

Rysowanie obrazu przy pomocy Transformaty Fouriera

Aleksandra Kuś, Karol Szeliga, Katarzyna Romańczyk

1. Temat

Celem projektu jest stworzenie programu, który tworzy obraz za pomocą dyskretnej transformaty Fouriera. Efektem jest animacja rysowania zdjęcia za pomocą obracających się kółek połączonych promieniami. Program został napisany w programie MATLAB, wersja R2020b.

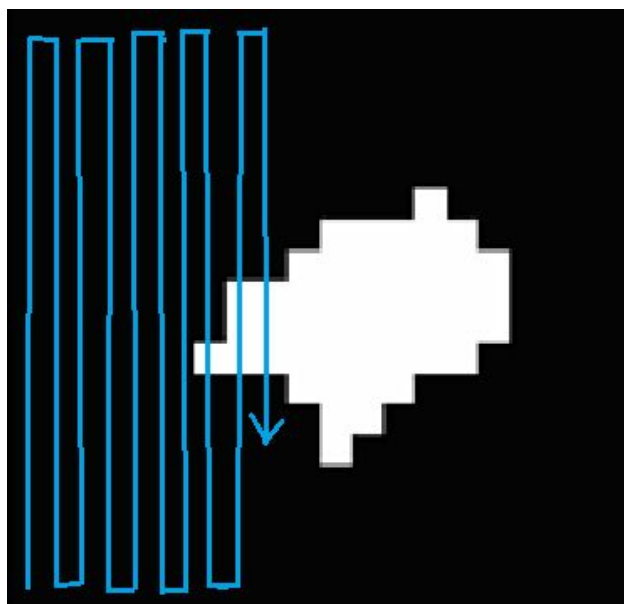
2. Użyte algorytmy

a. Kolejność rysowanych pikseli

Poddanie algorytmom transformaty i animacji obrazu bez ustalenia kolejności rysowania pikseli skutkuje tym, że obraz będzie się rysował kolejnymi kolumnami zaczynając od jednego z rogów, zgodnie z numeracją tablicy pikseli, przeskakując od końca kolumny do początku następnej za każdym razem. Wprowadzono zatem algorytmy zmieniające kolejność rysowania pikseli, aby temu zapobiec.

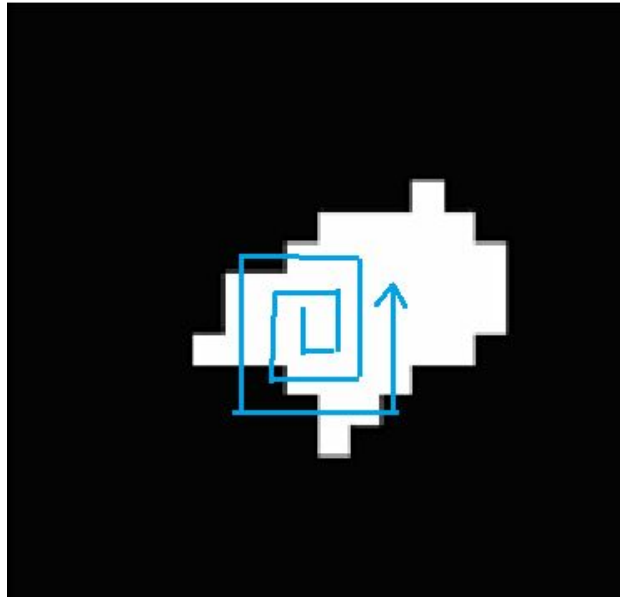
Na wejściu dostają obraz, a na wyjściu zwracają tablicę ze współrzędnymi kolejnych pikseli obrazu.

Pierwszy, nazwany ładowaniem prostym, bierze kolejne kolumny obrazu zmieniając za każdym razem zwrot, przez co eliminuje się wyraźny skok pomiędzy kolejnymi kolumnami.



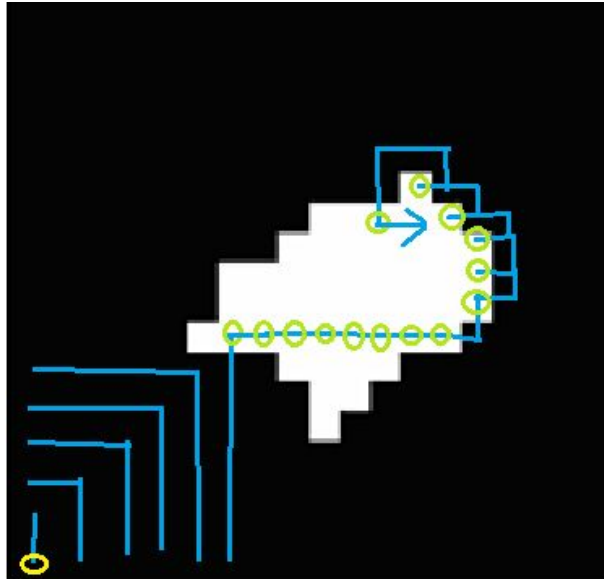
Rys. 1 Ładowanie Proste. Niebieska linia oznacza kolejność zapisu pikseli do tablicy wynikowej.

Drugi algorytm, ładowanie spiralne, polega na zbieraniu kolejnych pikseli w tablicy spiralą od środka obrazu. Jeśli obiekt ma osie symetrii, rysowanie go tym sposobem będzie płynne, bez większych przeskoków.



Rys. 2 Ładowanie Spiralne. Niebieska linia oznacza kolejność zapisu pikseli do tablicy wynikowej.

Trzeci sposób to ładowanie po linii. Rozpoczyna szukanie od prawego dolnego rogu robiąc kwadratową spiralę. Jeśli znajdzie pierwszy piksel obiektu, zaczyna nową spiralę, mającą środek w pierwszym pikselu. Za każdym razem, gdy znajdzie nowy piksel, zaczyna nową spiralę. Istnieje jednak możliwość zapisania jednego piksela wielokrotnie. Przez to za każdym razem, gdy znajdzie piksel, najpierw sprawdza czy nie został już zapisany. Zapewnia to poprawne wykonanie algorytmu, ale zarazem zwiększa znacznie czas jego wykonania.



Rys. 3 Ładowanie po linii. Niebieska linia oznacza kolejność szukania pikseli do zapisu w tablicy wynikowej. Żółte kółko oznacza początkową spiralę, a zielone kolejne środki spiral (i także kolejne piksele w tabeli wynikowej).

b. Transformata fouriera

Mając tablice ze współrzędnymi kolejnych pikseli można poddać je transformacie fouriera i uzyskać szukane okręgi rysujące.

Aby tego dokonać przekształcamy współrzędne do postaci zespolonej (współrzędna x to część rzeczywista, y to urojona). Następnie poddajemy je dyskretnej transformacie fouriera:

$$X_k(k) = \sum_{n=0}^N x_c(n) e^{\frac{2\pi i}{N} - (k-1)(n-1)}, \quad 1 \leq k \leq N-1$$

gdzie: i - jednostka urojona, k – numer harmonicznej, n – numer próbki sygnału, x_c – wartość próbki sygnału, N – liczba próbek. W naszym przypadku sygnał to piksele obrazu.

Możemy zatem z tego wyciągnąć wszystkie potrzebne informacje. Zmienna k oznacza częstotliwości obrotów kolejnych okręgów. Wartości bezwzględne uzyskanych współczynników $X_k(k)$ to długości promieni okręgów, a ich kąty fazowe to początkowe ustawienia okręgów. Te wszystkie dane, posortowane według długości promieni, zwraca algorytm.

c. Animacja

Algorytm transformaty dostarczył informacje o promieniach, częstotliwościach i fazach wszystkich kółek. Mając te trzy tablice można przygotować kolejne klatki animacji. Klatek jest tyle co okręgów. Najprościej zapisać w tablicy współrzędne środków kolejnych okręgów. Każda klatka tworzy się poprzez zapisanie do tablicy tych współrzędnych, biorąc przepisy na kolejne okręgi w danym stopniu obrotu (jeden obrót oznacza narysowanie całego obrazu). Powstaje tablica tablic wszystkich klatek, która umożliwia szybką animację. Do narysowania każdej klatki posłużyły optymalne funkcje rysujące okręgi (viscircles) i linie (plot bez ".").

3. Podział zadań
 - a. Karol Szeliga
 - i. transformata Fouriera
 - ii. rysowanie obrazu
 - iii. dokumentacja
 - b. Katarzyna Romańczyk
 - i. elementy edycji zdjęcia
 - ii. przygotowanie aplikacji
 - iii. dokumentacja
 - c. Aleksandra Kuś
 - i. edycja zdjęcia
 - ii. przygotowanie aplikacji
 - iii. dokumentacja
4. Analiza testowych wyników

Wyniki dla obrazów wybranych przez nas do testów:

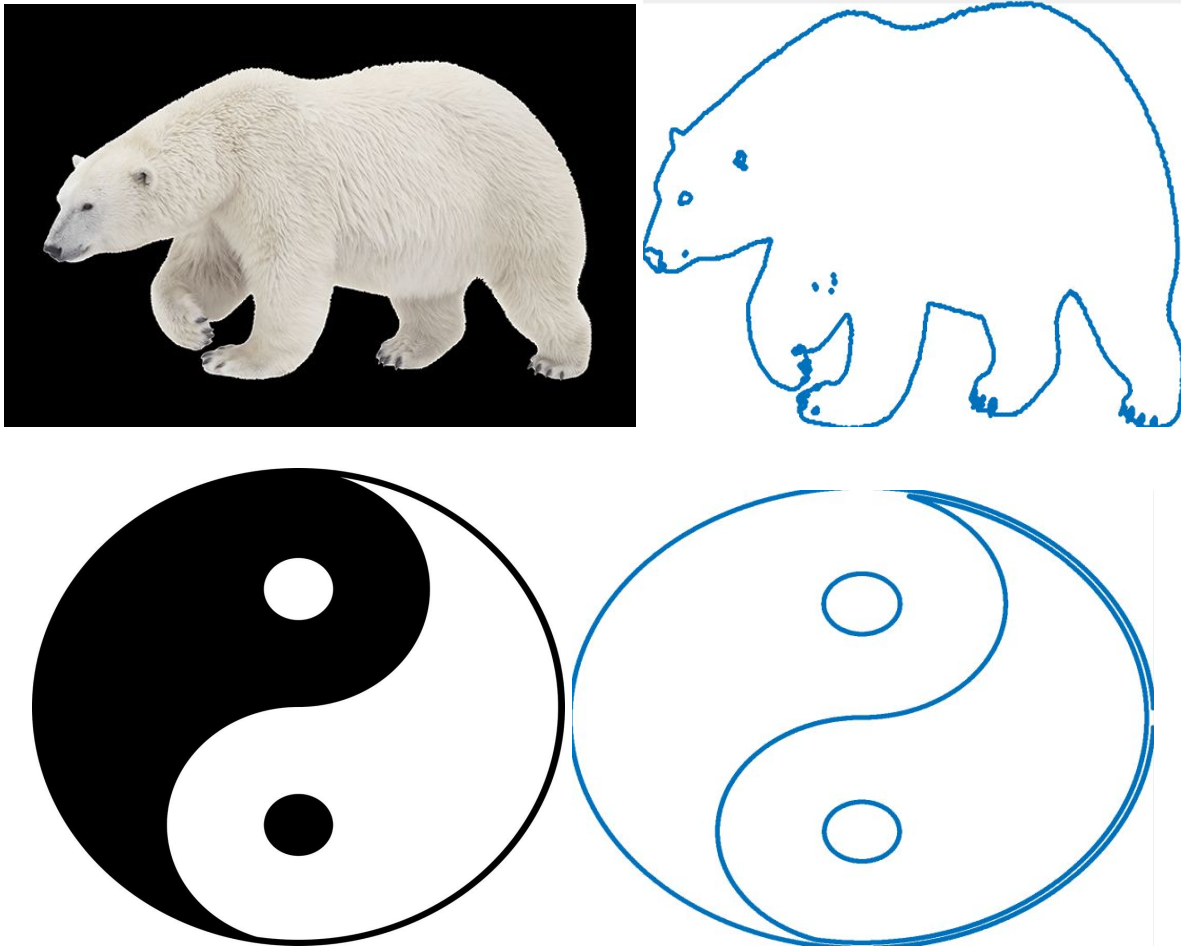
Obraz pierwszy, choć jest szczegółowy i brak na nim dokładnych konturów, udało się narysować w programie dzięki zmianie progu binaryzacji. W pokazanym poniżej przykładzie przy niewielkiej wartości progu liczba pikseli do narysowania nie jest zbyt duża, a w efekcie otrzymaliśmy obraz na którym zachowane zostało podobieństwo. W innych konfiguracjach (bez zastosowania zmiany progu lub zwiększając go) kontur obrazu okazuje się zbyt szczegółowy do narysowania naszym sposobem (zbyt długie obliczenia).



Inny efekt otrzymujemy stosując przed konturowaniem w niewielkim stopniu operację otwarcia. Nie musimy martwić się w tym przypadku o zbyt dużą ilość pikseli.



Przykłady obrazów, dla których dobry efekt daje tylko binaryzacja i konturowanie, przedstawiono poniżej. Na ostatnim zdjęciu można jednak zauważyć braki w konturze czy podwójną linię konturu.



5. Co się nie udało, co nie działa

Z analizy testowych obrazów widzimy, że ograniczeniem aplikacji jest rysowanie zbyt dużej ilości punktów. Przy próbie narysowania obrazu zbyt szczegółowego czas obliczeń staje się zbyt długi, czego przykładem jest obraz przedstawiający Fouriera.

Problematyczne jest również znalezienie wyraźnego lub dokładnego konturu dla pewnych obrazów. W niektórych przypadkach linia konturu jest podwójna, czy wręcz 'rozdwójona', co wydłuża czas obliczeń oraz rysowania.

6. Instrukcja użytkownika

Użytkownik uruchamia plik Draw_With_FT.mlapp, a następnie za pomocą przycisku wgrywa własny obraz lub jeden z testowych. Potem może użyć automatycznej

binaryzacji lub przy użyciu suwaków w sekcji edycji obrazu doprowadzić do oczekiwanego efektu, który cały czas można obserwować. Po stworzeniu konturów, użytkownik przechodzi do panelu animacji. Wybiera sposób rysowania obrazu, a także opcjonalnie margines i opóźnienie rysowania. Efekt można także zapisać do pliku (domyślnie zawsze ta sama nazwa). Po kliknięciu 'Oblicz i rysuj' obliczane są dane potrzebne do stworzenia animacji, która następnie zostaje wyświetlona.

7. Przykładowe dane

Przykładowe obrazy załączone są w folderze 'Obrazki'.