvedené v poli `faces` jako jednotlivé objekty. [35][36]

```
"faces": [
  {
    "age": [VĚK_DETEKOVANÉ_OSOBY],
    "gender": [POHLAVÝ_DETEKOVANÉ_OSOBY],
    "faceRectangle": {
      "left": [POZICE_OD_LEVÉHO_OKRAJE],
```



```
"top": [POZICE_OD_PRAVÉHO_OKRAJE],  
"width": [ŠÍŘKA_OBRAZCE],  
"height": [VÝŠKA_OBRAZCE]  
}  
]
```

### 3.1.2.13 Detekce typu obrázku

Tato dílčí funkce dokáže detekovat, zda je daný obrázek klipart nebo perokresba. Obrázkem klipart je myšlena jednoduchá digitální ilustrace. Perokresba pak představuje obrázek nakreslený rukou či pomocí grafického editoru, jenž je tvořen z čar o různých tloušťkách. Pro využití této dílčí funkce je obecný parametr `visualFeatures` nahrazen řetzcem `ImageType`. Při detekci jednoho z obrázků tato funkce vrací objekt `imageType`. Tento objekt je tvořen ze dvou klíčů `clipArtType` a `lineDrawingType`. Tyto klíče mohou nabývat čtyř číselných hodnot v závislosti na pravděpodobnost, s jakou funkce určila typ obrázku. V případě, že funkce nedetekuje žádný z uvedených typů obrázku hodnota je rovna číslu 0. Při nejednoznačné detekci je hodnota rovna 1. Detekce jednoho z typů je reprezentována číslem 3. Detekci s nejvyšší pravděpodobností pak představuje číslo 4. [36][37]

```
"imageType": {  
  "clipArtType": [ČÍSLO_PRAVDĚPODOBNOTI_URČENÍ_CLIPARTU],  
  "lineDrawingType": [ČÍSLO_PRAVDĚPODOBNOTI_URČENÍ_PEROKRESBY]  
}
```

### 3.1.2.14 Detekce objektů

Služba nabízí možnost detekce objektů v obrázku. Pro inicializaci této detekce je nutné obecný parametr funkce nahradit hodnotou `detect`. V případě detekce objektu funkce vrací informace o rámečku v pixelech, který tento objekt ohraničuje, název objektu, jeho pravděpodobnost určení tohoto objektu. Dále tato funkce poskytuje informaci o nadřazené kategorii tohoto objektu. [37]

```
"objects": [  
  {  
    "rectangle": {  
      "x": [SOUŘADNICE_X_POLOHY_OBRÁZKU],  
      "y": [SOUŘADNICE_Y_POLOHY_OBRÁZKU],  
      "w": [VÝŠKA_RÁMEČKU],
```

```
    "h": [ŠÍŘKA_RÁMEČKU]
  },
  "object": [NÁZEV_NALEZENÉHO_OBJEKTU],
  "confidence": [PRAVDĚPODOBNOST_URČENÍ_OBJEKTU],
  "parent": {
    "object": [NÁZEV_NADŘÁZENÉHO_OBJEKTU],
    "confidence": [PRAVDĚPODOBNOST_URČENÍ]
  }
}
]
```

## 3.2 Překlad textu

Jedná se o služby sloužící k překladu textu. Tento text může být ve formě prostého řetězce nebo ve formě HTML souboru. V případě HTML souboru služba překládá pouze text, přičemž HTML tagy ignoruje. Služby dále nabízejí možnost pro detekci jazyka nebo převod textu mezi znakovými sadami. [38]

### 3.2.1 Společnosti Google

Služba pro překlad textu je společností Google poskytována zdarma a to v případě, že počet znaků poskytnutých k překladu nepřesáhne množství 500 000 za měsíc. V případě nutnosti překládat více znaků je cena \$20 za každých dalších milion znaků. V případě detekce jazyka je tato funkce zpoplatněna \$20 za každých milion znaků. Co se týče množství provedených dotazů na překlad, služba nabízí možnost přeložit miliardu znaků za den a milion znaků za sto sekund. Tyto kvóty je možné navýšit, a to maximálně na deset milionu znaků za sto sekund. Denní limit je možné navýšit až na neomezené množství znaků. Jazyky pro tuto službu jsou definovány pomocí standartu ISO 639-1. Služba podporuje přes sto jazyků, které je možné detekovat nebo přeložit včetně češtiny. [38]

#### 3.2.1.1 Obecný popis rozhraní

Pro definování, jakou funkci této služby využít nebo poskytnutí informací jako je např. zdrojový jazyk textu využívá služba parametry URL adresy. Obecně lze adresu služby napsat následovně.

[https://translation.googleapis.com/language/translate/\[VERZE\]/\[FUNKCE\]/\[PARAMERTY\\_FUNKCE\]](https://translation.googleapis.com/language/translate/[VERZE]/[FUNKCE]/[PARAMERTY_FUNKCE])

- Verze – definuje jaká verze služby je využita, v současné době je služba dostupná ve verzi v2 a v3beta1

- Funkce – definuje jakou funkci má služba při dotazu provést
- Parametry Funkce – definují jednotlivé parametry jednotlivých funkcí

Požadavky na tuto službu jsou provedeny pomocí metod POST a GET. Této službě se v případě dotazu pomocí metody POST nepředávají žádné informace v těle požadavku. Znamená to, že veškeré informace, jež služba a jednotlivé funkce vyžadují jsou umístěny jako parametry URL adresy služby. [38]

### 3.2.1.2 Překlad textu

Jedná se o základní funkci této služby. Pro využití této funkce je obecný parametr funkce URL adresy služby nahrazen prázdným řetězcem. Funkce vyžaduje dva povinné parametry, a to text k překladu, umístěný na adrese jako hodnota parametru q, a cílový jazyk, do kterého se má vstupní text přeložit. Tento jazyk je pak definován jako hodnota parametru target. V případě překladu textu do více jazyků je tento parametr s příslušným kódem cílového jazyka použit opakovaně. Další parametry funkce jsou volitelné. [38]

Funkce disponuje možností překladu jak prostého textu, tak textů, jež jsou umístěny v HTML tazích. Tuto možnost typu textu lze ovlivnit pomocí parametru format, přičemž parametr se může rovnat hodnotě text nebo html. Funkce jako výchozí formát využívá HTML. V případě poskytnutí vstupního textu jako HTML stránku jsou přeloženy pouze přítomné texty a HTML tagy zůstávají beze změn. [38]

Další volitelnou možností pro tuto funkci je zvolení vstupního jazyka. Tento jazyk lze definovat pomocí parametru source. Posledním možným parametrem je volba modelu, jež je využit pro překlad pomocí parametru model. Tento parametr může nabývat hodnoty base pro překlad s využitím statického překladového modelu nebo hodnoty nmt pro překlad využívající neuronovou síť. [38]

Po úspěšném provedení překladu služba ve své odpovědi vrací objekt JSON. Ten obsahuje pole, jež nese jednotlivé překlady. Přeložené texty jsou definovány jako hodnota klíče translatedText. V případě využití automatické detekce zdrojového jazyka funkce vrací kód tohoto jazyka jako hodnotu klíče detectedSourceLanguage. [38]

```
“data”: {  
  “translations”: [  
    “detectedSourceLanguage”: [AUTOMATICKY_DETEKOVANÝ_JAZYK],  
    “translatedText”: [PŘELOŽENÝ_TEXT]  
  ]  
}
```

```
}
```

### 3.2.1.3 Detekování jazyka

Služba disponuje možností detekování jazyka, ve kterém je vstupní text napsán. Tato funkce však provádí pouze detekci, nikoliv překlad. Funkce vyžaduje jediný parametr a to vstupní text. Tento text je funkci poskytnut prostřednictvím parametru q v URL adrese služby. Jeli-kož může služba detekovat více jazyků v jednom poskytnutém textu, odpověď obsahuje pole jednotlivých detekovaných jazyků. Informace, jež služba o detekci poskytuje, jsou kód de-tekovaného jazyka, informace, zda je detekce spolehlivá a její pravděpodobnost. [39]

```
“detections”: [  
  {  
    “language”: [DETEKOVANÝ_JAZYK],  
    “isReliable”: [JE_DETEKCE_SPOLEHLIVÁ],  
    “confidence”: [PRAVDĚPODOBNOST_URČENÍ]  
  }  
]
```

### 3.2.2 Společnosti Microsoft

Služba podporuje přes 60 jazyků včetně češtiny. Co se týče množství dotazů na překlad, služba uživatele nijak neomezuje. Množství znaků k překladu v jednom dotazu na službu je omezeno na 5 000. Překladač je účtován za množství překládaných znaků. Jako u ostatních služeb společnosti Microsoft jsou jejich poplatky a limity rozděleny do pěti úrovní. Úroveň F0 nabízí možnost přeložit maximálně 2 miliony znaků za hodinu. Úroveň S1 a S2 maxi-málně 40 milionu znaků, S3 pak 120 milionu a S4 nabízí možnost přeložit 200 milionu znaků za hodinu. Služba dále nabízí možnost přeložit 2 miliony znaků za měsíc zcela zdarma, a to pro úroveň zpoplatnění F0. V případě úrovně S1 je cena překladu \$10 měsíčně. Pro úroveň S2 služba stojí \$2000 a je možné přeložit až 250 milionu znaků za měsíc. Úroveň S3 nabízí možnost přeložit miliardu znaků měsíčně za \$6 000 a úroveň S4 nabídne až 10 miliard znaků měsíčně za \$45 000. [39]

#### 3.2.2.1 Obecný popis rozhraní

Služba pro zprostředkování jednotlivých funkcí využívá parametry URL adresy služby. Obecně lze tuto adresu napsat následovně.

[https://api.cognitive.microsofttranslator.com/translate?apiversion=\[VERZE\]&\[PARAMETRY\]](https://api.cognitive.microsofttranslator.com/translate?apiversion=[VERZE]&[PARAMETRY])

- Verze – použitá verze nabývající hodnot v3 nebo v2

- Parametry – liší se dle použitých funkcí

Požadavek je prováděn pomocí metody POST, jehož tělo tvoří pole JSON. Toto pole obsahuje objekty nesoucí řetězce k překladu. Tyto řetězce jsou hodnotou klíče text. Textů na překlad může být v jednom dotazu na službu více.

V hlavičce požadavku je dále nutné uvést typ ověření, typ obsahu a délku požadavku. Pro ověření lze použít jednu z následujících možností. Použití API klíč pro službu překladu, který je definován jako hodnota klíče OCP-Apim-Subscription-Key. Další možností je využití API klíče pro balíček služeb Cognitive Service, v němž je služba překladu zahrnuta. Tento klíč je v hlavičce umístěn jako hodnota klíče OCP-Apim-Subscription-Region. Poslední možností je využití autorizačního tokenu. Tento token je v hlavičce uveden jako hodnota klíče Authorization. [39]

Volitelným údajem pro tento požadavek je poskytnutí identifikátoru GUID jako hodnotu klíče X-ClientTraceId. [39]

### 3.2.2.2 Překlad textu

Základní funkcí této služby je překlad textu. K provedení dotazu na překlad je nutné obecný parametr URL adresy nahradit následujícími parametry. Jediným povinným je v tomto případě parametr to. Ten definuje, do jakého jazyka se má poskytnutý řetězec přeložit. V případě poskytnutí pouze tohoto parametru je zdrojový jazyk řetězce určen automaticky. Zdrojový jazyk je možné definovat pomocí parametru from. Další možné parametry jsou nepovinné. Za zmínku stojí definice typu textu k čemuž slouží parametr textType. Typ textu může být buďto prostý text reprezentovaný hodnotou plain, jež je zároveň hodnotou výchozí pro tuto funkci nebo html soubor reprezentovaný hodnotou html.

V odpovědi této funkce jsou obsaženy následující informace. Detekovaný jazyk obsažený v objektu detectedLanguage jako hodnota klíče language spolu s pravděpodobností určení tohoto jazyka. Finální překlad je umístěn v poli translations jako objekt. Tento objekt obsahuje kód cílového jazyka a přeložený text. [39]

### 3.2.2.3 Převod mezi znakovými sadami

Služba nabízí možnost převodu mezi znakovými sadami v předem definovaném jazyce. Při použití této funkce je nutné parametry URL adresy zvolit následovně. Parametr language definuje, v jakém jazyce má být výsledný převod. Pomocí parametru fromScript je určena znaková sada vstupního textu a parametr toScript určuje výstupní znakovou sadu.

Odpověď obsahuje pole nesoucí objekt, jež ve svém těle nese převedený text jako hodnotu klíče text. Dále tento objekt obsahuje výstupní znakovou sadu, kterou služba poskytuje jako hodnotu klíče script. [39]

```
[
  {"text": [PŘEVEDENÝ_TEXT], "script": [ZNAKOVÁ_SADA]}
]
```

### 3.2.2.4 Detekce jazyka

Služba dále disponuje funkcí pro detekci jazyka zkoumaného textu. Tato funkce při dotazu nepřebírá žádné parametry URL adresy. Funkce na poskytnutém textu neprovede překlad, ale pouze detekuje zdrojový jazyk. Text pro detekci je funkci předán v těle požadavku POST jako hodnota klíče Text. Odpovědí je pak pole objektů. Detekovaných jazyků může být více v jednom požadavku. Objekt obsahuje detekovaný jazyk, informace, zda je služba schopna poskytnutý text přeložit či převést do jiné znakové sady a pravděpodobnost detekce jazyka. Odpověď dále disponuje polem alternativních detekovaných jazyků, jež obsahuje objekty ze stejnými informacemi jako primární detekovaný jazyk. [39]

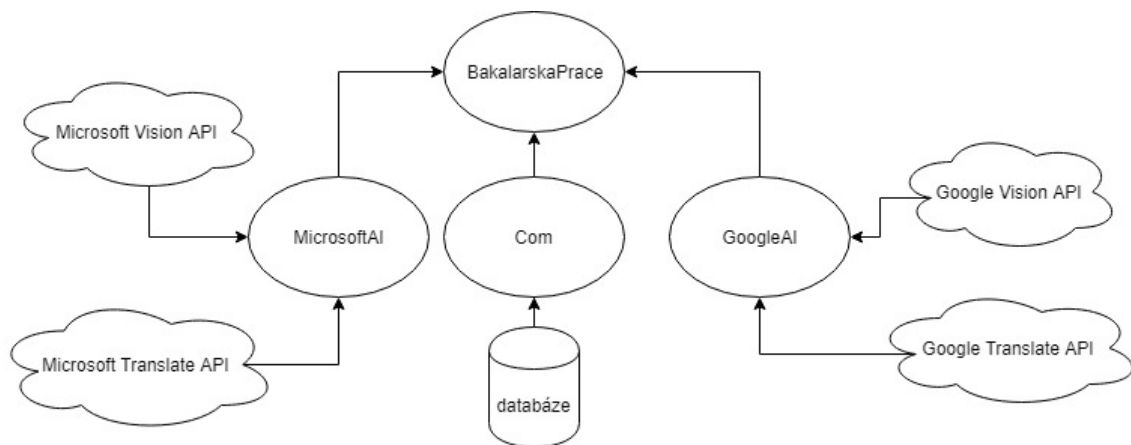
```
[
  {
    "language": [DETEKOVANÝ_JAZYK],
    "score": [PRAVDĚPODOBNOST_URČENÍ_JAZYKA],
    "isTranslationSupported": [JE_MOŽNÉ_PROVÉST_PŘEKLAD],
    "isTransliterationSupported": [JE_MOŽNÉ_PŘEVÉST_DO_JINÉ_ZNAKOVÉ_SADY],
    "alternatives": [
      [ALTERNATIVNÍ_DETEKOVANÉ_JAZYKY_SE_STEJNÝMI_PARAMETRY_JAKO_PRIMÁRNÍ_DETEKCE]
    ]
  }
]
```

## 4 PRAKTICKÉ UKÁZKY VYBRANÝCH SLUŽEB

Tato kapitola se zabývá analýzou dat získaných pomocí vytvořené aplikace. Jsou využity služby pro překlad textu a počítačové zpracování obrazu společností Google a Microsoft. Na základě získaných dat jsou jednotlivé funkce vyhodnoceny z hlediska jejich fungování a také srovnány.

### 4.1 Popis programového vybavení

Ukázkové příklady spolu s praktickou částí této práce jsou vyvinuty pomocí platformy ASP.NET Core. Aplikace je členěna do čtyř samostatných projektů. Důvodem je větší přehlednost a možnost implementace vytvořených funkcí v jiném řešení bez nutnosti většího zásahu. Program je rozdělen následovně.



Obrázek 4 Schéma architektury aplikace

#### **BakalarskaPrace**

Jedná se o webovou aplikaci ASP.NET Core, jež implementuje vzor MVC. Je využita jako koncový bod aplikace. Obsahuje tři controllery a to CarController, GoogleController a MicrosoftController.

#### **CarController**

- Zobrazení Indexu (titulní stránky aplikace)
- Detekce automobilu
- Detekce tisíce automobilů (test funkčnosti detekce automobilu a SPZ)
- Detekce státní poznávací značky

### GoogleController

- Detekce objektů
- Detekce známých míst
- Detekce dominantních barev

### MicrosoftController

- Detekce dominantních barvy (dominantní barva pozadí a popředí)
- Popis obrázku v celých větách
- Detekce objektů

## 4.2 Počítačové zpracování obrazu

Tato část práce je věnována ověření správné funkčnosti vybraných funkcí pro počítačové zpracování obrazu. Jsou využity funkce pro detekci objektů, detekci dominantní barvy, popis obrázku v celých větách v lidsky čitelné formě a jsou využity služby k vyškolení vlastních modelů pro kategorizaci obrázků.

### 4.2.1 Popis obslužných metod

Metoda `AnalyzeRemoteImageAsync` objektu `MsVision` umístěná v projektu `MicrosoftAI`, slouží k získání analýzy obrázku od společnosti Microsoft.

```
public async Task<ImageAnalysis> AnalyzeRemoteImageAsync(
    string url, List<VisualFeatureTypes> features)
{
    ComputerVisionClient computerVision = new ComputerVisionClient(
        new ApiKeyServiceClientCredentials(subscriptionKey),
        new System.Net.Http.DelegatingHandler[] { });
    computerVision.Endpoint = endPoint;

    return await computerVision.AnalyzeImageAsync(url, features);
}
```

Jedná se o asynchronní metodu, která přebírá URL adresu zkoumaného obrázku a list `VisualFeatureTypes` pro definování funkce, kterou má služba provést. Metoda dále vytváří nový objekt `ComputerVisionClient`, jemuž je předán API klíč, obslužné rutiny HTTP protokolu a adresa koncového bodu služby. Poté je proveden dotaz na službu, která provede příslušnou



funkci pomocí metody `AnalyzeImageAsync` objektu `ComputerVisionClient`. Po provedení požadavku tato metoda vrací výsledek analýzy jako objekt `ImageAnalysis`.

V případě služby společnosti Google je využito popsané rozhraní API bez využití knihovny. K provedení analýzy slouží metoda `GetData` objektu `Vision`, který je umístěn v projektu `GoogleAI`. Tato metoda vytvoří tělo požadavku pomocí metody `prepareRequestJson` a poté provede požadavek.

#### 4.2.2 Detekce objektů

U služeb pro počítačové zpracování obrazu byly využity funkce pro detekci objektů. Analýza byla provedena na deseti fotografiích. U každé z fotografií byl zaznamenán název objektu, jež služba detekovala s nejvyšší pravděpodobností. Pravděpodobnost detekce těchto objektů se vždy pohybovala okolo 99 %. Výsledek testu je uveden v tabulce níže.

Číslo obrázku	Microsoft Vision	Google Vision
1	Tree	Tree
2	Grass	Home
3	Grass	Sports
4	Couch	Room
5	Road	Land vehicle
6	Snow	Mountainous landforms
7	Sky	Landmark
8	Outdoor	Landmark
9	Cup	Caffè macchiato
10	Sofa	Couch

Tabulka 1 Výsledky detekce objektů

Jak je patrné z tabulky, služba společnosti Google dokázala objekty ve fotografii detekovat mnohem lépe. Toto tvrzení je demonstrováno na detailnější analýze fotografií, jež jsou v tabulce označeny číslem čtyři a pět.



Obrázek 5 Ukázkový obrázek pokoje [44]

Služba společnosti Google na obrázku číslo čtyři detekovala s největší pravděpodobností Room. Služba společnosti Microsoft pak detekovala Couch. Je patrné, že služba Googlu lépe pochopila kontext fotografie, přičemž služba Microsoftu se zaměřila na detail tedy gauč. V reálném provozu, kdy je po službě vyžadováno, aby co nejlépe popsala obrázek a objekty v něm je lepší služba Google, která správně dokázala určit, že se jedná o místnost, a ne pouze jen o gauč.



Obrázek 6 Ukázkový obrázek auta [45]

Na tomto obrázku služba společnosti Google detekovala Land vehicle. Služba společnosti Microsoft detekovala Road. Znovu je zde viditelná převaha služby Googlu. Tato služba správně rozpoznala dominantní objekt na fotografii, tedy auto, a poskytla příslušný název objektu. Služba Microsoftu s nejvyšší pravděpodobností detekovala silnici, která je v tomto případě naprosto nepodstatná.

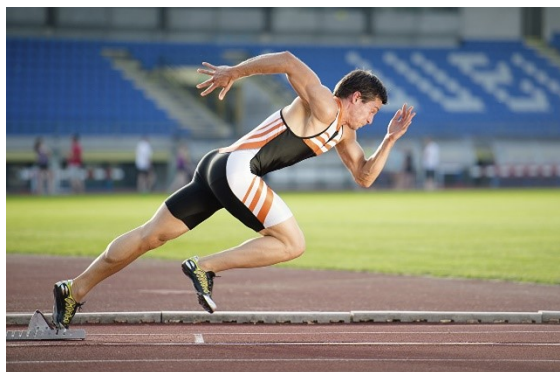
### 4.2.3 Popis obrázku v celých větách

Služba společnosti Microsoft disponuje funkcí pro popis obrázku v celých větách v anglickém jazyce. Znamená to, že je službě poskytnut obrázek k analýze. Služba následně poskytne popis obrázku v lidsky čitelné formě. Test této funkce byl proveden na deseti fotografiích. Výsledek testu je uveden v tabulce níže.

Číslo obrázku	Věta popisující obrázek	Pravděpodobnost
<b>1</b>	a large tree in a field	96 %
<b>2</b>	a house with a lawn in front of a building	97 %
<b>3</b>	a person jumping in the air	77 %
<b>4</b>	a hotel room	99 %
<b>5</b>	a car parked on a city street	97 %
<b>6</b>	a view of a snow covered mountain	96 %
<b>7</b>	a large tall tower with a clock at the top of Eiffel Tower	68 %
<b>8</b>	a large clock tower towering over Big Ben	93 %
<b>9</b>	a cup of coffee on a table	78 %
<b>10</b>	a green sofa in a room	86 %

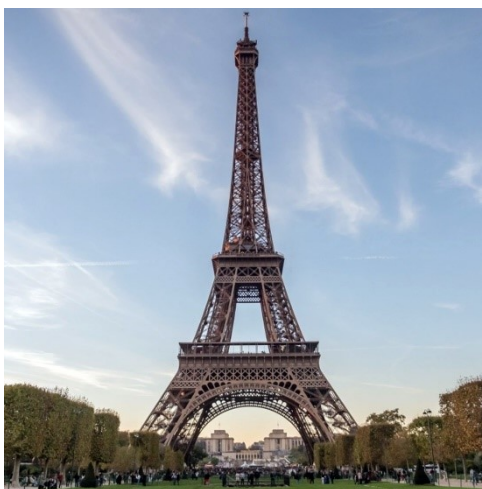
Tabulka 2 Výsledky vytvořených vět popisujících obrázků

V případě fotografie číslo jedna, na které se nachází strom, služba vytvořila správný popis obrázku, a to i v případě obrázku šest, kde se na obrázku skutečně nacházejí hory, obrázku devět, na kterém se nachází šálek kávy položený na stole nebo obrázku pět vyobrazujícím zaparkované auto. Horší popis provedla u obrázků tři a sedm.



Obrázek 7 Ukázkový obrázek běžce [46]

Na obrázku, jež je v tabulce označen číslem tři, je zachycen běžec při startu. Služba však tento obrázek popsala jako člověka, který skáče do vzduchu. Na první pohled se může jevit, že muž na obrázku skutečně vypadá jako by byl zachycen ve skoku. Nicméně po bližším prozkoumání zjistíme, že se jedná o startujícího běžce. K tomuto tvrzení nám může napomoci např. to, že na obrázku je viditelný startovací blok nebo to, že jsou viditelné vodící čáry trati a obrázek je pořízen a stadionu. Služba sice predikovala tento popis na 77 %, ale i přesto popis neodpovídá realitě.













Obrázek 8 Ukázkový obrázek Eiffelovy věže [47]

V případě obrázku, v tabulce označený číslem sedm, služba vytvořila tuto větu: velká vysoká věž s hodinami v horní části Eiffelovy věže, a to s pravděpodobností 68 %. Služba dokázala správně určit, že se jedná o Eiffelovu věž, ale také do popisu přidala že se v horní části věže nacházejí hodiny což není pravda. Hodiny se na vrcholu Eiffelovy věže nenachází ani reálně, ani nejsou tyto hodiny patrné na fotografii. Z těchto důvodů lze tvrdit, že popis odpovídá jeho pravděpodobnosti určení.

#### 4.2.4 Dominantní barvy na obrázku

U obou služeb byla využita funkce k detekci dominantní barvy. V tomto případě dokáže služba Microsoftu poskytnout více informací. Nejen, že dokáže detekovat dominantní barvu celého obrázku jako je tomu u služby společnosti Google, ale také dokáže detekovat dominantní barvu popředí, pozadí a celého obrázku. Další odlišností je, že služba Microsoftu poskytne detekované barvy jako anglickě slovo, kdežto služba Googlu pak jako hodnoty složek RGB. Z tohoto důvodu jsou barvy v tabulce s výsledky služby společnosti Googlu uvedeny jako hexadecimální číslo. Test této funkce byl proveden na deseti fotografiích. Výsledky testů jsou uvedeny níže v tabulkách.

Google Vision		
Číslo obrázku	Dominantní barva (HEX)	Pravděpodobnost určení
1	#6E8037 	35 %
2	#704A38 	15 %
3	#C8D283 	28 %
4	#A4967A 	26 %
5	#D1CB3D 	17 %
6	#507FAC 	35 %
7	#ACC0D8 	37 %
8	#8D9DB1 	21 %
9	#CACAC8 	19 %
10	#617D7E 	41 %

Tabulka 3 Výsledky detekce dominantní barvy společností Google

Microsoft Vision			
Číslo obrázku	Dominantní barva		
	popředí	pozadí	Celého obrázku
1	Green	Blue	Green
2	White	Green	Green
3	Gray	Gray	Grey
4	Black	Black	black
5	Gray	Black	Black
6	White	White	White
7	White	White	White
8	Brown	Grey	Brown
9	White	Brown	Brown
10	Grey	White	Grey

Tabulka 4 Výsledky detekce dominantních barev společnosti Microsoft

Na obrázcích, jež jsou označeny v tabulkách číslem jedna a deset je provedena podrobnější analýza a zjištění, zda služby dokázali dominantní barvy určit správně.



Obrázek 9 Ukázkový obrázek stromu [48]

V případě této fotografie označené číslem jedna služba Googlu špatně detekovala dominantní barvu. Detekovaná barva nijak neodpovídá realitě. V případě služby společnosti Microsoft jsou detekované barvy poměrně přesné. Dominantní barvu popředí služba správně

detekovala jako zelenou. Dominantní barvu pozadí také správně detekovala jako modrou a celkovou dominantní barvou je podle služby zelená, což odpovídá realitě.



Obrázek 10 Ukázkový obrázek pohovky [49]

V tomto případě služba společnosti Google dokázala přesně detekovat dominantní barvu. Nevýhodou služby Microsoftu je, že poskytuje informace o barvě pouze slovně. Pro druhý analyzovaný obrázek služba detekovala jako dominantní barvu šedou, což není úplně špatně, nicméně v tomto případě je výkonnější služba společnosti Google, jež dokázala dominantní barvu včetně odstínu určit přesně. Microsoft dále detekoval dominantní barvu pozadí jako bílou což je správně.

#### 4.2.5 Školení vlastního modelu

Kategorie	Počet fotografií
Interiér	249
Pravý bod	103
Levý bod	103
Přední část	313
Zadní část	267

Tabulka 5 Rozdělení obrázku do kategorií

Funkčnost služeb Google AutoML Vision a Microsoft Custom Vision sloužících pro vyškolení vlastních modelů k detekci objektů na fotografii je demonstrována na vyškolení modelu, určeného k zjištění, z jaké strany je vyfocen automobil. Hlavním cílem školení těchto modelů je s jistotou detekovat přední část automobilu. Nejprve je nutné definovat o jaké pozice jde. Je zvoleno pět základních pozic, a to přední část, pravý bok, levý bok, zadní část a interiér automobilu. K výcviku modelu bylo využito celkem 1035 fotografií. Tyto fotografie byly následně rozřazeny do příslušných kategorií. Rozložení mezi těmito kategoriemi je uvedeno v následující tabulce.

U obou služeb byl model školen 1 hodinu. Po vyškolení jsou oba modely u jednotlivých společností otestovány. Tento test je proveden na pěti fotografiích, na kterých je automobil vyfocen ze všech stran včetně interiéru.



Obrázek 11 Přední část automobilu

Na této fotografii je automobil vyfocen zepředu. V následující tabulce jsou uvedeny pravděpodobnost detekce jednotlivých kategorií obou služeb.

<b>Kategorie</b>	<b>Google AutoML Vision</b>	<b>Microsoft Custom Vision</b>
Přední část	99 %	100 %
Zadní část	0 %	0 %
Levý bok	0 %	0 %
Pravý bok	0 %	0 %
Interiér	0 %	0 %

Tabulka 6 Výsledky detekce přední části automobilu



V tomto případě služby správně detekovali, že se jedná o přední část auto v obou případech. Z tabulky je patrné, že služba společnosti Microsoft je v tomto případě přesnější. Na následující fotografii je vyobrazena zadní část automobilu.



Obrázek 12 Zadní část automobilu

Kategorie	Google AutoML Vision	Microsoft Custom Vision
Přední část	0 %	0 %
Zadní část	100 %	99 %
Levý bok	0 %	0 %
Pravý bok	0 %	0 %
Interiér	0 %	0 %

Tabulka 7 Výsledky detekce zadní části automobilu

V tomto případě jsou také obě služby při predikci zadní strany automobilu úspěšné. Vyhodnocení jednotlivých modelů je srovnatelné. Na následující fotografii je automobil vyfocen z levé strany.



Obrázek 13 Levý bok automobilu

Kategorie	Google AutoML Vision	Microsoft Custom Vision
Přední část	0 %	0 %
Zadní část	0 %	0 %
Levý bok	64 %	1 %
Pravý bok	53 %	98 %
Interiér	0 %	0 %

Tabulka 8 Výsledky detekce levé části automobilu

V případě fotografie levého boku automobilu nebyla služba společnosti Google schopna jednoznačně detekovat o kterou stranu automobilu se jedná. Služba v tomto případě dokáže rozlišit mezi přední či zadní částí auta od levého boku, ale nedokáže rozlišit mezi levým a pravým bokem automobilu. Služba společnosti Microsoft nedokázala správně určit, že se v tomto případě jedná o levou stranu automobilu a detekovala stranu pravou. Na následující fotografii je automobil vyfotografován z pravé strany.

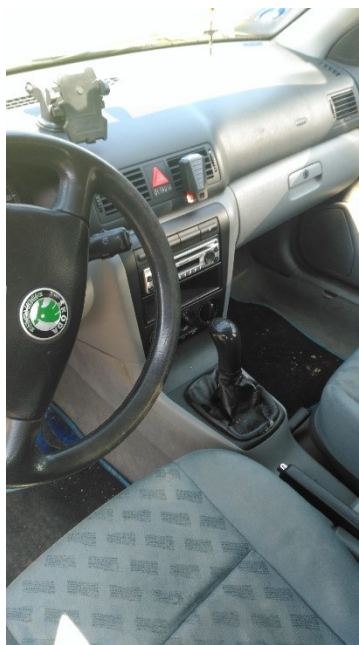


Obrázek 14 Pravý bok automobilu

Kategorie	Google AutoML Vision	Microsoft Custom Vision
Přední část	0 %	0 %
Zadní část	0 %	0 %
Levý bok	30 %	1 %
Pravý bok	73 %	98 %
Interiér	0 %	0 %

Tabulka 9 Výsledky detekce pravé části automobilu

V případě fotografie pravého boku je služba společnosti Microsoft schopna detekovat tuto část auta na 98 %, což je více než dostačující. Vyškolený model u společnosti Google v tomto případě dokázal detekovat pravý bok automobilu na 73 %. Nicméně služba detekovala levý bok na 30 %, což je nedostačující a jako v přechodím případě nebyla schopna jednoznačně rozlišit mezi levou a pravou stranou automobilu. Na následující fotografii je interiér automobilu.



Obrázek 15 Interiér automobilu

Kategorie	Google AutoML Vision	Microsoft Custom Vision
Přední část	0 %	0 %
Zadní část	0 %	0 %
Levý bok	0 %	0 %
Pravý bok	0 %	0 %
Interiér	100 %	100 %

Tabulka 10 Výsledky detekce interiéru automobilu

V tomto případě dokázali obě služby správně detekovat interiér automobilu bez detekce ostatních kategorií.

Vycvičené modely obou služeb dokázali s jistotou predikovat přední a zadní část automobilu i interiér. V případě bočních částí automobilu služba od společnosti Google nebyla schopna jednoznačně rozlišit levou a pravou stranu a pravděpodobnost určení u těchto fotografií je rozložena mezi tyto kategorie. Služba společnosti Microsoft v případě bočních stran automobilu dokázala správně rozlišit pouze pravou stranu. Toto nesprávné detekování stran automobilu by mohlo být způsobeno malým počtem fotografií u těchto kategorií v tréninkové sadě. Nicméně pro stanovený účel, jež je rozlišit přední část automobilu od ostatních částí tyto vycvičené modely postačují. Provedeným otestováním bylo dále zjištěno, že v případě vycvičení modelu za účelem kategorizace fotografií je služba Custom Vision společnosti Microsoft výkonnější.

### 4.3 Překlad textu

Pro ověření správnosti překladu textu jsou využity věty v anglickém jazyce. Tyto věty jsou následně odeslány k analýze dvou popsanych služeb určených k překladu textu. Překlad je proveden pomocí neuronového strojového překladu (NMT). Následné výsledky jsou porovnány s lidsky provedeným překladem. Popis obslužných tříd

Pro účely překladu textu zprostředkované společností Google je vytvořena třída Translate v projektu GoogleAI.

```
public string TranslateText(string languageCode, string text,
                           TranslationModel model)
{
    TranslationClient client = TranslationClient.Create(model: model);
    TranslationResult result = client.TranslateText(text, languageCode);
}
```

```
    return result.TranslatedText;
}

public string TranslateHtml(string languageCode, string html,
                           TranslationModel model)
{
    TranslationClient client = TranslationClient.Create(model: model);
    TranslationResult result = client.TranslateHtml(html, languageCode);

    return result.TranslatedText;
}

public Detection DetectLanguage(string text, TranslationModel model)
{
    TranslationClient client = TranslationClient.Create(model: model);
    Detection result = client.DetectLanguage(text);

    return result;
}
```

Třída Translate disponuje těmito třemi metodami:

- TranslateText – překlad textu ve tvaru řetězce
- TranslateHtml – překlad HTML souboru
- DetectLanguage – metoda sloužící pro detekci zdrojového jazyka textu

Všechny uvedené metody nejprve vytvoří novou instanci objektu TranslationClient pomocí volání metody Create. Jako parametr této metodě předávají objekt TranslationModel pro definici modelu, který bude použit k překladu. Následně podle zvolené funkce jsou volány příslušné metody. Metodám TranslateText a TranslateHtml jsou jako první parametr předány texty k překladu a jako druhý parametr kód cílového jazyka. Jejich výstupem je pak objekt TranslationResult. V případě detekce zdrojového jazyka je metodě DetectLanguage objektu TranslationClient předán pouze jeden parametr, a to text pro detekci. Výstupem metody je objekt Detection.

Pro účely komunikace se službou společnosti Microsoft je vytvořena třída MsVision v projektu MicrosoftAI.

```
public class MsTranslate : IMsTranslate
{
    private string host = "https://api.cognitive.microsofttranslator.com";
    private string subscriptionKey = "77e5dd7f341e463bb798af967394e813";
    private string queryTranslate = "/translate?api-version=3.0";
    private string queryDetect = "/detect?api-version=3.0";

    public string Translate(string text, string[] translateLanguages)
    {
        foreach(var translateLanguage in translateLanguages)
        {
            queryTranslate += string.Concat("&to=", translateLanguage);
        }
    }
}
```

```
    }

    var requestBody = JsonConvert.SerializeObject(
        new Object[] { new { Text = @text } });

    using (var client = new HttpClient())
    using (var request = new HttpRequestMessage() {
        Method = HttpMethod.Post,
        RequestUri = new Uri(host + queryTranslate),
        Content = new StringContent(requestBody, Encoding.UTF8, "application/json")
    })
    {
        request.Headers.Add("Ocp-Apim-Subscription-Key", subscriptionKey);
        var response = client.SendAsync(request).Result;
        return response.Content.ReadAsStringAsync().Result;
    }
}

public string DetectLanguage(string text)
{
    var requestBody = JsonConvert.SerializeObject(
        new Object[] { new { Text = @text } });

    using (var client = new HttpClient())
    using (var request = new HttpRequestMessage()
    {
        Method = HttpMethod.Post,
        RequestUri = new Uri(host + queryDetect),
        Content = new StringContent(requestBody, Encoding.UTF8, "application/json")
    })
    {
        request.Headers.Add("Ocp-Apim-Subscription-Key", subscriptionKey);
        var response = client.SendAsync(request).Result;
        return response.Content.ReadAsStringAsync().Result;
    }
}
}
```

Třída je tvořena ze dvou metod. Metoda Translate, sloužící pro překlad textu, přebírá jako svůj vstupní parametr text pro překlad datového typu string a kod cílového jazyka je definován jako pole datového typu string. Jelikož jsou informace o cílovém jazyce službě poskytnuty jako parametry URL adresy a těchto jazyků může být v jednom požadavku více, program nejprve sestaví požadované parametry o jazyce, které označí jako „to“. Poté tyto parametry spojí s adresou na koncový bod služby, která je uvedena jako privátní proměnná třídy se jménem queryTranslate. Následně je text převeden do formátu JSON. Poté je vytvořen nový HttpClient a nový http požadavek. Požadavku je definován jako POST, je mu předána URL adresa služby a do jeho tělu přiřazen vytvořený JSON. Poté je do hlavičky požadavku přidán API klíč a požadavek je odeslán ze zpracování. Po přijetí odpovědi jsou data v jejím těle převedeny do datového typu string z níž je vybrána proměnná Result a vrácena jako odpověď na volání této třídy.

Metoda DetectLanguage sloužící o určení zdrojového jazyka textu. Funguje obdobným způsobem. Jediným rozdílem je zde jiná adresa koncového bodu služby a absence parametrů pro definici cílového jazyka překladu.

### 4.3.1 Analýza výsledků a srovnání služeb

Pro analýzu funkčnosti těchto překladových služeb bylo použito dvou vět z anglické učebnice v anglickém jazyce. Tyto věty jsou následně poskytnuty službám k překladu do českého jazyka. Výsledky jsou porovnány s lidsky provedeným překladem.

#### Věta číslo jedna

„Conductor William Cristie stopped the performance, turned, pointed at the phone owner, and **shouted**, 'Out! You have just ruined one of the most beautiful **passages** of one of the most beautiful works ever written!'.“

#### Lidský překlad

Dirigent William Cristie přerušil představení, otočil se, ukázal na majitele telefonu a zařval: „Ven! Zrovna jste zkazil jednu z nejkrásnějších **pasáží** jednoho z nejkrásnějších děl, které kdy bylo **napsáno!**“

#### Překlad Společností Google

Dirigent William Cristie zastavil představení, otočil se, ukázal na majitele telefonu a vykřikl: Ven! Právě jste zničili jednu z nejkrásnějších **pasáží** jednoho z nejkrásnějších děl, která kdy **byla** **napsána**.

#### Překlad společností Microsoft

Dirigent William CRISTIE ukončil představení, otočil se, ukázal na majitele telefonu a vykřikl: Právě jsi zničil jednu z nejkrásnějších **chodeb** jedné z nejkrásnějších děl, která **byla** **kdy** **napsána**.

U překladu společnosti Microsoft vidíme, že slovo „passages“ detekoval mylně a ačkoli v jiném významu může znamenat chodby, zde tomu tak není. Když pomineme skloňování služeb v poslední části věty, můžeme říci, že obě služby přeložily text obstojně, avšak služba Googlu lépe.

### Věta číslo dvě

„New or inexperienced users will love the brand new, friendly interface which allows anyone to get productive even more rapidly than before, and the re-designed Help menu comes with wizards and tutorials that explain complex procedures better and more clearly than any other image-editing program on the market.“

### Lidský překlad

Noví nebo nezkušení uživatelé **si zamilují** toto nové, přátelské prostředí, které umožňuje komukoli **se stát produktivním** ještě rychleji než předtím a nově navržené menu nápovědy přichází s průvodci a tutoriály, jež vysvětlují složité postupy lépe a jasněji než jakýkoli jiný **program na úpravu fotek** na trhu.

### Překlad společností Google

Noví nebo nezkušení uživatelé **budou milovat** zbrusu nové, přátelské rozhraní, které umožní komukoli **získat produktivitu** ještě rychleji než dříve, a přepracované menu Nápověda je dodáváno s průvodci a tutoriály, které vysvětlují složitější postupy lépe a jasněji než jakékoli jiné **úpravy obrázků programu** na trhu.

### Překlad společností Microsoft

Noví nebo nezkušení uživatelé **budou milovat** zcela nové, přátelské rozhraní, které umožní komukoli **zvýšit produktivitu** ještě rychleji než dříve, a nově navržená nabídka nápovědy je dodávána s průvodci a výukovým postupem, který objasní složité postupy lépe a jasněji než jakýkoli jiný **program pro úpravu obrázků** na trhu.

Druhá věta byla pro služby připravena mnohem komplikovanější, ale i zde můžeme říci, že význam zůstal zachován. V překladu lidským faktorem můžeme vidět „lidštější“ spojení „si zamilují“, kdežto obě služby zvolily poněkud kostrbaté „budou milovat“. Fráze „to get productive“ je přeložena třikrát jinak, a přesto ani jeden z překladů není nemístný, což se nedá říci „úpravy obrázků programu“ z překladu Googlu. V tomto případě byl tedy překlad služby Microsoft vhodnější.



## **II. PRAKTICKÁ ČÁST**

Praktická část práce se zabývá popisem vytvořené aplikace k detekci automobilu na fotografii pomocí obou popsaných služeb pro počítačové zpracování obrazu. Následně jsou tyto služby porovnány na základě získaných dat a je vybrána ta s lepšími výsledky. Poté je vybraná služba využita pro detekci a skrytí státní poznávací značky. Následně je tato aplikace podrobena několika testům funkčnosti a vyhodnocena. Poslední část je věnována popisu praktického využití aplikace a popisu dalších možných implementací.

## 5 POPIS PROGRAMU PRO DETEKCI AUTOMOBILU

Na službu pro počítačové zpracování obrazu jsou vzneseny tyto požadavky. Služba by měla být schopna detekovat automobil na obrázku, který má minimální velikost 400 x 400 *px*. Detekce by měla být provedena alespoň z 90 % pravděpodobností a automobil by měl být detekován, ať už je vyfotografován z jakéhokoliv úhlu.

Pro tyto účely jsou v programu vytvořeny metody `IfCarReturnScore` pro obě popsané služby společností Google a Microsoft v příslušných projektech. Tyto metody jsou co do funkčnosti totožné, proto je následující popis proveden pouze na jedné z nich, a to na metodě využívající službu společnosti Google.

```
public float? IfCarReturnScore(Dictionary<string, float> labels)
{
    List<string> carLabels = new List<string>() {
        "land vehicle", "vehicle", "car"
    };

    List<string> undesirableLabels = new List<string>()
    {
        "toy", "parking lot", "Toy vehicle", "Car dealership", "parking"
    };

    foreach (var undesirableLabel in undesirableLabels)
    {
        foreach (var label in labels)
        {
            if (undesirableLabel == label.Key.ToLower()) return null;
        }
    }

    foreach (var label in labels)
    {
        foreach (var carLabel in carLabels)
        {
            if ((carLabel == label.Key.ToLower()) && (label.Value > 0.90))
                return label.Value;
        }
    }

    return null;
}
```

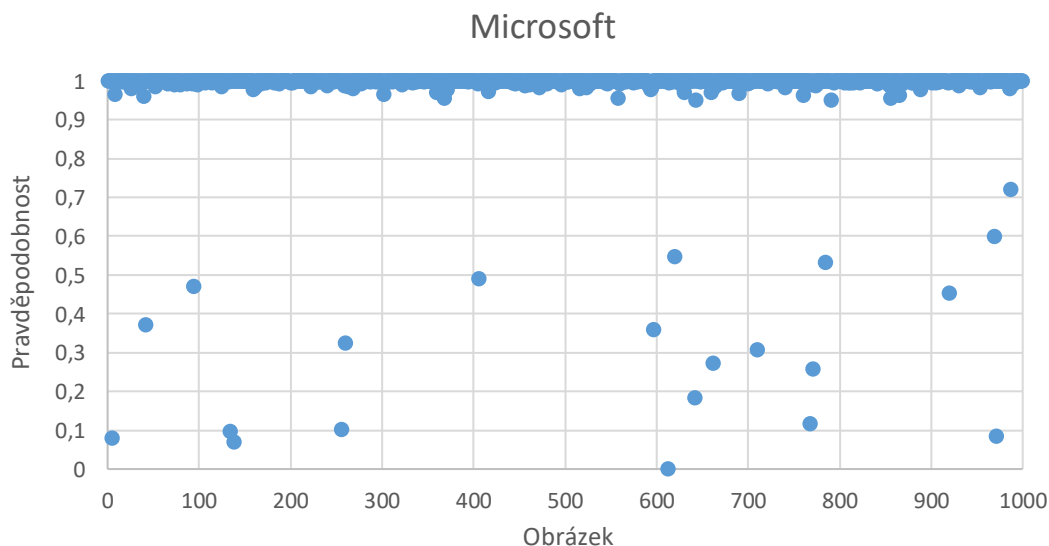
Metoda přebírá detekované objekty na fotografii spolu s jejich pravděpodobností určení. Metody určené pro detekci jsou popsány v kapitole 4.2.2. Tato metoda nejprve vytvoří dva listy datového typu `string`. Ty slouží pro definici vhodných a nevhodných štítků. Z předběžné analýzy fotografií je zjištěno, že služby při detekci automobilu nejčastěji vracejí kategorie `land vehicle`, `vehicle` a `car`. Ty byli zahrnuty do skupiny akceptovaných štítků. Při analýze nevhodných fotografií, na kterých se však automobil, či více automobilů nachází, služby

detekují kategorie toy, parking lot, toy vehicle či car dealership. Ty byli zahrnuty do skupiny nežádoucích kategorií.

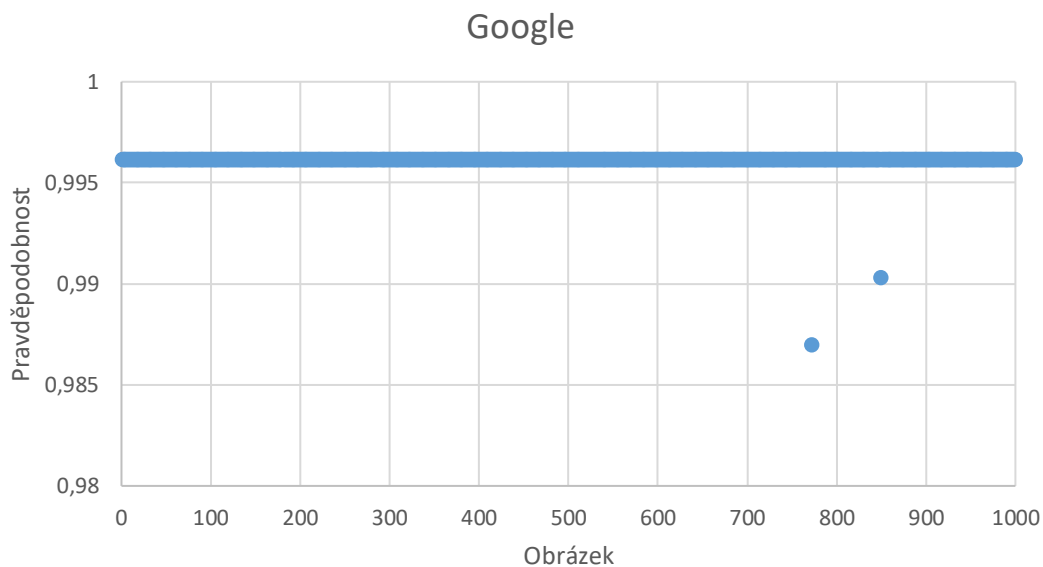
Metoda projde detekované kategorie a porovná je s nežádoucími. Pokud narazí na shodu nepokračuje dál a vrací null. V opačném případě pokračuje porovnáním vyhovujících kategorií. Při shodě metoda vrací pravděpodobnost, s jakou byla kategorie detekována. Z důvodu analýzy výkonosti obou služeb byla podmínka alespoň 90 % pravděpodobnosti určení dané kategorie pro účely analýzy vyřazena. Ve finálním programu pak metoda spolu s akceptovanými kategoriemi porovnává, zda je jejich pravděpodobnost určení vyšší než 90 %. V případě nalezení shody s akceptovanými štítky vrací metoda pravděpodobnost určení.

## 5.1 Výběr vhodné služby

Pomocí popsaných metod byla provedena analýza na tisíci fotografiích automobilů. Při této analýze byla zaznamenána pravděpodobnost, s jakou byl automobil detekován. Následně byly vytvořeny grafy závislosti pravděpodobnosti na jednotlivých snímcích pro obě služby, které jsou uvedeny níže.



Graf 1 Pravděpodobnost detekce automobilu společností Microsoft



Graf 2 Pravděpodobnost detekce automobilu společností Google

Z grafů je zřetelně viditelná převaha služby společnosti Google. Detekce automobilu byla v drtivé většině provedena s 99,6 % pravděpodobností, což je více než dostačující. Ve dvou případech jsou patrné mírné odchylky, které jsou však v desetinách procent, což v této konkrétní aplikaci nepředstavuje žádný problém. Oproti tomu služba společnosti Microsoft vykazuje velké propady pravděpodobnosti, s jakou detekuje automobil, což je nežádoucí. Z tohoto důvodu je pro finální realizaci aplikace pro detekci auta a detekci státní poznávací značky vybrána služba společnosti Google.

## 5.2 Detekce státní poznávací značky

Pro detekci státní poznávací značky je, jako v případě detekce automobilu, využita služba pro počítačové zpracování obrazu, konkrétně společnosti Google, což vyplývá z předchozí analýzy. V tomto případě je však využita její funkce pro rozpoznání tištěného textu. Jelikož tato funkce detekuje veškeré texty nalezené na fotografii je nutné vytvořit rozlišovací mechanismus kdy se skutečně jedná o poznávací značku a určit, které typy značek budou detekovány. Dle vyhlášky č. 343/2014 Sb. musí státní poznávací značka obsahovat alespoň 5 znaků. Horní hranice počtu znaků je 8. Tyto znaky musí být tvořeny velkými písmeny latinské abecedy nebo arabskými číslicemi. Dále musí značka obsahovat alespoň jednu číslici a jedno písmeno, přičemž jejich umístění v množině znaků poznávací značky je libovolné.

[50]

V realitě se však při registraci vozidel postupuje systematicky a rozmístění znaků libovolné není. První písmeno zleva označuje kraj, kde je automobil registrován. Jedná o druhou pozici levé poloviny značky. Na první pozici levé poloviny je číslice. Třetí pozici může tvořit, jak číslo, tak písmeno, a to dle počtu registrovaných značek v jednotlivých krajích. Kupříkladu při nové registraci vozidla ve Středočeském či Moravskoslezském kraji jsou značky s číslicí na třetí pozici levé strany poznávací značky již všechny registrovány a je tedy nutné nové značky vydávat s písmenem na této pozici. Pravá strana značky je pak v současné době tvořena číslicemi. Dalším aspektem je typ poznávací značky. Klasický osobní automobil je v drtivé většině označen poznávací značkou typu 101. Jedná se o značku o rozměrech 520 x 110 mm, jež obsahuje černé znaky na bílém pozadí. Touto značkou je automobil označen zepředu i zezadu. Dalším typem značek pro osobní automobily je typ 105. Jedná se o značku o rozměrech 280 x 210 mm, taktéž s černými písmeny na bílém pozadí, které také označuje automobil z obou stran. Na rozdíl od značky typu 101, je tato značka dvouřádková. [51]

Jelikož je tato aplikace vyvinuta pro C2C platformu NajdiKaru s.r.o. určenou k prodeji aut, na níž se vyskytují pouze osobní automobily s českými poznávacími značkami, převážně typu 101, je program přizpůsoben těmto potřebám.

```
public SPZ ParseSpz(JObject json)
{
    if (!json["responses"].First.HasValues) return new SPZ();

    bool isNumber = false;
    SPZ spz = new SPZ();

    var properties = json.SelectToken("responses[0].textAnnotations");

    foreach (var property in properties)
    {
        List<Position> position = new List<Position>();
        var description = property["description"].Value<string>().ToCharArray();
        var positions = property.SelectToken("boundingPoly.vertices");

        foreach (var p in positions)
        {
            position.Add(new Position(p["x"].Value<int>(), p["y"].Value<int>()));
        }

        // is left part of spz
        if ((description.Length == 3) &&
            Char.IsNumber(description[0]) && !Char.IsNumber(description[1]))
        {
            spz.Left = new PartOfSPZ();
            spz.Left.Text = new string(description);
            spz.Left.Position = position;
            spz.Left.Region = GetRegion(spz.Left.Text);
            continue;
        }

        // is right part of spz
        if (description.Length == 4)
```

```
{
    foreach (var c in description)
    {
        isNumber = Char.IsNumber(c);
    }

    if (!isNumber) continue;

    if(spz.Left != null)
    {
        var positionX = position[0].x - spz.Left.Position[1].x;
        var positionY = position[0].y - spz.Left.Position[1].y;
        if (positionX < -100 || positionX > 100 || positionY > 20 || positionY < -20)
            continue;
    }

    spz.Right = new PartOfSPZ();
    spz.Right.Text = new string(description);
    spz.Right.Position = position;
    continue;
}
}

return spz;
}
```

Metoda ParseSpz jako svůj vstupní parametr přebírá odpověď služby k detekci textu. Nejprve metoda zkontroluje, zda vstupní parametr obsahuje požadované hodnoty. Dále pokračuje deklarací pomocných proměnných.

K tomuto účelu jsou v programu vytvořeny tři modely SPZ, Position a PartOfSPZ. Jelikož služba text na poznávací značce detekuje jako levou a pravou stranu zvlášť, model SPZ obsahuje právě dvě proměnné, jež jsou datového typu PartOfSPZ. Tento vnořený model uchovává detekovaný text a list datového typu Position pro uložení pozice vrcholů detekovaného textu.

Dále metoda ParseSpz pomocí metody SelectToken objektu JObject vyseparuje všechny detekované texty. Následně tyto texty prochází pomocí cyklu Foreach. U jednotlivých detekovaných textů vybere jejich znění a pozici. Poté nastává samotné určení levé a pravé strany poznávací značky. V případě, že detekovaný text obsahuje tři znaky, přičemž na první pozici je číslice a na druhé pozici písmeno, metoda tento text vyhodnotí jako levou stranu poznávací značky. Informace o tomto textu přiřadí příslušným proměnným modelu SPZ. V případě, že text obsahuje čtyři znaky, které jsou všechny tvořeny číslicemi, metoda tento text vyhodnotí jako pravou stranu poznávací značky. Pokud je při detekci pravé strany značky již detekována strana levá, je ověřeno, zda je detekovaný text na správné pozici. V případě, že se skutečně jedná o pravou část značky jsou informace o ní přiřazeny příslušné proměnné modelu SPZ. Tento model je pak návratovou hodnotou této metody.

Metoda GetRegion slouží k určení kraje, ve kterém je automobil registrován. Přebírá jako svůj vstupní parametr text levé strany poznávací značky. Poté definuje jednotlivé kraje a k nim příslušné kódové označení. Poté ověří shodu mezi těmito kódy a druhým znakem levé strany poznávací značky. V případě shody vrací název kraje. V opačném případě vrací null.

```
private string GetRegion(string spzLeft)
{
    var regions = new Dictionary<char, string>()
    {
        { 'A', "Praha"},
        { 'B', "Jihomoravský kraj"},
        { 'C', "Jihočeský kraj"},
        { 'E', "Pardubický kraj"},
        { 'H', "Královehradecký kraj"},
        { 'J', "Kraj Vysočina"},
        { 'K', "Karlovarský kraj"},
        { 'L', "Liberecký kraj"},
        { 'M', "Olomoucký kraj"},
        { 'P', "Plzeňský kraj"},
        { 'S', "Středočeský kraj"},
        { 'T', "Moravskoslezský kraj"},
        { 'U', "Ústecký kraj"},
        { 'Z', "Zlínský kraj"}
    };

    foreach (var region in regions)
    {
        if (region.Key == spzLeft[1]) return region.Value;
    }
    return null;
}
```

Po správné detekci alespoň jedné strany poznávací značky je pomocí metody DeleteSpz objektu Vision ve fotografii tato část skryta pomocí bílého obdélníku.

```
public async Task DeteleSpz(SpZ spz, HttpResponseMessage response)
{
    Point leftBottom = new Point();
    Point rightBottom = new Point();
    Point rightTop = new Point();
    Point leftTop = new Point();

    if (spz.Left == null && spz.Right == null) return;

    if(spz.Left != null && spz.Right != null)
    {
        leftBottom = new Point(spz.Left.Position[0].x, spz.Left.Position[0].y);
        rightBottom = new Point(spz.Right.Position[1].x, spz.Right.Position[1].y);
        rightTop = new Point(spz.Right.Position[2].x, spz.Right.Position[2].y);
        leftTop = new Point(spz.Left.Position[3].x, spz.Left.Position[3].y);
    }
    else if(spz.Left != null)
    {
        leftBottom = new Point(spz.Left.Position[0].x, spz.Left.Position[0].y);
        rightBottom = new Point(spz.Left.Position[1].x, spz.Left.Position[1].y);
        rightTop = new Point(spz.Left.Position[2].x, spz.Left.Position[2].y);
        leftTop = new Point(spz.Left.Position[3].x, spz.Left.Position[3].y);
    }
    else if(spz.Right != null)
    {

```



```
    leftBottom = new Point(spz.Right.Position[0].x, spz.Right.Position[0].y);
    rightBottom = new Point(spz.Right.Position[1].x, spz.Right.Position[1].y);
    rightTop = new Point(spz.Right.Position[2].x, spz.Right.Position[2].y);
    leftTop = new Point(spz.Right.Position[3].x, spz.Right.Position[3].y);
}
var stream = await response.Content.ReadAsStreamAsync();
var btm = new Bitmap(stream);
Graphics g = Graphics.FromImage(btm);

Point[] curvePoints = { leftBottom, rightBottom, rightTop, leftTop };
g.FillPolygon(Brushes.White, curvePoints);

btm.Save("car.jpg", ImageFormat.Jpeg);
}
```

Metoda DeleteSpz přebírá jako své vstupní parametry model spz a zkoumanou fotografii. Nejprve tato metoda vytvoří čtyři prázdné objekty Point, které slouží k určení souřadnic vrcholů obdélníku. Poté pomocí If-Else stromu rozhodne, zda byla detekována celá poznávací značka nebo pouze její levá či pravá část a naplní objekty Point příslušnými souřadnicemi. Následně získá fotografii v podobě streamu a vytvoří novou bitmapu. Tuto bitmapu použije pro vytvoření nové instance objektu Graphics, jež použije k nakreslení bílého obdélníku na fotografii. Následně tuto bitmapu uloží pod názvem car ve formátu Jpeg do projektu.

## 6 VYHODNOCENÍ VYTVOŘENÉ APLIKACE

Tato kapitola práce se zabývá vyhodnocením vytvořené aplikace. Je rozdělena na dvě podkapitoly, tedy na dva samostatné testy funkčnosti. První z testů se zabývá funkčností detekce automobilu na fotografii. Druhý test je zaměřen na vyhodnocení funkčnosti detekce státní poznávací značky. Nakonec je vyhodnocení shrnuto a jsou navrženy možné implementace.

### 6.1 Detekce automobilu na obrázku

Vyhodnocení vytvořené aplikace pro detekci automobilu na fotografii je rozděleno na dva samostatné testy. První test demonstruje, zda aplikace dokáže na obrázku odlišit osamocený automobil od více automobilů na parkovišti. V druhém testu je na vzorku padesáti fotografií demonstrováno, zda aplikace dokáže odlišit automobil od ostatních objektů.

V prvním testu jsou použity dvě fotografie. Na první fotografii je osamocený automobil, na druhé fotografii je automobil na parkovacím místě obklopený dalšími automobily. V prvním případě je vyžadováno, aby aplikace tuto fotografii vyhodnotila jako automobil. V druhém případě je po aplikaci požadováno, aby vyhodnotila, že se o automobil nejedná.



Obrázek 16 Fotografie jediného automobilu

V prvním případě aplikace skutečně vyhodnotila, že se jedná o automobil. Následující tabulka obsahuje prvních pět kategorií, které služba pro počítačové zpracování obrazu u této fotografie detekovala.

Detekovaná kategorie	Pravděpodobnost určení kategorie
Land vehicle	99,6 %
Vehicle	99 %
Car	98,7 %
Green	90 %
Automotive design	72 %

Tabulka 11 Přehled detekovaných objektů na fotografii s automobilem

Jak se patrně z tabulky první tři detekované kategorie jsou současně kategoriemi zahrnutými do skupiny akceptovaných. Z tohoto důvodu aplikace správně vyhodnotila, že se jedná o automobil.



Obrázek 17 Fotografie automobilů na parkovišti

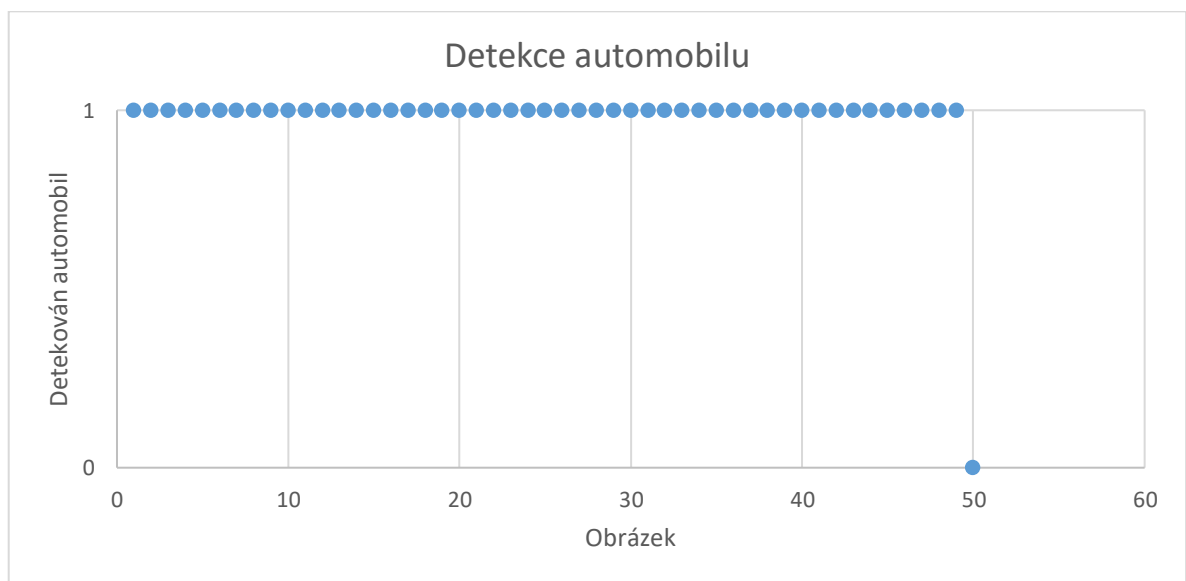
V druhém případě aplikace správně určila, že se o automobil nejedná. V Následující tabulce je uvedeno prvních pět detekovaných kategorií.

Detekovaná kategorie	Pravděpodobnost určení kategorie
Land vehicle	99,6 %
Vehicle	99 %
Car	98,7 %
Parking	91,3 %
Asphalt	85,2 %

Tabulka 12 Přehled detekovaných objektů na fotografii s více automobily

V tomto případě sice služba detekovala akceptované kategorie Land vehicle s nejvyšší pravděpodobností, současně však detekovala i nežádoucí kategorii Parking. Z tohoto důvodu aplikace správně vyhodnotila, že zde o automobil nejde.

V druhém testu byla funkčnost aplikace ověřena na vzorku padesáti fotografií. Na poslední z fotografií není automobil. V následujícím grafu, kde 1 představuje detekci automobilu, je uveden výsledek tohoto testu.



Graf 3 Detekce automobilu

Aplikace správně detekovala všechny automobily. V posledním případě správně rozlišila, že se o automobil nejedná.

Na základě provedených testů je možné s jistotou tvrdit, že aplikace pro detekci automobilu je spolehlivá a je možné ji využít pro účely C2C platformy NajdiKaru s.r.o.

## 6.2 Detekce státní poznávací značky

Pro vyhodnocení funkce pro detekci a skrytí státní poznávací značky jsou provedeny dva testy. V prvním testu je demonstrováno, pod jakým úhlem vyfocení automobilu je služba schopna poznávací značku detekovat a zda dokáže skrýt pouze jednu stranu značky v případě, kdy je značka neúplná. V druhém testu je použit vzorek padesáti aut, na kterém je demonstrováno, ve kterých případech aplikace poznávací značku dokáže detekovat.

Pro první test bylo použito 5 fotografií. Na těchto fotografiích se nachází jeden automobil, který je vyfocen pod různými úhly.



Obrázek 18 Výsledek detekce poznávací značky 1

Na první fotografii je automobil vyfocen přímo zepředu. Aplikace správně detekovala obě strany poznávací značku a tuto značku skryla.



Obrázek 19 Výsledek detekce poznávací značky 2

V druhém případě, kdy je automobil vyfocen pod mírným úhlem aplikace také správně dokázala detekovat obě strany poznávací značky a tu následně skrýt.



Obrázek 20 Výsledek detekce poznávací značky 3

Ve třetím případě služba vůbec nedokázala detekovat text na poznávací značka. Aplikace tedy neměla informace o textu na obrázku a značku nemohla skrýt.



Obrázek 21 Výsledek detekce levé strany poznávací značky

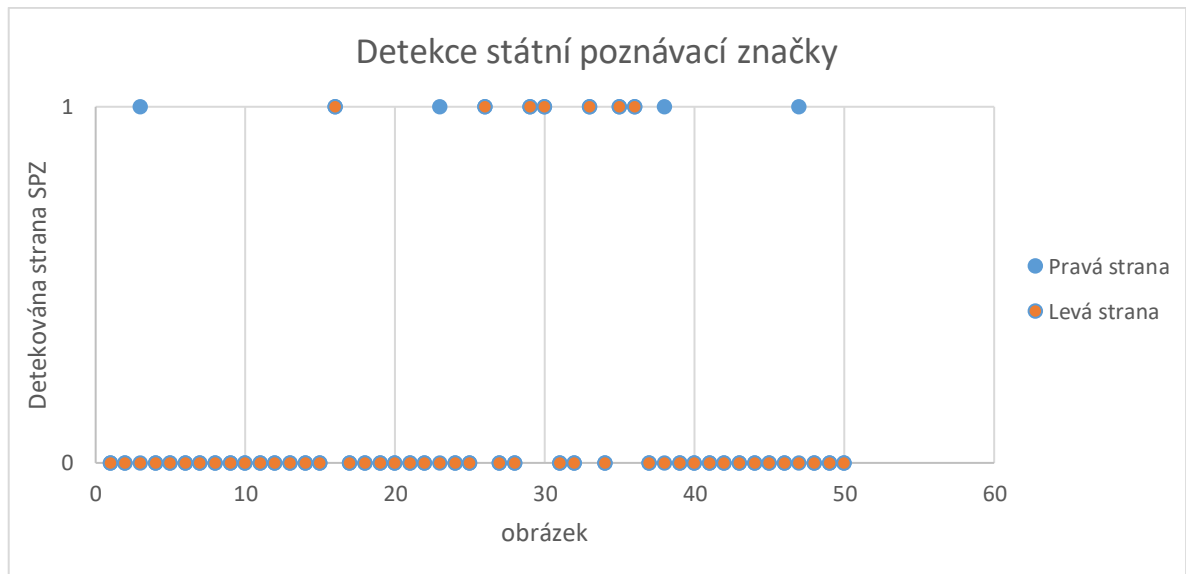
Na čtvrté fotografii je viditelná pouze levá strana poznávací značky. Jak je patrné, aplikace dokázala správně detekovat levou stratu poznávací značky a tu následně skrýt. Jelikož je pravá strana značky neúplná, aplikace ji v tomto případě ignorovala.



Obrázek 22 Výsledek detekce pravé strany poznávací značky

Podobně jako na přechodí fotografii je vidět pouze část automobilu, v tomto případě pravá strana. I v tomto případě aplikace dokázala správně detekovat viditelnou část poznávací značky a tu poté skrýt. Levá část značky je z důvodu neúplnosti ignorována.

V druhém testu je použit vzorek 50 fotografií automobilů. Na tomto vzorku byla provedena detekce poznávací značky. Výsledky detekce jsou uvedeny v grafu níže, kde 1 představuje detekci konkrétní strany poznávací značky.



Graf 4 Detekce státní poznávací značky

Jak je patrné z grafu, funkčnost aplikace pro detekci poznávací značky není ideální. Ve většině případů aplikace značku nedokázala detekovat. Je to způsobeno mnoha faktory. Jedním z faktorů ovlivňující znemožnění detekce jsou špatné světelné podmínky na fotografii. Dalším problémem při detekci je úhel vyfotografování automobilu. Tento problém je demonstrován v prvním testu. V některých případech však nebyl problém na straně aplikace, ale byl způsoben tím, že na fotografii byla poznávací značka již skryta nebo nebyla vůbec přítomna. Důvodem použití i těchto fotografií je demonstrace funkčnosti detekce v reálném provozu.

### 6.3 Celkové zhodnocení a návrh reálného využití

Pomocí provedených testů bylo zjištěno, že aplikace dokáže spolehlivě detekovat automobil na obrázku. Je schopna rozlišit automobil od ostatních objektů a také dokáže na fotografii odlišit více automobilů od jediného. Z těchto důvodů lze tvrdit, že aplikace je plně připravena k implementaci. V případě detekce poznávací značky nelze aplikaci považovat za provozuschopnou v reálném využití. V ideálních podmínkách, kdy je poznávací značka dobře viditelná a na značce se neobjevuje žádný stín je aplikace schopna tuto značku detekovat. V reálných případech však služba pro detekci textu na obrázku nebyla schopna znaky značky identifikovat nebo byla schopna detekovat pouze znaky jedné strany poznávací značky. Možným řešením tohoto problému by mohlo být využití služby AutoML Vision. Pomocí této služby vycvičit model k detekci a určení polohy poznávací značky, kterou by bylo možné dle poskytnuté polohy skrýt.

Jak již bylo zmíněno, aplikace je určena pro C2C platformu NajdiKáru s.r.o. Pomocí této aplikace se na platformě ověřuje, zda se na uživatelsky nahrávaných fotografiích skutečně vyskytuje automobil. V případě, že aplikace automobil nedetekuje je administrátorovi odesláno upozornění prostřednictvím emailu. Dalším využitím této aplikace na platformě NajdiKaru s.r.o. je možnost vymazat poznávací značku. Uživatel na tuto platformu nahraje fotografii svého automobilu. Tato fotografie je následně pomocí aplikace analyzována a v případě detekce alespoň jedné části poznávací značky nabídne uživateli možnost značku skrýt.

Další možnou implementací aplikace by mohlo být umožnění vjezdu aut na privátní parkoviště. Aplikace by při příjezdu automobilu na toto parkoviště detekovala poznávací značku. Tuto SPZ by porovnávala se značkami v databázi, jež mají vjezd povolen a následně by těmto automobilům otevřela závoru. Také by ale dokázala zabránit vjezdu automobilům, jež nemají povolení parkovat v této privátní části.

Další implementací aplikace by mohlo být počítadlo automobilů na veřejném parkovišti. Při vjezdu auta na toto parkoviště by aplikace automobil detekovala a upravila počet volných parkovacích míst na veřejném parkovišti. Počet volných míst by aplikace upravovala i při odjezdu automobilu z parkoviště. Dále by mohla detekovat poznávací značku a zaznamenávat čas příjezdu. S těmito údaji by bylo možné automaticky určit, jak dlouho byl automobil na parkovišti a vypočítat cenu za parkování. Řidič by poté mohl za parkování zaplatit pomocí aplikace nebo kreditní kartou při odjezdu z parkoviště.

Další možností, jak využít aplikaci, by také mohlo být usnadnění průjezdu vozidel integrovaného záchranného systému skrze město. Tento hypotetický scénář předpokládá rozsáhlý kamerový systém umístěn na signalizačních sloupech světelných křižovatek. Aplikace by v reálném čase byla schopna detekovat poznávací značky automobilů a porovnávat je se značkami patřícími automobilům integrovaného záchranného systému. V případě detekování tohoto vozidla by upozornila řidiče, jež brání průjezdu, aby uvolnily prostor, a to vše ještě před tím, než by řidiči stačili zaznamenat blížící se automobil se sirénami. Podobným způsobem by bylo možné realizovat plynulejší průjezd vozidel městské hromadné dopravy.

Další možností využití aplikace by mohlo být automatické hlídání pěších zón, kde je možný vjezd pouze s povolením. V tomto případě by aplikace detekovala poznávací značku a pokud by detekovala nepovolené značky, upozornila by na tento automobil policii. V kombinaci s případnými vysouvacími bariérami, by se dalo i reagovat na nechvalně známý fenomén dnešní doby, a to kdy samozvaný terorista vjede do davu lidí a případně tomu zabránit.



## ZÁVĚR

Cílem této práce bylo analyzovat a porovnat dostupné Cloudové služby využívající ke svému fungování umělou inteligenci. Dále využít vybrané služby a provést demonstrace jejich fungování. Praktická část práce se zabírala vytvořením aplikace, s jejíž pomocí je možné detekovat automobil na poskytnutých obrázcích a na těchto obrázcích se pokusit skrýt státní poznávací značku.

Pro účely analýzy byly využity dostupné služby poskytované společnostmi Microsoft, Google a Amazon. U jednotlivých služeb byl popsán jejich význam, základní vlastnosti a možnosti použití. Následně byly služby těchto společností určeny k řešení stejných problémů, porovnány a byli zmíněny jejich silné a slabé vlastnosti.

Následně byly zvoleny služby pro počítačové zpracování obrazu a překlad společnosti Google a Microsoft. U těchto služeb byla provedena podrobná analýza, výčet všech dílčích funkcionalit a možnosti komunikace s těmito službami pomocí rozhraní REST API. Poté byli zvoleny funkce pro detekci objektů, detekci dominantní barvy, funkce pro popis obrázku v celých větách a překlad, jež byli prakticky využity. Výsledky byly následně analyzovány.

V praktické části práce byla vybrána služba pro počítačové zpracování obrazu společnosti Google pro vytvoření aplikace k detekci automobilů na obrázku. V případě detekce se aplikace pomocí stejné služby pokusila detekovat a skrýt státní poznávací značku. Poslední kapitola práce se zaměřuje na podrobnou analýzu této aplikace a návrhy možností její reálné implementace.

## SEZNAM POUŽITÉ LITERATURY

1. Chapter 10. Neural Networks. *The nature of code* [online]. [cit. 2019-05-01]. Dostupné z: <https://natureofcode.com/book/chapter-10-neural-networks/>
2. MAŘÍK, Vladimír, Jiří LAŽANSKÝ a Olga ŠTĚPÁNKOVÁ. Umělá inteligence 6. Academia, 2013. ISBN 978-80-200-2276-9
3. Perceptron. *Towards Data Science* [online]. [cit. 2019-05-01]. Dostupné z: <https://towardsdatascience.com/perceptron-the-artificial-neuron-4d8c70d5cc8d>
4. Sigmoid Function. *Wolfram Mathworld* [online]. [cit. 2019-05-01]. Dostupné z: <http://mathworld.wolfram.com/SigmoidFunction.html>
5. A Complete Guide to Types of Neural Networks. *Digital Vidya* [online]. [cit. 2019-05-01]. Dostupné z: <https://www.digitalvidya.com/blog/types-of-neural-networks/>
6. The Turing Test, 1950. *The Alan Turing Internet Scrapbook* [online]. [cit. 2019-05-01]. Dostupné z: <https://www.turing.org.uk/scrapbook/test.html>
7. Minds, Brains, and Programs. *Web.archive.org* [online]. 2007 [cit. 2019-05-01]. Dostupné z: <https://towardsdatascience.com/mcculloch-pitts-model-5fdf65ac5dd1>
8. Filosofické problémy umělé inteligence. *Katedra kybernetiky* [online]. [cit. 2019-05-01]. Dostupné z: [http://www.kky.zcu.cz/uploads/courses/zui/filosofie\\_ui\\_v1.pdf](http://www.kky.zcu.cz/uploads/courses/zui/filosofie_ui_v1.pdf)
9. API Design guidance. *Microsoft Docs* [online]. 2018 [cit. 2019-05-01]. Dostupné z: <https://docs.microsoft.com/en-us/azure/architecture/best-practices/api-design>
10. Cloud computing: A complete guide. *The nature of code* [online]. [cit. 2019-05-01]. Dostupné z: <https://www.ibm.com/cz-cs/cloud/learn/what-is-cloud-computing>
11. Co je Azure?. *Microsoft Docs* [online]. [cit. 2019-05-01]. Dostupné z: <https://azure.microsoft.com/cs-cz/overview/what-is-azure/>
12. Co je Azure Machine Learning Studio?. *Microsoft Docs* [online]. [cit. 2019-05-01]. Dostupné z: <https://docs.microsoft.com/cs-cz/azure/machine-learning/studio/what-is-ml-studio>
13. Azure Machine Learning Studio: algorithm and module help. *Microsoft Docs* [online]. [cit. 2019-05-01]. Dostupné z: <https://docs.microsoft.com/cs-cz/azure/machine-learning/studio-module-reference/index>

14. Speech API. *Microsoft Docs* [online]. [cit. 2019-05-01]. Dostupné z: <https://docs.microsoft.com/en-us/azure/cognitive-services/speech-service/overview>
15. LAŽANSKÝ, Jiří, Vladimír MAŘÍK a Olga ŠTĚPÁNKOVÁ. Umělá inteligence 5. Academia, 2007. ISBN 978-80-200-1470-2
16. *Google Cloud Platform (GCP)* [online]. [cit. 2019-05-01]. Dostupné z: <https://searchcloudcomputing.techtarget.com/definition/Google-Cloud-Platform>
17. Basics. *Google Docs* [online]. [cit. 2019-05-01]. Dostupné z: <https://cloud.google.com/speech-to-text/docs/basics>
18. Cloud AutoML. *Google Docs* [online]. [cit. 2019-05-01]. Dostupné z: <https://cloud.google.com/automl/docs/>
19. AutoML Natural Language. *Google Docs* [online]. [cit. 2019-05-01]. Dostupné z: <https://cloud.google.com/natural-language/automl/docs/>
20. Dialogflow: How it works. *Google Docs* [online]. [cit. 2019-05-01]. Dostupné z: <https://cloud.google.com/dialogflow-enterprise/docs/how>
21. Global infrastructure. *Amazon Docs* [online]. [cit. 2019-05-01]. Dostupné z: <https://aws.amazon.com/about-aws/global-infrastructure/>
22. Amazon Forecast. *Amazon Web Services* [online]. [cit. 2019-05-01]. Dostupné z: <https://aws.amazon.com/forecast/>
23. Amazon Polly. *Amazon Web Services* [online]. [cit. 2019-05-01]. Dostupné z: <https://aws.amazon.com/polly/features/>
24. Amazon Textract. *Amazon Docs* [online]. [cit. 2019-05-01]. Dostupné z: <https://docs.aws.amazon.com/textract/latest/dg/what-is.html>
25. Supported Images. *Google docs*. [online]. [cit. 2019-05-01]. Dostupné z: <https://cloud.google.com/vision/docs/supported-files>
26. Quotas and Limits. *Google docs*. [online]. [cit. 2019-05-01]. Dostupné z: <https://cloud.google.com/vision/quotas>
27. Cloud Vision API: Method: images.annotate. *Google Docs* [online]. [cit. 2019-05-01]. Dostupné z: <https://cloud.google.com/vision/docs/reference/rest/v1p4beta1/images/annotate>

28. Cloud Vision API: Method: files.annotate. *Google Docs* [online]. [cit. 2019-05-01]. Dostupné z: <https://cloud.google.com/vision/docs/reference/rest/v1p4beta1/files/annotate>
29. Cloud Vision API: AnnotateImageRequest. *Google Docs* [online]. [cit. 2019-05-01]. Dostupné z: <https://cloud.google.com/vision/docs/reference/rest/v1p4beta1/AnnotateImageRequest>
30. Cloud Vision API: Feature. *Google Docs* [online]. [cit. 2019-05-01]. Dostupné z: <https://cloud.google.com/vision/docs/reference/rest/v1p4beta1/Feature>
31. Cloud Vision API: ImageContext. *Google Docs* [online]. [cit. 2019-05-01]. Dostupné z: <https://cloud.google.com/vision/docs/reference/rest/v1p4beta1/ImageContext>
32. Cloud Vision API: InputConfig. *Google Docs* [online]. [cit. 2019-05-01]. Dostupné z: <https://cloud.google.com/vision/docs/reference/rest/v1p4beta1/InputConfig>
33. Cloud Vision API. *Google Docs* [online]. [cit. 2019-05-01]. Dostupné z: <https://cloud.google.com/vision/docs>
34. Auto ML. *Google Docs* [online]. [cit. 2019-05-01]. Dostupné z: <https://cloud.google.com/vision/automl/docs/before-you-begin>
35. Cognitive Services. *Microsoft Docs* [online]. [cit. 2019-05-01]. Dostupné z: <https://westus.dev.cognitive.microsoft.com/docs/services/5adf991815e1060e6355ad44/operations/56f91f2e778daf14a499e1fe>
36. Microsoft Vision API. *Microsoft Docs* [online]. 2019 [cit. 2019-05-01]. Dostupné z: <https://docs.microsoft.com/cs-cz/azure/cognitive-services/computer-vision>
37. Co je to klipart?. *Shutterstock* [online]. 2019 [cit. 2019-05-01]. Dostupné z: <https://www.shutterstock.com/cs/support/article/Co-je-to-klipart>
38. Cloud Translation API [online]. [cit. 2019-05-01]. Dostupné z: <https://cloud.google.com/translate>
39. Language and region support for the Translator Text API. *Microsoft Docs* [online]. 2019 [cit. 2019-05-01]. Dostupné z: <https://docs.microsoft.com/en-us/azure/cognitive-services/translator>

40. Amazon Rekognition Developer Guide. *AWS Documentation* [online]. ©2018 [cit. 2018-11-01]. Dostupné z: <https://docs.aws.amazon.com/rekognition/latest/dg/rekognition-dg.pdf>
41. Co je počítačové zpracování obrazu?. *Docs.microsoft.com/cs-cz/azure/* [online]. [cit. 2018-11-01]. Dostupné z: <https://docs.microsoft.com/cs-cz/azure/cognitive-services/computer-vision/home>
42. Text detection samples. *Cloud.google.com* [online]. [cit. 2018-11-01]. Dostupné z: <https://cloud.google.com/vision/docs/detecting-text>
43. Cloud Speech-to-Text Basics. *Cloud.google.com* [online]. [cit. 2018-11-01]. Dostupné z: <https://cloud.google.com/speech-to-text/docs/basics>
44. *Fotografie pokoje* [online]. [cit. 2019-05-01]. Dostupné z: <http://d2e5ushqwiltxm.cloudfront.net/wp-content/uploads/sites/12/2016/02/09120423/Pullman-Executive-Room-King-Bed-1.jpg>
45. *Fotografie automobilu* [online]. [cit. 2019-05-01]. Dostupné z: <https://www.najdi-karu.cz/uploads/2019-05-08/f92ab2d0-3c76-4525-a0dd-8422df5944f7.jpg>
46. *Fotografie běžce* [online]. [cit. 2019-05-01]. Dostupné z: [http://s1.lzoom.me/big0/70/Men\\_sprintingRun\\_471584.jpg](http://s1.lzoom.me/big0/70/Men_sprintingRun_471584.jpg)
47. *Fotografie Eiffelovy věže* [online]. [cit. 2019-05-01]. Dostupné z: <http://d2e5ushqwiltxm.cloudfront.net/wp-content/uploads/sites/12/2016/02/09120423/Pullman-Executive-Room-King-Bed-1.jpg>
48. *Fotografie stromu* [online]. [cit. 2019-05-01]. Dostupné z: [https://imagesvc.meredithcorp.io/v3/mm/image?url=https%3A%2F%2Fimg1.southernliving.timeinc.net%2Fsites%2Fdefault%2Ffiles%2Fstyles%2F4\\_3\\_horizontal\\_inbody\\_900x506%2Fpublic%2Fimage%2F2018%2F06%2Fmain%2Fgettyimages-666747504.jpg](https://imagesvc.meredithcorp.io/v3/mm/image?url=https%3A%2F%2Fimg1.southernliving.timeinc.net%2Fsites%2Fdefault%2Ffiles%2Fstyles%2F4_3_horizontal_inbody_900x506%2Fpublic%2Fimage%2F2018%2F06%2Fmain%2Fgettyimages-666747504.jpg)
49. *Fotografie pohovky* [online]. [cit. 2019-05-01]. Dostupné z: [https://www.olsson-gerthel.se/sites/all/files/styles/product\\_image\\_zoom/public/produktvisning/main\\_image/eilersen-skagen-soffa.png](https://www.olsson-gerthel.se/sites/all/files/styles/product_image_zoom/public/produktvisning/main_image/eilersen-skagen-soffa.png)
50. Vyhláška č. 343/2014 Sb. *Zákony pro lidi* [online]. 2018 [cit. 2019-05-01]. Dostupné z: <https://zakonyprolidi.cz/cs/2014-343>

51. Kódy krajů - registrační značky. *Ministerstvo dopravy* [online]. 2018 [cit. 2019-05-01]. Dostupné z: <https://www.mdcz.cz/Dokumenty/Silnicni-doprava/Registrace-vozidel/Kody-kraju-registracni-znacky>

**SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK**

HTML	HyperText Markup Language (odkazovací značkový jazyk)
HTTP	Internetový protokol (Hypertext Transfer Protocol)
Cloud	Na internetu založený model použití počítačových technologií
REST	Architektura rozhraní (Representational State Transfer)
API	Rozhraní pro programování aplikací (Application Programming Interface)
PDF	Portable Document Format (přenosný formát dokumentů)
JPG	Formát ztrátové komprese obrázků
GIF	Graphics Interchange Format (grafický formát určený pro rastrovou grafiku)
PNG	Portable Network Graphics (grafický formát určený pro rastrovou grafiku)

**SEZNAM GRAFŮ**

Graf 1 Pravděpodobnost detekce automobilu společností Microsoft .....	72
Graf 2 Pravděpodobnost detekce automobilu společností Google .....	73
Graf 3 Detekce automobilu .....	80
Graf 4 Detekce státní poznávací značky .....	81



**SEZNAM OBRÁZKŮ**

Obrázek 1 Perceptron [3] .....	13
Obrázek 2 Sigmoidální funkce [4] .....	13
Obrázek 3 Ukázka grafického prostředí Machine Learning Studia .....	19
Obrázek 4 Schéma architektury aplikace .....	47
Obrázek 5 Ukázkový obrázek pokoje [44] .....	50
Obrázek 6 Ukázkový obrázek auta [45] .....	50
Obrázek 7 Ukázkový obrázek běžce [46] .....	52
Obrázek 8 Ukázkový obrázek Eiffelovy věže [47] .....	52
Obrázek 9 Ukázkový obrázek stromu [48] .....	54
Obrázek 10 Ukázkový obrázek pohovky [49] .....	55
Obrázek 11 Přední část automobilu .....	56
Obrázek 12 Přední část automobilu .....	57
Obrázek 13 Levý bok automobilu .....	58
Obrázek 14 Pravá bok automobilu .....	58
Obrázek 15 Interiér automobilu .....	59
Obrázek 16 Fotografie jediného automobilu .....	74
Obrázek 17 Fotografie automobilů na parkovišti .....	75
Obrázek 18 Výsledek detekce poznávací značky 1 .....	77
Obrázek 19 Výsledek detekce poznávací značky 2 .....	77
Obrázek 20 Výsledek detekce poznávací značky 3 .....	77
Obrázek 21 Výsledek detekce levé strany poznávací značky .....	78
Obrázek 22 Výsledek detekce pravé strany poznávací značky .....	78

**SEZNAM TABULEK**

Tabulka 1 Výsledky detekce objektů .....	53
Tabulka 2 Výsledky vytvořených vět popisujících obrázků .....	55
Tabulka 3 Výsledky detekce dominantní barvy společností Google .....	57
Tabulka 4 Výsledky detekce dominantních barev společnosti Microsoft .....	58
Tabulka 5 Rozdělení obrázku do kategorií .....	59
Tabulka 6 Detekce přední části automobilu .....	60
Tabulka 7 Výsledky detekce zadní části automobilu .....	61
Tabulka 8 Výsledky detekce levé části automobilu .....	62
Tabulka 9 Výsledky detekce pravé části automobilu .....	62
Tabulka 10 Výsledky detekce interiéru automobilu .....	63
Tabulka 11 Přehled detekovaných objektů na fotografii s autem .....	79
Tabulka 12 Přehled detekovaných objektů na fotografii s více auty .....	79

## SEZNAM PŘÍLOH

Příloha P I: CD obsahuje zdrojové kódy a tabulky s naměřenými daty