

Wydział Elektroniki i Technik Informacyjnych  
Politechnika Warszawska

Percepcja Maszyn

Sprawozdanie z ćwiczenia laboratoryjnego nr 3

Mateusz Palczuk

Warszawa, 2022

# Spis treści

1. Wstęp . . . . .	2
2. Odczytywanie sygnału z obrazów . . . . .	3
3. Odszumianie sygnału . . . . .	4
4. Znajdowanie tętna autokorelacją . . . . .	5
5. Wyniki i wnioski . . . . .	6
5.1. Wyniki . . . . .	6
5.2. Rozdzielczość . . . . .	6
5.3. Wnioski . . . . .	6

# 1. Wstęp

W niniejszym sprawozdaniu opisany został przebieg prac związanych z zadaniem laboratoryjnym nr 3. Zadanie to polegało na wykorzystaniu autokorelacji sygnału do wyznaczenia tętna z filmu.

Do realizacji został wykorzystany jedynie pakiet MATLAB.

## 2. Odczytywanie sygnału z obrazów

Poniżej znajduje się kod odpowiedzialny za odczytywanie sygnału z obrazów:

```
% Liczba ramek do wczytania (przy 10 sekundach i 30 FPS będzie to 300)
N = 321;

% wektor jasności
br = zeros(1, N);

% lista obrazów do analizy
imds = imageDatastore(' ../lab1/movie', 'FileExtension', '.jpg');

% wczytanie pierwszych N obrazów i analiza jasności
for i=1:N
    % wczytujemy obraz i przekształcamy go do skali szarości
    I = rgb2gray(imread(imds.Files{i}));

    % wyznaczamy średnią z całego obrazu
    br(i) = mean(I, 'all');
end

% dla ułatwienia późniejszej analizy od razu można odjąć od sygnału składową stałą
br = br - mean(br);
```

### 3. Odszumianie sygnału

Część programu do odszumiania sygnału wybranym filtrem została umieszczona poniżej:

```
% filtrowanie sygnału
f3 = ones(1,3) / 3;
c3 = conv(br, f3, 'same');

f15 = ones(1,15) / 15;
c15 = conv(br, f15, 'same');

g3 = fspecial('gaussian', [1, 3], 1);
cg3 = conv(br, g3, 'same');

g15 = fspecial('gaussian', [1, 15], 3);
cg15 = conv(br, g15, 'same');

% dodaj filtrowanie
br = cg15;
```

## 4. Znajdowanie tętna autokorelacją

Pierwszym etapem było zastosowanie autokorelacji i odfiltrowanie z niej jedynie ważnych elementów. Następnie znalezione zostały maksima lokalne autokorelacji. To w nich sygnał najbardziej pokrywa się sam ze sobą, ale po przesunięciu. Zatem pierwsze takie maksimum lokalne będzie znajdowało się w miejscu gdzie występuje przesunięcie o jedną fazę.

Zatem w programie znajdującym się poniżej dokonano właśnie tego, znelziono pierwsze maksimum lokalne autokorelacji i wyciągnięto z tego informację o częstotliwości sygnału, czyli tętnie.

```
[r1, lags] = xcorr(br);  
% wycięcie jedynie dodatnich przesunięć  
r1 = r1(lags >= 0);  
lags = lags(lags >= 0);  
  
[pks, loc] = findpeaks(r1, "MinPeakDistance", 10, "MinPeakProminence", 20);  
fs = 30;  
  
% przesunięcie w sekundach  
lag_s = lags(loc(1)) * 1/fs;  
  
% częstotliwość bazowa  
freq = 1/lag_s;  
bmp = freq * 60  
  
figure;  
subplot(2, 1, 1);  
plot(br);  
subplot(2, 1, 2);  
plot(r1);
```

## 5. Wyniki i wnioski

### 5.1. Wyniki

Na rys 5.1 - 5.5 pokazane zostały sygnały zastosowane do autokorelacji i wyniki autokorelacji.

W pierwszej kolejności przeprowadzono eksperyment bez żadnego filtra i uzyskano wynik przedstawiony na rys. 5.1. W tym eksperymencie uzyskane tętno wyniosło: 69,2308.

Następnie sygnał oryginalny poddano filtracji filtrem uśredniającym o długości 3 czego wyniki można zobaczyć na rys. 5.2. Zmierzone tętno wyniosło 69,2308.

Kolejnym eksperymentem było poddanie sygnału filtracji filtrem uśredniającym o długości 15. Wyniki tych działań zostały przedstawione na rys. 5.3. Przełożyło się to na tętno 66,6667.

Potem zastosowano filtr Gaussa o długości 3 i uzyskano wyniki jak na rys. 5.4. Tym razem tętno zmierzone wyniosło 69,2308.

Ostatnim eksperymentem było zastosowanie filtra Gaussa, ale tym razem o długości 15. Uzyskane wyniki zostały przedstawione na rys. 5.5, a tętno zostało zmierzone na poziomie 69,2308.

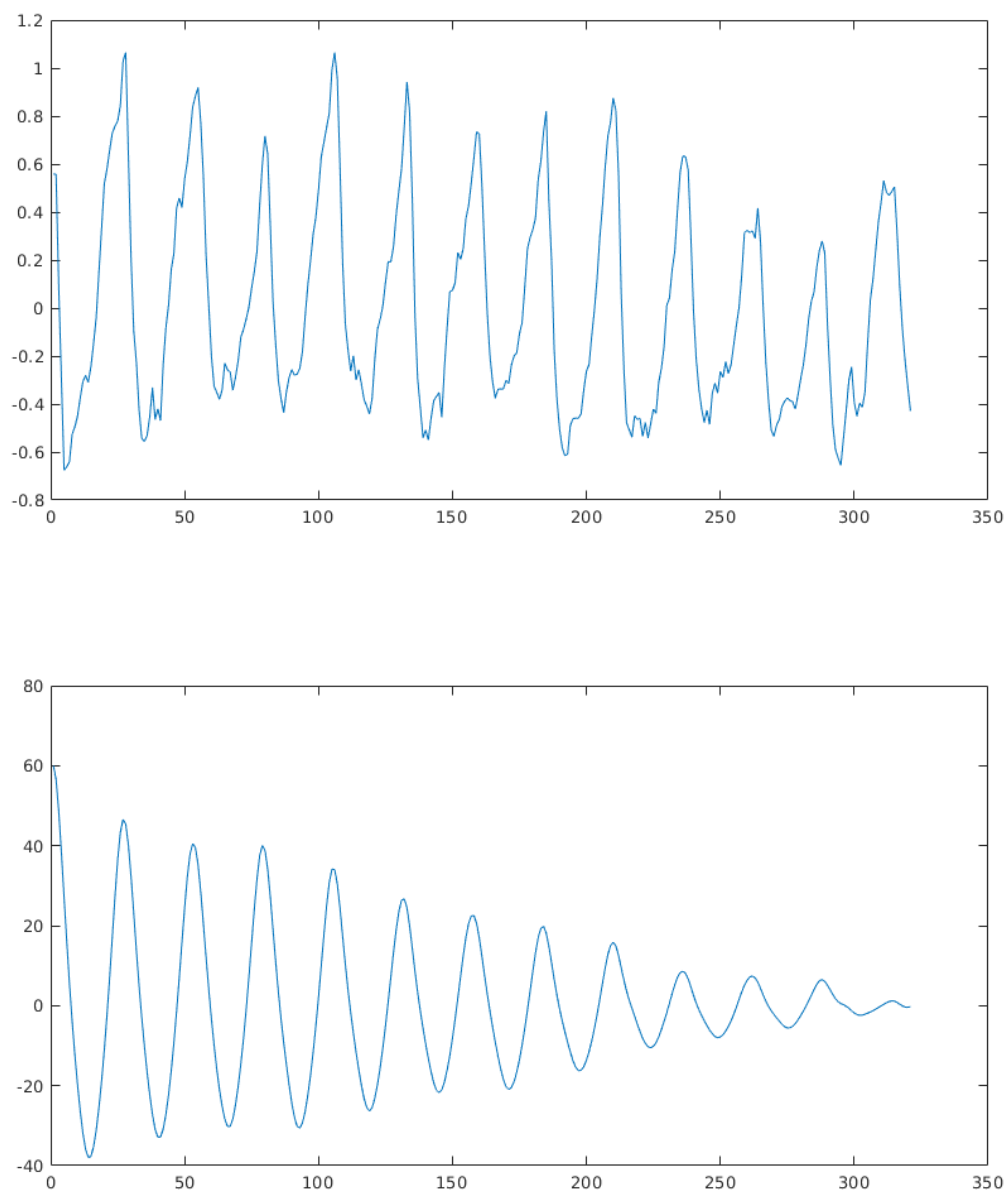
### 5.2. Rozdzielczość

Na dokładność uzyskanych wyników wpływa rozdzielczość samego procesu autokorelacji. Znajdowanie kolejnych wyników autokorelacji odbywa się poprzez przesuwanie kopii sygnału krok po kroku i sprawdzanie podobieństwa z sygnałem oryginalnym. Tak więc rozdzielczość zależy od wielkości kroku procesu autokorelacji. Im ten krok jest mniejszy tym rozdzielczość wyniku będzie większa.

W przypadku tego zadania i jego rozwiązania została zastosowana funkcja `xcorr` pakietu MATLAB. Jeden krok miał długość  $\frac{1}{30} \approx 0,0333$ , co oznacza, że z taką dokładnością została zmierzona częstotliwość w Hz. Na tętno mierzone w uderzeniach na minutę przełożyło się to na dokładność do  $\frac{60}{30} = 2$  bmp.

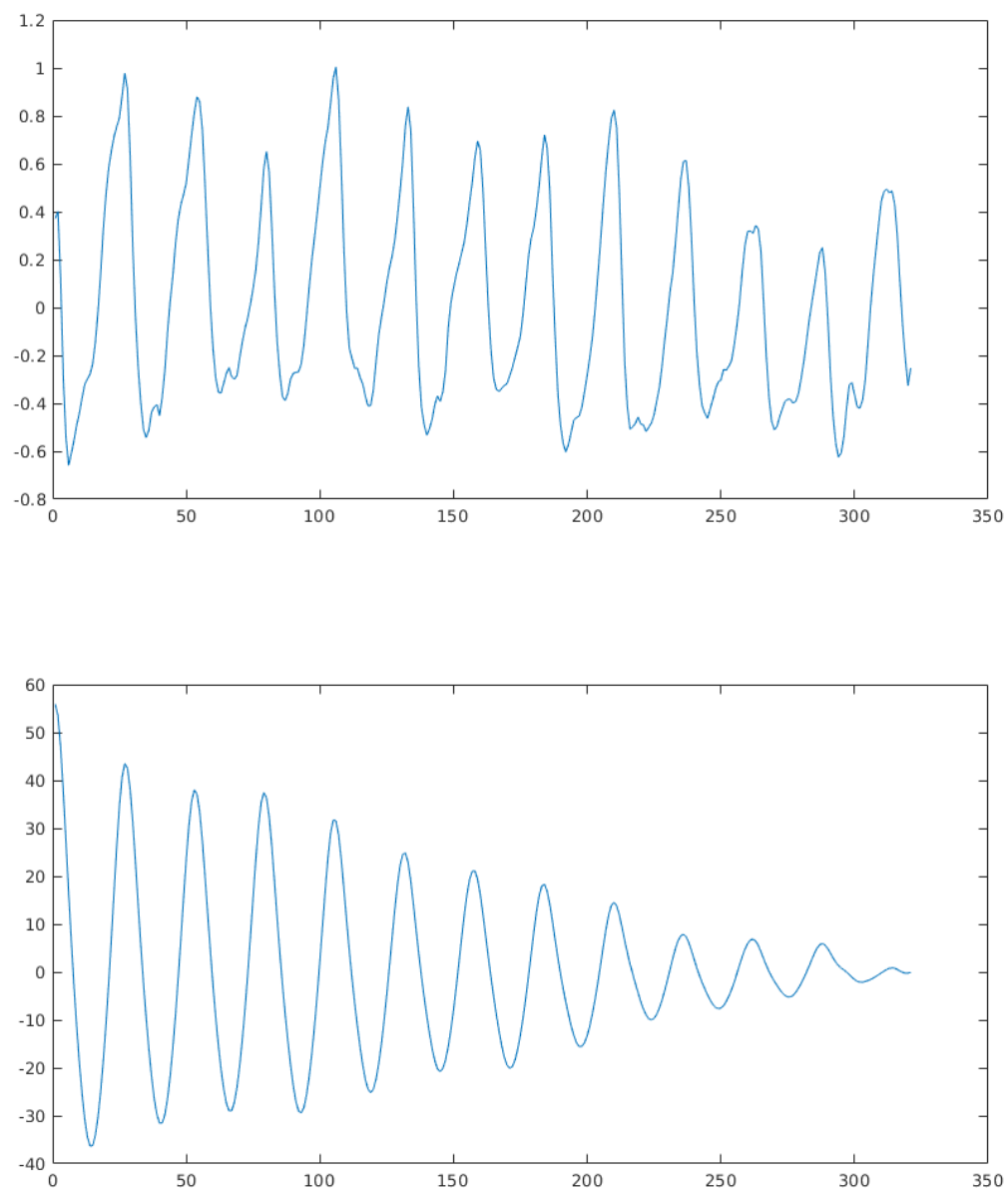
### 5.3. Wnioski

Jak widać wyniki wszystkich tych metod są bardzo podobne, a tak naprawdę, z jednym wyjątkiem, takie same jeżeli chodzi o poziom dokładności zmierzonego tętna. Same przebiegi z kolei są dla krótszych filtrów już zdecydowanie ładniejsze niż dla braku ich braku, ale widać jeszcze pewne niedoskonałości. Mogą one jednak już wynikać z niedokładnie wykonanego pomiaru, a niekoniecznie z samego szumu pomiarowego generowanego przez sprzęt. Za to długi filtr, zarówno uśredniający jak i Gaussa, bardzo mocno przybliżają sygnał do sinusoidy bez mniejszych zniekształceń, ale powoduje to też częściową utratę szczegółowych informacji o mikro-zmianach. Dlatego zresztą też tętno zmierzone z sygnału przepuszczonego przez filtr uśredniający o długości 15 (rys. 5.3) różni się od pozostałych. Sygnał został dostatecznie mocno zniekształcony, że wynik się zmienił.

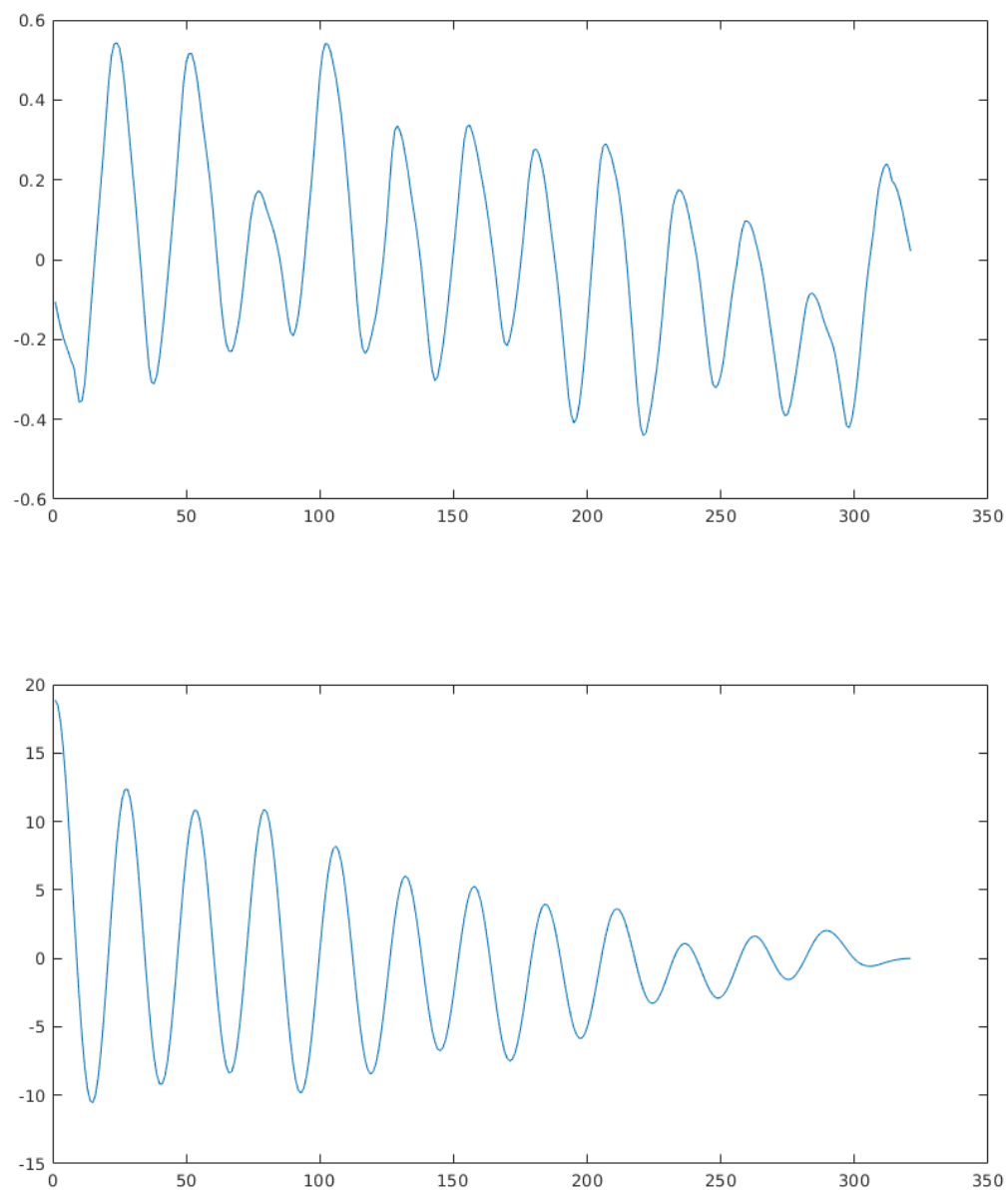


Rys. 5.1. Na górze: sygnał oryginalny uzyskany ze zdjęć  
Na dole: autokorelacja uzyskana z tego sygnału

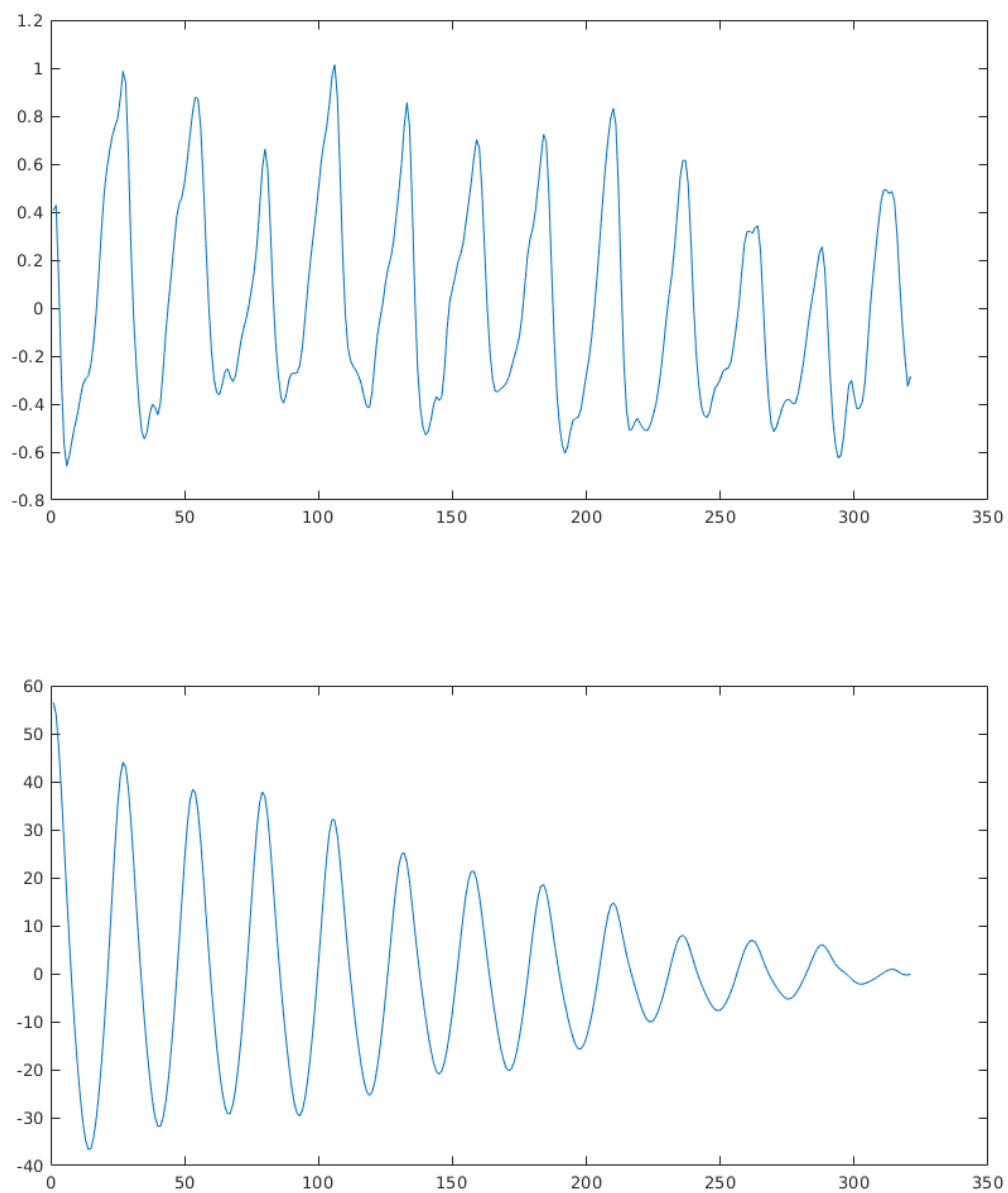




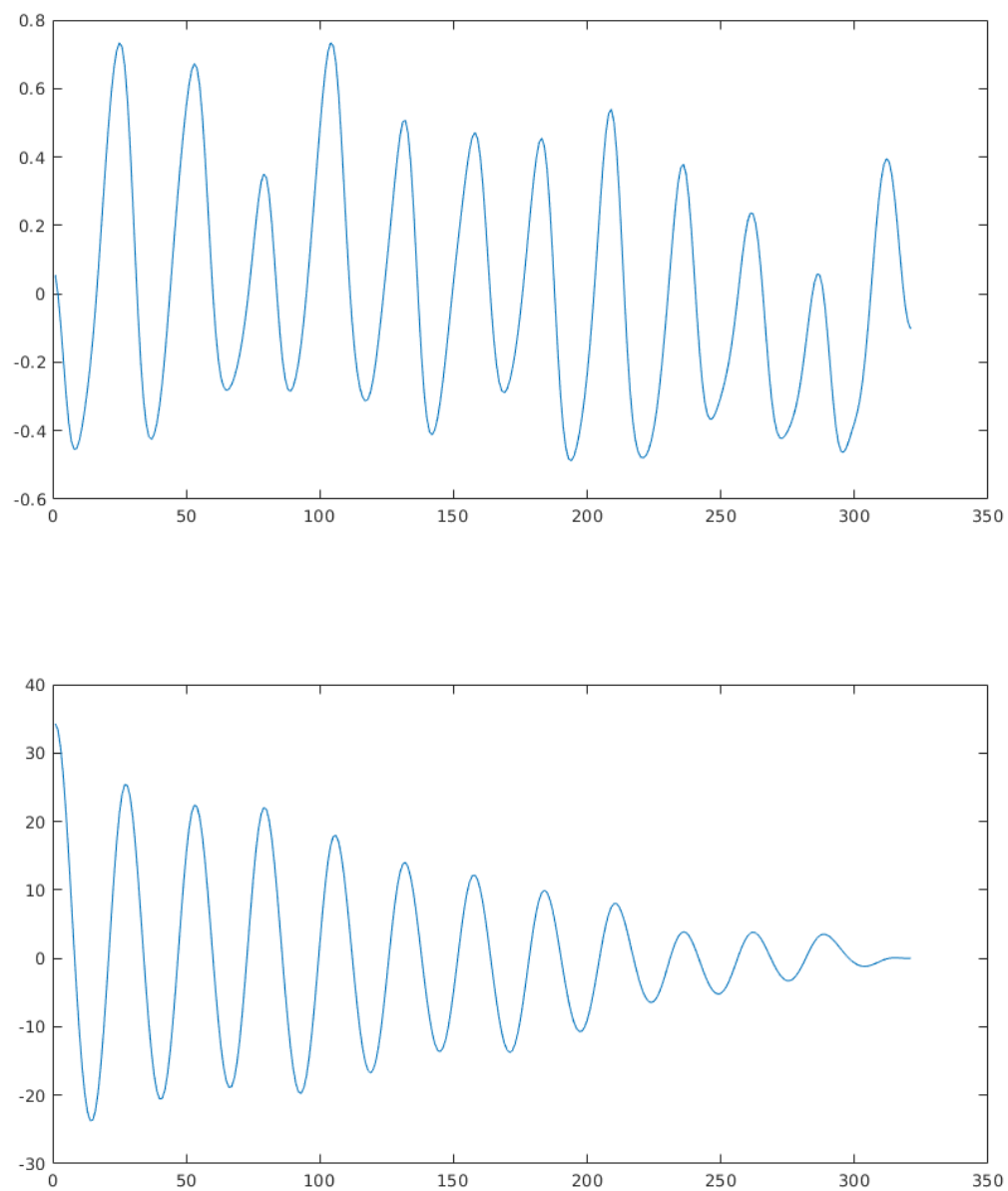
Rys. 5.2. Na górze: sygnał uzyskany po zastosowaniu filtra uśredniającego o długości 3  
Na dole: autokorelacja uzyskana z tego sygnału



Rys. 5.3. Na górze: sygnał uzyskany po zastosowaniu filtra uśredniającego o długości 15  
Na dole: autokorelacja uzyskana z tego sygnału



Rys. 5.4. Na górze: sygnał uzyskany po zastosowaniu filtra Gaussa o długości 3  
Na dole: autokorelacja uzyskana z tego sygnału



Rys. 5.5. Na górze: sygnał uzyskany po zastosowaniu filtra Gaussa o długości 15  
Na dole: autokorelacja uzyskana z tego sygnału