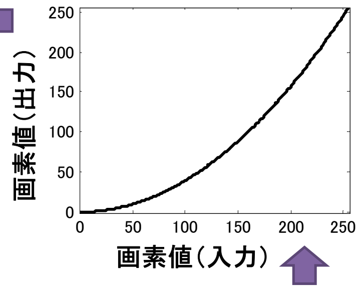
“トーンマッピング”とは？

暗い　‘Lena’

tone mapped image



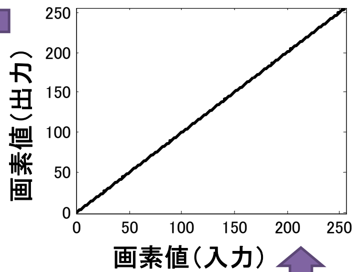
$$y = x^{1/\gamma}$$
$$\gamma = 1/2$$

original image

All Rights Reserved / M.Iwahashi / Nagaoka University of Technology

元の　‘Lena’

tone mapped image



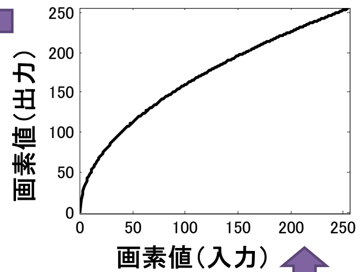
$$y = x^{1/\gamma}$$
$$\gamma = 1$$

original image

All Rights Reserved / M.Iwahashi / Nagaoka University of Technology

明るい‘Lena’

tone mapped image



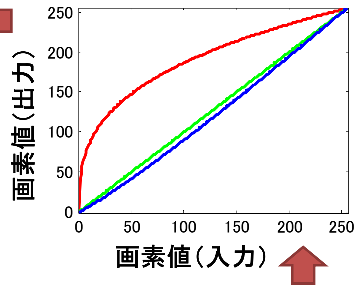
$$y = x^{1/\gamma}$$
$$\gamma = 2$$

original image

All Rights Reserved / M.Iwahashi / Nagaoka University of Technology

赤い　‘Lena’

tone mapped
image



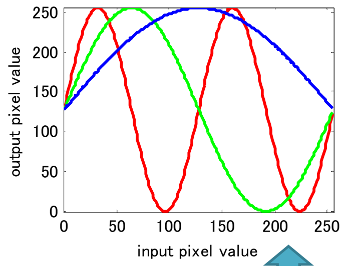
$$y = x^{1/\gamma}$$

$$\begin{cases} \gamma_R = 3 \\ \gamma_G = 1 \\ \gamma_B = 0.9 \end{cases}$$

original
image

ソラリゼーション

tone mapped
image



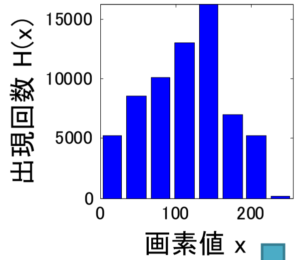
original
image

ヒストグラム

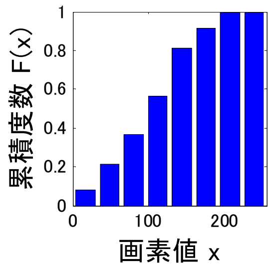
確率密度と累積度数

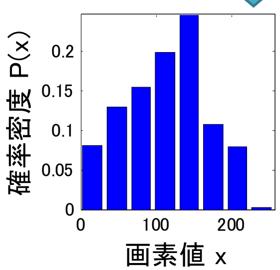
ヒストグラム （8 個のビン）

原画像



ヒストグラム
H(x)





確率密度
P(x)

確率密度 と 累積度数

ヒストグラム
H(x)
x=Min,･･･,Max

$$\sum_{x=Min}^{Max} H(x) = \text{画素の総数}$$

画素の総数で割る

確率密度
P(x)
x=Min,･･･,Max

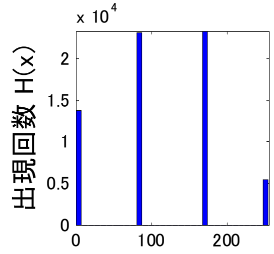
$$\sum_{x=Min}^{Max} P(x) = 1$$

累積度数

$$F(x) = \sum_{u=Min}^{x} P(u)$$

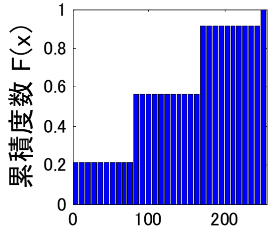
4 階調の ‘Lena’

原画像

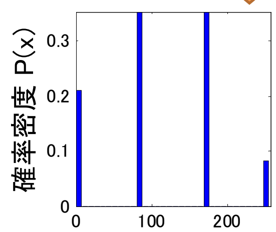


画素値 x

ヒストグラム
H(x)



画素値 x

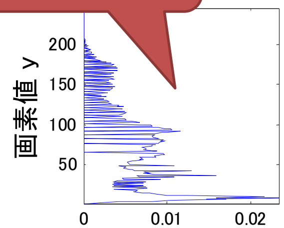


画素値 x

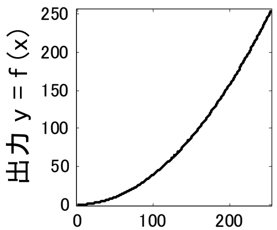
確率密度
P(x)

暗い ‘Lena’

f(x) に依って
どう変わるか



確率密度 Q(y)

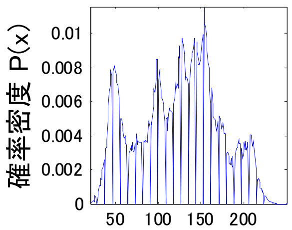


入力 x

$$y = x^{1/\gamma}$$

$$\gamma = 1/2$$

画像 y



画素値 x

確率密度
P(x)

トーンマッピングによる確率密度の変化

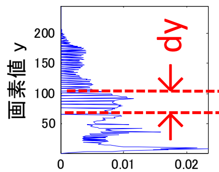
確率密度

Q(y)

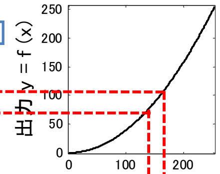
を f(x)

と P(x)

で表せ



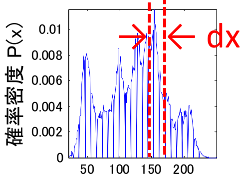
確率密度 Q(y)



入力 x

$$y = f(x)$$

画像 y



画素値 x

確率密度
P(x)

ヒント $\boxed{Q(y)\,dy = P(x)\,dx}$

トーンマッピングによる確率密度の変化

$$Q(y)dy = P(x)dx$$

より、

$$Q(y) \quad = P(x)\frac{dx}{dy} \quad = \frac{P(x)}{\dfrac{dy}{dx}}$$

ここで、

$$y = f(x), \quad \frac{dy}{dx} = f'(x)$$

なので、

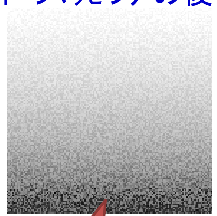
$$Q(y) = \boxed{\frac{P(x)}{f'(x)}}$$

確率密度 Q(y) を
f(x) と P(x) で表せた

トーンマッピングが
誤差
に及ぼす影響

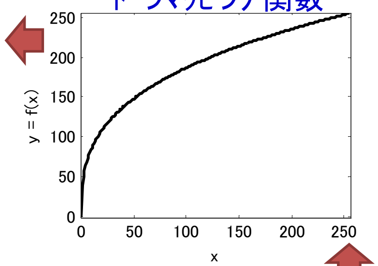
テスト画像　＋　誤差

トーンマッピングの後



暗い
領域

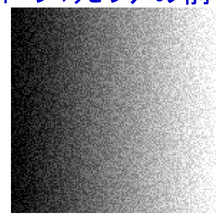
トーンマッピング関数



$$y = x^{1/\gamma}$$

$$\gamma = 3$$

トーンマッピングの前



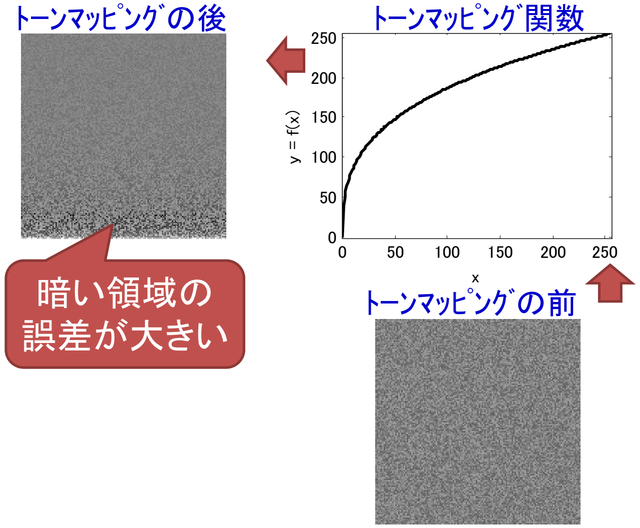
信号＋誤差

x + e(x)

MATLAB

```
clear all; close all;

% Fn1='C:¥Users¥iwa¥岩橋¥ソフトウェア¥標準画像¥mono¥';
% Fn2='lena.bmp';
% X=imread(strcat(Fn1,Fn2),'bmp');

X=zeros(256,256);
for i=1:256;for j=1:256;
   X(i,j)=double(j-1);
end;end;

x=0:255;
X=double(X);
Xd=X+(rand(size(X))-0.5)*64;

G=3;
y =x .^(1/G);
Y =X .^(1/G);
Yd=Xd.^(1/G);

Mx=max(y);
y =y /Mx*255;
Y =Y /Mx*255; Y =flipud(Y');
Yd=Yd/Mx*255; Yd=flipud(Yd');

% [Hx Bx]=hist(Xd(:),256);
% [Hy By]=hist(Yd(:),256);   Hy(1)=0;
% Hx=Hx/sum(Hx);
% Hy=Hy/sum(Hy);
% subplot(2,2,1);plot(Hy,By,'.');
% xlabel('P(y)'); ylabel('y'); axis('tight');
% subplot(2,2,4);plot(Bx,Hx,'.');
% xlabel('x'); ylabel('P(x)'); axis('tight');

subplot(2,2,1);imshow(uint8(Yd));
subplot(2,2,4);imshow(uint8(Xd));

subplot(2,2,2); plot(x,y,'K','LineWidth',1.5); hold on;
axis([-1 256 -1 256]);
xlabel('x');
ylabel('y = f(x)');
```

誤差のみ



暗い領域の誤差が大きい

$y = x^{1/\gamma}$

$\gamma = 3$

誤差

e(x)

数理的な裏付け

$$f(x+e(x)) \approx f(x) + \frac{df(x)}{dx} \cdot e(x) \qquad \leftarrow \quad x > e(x) \text{ を仮定}$$

なので

$$\Delta y = f(x+e(x)) - f(x)$$

$$\approx f(x) + \frac{df(x)}{dx} \cdot e(x) - f(x)$$

$$= \boxed{}\, e(x)$$

接線の傾き

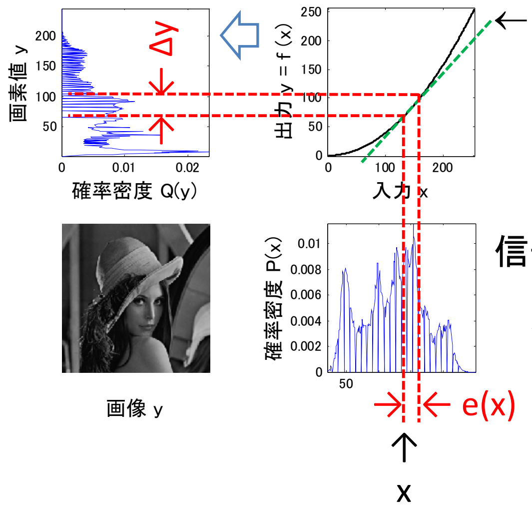
出力の誤差 Δy を
f(x) と e(x) で表せた

まとめ

出力の誤差

$$\Delta y = \boxed{}\, e(x)$$

接線の傾き



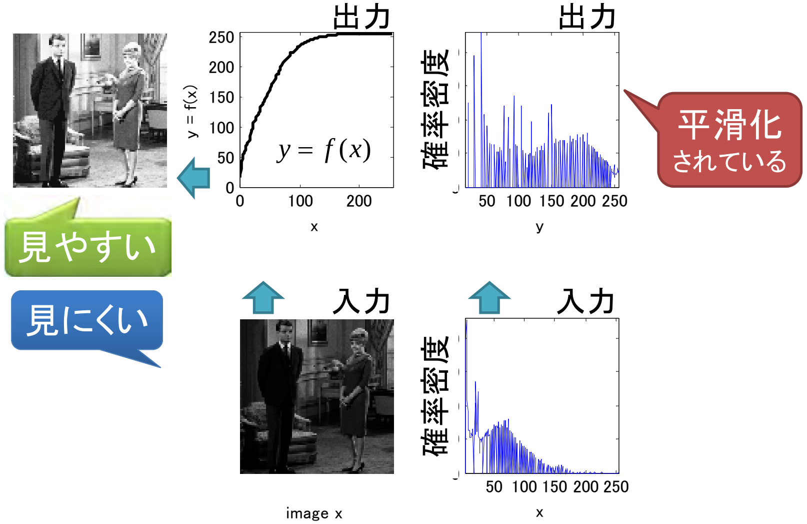
$\leftarrow \dfrac{df(x)}{dx}$

信号＋誤差

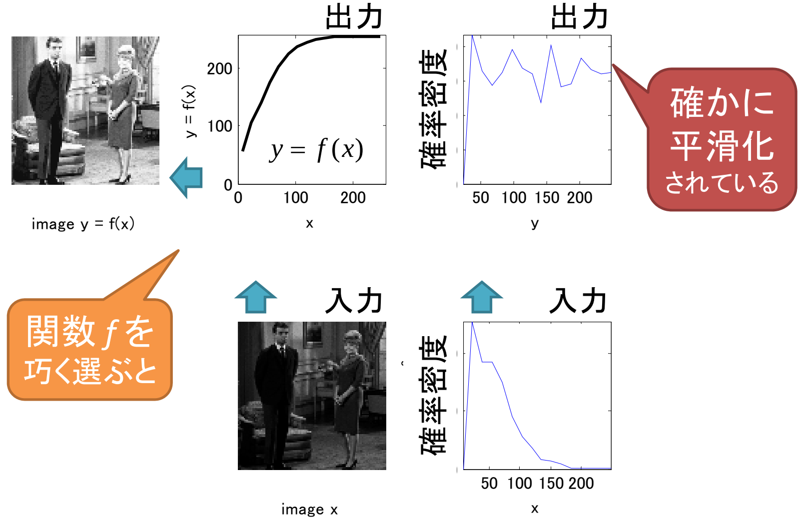
x + e(x)

ヒストグラムを
平滑化する

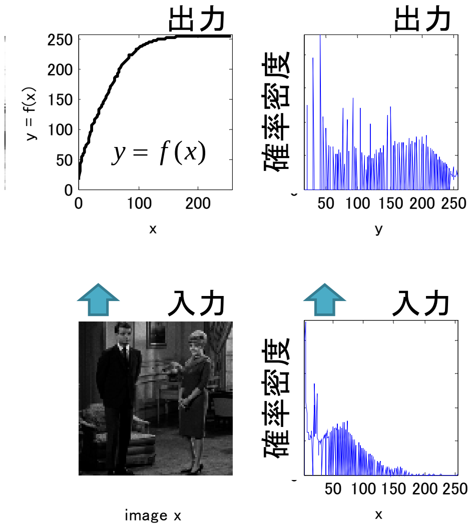
ヒストグラムの平滑化



ビンを　16個　に減らすと…



トーンマッピング関数の決定法



確率密度　が　一定となる

$Q(y)$

様に f(x) を

決めよ。

$y = f(x)$

確率密度

$P(x)$

トーンマッピング関数の決定法

$$Q(y) = \frac{P(x)}{f'(x)} = Const.$$

$\leftarrow$ $Q(y)dy = P(x)dx$

$dy/dx = f'(x)$

より

なので、

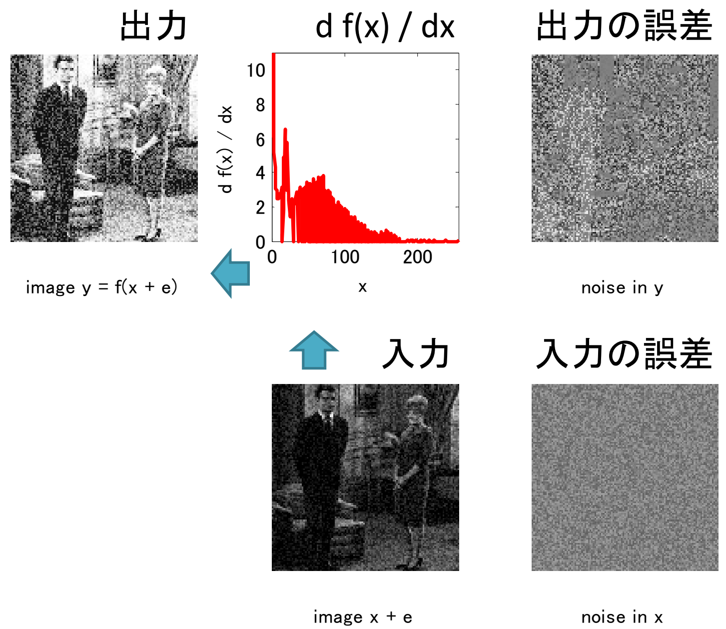
$$f'(x) = \frac{P(x)}{Const.}$$

従って、

$$f(x) = \boxed{\int_{\min}^{x} \frac{P(u)}{Const.} du}$$

$\leftarrow$ 結果は累積度数！

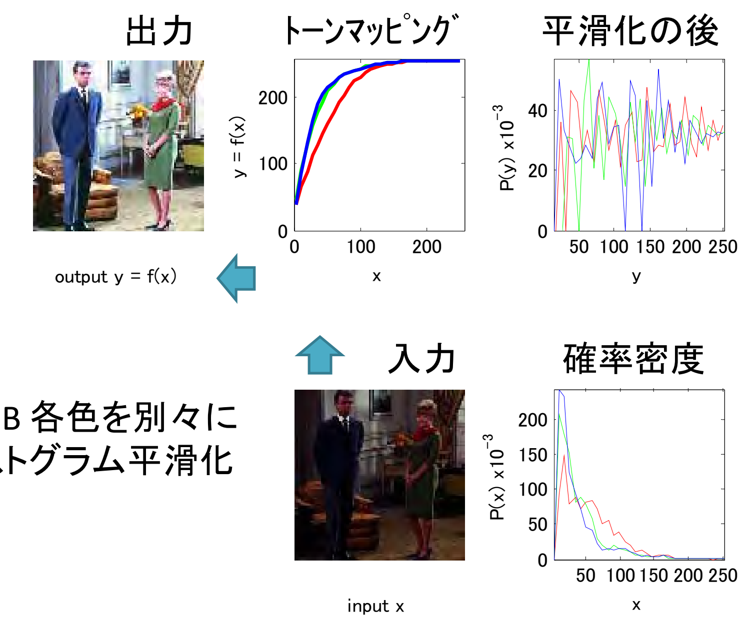
確率密度 Q(y) が一定となる
様に f(x) が決定された



MATLAB

```
clear all; close all;
%
SwTm=1;
for SwIn=1:8;

  switch(SwIn);
     case 1; Fn2='couple.bmp';
     case 2; Fn2='girl.bmp';
     case 3; Fn2='lax.bmp';
     case 4; Fn2='airplane.bmp';
     case 5; Fn2='bridge.bmp';
     case 6; Fn2='barbara.bmp';
     case 7; Fn2='lena.bmp';
     case 8; Fn2='test.bmp';
  end;

  Fn1='C:¥Users¥iwa¥岩橋¥ソフトウェア¥標準画像¥mono¥';
  Xs=imread(strcat(Fn1,Fn2),'bmp');
  Xs=NrmImg(double(Xs))*255;        % signal
  Xd=Xs+(rand(size(Xs))-0.5)*64;    % signal + noise

  switch(SwTm);
   case 1;
     [Hx Bx]=hist(Xs(:),256); Hx=Hx/sum(Hx);
     f=0; for i=1:256; f=f+Hx(i); F(i)=f; end;
  end;
  F =F *255;

Ys=F(uint8(Xs)+1);
Yd=F(uint8(Xd)+1);

[Hy By]=hist(Ys(:),2^8); Hy(1)=0; Hy=Hy/sum(Hy); Hy=Hy*1000;
[Hx Bx]=hist(Xs(:),2^8); Hx(1)=0; Hx=Hx/sum(Hx); Hx=Hx*1000;

figure(1);%--------------------- signal
subplot(2,3,3);plot(By,Hy,'-');
xlabel('y'); ylabel('P(y) x10^{-3}'); axis('tight'); axis('square');
subplot(2,3,6);plot(Bx,Hx,'-');
xlabel('x'); ylabel('P(x) x10^{-3}'); axis('tight'); axis('square');

subplot(2,3,1);imshow(uint8(Ys)); xlabel('image y = f(x)');
subplot(2,3,5);imshow(uint8(Xs)); xlabel('image x');

x=floor(Bx);
y=F(x+1);
subplot(2,3,2); plot(x,y,'K','LineWidth',1.5); axis('square');
axis([-1 256 0 256]); xlabel('x'); ylabel('y = f(x)');
%
% figure(2);%--------------------- signal + noise
% subplot(2,3,3);imshow(uint8(Yd-Ys+128));
% xlabel('noise in y');
% subplot(2,3,6);imshow(uint8(Xd-Xs+128));
% xlabel('noise in x');
%
% subplot(2,3,1);imshow(uint8(Yd)); xlabel('image y = f(x + e)');
% subplot(2,3,5);imshow(uint8(Xd)); xlabel('image x + e');
%
% subplot(2,3,2); plot(x,Df(x,y),'-R','LineWidth',1.5);
% hold on; axis('square');
% axis('tight'); xlabel('x'); ylabel('d f(x) / dx');

pause; close all;
end;
```

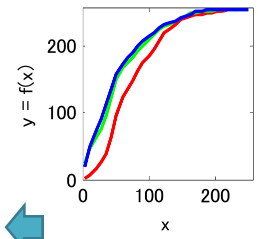

カラー画像の各色 {R,G,B} それぞれを ヒストグラム平滑化

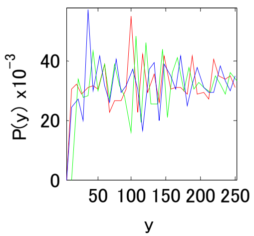


R,G,B 各色を別々に
ヒストグラム平滑化

出力　トーンマッピング　平滑化の後

output y = f(x)

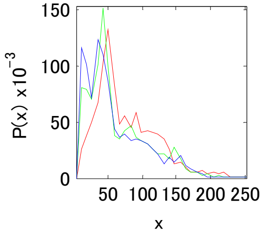




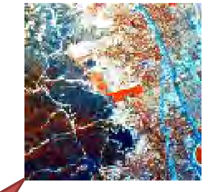
入力　確率密度

R,G,B 各色を別々に
ヒストグラム平滑化

input x

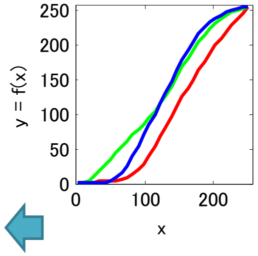


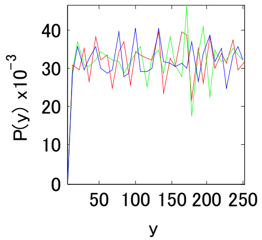
出力　トーンマッピング　平滑化の後



模様が
明瞭化

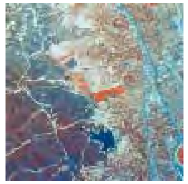
output y = f(x)



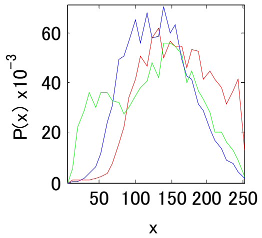


入力　確率密度

R,G,B 各色を別々に
ヒストグラム平滑化



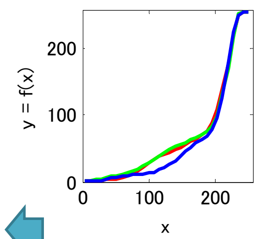
input x

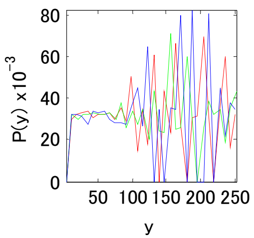


出力　トーンマッピング　平滑化の後

模様が
明瞭化

output y = f(x)

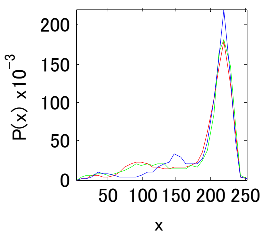




入力　確率密度

R,G,B 各色を別々に
ヒストグラム平滑化

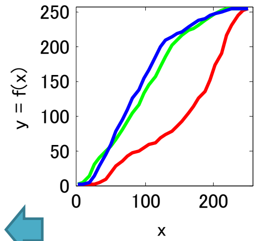
input x

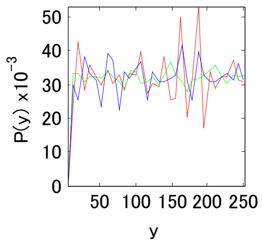


出力　トーンマッピング　平滑化の後

色調は
不自然

output y = f(x)

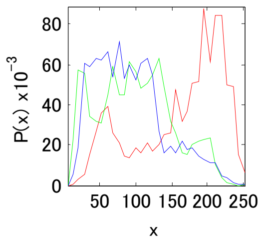


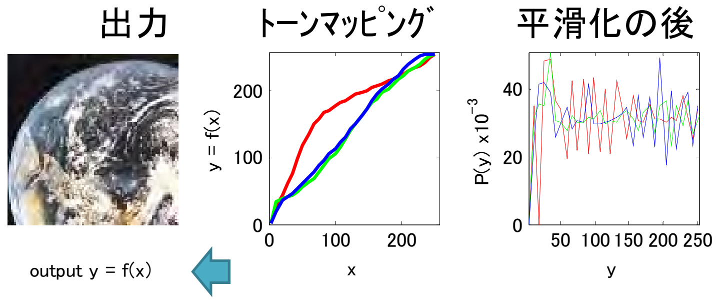


入力　確率密度

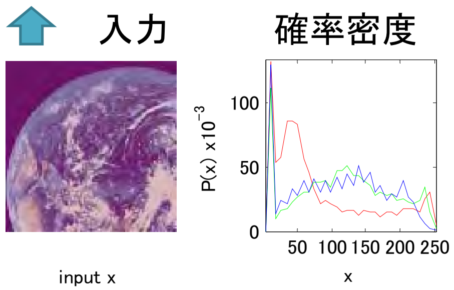
R,G,B 各色を別々に
ヒストグラム平滑化

input x





R,G,B 各色を別々に
ヒストグラム平滑化



MATLAB

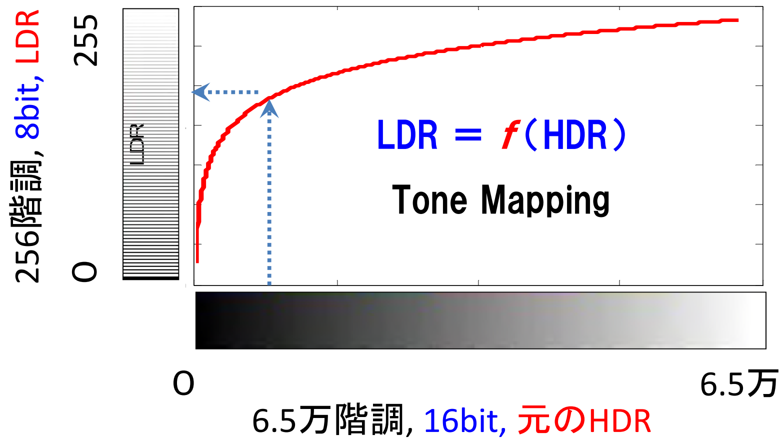
```
clear all; close all;
%
SwTm=1;
for SwIn=1:1;
%   switch(SwIn);
%      case 1; Fn2='couple.bmp'; case 2; Fn2='girl.bmp';  case 3; Fn2='lax.bmp';  case 4; Fn2='airplane.bmp';
%      case 5; Fn2='bridge.bmp'; case 6; Fn2='barbara.bmp'; case 7; Fn2='lena.bmp';  case 8; Fn2='test.bmp';
%   end;
  switch(SwIn);
     case 1; Fn2='couple.bmp'; case 2; Fn2='girl.bmp'; case 3; Fn2='aerial.bmp';
     case 4; Fn2='airplane.bmp'; case 5; Fn2='parrots.bmp'; case 6; Fn2='mandrill.bmp';
     case 7; Fn2='lena.bmp'; case 8; Fn2='earth.bmp'; case 9; Fn2='balloon.bmp';
     case 10; Fn2='milkdrop.bmp'; case 11; Fn2='sailboat.bmp'; case 12; Fn2='pepper.bmp';
     case 13; Fn2='tree_log.bmp';
  end;
  Fn1='C:¥Users¥iwa¥岩橋¥研究室¥ソフトウェア¥標準画像¥color¥';
%   Fn1='C:¥Users¥iwa¥岩橋¥ソフトウェア¥標準画像¥color¥';
%   Fn1='C:¥Users¥iwa¥岩橋¥ソフトウェア¥標準画像¥mono¥';
  X0=imread(strcat(Fn1,Fn2),'bmp');

for c=1:3;
  Xs=X0(:,:,c);
  Xs=NrmImg(double(Xs))*255;        % signal
  switch(SwTm);
   case 1;
     [Hx Bx]=hist(Xs(:),256); Hx=Hx/sum(Hx);
     f=0; for i=1:256; f=f+Hx(i); F(i)=f; end;
  end;
  F =F *255;
  Ys=F(uint8(Xs)+1);
  [Hy By]=hist(Ys(:),2^5); Hy(1)=0; Hy=Hy/sum(Hy); Hy=Hy*1000;
  [Hx Bx]=hist(Xs(:),2^5); Hx(1)=0; Hx=Hx/sum(Hx); Hx=Hx*1000;

  figure(1);%---------------------- signal
  subplot(2,3,3);
  if(c==1); plot(By,Hy,'-R'); end;
  if(c==2); plot(By,Hy,'-G'); end;
  if(c==3); plot(By,Hy,'-B'); end;
  hold on;
  xlabel('y'); ylabel('P(y) x10^{-3}'); axis('tight'); axis('square');
  subplot(2,3,6);
  if(c==1); plot(Bx,Hx,'-R'); end;
  if(c==2); plot(Bx,Hx,'-G'); end;
  if(c==3); plot(Bx,Hx,'-B'); end;
  hold on;
  xlabel('x'); ylabel('P(x) x10^{-3}'); axis('tight'); axis('square');

  x=floor(Bx);
  y=F(x+1);
  subplot(2,3,2);
  if(c==1); plot(x,y,'R','LineWidth',1.5); end;
  if(c==2); plot(x,y,'G','LineWidth',1.5); end;
  if(c==3); plot(x,y,'B','LineWidth',1.5); end;
  hold on;
  axis('square'); axis([-1 256 0 256]); xlabel('x'); ylabel('y = f(x)');

  Y0(:,:,c)=Ys;
end;

  subplot(2,3,1); imshow(uint8(Y0)); xlabel('output y = f(x)');
  subplot(2,3,5); imshow(uint8(X0)); xlabel('input x');

  pause; close all; clear X0 Y0;
end;
```


ハイダイナミックレンジ

（HDR）画像を

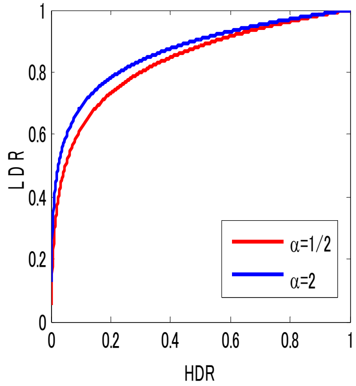
トーンマッピングする

HDR 画像から LDR 画像へ

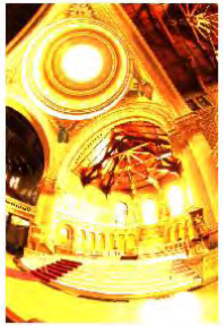


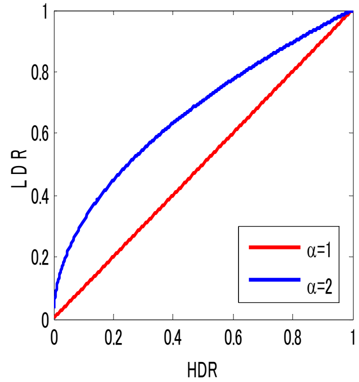
対数関数　$\log(1+\alpha x)$



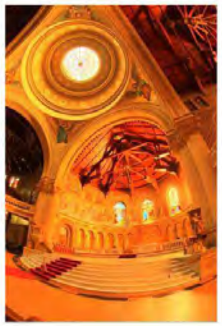
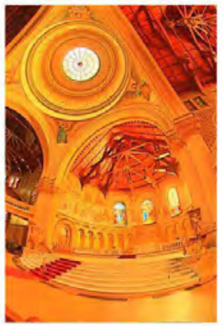
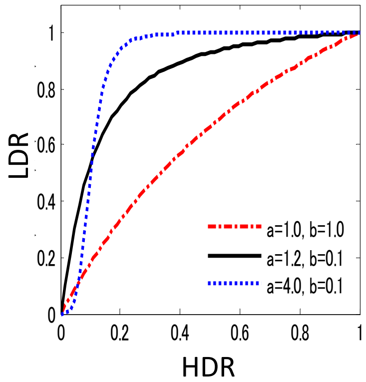


べき関数　$x^{1/\alpha}$





Hill 関数　$\dfrac{x^a}{x^a+b^a}$

カラー画像のトーンマッピング


R,G,B,それぞれを
別々にﾄｰﾝﾏｯﾋﾟﾝｸﾞ


色褪せない
ﾄｰﾝﾏｯﾋﾟﾝｸﾞ

色褪せないための工夫

HDRの赤,緑,青 $\rightarrow \begin{bmatrix} x_{H,R} & x_{H,G} & x_{H,B} \end{bmatrix}$

LDRの赤,緑,青 $\rightarrow \begin{bmatrix} x_{L,R} & x_{L,G} & x_{L,B} \end{bmatrix}$

$$x_{L,c} = f(x_{H,c})$$

$$c \in \{R, G, B\}$$

R,G,B,それぞれを
別々にトーンマッピング

$$x_{L,c} = \frac{f(x_{H,Y})}{x_{H,Y}} x_{H,c}$$

HDR白黒の輝度 $x_{H,\mathrm{Y}}$

色褪せない
トーンマッピング

Hill 関数 & フィルタ

＆低域通過フィルタ　　Hill 関数　　＆高域強調フィルタ

Hill 関数＆フィルタ