Wstęp

Obecnie coraz więcej dokumentów przechowujemy w sposób elektroniczny. Od 2020 roku pojawiła się nawet możliwość załatwienia spraw urzędowych przez Internet i składanie wniosków bez wychodzenia z domu. Trzymanie dokumentów na komputerze ubezpiecza nas od możliwej straty informacji z powodu fizycznych szkód, a w przypadku ich dużej ilości wielką zaletą jest szybkość, podczas sortowania, filtracji i wyszukiwania danych. Danymi, które te dokumenty mogą zawierać, są tekst i obrazy. Tekst - jest zbiorem zdań, zdania jest zbiorem słów, a słowo składa się z liter.

Wprowadzanie tekstu, zawartego w dokumencie do komputera, odbywa się za pomocą klawisz klawiatury, każda z których jest oznakowana pojedynczymi literami, cyframi lub symbolami interpunkcyjnymi. Komputer traktuje wciśnięta klawisze jako unikalny kod binarny lub liczbowy, a następnie używa wybranego przez system pliku czcionki, w celu wyświetlenia powiązanego z nim znaku na ekranie monitora. Stworzenie elektronicznej kopii dokumentu poprzez ręczne wpisywania każdego znaku jest dość czasochłonne w przypadku, gdy składa się on z więcej niż jednej strony, ale takie rozwiązanie powoduje, że w wyniku dostajemy stworzony edytowalny tekst.

Szybszym sposobem na przekazanie dokumentu komputerowi jest stworzenie jego wersji cyfrowej za pomocą skanera lub kamery. Taki dokument zostanie umieszczony w pamięci komputera jako plik graficzny, zapisany w odpowiednim formacie, najczęściej którymi są: JPEG, PDF, PNG albo TIFF. Tego typu plik przedstawia obraz na ekranie za pomocą zawartej informacji o ilości pikseli oraz ich położenia i koloru. Z tego wynika, że taka wersja dokumentu elektronicznego, służy tylko do podglądu jego treści.

Od lat 1950 ciągle się rozwija rozwiązanie, za pomocą którego w dzisiejszych czasach można uzyskać tekst z każdego pliku graficznego. Jest to oprogramowanie komputerowe, które się nazywa „OCR”. W swojej pracy pokażę jego zastosowanie we własnej aplikacji, cel której, jest odczytywanie danych z najczęściej spotykanych wśród ludzi dokumentów. Jest to dokument potwierdzający sprzedaż – „Paragon Fiskalny”.

**Rozdział I**

1. Optyczne rozpoznawanie znaków

Optyczne rozpoznawanie znaków (Optical Character Recognition, OCR), jest to technologia, która automatycznie identyfikuję liczby, litery oraz znaki interpunkcyjne w plikach graficznych i konwertuje ich do postaci danych, zrozumiałych komputerowi. Na wejściu podawany jest wcześniej przygotowany plik rastrowy, który poddaje się analizie, w celu odnajdywania w nim zbioru pikseli, wizualizujących symbol, a następnie do każdego znalezionego, przypisuje się kod binarny lub liczbowy z tablicy znaków. W wyniku, zwracany jest stworzony, edytowalny tekst. Przy użyciu skanera lub urządzenia wyposażanego w kamerę, rozpoznawanie tekstu może odbyć się w trakcie generowania pliku graficznego. W innym przypadku, możliwe jest przekazanie przygotowanej wcześniej grafiki, zawierającej tekst, do systemu OCR.

Obecnie nie istnieję idealnego oprogramowania, które zawsze potrafi rozpoznać tekst z dokładnością do 100%. Prawdopodobieństwo tego, że OCR odniesie sukces podczas pracy, zależy w pierwszej kolejności od jakości materiału wejściowego. Im większa jest rozdzielczość obrazu i jego kontrastowość, tym większa szansa, że tekst zostanie poprawnie rozpoznany. Drugim ważnym czynnikiem jest samo oprogramowanie, składające się z po kolei wywoływanych algorytmów służących do poprawy obrazu, segmentacji tekstu, rozpoznania znaków i analizie wyniku. Dziś można zaobserwować dużą ilość gotowych rozwiązań, które używają własne zaimplementowane sieci neuronowe, za pomocą których dokładność rozpoznawania znaków zrasta. Taki program jest uniwersalny, ponieważ pojawia się możliwość dostosować go do odpowiedniego problemu, i nauczyć rozumieć pismo ręczne lub tekst o dowolnej czcionce.**Proces rozpoznawania znaków**

Diagram

Description automatically generatedW celu wyodrębnienia ciągu znakowego z obrazu, używane są różne techniki. Każda technika składa się z procesów, na wejście do których jest podawany wynik poprzedniego kroku, zaczynając od pliku graficznego i kończąc sformatowanym, edytowalnym tekstem.

Rysunek 1.https://www.geeksforgeeks.org/tesseract-ocr-with-java-with-examples/

Praca systemów OCR w większości wypadków jest podzielona na cztery etapy:

1. **Przetwarzanie wstępne (ang. Pre-processing)**

Przekazany do OCR obraz, po procesie skanowania może zawierać pewną ilość szumów. Plamy, brud, zagięty lub pomarszczony papier, uszkodzona matryca urządzenia – jest to czynniki powodujące pojawienie w pliku rastrowym nie potrzebnych pikseli, utrudniających rozpoznawanie. Za duży poziom jasności, słaba jakość lub kontrast spowodują zniszczenie kształtu znaków, lub ich zamazanie. Dlatego, przed procesem identyfikacji znaków, na etapie wstępnym, implementuje się do obrazu metody programowe, które mają na celu polepszyć jakość pliku rastrowego oraz usunąć defekty, co spowoduję zwiększenie współczynnika rozpoznawania tekstu.

Obraz – jest zbiorem pikseli o różnych kolorach, aby rozgraniczyć, gdzie znajdują się w nim znaki, oprogramowanie OCR przyjmuję założenie, że są oni zawsze białego koloru, a wszystko pozostałe jest czarne. W fazie przetwarzania wstępnego, do zamiany kolorów na skalę szarości używany jest algorytm nazywany binaryzacja (z ang. binarization). Jego działanie wygląda następująco:

- przyjmuje się, że piksel białego koloru jest równy wartości 255, a czarny = 0. Następnie, odbywa się zaokrąglenie każdego pikselu według ustawionego progu (domyślne jest to połowa wartości 255, czyli 127). Jeśli wartość pikselu jest mniejsza od wartości progu, to taki piksel jest uznawany za czarny, w przeciwnym przypadku jego kolor zmieni się na biały.

1. **Segmentacja tekstu (ang. Text segmentation)**

Zanim program zacznie rozpoznawać znaki, istotnie jest najpierw rozbić zlokalizowany na obrazie tekst na pojedyncze linie, a te na słowa oraz litery. Dla tego celu, można użyć techniki histogramowej projekcji, która polega na wyliczaniu rzutów obrazu.

Najpierw, wykonuje się segmentacja linii tekstu, aby podzielić obszar tekstowy na pojedyncze linie. Odbywa się to za pomocą metody do wyliczania rzutu poziomowego (Horizontal Histogram Projection). Ta metoda przyjmuje na wejście obraz binarny otrzymany z poprzedniego kroku, w którym białe pikseli - są uznawane za ważną informację, a czarne - za tło. Następnie, oblicza się suma białych pikseli w każdym wierszu obrazu. Wynikiem działania metody jest lista o długości równej wysokości obrazu. Jeśli wartość elementu listy jest duża pod względem pozostałych elementów, to znaczy, że jest to część linii tekstu. W przeciwnym wypadku, elementy o malej wartości mogą być uznane za obszary znajdujące się pomiędzy liniami tekstu i służyć jako wskaźniki na miejsca, gdzie należy podzielić obraz.

Kolejnym zadaniem jest segmentacja słów. Używa się do tego metoda wyliczania rzutu pionowego (Vertical Histogram Projection), do której przekazywany jest wycięty w poprzednim kroku obraz, zawierający pojedynczą linię tekstu. Jej działanie jest podobne do metody wykonywanej podczas segmentacji linii tekstu, jedną różnicą jest to, że tutaj oblicza się suma białych pikseli w każdej kolumnie obrazu a nie wierszu, jak to było wcześniej. Wynik – przedstawia listę o długości równej szerokości obrazu, a jej elementy w zależności od posiadanej wartości, mogą być częścią słów – jeśli są o dużej wartości lub polem pomiędzy nimi – w przypadku wartości niskich.

Proces segmentacji znaków jest opcjonalnym, jego użycie zależy od typu tekstu na obrazie. Jeśli znaki tworzące słowo są niezależne i nie łączą się pomiędzy sobą, to w takim wypadku można użyć metody używanej podczas segmentacji słow. Podczas wyliczania rzutu pionowego, elementy o niskich wartościach . Odstęp pomiędzy słowami jest większy niż odstęp pomiędzy znakami, dla tego aby poprawnie wyliczyć można ustawić próg szerokości uznawany za

1. **Character recognition**

Oprogramowanie OCR rozpoznaje tekst konsekwentnie. Najpierw łączy pikseli w ewentualne znaki, a znaki – w ewentualne słowa. Następnie, program sprawdza czy jest odpowiednik tego słowa w słowniku. Jeżeli tak, to zostaje ono oznaczone jako rozpoznane. W innym przypadku, program zaproponuje najbardziej prawdopodobny wynik, tym samym zmniejszy współczynnik rozpoznawania tekstu.

1. **Post Processing**
2. **Sieci neuronowe**
3. **Long short-term memory (LSTM)**
4. **Tesseract**

**Rozdział II**

**|------------------**

**Użyć w Drugim rozdziale podczas opisywania własnych algorytmów poprawiających obraz (i dołączyć przykładowe zdjęcia)**

**Binaryzacja (ang. binarization) -** Algorytm binaryzacji konwertuje obraz kolorowy lub z odcieniem szarości na obraz binarny, w którym pikseli mogą przyjmować jedną z dwóch wartości. Celem tego procesu jest oddzielenie pierwszego planu, czyli tekstu, od tła.

**Usuwanie szumów (ang. De-speckle) -** Jedną z najczęściej pojawiających się problemów, powiązanych z jakością obrazu jest powstanie różnego rodzaju szumów. Jednym z powodów ich pojawienia, może być zanieczyszczenia, znajdujące się na obiektywie kamery cyfrowej lub szubie skanera. Brud lub rysy znajdujące się na dokumencie też sprzyjają pojawieniu szumów na obrazie.

**Poziomowanie (ang. de-skew) -** Podawane na wejściu dokumenty, czasami mogą być zeskanowane lub sfotografowane ukośnie. Tekst w podobnych materiałach jest trudniejszy do identyfikacji programami OCR. W związku z tym, żeby uniknąć podobnych problemów i zwiększyć szanse na prawidłowe rozpoznanie znaków, do obrazu stosują się algorytmy które wyprostowują linii tekstu.

**Analiza układu dokumentu (ang. document layout analysis)** - program zacznie rozpoznawać znaki, najpierw istotnie jest zidentyfikować i sklasyfikować elementy występujące w dokumencie. Zazwyczaj takimi elementami mogą być zdjęcia, grafika, tabeli, linii, tekst, tytuł, akapity, łamy. Bez tej analizy kolejność zdań w rozpoznanym tekście może się różnić od dokumentu wejściowego, a wyrazy przenoszone mogą zostać podzielone na pól.

**-------------------------------------------|**