subsample.m

```
function [ subimage ] = subsample(image, factor)

% calculate the new dimension, make each new pixel = old pixel * factor
scale = [sqrt(factor) sqrt(factor)];
oldSize = size(image);
newSize = max(floor(scale.*oldSize(1:2)),1);

% Compute the new index

rowIndex = min(round((1:newSize(1))./scale(1)),oldSize(1));
colIndex = min(round((1:newSize(2))./scale(2)),oldSize(2));

% Put new index into new image
subimage = image(rowIndex,colIndex,:);

return;

end
```

shrink.m

```
function [ shrinkIM ] = shrink(image)

% calculate the new dimension, reduce the length of both col and row by 2
factor=0.5;
scale = [factor factor];
oldSize = size(image);
newSize = max(floor(scale.*oldSize(1:2)),1);

% Compute the new index
rowIndex = min(round((1:newSize(1))./scale(1)),oldSize(1));
colIndex = min(round((1:newSize(2))./scale(2)),oldSize(2));

% Put new index into new image
shrinkIM = image(rowIndex,colIndex,:);

end
```

zoom.m

```
function [ zoomIM ] = zoom( image )
% calculate the new dimension, twice the length of both col and row
factor=2;
scale = [factor factor];
oldSize = size(image);
newSize = max(floor(scale.*oldSize(1:2)),1);

% Compute the new index

rowIndex = min(round((1:newSize(1))./scale(1)),oldSize(1));
colIndex = min(round((1:newSize(2))./scale(2)),oldSize(2));

% Put new index into new image

zoomIM = image(rowIndex,colIndex,:);


end
```

myrotate.m

```
function [ myrotateIM ] = myrotate( image )
%rotate 90 degrees clockwise
theta = pi/2;
rmat = [
cos(theta) sin(theta) 0
-sin(theta) cos(theta) 0
0           0          1];

mx = size(image,2);
my = size(image,1);
corners = [
    0  0  1
    mx 0  1
    0  my 1
    mx my 1];
new_c = corners*rmat;

% transform matrix
T = maketform('affine', rmat);

myrotateIM = imtransform(image, T, ...
    'XData',[min(new_c(:,1)) max(new_c(:,1))],...
    'YData',[min(new_c(:,2)) max(new_c(:,2))]);

end
```

reflect.m

```
function [ reflectIM ] = reflect( image )
% reflect image about the vertical axis
reflectIM = image(:,end:-1:1,:);
end
```

dim.m

```
function [ dimIM ] = dim( image, fraction )
% scale every pixel in the image by fraction
dimIM=image(:, :, :)*fraction;

end
```

contrast_compress.m

```
function [ contrastIM ] = comtrast_compress( image )
% take the square root of all the pixels in every color plane and add them
% to one plane(compress)
adjIM=(sqrt(double(image(:, :, 1))) + sqrt(double(image(:, :, 2))) +
sqrt(double(image(:, :, 3))));
% scale every pixel to [0,1] intensity
contrastIM=(double(adjIM(:,:)))/255;
end
```

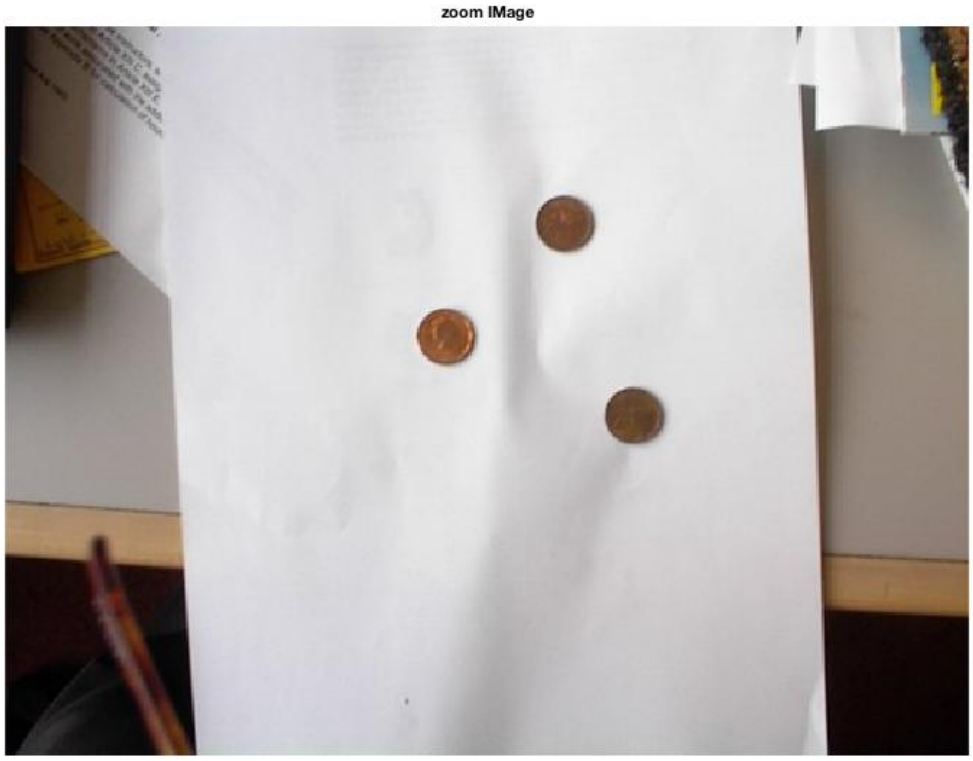
assign1.m

```
%cmpt412-assignment1
%author: Janet Xuan


image=imread('ThreePenniesAreduced.jpg');
figure
imshow(image)
title('Original Image')

%%%%%%%%%%%%%%%    subimage          %%%%%%%%%%%%%%%
[subimage] = subsample(image, 0.3);
figure
imshow(subimage)
title('subimage IMage')

%%%%%%%%%%%%%%%    shrink         %%%%%%%%%%%%%%%%%%
 [shrinkIM] = shrink(image);
 figure
 imshow(shrinkIM)
 title('shrink IMage')

%%%%%%%%%%%%%%%    zoom         %%%%%%%%%%%%%%%%%%
[zoomIM] = zoom(image);
figure
imshow(zoomIM)
title('zoom IMage')

%%%%%%%%%%%%%%%    rotate       %%%%%%%%%%%%%%%%%
[myrotateIM] = myrotate(image);
figure
imshow(myrotateIM)
title('rotate IMage 90 degrees clockwise')


%%%%%%%%%%%%%%%    reflect      %%%%%%%%%%%%%%%%%
[reflectIM] = reflect(image);
figure
imshow(reflectIM)
title('reflect IMage about the vertical axis')

%%%%%%%%%%%%%%%    dim          %%%%%%%%%%%%%%%
[dimIM] = dim(image, 0.3);
figure
imshow(dimIM)
title('darker Image by the fraction')

%%%%%%%%%%%%%%%    contrast_compress    %%%%%%%%%
[ contrastIM ] = comtrast_compress( image );
figure
imshow(contrastIM)
title('comtrast and compress image')
```

Test results:
subsample:

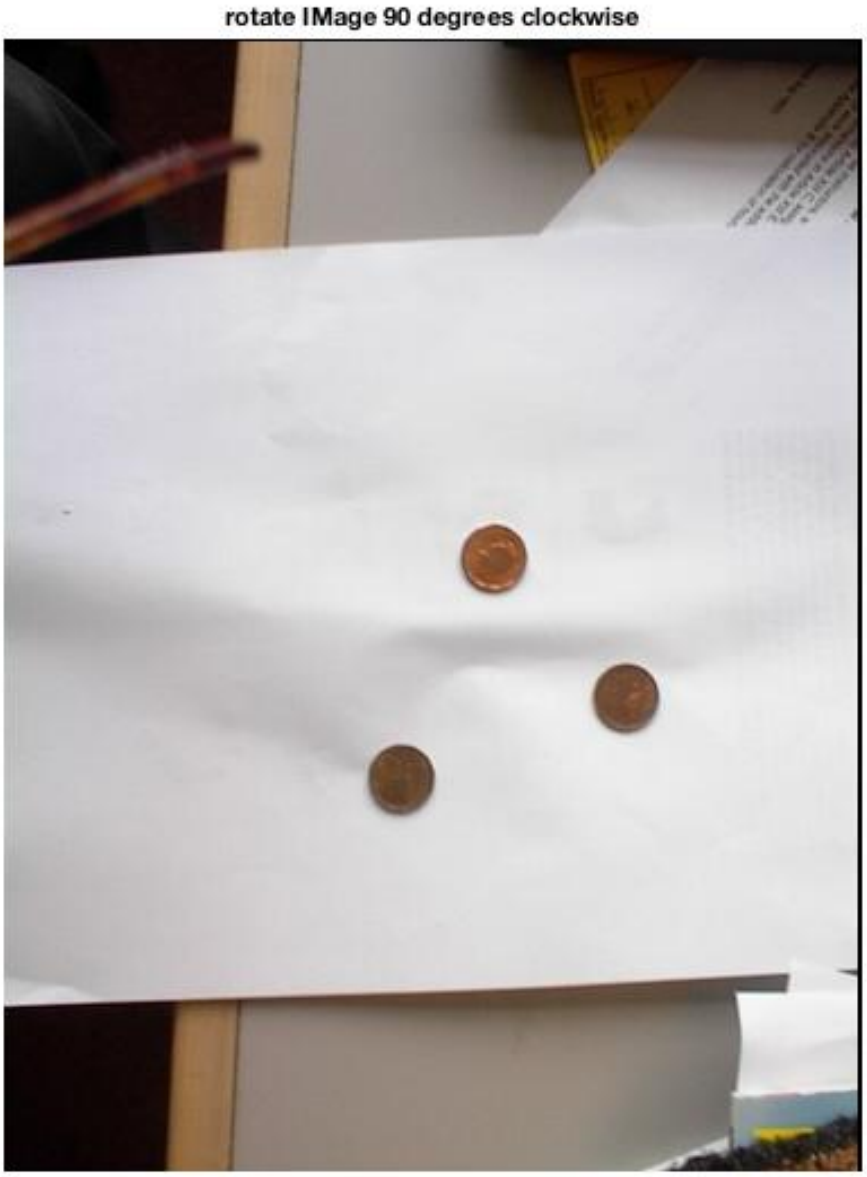
shrink:

zoom




```
>> assign1
Warning: Image is too big to fit on screen; displaying at 67%
> In images.internal.initSize (line 71)
  In imshow (line 328)
  In assign1 (line 25)
```

myrotate



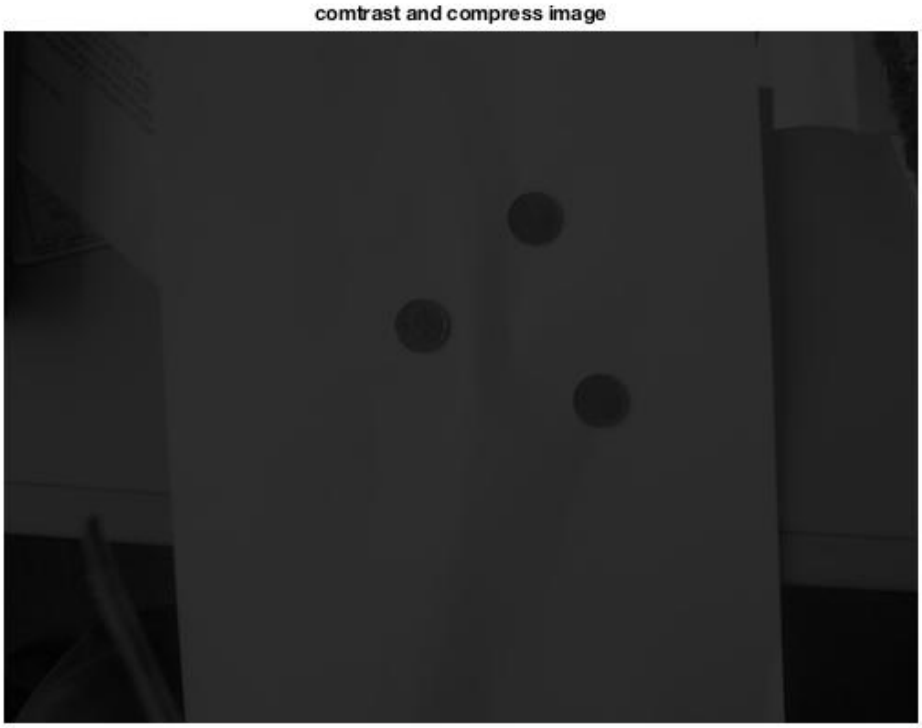
reflect


reflect IMage about the vertical axis

dim


darker Image by the fraction

contrast_compress



Test results for very small images:

```
image = zeros(6,6,3);
image(:,:,1)=magic(6);
image(:,:,2)=magic(6)';
image(:,:,3)=ones(6)/2;
image=image/max(image(:));
```

original image:

val(:,:,1) =

0.9722	0.0278	0.1667	0.7222	0.5278	0.6667
0.0833	0.8889	0.1944	0.5833	0.6389	0.6944
0.8611	0.2500	0.0556	0.6111	0.7500	0.5556
0.2222	0.7778	0.9167	0.4722	0.2778	0.4167
0.8333	0.1389	0.9444	0.3333	0.3889	0.4444
0.1111	1.0000	0.8056	0.3611	0.5000	0.3056

val(:,:,2) =

0.9722	0.0833	0.8611	0.2222	0.8333	0.1111

0.0278	0.8889	0.2500	0.7778	0.1389	1.0000
0.1667	0.1944	0.0556	0.9167	0.9444	0.8056
0.7222	0.5833	0.6111	0.4722	0.3333	0.3611
0.5278	0.6389	0.7500	0.2778	0.3889	0.5000
0.6667	0.6944	0.5556	0.4167	0.4444	0.3056

val(:,:,3) =

0.0139	0.0139	0.0139	0.0139	0.0139	0.0139
0.0139	0.0139	0.0139	0.0139	0.0139	0.0139
0.0139	0.0139	0.0139	0.0139	0.0139	0.0139
0.0139	0.0139	0.0139	0.0139	0.0139	0.0139
0.0139	0.0139	0.0139	0.0139	0.0139	0.0139
0.0139	0.0139	0.0139	0.0139	0.0139	0.0139

subimage:

val(:,:,1) =

0.8889	0.5833	0.6389
0.7778	0.4722	0.2778
0.1389	0.3333	0.3889

val(:,:,2) =

0.8889	0.7778	0.1389
0.5833	0.4722	0.3333
0.6389	0.2778	0.3889

val(:,:,3) =

0.0139	0.0139	0.0139
0.0139	0.0139	0.0139
0.0139	0.0139	0.0139

shrink:

val(:,:,1) =

0.8889	0.5833	0.6944
0.7778	0.4722	0.4167
1.0000	0.3611	0.3056

val(:,:,2) =

0.8889	0.7778	1.0000
0.5833	0.4722	0.3611
0.6944	0.4167	0.3056

val(:,:,3) =

0.0139	0.0139	0.0139
0.0139	0.0139	0.0139
0.0139	0.0139	0.0139

zoom

val(:,:,1) =

Columns 1 through 9

0.9722	0.9722	0.0278	0.0278	0.1667	0.1667	0.7222	0.7222	0.5278
0.9722	0.9722	0.0278	0.0278	0.1667	0.1667	0.7222	0.7222	0.5278
0.0833	0.0833	0.8889	0.8889	0.1944	0.1944	0.5833	0.5833	0.6389
0.0833	0.0833	0.8889	0.8889	0.1944	0.1944	0.5833	0.5833	0.6389
0.8611	0.8611	0.2500	0.2500	0.0556	0.0556	0.6111	0.6111	0.7500
0.8611	0.8611	0.2500	0.2500	0.0556	0.0556	0.6111	0.6111	0.7500
0.2222	0.2222	0.7778	0.7778	0.9167	0.9167	0.4722	0.4722	0.2778
0.2222	0.2222	0.7778	0.7778	0.9167	0.9167	0.4722	0.4722	0.2778
0.8333	0.8333	0.1389	0.1389	0.9444	0.9444	0.3333	0.3333	0.3889
0.8333	0.8333	0.1389	0.1389	0.9444	0.9444	0.3333	0.3333	0.3889
0.1111	0.1111	1.0000	1.0000	0.8056	0.8056	0.3611	0.3611	0.5000
0.1111	0.1111	1.0000	1.0000	0.8056	0.8056	0.3611	0.3611	0.5000

Columns 10 through 12

0.5278	0.6667	0.6667

0.5278	0.6667	0.6667
0.6389	0.6944	0.6944
0.6389	0.6944	0.6944
0.7500	0.5556	0.5556
0.7500	0.5556	0.5556
0.2778	0.4167	0.4167
0.2778	0.4167	0.4167
0.3889	0.4444	0.4444
0.3889	0.4444	0.4444
0.5000	0.3056	0.3056
0.5000	0.3056	0.3056

val(:,:,2) =

Columns 1 through 9

0.9722	0.9722	0.0833	0.0833	0.8611	0.8611	0.2222	0.2222	0.8333
0.9722	0.9722	0.0833	0.0833	0.8611	0.8611	0.2222	0.2222	0.8333
0.0278	0.0278	0.8889	0.8889	0.2500	0.2500	0.7778	0.7778	0.1389
0.0278	0.0278	0.8889	0.8889	0.2500	0.2500	0.7778	0.7778	0.1389
0.1667	0.1667	0.1944	0.1944	0.0556	0.0556	0.9167	0.9167	0.9444
0.1667	0.1667	0.1944	0.1944	0.0556	0.0556	0.9167	0.9167	0.9444
0.7222	0.7222	0.5833	0.5833	0.6111	0.6111	0.4722	0.4722	0.3333
0.7222	0.7222	0.5833	0.5833	0.6111	0.6111	0.4722	0.4722	0.3333
0.5278	0.5278	0.6389	0.6389	0.7500	0.7500	0.2778	0.2778	0.3889
0.5278	0.5278	0.6389	0.6389	0.7500	0.7500	0.2778	0.2778	0.3889
0.6667	0.6667	0.6944	0.6944	0.5556	0.5556	0.4167	0.4167	0.4444
0.6667	0.6667	0.6944	0.6944	0.5556	0.5556	0.4167	0.4167	0.4444

Columns 10 through 12

0.8333	0.1111	0.1111
0.8333	0.1111	0.1111
0.1389	1.0000	1.0000
0.1389	1.0000	1.0000
0.9444	0.8056	0.8056
0.9444	0.8056	0.8056
0.3333	0.3611	0.3611
0.3333	0.3611	0.3611
0.3889	0.5000	0.5000
0.3889	0.5000	0.5000
0.4444	0.3056	0.3056
0.4444	0.3056	0.3056

val(:,:,3) =

Columns 1 through 9

0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139
0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139
0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139
0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139
0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139
0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139
0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139
0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139
0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139
0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139
0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139
0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0139

Columns 10 through 12

0.0139	0.0139	0.0139
0.0139	0.0139	0.0139
0.0139	0.0139	0.0139
0.0139	0.0139	0.0139
0.0139	0.0139	0.0139
0.0139	0.0139	0.0139
0.0139	0.0139	0.0139
0.0139	0.0139	0.0139
0.0139	0.0139	0.0139
0.0139	0.0139	0.0139
0.0139	0.0139	0.0139
0.0139	0.0139	0.0139

myrotate:

val(:,:,1) =

0	0	0	0	0	0	0
0.1111	0.8333	0.2222	0.8611	0.0833	0.9722	0
1.0000	0.1389	0.7778	0.2500	0.8889	0.0278	0
0.8056	0.9444	0.9167	0.0556	0.1944	0.1667	0.0000
0.3611	0.3333	0.4722	0.6111	0.5833	0.7222	0.0000
0.5000	0.3889	0.2778	0.7500	0.6389	0.5278	0.0000
0.3056	0.4444	0.4167	0.5556	0.6944	0.6667	0.0000

val(:,:,2) =

0	0	0	0	0	0	0
0.6667	0.5278	0.7222	0.1667	0.0278	0.9722	0
0.6944	0.6389	0.5833	0.1944	0.8889	0.0833	0
0.5556	0.7500	0.6111	0.0556	0.2500	0.8611	0.0000
0.4167	0.2778	0.4722	0.9167	0.7778	0.2222	0.0000
0.4444	0.3889	0.3333	0.9444	0.1389	0.8333	0.0000
0.3056	0.5000	0.3611	0.8056	1.0000	0.1111	0.0000

val(:,:,3) =

0	0	0	0	0	0	0
0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0
0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0
0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0000
0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0000
0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0000
0.0139	0.0139	0.0139	0.0139	0.0139	0.0139	0.0000

reflect:

val(:,:,1) =

0.6667	0.5278	0.7222	0.1667	0.0278	0.9722
0.6944	0.6389	0.5833	0.1944	0.8889	0.0833
0.5556	0.7500	0.6111	0.0556	0.2500	0.8611
0.4167	0.2778	0.4722	0.9167	0.7778	0.2222
0.4444	0.3889	0.3333	0.9444	0.1389	0.8333
0.3056	0.5000	0.3611	0.8056	1.0000	0.1111

val(:,:,2) =

0.1111	0.8333	0.2222	0.8611	0.0833	0.9722
1.0000	0.1389	0.7778	0.2500	0.8889	0.0278
0.8056	0.9444	0.9167	0.0556	0.1944	0.1667
0.3611	0.3333	0.4722	0.6111	0.5833	0.7222
0.5000	0.3889	0.2778	0.7500	0.6389	0.5278
0.3056	0.4444	0.4167	0.5556	0.6944	0.6667

val(:,:,3) =

0.0139	0.0139	0.0139	0.0139	0.0139	0.0139
0.0139	0.0139	0.0139	0.0139	0.0139	0.0139
0.0139	0.0139	0.0139	0.0139	0.0139	0.0139
0.0139	0.0139	0.0139	0.0139	0.0139	0.0139
0.0139	0.0139	0.0139	0.0139	0.0139	0.0139
0.0139	0.0139	0.0139	0.0139	0.0139	0.0139

dim

val(:,:,1) =

0.2917	0.0083	0.0500	0.2167	0.1583	0.2000
0.0250	0.2667	0.0583	0.1750	0.1917	0.2083
0.2583	0.0750	0.0167	0.1833	0.2250	0.1667
0.0667	0.2333	0.2750	0.1417	0.0833	0.1250
0.2500	0.0417	0.2833	0.1000	0.1167	0.1333
0.0333	0.3000	0.2417	0.1083	0.1500	0.0917

val(:,:,2) =

0.2917	0.0250	0.2583	0.0667	0.2500	0.0333
0.0083	0.2667	0.0750	0.2333	0.0417	0.3000
0.0500	0.0583	0.0167	0.2750	0.2833	0.2417
0.2167	0.1750	0.1833	0.1417	0.1000	0.1083
0.1583	0.1917	0.2250	0.0833	0.1167	0.1500
0.2000	0.2083	0.1667	0.1250	0.1333	0.0917

val(:,:,3) =

0.0042	0.0042	0.0042	0.0042	0.0042	0.0042
0.0042	0.0042	0.0042	0.0042	0.0042	0.0042
0.0042	0.0042	0.0042	0.0042	0.0042	0.0042
0.0042	0.0042	0.0042	0.0042	0.0042	0.0042
0.0042	0.0042	0.0042	0.0042	0.0042	0.0042
0.0042	0.0042	0.0042	0.0042	0.0042	0.0042

contrast_compress

0.00819559891985999	0.00224781541748721	0.00570219665803905	0.00564349896701015	0.00689100084952868	0.00497129821356399
0.00224781541748721	0.00785674201318386	0.00415219483153669	0.00691580725377227	0.00505817404052226	0.00765170377855330
0.00570219665803905	0.00415219483153669	0.00231080647446584	0.00728240861931242	0.00766941901878332	0.00690484153158211
0.00564349896701015	0.00691580725377227	0.00728240861931242	0.00585184185125612	0.00479312814149849	0.00535009242016860
0.00689100084952868	0.00505817404052226	0.00766941901878332	0.00479312814149849	0.00535321670244080	0.00584950814921950
0.00819559891985999	0.00224781541748721	0.00570219665803905	0.00564349896701015	0.00689100084952868	0.00497129821356399