1 Úvod do problematiky

1.1 Získanie textu z naskenovaných dokumentov

Na to, aby sme dokumenty mohli akokoľvek analyzovať, potrebujeme ich mať uložené v takom formáte, aby boli strojom čitateľné. Musia byť teda uložené ako text. Všetky dokumenty, ktoré v našej práci spracúvame sú uložené vo formáte PDF. Avšak iba malá časť je uložená ako text. Vo väčšine prípadov sa jedna o obrázok – naskenovaný dokument. Preto prvým krokom je získanie textu z obrázku pomocou technológie OCR. Táto technológia nie je ťažiskom práce, no je jej nevyhnutnou súčasťou.

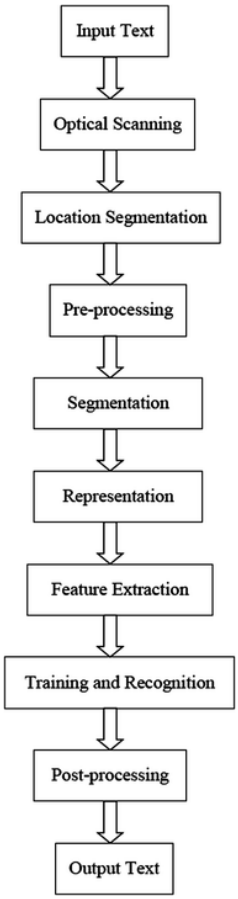
OCR alebo optické rozoznávanie znakov sa zaoberá problémom klasifikácie optických vzorov v digitálnom obraze do príslušných alfanumerických či iných znakov [1]. Vďaka tomu sa uľahčí ich ukladanie, keďže namiesto množstva pixelov môžeme uložiť jeden alebo viac znakov. Ešte väčším benefitom je ale to, že takto uložený text môžeme prehľadávať, analyzovať a ľahko upravovať.

Tento problém bol na začiatku výskum rozpoznávania vzorov považovaný za jednoduchý [2]. Znakov je pomerne malé množstvo a ľahko sa s nimi pracuje. Problém prichádza ak sa neobmedzíme iba na latinku a jeden font – v tom prípade množstvo vzorov narastá. Ešte väčší problém ale spôsobuje ručne písaný text a kvalita spracovaného obrazu. Problematika rozpoznávania znakov sa ukázala byť komplikovanejšou. V súčasnosti je už ale dostupných množstvo komerčných aplikácii vykonávajúcich OCR na pomerne dobrej úrovni [1]. Tieto aplikácie sú ale stále veľmi závisle na kvalite vstupného obrazu, preto obzvlášť pri spracovaní menej kvalitného obrazu sa stále nemôžu porovnávať v presnosti rozpoznania znakov s ľudskými schopnosťami [1]. V našej práci používame open source OCR engine Tesseract [3].

Najskoršie generácie OCR systémov sa spoliehali predovšetkým na techniky rozpoznávania vzorcov a spracovania obrazu [2], veľké zlepšenie ale prinieslo zapojenie metód umelej inteligencie [1]. V nedávnej minulosti prinieslo ďalšie zlepšenie, podobne ako v iných oblastiach, využitie umelých neurónových sieti (ANN). [1]

Obrázok 1 ukazuje celú postupnosť štandardných krokov OCR systémov. Za hlavné kroky OCR by sa dali označiť [4]

- analýza rozloženia dokumentu
- rozpoznanie znakov
- post-processing



Obrázok 1: Detailná postupnosť krokov OCR systému [1]

1.1.1 Prípravné kroky pred rozpoznávaním znakov

Shafait považuje analýzu rozloženia dokumentu (layout analysis) za prvý krok, ktorý OCR systém vykonáva [4]. Chaudhuri tento krok nazýva *location segmentation*, no ešte pred ním uvádza krok *optical scanning* [1].

Krok alebo komponent *optical scanning* vytvorí digitálnu verziu pôvodného dokumentu. Okrem toho je možné do tohto kroku zahrnúť techniku spracovania obrazu – prahovanie. Pomocou prahovania sa zo šedotónového obrazu vytvorí dvojúrovňový – čiernobiely obraz. Prahovanie môže byť lokálne alebo

globálne. Pri globálnom sa nájde jedna konštanta slúžiaca ako prah pre celý dokument. Pri lokálnom sa naopak vyberá vytvorí viacero oblastí, ktoré majú samostatné prahové konštanty. V niektorých implementáciách môže mať každý pixel vlastnú oblasť a teda aj vlastnú prahovú konštantu. [1].

Layout analysis alebo *location segmentation* sa snaží v dokumente identifikovať presne tie oblasti, ktoré obsahujú text. Takto sa odfiltruje napr. ilustračný obrázok umiestnený pri texte od samotného textu, ale aj biele oblasti, kde sa žiaden text nenachádza. Výstupom je obraz rozdelený na bloky, ktoré by mali obsahovať iba text. V závislosti od implementácie a dokumentu môže jeden blok obsahovať jedno slovo až jeden stĺpček. Komplikáciou je ak súčasťou dokumentu sú aj tabuľky, čo sa týka aj časti nami analyzovaných dokumentov [1] [4].

Pre nás je najzaujímavejší krok prespracovanie (*pre-processing)*, keďže v tomto kroku sa snažíme zlepšiť rozpoznanie textu niektorých dokumentov. Dokumentácia Tesseract enginu, ktorý v práci používame, totiž predspracovanie odporúča pre zlepšenie výsledku vykonať pre- predspracovanie ešte pred spustením Tesseractu – aj keď tento OCR systém sám používa rôzne metódy spracovania obrazu [3].

Metód predspracovania je mnoho v závislosti od problému, či problémov daného obrazu. Jeho základným cieľom ale je aby bol obraz čitateľnejší pre ďalšie komponenty OCR. V krátkosti predstavíme niekoľko problémov, ktoré predspracovanie rieši. Konkrétne riešenia ale vynecháme a v neskorších kapitolách predstavím tie z nich, ktoré použijeme odchýliac sa od štandardných techník Tesseractu.

- Jednou z najbežnejších metód je *odstránenie šumu*, ktorý každý digitálny obraz obsahuje. Časť šumu sa odstráni prahovaním [3], no po ňom môžu ostať napr. diery v čiarach, zaoblené rohy písmen a podobné artefakty [1].
- Predovšetkým pri rozpoznávaní rukou písaného textu môže byť nápomocná *normalizácia*. Hoci aj súvislý rukou písaný text môže meniť otočenie, veľkosť či rovinu (riadok) na ktorej je text písaný. Okrem toho, častým problémom obzvlášť pri hrubších knihách býva zatočenie textu, ktorý bol predtým písaný v jednej rovine. Historické dokumenty zas môžu mať problém s vypúšťaním príliš veľkého množstva atramentu a teda s príliš hrubým písmom. Túto skupinu problémov rieši normalizácia. [1], [3]
- Bežné kompresné techniky pre obrázky nie sú vhodné pre rozpoznávanie znakov. Zároveň, určitá kompresia môže byť žiadúca pre zvýšenie rýchlosti spracovania, resp. učenia sa. Aj po kompresii musí byť OCR systém schopný rozpoznať tvar jednotlivých znakov. Preto ako kompresná technika sa používa už spomínané prahovanie alebo *thinning*. Vďaka prahovaniu sa dramaticky zmenší potrebné miesto na uloženie farby jedného pixelu. Cieľom však je zachytiť celý pôvodný znak, hoci v praxi sa často stáva, že niektoré šedé pixely z kraja jednotlivých znakov sa prahovaním vymažú. *Thinning* sa naopak nesnaží zachytiť celý pôvodný znak ale iba jeho kostru. [1]

Ako ukazuje Obrázok 2Obrázok 1, ďalej nasledujú kroky, ktoré sa snažia po vyčistení dát v predspracovaní nájsť vhodnú reprezentáciu častí obrázku, pomocou ktorej systém ľahšie rozpozná znaky.

2.1.1 Rozpoznávanie znakov

Snahou je zaradiť rozpoznávanú vzorku do správnej triedy. Samotné rozpoznávanie jednotlivých znakov môže OCR systém vykonávať štyrmi základnými prístupmi, resp. ich kombináciou. Každý z týchto prístupov môže používať holistické alebo analytické stratégie. Holistické stratégie si nevyžadujú

segmentáciu a pristupujú najprv k celému slovu až potom k jednotlivým znakom. Efektívne sú najmä pri ťažko segmentovateľných textoch, napr. písaných kurzívou. Analytické stratégie naopak vyžadujú segmentácie, keďže postupujú od jednotlivých znakov, či dokonca ich čŕt nahor k slovám a následne textu. [1], [5]

a) *Template matching*
   Tento prístup bol historicky prvý vo vývoji OCR systémov. Vytvoria sa pri ňom prototypy jednotlivých tried, ktoré môžu mať rôznu reprezentáciu. V závislosti od toho sa môže porovnávať miera zhody skupiny pixelov, zakrivenia či primitív. Bez ohľadu na spôsob reprezentácie, porovnávajú sa jednotlivé prototypy s obrázkom, ktorý sa má rozpoznať. V závislosti od komplexnosti systému sa môže porovnávať priamo s prototypmi alebo sa tieto prototypy môžu rôzne deformovať. Každopádne, rozpoznaný obraz sa zaradí do triedy s ktorej prototypom sa najlepšie zhoduje. [1], [2]
b) *Štatistický prístup*
   Pri tomto prístupe je nevyhnutné každý rozpoznávaný obrázok reprezentovať ako množinu vlastností. Tieto vlastností by mali byť vybrané tak, aby dovoľovali zaradenie do viacerých tried. Zároveň, podľa týchto vlastností by mali byť jednotlivé triedy separovateľné. Cieľom je naučiť sa na trénovacej množine tieto hranice. Prostriedkom môže byť klasterizácia, Bayesov či Markovov model. [1], [6]
c) *Syntakticko-štrukturálny prístup*
   Za týmto prístupom je snaha rekurzívne rozdeliť rozpoznávaný obrázok na primitíva. Komplexný obraz je tak reprezentovaný primitívami a vzťahmi medzi nimi. Z primitív sa pomocou pravidiel dajú vytvoriť inštancie jednotlivých tried. [1], [6]
d) *ANN*
   V súčasnosti najpoužívanejším prístupom v OCR sú umelé neurónové siete. ANN poskytujú predovšetkým možnosť masívneho množstvo paralelných výpočtov. Napriek rôznosti možných architektúr sa dá dokázať, že väčšina je ekvivalentná štatistickým metódam. V OCR systémoch sú najpoužívanejšími architektúrami dopredný viacvrstvový perceptrón a *self-organizing map*.

3.1.1 Post-processing

Posledným krokom, ktorý OCR systém vykonáva je *post-processing*. Využíva pritom techniky NLP na odhalenie a opravenie chýb, ktorých sa dopustil pri rozpoznávaní. Tieto chyby môžu byť také, že ako výstup rozpoznávania dostaneme slovo, ktoré

a) nie je skutočným slovom z daného jazyka
b) je slovom daného jazyka, no nie tým, ktoré bolo v rozpoznávanom dokumente

Ak sa jedná o prvý prípad, korekcia môže byť pomerne jednoduchá. Môže sa prehľadať slovník jazyka, prípadne vypočítať pravdepodobnosť, že niektoré písmena budú pri sebe. Napr. v slovenčine je nulová pravdepodobnosť, že *ď* a *y* budú vedľa seba a tak ak niečo také počas post-processingu nájdeme, s istotou bolo niektoré písmeno rozpoznané zle. Komplikovanejšia, no nie nemožná je korekcia v druhom prípade. Na to potrebujeme analyzovať kontext daného slova. Na to môžeme použiť rôzne štatistické modely ako ukazuje Tong . [1], [7]

2.1 Tesseract

V našej práci používame open-source OCR systém Tesseract, ktorý bol vyvíjaný najskôr ako PhD projekt v spoločnosti HP. Táto spoločnosť neskôr prebrala vývoj systému až kým sa nestal open-source. Krátko na to nad ním prebrala záštitu spoločnosť Google, ktorá ho naďalej vyvíja ako open-source projekt. [8]–[10]

Tesseract po predspracovaní extrahuje komponenty obrázku a ich obrysy organizuje do tzv. *Blobov*. *Bloby* sú organizované do riadkov textu. Riadky sú následné analyzované pre fixnú výšku textu. Rozdelenie riadku na slová sa vykonáva s prihliadnutím na rovnomerné, ale aj nerovnomerné medzery. [8]

Samotné rozpoznávanie je dvojfázové, keďže Tesseract používa adaptívne rozpoznávanie. V prvej fáze sa rozpoznávajú rad za radom všetky slová. Tie, ktoré sú rozpoznané dostatočne dobre sa následne uložia ako dáta na trénovanie adaptívneho klasifikátora. Ten sa používa až v druhej fáze, keď sa opäť prejde celá strana. Pri tomto druhom prechode sa už ale rozpoznávajú iba tie slová, ktoré neboli v prvej fáze rozpoznané dostatočne dobre. Nakoniec sa riešia nejasné medzery a alternatívne hypotézy pre výšku jednotlivých riadkov. Lingvistický post-processing je v Tesseracte iba minimálny. [8]

3.1 Klasifikácia dokumentu

1.3.1 Konečný užívateľ výhod

Dokumenty, ktoré sú v našej práce analyzované vznikajú na základe zákona o registri partnerov verejného sektora a o zmene a doplnení niektorých zákonov z 25. októbra 2016[11]. Pre našu prácu je zaujímavý §11, ktorý hovorí o identifikácii konečného užívateľa výhod. Ten je ale definovaný zákonom o ochrane pred legalizáciou príjmov z trestnej činnosti a o ochrane pred financovaním terorizmu a o zmene a doplnení niektorých zákonov. Podľa tohto zákona sa ako konečný užívateľ výhod (ďalej len KUV) označí

a) „fyzická osoba ktorá skutočne ovláda alebo kontroluje právnickú osobu, fyzickú osobu – podnikateľa alebo združenie majetku, a každá fyzická osoba, v prospech ktorej tieto subjekty vykonávajú svoju činnosť alebo obchod“ [12]
b) členovia jej vrcholového manažmentu ak žiadna fyzická osoba nespĺňa kritéria aby mohla byť zapísaná podľa a) [12]
c) „Konečným užívateľom výhod je aj fyzická osoba, ktorá sama nespĺňa kritériá podľa odseku 1 písm. a), b) alebo písm. c) druhého a štvrtého bodu, avšak spoločne s inou osobou konajúcou s ňou v zhode alebo spoločným postupom spĺňa aspoň niektoré z týchto kritérií.“[12]

Kritéria na zapísanie KUV podľa a) sa líšia v závislosti od typu spoločnosti. Ak ide o právnickú osobu, ktorá nie je združením majetku ani emitentom cenných papierov, fyzická osoba je označená ako KUV podľa a) ak

1. má priamy alebo nepriamy podiel alebo ich súčet najmenej 25 % na hlasovacích právach v právnickej osobe alebo na jej základnom imaní vrátane akcií na doručiteľa, [12]
2. má právo vymenovať, inak ustanoviť alebo odvolať štatutárny orgán, riadiaci orgán, dozorný orgán alebo kontrolný orgán v právnickej osobe alebo akéhokoľvek ich člena, [12]
3. ovláda právnickú osobu iným spôsobom, ako je uvedené v prvom a druhom bode, [12]
4. má právo na hospodársky prospech najmenej 25 % z podnikania právnickej osoby alebo z inej jej činnosti. [12]

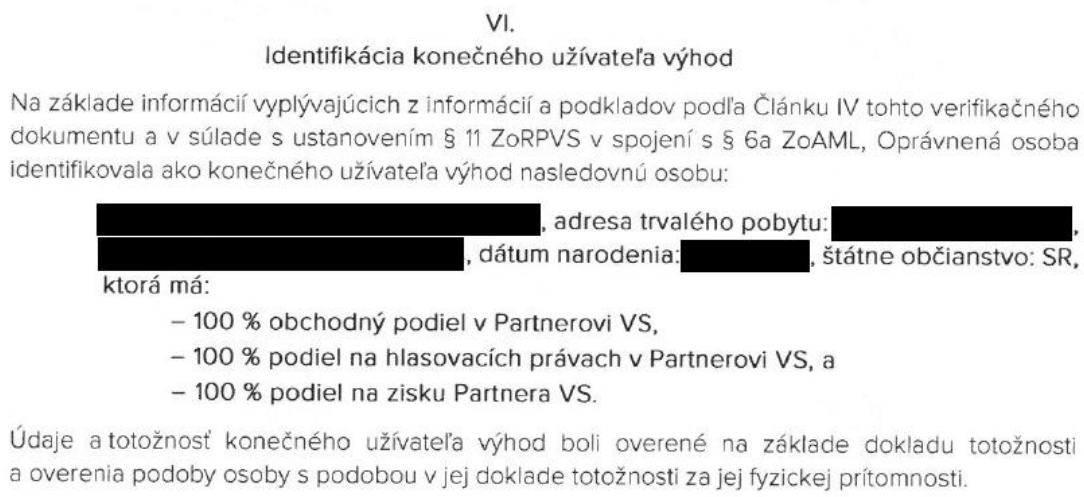
„Ak ide o fyzickú osobu – podnikateľa, fyzická osoba, ktorá má právo na hospodársky prospech najmenej 25 % z podnikania fyzickej osoby – podnikateľa alebo z inej jej činnosti“[12] bude označená za KUV.

Pokiaľ sa jedná o združenie majetku, fyzická osoba je označená ako KUV podľa a) ak

1) je zakladateľom alebo zriaďovateľom združenia majetku
2) má právo vymenovať, inak ustanoviť alebo odvolať štatutárny orgán, riadiaci orgán, dozorný orgán alebo kontrolný orgán združenia majetku alebo ich člena alebo je členom orgánu, ktorý má právo vymenovať, inak ustanoviť alebo odvolať tieto orgány alebo ich člena,
3) je štatutárnym orgánom, riadiacim orgánom, dozorným orgánom, kontrolným orgánom alebo členom týchto orgánov,
4) je príjemcom najmenej 25 % prostriedkov, ktoré poskytuje združenie majetku, ak boli určení budúci príjemcovia týchto prostriedkov; ak neboli určení budúci príjemcovia prostriedkov združenia majetku, za konečného užívateľa výhod sa považuje okruh osôb, ktoré majú významný prospech zo založenia alebo pôsobenia združenia majetku.[1][12]

To, kto je zapísaný ako KUV pre danú firmu je dostupné na rpvs.sk spolu s dokumentom, ktorý popisuje ako bol KUV identifikovaný a zdôvodňuje, prečo bola daná osoba zapísaná ako KUV. Zjednodušene, v našej práci sa snažíme rozdeliť spoločnosti na také

- ktorých KUV je skutočným vlastníkom
- ktorých skutočný vlastník nie je zo štruktúrovaných dát známy, keďže ako KUV je zapísaný štatutár



VI.
Identifikácia konečného užívateľa výhod

Na základe informácií vyplývajúcich z informácií a podkladov podľa Článku IV tohto verifikačného dokumentu a v súlade s ustanovením § 11 ZoRPVS v spojení s § 6a ZoAML, Oprávnená osoba identifikovala ako konečného užívateľa výhod nasledovnú osobu:

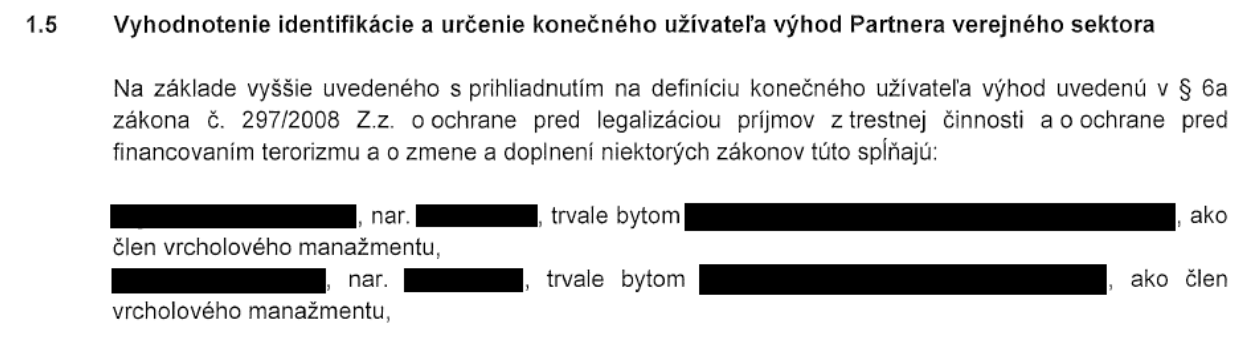
[redacted], adresa trvalého pobytu: [redacted], [redacted], dátum narodenia: [redacted], štátne občianstvo: SR, ktorá má:

– 100 % obchodný podiel v Partnerovi VS,
– 100 % podiel na hlasovacích právach v Partnerovi VS, a
– 100 % podiel na zisku Partnera VS.

Údaje a totožnosť konečného užívateľa výhod boli overené na základe dokladu totožnosti a overenia podoby osoby s podobou v jej doklade totožnosti za jej fyzickej prítomnosti.

Obrázok 2: Príklad ako môže vyzerať časť dokumentu v prípade, že ako KUV bol zapísaný majiteľ

[1] Pre viac informácii o KUV a spôsobe identifikácie pozri 297/2008 Z.z § 6a



1.5 **Vyhodnotenie identifikácie a určenie konečného užívateľa výhod Partnera verejného sektora**

Na základe vyššie uvedeného s prihliadnutím na definíciu konečného užívateľa výhod uvedenú v § 6a zákona č. 297/2008 Z.z. o ochrane pred legalizáciou príjmov z trestnej činnosti a o ochrane pred financovaním terorizmu a o zmene a doplnení niektorých zákonov túto spĺňajú:

[redacted], nar. [redacted], trvale bytom [redacted], ako člen vrcholového manažmentu,
[redacted], nar. [redacted], trvale bytom [redacted], ako člen vrcholového manažmentu,

Obrázok 3: Príklad ako môže vyzerať časť dokumentu v prípade, že ako KUV bol zapísaný štatutár

Mám vstup zo štrukturovaných dát. Potrebujem funkciu jeKonečnýUžívateľ(Zoznam ľudí), ktorá zistí, či tí ľudia sú skutočnými majiteľmi, prípadne či ovládajú spoločnosť (teda splňajú znaky KUV podľa nejakého zákona).

Potrebujem Name Entity Recognociosion. Potrebujem nahradiť zapísaného KUV ako KUV -> nech sa model učí potvrdiť/poprieť výrok „KUV je skutočným vlastníkom“. Pre jeKonečnýUžívateľ(jeKonečnýUžívateľ)

[1] A. Chaudhuri, K. Mandaviya, P. Badelia, and S. K. Ghosh, “Optical Character Recognition Systems,” *Optical Character Recognition Systems for Different Languages with Soft Computing*, pp. 9–41, 2017, doi: 10.1007/978-3-319-50252-6_2.

[2] S. Mori, C. Y. Suen, and K. Yamamoto, “Historical review of OCR research and development,” *Proceedings of the IEEE*, vol. 80, no. 7, pp. 1029–1058, Jul. 1992, doi: 10.1109/5.156468.

[3] “Tesseract documentation,” *Tesseract OCR*. http://tesseract-ocr.github.io/ (accessed Jan. 21, 2021).

[4] F. Shafait and R. Smith, “Table Detection in Heterogeneous Documents,” *Detecting tables in document images is important since not only do tables contain important information, but also most of the layout analysis methods fail in the presence of tables in the document image. Existing approaches for table detection mainly focus on detecting tables in single columns of text and do not work reliably on documents with varying layouts. This paper presents a practical algorithm for table detection that works with a high accuracy on documents with varying layouts (company reports, newspaper articles, magazine pages, . . .). An open source implementation of the algorithm is provided as part of the Tesseract OCR engine. Evaluation of the algorithm on document images from publicly available UNLV dataset shows competitive performance in comparison to the table detection module of a commercial OCR system.*, p. 9.

[5] F. M. A. Nashwan, M. A. A. Rashwan, H. M. Al-Barhamtoshy, S. M. Abdou, and A. M. Moussa, “A Holistic Technique for an Arabic OCR System,” *Journal of Imaging*, vol. 4, no. 1, Art. no. 1, Jan. 2018, doi: 10.3390/jimaging4010006.

[6] A. K. Jain, R. P. W. Duin, and Jianchang Mao, “Statistical pattern recognition: a review,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 1, pp. 4–37, Jan. 2000, doi: 10.1109/34.824819.

[7] X. Tong and D. A. Evans, “A Statistical Approach to Automatic OCR Error Correction in Context,” 1996, Accessed: Jan. 30, 2021. [Online]. Available: https://www.aclweb.org/anthology/W96-0108.

[8] R. Smith, “An Overview of the Tesseract OCR Engine,” in *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, Sep. 2007, vol. 2, pp. 629–633, doi: 10.1109/ICDAR.2007.4376991.

[9] C. Patel, A. Patel, and D. Patel, “Optical Character Recognition by Open source OCR Tool Tesseract: A Case Study,” *International Journal of Computer Applications*, vol. 55, pp. 50–56, Oct. 2012, doi: 10.5120/8794-2784.
[10] “Announcing Tesseract OCR - The official Google Code blog,” *Announcing Tesseract OCR - The official Google Code blog*, Aug. 30, 2006. http://googlecode.blogspot.com/2006/08/announcing-tesseract-ocr.html (accessed Feb. 01, 2021).
[11] Slov-lex, “315/2016 Z.z. - Zákon o registri partnerov verejnéh...,” *Slov-lex*. https://www.slov-lex.sk/pravne-predpisy/SK/ZZ/2016/315/20170201 (accessed Jan. 20, 2021).
[12] Slov-lex, “297/2008 Z.z. - Zákon o ochrane pred legalizáciou p...,” *Slov-lex*. https://www.slov-lex.sk/pravne-predpisy/SK/ZZ/2008/297/ (accessed Jan. 20, 2021).