Analiza obrazów

**Projekt: rozpoznawanie tekstu ze zdjęcia**

Piotr Kowalczyk

Maciej Dąbrowski

Zuzanna Sulima

# Spis treści:

1. [Tytuł projektu i jego autorzy](#_bookmark0)
2. [Opis projektu](#_bookmark1)
3. [Złożenia wstępne przyjęte w realizacji projektu](#_bookmark2)
4. [Analiza projektu](#_bookmark3)
   1. [Specyfikacja danych wejściowych](#_bookmark4)
   2. [Opis oczekiwanych danych wyjściowych](#_bookmark5)
   3. [Zdefiniowanie struktur danych](#_bookmark6)
   4. [Specyfikacja interfejsu użytkownika](#_bookmark7)
   5. [Wyodrębnienie i zdefiniowanie zadań](#_bookmark8)
   6. [Decyzja o wyborze narzędzi programistycznych](#_bookmark9)
5. [Podział pracy i analiza czasowa](#_bookmark10)
6. [Opracowanie i opis niezbędnych algorytmów](#_bookmark11)
7. [Kodowanie](#_bookmark12)
8. [Testowanie](#_bookmark13)
9. [Wdrożenie, raport i poprawki](#_bookmark14)

# Tytuł i autorzy projektu

Wykonano projekt pod tytułem „Rozpoznawanie tekstu ze zdjęcia”, którego autorami są:

* Piotr Kowalczyk
* Maciej Dąbrowski
* Zuzanna Sulima

# Opis projektu

Celem projektu było stworzenie programu umożliwiającego rozpoznawanie tekstu zawartego na wgranym obrazie. Użytkownik może wgrać dowolny obraz zawierający tekst, a następnie przygotować go do rozpoznawania. Otrzymany przetworzony obraz, może przekonwertować na tekst.

# Założenia wstępne przyjęte w realizacji projektu

Kierując się wcześniej zdobytym doświadczeniem do wykonania projektu zastosowano program MATLAB. Powstała aplikacja umożliwia rozpoznawanie tekstu zawartego na obrazie wczytanym przez użytkownika.

* h

# Analiza projektu

## Specyfikacja danych wejściowych

Danymi wejściowymi przyjmowanymi przez program są obrazy o rozszerzeniu PNG oraz JPG zawierające tekst drukowany.

## Opis oczekiwanych danych wyjściowych

## Dane wyjściowe będzie reprezentować rozpoznany ciąg liter/znaków z obrazu wejściowego.

## Zdefiniowanie struktur danych

## W programie są zdefiniowane następujące struktury danych:

## originalImage – obiekt reprezentujący oryginalny obraz, wgrany przez użytkownika

## binaryImage – zmienna globalna reprezentujący obraz przygotowany do rozpoznawania tekstu

## objects – obraz binarny reprezentujący obiekty (linijki tekstu)

## merged – obraz, gdzie każdy znak otrzymał unikalną etykietę

## objectsPosition – tablica obiektów reprezentująca pozycje obiektów (znaków)

## labeledObjects – tablica obiektów reprezentująca etykietowane obiekty

## mainWindow – obiekt reprezentujący interfejs użytkownika i będący kontenerem dla pozostałych elementów interfejsu użytkownika

## Specyfikacja interfejsu użytkownika

## Na interfejs użytkownika składają się trzy główne elementy: przycisk „Wybierz zdjęcie”, po kliknięciu którego użytkownik może wybrać obraz, który będzie analizowany przez aplikację, przycisk „Przygotuj zdjęcie”, po kliknięciu którego aplikacja przetwarza oryginalny obraz i przygotowuje go do procesu rozpoznawania tekstu oraz pole tekstowe, w którym rozpoznany tekst jest wyświetlany.

## Wyodrębnienie i zdefiniowanie zadań

Planując sposób realizacji projektu zdecydowano się na podział na mniejsze, niezależne moduły:

* Stworzenie bazy danych do trenowania sieci
* Opracowanie algorytmu do trenowania sieci
* Stworzenie interfejsu użytkownika
* Przygotowanie obrazu do rozpoznawania tekstu
* Opracowanie algorytmu odczytywania z obrazu
* Opracowanie dokumentacji opisującej projekt

## Decyzja o wyborze narzędzi programistycznych

Zdecydowano, aby przy realizacji projektu korzystać z programu MATLAB. Narzędzie to wybrano ze względu na doświadczenie w jego stosowaniu zdobyte podczas tego semestru.

# Podział pracy i analiza czasowa

Po wspólnym omówieniu sposobu realizacji projektu, zdecydowano o podziale

na mniejsze moduły ułatwiające niezależną pracę.

|  |  |
| --- | --- |
| Piotr Kowalczyk | Stworzenie bazy danych do trenowania sieci, Opracowanie algorytmu do trenowania sieci |
| Maciej Dąbrowski | Stworzenie interfejsu użytkownika, Przygotowanie obrazu do rozpoznawania tekstu, Opracowanie algorytmu odczytywania z obrazu |
| Zuzanna Sulima | Opracowanie dokumentacji opisującej projekt |

# Opracowanie i opis niezbędnych algorytmów

**Algorytm binaryzacji** (funkcja **prepareImage**) – używany w celu przetworzenia oryginalnego obrazu na obraz binarny wykorzystywany do rozpoznawania tekstu. Polega na przekształceniu obrazu w skali szarości na obraz, w którym piksele są białe bądź czarne, w zależności od tego czy przekroczą progowanie.

**Algorytm OCR** (Optical Character Recognition) (funkcja **translateImage**) – używany do rozpoznawania i konwersji tekstu z obrazów na tekst czytelny dla maszyny

**Algorytm scalania obiektów** (funkcja **mergeObjects**) – polega na scaleniu obiektów, które się na siebie nakładają, co pozwala na łatwiejsze przetwarzanie tekstu

**Algorytm analizy obiektów** (funkcja **regionprops**) – pozwala na analizę obiektów znajdujących się na obrazie (w tym wypadku analiza ich pozycji)

**Algorytm etykietowania** (funkcja **bwlabel**) – polega na oznaczeniu obiektów po ich scaleniu dzięki przypisaniu im unikalnej etykiety

# Kodowanie

* **prepareImage**– funkcja przygotowująca obraz do rozpoznawania tekstu. Przyjmuje trzy parametry: obraz wejściowy, wartość logiczną mówiącą czy tekst jest za ciemny oraz poziom wrażliwości progowania. Zwraca ona zbinaryzowany obraz
* **OCR** – główna funkcja uruchamiająca okno aplikacji
* **loadImage** – funkcja wywoływana po naciśnięciu „Wybierz zdjęcie”. Otwiera okno dialogowe do wyboru pliku PNG lub JPG i ładuje wybrany obraz
* **loadPreparedImage** – funkcja wywoływana po naciśnięciu „Przygotuj zdjęcie”. Przetwarza oryginalny obraz i wyświetla wynik
* **translateImage** – funkcja wywoływana po naciśnięciu „Konwertuj na tekst”. Rozpoznaje tekst na przygotowanym obrazie i wyświetla wynik w polu tekstowym
* **gridLayout** – obiekt reprezentujący siatkę dla elementów interfejsu użytkownika. Jest odpowiedzialny za wymiary oraz rozmieszczenie elementów na ekranie
* **originalImage** – obiekt reprezentujący oryginalny obraz wczytany przez użytkownika
* **preparedImage** – obiekt reprezentujący obraz po przetworzeniu przez funkcję prepareImage
* **textArea** – obiekt reprezentujący pole tekstowe, w którym wyświetlany jest rozpoznany tekst
* **selectButton, prepareButton, converButton** – obiekty reprezentujące przyciski wywołujące funkcje po naciśnięciu
* **mergeObjects–** funkcja przyjmująca obraz binarny z obiektami (znakami) i scalająca nachodzące na siebie obiekty, co pozwala na łatwiejsze przetwarzanie tekstu

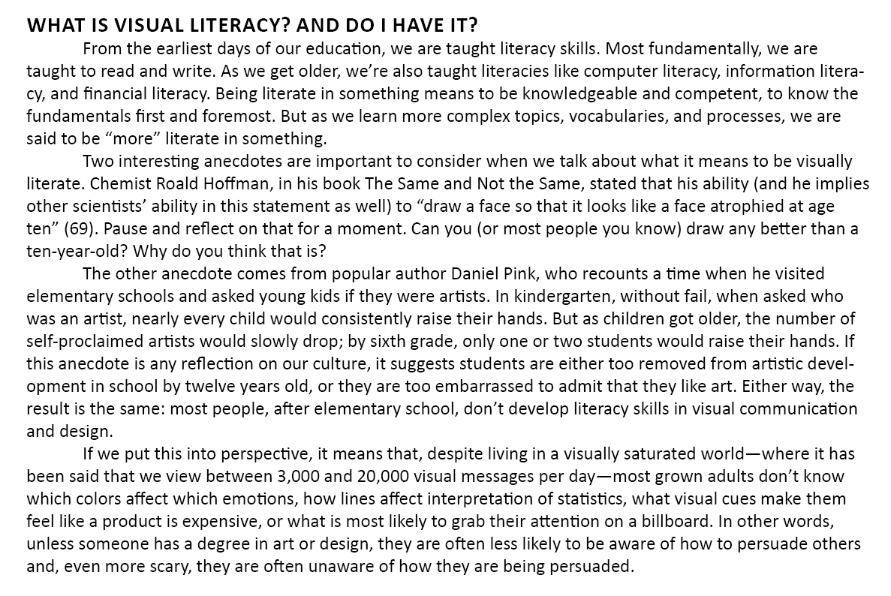
# Testowanie

Program testowano manualnie, sprawdzając kolejne zaimplementowane funkcjonalności.

Sprawdzono poprawność działania wczytania obrazu z pliku PNG lub JPG, przygotowanie zdjęcia do rozpoznawania tekstu oraz konwersję obrazu na tekst.

# Wdrożenie, raport i wnioski

# Stwierdzono, że wczytywanie zdjęcia oraz jego przygotowanie są przeprowadzane bezproblemowo. Konwersja na tekst nie działa jednak w pełni idealnie. Niektóre znaki są rozpoznawane niepoprawnie, lecz jest to mały procent całości tekstu.



**Obraz wczytany w celu rozpoznania tekstu**

Obraz zawierający tekst

Opis wygenerowany automatycznie

**Tekst rozpoznany przez aplikację**