به نام خدا

دانشگاه تهران

دانشکده مهندسی برق و کامپیوتر

پردازش سیگنال‌های زمان - گسسته

تمرین سری **4**

نام و نام خانوادگی	نرجس نورزاد
شماره دانشجویی	810196626
تاریخ ارسال گزارش	13 مرداد 1399

فهرست گزارش سوالات

3 ..... Spatial domain filtering

13.....Frequency domain filtering

21.....Face recognition

عملکرد kernel ها :

برای هر بلوک 3 × 3 از پیکسل ها ، هر پیکسل در ماتریس 3 ×3 ای که ما تعریف کردیم ضرب میشود ، سپس جمع تک تک درایه های آن گرفته میشود و این حاصل جمع، پیکسل جدید را میسازد .

- Sharpen :

این نوع kernel تفاوت های میان مقدار پیکس های همسایه را برجسته تر میکند پس باعث میشود که گوشه ها در هر پیکسل نمایان تر شود. برجسته کردن نواحی روشن و تاریک باعث اضافه کردن contrast و برجسته شدن گوشه ها میشود .

- Blur :

این kernel با عبور یک Moving Average ک مقدار هر خانه را با میانگین مقدار خانه‌های همسایه‌اش جایگزین می‌کند، می‌توان تصویر را محو کرد.

- Outline:

این kernel که به آن edge کرنل هم گفته میشود برای هایلایت کردن تفاوت های بزرگ میان پیکسل های مختلف است در واقع میتوان خطوط اطراف هر شی در تصویر را به کمک این kernel پررنگ تر و برجسته تر کرد .

- Gauss :

این نوع kernel ، همانند blur kernel عمل میکند با این تفاوت که بر تابع Gaussian وابسته است ، تابعی که یک مقادیر را در اطراف یک نقطه ی مرکزی توزیع میکند ، پس اعمال کردن این نوع kernel

- Average moving :

این نوع kernel نیز همانند کرنل blur عمل میکند برای همین با جایگزینی مقدار هر خانه با میانگین خانه های اطراف باعث تار شدن کل تصویر میشود .

- H line :

این kernel ، یکی از انواع line detection کرنل هاست .این نوع کرنل ها ، باعث detect شدن خطوط با قطر مشخص و زاویه مشخص میشوند .

این کرنل ها باعث نمایان کردن خطوط با زمینه ی تیره و خطوط روشن میشوند .

H lineخطوط عمودی در تصویر را برجسته میکند

- V line :

این kernel ، یکی از انواع line detection کرنل هاست .این نوع کرنل ها ، باعث detect شدن خطوط با قطر مشخص و زاویه مشخص میشوند .

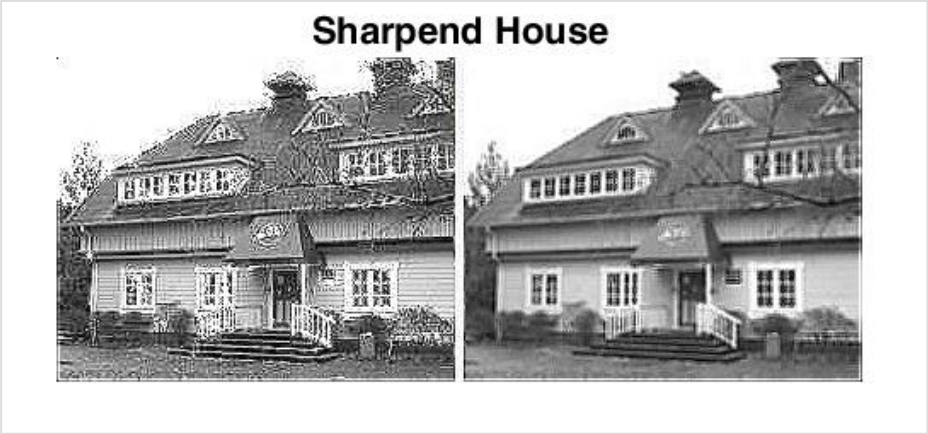
این کرنل ها باعث نمایان کردن خطوط با زمینه ی تیره و خطوط روشن میشوند .

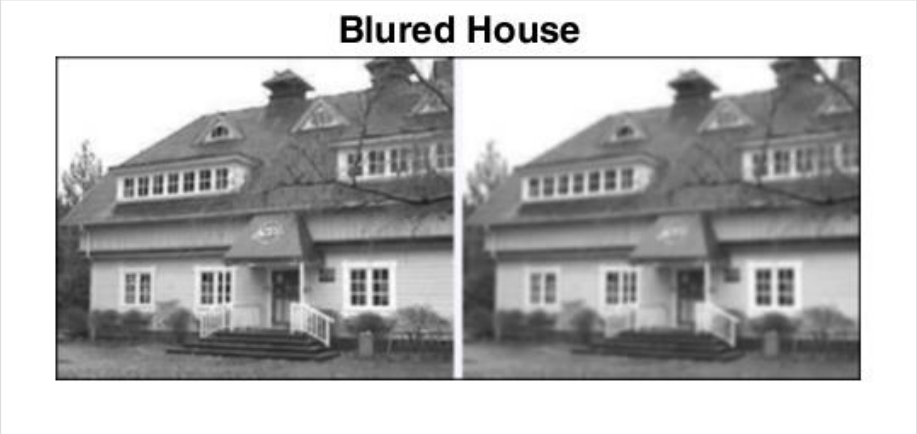
این kernel خطوط افقی در تصویر را برجسته میکند

- Identify:

این kernel که به آن do-nothing کرنل هم گفته میشود ، در واقع باعث ایجاد هیچ گونه تفاوتی نمیشود و کار خاصی روی تصویر انجام نمیدهد.

HOUSE





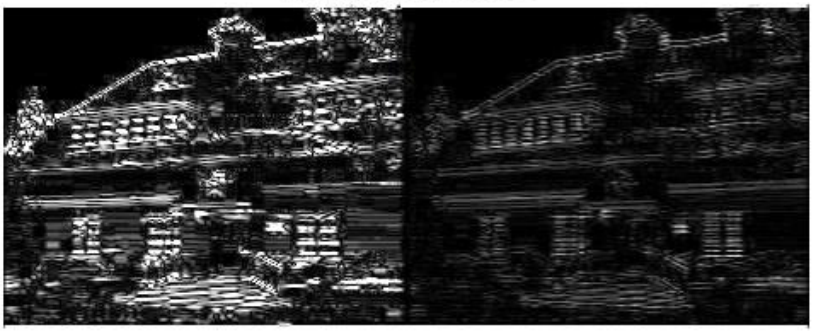
Outlined House

Gaussed House

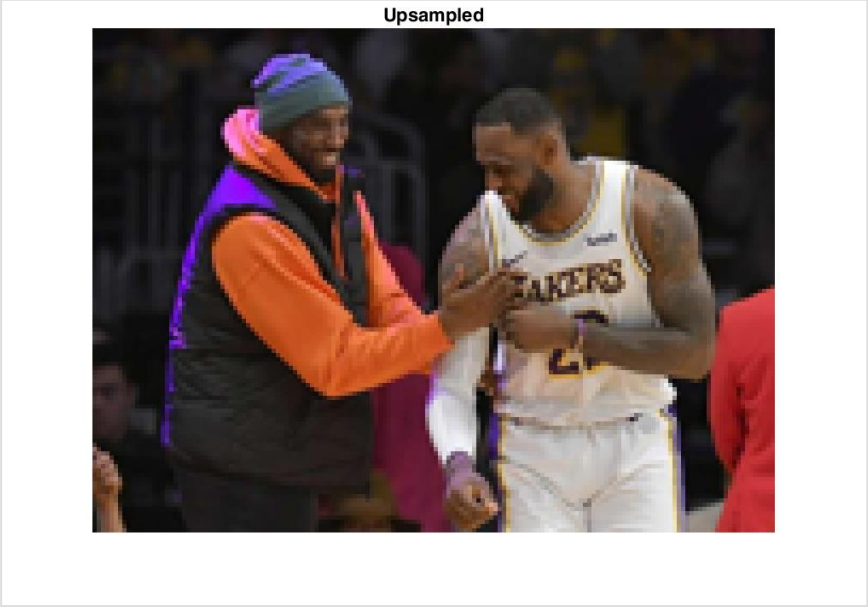
Moving avged House

V lined House

H lined House



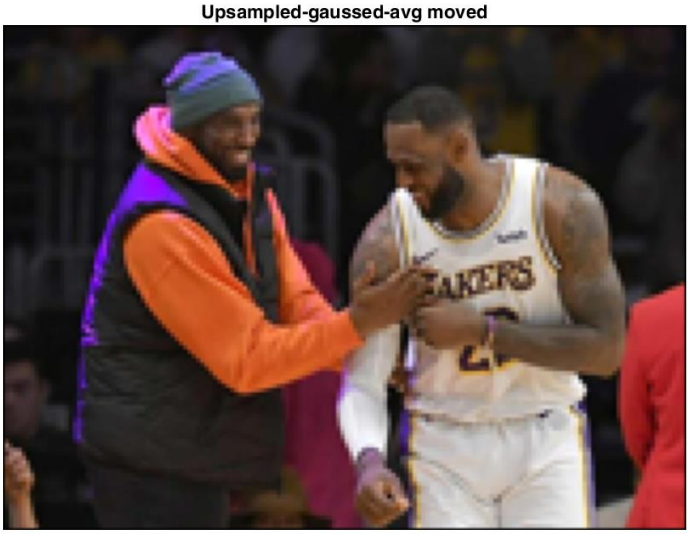
Identified House



علت تفاوت میان دو تصویر این است که ، بعد از downsample کردن ، ما داده هارا دور میریزیم و با upsample کردن صرفا جای داده هایی ک نداریم 0 قرار میدهیم ، به همین دلیل باعث مشود که تصویر وضوح اولیه را نداشته باشد و تار باشد.

تصویر زیر ابتدا کرنل گوسی اعمال میشود ، سپس کرنل avg moving که باعث مشود خانه هایی ک قبلا در آن ها 0 قرار گرفته با میانگین خانه های اطراف خود جایگزین شوند که باعث میشود اندکی از تاری تصویر کاهیده شود .

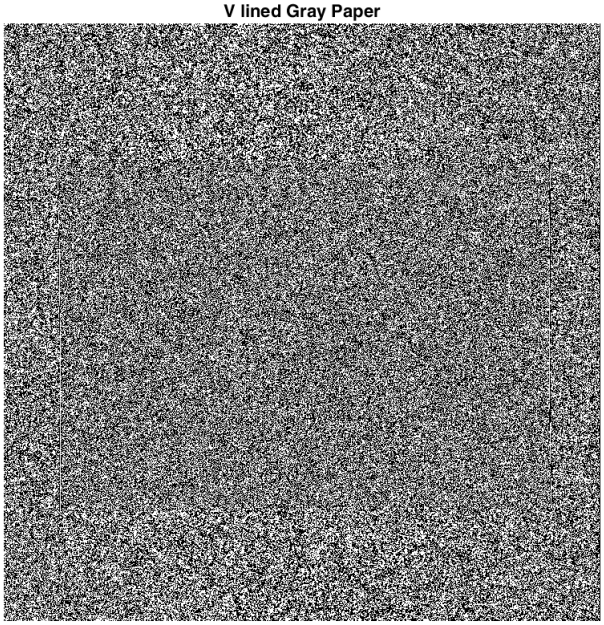
Upsampled-gaussed-avg moved

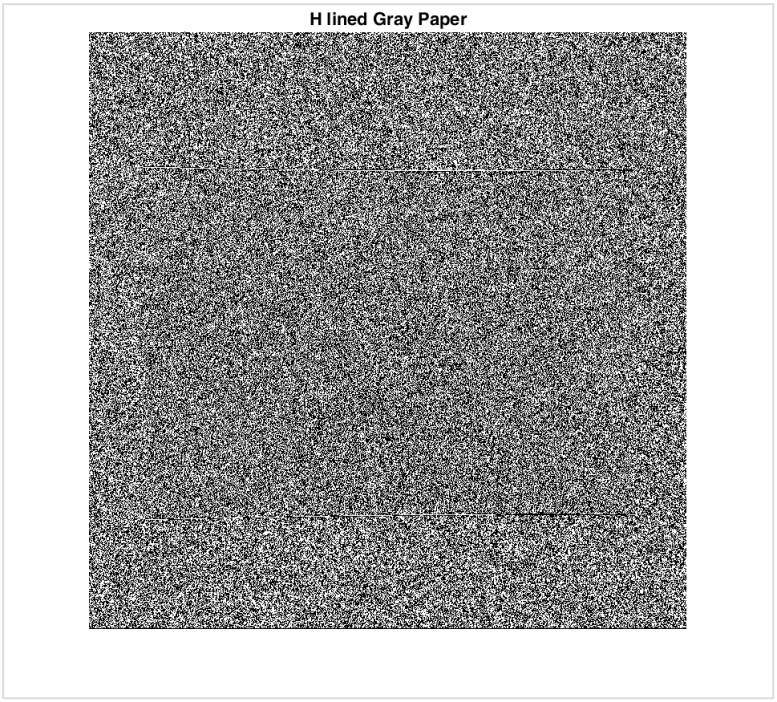


CAM SCANNER

برای طراحی چیزی مانند Camscanner ، ابتدا تصویر را به حوزه ی gray scale میبریم . سپس به کمک کرنل هایی که از قبل داشتیم ، خطوط عمودی و افقی را مشخص کنیم .

V lined Gray Paper





میتوان دید با استفاده از کرنل ها ، خطوط سفید رنگی در تصویر ایجاد شده اند .

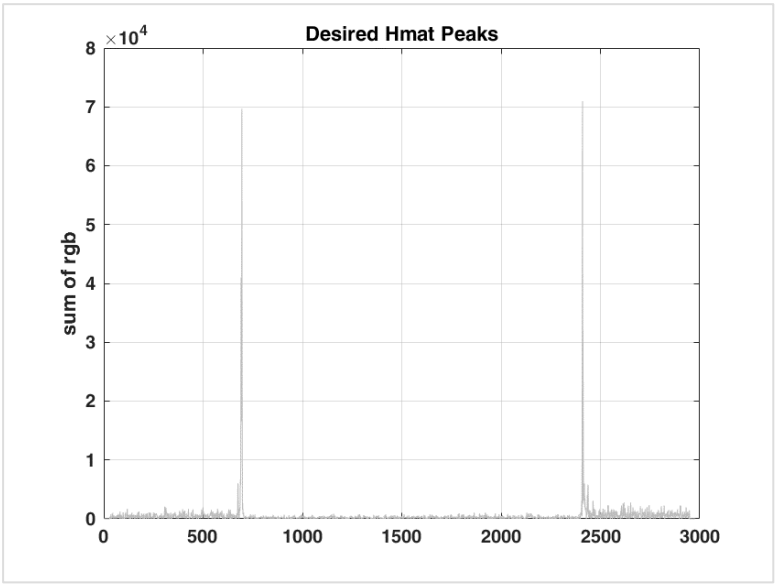
ابتدا به کمک فیلتر median دو بعدی ، میخواهیم نویز های موجود در تصویر را از بین ببریم . در واقع این نوع فیلتر مانند پنجره ای عمل میکند که رویentry سیگنال حرکت میکند و median همسایه های هر entryرا با اصل مقدار آن جایگزین میکند .

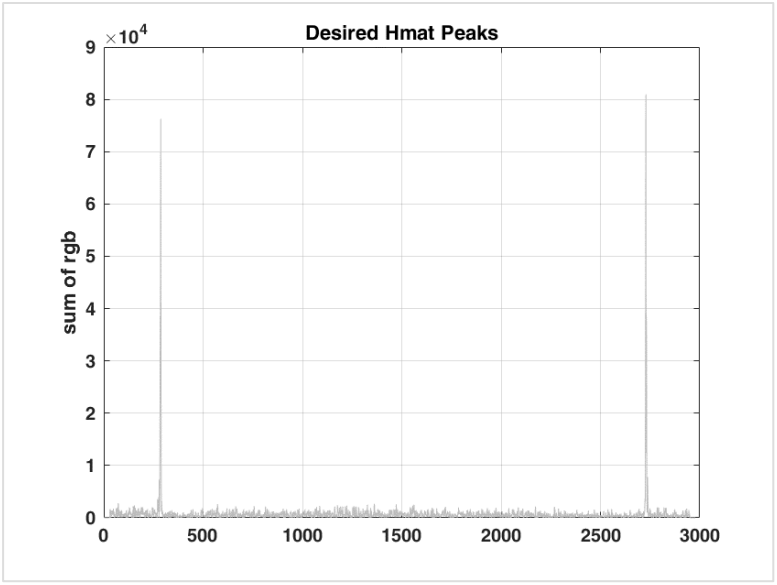
سپس ماتریس desired_Hmat (desired_Vmat) را به صورت افقی (عمودی) جمع میکنیم تا یک ماتریس 1*2978 حاصل شود . در این ماتریس ، 2 تا از درایه ها بیشترین مقدار را دارند که نشان میدهد خطوط اصلی کاغذ ما در کجا قرار دارند .

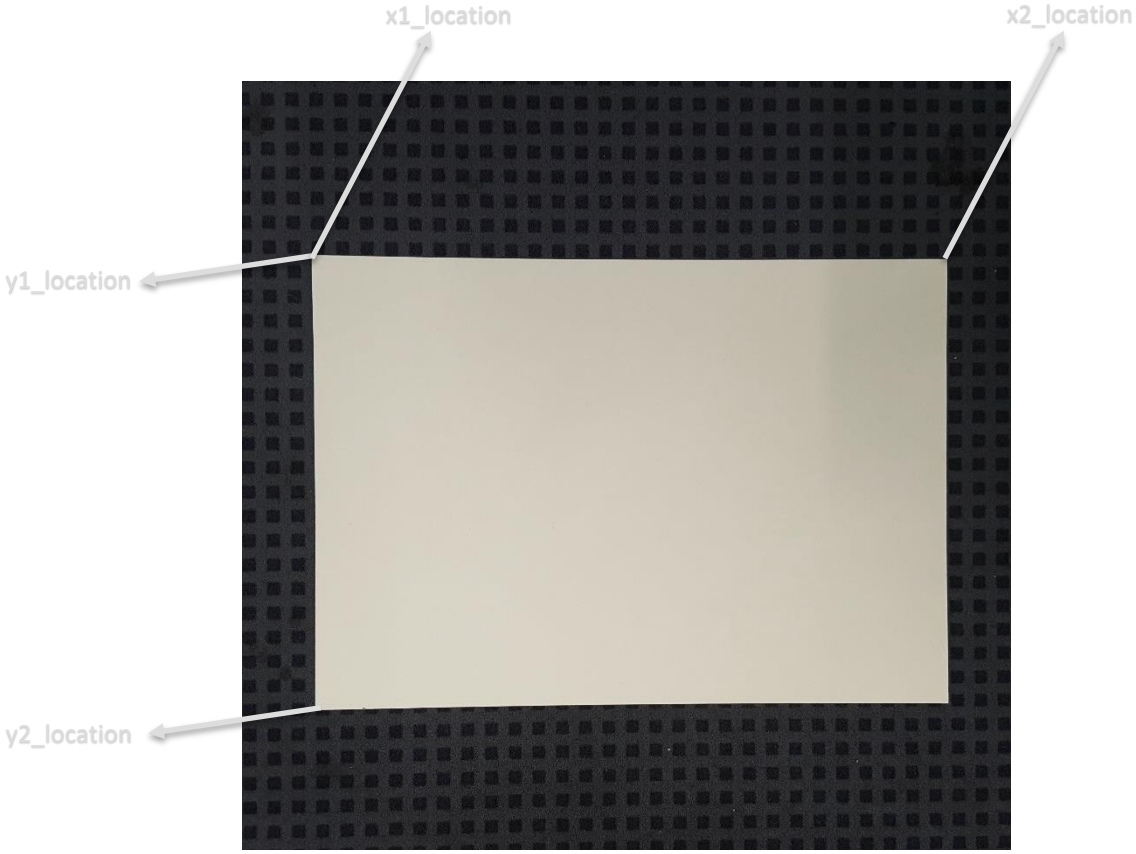
سپس با استفاده از تابع remove_unwanted_data ، اول و آخر ماتریس را صفر میکنیم چون پیک های نامربوط میزند که مورد نظر ما نیست .

سپس ماتریس را رسم میکنیم . دو پیک به وضوح قابل دیدن است .

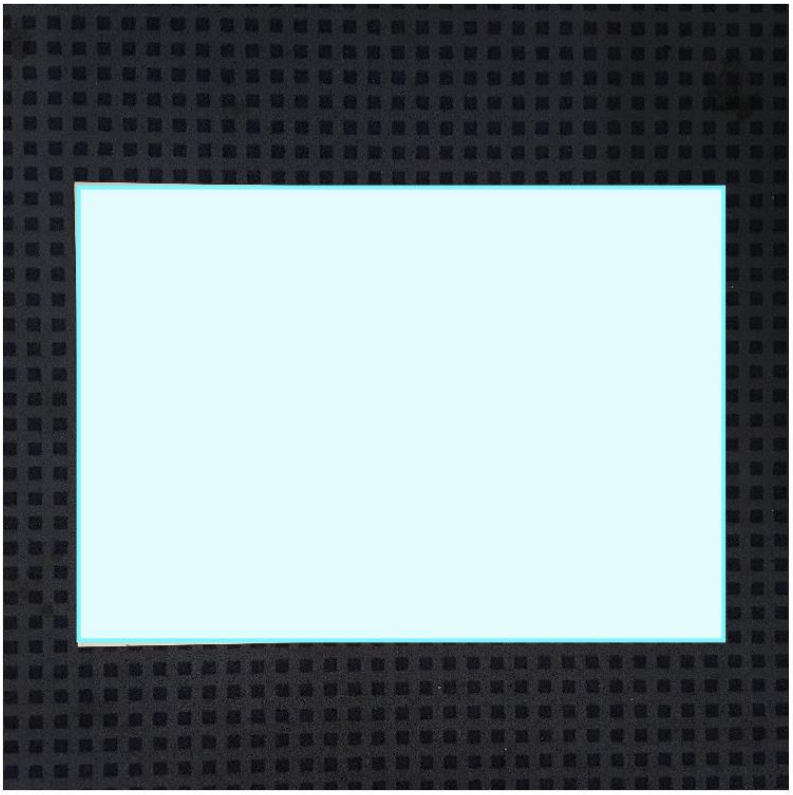
با استفاده از تابع find_position ، به راحتی میتوان مکان پیک هایی که چه از نظر افقی و چه از نظرعمودی میزند را یافت .


Desired Hmat Peaks
×10^4
sum of rgb
0
1
2
3
4
5
6
7
8
0
500
1000
1500
2000
2500
3000


Desired Hmat Peaks
×10^4
sum of rgb
0
1
2
3
4
5
6
7
8
9
0
500
1000
1500
2000
2500
3000

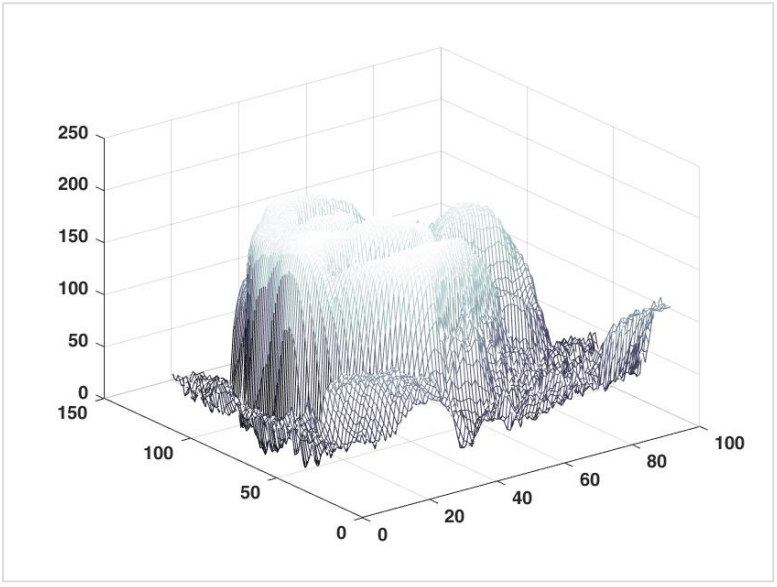


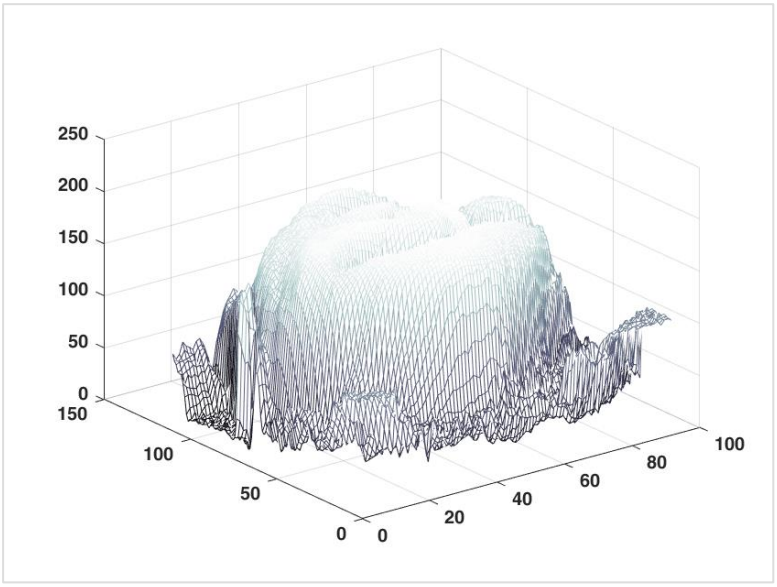
بعد از اینکه مختصات 4 نقطه ی تصویر بدست آمد ، به کمک تابع rectangle یک مستطیل میکشیم که انگار کاغذ ما detect شده است.

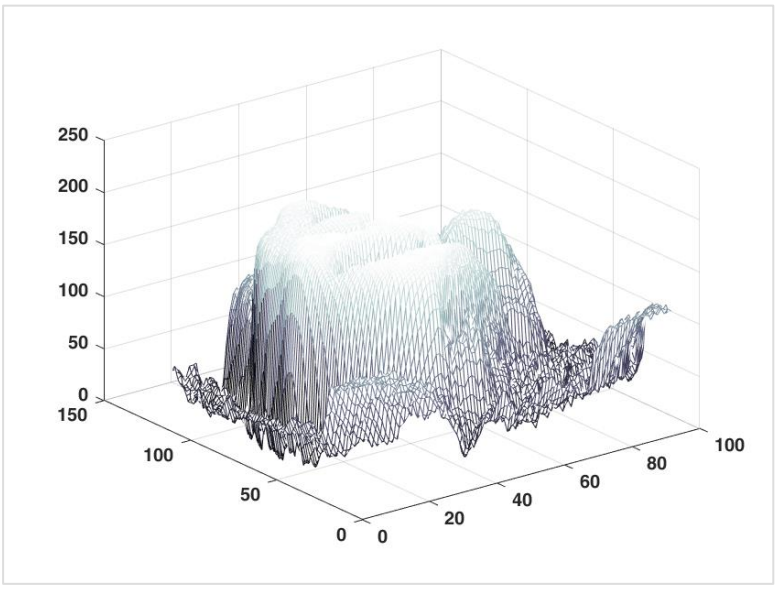


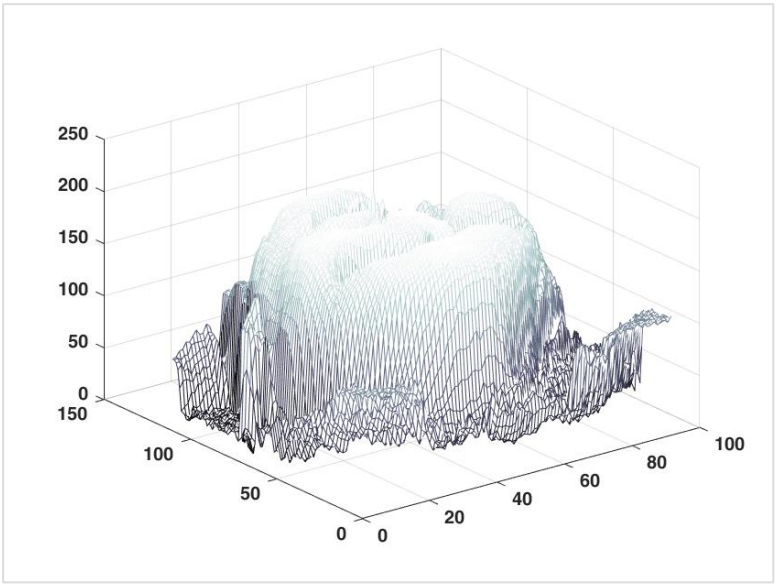
FREQUENCY DOMAIN FILTERING

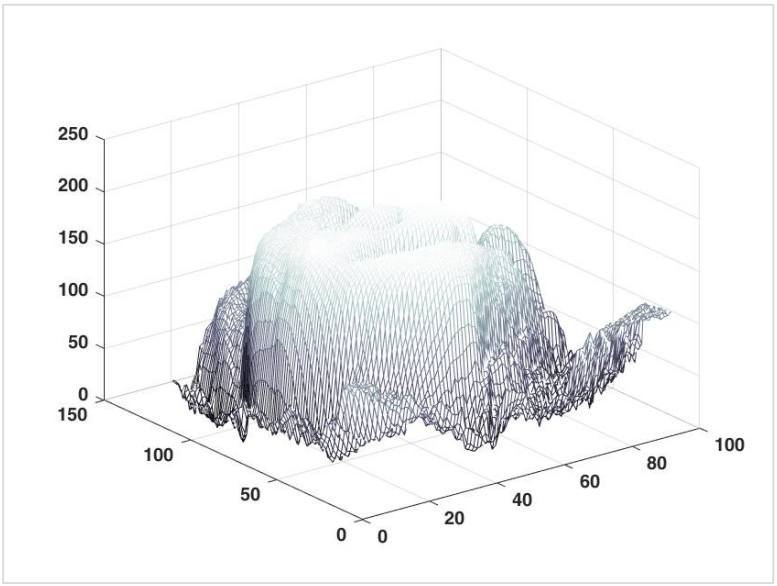
.1

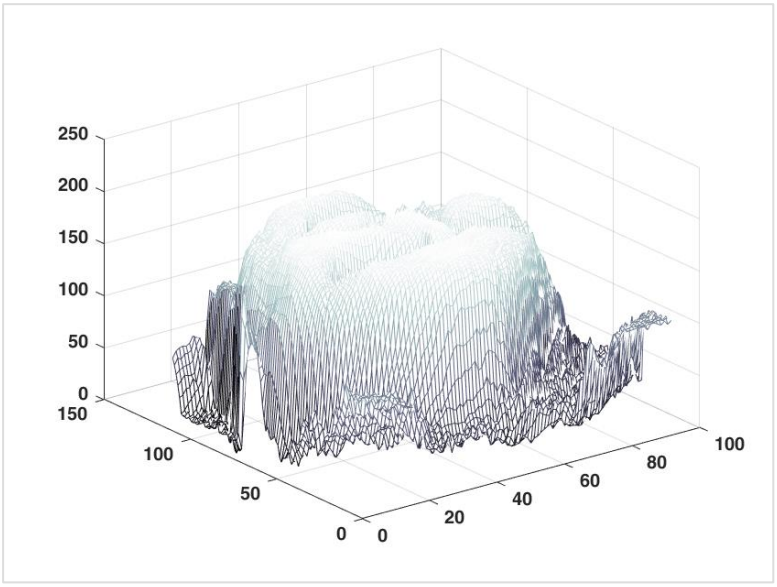


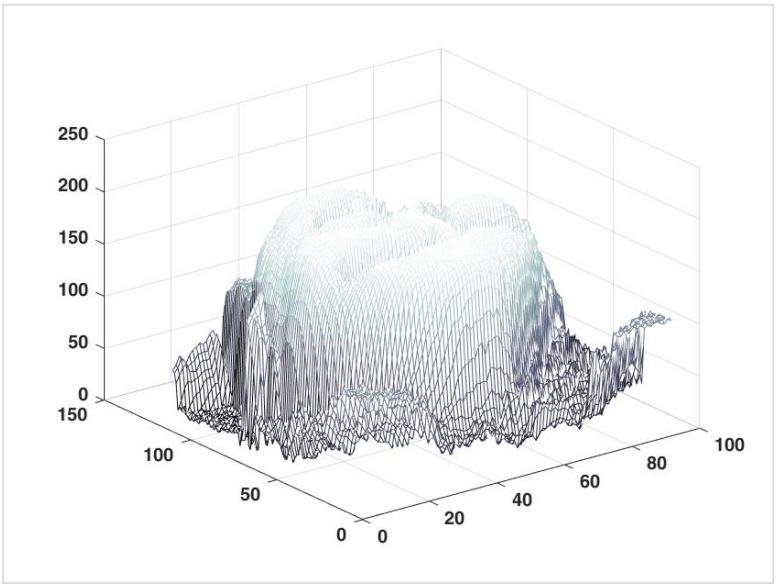

250
200
150
100
50
0
150
100
50
0
0
20
40
60
80
100

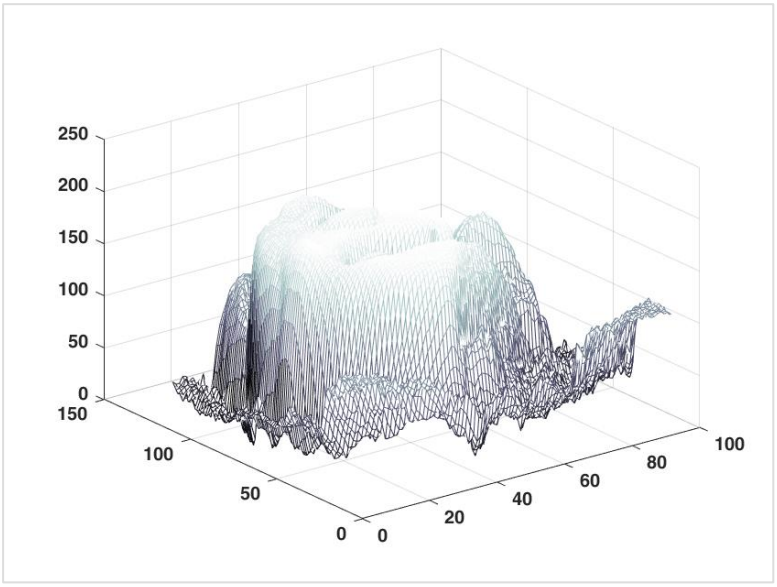

250
200
150
100
50
0
150
100
50
0
0
20
40
60
80
100

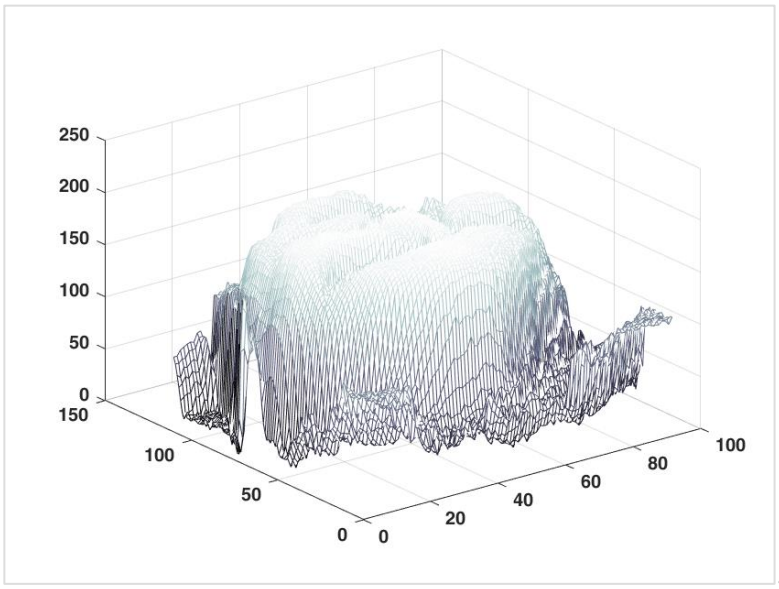

250
200
150
100
50
0
150
100
50
0
0
20
40
60
80
100


250
200
150
100
50
0
150
100
50
0
0
20
40
60
80
100

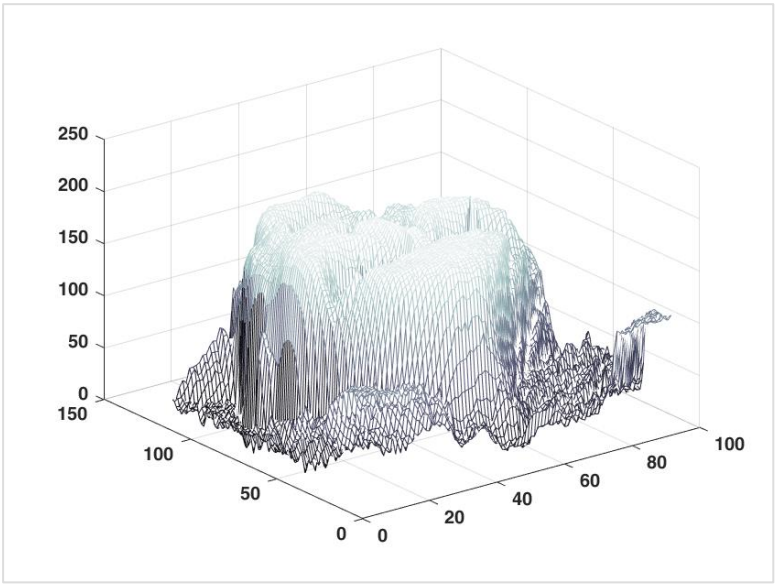

250
200
150
100
50
0
150
100
50
0
0
20
40
60
80
100

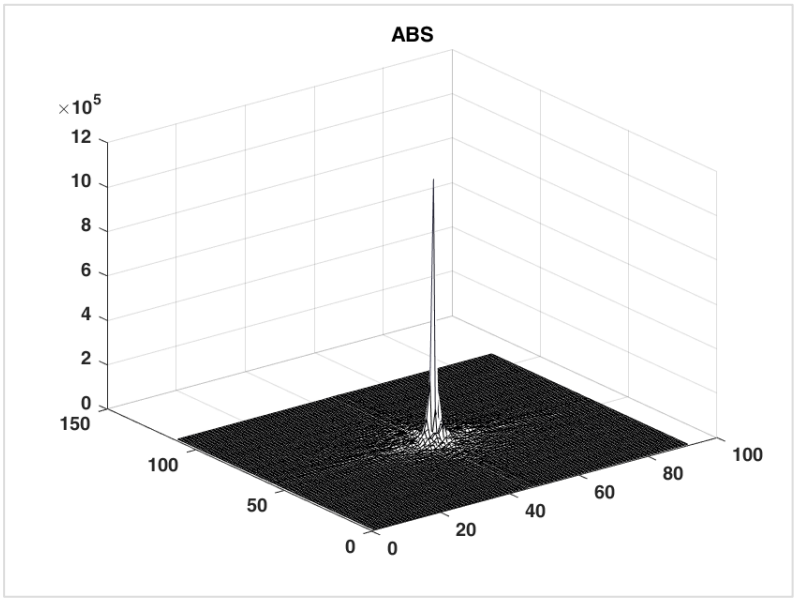

250
200
150
100
50
0
150
100
50
0
0
20
40
60
80
100

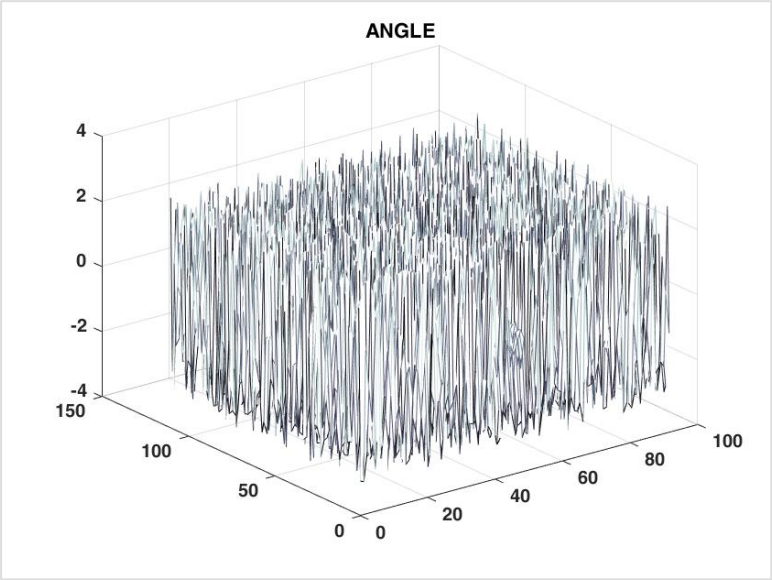

250
200
150
100
50
0
150
100
50
0
0
20
40
60
80
100


250
200
150
100
50
0
150
100
50
0
0
20
40
60
80
100

4


250
200
150
100
50
0
150
100
50
0
0
20
40
60
80
100


ABS
×10^5
12
10
8
6
4
2
0
150
100
50
0
0
20
40
60
80
100



دو تصویر بالا فقط اندازه و فاز تصویر اول هستند و به دلیل تشابه بقیه موراد آورده نشدند .

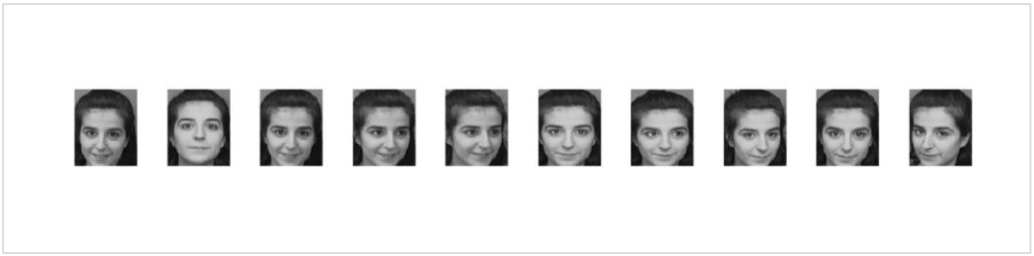
واضح است که مخدوش کردن فاز به نسبت بسیار خوبی تصاویر را نیز مخدوش کرده . افزایش مقدار snr روی تصاویر مختلف تاثیرات مختلفی گذاشته است .

2.

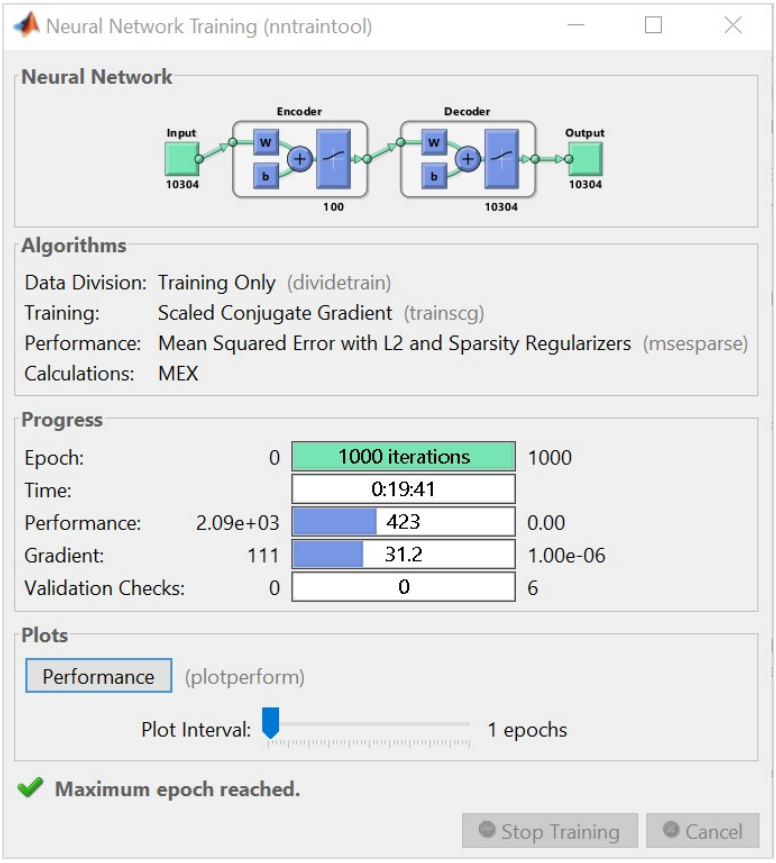
همانطور ک واضح است مخدوش کردن اندازه تاثیر انچنانی ای روی تصاویر نداشته است و صرفا باعث اندکی تیرگی و روشنی شده است .

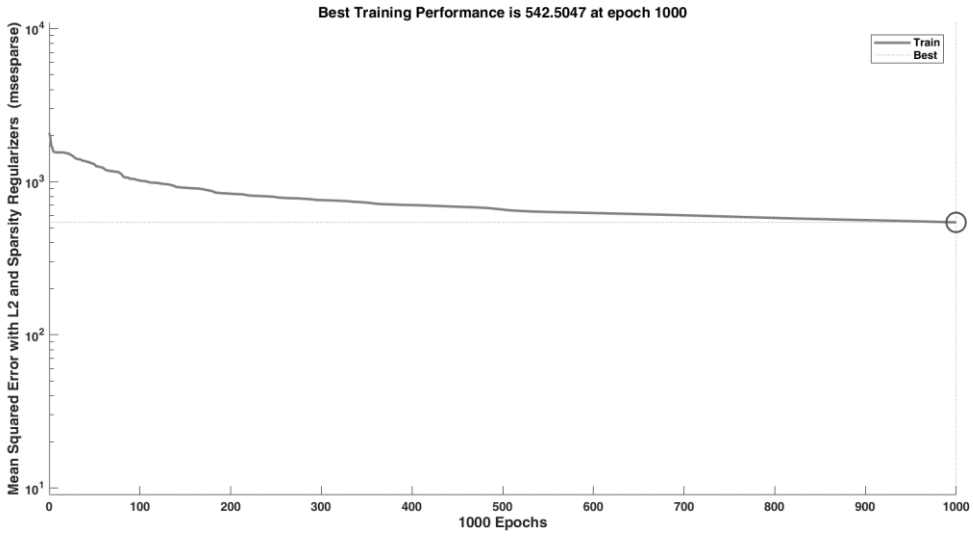
پس میتوان نتیجه گرفت که اطلاعات مهم و اساسی در فاز تصویر وجود دارد .

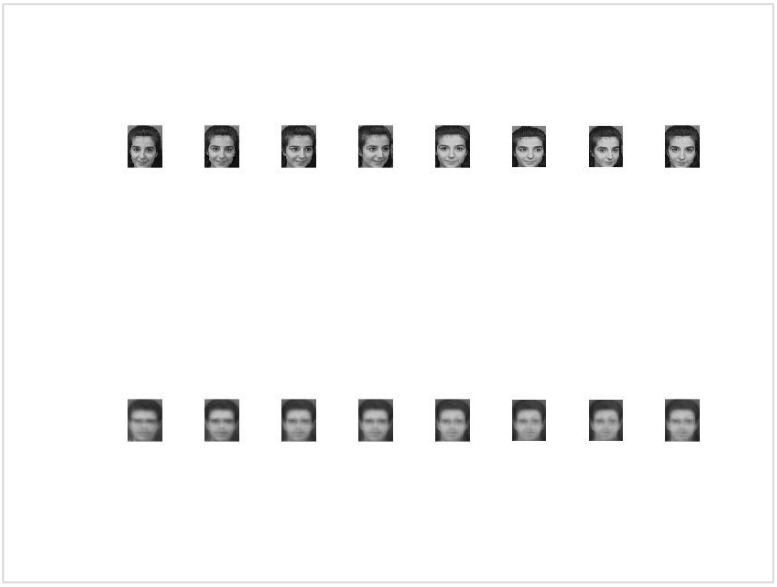
3.



از بررسی تمام تصاویر بالا میتوان متوجه شد بخش قابل توجه و اساسی عکس در فاز آن قرار دارد و اطلاعات بیشتری در مورد تصویر چهره به ما میدهد.


Neural Network Training (nntraintool)
Neural Network
Encoder
Decoder
Input
Output
W
b
10304
100
10304
10304
Algorithms
Data Division: Training Only (dividetrain)
Training: Scaled Conjugate Gradient (trainscg)
Performance: Mean Squared Error with L2 and Sparsity Regularizers (msesparse)
Calculations: MEX
Progress
Epoch: 0 1000 iterations 1000
Time: 0:19:41
Performance: 2.09e+03 423 0.00
Gradient: 111 31.2 1.00e-06
Validation Checks: 0 0 6
Plots
Performance
(plotperform)
Plot Interval: 1 epochs
Maximum epoch reached.
Stop Training
Cancel


Best Training Performance is 542.5047 at epoch 1000
Train
Best
Mean Squared Error with L2 and Sparsity Regularizers (msesparse)
1000 Epochs



کد به جواب دقیقی نرسید.