

Afshar - Q2

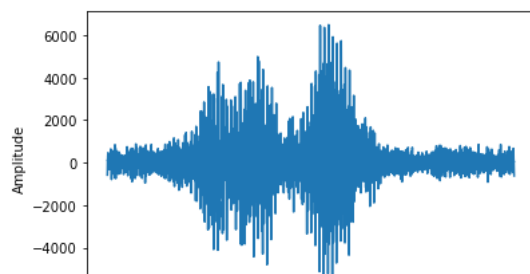
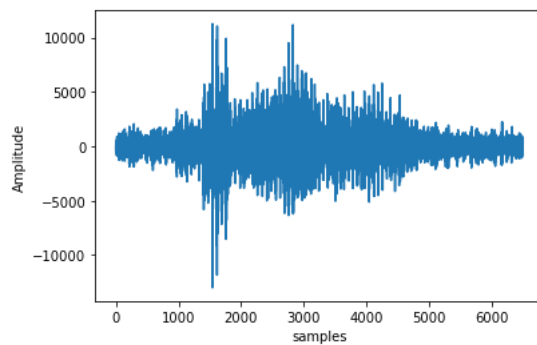
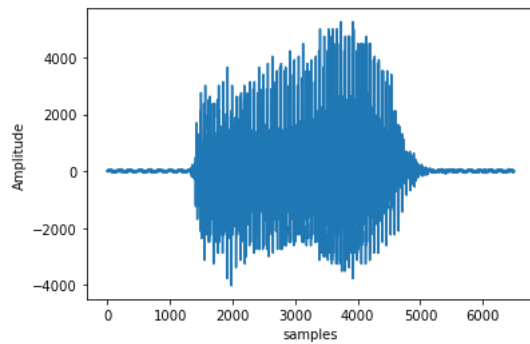
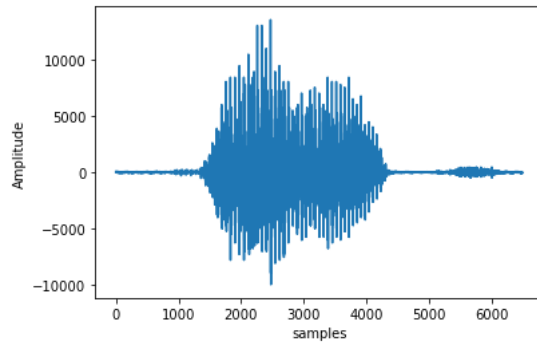
در این سوال ابتدا دو فایل صوتی خوانده میشود سپس باید تبدیل فوریه این دو فایل محاسبه شود و جای فاز و دامنه صوت ها عوض شود، در نهایت سیگنال های جدید ساخته می شوند. سپس مجدد این کار برای دو عکس داده شده انجام می شود که نشان دهنده تاثیر جابجایی فاز و دامنه است. خروجی شامل صوت های تغییر کرده و همچنین تصاویر تغییر یافته می باشد. در مرحله آخر هم تاثیر اضافه کردن نویز به فاز و دامنه مورد بررسی قرار گرفته است.

در ۲ تصویر اول شکل زمانی دو سیگنال صوتی داده شده را مشاهده میکنید.

دو تصویر آخر به ترتیب خروجی حاصل از ترکیب دامنه سیگنال اول و فاز سیگنال دوم و ترکیب دامنه سیگنال دوم و فاز سیگنال اول است.

```
1 import matplotlib.pyplot as plt
2 from scipy.io import wavfile
3 import numpy as np
4 import os
5
6 # part1
7 fs1, x1 = wavfile.read('1.wav')
8 fs2, x2 = wavfile.read('2.wav')
9
10 plt.figure()
11 plt.plot(x1)
12 plt.xlabel('samples')
13 plt.ylabel('Amplitude')
14
15 plt.figure()
16 plt.plot(x2)
17 plt.xlabel('samples')
18 plt.ylabel('Amplitude')
19
20 X1 = np.fft.fft(x1)
21 X2 = np.fft.fft(x2)
22
23 magX1 = np.abs(X1)
24 magX2 = np.abs(X2)
25
26 angX1 = np.angle(X1)
27 angX2 = np.angle(X2)
28
29 X3 = magX1*np.exp(1j*angX2)
30 X4 = magX2*np.exp(1j*angX1)
31
32 x3 = np.fft.ifft(X3)
33 x4 = np.fft.ifft(X4)
34
35 plt.figure()
36 plt.plot(np.real(x3))
37 plt.xlabel('samples')
38 plt.ylabel('Amplitude')
39
40 plt.figure()
41 plt.plot(np.real(x4))
42 plt.xlabel('samples')
43 plt.ylabel('Amplitude')
44
```

Text(0, 0.5, 'Amplitude')



لازم به ذکر است

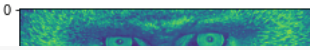
برای شنیدن هر دو صوت، باید یکی را کامنت کنید تا دیگری را بشنوید.

```
1 #part 2
2 from IPython.display import Audio
3
4 Audio(x3, rate=fs1)
5 Audio(x4, rate=fs2)
6
```

Audio x4 is like 1.wav

Audio x3 is like 2.wav

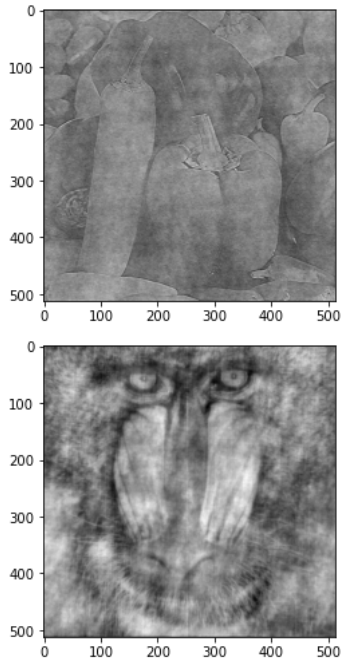
```
1 #part3
2 from scipy.fftpack import fft2, ifft2, fftshift
3 from matplotlib import pyplot as plt
4 from PIL import Image
5
6 # Read the images
7 I1 = Image.open('Baboon.bmp')
8 I2 = Image.open('Peppers.bmp')
9
10 # Show the images
11 plt.figure()
12 plt.imshow(I1)
13
14 plt.figure()
15 plt.imshow(I2)
16
17 # Compute the FFTs and magnitudes
18 FI1 = fftshift(fft2(I1))
19 FI2 = fftshift(fft2(I2))
20
21 magI1 = np.abs(FI1)
22 magI2 = np.abs(FI2)
23
24 angI1 = np.angle(FI1)
25 angI2 = np.angle(FI2)
26
27 # Compute the new FFTs
28 FI3 = magI1 * np.exp(1j*angI2)
29 FI4 = magI2 * np.exp(1j*angI1)
30
31 # Compute the new images
32 I3 = np.real(ifft2(fftshift(FI3)))
33 I4 = np.real(ifft2(fftshift(FI4)))
34
```



```

1 plt.figure()
2 plt.imshow(I3, cmap='gray')
3
4 plt.figure()
5 plt.imshow(I4, cmap='gray')
6
7 plt.show()
8

```



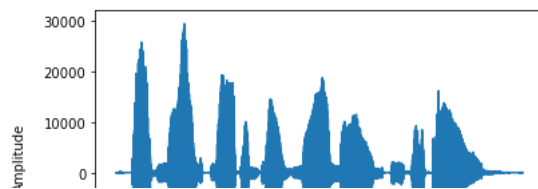
خروجی دوم شباهت بیشتری به عکس حیوان دارد و خروجی اول بیشتر شبیه به عکس فلفل است.

```

1 import scipy.io.wavfile as wav
2 import matplotlib.pyplot as plt
3 from scipy.fftpack import fft, ifft, fftshift
4 from scipy.signal import chirp
5
6 # Read the wav file
7 Fs, x_test = wav.read('test.wav')
8
9 # Plot the samples
10 plt.figure()
11 plt.plot(x_test)
12 plt.xlabel('Samples')
13 plt.ylabel('Amplitude')
14
15 # Compute the FFT
16 X_test = fftshift(fft(x_test))
17
18 # Compute the magnitude and angle
19 mag_X_test = np.abs(X_test)
20 ang_X_test = np.angle(X_test)
21
22 # Add noise to the magnitude and angle
23 mag_X_test_n = np.random.normal(0, np.std(mag_X_test)/5, len(mag_X_test)) + mag_X_test

```

```
24 ang_X_test_n = np.random.normal(0,np.std(ang_X_test)/5, len(ang_X_test)) + ang_X_test
25
26 # Recovered signals
27 AMP_rec = mag_X_test_n * np.exp(1j*ang_X_test)
28 ANG_rec = mag_X_test * np.exp(1j*ang_X_test_n)
29
30 # Compute the ifft
31 amp_rec = np.real(iff(fftshift(AMP_rec)))
32 ang_rec = np.real(iff(fftshift(ANG_rec)))
33
34 # Plot the recovered signals
35 plt.figure()
36 plt.plot(amp_rec)
37 plt.xlabel('Samples')
38 plt.ylabel('Amplitude')
39 plt.title('Noisy Amplitude')
40
41 plt.figure()
42 plt.plot(ang_rec)
43 plt.xlabel('Samples')
44 plt.ylabel('Amplitude')
45 plt.title('Noisy Phase')
46
47 # Play the sound
48 from IPython.display import Audio
49
50 Audio(amp_rec, rate=Fs)
51 Audio(ang_rec, rate=Fs)
52
```



همان‌طور که دیده می‌شود در تصاویر تغییر دامنه و فاز تأثیر گذار بوده و تصویر با تصویر قبل خیلی متفاوت می‌شود. در صوت نیز تأثیر فاز تقریباً ناچیز است.

ولی به شخصه در این مثال در صوتی که شخص می‌گفت: یک، شاید اگر فایل صوت اصلی را نمی‌شنیدم متوجه نمی‌شدم که در صوت دومی که در خروجی شنیدم چه می‌گوید.

1

