Łukasz Szczygielski  
Jakub Turliński

Sprawozdanie z projektu  
Steganografia - ukrywanie obrazu w spektrogramie pliku audio  
 zrealizowanego na potrzeby przedmiotu Technika Obrazowa w semestrze 2021Z

**1. Cel projektu**

Celem projektu było stworzenie programu w języku Python, który miałby umieszczać podany przez użytkownika obraz w spektrogramie dźwięku(np. utwór muzyczny). Praktycznym zastosowaniem takiego zabiegu mogłoby być umieszczanie znaków wodnych w plikach audio celem potwierdzenia ich autora.  **2. Użyte technologie**

Poza językiem python w projekcie został użyte następujące technologię:  
**Qt** – framework odpowiadających za działanie GUI

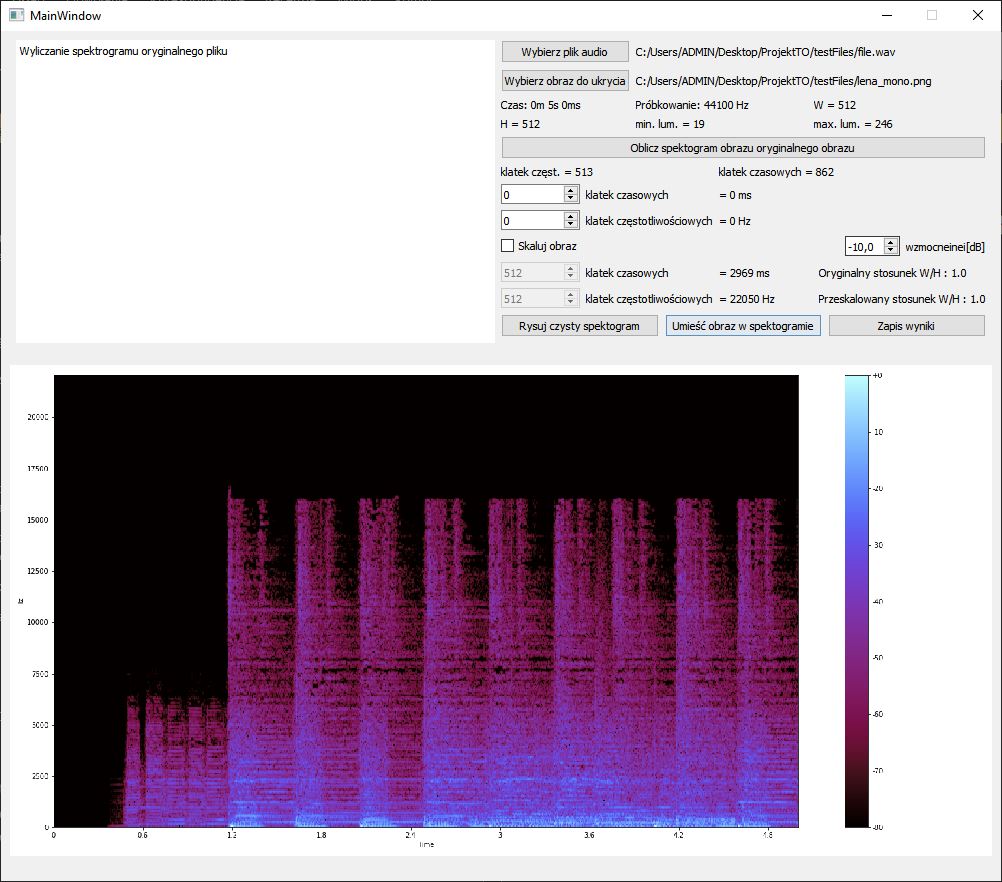
**Librosa** **-** pakiet odpowiadający za konwersje sygnału audio między dziedziną czasu a częstotliwości za pomocą transformaty Fouriera krótkiego czasu.

**OpenCV** -pakiet odpowiadający za odczyt i zapis plików graficznych.

**pyplot** - pakiet odpowiadający za rysowanie spektrogramów

**3. Działanie**

Program składa się z jednego okna w którym użytkownik musi wybrać plik w formacie .wav w którym ma zostać obsadzony obraz w formacie .png. W momencie wybrania obu plików użytkownik może obliczyć spektrogram pliku audio który zostanie wyświetlony w oknie programu. Spektrogram jest rysowany w skali decybelowej. Działania programów są raportowane w oknie logów. Użytkownik może wybrać pozycje(lewy dolny róg) w której zostanie umieszczony obraz oraz może przeskalować obraz wedle uznania. Jednostkami którymi posługuje się użytkownik są klatki czasu(oś pozioma) oraz klatki częstotliwości(oś pionowa). Wprowadzenie niepoprawnych wartości(takich które oznaczałby wyjście obrazu poza obszar spektrogramu zostanie zakomunikowane. Wciśniecie przycisku „Umieść obraz w spektrogramie” spowoduje umieszczenie obrazu wewnątrz spektrogramu poprzez mapowanie jaskrawości pikseli obrazu na moc sygnału audio w odpowiednich jego próbach, użytkownikowi zostanie zaprezentowane teoretyczny(bez zniekształceń) wygląd spektrogramu. Zmodyfikowany obraz jest tylko tymczasowy i użytkownik może dopasować pozycje i rozmiar obrazu wedle uznania. Wciśniecie przycisku „Zapisz wyniki” spowoduje zapisanie w wybranym przez użytkownika miejscu następujących plików:  
1) plik audio wzbogacony o obraz w pliku .wav  
2) plik .ini zawierający informacje potrzebne do ponownego odtworzenia obrazu z pliku audio.  
3) plik .png z spektrogramem pliku dźwiękowego bez modyfikacji  
4) plik .png z teoretycznym spektrogramem zmodyfikowanego pliku dźwiękowego   
5) plik .png z rzeczywistym spektrogramem pliku zmodyfikowanego dźwiękowego   
6) plik .png z obrazem odczytanym z pliku audio  
7) plik .png z obrazem z obrazem przed wprowadzeniem go do sygnału audio  
Po zapisie plików program wczyta spektrogram zmodyfikowanego audio oraz wyliczy miary błędów między obrazem oryginalnym a tym wydobytym z pliku audio.

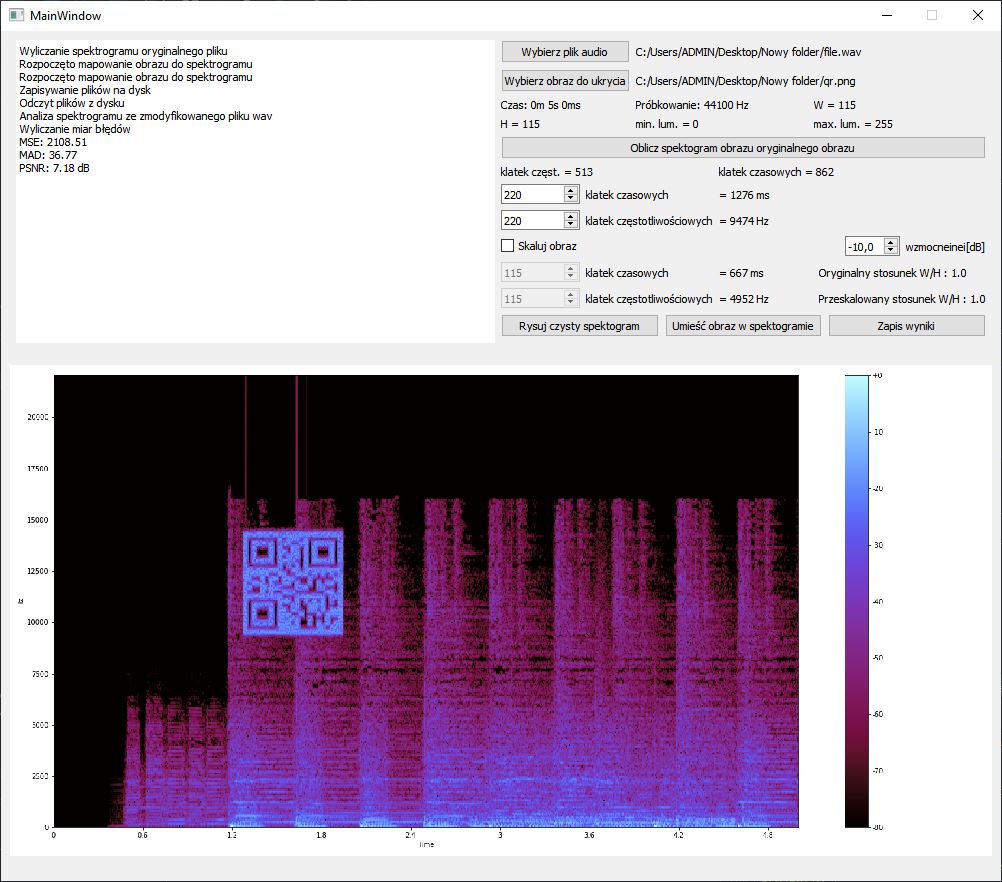


*Główne okno programu, przed umieszczeniem obrazu*

**

*Obraz zawierający osoba, kobieta

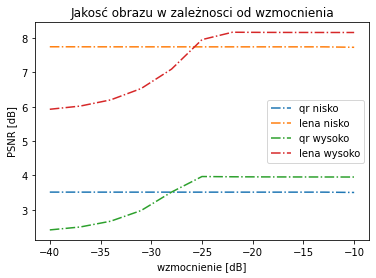
Opis wygenerowany automatycznie  
Porównanie od lewej: Obraz oryginalny, obraz wstawiany do spektrogramu, obraz odzyskany ze zmodyfikowanego pliku audio(wzmocnienie = -10dB).*

**

*Główne okno programu, po umieszczeniem obrazu*

**4.Szczegóły techniczne**

Oś pozioma jest dzielona z rozdzielczością 5,8ms na klatkę, jest to wartość standardowa dla próbkowania sygnału audio 44,1kHz. Oś pionowa zostaje podzielony na 513 klatek częstotliwości, maksymalna częstotliwość wynosi 22,1kHz co daje rozdzielczość 43,08Hz na klatkę, te wartości zostały wybrane metodą prób i błędów a wynikają z doboru parametrów transformaty Fouriera takich jak długość okna który wynosi 1024 próbki i długość przeskoku która wynosi 256 próbek. Do próbkowania sygnału zostało użyte okno typu „Nuttall” zapewniające dobrą filtrację sygnału kosztem nieznacznie wydłużonego czasu obliczeń. Użytkownik dodatkowo może wybrać poziom wzmocnienia mocy obrazu w zakresie od -40dB do -10dB, zmniejszenie mocy sygnału będzie skutkowało zmniejszeniem wpływu mocy sygnału na pozostałe próbki spektrogramu na danej częstotliwości, a tym samym będzie zmniejszało zniekształcenia wprowadzone do pliku audio przez obsadzenie obrazu, jednakże spowoduje to silne zmniejszenie różnicy między kolejnymi poziomami jaskrawości osadzanego obrazu.

**5. Jakość obrazów odczytanych ze zmodyfikowanego dźwięku   
  
**

„nisko” oznacza że dolna granica obrazka jest na częstotliwości 215Hz i jest otoczony silnym sygnałem  
„wysoko” oznacza że dolna granica obrazka jest na częstotliwości 15kHz i jest otaczający sygnał jest dużo słabszym

Z wykresu powyżej można zauważyć że:  
1) Obrazy binarne mają większą odporność jakościową na mniejsze poziomy, dla tłumienia o wartości 40dB QR traci zaledwie 1,5db PSNRu względem PSNR maksymalnego. Obraz Lena traci 2,3dB(około 50% więcej)

2) W otoczeniu silnego sygnału audio wzmocnienie obrazu nie ma znaczenia, artefakty sygnału audio zbyt mocno wpływają na obraz żeby jego własne tłumienie miało jakość, zatem użytkownikowi który chce umieścić obraz na niskich częstotliwościach zaleca się zastosować duże tłumienie gdyż bez szczególnej utraty jakości może lepiej zamaskować obraz.

3) Dla obu typów obrazu umieszczonych na wysokich częstotliwościach można zaobserwować takie wartości tłumień w których są one neutralne dla jakości. Użytkownikowi który chce umieścić obraz na wysokich częstotliwościach zaleca się użycie średniego tłumienia które da najlepszą jakością przy zauważalnym spadku odróżnialności sygnału zmodyfikowanego od oryginalnego.

***Progowanie obrazu Lena, górny rząd reprezentuje obraz odczytany z audio a dolny progowanie tego obrazu do określonej liczby bitów, kolejno od lewej od 2 do 8 bitów***

**Obraz zawierający czarny, roślina

Opis wygenerowany automatycznie**

Obraz można też poddać progowaniu, w tym wypadku obraz Lena zdecydowanie zyskuje na wrażeniach wizualnych

***Kolejne przekształcenia obrazu QR***

**Obraz zawierający tekst

Opis wygenerowany automatycznie**Progowanie można też użyć w bardziej praktycznym celu. Powyższy obraz przedstawia wygląd kodu QR który został zamaskowany w audio a następnie poddany próbie odtworzenia. Uważny obserwator może zauważyć że piksele kodu oryginalnego nie są identyczne z pikselami kodu odtworzonego, jednakże maskując kodu QR które zostały wygenerowane z nadmiarowością można odczytać informacje z odzyskanego kodu pomimo kliku przekłamanych pikseli. Daje to dość możliwość zapisywania dużej ilości informacji w niewielkim fragmencie audio.   
 **6. Materiały**Do testów posłużyły  
1) obraz Lena z laboratorium tego przedmiotu  
2) nadmiarowy kod QR  
3) 5 pierwszych sekund piosenki Kultu „Panie Waldku”