

Analiza i Przetwarzanie Obrazów

# DOKUMENTACJA PROJEKTU

05.07.2023

Dominik Dziuba  
Łukasz Wajda  
Mariusz Biegański  
Mateusz Niepokój  
Przemysław Rodzik  
Szymon Pawelec

### 1. Cele i założenia projektu

Celem projektu było stworzenie programu pozwalającego użytkownikowi na wyszukanie najkrótszej drogi pomiędzy zaznaczonymi na mapie punktami. Powinien on zapewniać odpowiedni interfejs, który umożliwia wczytanie wybranej przez siebie mapy, ustawienie punktu początkowego i końcowego, a algorytm wyszukujący trasę powinien brać pod uwagę szerokość dostępnych dróg. Efektem oczekiwanym działania programu ma być wizualizacja na przekazanym obrazku optymalnej z punktu widzenia algorytmu trasy.

### 2. Opis zrealizowanego rozwiązania

Projekt został zrealizowany jako aplikacja desktopowa (.exe) z graficznym interfejsem użytkownika. Umożliwia ona wczytanie wybranego przez użytkownika obrazu, jak również konfigurację parametrów odpowiadających za odpowiednie przekształcenie obrazu i wyszukanie optymalnej ścieżki. Ustawienie punktów początkowego i końcowego szukanej trasy możliwe jest poprzez zaznaczenie wybranych miejsc na wczytanej mapie.

### 3. Generowanie danych

Na potrzeby stworzenia algorytmu wyszukującego najkrótszą drogę, wygenerowano zbiór obrazów przy pomocy programu Easy Diffusion. Jest to narzędzie oparte na sztucznej inteligencji, podobne do Stable Diffusion, które ma za zadanie przekształcać tekst na obrazy.

W pierwszym etapie procesu wybrano fragment mapy bez etykiet ze strony <https://snazzymaps.com/style/24088/map-without-labels>. Ten konkretny obraz został wykorzystany jako podstawa do generowania nowych obrazów przy użyciu narzędzia Easy Diffusion. Z wygenerowanych obrazów wybrano jeden, który został wykorzystany do stworzenia końcowego zbioru obrazów. Zbiór ten użyty został do testowania algorytmu wyznaczania trasy.

Ostateczny prompt, który służył do tego celu, prezentuje się następująco:

*„map with white roads, only roads are white, city, road with difference width, very wide road, clear roads, all roads connected, black and white”.*

Zdefiniowany został również negative prompt:

*„label, text, characters, many colors, colored roads, color blending”.*

Dodatkowe parametry ustawione w generatorze to:

- Seed: 1
- Number of Images: 32
- Model: sd-v1-4
- Image size: 1280x768
- Inference Steps: 25
- Guidance Scale: 28
- Prompt Strength: 0,82
- Image Quality: 95


Enter Prompt or Load from a file

map with white roads, only roads are white, city, road with difference width, very wide road, clear roads, all roads connected, black and white

Negative Prompt (optional)

label, text, characters, many colors, colored roads, color blending

Initial Image (img2img) (optional)



2048 x 2048

Browse

Draw

Inpaint ☐

☐ Preserve color profile (helps during inpainting)

Image Modifiers 3

Cel Shading

Hard Edge Painting

Linocut

Make 32 Images

Image Settings

Image Settings

Seed:  ☐ Random

Number of Images:  (total)  (in parallel)

Model: sd-v1-4

Custom VAE: None

Image Size: 1280 (width) 768 (\*) (height)

Inference Steps:

Guidance Scale:

Prompt Strength:

Hypernetwork: None

Output Format: png

Image Quality:

Render Settings

☐ Show a live preview (uses more VRAM, slower images)

☐ Fix incorrect faces and eyes GFPGANv1.3

☐ Scale up by 4x with RealESRGAN\_x4plus

☒ Show only the corrected/upscaled image

Scr. 1: Generowanie obrazów

## 4. Szczegółowy opis algorytmu

Algorytm wyszukiwania ścieżki można podzielić na kilka etapów. W pierwszym z nich, sprawdzany jest poprawny wybór punktów i dostępność wczytanego przez użytkownika obrazu. Ustawiane są również parametry takie jak kolor znalezionej ścieżki, czy jej szerokość. Następnie obraz jest odpowiednio przygotowywany do wykonania procesu wyszukiwania drogi.

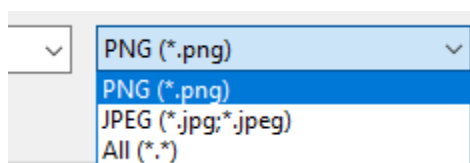
Najpierw obraz mapy poddawany jest przekształceniu do skali szarości, dzięki czemu zwiększana jest czytelność dróg w stosunku do tła. Następnie, z wszystkich zaznaczonych przez użytkownika punktów pobierane są wartości pikseli. Pozwala to na określenie minimalnej i maksymalnej wartości, czyli zakresu, w jakim algorytm ma wyszukiwać ścieżkę. Przeprowadzone dla każdego punktu progowanie scalane jest w jeden obraz za pomocą maski bitowej or. W kolejnym kroku, przy pomocy filtra uśredniającego i maski bitowej and, wyznaczane są lokalne szerokości dróg. Ponadto, ustawiana jest minimalna wartość pikseli na obrazie.

Dla tak przygotowanych danych, przeprowadzany jest zmodyfikowany algorytm Dijkstry. Polega on na wyszukaniu optymalnej ścieżki pomiędzy 2 punktami w taki sposób, jakby szerokie drogi miały mniejszy koszt przejścia. Otrzymana w ten sposób trasa jest następnie rysowana i nakładana na obraz wejściowy.

## 5. Interfejs użytkownika

Interfejs użytkownika zawiera szereg parametrów. Dwa pierwsze z nich, *Max Image Width* i *Max Image Height* odpowiadają za rozmiar wczytanego obrazka.

Przy pomocy przycisku *Load Photo*, użytkownik może wczytać wybrany przez siebie obraz mapy, domyślnie z rozszerzeniem .png. Możliwe jest jednak wybranie rozszerzenia .jpg, .jpeg, lub jakiegokolwiek innego.



Scr. 2: Wczytywanie obrazu

Kolejny parametr, *Minimum Pixel Weight*, odpowiada za minimalną wartość piksela na obrazie. (?)

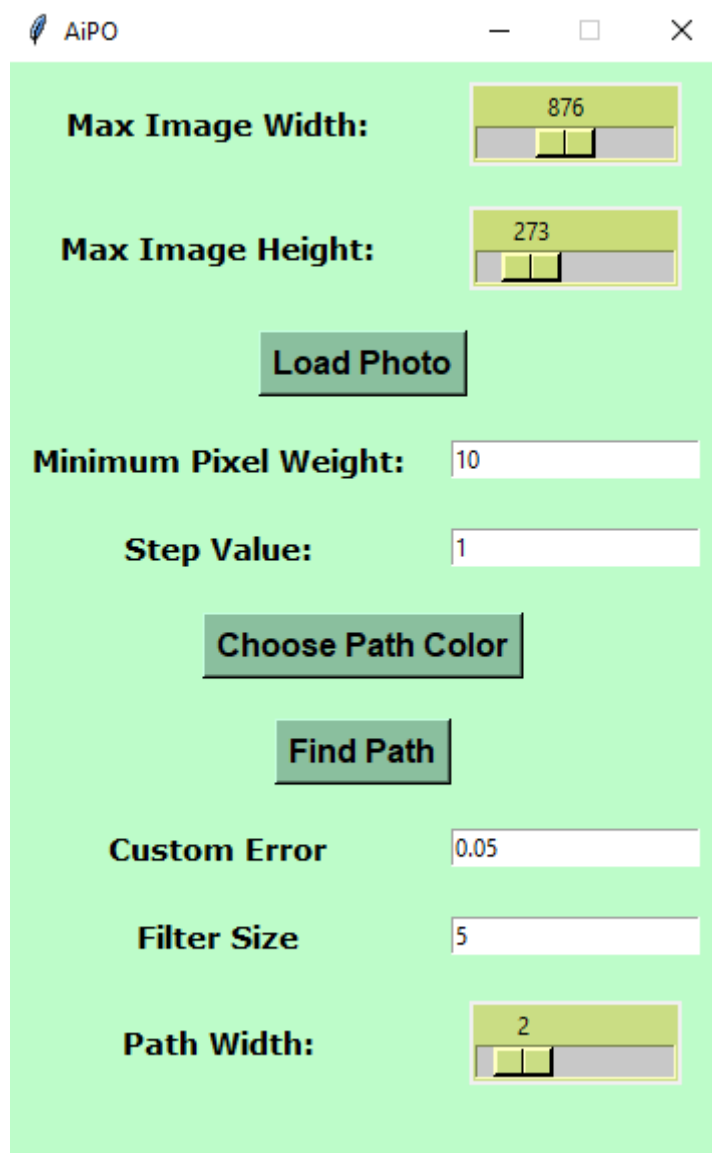
Parametr *Step Value* związany jest z procesem rysowania znalezionej przez algorytm ścieżki. Im większa wartość parametru, tym większa odległość pomiędzy zaznaczanymi punktami.

Przycisk *Choose Path Color* służy do wybrania koloru rysowanej ścieżki.

*Find Path* uruchamia algorytm.

Wartości parametrów *Custom Error* i *Filter Size* odpowiadają kolejno za dopuszczalny błąd, czyli tolerancję przy wyznaczaniu wartości progowych pikseli oraz za rozmiar filtra uśredniającego.

Ostatni element, *Path Size*, określa szerokość rysowanej ścieżki.



The screenshot shows the AiPO application window with a light green background. At the top, there is a title bar with the AiPO logo and standard window controls. The interface contains several input fields and buttons:

- Max Image Width:** A slider control with a value of 876.
- Max Image Height:** A slider control with a value of 273.
- Load Photo:** A button with a dark green background and white text.
- Minimum Pixel Weight:** A text input field containing the value 10.
- Step Value:** A text input field containing the value 1.
- Choose Path Color:** A button with a dark green background and white text.
- Find Path:** A button with a dark green background and white text.
- Custom Error:** A text input field containing the value 0.05.
- Filter Size:** A text input field containing the value 5.
- Path Width:** A slider control with a value of 2.

Scr. 3: Interfejs użytkownika