

Ćwiczenie PULS

Detekcja pulsu - wideopletyzmografia

Cele laboratorium:

- Zapoznanie się z metodami wideopletyzmografii.

Przygotowanie

W przypadku realizacji ćwiczenia w laboratorium: w katalogu wskazanym przez prowadzącego utwórz podkatalog, w którym będą zapisywane wszystkie dane podczas ćwiczenia. Nazwa katalogu powinna zawierać: **<nr ćwiczenia>_<data>_<inicjały osoby realizującej ćwiczenie>**; zapisz i rozpakuj pliki ćwiczenia (pobrane z UPEL) do utworzonego katalogu; po skończonych zajęciach pamiętaj o usunięciu danych z dysku.

W przypadku realizacji ćwiczenia zdalnie:

- wymagania: MATLAB R2023b z modułem: Image Acquisition Toolbox, Image Processing Toolbox, Audio Toolbox, DSP System Toolbox, kamera internetowa, mikrofon, następujące AddOns: MATLAB Support Package for USB Webcams, Image Acquisition Toolbox Support Package for OS Generic Video Interface
- Opcjonalnie – urządzenie do pomiaru pulsu (pulsoksymetr, smartwatch)

Informacje na temat sprawozdania z ćwiczenia zostały zamieszczone na końcu instrukcji

Proszę o zgłaszanie nieścisłości w instrukcji drogą emailową do prowadzącego zajęcia.

Część I – analiza sygnału VPG

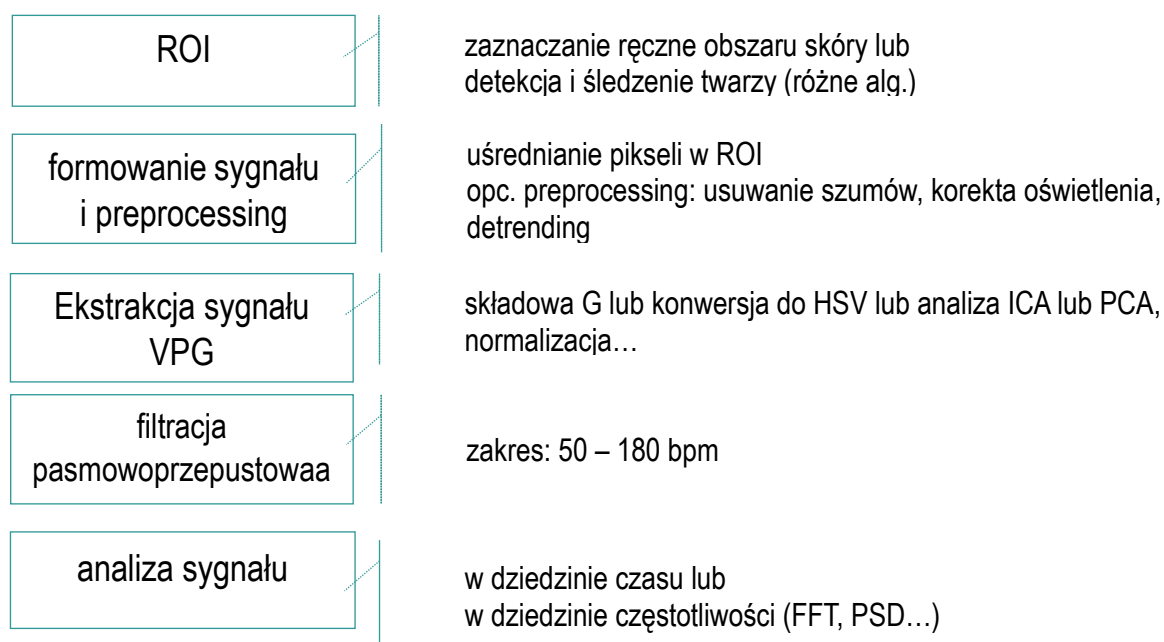
Gdzie szukać dodatkowych informacji:

- Internet (google scholar) – słowa kluczowe: videoplethysmography
- Algorithmic Principles of Remote PPG: <https://ieeexplore.ieee.org/document/7565547>
- [1] Pulse rate estimation using imaging photoplethysmography: generic framework and comparison of methods on a publicly available dataset: <https://arxiv.org/abs/1710.08369>
- Continuous Distant Measurement of the User's Heart Rate in Human-Computer Interaction Applications : <https://www.mdpi.com/1424-8220/19/19/4205>
- [2] de Haan G, Jeanne V (2013). Robust pulse rate from chrominance-based rPPG, IEEE Transactions on Biomedical Engineering 60(10): 2878-2886.
http://www.es.ele.tue.nl/~dehaan/pdf/169_ChrominanceBasedPPG.pdf

Informacje ogólne:

Fotopletyzmografia jest badaniem, które ocenia miejscowe zmiany zawartości krwi w tkankach. Do skóry badanego przykładana się czujnik wyposażony w źródło emitujące światło w zakresie podczerwieni oraz system sensorów optycznych. Fizyczną podstawę tej metody stanowi zjawisko odbijania i rozpraszania strumienia fotonów, jakie zachodzi w trakcie przejścia promieniowania przez tkanki organizmu. Do tkanek tłumiących możemy zaliczyć m.in: hemoglobinę krwi tętniczej oraz żyłnej, kości czy melaninę występującą w skórze. W pomiarach wykorzystywany jest efekt zmiennego pochłaniania fal świetlnych, które zależy od aktualnego wypełnienia łożyska naczyńowego w badanym obszarze.

Wideopletyzmografia wykorzystuje kamerę jako sensor, natomiast źródłem światła jest naturalne oświetlenie otoczenia. Dzięki odpowiednim algorytmom możliwe jest sformowanie sygnału zawierającego podobne informacje jak sygnał pletyzmograficzny. Rysunek 1 przedstawia ogólny schemat analizy:

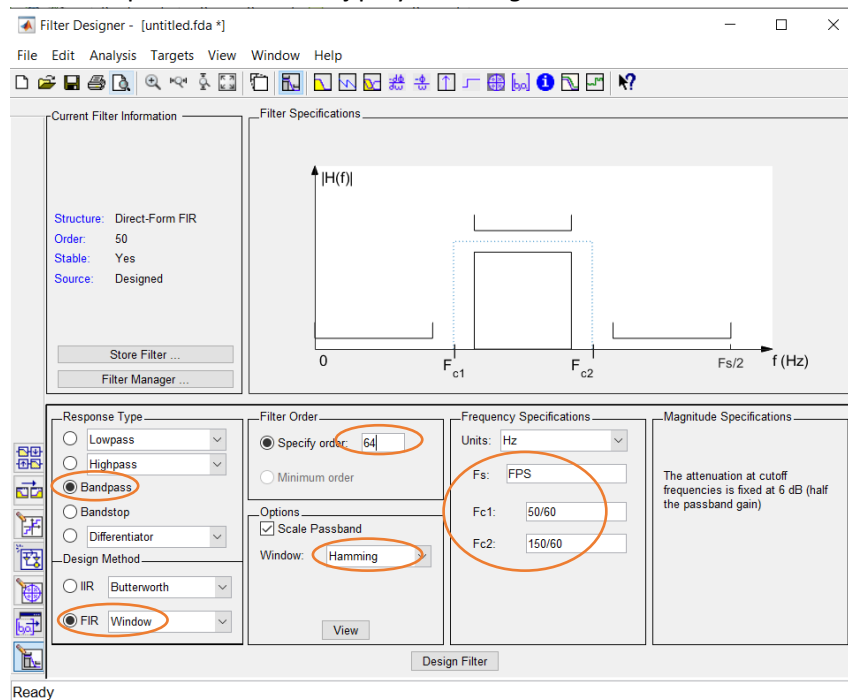


Rysunek 1 Ogólny schemat analizy

Instrukcja:

1) Projektowanie filtru cyfrowego

- Uruchom narzędzie `filterDesigner`
- Ustaw parametry generowanego filtru:
 - filtr pasmowo-przepustowy (*bandpass*)
 - metoda projektowania: *FIR->Window*
 - rząd filtru: *order=64*
 - typ okna: *Hamming*
 - częstotliwość próbkowania: $F_s = FPS$ (w tym przypadku 60)
 - częstotliwości odcięcia: $F_{c1} = 50/60$, $F_{c2} = 150/60$ (zwróć uwagę że częstotliwości te są w Hz i wyznaczamy je dzieląc min/max spodziewaną wartość pulsu przez 60 ponieważ puls podawany jest w uderzeniach na minutę – nie należy mylić tego z częstotliwością próbkowania w tym przypadku)
- Po ustawieniu parametrów kliknij przycisk *Design Filter*



- W sprawozdaniu zamieść wykres charakterystyki filtru (*magnitude response*).
- Zapisz sesję narzędzia do pliku `my_bandpass_filter.fda`
- Wygeneruj funkcję o nazwie `my_bandpass_filter.m` z tak zaprojektowanego filtru:
File->Generate MATLAB Code->Filter Design function
- Otwórz wygenerowany plik i zapoznaj się z nim (w dokumentacji sprawdź funkcje `fir1` i `hamming`).

2) Otwórz skrypt `VPG_przyklad_1.m` i wykonuj kolejne sekcje kodu notując w sprawozdaniu rezultaty oraz wnioski (patrz komentarze w skrypcie).

- (1) Wczytanie danych: zwróć uwagę na strukturę danych

- (2) Wizualizacja sygnału VPG: powiększ fragment sygnału i zwróć uwagę czy jest widoczna jego okresowość.
- (3) Wybór fragmentu sygnału: do analizy wybieramy fragment sygnału o określonej długości (większej lub równej niż długość FFT). Zamieść wykres w sprawozdaniu.
- (4) Preprocessing sygnału VPG - usuwanie trendu. Zamieść wykres w sprawozdaniu.
- (5) Odczytanie wartości pulsu z wykresu – uzupełnij odpowiednie fragmenty kodu
- (6) Preprocessing sygnału VPG - filtracja pasmowoprzepustowa (wykorzystaj wcześniej zaprojektowany filtr cyfrowy)
- (7) Analiza sygnału VPG - transformata Fouriera sygnału
- (8) Analiza sygnału VPG - wyznaczenie wartości pulsu

Część II – testowanie algorytmu na własnych danych

W tej części celem będzie zebranie własnych danych oraz detekcja pulsu. W tym celu potrzebna jest kamera oraz jakikolwiek sposób pomiaru referencyjnego pulsu (pulsometr, pulsoksymetr, smartwatch). Jeżeli nie dysponujesz urządzeniem do pomiaru pulsu możesz zmierzyć tętno „ręcznie”.

Położ palec wskazujący i środkowy na wewnętrznej stronie ręki, nad nadgarstkiem. Przebiega tam tuż pod skórą tętnica promieniowa. Uciśnij to miejsce. Spójrz na zegarek i licz puls przez 15 sekund. Gdy pomnożysz tę liczbę przez cztery, otrzymasz wartość tętna.



Autor: Getty Images

Puls możesz także zmierzyć na tętnicy szyjnej (poszukaj miejsca w bok od krtani, w którym poczujesz pulsującą krew). Łatwiej na niej wyczuć tętno.

- 1) Otwórz skrypt `testHarness_PULS.m` i ustaw parametry akwizycji video tak aby szybkość akwizycji (FPS) była stabilna. Zapoznaj się z poszczególnymi funkcjami:
 - `myAlgorithmPULS.m` - detekcja i śledzenie twarzy, wyznaczanie sumy pikseli dla poszczególnych składowych (R,G,B) w oknie ROI (śledzona twarz).
 - `myExportDataPULS.m` - eksport sumy pikseli oraz czasu przetwarzania ramek do pliku TXT (`'dane_puls.txt'`)
 - `myVisualizationPULS.m` - wizualizacja sumy pikseli na wykresie
- 2) Ustaw się nieruchomo przed kamerą, tak aby twarz była widoczna i poprawnie wykrywana. Uruchom skrypt `testHarness_PULS.m` tak aby zebrać **minimum 1024 ramki** video. Zanotuj w sprawozdaniu czas akwizycji, wyznacz średnie FPS oraz zanotuj referencyjną wartość pulsu w trakcie akwizycji (średnia). Upewnij się, czy twarz jest poprawnie wykrywana na kolejnych ramkach. Dane sumy pikseli oraz czasu przetwarzania ramek zapisywane są do pliku tekstowego `'dane_puls.txt'` który zawiera następujące kolumny danych: `iter`, `t2`, `sumr`, `sumg`, `sumb`
- 3) Zmień nazwę pliku tekstowego na `'dane_puls1.txt'`.
- 4) Otwórz skrypt `VPG_przyklad_2.m` i wykonuj kolejne sekcje kodu notując w sprawozdaniu rezultaty oraz wnioski (patrz komentarze w skrypcie).

- (1) Wczytanie danych: zwróć uwagę na średni FPS oraz na to czy na wszystkich ramkach wykryto twarz
- (2) Wizualizacja sygnału VPG
- (3) Wybór fragmentu sygnału: do analizy wybierz stabilny fragment sygnału o długości 1024 próbki (większej lub równej niż długość FFT). Czy widoczna jest okresowość na sygnale? W sprawozdaniu zamieść wykres i parametry.
- (4) Preprocessing sygnału VPG - usuwanie trendu. W sprawozdaniu zamieść wykres i parametry.
- (5) Preprocessing sygnału VPG - filtracja pasmowoprzepustowa . Przy użyciu narzędzia `"filterDesigner"` zaprojektuj filtr pasmowoprzepustowy i wygeneruj z niego funkcję `"my_bandpass_filter_2.m"`
UWAGA - ustaw dla filtru odpowiednią częstotliwość próbkowania FPS !!!
Czy po filtracji widoczna jest okresowość na sygnale? W sprawozdaniu zamieść wykresy oraz odpowiedź filtru.
- (6) Analiza sygnału VPG - transformata Fouriera sygnału. Zanotuj w sprawozdaniu na której składowej występuje wyraźny "peak" dla częstotliwości zbliżonej do referencyjnej wartości pulsu.
- (7) Analiza sygnału VPG - wyznaczenie wartości pulsu. Zanotuj parametry oraz rezultaty detekcji pulsu. Czy wyznaczona wartość pulsu jest podobna do pulsu zmierzonego pulsometrem (lub ręcznie)? Wyznacz błąd pomiaru pulsu (w jednostkach [bpm]).

Raport z ćwiczenia

Raport z ćwiczenia należy dostarczyć poprzez system UPEL, w formacie **PDF**. Raport składa się z 3 części. Pierwsza część zawiera rezultaty otrzymane w trakcie realizacji ćwiczenia i należy ją umieścić na UPEL pod koniec ćwiczeń. Kolejne części (analiza i wnioski oraz odpowiedzi na pytania), należy umieścić na UPEL w terminie określonym przez prowadzącego zajęcia (najczęściej jest to tydzień czasu – przed następnym ćwiczeniem).

W sprawozdaniu proszę zamieścić:

Data realizacji ćwiczenia i godzina: Imię i nazwisko:
<u>Rezultaty</u> 1) Cz.I: Zamieść rezultaty oraz fragmenty kodu uzupełnione w trakcie ćwiczenia. 2) Cz.II: Zamieść rezultaty oraz fragmenty kodu uzupełnione w trakcie ćwiczenia.
<u>Analiza i wnioski</u> 1) Cz.I: Zamieść wnioski z analizy sygnału VPG 2) Cz.I: Na czym polega metoda MCwS (Mean-Centering-and-Scaling [1][2])? 3) Cz.II: Zastanów się w jaki sposób można byłoby poprawić estymację pulsu poprzez odpowiednie zbieranie danych z okna ROI twarzy (<code>myAlgorithmPULS.m</code>). Czy wyznaczanie sumy wszystkich pikseli jest optymalne? 4) Cz.II: Jaki musi być czas akwizycji video (w sekundach) przy określonym FPS, aby zebrać 1024 próbki? 5) Cz.II: zanotuj wnioski.
<u>Pytania</u> 1) Wymień i krótko uzasadnij 3 potencjalne praktyczne zastosowania algorytmów przedstawionych w ćwiczeniu. 2) Wymień i krótko uzasadnij 2 elementy z ćwiczenia, które zwróciły Twoją uwagę lub są Twoim zdaniem istotne do zrozumienia działania opisanych algorytmów. 3) Zadaj 1 pytanie związane z ww przykładami - co jest dla Ciebie niezrozumiałe lub niejasne.