# Dokumentacja - Projekt na Analizę Obrazów

# Prosty czytnik OCR

Jakub Kawka, Daria Kokot, Kacper Duda, Aleksandra Śliwska

22.01.24r.

## Spis treści

# 1 O projekcie

Projekt ten ma charakter prostego narzędzia optycznego rozpoznawania znaków (OCR), którego głównym celem jest identyfikacja tekstu ustrukturyzowanego, tj. tekstu napisanego standardowymi czcionkami przewidzianymi do druku. Aplikacja umożliwia użytkownikowi pozyskanie kopiowalnego i edytowalnego tekstu ze zdjęcia, zrzutu ekranu lub skanu ciemnego drukowanego tekstu na jasnym tle.

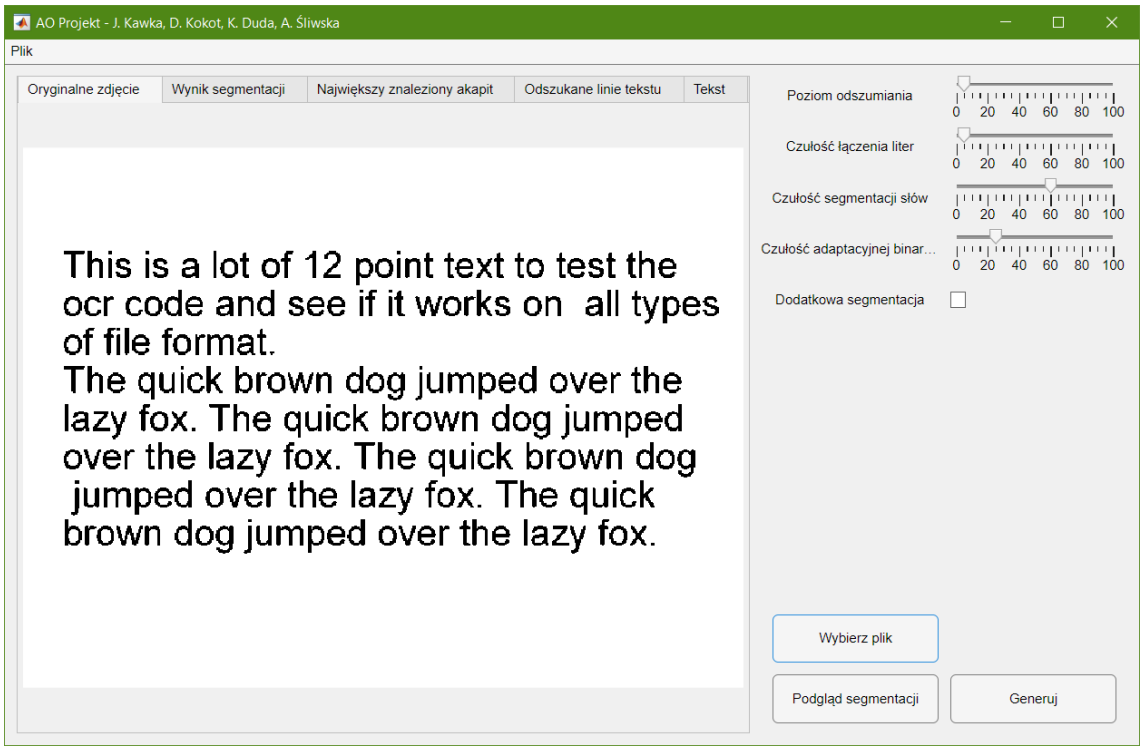
Rdzeń projektu stanowią dwa algorytmy:

- pierwszy - opierający się o analizę obrazu w celu znalezienia poszczególnych akapitów, linii i map binarnych liter ze zdjęcia
- oraz drugi - rozpoznający litery na pozyskanych obrazach poprzez przepuszczanie ich przez wytrenowany model sieci neuronowej i składający je w komputerowy tekst.

Optyczne rozpoznawanie znaków to technologia, która ma ogromny potencjał i znacznie ułatwia, między innymi, digitalizację dokumentów oraz może poprawiać jakość życia osób niedowidzących.

# 2 Opis działania aplikacji

## 2.1 Interakcja z użytkownikiem i przepływ danych




Rysunek 1: Wygląd aplikacji.

Aplikacja prezentuje się jak na obrazku powyżej. Przepływ danych przez nią jest następujący:

1. Użytkownik wybiera obraz z tekstem do analizy po wciśnięciu przycisku „Wybierz plik". Obraz ten pojawia się w zakładce „Oryginalne zdjęcie".
2. Użytkownik ma także opcję zmiany parametrów segmentacji liter i słów poprzez ręczną manipulację suwakami po prawej stronie okna oraz wczytywanie zapisanych wcześniej plików z parametrami segmentacji z paska menu (Plik > Wczytaj parametry).

3. W zakładce „Wynik segmentacji” pojawia się wynik wciśnięcia przycisku „Podgląd segmentacji” - wysłania obrazu i parametrów do funkcji „preview()” separatora. Dzięki temu użytkownik może dostosować parametry z krótkim czasem ładowania. Detale na zdjęciu można przybliżyć poprzez używanie na jego regionach jednego z trzech przycisków myszy.
4. Po satysfakcjonującym ustawieniu parametrów segmentacji i wciśnięciu przycisku „Generuj” następują po sobie odpowiednie wydażenia:
   (a) Do funkcji separatora „separate()” wysyłane są parametry i obraz.
   (b) Jej wyniki wyświetlają się w zakładkach „Wynik segmentacji”, „Największy znaleziony akapit” oraz „Odszukane linie tekstu”.
   (c) Dane o zawartości znalezionych paragrafów uzyskane przez separator wysyłane są do funkcji „recognize()” sieci neuronowej.
   (d) Każda litera jest klasyfikowana i dołączana do wynikowego tekstu wyświetlanego w zakładce „Tekst”.
5. Użytkownik ma opcję edycji tekstu wynikowego oraz zapisanie go do pliku tekstowego z paska menu w lewym górnym rogu okna aplikacji. Może tam także zapisać aktualnie ustawione parametry segmentacji.

## 2.2 Separator

Zadaniem separatora jest, w następującej kolejności:

- Segmentacja obrazu wraz z odszumianiem w celu wyszukania liter
- Separacja liter w linie tekstu
- Łączenie znaków diakrytycznych z literą w obrębie linii
- Identyfikacja poszczególnych słów w linii tekstu
- Odpowiednie przetworzenie liter, opatrzenie ich w metadane i przesłanie ich do sieci neuronowej

Separator działa przy spełnieniu następujących założeń:

- Tekst przesłany do separatora nie zawiera obrazów ani innych elementów które nie są literami (musi być dokumentem takim jak np. zeskanowana strona książki)
- Czcionka którą zapisany jest tekst posiada odpowiednie odstępy pomiędzy literami
- Odstępy pomiędzy liniami w tekście są odpowiednio duże aby można było je odróżnić i aby nie pokrywały się ze sobą
- Tekst jest zapisany od lewej do prawej w akapitach które nie są pochylone względem osi obrazu
- W przypadku skanów tekstu drukowanego czcionka powinna być wydrukowana czytelnie tj. nie ma przerw w wydruku

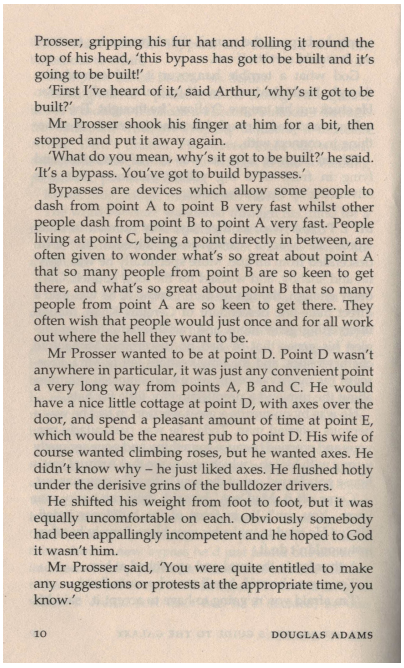
Są to bardzo silne założenia, jednak separator posiada pewne dostosowywalne wewnętrzne funkcjonalności, które pozwalają na odczytanie tekstu nie spełniającego ich w 100%.

### 2.2.1 Segmentacja obrazu

Segmentacja, lub preprocessing, zaczyna się od odebrania od użytkownika odpowiednich parametrów:

- Poziom odszumiania - steruje przekształceniami morfologicznymi, które mają na celu usunięcie szumów z obrazu - zbyt duża wartość uszkodzi tekst, a zbyt mała może spowodować przedostanie się pikseli szumu.
- Czułość łączenia liter - niektóre czcionki lub style druku mogą zostawiać odstępy wewnątrz liter, które w wyniku takiego błędu mogą zostać podzielone na dwa obiekty. Łączenie liter steruje operacją dylatacji, która ma niwelować takie błędy.
- Opcja dodatkowej segmentacji - Pozwala na drugie przejście segmentacji po obszarach, w których wykryto zniekształcenia liter aby uzyskać lepszą separację.
- Czułość adaptacyjnej binaryzacji - czułość algorytmu adaptacyjnego wyboru progu binaryzacji w segmentacji. Wyższe wartości mają tendencję do wprowadzenia szumu oraz łączenia ze sobą liter, jednak w niektórych dokumentach jest to potrzebne, by zwalczyć niedokładności wydruku.

Preprocessing odbywa się na obrazie, który został poddany operacji rgb2gray. Następnie obraz jest binaryzowany adaptatywnie. Pozwala to usunąć zniekształcenia wprowadzone przez gradient oświetlenia. Próg adaptatywny pozwala także na lepszą identyfikację liter, ponieważ jest wyczulony na lokalne różnice liter od tła.



Prosser, gripping his fur hat and rolling it round the top of his head, 'this bypass has got to be built and it's going to be built!'

'First I've heard of it,' said Arthur, 'why's it got to be built?'

Mr Prosser shook his finger at him for a bit, then stopped and put it away again.

'What do you mean, why's it got to be built?' he said. 'It's a bypass. You've got to build bypasses.'

Bypasses are devices which allow some people to dash from point A to point B very fast whilst other people dash from point B to point A very fast. People living at point C, being a point directly in between, are often given to wonder what's so great about point A that so many people from point B are so keen to get there, and what's so great about point B that so many people from point A are so keen to get there. They often wish that people would just once and for all work out where the hell they want to be.

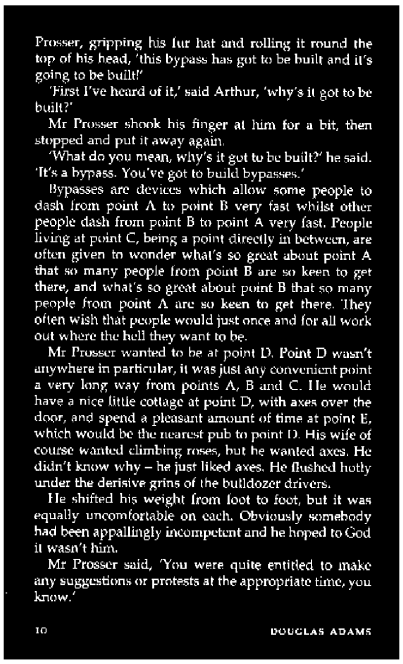
Mr Prosser wanted to be at point D. Point D wasn't anywhere in particular, it was just any convenient point a very long way from points A, B and C. He would have a nice little cottage at point D, with axes over the door, and spend a pleasant amount of time at point E, which would be the nearest pub to point D. His wife of course wanted climbing roses, but he wanted axes. He didn't know why – he just liked axes. He flushed hotly under the derisive grins of the bulldozer drivers.

He shifted his weight from foot to foot, but it was equally uncomfortable on each. Obviously somebody had been appallingly incompetent and he hoped to God it wasn't him.

Mr Prosser said, 'You were quite entitled to make any suggestions or protests at the appropriate time, you know.'

10 DOUGLAS ADAMS

(a) Obraz wejściowy algorytmu - zeskanowana strona książki „The Hitchhiker's Guide to the Galaxy" Douglasa Adamsa



Prosser, gripping his fur hat and rolling it round the top of his head, 'this bypass has got to be built and it's going to be built!'

'First I've heard of it,' said Arthur, 'why's it got to be built?'

Mr Prosser shook his finger at him for a bit, then stopped and put it away again.

'What do you mean, why's it got to be built?' he said. 'It's a bypass. You've got to build bypasses.'

Bypasses are devices which allow some people to dash from point A to point B very fast whilst other people dash from point B to point A very fast. People living at point C, being a point directly in between, are often given to wonder what's so great about point A that so many people from point B are so keen to get there, and what's so great about point B that so many people from point A are so keen to get there. They often wish that people would just once and for all work out where the hell they want to be.

Mr Prosser wanted to be at point D. Point D wasn't anywhere in particular, it was just any convenient point a very long way from points A, B and C. He would have a nice little cottage at point D, with axes over the door, and spend a pleasant amount of time at point E, which would be the nearest pub to point D. His wife of course wanted climbing roses, but he wanted axes. He didn't know why – he just liked axes. He flushed hotly under the derisive grins of the bulldozer drivers.

He shifted his weight from foot to foot, but it was equally uncomfortable on each. Obviously somebody had been appallingly incompetent and he hoped to God it wasn't him.

Mr Prosser said, 'You were quite entitled to make any suggestions or protests at the appropriate time, you know.'

10 DOUGLAS ADAMS

(b) Obraz wejściowy poddany segmentacji i odszumianiu

Rysunek 2: Algorytm podwójnej segmentacji

Następnie obraz jest oczyszczany operacją usunięcia obiektów połączonych z brzegiem obrazu, operacją otwarcia i filtrem medianowym. Operacja usuwania obiektów dotykających brzegu jest następnie wykonywana jeszcze raz. Rozmiar obszaru sąsiedztwa w tych operacjach jest definiowany przez poziom odszumiania.

Kolejnym etapem odszumiania jest odszumianie obszarowe, ma ono na celu usunięcie zanieczyszczeń w postaci grup pikseli, które przetrwały odszumianie. Za pomocą polecenia regionprops wyszukiwany jest średni obszar liter na obrazie, następnie usuwane są elementy które odstają od tego obszaru o -2.25 odchylenia standardowe. Znak jest brany pod uwagę, aby nie usunąć wielkich liter, lub liter w innych sekcjach obrazu zapisanych większą czcionką.

Po odszumianiu wykonywana jest operacja ponownego łączenia liter poprzez dylatację, jest ona sterowana przez parametr czułości łączenia liter.

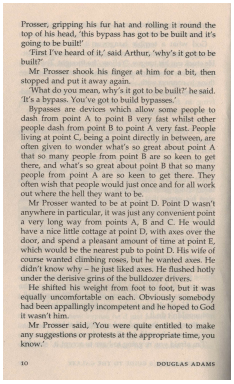
W niektórych czcionkach, które nie spełniają wymagań istnieje ryzyko połącznia się niektórych ciągów liter w spójne obszary. Jeżeli zaznaczono opcję dodatkowej separacji, to jest ona przeprowadzana na obiektach, które odstają swoją szerokością o 1.5 odchylenia standardowego od reszty obszarów. Operacja ta może także próbować segmentować wielkie litery, więc zaleca się z niej nie korzystać, jeżeli nie jest to potrzebne. Separator segmentuje takie obszary jeszcze raz z większą czułością. Wykorzystanie tej opcji wymaga ustawienia poziomu odszumiania oraz czułości łączenia na większą wartość w przypadku błędów segmentacji.

(a) Litery h oraz i sklejone ze sobą tworząc błąd segmentacji nazywany przez nas „wieloznakiem”

(b) Wieloznak rozdzielony podwójną segmentacją

Rysunek 3: Algorytm usuwania wieloznaków dodatkową segmentacją

Ostatnim zadaniem preprocesowania jest ustalenie granic akapitów. Tworzona jest kopia zbinaryzowanego obrazu i jest przeprowadzana operacja thicken, zamknięcia i wypełnienia dziur aby połaczyć sąsiadujące ze sobą linie tekstu. Dzięki temu dalsza separacja nie będzie psuta przez np. dwie kolumny tekstu po dwóch stronach dokumentu.



Prosser, gripping his fur hat and rolling it round the top of his head, 'this bypass has got to be built and it's going to be built!'

'First I've heard of it,' said Arthur, 'why's it got to be built?'

Mr Prosser shook his finger at him for a bit, then stopped and put it away again.

'What do you mean, why's it got to be built?' he said, 'It's a bypass. You've got to build bypasses.'

Bypasses are devices which allow some people to dash from point A to point B very fast whilst other people dash from point B to point A very fast. People living at point C, being a point directly in between, are often given to wonder what's so great about point A that so many people from point B are so keen to get there, and what's so great about point B that so many people from point A are so keen to get there. They often wish that people would just once and for all work out where the hell they want to be.

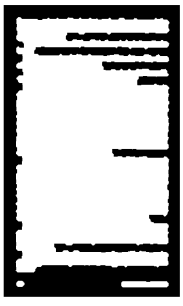
Mr Prosser wanted to be at point D. Point D wasn't anywhere in particular, it was just any convenient point a very long way from points A, B and C. He would have a nice little cottage at point D, with axes over the door, and spend a pleasant amount of time at point E, which would be the nearest pub to point D. His wife of course wanted climbing roses, but he wanted axes. He didn't know why – he just liked axes. He flushed hotly under the derisive grins of the bulldozer drivers.

He shifted his weight from foot to foot, but it was equally uncomfortable on each. Obviously somebody had been appallingly incompetent and he hoped to God it wasn't him.

Mr Prosser said, 'You were quite entitled to make any suggestions or protests at the appropriate time, you know.'

10 DOUGLAS ADAMS

(a) Obraz wejściowy dla porównania



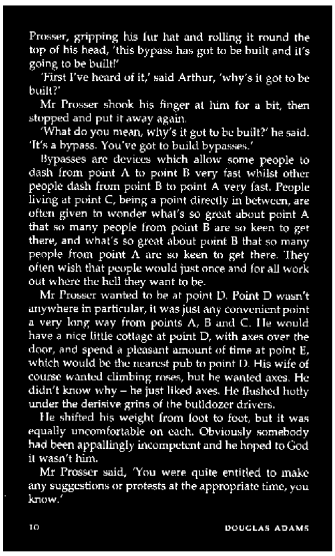
(b) Odszukane akapity tekstu

Rysunek 4: Algorytm szukania akapitów

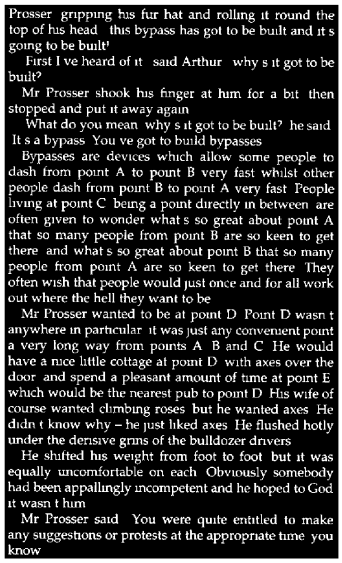
### 2.2.2 Przetwarzanie akapitów

Każdy uzyskany w następujący sposób akapit tekstu jest osobno analizowany. Separator jako argument dodatkowo przyjmuje czułość separacji słów.

Pierwszym krokiem przetwarzania jest odizolowanie kropek i znaków diakrytycznych do osobnego obrazu. Są one izolowane za pomocą wyszukania obszarów, które odstają o -1.5 odchylenia standardowego swoim polem od innych. Wartość ta została ustalona eksperymentalnie.




(a) Obraz wejściowy

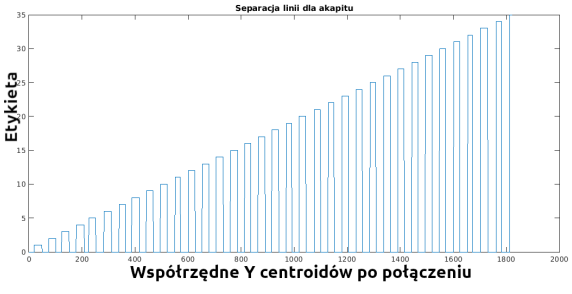



(b) Największy akapit bez kropek

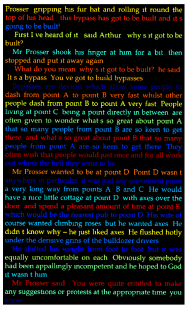
Rysunek 5: Algorytm usuwania kropek

Następnie na obrazie binarnym zawierającym tylko litery dokonywana jest analiza centroidów tych obszarów oraz ich obramowań. Ustalana jest średnia wysokość znaku, a następnie centroidy są rzutowane na oś Y obrazu. Na uzyskanym rzutowaniu dokonywana jest operacja dylatacji o sąsiedztwie równym połowie średniej wysokości znaku. Uzyskana oś jest etykietowana.

Obszary są następnie dzielone na etykiety ze względu na to, w którym obszarze na osi leżą ich centroidy. Pozwala to na dosyć niezawodne separowanie linii tekstu tak długo, jak tekst nie jest wystarczająco pochylony, by zepsuć cały proces.




(a) Centroidy rzutowane na oś po przekształceniach

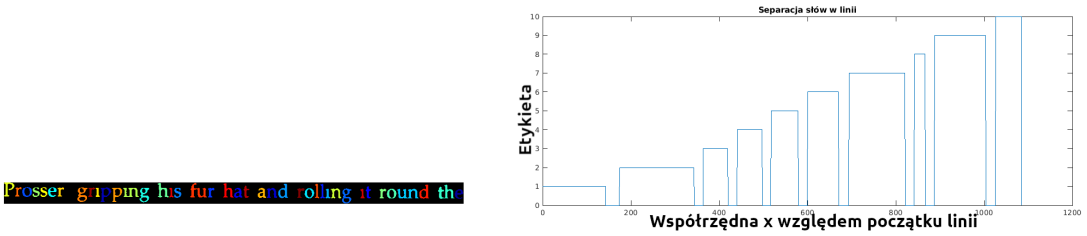


(b) Rozseparowane linie

Rysunek 6: Algorytm separacji linii

Następnie poprzednio odseparowane kropki są przypiswane liniom, w których leżą.

Najwazniejszym krokiem separacji jest połączenie liter i znaków diakrytycznych w obrębie linii oraz analiza słów. Każda uzyskana poprzednio linia tekstu jest rzutowana pikselami na oś X. Na osi X jest nastepnie dokonywana operacja zamknięcia o sąsiedztwie równym czułości sepracji słów - im większa wartość argumentu, tym mniejsze są odstępy między znakami uznawane za spację. Litery, które przed sobą mają spację są oznaczane odpowiednią flagą. Odpowiednią flagą są także oznaczane litery znajdujące się na końcu linii.




(a) Linia tekstu poddana dalszej analizie

(b) Separacja słów w linii

Rysunek 7: Algorytm szukania słów

Każda litera jest łączona z kropkami znajdującymi się w obszarze o szerokości tej litery oraz o wysokości linii - ma to znaczenie dla liter „i" i „j". Następnie tak utworzona litera jest etykietowana i przenoszona na obszar bufora o wielkości całego akapitu.

Ostatnim krokiem separatora jest przygotowanie uzyskanych liter do analizy. Litery w kolejności takiej, jak w tekście, są rzutowane na bufor o rozmiarach 1.25 * największego rozmiaru litery, a następnie tak uzyskane obszary są skalowane do obrazów 32x32px.

## 2.3 Sieć neuronowa

### 2.3.1 Wejście

Litery z części separatora przesyłane są do części recognizera. Wejściem jest tablica liter, które są obrazkami w formacie PNG o rozmiarze 32x32px.

### 2.3.2 Typ sieci

Do wykrywania liter użyto siatki neuronowej dostępnej w Matlabie. Ponieważ zagadnienie dotyczy przetwarazania obrazu - wybrano pracę z siecią konwolucyjną.

### 2.3.3 Architektura sieci

Struktura warstw sieci neuronowej:

1. **imageInputLayer([32 32 1])** - Ta warstwa jest punktem wejściowym dla sieci neuronowej. Akceptuje ona obrazy o wymiarach 32x32 pikseli z jednym kanałem kolorów (skala szarości).
2. **convolution2dLayer(3, 16, 'Padding', 'same')** - Jest to warstwa konwolucyjna, która stosuje zestaw 16 filtrów konwolucji o rozmiarze 3x3 pikseli do obrazów wejściowych. Wypełnienie ('Padding') jest ustawione na "same", co oznacza, że obraz wyjściowy ma te same wymiary co obraz wejściowy.
3. **batchNormalizationLayer()** - Ta warstwa normalizuje aktywacje poprzedzającej warstwy dla każdej grupy próbek na tej samej warstwie, zmniejszając różnorodność aktywacji.
4. **reluLayer()** - Warstwa aktywacji ReLU (Rectified Linear Unit) stosuje funkcję nieliniową max(0,x) do wszystkich jej wejść, eliminując ujemne wartości.

5. **maxPooling2dLayer(2, 'Stride', 2)** - Ta warstwa wykonuje operację max pooling, zmniejszając wymiary obrazu poprzez selekcję maksymalnych wartości z kwadratów 2x2 pikseli z krokiem (stride) 2.
6. **flattenLayer()** - Ta warstwa spłaszcza wejście do jednowymiarowego wektora, co jest niezbędne do przekazania danych do pełnej warstwy połączeń (fullyConnectedLayer).
7. **fullyConnectedLayer(256)** - Ta warstwa łączy wszystkie neurony (odpowiadające wektorowi spłaszczonego obrazu) z 256 neuronami na kolejnej warstwie.
8. **dropoutLayer(0.3)** - Ta warstwa pomaga zapobiegać przetrenowaniu (overfitting) sieci neuronowej poprzez losowe wyłączanie 30% neuronów podczas treningu, co zmusza sieć do nauczania się bardziej rozbudowanych cech.
9. **fullyConnectedLayer(52)** - Kolejna warstwa pełnych połączeń, która łączy neurony z poprzedniej warstwy z 52 neuronami na tej warstwie, co odpowiada liczbie klas, które chcemy, aby nasza sieć rozpoznawała.
10. **softmaxLayer()** - Ta warstwa zmienia wektor liczb rzeczywistych (końcowa warstwa) na prawdopodobieństwo rozkładu poprzez funkcję softmax, co oznacza, że po zastosowaniu tej funkcji wszystkie elementy wyniku sumują się do 1.
11. **classificationLayer()** - Jest to końcowa warstwa sieci neuronowej, która tworzy końcową klasyfikację modelu na podstawie prawdopodobieństw wygenerowanych przez warstwę softmax.

#### 2.3.4 Dane treningowe

1. Wykorzystano czcionki dostępne w systemie Windows, uwzględniając różne typy, takie jak: standardowe, pochyłe oraz pogrubione.
2. Stanowiły one bazę do wygenerowania licznych grafik reprezentujących poszczególne litery.
3. Kolejnym etapem było manipulowanie generowanymi grafikami – rotacja, przesunięcie, dylatacja itp., w celu jak najbardziej wiernego odwzorowania form liter na wejściu z separatora.
4. Finalnym krokiem była prezentacja wygenerowanych, czarnych liter na białym tle o wymiarach 32x32 piksele, zapisane w formacie PNG.
5. Nazwa każdego pliku tworzona była według formuły: ASCII_*.png, gdzie ASCII reprezentował numer odpowiadający danej literze, natomiast „*" symbolizował odpowniedni inny ciąg znaków.
6. Finalnie zostało wygenerowanych 24,978 plików.

#### 2.3.5 Trenowanie sieci

1. Do treningu sieci neuronowej zastosowano gotowe rozwiązanie dostarczone przez Matlab, czyli funkcję trainNetwork.
2. Proces treningu został skonfigurowany z parametrem batchsize ustawionym na 128.
3. Ilość epok, przez które sieć była trenowana, została ustawiona na 50.
4. Learning rate został ustawiony na 0.005

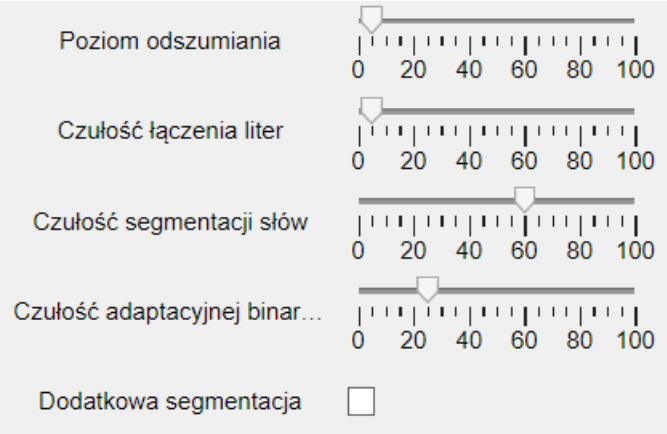
#### 2.3.6 Rezultaty sieci

Udało się osiągnąć 97.28% dokładności danych walidacyjnych.

# 3 Instrukcja użytkownika

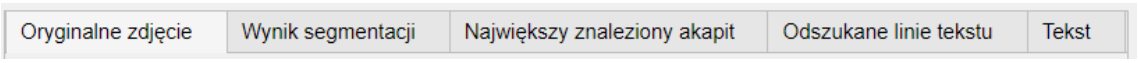
## 3.1 Dobór ustawień

Program po uruchomieniu zawiera zalecane ustawienia, jednak dla niektórych obrazów może zaistnieć potrzeba ich modyfikacji.




Rysunek 8: Parametry segmentacji.

Obraz poddany segmentacji można obejrzeć w zakładce „Wynik segmentacji", natomiast wyniki identyfikacji liter i linii znajdują się w zakładkach „Największy znaleziony akapit" oraz „Odszukane linie tekstu".




Rysunek 9: Zakładki.

- Jeżeli obraz po segmentacji zawiera szum, należy zwiększyć poziom odszumiania. Zbyt wysoka wartość może zniszczyć niektóre obszary.
- Jeżeli litery po segmentacji są podzielone na części zbliżone do siebie, należy zwiększyć czułość łączenia liter, zbyt wysoka wartość może łączyć ze sobą inne litery.
- Jeżeli litery w tekście są ze sobą połączone, należy włączyć ustawienie dodatkowej segmentacji. Przy korzystaniu z dodatkowej segmentacji należy obniżyć poziom odszumiania i zwiększyć czułość łączenia liter, aby uzyskać lepsze wyniki.
- Jeżeli tekst wynikowy zawiera wiele słów połączonych ze sobą - należy zwiększyć czułość segmentacji słów. Jeżeli słowa są podzielone na litery - należy ją zmniejszyć.
- W zależności od obrazu może zaistnieć potrzeba zmiany czułości adaptacyjnej binaryzacji, jednak modyfikacja tego ustawienia nie jest zalecana bez próby uzyskania lepszej segmentacji poprzednimi metodami.
- Jeżeli nadal przy zmianie opcji segmentacji występują wieloznaki, to użycie opcji dodatkowej segmentacji może pomóc je rozdzielić - jednak przy dużym ryzyku uszkodzenia segmentacji wielkich liter, takich jak np. „W".

## 3.2 Uruchomienie

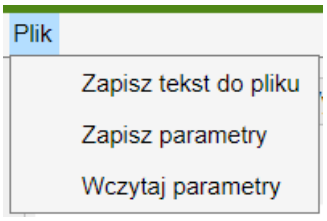
Najpierw należy załadować plik za pomocą przycisku „Wybierz plik", po czym dostosować parametry segmentacji zgodnie z instrukcją powyżej (można też załadować zapisane wcześniej pliki z parametrami).

Aby ułatwić dobór ustawień można zamiast generatora uruchomić podgląd segmentacji. Pozwala on w zakładce „Wynik segmentacji” zobaczyć jak zidentyfikowane zostały spójne obszary w tekście bez uruchamiania algorytmu wykrywania linii, który trwa znacznie dłużej.

Po dobraniu ustawień należy nacisnąć przycisk „Generuj” i poczekać, aż zmieni on kolor z żółtego (oznaczającego ładowanie) na chwilę na zielony (oznaczającego zakończenie procesu). W zakładkach „Wynik segmentacji”, „Największy znaleziony akapit”, oraz „Odszukane linie tekstu” zostaną przedstawione etapy pracy identyfikacji liter, natomiast w zakładce „Tekst” będzie można wyświetlić tekst odczytany przez sieć neuronową na bazie przesłanych do niej liter.

**Jeżeli segmentacja nie powiodła się całkowicie (np. przez niekompatybilność obrazu z segmentatorem lub zły dobór ustawień) program zamiast przykładu akapitu zwróci całkowicie biały obszar w zakładce „Największy znaleziony akapit”.**

Wygenerowany tekst można zapisać do pliku za pomocą przycisków w pasku menu na górze okna aplikacji. Jest tam też możliwość zapisania oraz wczytania parametrów segmentacji.




Rysunek 10: Menu „Plik”.

# 4 Instalacja

Otworzenie aplikacji ze środowiska Matlab (pliki .m):

1. Pobierz i wypakuj pliki do wybranego folderu.
2. Zainstaluj poniższe rozszerzenia do Matlaba:
   - Deep Learning Toolbox
   - Image Processing Toolbox
3. W Matlabie ustaw katalog roboczy na ten, w którym znajdują się wypakowane pliki
4. Dodaj do ścieżki wszystkie foldery i podfoldery.
5. Uruchom plik ui/AppUI_exported_ver1.m

W folderze „examples” znajdują się obrazy, na których testowano działanie aplikacji. Pliki .txt zawierają parametry segmentacji do obrazów o tej samej nazwie (jeżeli nie istnieje taki plik tekstowy, to do obrazu pasują parametry z basic_example.txt).

# 5 Podział pracy

- Interfejs Graficzny UI - Aleksandra Śliwska
- Separator - Jakub Kawka
- Recognizer/sieć neuronowa - Daria Kokot i Kacper Duda
- Dokumentacja - wspólny wysiłek

# 6 Problemy, pomysły na rozwój

## 6.1 Problemy separatora

Odpowiednia segmentacja obrazu i identyfikacja poszczególnych linii oraz słów nie jest prosta – separator przez cały proces powstawania programu był testowany pod kątem różnych błędów, które powodują nieprawidłową separację. Aby usunąć część z tych problemów, zostały dodane elementy UI, które pozwalają na dobranie odpowiednich parametrów separacji.

- Jeżeli tekst jest zbyt pochyły separator może mieć problemy z identyfikacją linii.
- Tekst zawierający obrazy lub inne elementy graficzne może zepsuć separację.
- Tekst o zbyt małych odstępach pomiędzy liniami może zepsuć separację.
- Tekst o zbyt małych odstępach liter może powodować, że kawałki niektórych liter zostaną zabrane przez inne litery.
- Tekst o zbyt małych ostępach może powodować, że niektóre grupy liter mogą zostać połączone w jedną. Opcja dodatkowej segmentacji próbuje walczyć z tym problemem.
- Opcja dodatkowej segmentacji nie działa, jeżeli nie jest ustawiona odpowiednio wysoka wartość odszumiania i łączenia liter.
- Skany lub zdjęcia tekstu drukowanego, w zależności od jakości wydruku, mogą być segmentowane tak, że braki tuszu w obrębie litery spowodują podzielenie jej na kilka obszarów.
- Zbyt wysoka wartość odszumiania niszczy litery.
- Zbyt wysoka czułość łączenia liter łączy sąsiadujące litery ze sobą.
- Złe ustawienie paremetru czułości segmentacji słów może doprowadzić do połączenia ze sobą wyrazów lub rozdzielenia liter wewnątrz wyrazów.
- Problemy z separacją mogą powodować obrazy, w których części tekstu są zapisane różnymi wielkościami czcionki

Większość problemów separatora jest powodowana przez podejście do problemu - wykorzystywane są praktycznie tylko operacje morfologiczne oraz analiza własności obszarów do przeprowadzania segmentacji.
Wykorzystanie sieci neuronowej do identyfikacji obiektów litera/szum oraz do odpowiedniej separacji wieloznaków znacznie zwiększyłaby jakość separowanego tekstu.
Część zmiennych sterująca separacją została także ustawiona wewnątrz programu „na stałe". Znacznym ulepszeniem separatora byłoby dobieranie tych wartości w zależności od analizowanego tekstu.

Przykład, który sprawia najwięcej problemów separatorowi i powoduje łączenie się liter mimo ustawienia dodatkowej segmentacji jest skan strony z książki „Rok 1984" Georgea Orwella – litery są często wydrukowane połączone ze sobą pełną czernią atramentu. Podobne trudności sprawia skan wykorzystany w sekcji przedstawiającej działanie separatora.
Inne trudności sprawia przykład skanu z książki science fiction dostępny w katalogu przykładów o nazwie ALIEN.jpg – papier ma bardzo specyficzną nadrukowaną teksturę która dodaje duże ilości szumu przy próbie bardziej czułej segmentacji - która jest potrzebna, ponieważ nierówny rozkład tuszu w niektórych literach powoduje podzielenie ich przez segmentację na pół.

## 6.2 Problemy recognizera

Regonizer uznajemy za dobrze skalibrowany, ale nadal ma problemy z:

- rozpoznawaniem niektórych małych i dużych liter
- myleniem „l" (małe „L") z „I" (duże „i")
- obsługą wielu czcionek i mocno zdeformowanych liter.

Dodatkowo nie rozpoznaje cyfr oraz polskich znaków.

## 6.3 Dalszy rozwój i wnioski

Dobrym pomysłem byłoby rozwiązanie problemów wmienionych w poprzednich punktach, ale program mógłby też w przyszłości:

- być podłączony do API dużego modelu językowego LLM, takiego jak ChatGPT 4, w celu usunięcia błędów związanych z niepoprawnym rozpoznaniem pojedynczych liter - na przykład z poleceniem „To surowy wynik programu OCR, napraw błędy w tekście i popraw jakość formatowania"
- zostać przeportowany na inne urządzenia, jak smartfony, którymi łatwiej robić zdjęcia - obraz byłby przesyłany na serwer w celu przetworzenia, a rezultat w postaci tekstu byłby odesłany spowrotem do użytkownika
- przetwarzać wiele plików na raz w celu szybkiej obsługi dużej ilości dokumentów, na przykład wielostronicowych skanów.
- porzucić metody morfologiczne na rzecz dobrze wytrenowanej sieci neuronowej do wykrywania tekstu, a dopiero po ekstrakcji tekstu dalej przetwarzać obrazy i analizować tekst innymi sieciami neuronowymi
- wykorzystać znacznie szerszy dataset do trenowania sieci, a także rozpoznawać pismo odręczne i szerszy zakres znaków, niż same litery a-z A-Z
- rozpoznawać symbole matematyczne w pracach naukowych i generować na ich podstawie kod LaTeX