Document for : **MATLAB Central , File Exchange**

Author : KHMOU Youssef 25, July 2012

Peak Signal-to-Noise Ratio for RGB Images

The PSNR (Peak Signal-to-Noise Ratio) **[1], [2]** is a tool for measuring the distortion between the original and the recovered signals, "recovered" refers to either denoising operation, decompression, reconstruction or any other engineering manipulation. It is evaluated in logarithmic decibel scale .

We consider two 2D signals : X(n,m) is the original signal and Y(n,m) is the recovered signal, the PSNR is evaluated as follows :

1. we calculte the Mean Square Error between the two signals :

$$MSE(X,Y)=\frac{1}{m.n}\sum_{i=1}^{i=n}\sum_{j=1}^{j=m}[X(i,j)-Y(i,j)]^2 \qquad (1)$$

2. the psnr is given by :

$$PSNR(X,Y)=10.\log_{10}\left(\frac{max(X)^2}{MSE(X,Y)}\right) \qquad (2)$$

With $max(X)$ is the maximum possible value of the signal X : For type unsigned integer <uint8> , n= 8, $max(X)=2^n-1=255$.

Many programmed PSNR functions, for grayscale images, are in the File exchange [3],[4],[5],[6],[7] .

We are interested in using the PSNR to measure the difference for RGB images, for that purpose we consider two RGB images : X(n,m,p) the original image and Y(n,m,p) the noisy or the denoised image, (**p= 3**) the Mean Square Error is given by :

$$MSE(X,Y)=\frac{1}{(p.n.m)}\sum_{i=1}^{i=n}\sum_{j=1}^{j=m}\sum_{k=1}^{k=p}[X(i,j,k)-Y(i,j,k)]^2 \qquad (3)$$

We compute the "dynamic" of the two signals; theirs maximum values

$$m1=argmax(X(n,m,p)) \qquad (4)$$

$$m2=argmax(Y(n,m,p)) \qquad (5)$$

Then we choose the maximum of the two values : $D=max([m1\,m2])$ (6)

Note: other versions take 1 or 255 ...instead of max(max(X),max(Y)).

finally the PSNR is given by : $PSNR(X,Y)=10.\log\left(\frac{D^2}{MSE(X,Y)}\right)$ (7)

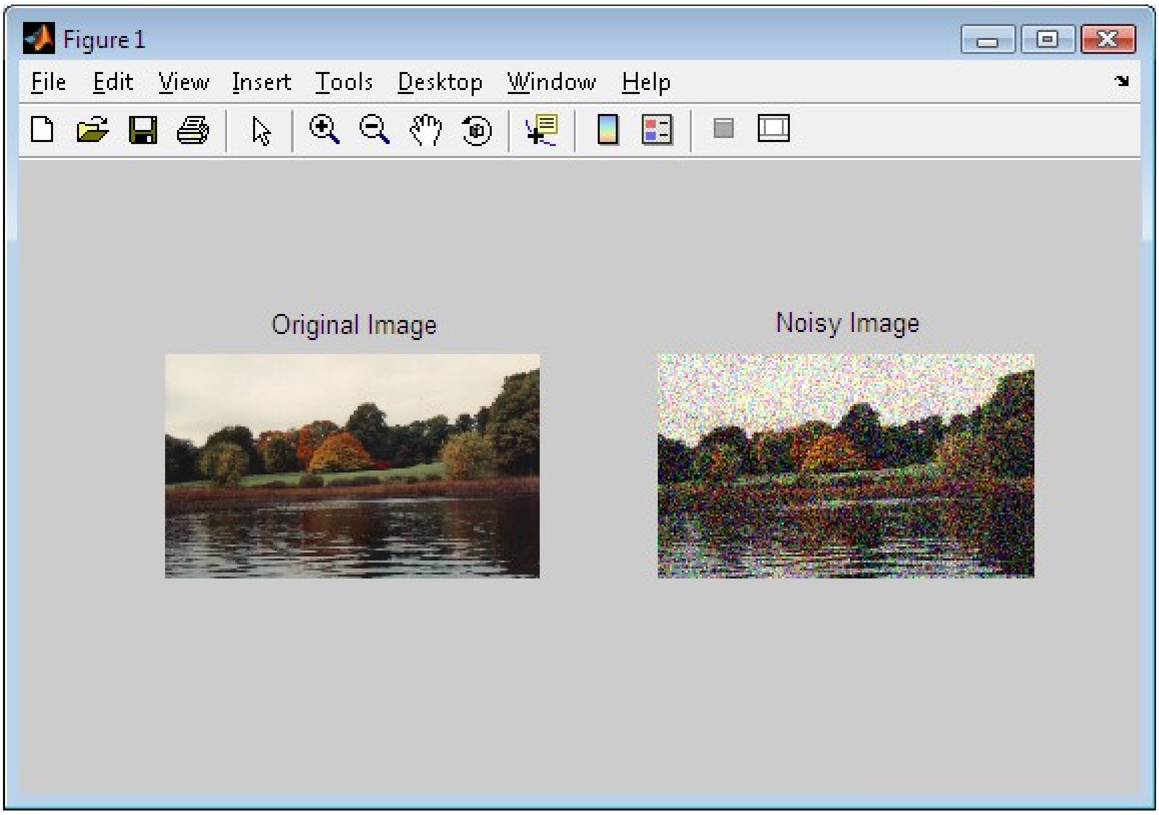
Application in MATLAB :

1.First test :

```
>>A=rand(260,260,3);
>>B=0.9*A;
>>PSNR_RGB(A,B)
ans =

   57.0273
```

2.Second test :

```
>>I=imread('autumn.tif');
>> whos I
  Name       Size              Bytes  Class    Attributes

  I        206x345x3         213210  uint8

>> I=im2double(I);
>>noisyI=imnoise(I,'Gaussian',0,0.03);
>>subplot(1,2,1), imshow(I), title(' Original Image')
>> subplot(1,2,2), imshow(noisyI), title(' Noisy Image')
```



```
>> PSNR_RGB(I,noisyI)

ans =

   37.6402
```

Functions:

```
================================================================================
```

PSNR_RGB.m :

```
function y=PSNR_RGB(X,Y)

% Y= PSNR_RGB(X,Y)
% Computes the Peak Signal to Noise Ratio for two RGB images
% Class input : double [0,1] ,
% july ,25, 2012
% KHMOU Youssef



if size(X)~=size(Y)
    error('The images must have the same size');
end

%if ~isa(X,'double')
%   X=double(X)./255.00;
%end
%if  ~isa(Y,'double')
%    Y=double(Y)./255.00;
%end

% begin
d1=max(X(:));
d2=max(Y(:));
d=max(d1,d2);
sigma=mean2((X-Y).^2);

y=10*log((d.^2)./sigma);
================================================================================
```

PSNR.m

```
function y=PSNR(noisyImage,restoredImage)

% Compute the PSNR of two gray scale image
% Traditional progarmming using loops
% Class input : [0,1]
% july, 25 , 2012
% KHMOU Youssef

N=size(noisyImage);
if length(N)> 2
    error('Input must be grayscale image');
end
if size(noisyImage)~=size(restoredImage)
    error('The images must have the same size');
end

%if ~isa(noisyImage,'double')
%   noisyImage=double(noisyImage)./255.00;
```

```
%end
%if  ~isa(restoredImage,'double')
%    restoredImage=double(restoredImage)./255.00;
%end

% begin

d1=max(noisyImage(:));
d2=max(restoredImage(:));
d=max(d1,d2);

MSE=0;
for i=1:N(1)
    for j=1:N(2)
        if isnan(noisyImage(i,j)) || isinf(restoredImage(i,j))...
                || isnan(restoredImage(i,j)) || isinf(noisyImage(i,j))
            continue;
        end
        MSE=MSE+((abs(noisyImage(i,j)-restoredImage(i,j))).^2);
    end
end

MSE=MSE./(N(1)*N(2));

y=10*log10((d.^2) /MSE)
```

==

References :

[1] : http://en.wikipedia.org/wiki/Peak_signal-to-noise_ratio

[2] : PirahanSiah, F, "Adaptive image segmentation based on peak signal-to-noise ratio for a license plate recognition system", Computer applications and industrial Electronics, (ICCAIE), 2010 International Conference.

[3] : http://www.mathworks.com/matlabcentral/fileexchange/37326-psnr-image-processing

[4] : http://www.mathworks.com/matlabcentral/fileexchange/36291-psnr-for-2d-images

[5] : http://www.mathworks.com/matlabcentral/fileexchange/34204-peak-signal-to-noise-ratio

[6] : http://www.mathworks.com/matlabcentral/fileexchange/27862-psnr-calculator

[7] : http://www.mathworks.com/matlabcentral/fileexchange/135-psnr