

Sprawozdanie z projektu:

„Steganografia - ukrywanie obrazu w spektrogramie pliku audio”

zrealizowanego na potrzeby przedmiotu Technika Obrazowa

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 technologie:

- **Qt** – framework odpowiadających za działanie GUI
- **Librosa** - pakiet odpowiadający za konwersje sygnału audio między dziedziną czasu i 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 wybiera plik w formacie *.wav*. W pliku tym zostanie osadzony obraz w formacie *.png*. W momencie wyboru 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.

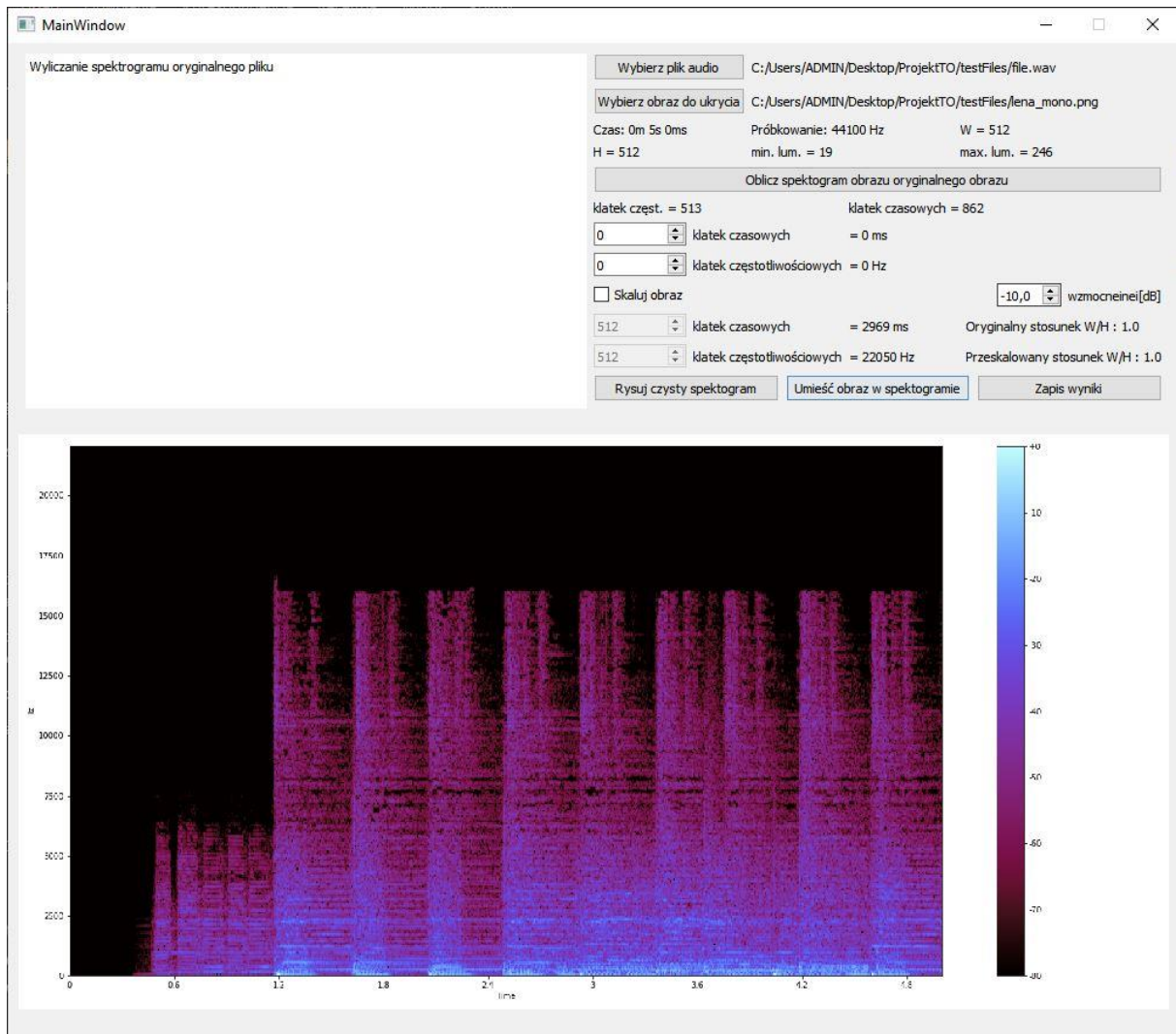
Użytkownik może wybrać pozycje (lewy dolny róg) w której zostanie umieszczony obraz oraz może przeskalować obraz wedle uznania. Jednostkami którymi posługuje się użytkownik są klatki czasu (oś pozioma) oraz klatki częstotliwości (oś pionowa). Wprowadzenie wartości „niepoprawnych”, czyli takich które oznaczałyby wyjście obrazu poza obszar spektrogramu spowoduje wyłapanie wyjątku.

Wciśnięcie 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 poglądowy (teoretyczny i bez zniekształceń) wygląd spektrogramu. Zmodyfikowany obraz jest tylko tymczasowy i użytkownik może dopasować pozycje i rozmiar obrazu wedle uznania. Wciśnięcie przycisku „Zapisz wyniki” spowoduje zapisanie w wybranym przez użytkownika miejscu następujących plików:

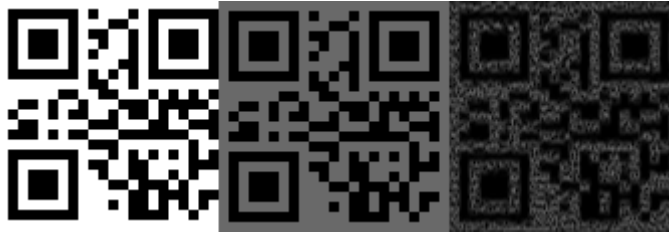
- 1) plik audio wzbogacony o obraz w pliku *.wav*
- 2) plik *.ini* zawierający informacje potrzebne do ponownego odtworzenia obrazu z pliku audio.
- 3) plik *.png* z spektrogramem pliku dźwiękowego bez modyfikacji
- 4) plik *.png* z teoretycznym spektrogramem zmodyfikowanego pliku dźwiękowego
- 5) plik *.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

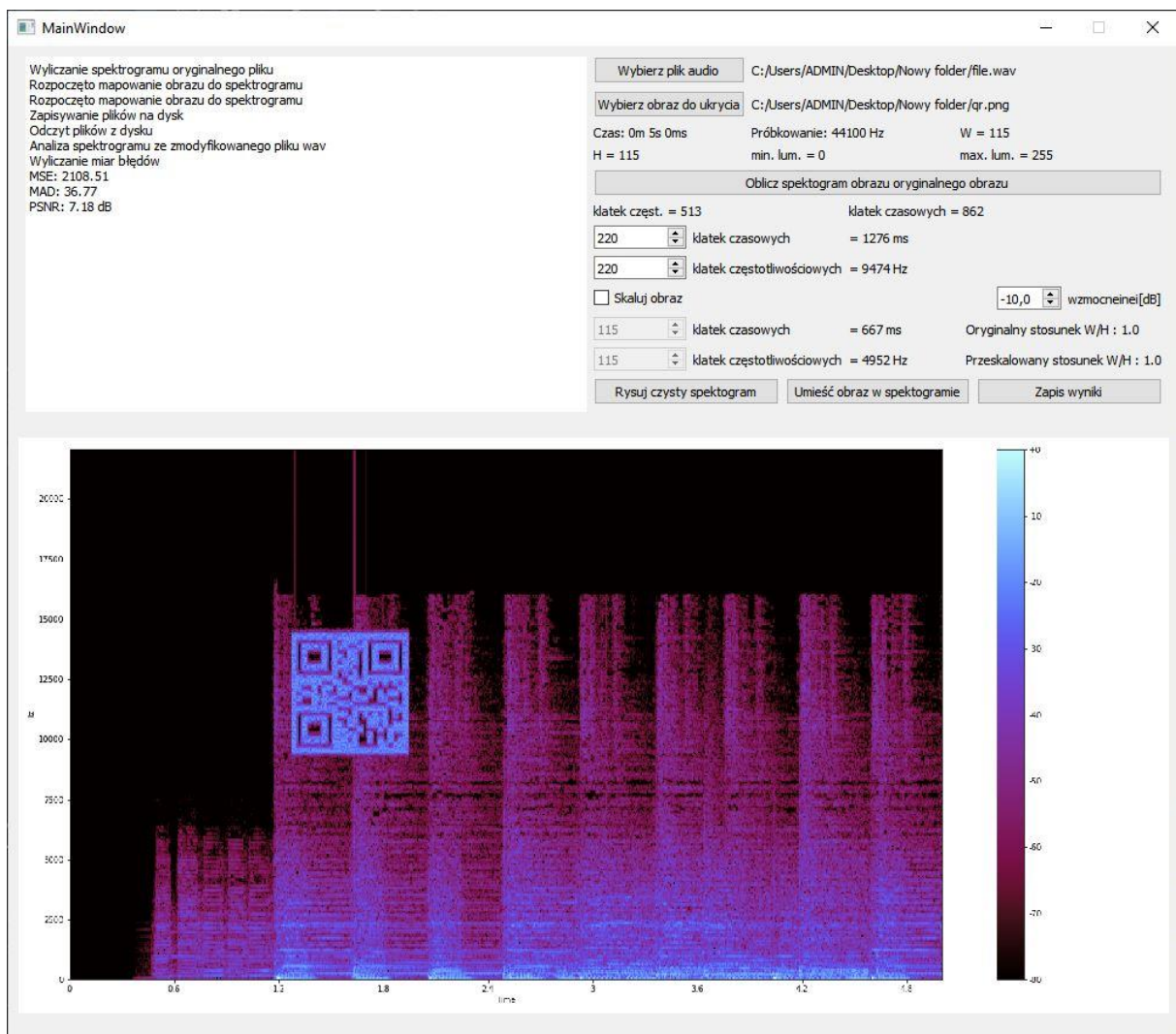
Po zapisie plików program wczyta spektrogram zmodyfikowanego audio oraz wyliczy miary błędów między obrazem oryginalnym a tym wydobytym z pliku audio.



Główne okno programu, przed umieszczeniem obrazu



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



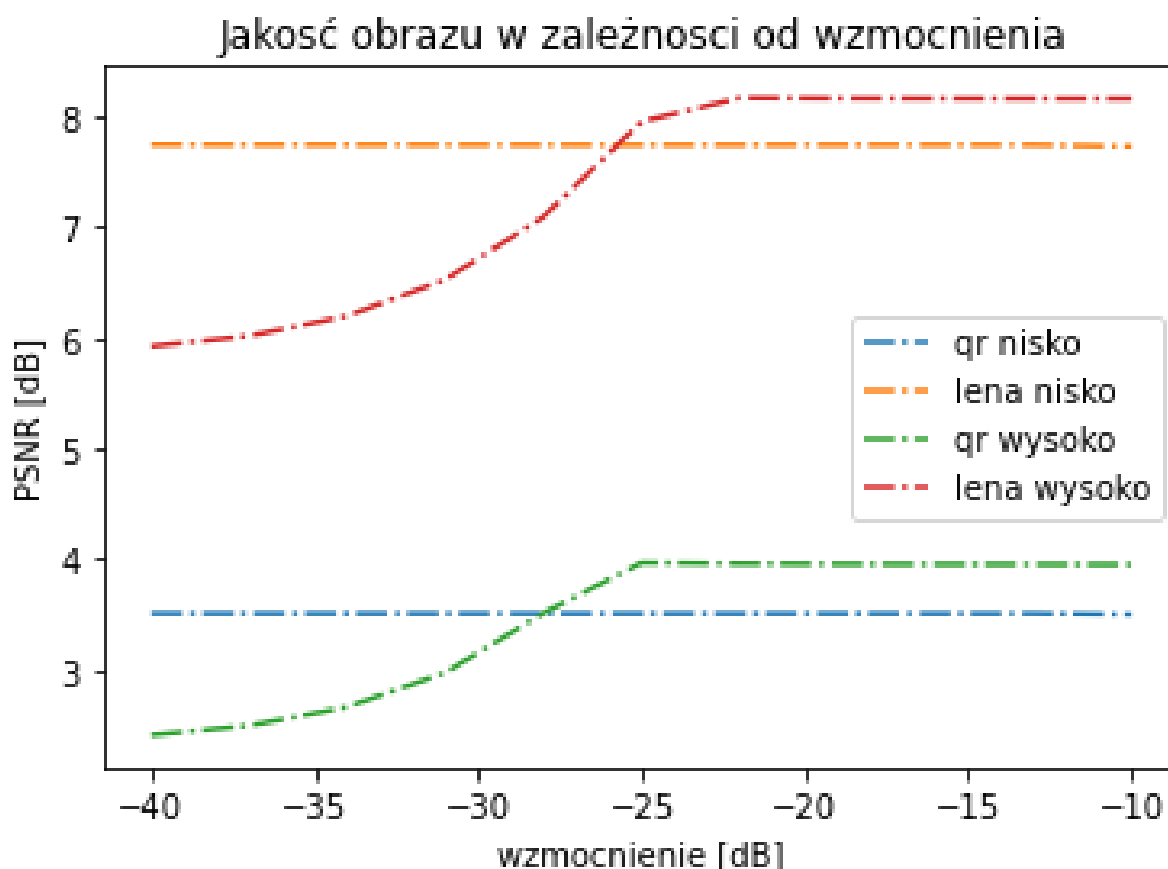
Główne okno programu, po umieszczeniu obrazu

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. Owe wartości zostały wybrane metodą prób i błędów i wynikają z doboru parametrów transformaty Fouriera takich jak długość okna, którego długość 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 może dodatkowo 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. Jednocześnie zmiana wzmocnienia mocy spowoduje to wyraźne zmniejszenie obserwowanego kontrastu w osadzonym obrazie, ze względu na wykorzystywany mniejszy zakres poziomów jasności przez obraz osadzony.

5. Jakość obrazów odczytanych ze zmodyfikowanego dźwięku

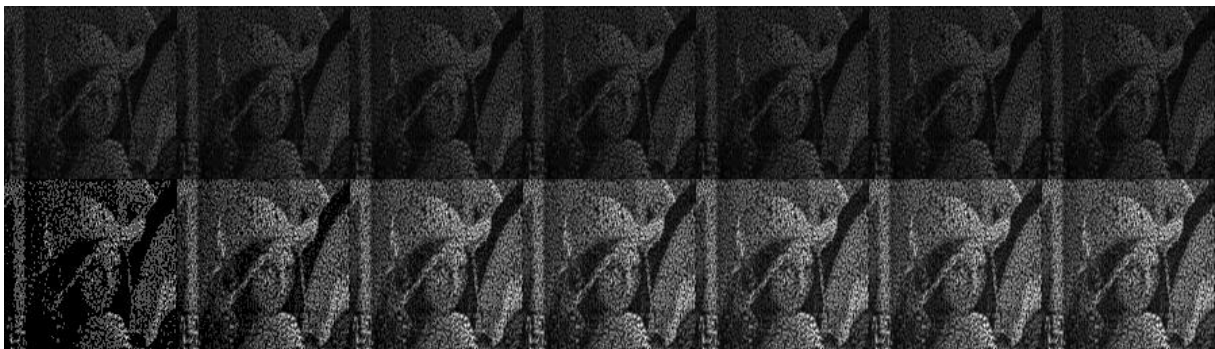


„nisko” oznacza, że dolna granica obrazka jest na częstotliwości 215Hz i jest otoczony silnym sygnałem

„wysoko” oznacza, że dolna granica obrazka jest na częstotliwości 15kHz i jest otoczony sygnałem zauważalnie słabszym

Wnioski z powyższego wykresu:

- 1) Obrazy binarne mają większą odporność jakościową na mniejsze poziomy wzmacnienie. Dla tłumienia o wartości 40dB wartość PSNR kodu QR spada o 1,5 dB względem PSNR maksymalnego. Dla obrazu Lena wartość PSNR spada o 2,3dB (około 50% więcej)
- 2) W otoczeniu silnego sygnału audio wzmacnienie obrazu nie ma znaczenia, artefakty sygnału audio zbyt mocno wpływają na obraz, aby tłumienie pozwoliło zachować jakość obrazu. Użytkownik, który chce umieścić obraz na niskich częstotliwościach powinien 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 tłumienia, które nie wpływają na jakość obrazu. Użytkownikowi, który chce umieścić obraz na wysokich częstotliwościach zaleca się użycie średniego tłumienia. Efektem czego będzie możliwie najlepszą jakość obrazu 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 można też poddać progowaniu, w tym wypadku obraz Lena zdecydowanie „zyskuje na wrażeniach wizualnych” zachowując niektóre szczegóły obrazu oryginalnego.



Kolejne przekształcenia obrazu QR

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, który został wygenerowany z nadmiarowością można odczytać informacje z odtyskanego kodu pomimo kilku przekłamanych pikseli. Umożliwia to zapisywanie większej ilości informacji w niewielkim fragmencie audio.

6. Użyte obrazy i pliki audio

- obraz Lena z laboratorium tego przedmiotu
- nadmiarowy kod QR
- 5 pierwszych sekund piosenki Kultu „Panie Waldku”