



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ

Информатика и системы управления

КАФЕДРА

Проектирования и технологии производства ЭА

ОТЧЕТ

По семинару №4

по курсу Киберфизические системы

Студент

Фан К.

(Подпись, дата)

(И.О.Фамилия)

Руководитель задания

Леонидов В. В.

(Подпись, дата)

(И.О.Фамилия)

Москва, 2023

Задание 1: Пример корреляционной функции двух произвольных сигналов

```
clear, clc, close all;

fs = 100;
ts = 0:1/fs:2-2/fs;
N =length(ts);

x1 = sin(2*pi*10*ts);
x2 = sin(2*pi*10*ts+pi);

figure;
subplot(2,1,1);
plot(x1), grid on;
subplot(2,1,2);
plot(x2), grid on;

figure;
[r12, lags] = xcorr(x1, x2);
subplot(3,1,1);
plot(lags, r12), grid on;
xlabel('Временной сдвиг, j'), ylabel('r_{12}[j]');
title('Корреляционная функция с краевым эффектом');

% способ убрать влияние краевого эффекта - увеличить длину одного из
% сигналов
x2 = [x2 x2];

r12 =zeros(1,N);
p12 = zeros(1,N);

for j = 1:N-1
    r12(j) = sum(x1.*x2(j:j+N-1));
    p12(j) = r12(j)/(sqrt(sum(x1.^2)*sum(x2(j:j+N-1).^2)));
end

subplot(3,1,2);
plot(r12), grid on;
xlabel('Временной сдвиг, j'), ylim([-100 100]), ylabel('r_{12}[j]');
title('Корреляционная функция');

subplot(3,1,3);
plot(p12), grid on;
xlabel('Временной сдвиг, j'), ylim([-1 1]), ylabel('p_{12}[j]');
title('Нормированная корреляционная функция');
```

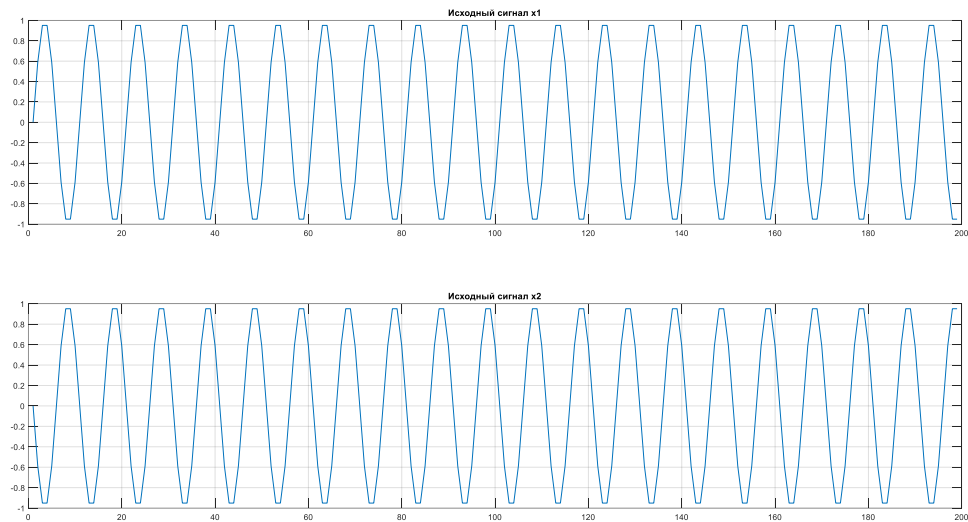


Рисунок 1.1 - Исходные сигналы

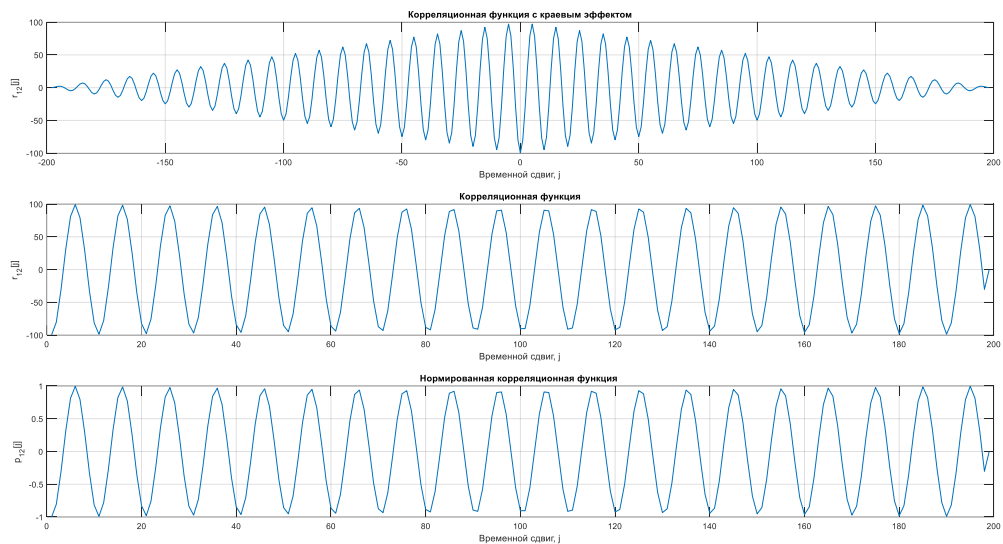


Рисунок 1.2 - Корреляционная функция и нормированная корреляционная функция

Вывод:

В данном задании возьмём два синусоидальных периодических сигнала, сдвинутых по фазе на 180° и посчитаем их корреляцию. Из-за того, что сигналы $x1$ и $x2$ имеют конечную длину, получаем корреляционную функцию с краевым эффектом. Чтобы убрать влияние краевого эффекта увеличиваем длину одного из сигналов (например, $x2$) в два раза. С помощи нормированной корреляционной функции можем оценить степень схожести сигналов. Значению 1 соответствует полная корреляция, нулю соответствует полное отсутствие корреляции и -1 соответствует полная противоположная корреляция.

Задание 2: Автокорреляция шумного сигнала в среде MATLAB

```
clear, clc, close all;

fs = 100;
ts = 0:1/fs:2-1/fs;
N =length(ts);

x = sin(2*pi*10*ts);
x = awgn(x,5); % Добавить белый Гауссов шум
subplot(2,1,1);
plot(x), grid on,title ('Исходный сигнал');

[xc, lags] = xcorr(x);
subplot(2,1,2);
plot (lags, xc), grid on ,ylim([-max(xc) max(xc)]);
title ('Автокорреляционная функция');
```

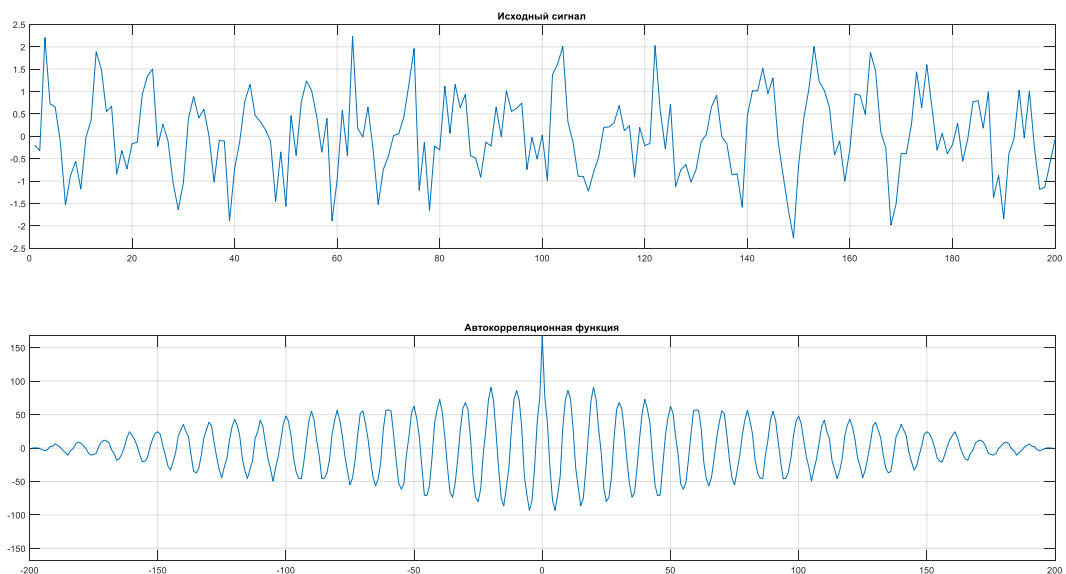


Рисунок 2 - Автокорреляция шумного сигнала

Вывод:

В данном задании возьмём зашумлённый синусоидальный сигнал и построим график автокорреляционной функции этого сигнала. Из графика исходного сигнала сложно сделать вывод о периодичности входного сигнала. С помощью автокорреляционной функции можем обнаружить периодичность входного сигнала.

Задание 3: Автокорреляция случайного сигнала в среде MATLAB

```
clear, clc, close all;

N = 200;
a = -4;
b = 4;

% случайный сигнал
x = (a + (b-a)* rand(1,N));
subplot(2,1,1);
plot(x), grid on, title('Случайный сигнал')
[xc,lags] = xcorr(x,'coeff');
subplot(2,1,2);
plot (lags, xc), grid on;
title ('Автокорреляционная функция случайного сигнала');
```

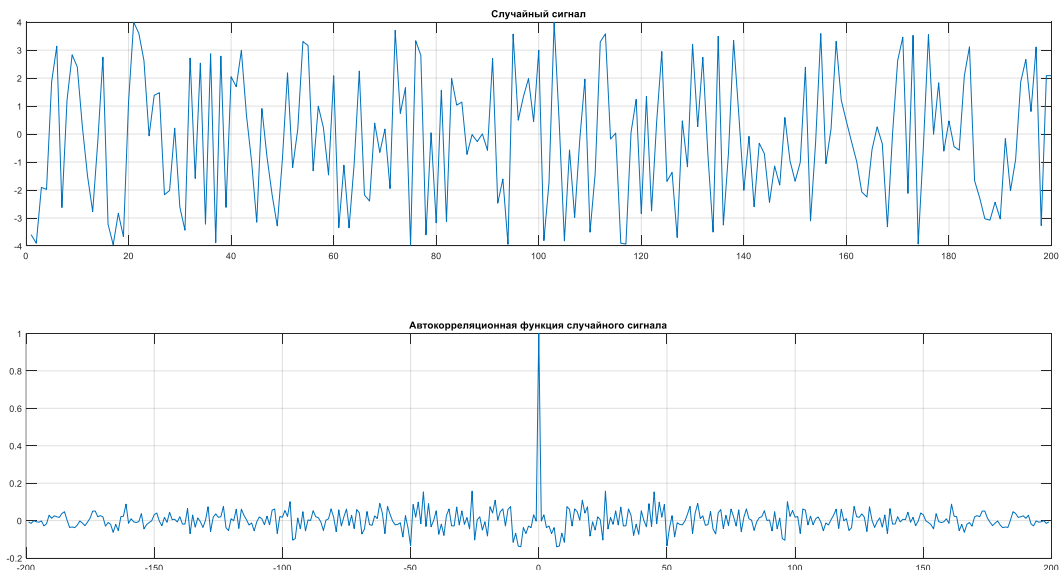


Рисунок 3 - Автокорреляция случайного сигнала

Вывод:

В данном задании возьмём случайный сигнал и построим график автокорреляционной функции этого сигнала. Из рисунка видно, что для случайных сигналов график автокорреляционной функции имеет свой максимум при $j=0$ и стремится к нулю с увеличением сдвига j .

Задание 4: Удаление эффекта “Echo” в среде MATLAB

```
clear, clc, close all

[x, fs] = audioread('bas-gitara-zvuk-zvuk-42332.wav');

k = 0.5;
d = 10000; % Задержка

y = zeros(size(x));

for i = d+1 : length(x)
    y(i,:) = x(i,:) + k*x(i-d,:);
end

figure;

[xc, lags] = xcorr(y, 'coeff');
xc = xc(lags>0);
lags = lags(lags>0);

subplot(2,1,1);
plot(lags/fs,xc), xlim([0 length(lags)/fs]), grid on;
xlabel('Временной сдвиг, с');
title('Автокорреляционная функция');

subplot(2,1,2);
findpeaks(xc,lags/fs, 'MinPeakHeight', 0.3), grid on;
xlabel('Временной сдвиг, с');
title('Пики функции');
[peaks,d1] =
findpeaks(xc(200:length(xc)),lags(200:length(xc)), 'MinPeakHeight', 0.3);
% БИХ-фильтр
y_clean = filter(1,[1 zeros(1,d1-1) 0.5],y);

figure;
subplot(3,1,1);
plot(x), grid on;
title('Исходный сигнал')

subplot(3,1,2);
plot(y), grid on;
title('Сигнал с эффектом “Echo”')

subplot(3,1,3);
plot(y_clean), grid on;
title('Сигнал без эффекта “Echo”')
```

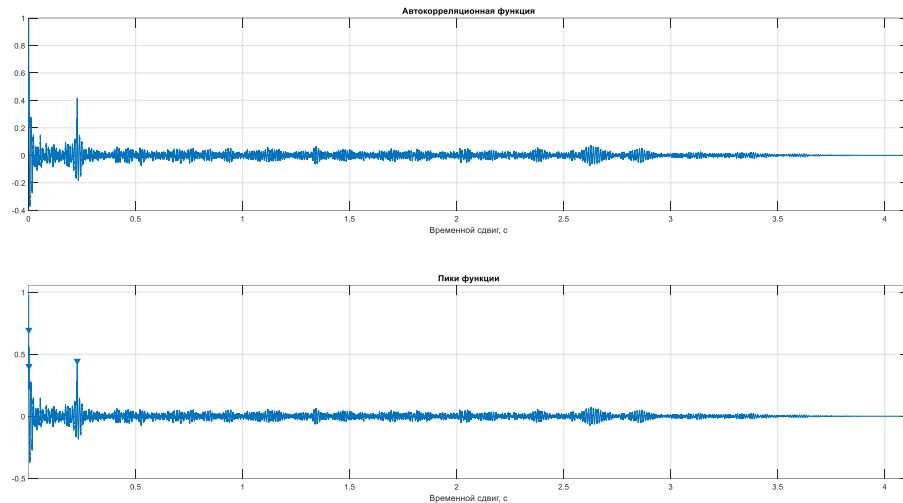


Рисунок 4.1 - Автокорреляционная функция и функция поиска пиков

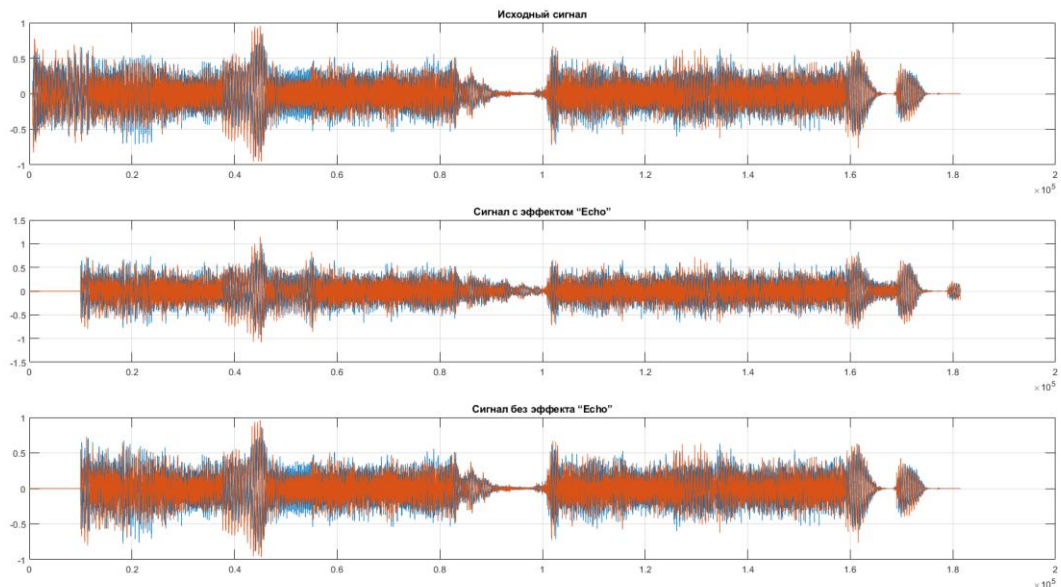


Рисунок 4.2 - Графики звуковых сигналов

Вывод:

В данном задании создадим звуковой сигнал с эффектом “Echo” и построим график автокорреляционной функции этого сигнала. С помощью функции поиска пиков обнаружим пик, откуда идет эффект "Echo". Затем мы используем фильтр БИХ для удаления эффекта "Echo".

Задание 5: Корреляция изображений в среде MATLAB

```
clear, clc, close all;

cat = imread('cat.jpg');
catGray = rgb2gray(cat);

owls = imread('owls.jpg');
owlsGray = rgb2gray(owls);

montage({owls,cat})

figure
imshow(owls), hold on;

rCat = normxcorr2(catGray,owlsGray);
[maxVal,maxIndex] = max(abs(rCat(:)));
[max_Y,max_X] = ind2sub(size(rCat),maxIndex);

rectangle('Position',[(max_X-102) (max_Y-75) 102 75], ...
          'LineWidth',4,'EdgeColor','g','Curvature',[0 0]);

figure;

srf = surf(rCat);
set(srf, 'LineStyle', 'none');
```



Рисунок 5.1 – Исходное изображение и искомое изображение

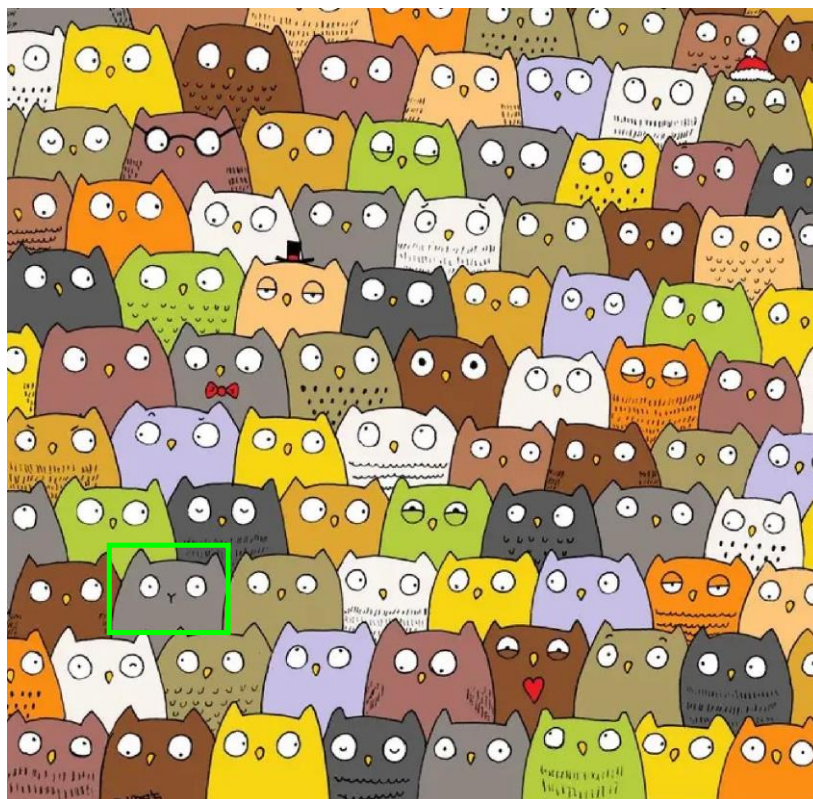


Рисунок 5.2 – Обнаруженное изображение

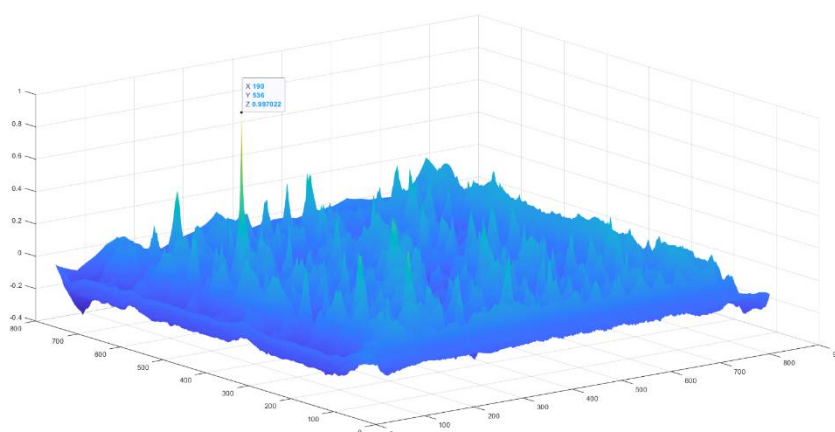


Рисунок 5.3 – График результатов корреляционного анализа

Вывод:

В этом задании найдем искомое изображение в заданном изображении. С помощью функции `normxcorr2` мы можем найти корреляцию между этими двумя изображениями, а затем найти координаты нужного искомого изображения.