



## 第5章 图像增强 Part2

主讲教师 张涛

电子信息与电气工程学院



# 主要内容

5.3 基于照度反射模型的图像增强

5.4 基于模糊技术的图像增强

5.5 基于伪彩色处理的图像增强

5.6 其他图像增强方法



## 5.3 基于照度反射模型的图像增强

### 5.3.1 基于同态滤波的增强

### 5.3.2 基于Retinex理论的增强



## 5.3.1 基于同态滤波的增强

### 基于照度反射模型的图像增强

#### ■ 背景

若物体受到照度明暗不匀的时候，图像上对应照度暗的部分，其细节就较难辨别。

#### ■ 主要目的

消除不均匀照度的影响，增强图像细节。

#### ■ 基本原理

根据图像的照度-反射模型，对原始图像  $f(x,y)$  中的 反射分量  $r(x,y)$  进行扩展，光照分量  $i(x,y)$  进行压缩，以获得所要求的增强图像



## 5.3.1 基于同态滤波的增强

### 基于照度反射模型的图像增强

#### ■ 算法步骤

##### □ 对图像函数 $f(x,y)$ 取对数变换

$$\begin{aligned} z(x,y) &= \ln[f(x,y)] = \ln[i(x,y) \cdot r(x,y)] \\ &= \ln[i(x,y)] + \ln[r(x,y)] \end{aligned}$$

##### □ 傅立叶变换

$$\begin{aligned} Z(u,v) &= DFT\{\ln[i(x,y)]\} + DFT\{\ln[r(x,y)]\} \\ &= I(u,v) + R(u,v) \end{aligned}$$

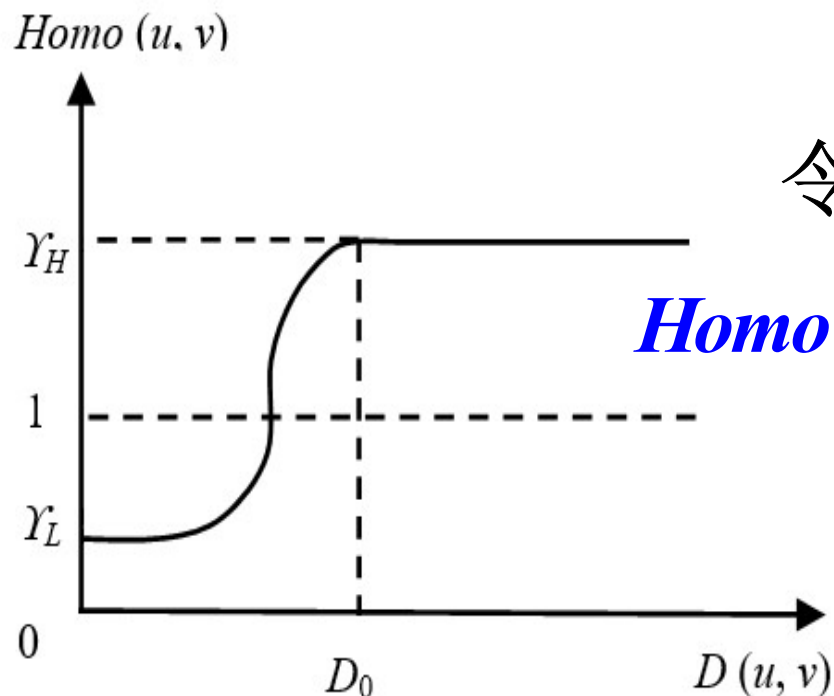


## 5.3.1 基于同态滤波的增强

### 基于照度反射模型的图像增强

□ 同态滤波---衰减低频段，增强高频段

$$\begin{aligned} S(u, v) &= Homo(u, v) \cdot Z(u, v) \\ &= Homo(u, v) \cdot I(u, v) + Homo(u, v) \cdot R(u, v) \end{aligned}$$



令  $High(u, v)$  为高通滤波函数

$$Homo(u, v) = (\gamma_H - \gamma_L) * High(u, v) + \gamma_L$$

$$\gamma_H > 1, \quad 0 < \gamma_L < 1$$



## 5.3.1 基于同态滤波的增强

### 基于照度反射模型的图像增强

□ 求傅立叶逆变换

$$\begin{aligned} s(x, y) &= DFT^{-1} \{ Hmo(u, v) \cdot I(u, v) \} + DFT^{-1} \{ Hmo(u, v) \cdot R(u, v) \} \\ &= i'(x, y) + r'(x, y) \end{aligned}$$

□ 进行指数变换，得到输出图像

$$\begin{aligned} g(x, y) &= e^{\{s(x, y)\}} = e^{\{i'(x, y) + r'(x, y)\}} \\ &= i_0(x, y) + r_0(x, y) \end{aligned}$$



## 5.3.1 基于同态滤波的增强

### 基于照度反射模型的图像增强

#### ■ 例程—代码

```
Image=double(rgb2gray(imread('gugong1.jpg')));  
logI=log(Image+1);  
sigma=1.414;    filtersize=[7 7];  
lowfilter=fspecial('gaussian',filtersize,sigma);  
highfilter=zeros(filtersize);  
highpara=1;    lowpara=0.4;  
highfilter(ceil(filtersize(1,1)/2),ceil(filtersize(1,2)/2))=1;  
highfilter=highpara*highfilter-  
            (highpara-lowpara)*lowfilter;
```





## 5.3.1 基于同态滤波的增强

### 基于照度反射模型的图像增强

```
highpart=imfilter(logI,highfilter,'replicate','conv');  
NewImage=exp(highpart);  
top=max(NewImage(:));  
bottom=min(NewImage(:));  
NewImage=(NewImage-bottom)/(top-bottom);  
NewImage=1.5.*(NewImage);  
figure,imshow((NewImage));title('基于同态滤波的  
增强图像');
```

## 5.3.1 基于同态滤波的增强

### 基于照度反射模型的图像增强

#### ■ 例程—效果



(a) 原图



(b) 同态滤波效果



## 5.3.2 基于Retinex理论的增强

### 基于照度反射模型的图像增强

Retinex理论的基本原理模型是以人类视觉系统为出发点发展而来的一种基于颜色恒常性的色彩理论，该理论认为：

- 人眼对物体颜色的感知与物体表面的反射性质有着密切关系，即反射率低的物体看上去较暗，反射率高的物体看上去是较亮。
- 人眼对物体色彩的感知具有一致性，不受光照变化的影响。



## 5.3.2 基于Retinex理论的增强

### 基于照度反射模型的图像增强

#### ■ 基于Retinex理论的增强的基本原理:

根据图像的照度-反射模型，通过从原始图像中估计光照分量，然后设法去除（或降低）光照分量，获得物体的反射性质，从而获得物体的本来面貌。



## 5.3.2 基于Retinex理论的增强

### 基于照度反射模型的图像增强

#### ■ 中心环绕Retinex方法

估计光照分量 $i'_c(x,y)$ 的计算式:

$$i'_c(x,y) = F(x,y) * f_c(x,y), C \in \{R, G, B\}$$

其中,  $F(x,y)$ 是中心环绕函数, 定义:

$$F(x,y) = K \cdot e^{-\frac{x^2+y^2}{\sigma^2}}$$

$$\iint F(x,y) dx dy = 1$$

$\sigma$ 为标准差, 表示高斯环绕函数的尺度常数, 决定了卷积核的作用范围



## 5.3.2 基于Retinex理论的增强

### 基于照度反射模型的图像增强

#### (1) 单尺度Retinex增强

##### ■ 算法步骤

□ 计算第C颜色通道的光照分量估计值 $i'_c(x,y)$

□ 对数变换  $\ln[f_c(x,y)] = \ln[i_c(x,y) \cdot r_c(x,y)]$   
 $= \ln[i_c(x,y)] + \ln[r_c(x,y)]$

□ 计算反射分量，获得单尺度Retinex增强图像

$$R_c(x,y) = \ln[r'_c(x,y)] = \ln[f_c(x,y)] - \ln[i'_c(x,y)]$$
$$= \ln[f_c(x,y)] - \ln[F(x,y) * f_c(x,y)]$$





## 5.3.2 基于Retinex理论的增强

### 基于照度反射模型的图像增强

#### ■ 例程—程序

```
Image=(imread('gugong1.jpg'));  
imshow(Image); title('原始图像');  
[height,width,c]=size(Image);  
RI=double(Image(:,:,1));    GI=double(Image(:,:,2));  
BI=double(Image(:,:,3));  
sigma=100; filtersize=[height,width];  
gaussfilter=fspecial('gaussian',filtersize,sigma);  
Rlow=imfilter(RI,gaussfilter,'replicate','conv');  
Glow=imfilter(GI,gaussfilter,'replicate','conv');  
Blow=imfilter(BI,gaussfilter,'replicate','conv');
```



## 5.3.2 基于Retinex理论的增强

### 基于照度反射模型的图像增强

```
minRL=min(min(Rlow));      minGL=min(min(Glow));  
minBL=min(min(Blow));      maxRL=max(max(Rlow));  
maxGL=max(max(Glow)); maxBL=max(max(Blow));  
RLi=(Rlow-minRL)/(maxRL-minRL);  
GLi=(Glow-minGL)/(maxGL-minGL);  
BLi=(Blow-minBL)/(maxBL-minBL);  
Li=cat(3,RLi,GLi,BLi);  
figure;imshow(Li);title('估计光照分量');  
Rhigh=log(RI./Rlow+1);      Ghigh=log(GI./Glow+1)  
Bhigh=log(BI./Blow+1);  
SSRI=cat(3,Rhigh,Ghigh,Bhigh);  
figure;imshow((SSRI));title('单尺度Retinex增强');
```



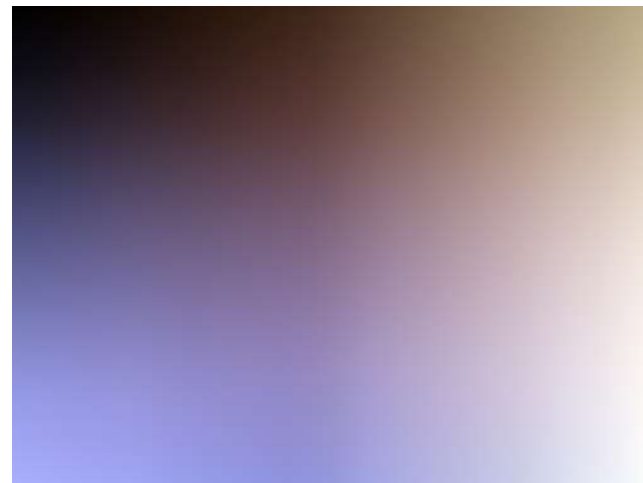
## 5.3.2 基于Retinex理论的增强

### 基于照度反射模型的图像增强

#### ■ 例程—效果



(a) 原图



(b) 估计光照分量



(c) 单尺度Retinex增强



## 5.3.2 基于Retinex理论的增强

### 基于照度反射模型的图像增强

#### (2) 多尺度Retinex增强

##### ■ 算法步骤

- 设置不同尺度 $\sigma_n, n=1,2,\dots$
- 计算不同尺度的中心环绕函数 $F_n(x,y)$
- 求不同尺度的Retinex增强输出

$$R_{n_c}(x,y) = \left\{ \ln[f_c(x,y)] - \ln[F_n(x,y) * f_c(x,y)] \right\}$$

$R_{n_c}(x,y)$  是第 $C$ 通道第 $n$ 个尺度Retinex增强输出



## 5.3.2 基于Retinex理论的增强

### 基于照度反射模型的图像增强

□ 对多个不同尺度Retinex增强输出加权平均

$$R_{M_C}(x, y) = \left[ \sum_{n=1}^N w_n R_{n_C}(x, y) \right] \cdot \gamma_C(x, y)$$

$w_n$  是给不同尺度  $\sigma_n$  分配的权重因子。

$\gamma_C$  是第  $C$  颜色通道的色彩恢复系数。

$$\gamma_C(x, y) = \eta \cdot \ln \left[ \beta \cdot \frac{f_C(x, y)}{\sum_{C \in (R, G, B)} f_C(x, y)} \right]$$

$\eta$  为增益常数， $\beta$  为非线性强度的控制因子



## 5.3.2 基于Retinex理论的增强

### 基于照度反射模型的图像增强

#### ■ 例程—程序

```
Image=(imread('gugong1.jpg'));  
imshow(Image); title('原始图像');  
[height,width,c]=size(Image);  
RI=double(Image(:,:,1));  GI=double(Image(:,:,2));  
BI=double(Image(:,:,3));  
beta=0.4;  alpha=125;  
CR=beta*(log(alpha*(RI+1))-log(RI+GI+BI+1));  
CG=beta*(log(alpha*(GI+1))-log(RI+GI+BI+1));  
CB=beta*(log(alpha*(BI+1))-log(RI+GI+BI+1));  
Rhigh=zeros(height,width);  
Ghigh=zeros(height,width);
```



## 5.3.2 基于Retinex理论的增强

### 基于照度反射模型的图像增强

```
Bhigh=zeros(height,width);  
sigma=[15 80 250];    filtersize=[height,width];  
for i=1:3  
    gaussfilter=fspecial('gaussian',filtersize,sigma(i));  
    Rlow=imfilter(RI,gaussfilter,'replicate','conv');  
    Glow=imfilter(GI,gaussfilter,'replicate','conv');  
    Blow=imfilter(BI,gaussfilter,'replicate','conv');  
    Rhigh=1/3*(CR.*log(RI./Rlow+1)+Rhigh);  
    Ghigh=1/3*(CG.*log(GI./Glow+1)+Ghigh);  
    Bhigh=1/3*(CB.*log(BI./Blow+1)+Bhigh);  
end  
MSRI=cat(3,Rhigh,Ghigh,Bhigh);  
figure;imshow(MSRI);title('多尺度Retinex增强');
```



## 5.3.2 基于Retinex理论的增强

### 基于照度反射模型的图像增强

#### ■ 例程—效果



(a) 原图



(b) 多尺度Retinex增强图像



## 5.4 基于模糊技术的图像增强

令 $U$ 为元素（对象）集， $u$ 表示 $U$ 的一类元素，即 $U=\{u\}$ ，则该集合称为论域 $U$ 。

论域 $U$ 到 $[0,1]$  闭区间的任一映射 $\mu_A$ ：

$$\mu_A : U \rightarrow [0,1] \quad u \rightarrow \mu_A(u)$$

都确定 $U$ 的一个模糊集合 $A$ ， $\mu_A$ 称为模糊集合的隶属函数。

$\mu_A(u)$ 称为 $u$ 对于 $A$ 的隶属度，取值范围为 $[0,1]$



$$A = \{u, \mu_A(u) \mid u \in U\}$$



## 5.4 基于模糊技术的图像增强

### 5.4.1 图像的模糊特征平面

### 5.4.2 图像的模糊增强





## 5.4.1 图像的模糊特征平面

### 基于模糊技术的图像增强

#### (1) 定义

$$X = \begin{bmatrix} \mu_{11}(x_{11}) & \mu_{12}(x_{12}) & \text{L} & \mu_{1N}(x_{1N}) \\ \mu_{21}(x_{21}) & \mu_{22}(x_{22}) & \text{L} & \mu_{2N}(x_{2N}) \\ \text{M} & \text{M} & \text{M} & \text{M} \\ \mu_{M1}(x_{M1}) & \mu_{M2}(x_{M2}) & \text{L} & \mu_{MN}(x_{MN}) \end{bmatrix}$$

或

$$X = \bigcup_{m=1}^M \bigcup_{n=1}^N \mu_{mn}(x_{mn})$$

$\mu_{mn}(x_{mn})$ 表示图像中像素灰度 $x_{mn}$ 相对于某些特定灰度级 $x$ 的隶属度，且 $0 \leq \mu_{mn} \leq 1$ 。

定义隶属度函数 $\mu_{mn}(x_{mn})$ ，用于表征图像模糊特征，实现图像从空间灰度域变换到模糊域。



## 5.4.2 图像的模糊增强

### 基于模糊技术的图像增强

#### (2) 算法步骤

- 将图像从空间灰度域变换到模糊域

$$\mu_{mn} = T(x_{mn}) = \left[ 1 + \frac{x_{\max} - x_{mn}}{F_d} \right]^{-F_e}$$

- ✓  $F_e$  为指数模糊因子。一般情况下， $F_e$  取为 2。
- ✓  $F_d$  为分数模糊因子，定义为

$$F_d = \frac{x_{\max} - x_c}{2^{1/F_e} - 1}$$

- ✓ 其中， $x_c$  为渡越点，其取值需要满足：

$$\mu_c = T(x_c) = 0.5 \quad \text{且} \quad x_c \in X$$



## 5.4.2 图像的模糊增强

### 基于模糊技术的图像增强

- 在模糊域，对模糊特征进行一定的增强

$$\begin{cases} \mu'_{mn} = I_r(\mu_{mn}) = \begin{cases} 2\mu_{mn}^2, & 0 \leq \mu_{mn} < 0.5 \\ 1 - 2(1 - \mu_{mn})^2, & 0.5 \leq \mu_{mn} < 1 \end{cases} \\ I_r(\mu_{mn}) = I_1(I_{r-1}(\mu_{mn})) \end{cases}$$

- ✓  $\mu'_{mn}$  为增强后的模糊域像素灰度值；
- ✓  $r$  为正整数，表示迭代次数；

- 逆变换，得到新的模糊增强后的输出图像

$$z_{mn} = I^{-1}(\mu'_{mn}) = x_{\max} - F_d \left[ (\mu'_{mn})^{\frac{1}{F_e}} - 1 \right]$$



## 5.4.2 图像的模糊增强

### 基于模糊技术的图像增强

#### (3) 例程

##### ■ 程序

```
Image=imread('Beautiful.jpg');  
[height width]=size(Image);  
Image=double(Image);    xmax=max(max(Image));  
xc=mean2(Image);  
Fe=2;    Fd=(xmax-xc)/(2^(1/Fe)-1);  
u=(1+(xmax-Image)/Fd).^(-Fe);  
times=2; %设置迭代次数  
for k=1:times  
    for i=1:height  
        for j=1:width
```



## 5.4.2 图像的模糊增强

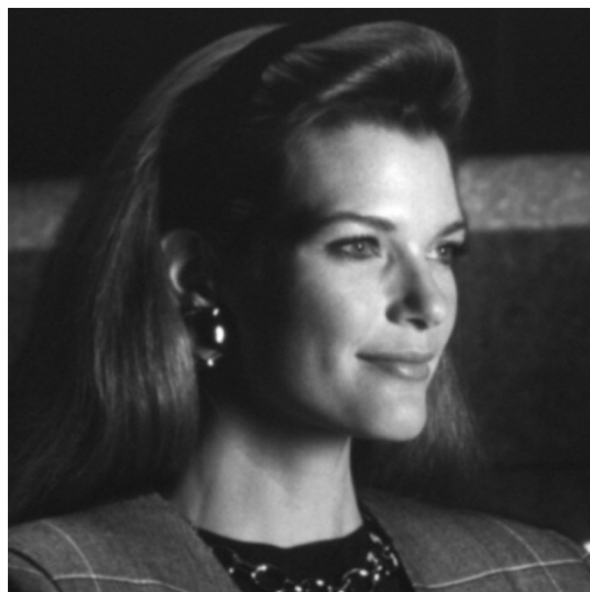
### 基于模糊技术的图像增强

```
if u(i,j)<0.5
    u(i,j)=2*u(i,j)^2;
else
    u(i,j)=1-2*(1-u(i,j))^2;
end
end
end
end
NewImage=xmax-Fd.*(u.^(-1/Fe)-1);
figure;imshow(uint8(NewImage));title('基于模糊技术的增强');
```

## 5.4.2 图像的模糊增强

### 基于模糊技术的图像增强

#### ■ 效果



(a) 原始图像



(b) 模糊增强后图像



## 5.5 基于伪彩色处理的图像增强

### ■ 为什么需要彩色增强处理图像？

- 色彩中含有很多信息，使从一个场景中识别和抽取目标变得简易些。
- 人眼对色彩敏感：  
可以辨别上千种不同颜色；  
但只能辨别十几到二十几种灰度
- 将灰度图像变成彩色图像，能够有效提高图像可鉴别性。
- 伪彩色增强就是一种灰度到彩色的映射技术。



## 5.5 基于伪彩色处理的图像增强

5.5.1 密度分割法

5.5.2 空间域灰度级-彩色变换

5.5.3 频域伪彩色增强

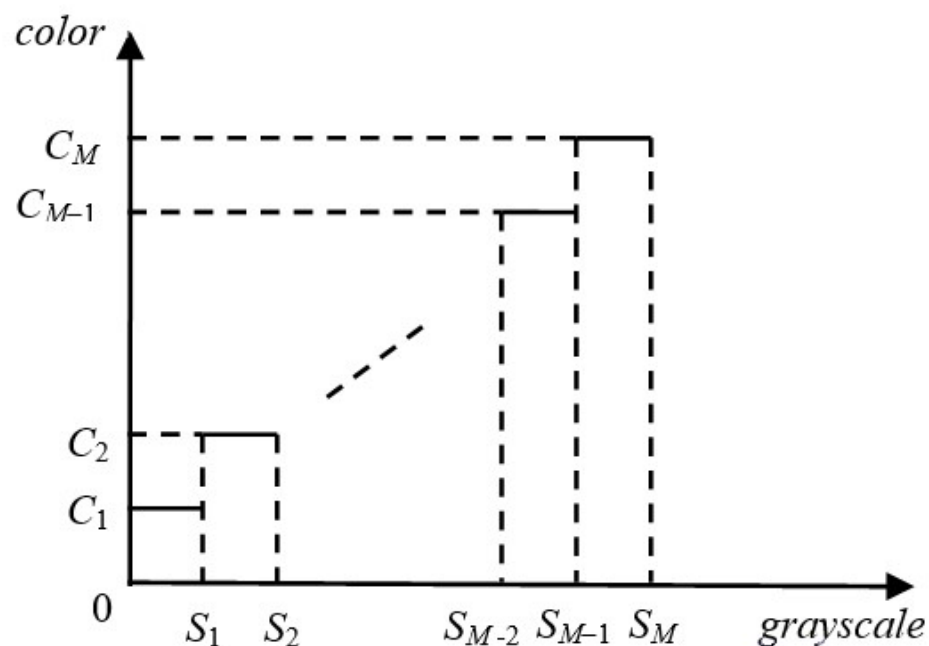
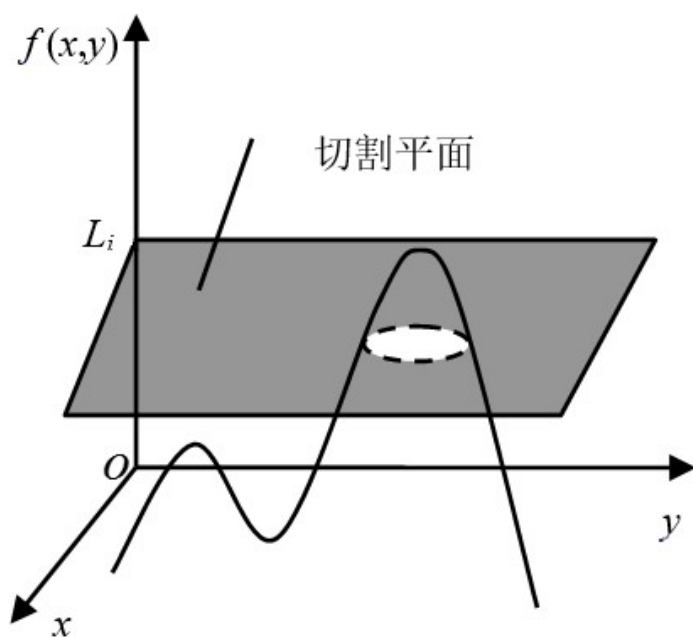


## 5.5.1 密度分割法

### 基于伪彩色处理的图像增强

#### (1) 原理

- 一幅图像描述为三维函数  $(x, y, f(x, y))$
- 在  $f(x, y) = L_i, i=1, 2, \dots, M$  处放置平行  $xoy$  坐标面的  $M$  个切割平面
- 分割  $M$  个不同灰度级区域，分配  $M$  种不同颜色





## 5.5.1 密度分割法

### 基于伪彩色处理的图像增强

#### (2) 例程

##### ■ 程序

```
Image = double(imread('yaogan1.bmp'));  
[height,width]=size(Image);  
NewImage=zeros(height,width,3);  
for i=1:height  
    for j=1:width  
        if Image(i,j)<52    %灰度级位于[0,52)  
            NewImage(i,j,1)=16;  
            NewImage(i,j,2)=25;  
            NewImage(i,j,3)=64;
```



## 5.5.1 密度分割法

### 基于伪彩色处理的图像增强

```
elseif Image(i,j)<92    %灰度级位于[52,92)
    NewImage(i,j,1)=27;
    NewImage(i,j,2)=45;
    NewImage(i,j,3)=125;
elseif Image(i,j)<115  %灰度级位于[92,115)
    NewImage(i,j,1)=101;
    NewImage(i,j,2)=146;
    NewImage(i,j,3)=79;
elseif Image(i,j)<170  %灰度级位于[115,170)
    NewImage(i,j,1)=115;
    NewImage(i,j,2)=156;
    NewImage(i,j,3)=142;
```



## 5.5.1 密度分割法

### 基于伪彩色处理的图像增强

```
Else    %灰度级位于[179,255)
```

```
    NewImage(i,j,1)=213;
```

```
    NewImage(i,j,2)=222;
```

```
    NewImage(i,j,3)=159;
```

```
end
```

```
end
```

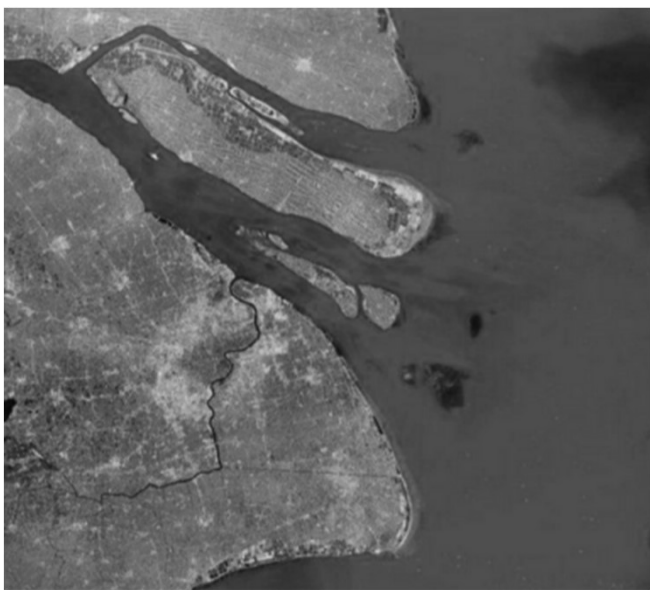
```
end
```

```
figure;imshow(uint8(NewImage));title(' 密 度 分  
割的伪彩色增强');
```

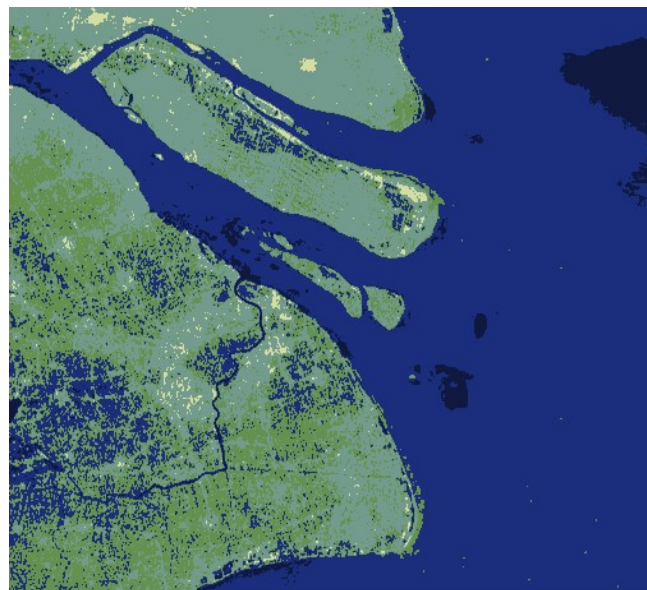
## 5.5.1 密度分割法

### 基于伪彩色处理的图像增强

#### ■ 效果



(a) 原始图像



(b) 伪彩色增强后图像

缺点是：变换出的彩色信息有限，且变换后的图像通常会显得不够细腻。

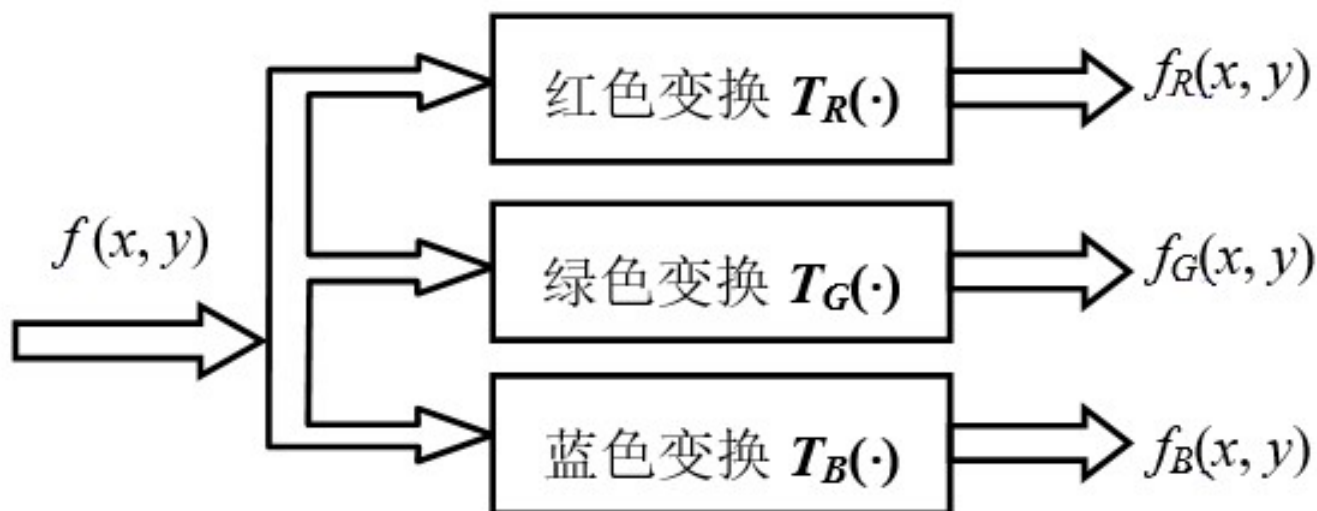


## 5.5.2空间域灰度级—彩色变换

### 基于伪彩色处理的图像增强

#### (1) 原理

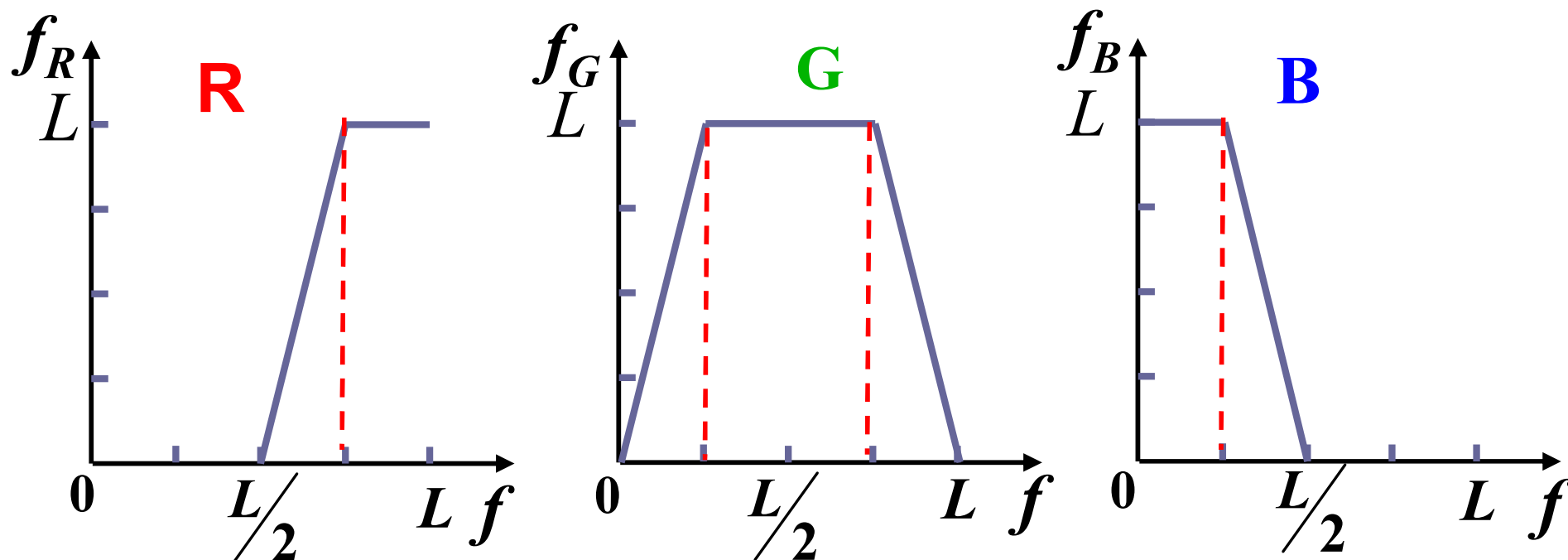
- 将图像 $f$ 送入具有不同变换特性的红、绿、蓝3个变换器；
- 产生的输出 $f_R$ 、 $f_G$ 、 $f_B$ 作为彩色图像的红、绿、蓝三个色彩分量；
- 合成一幅彩色图像。





## 5.5.2空间域灰度级—彩色变换

### 基于伪彩色处理的图像增强 (2) 常用的典型的灰度级-彩色变换



当  $f < L/4$  时,

$$R=0$$

$$G=4f$$

$$B=L$$

当  $L/4 < f < L/2$  时,

$$R=0$$

$$G=L$$

$$B=2L-4f$$

当  $L/2 < f < 3L/4$  时,

$$R=4f-2L$$

$$G=L$$

$$B=0$$

当  $f > 3L/4$  时,

$$R=L$$

$$G=4L-4f$$

$$B=0$$

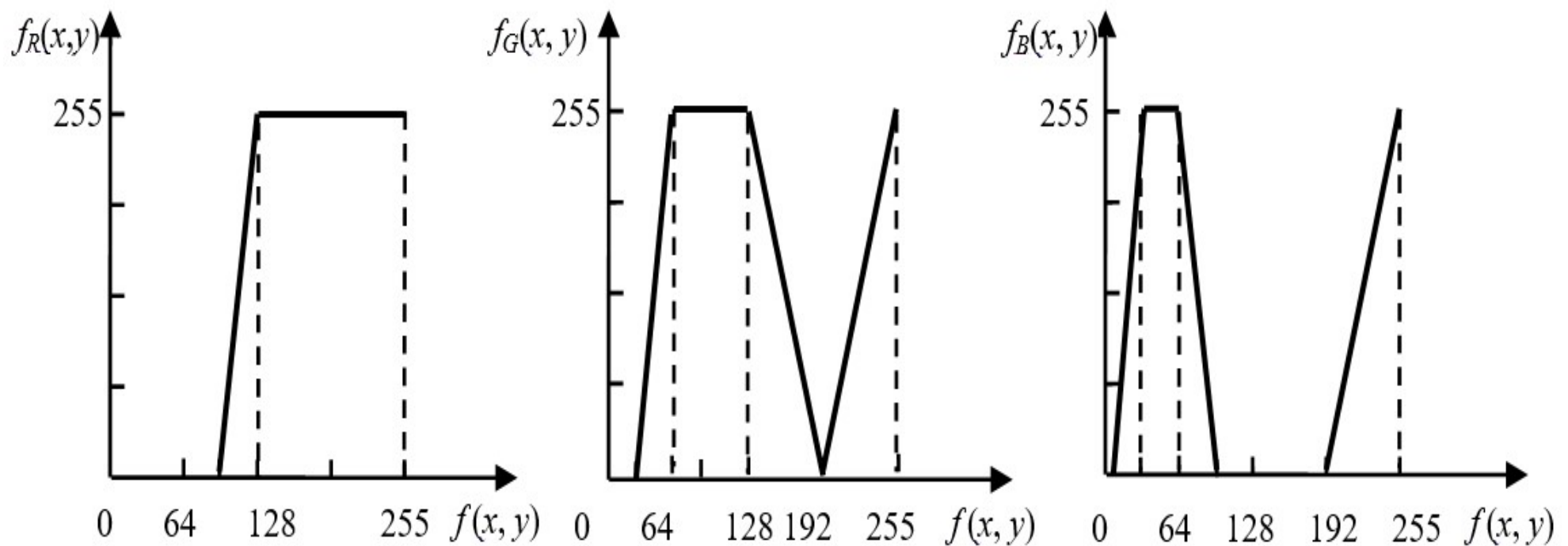




## 5.5.2 空间域灰度级—彩色变换

基于伪彩色处理的图像增强

### (3) 彩虹编码的灰度级-彩色变换



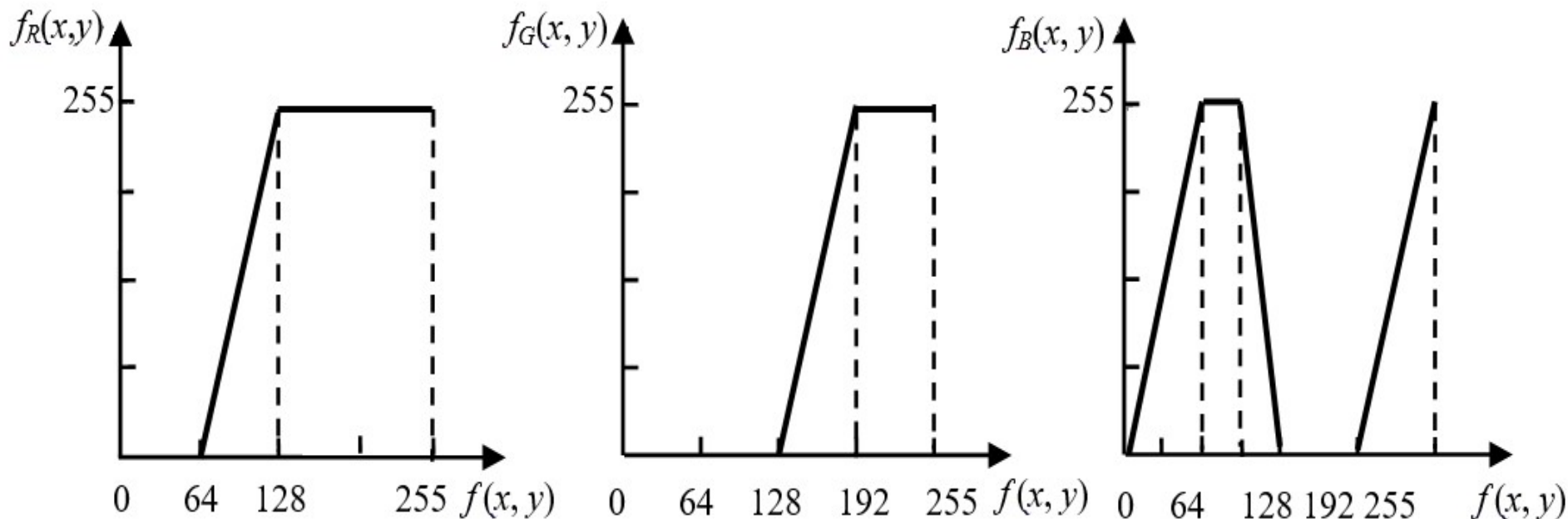




## 5.5.2空间域灰度级—彩色变换

基于伪彩色处理的图像增强

### (4) 热金属编码的灰度级-彩色变换





## 5.5.2空间域灰度级—彩色变换

### 基于伪彩色处理的图像增强

#### (5) 例程

##### ■ 程序

```
Image=double(imread('Brain.jpg'));  
[height,width]=size(Image);  
NewImage=zeros(height,width,3);  
L=255;  
for i=1:height  
    for j=1:width  
        if Image(i,j)<=L/4  
            NewImage(i,j,1)=0;
```



## 5.5.2空间域灰度级—彩色变换

### 基于伪彩色处理的图像增强

```
NewImage(i,j,2)=4*Image(i,j);  
NewImage(i,j,3)=L;  
else if Image(i,j)<=L/2  
    NewImage(i,j,1)=0;  
    NewImage(i,j,2)=L;  
    NewImage(i,j,3)=-4*Image(i,j)+2*L;  
else if Image(i,j)<=3*L/4  
    NewImage(i,j,1)=4*Image(i,j)-2*L;  
    NewImage(i,j,2)=L;  
    NewImage(i,j,3)=0;
```



## 5.5.2空间域灰度级—彩色变换

### 基于伪彩色处理的图像增强

**Else**

**NewImage(i,j,1)=L;**

**NewImage(i,j,2)=-4\*Image(i,j)+4\*L;**

**NewImage(i,j,3)=0;**

**end**

**end**

**end**

**end**

**end**

**figure;imshow(uint8(NewImage));title('灰度级-  
彩色变换伪彩色增强图像');**

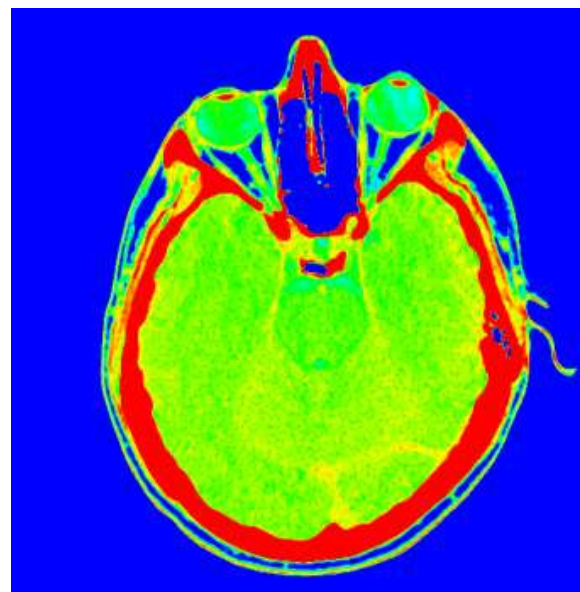
## 5.5.2空间域灰度级—彩色变换

### 基于伪彩色处理的图像增强

#### ■ 效果



(a) 原始图像



b) 伪彩色增强后图像



## 5.5.3 频域伪彩色增强

### 基于伪彩色处理的图像增强

#### (1) 原理

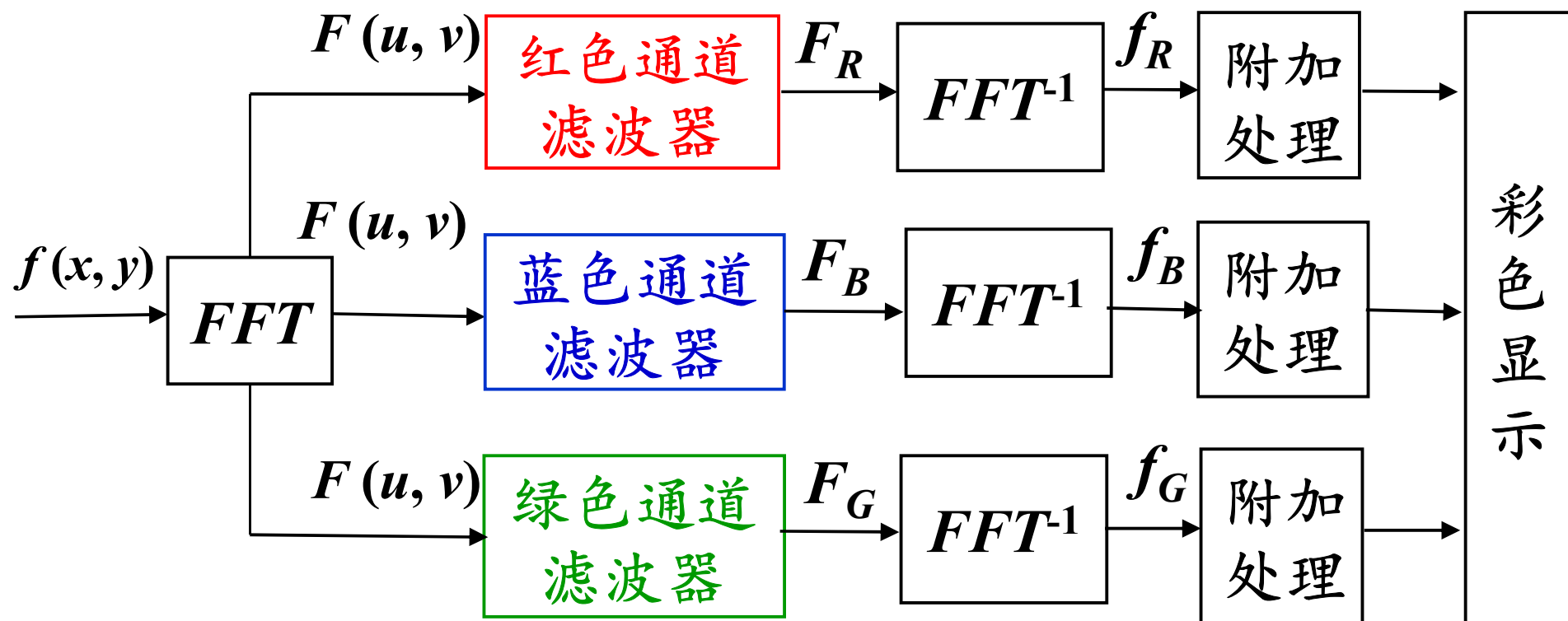
- 图像经傅里叶变换到频率域；
- 用三个不同传递特性的滤波器分离成三个独立分量；
- 逆傅里叶变换，得到三幅不同频率分量的单色图像；
- 进一步的附加处理（如直方图均衡化等）；
- 合成得到彩色图像，实现基于频域的伪彩色增强。

典型的频域伪彩色增强方法是设计相应的低通、带通和高通三种滤波器，



## 5.5.3 频域伪彩色增强

### 基于伪彩色处理的图像增强







## 5.5.3 频域伪彩色增强

### 基于伪彩色处理的图像增强

#### (2) 例程

##### ■ 程序

```
Image=imread('qiguan.bmp');  
[height,width]=size(Image);  
NewImage=zeros(height,width,3);  
G=fft2(Image);  
n=2;   RedD0=150;   GreenD0=200;  
Bluecenter=150;   Bluewidth=100;  
Blueu0=10;        Bluev0=10;  
for u=1:height  
    for v=1:width  
        D(u,v)=sqrt(u^2+v^2);  
        RedH(u,v)=1/(1+(sqrt(2)-1)*(D(u,v)/RedD0)^(2*n));
```



## 5.5.3 频域伪彩色增强

### 基于伪彩色处理的图像增强

$\text{GreenH}(u,v) = 1 / (1 + (\sqrt{2} - 1) * (\text{GreenD0}/D(u,v))^{2*n});$

$\text{BlueD}(u,v) = \sqrt{(u - \text{Blueu0})^2 + (v - \text{Bluev0})^2};$

$\text{BlueH}(u,v) = 1 - 1 / (1 + \text{BlueD}(u,v) * \text{Bluewidth} / ((\text{BlueD}(u,v))^{2 - (\text{Bluecenter})^2})^{2*n});$

end

end

$\text{Red} = \text{RedH} * G; \quad \text{Green} = \text{GreenH} * G; \quad \text{Blue} = \text{BlueH} * G;$

$\text{RedC} = \text{ifft2}(\text{Red}); \quad \text{GreenC} = \text{ifft2}(\text{Green}); \quad \text{BlueC} = \text{ifft2}(\text{Blue});$

$\text{NewImage}(:, :, 1) = \text{uint8}(\text{real}(\text{RedC}));$

$\text{NewImage}(:, :, 2) = \text{uint8}(\text{real}(\text{GreenC}));$

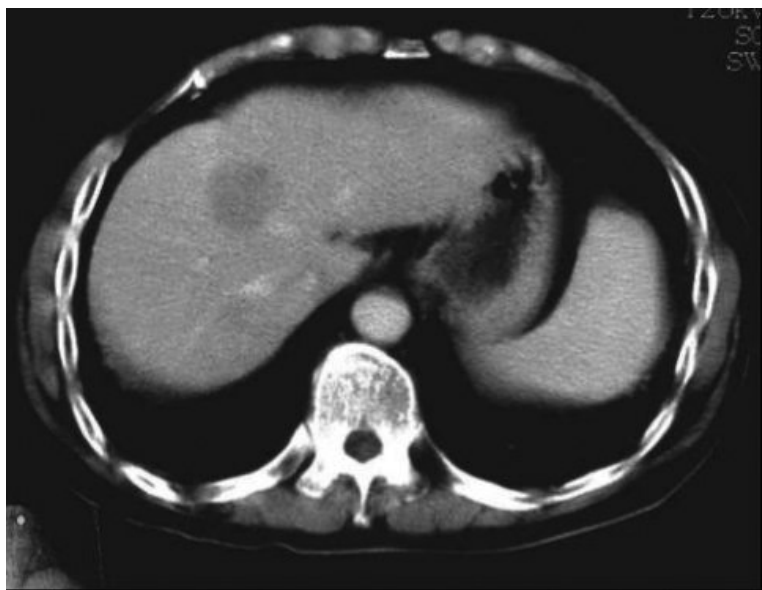
$\text{NewImage}(:, :, 3) = \text{uint8}(\text{real}(\text{BlueC}));$

$\text{figure}; \text{imshow}(\text{NewImage}); \text{title}(' \text{频域伪彩色增强} ');$

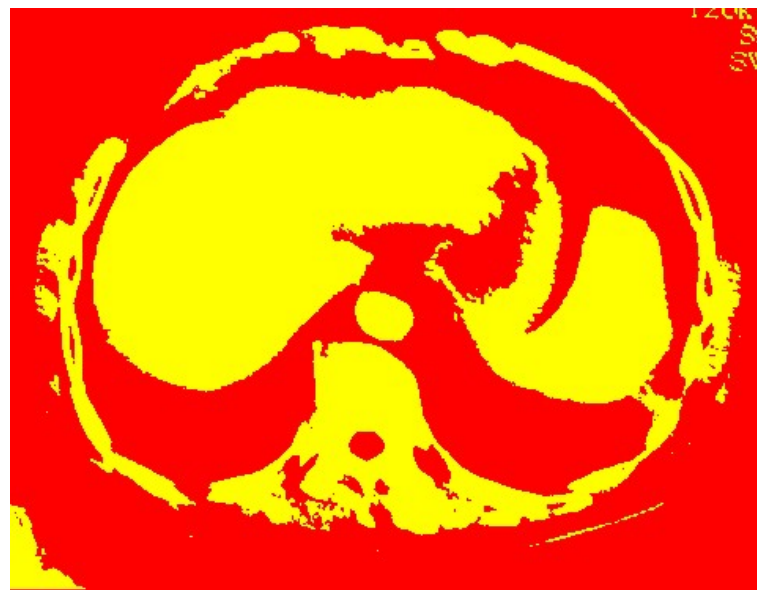
## 5.5.3 频域伪彩色增强

### 基于伪彩色处理的图像增强

#### ■ 效果



(a) 原始图像



(b) 伪彩色增强后图像



## 5.6 其他图像增强方法

### 5.6.1 基于对数图像处理模型的图像增强

### 5.6.2 图像去雾增强



## 5.6.1 基于对数图像处理模型的图像增强

### 其他图像增强方法

通常，在对图像进行线性运算处理过程中，一般的标量乘法“ $\times$ ”和加法“ $+$ ”运算并不通用于图像的形成法则：

- 可能产生“超区间值”问题，导致信息丢失；
- 线性运算结果与人类视觉感官有所偏差；

因此，提出对数图像处理（LIP）模型。



## 5.6.1 基于对数图像处理模型的图像增强

### 其他图像增强方法

#### (1) 对数图像处理模型

图像 $I(x,y)$ 用灰度色调函数 $f(x,y)$ 表示为：

$$f(x,y) = M - I(x,y)$$

- $M$  是正值。对于8比特的图像， $M$ 取为256。
- $f(x,y) \in [0,M]$

灰度色调函数的物理意义为光线通过一个光强滤波器形成进入人眼的透射光而成像，本质是把原图像转换到对应的光强滤波函数来表示，该光强滤波函数就定义为灰度色调函数。



## 5.6.1 基于对数图像处理模型的图像增强

### 其他图像增强方法

#### ■ LIP基础向量运算

##### □ 对数加法运算

$$f_1(x, y) \oplus f_2(x, y) = f_1(x, y) + f_2(x, y) - \frac{f_1(x, y) \cdot f_2(x, y)}{M}$$

##### □ 对数乘法运算

$$\lambda \otimes f_1(x, y) = M - M \left( 1 - \frac{f_1(x, y)}{M} \right)^\lambda$$

##### □ 对数减法运算

$$f_1(x, y) \ominus f_2(x, y) = M \cdot \frac{f_1(x, y) - f_2(x, y)}{M - f_2(x, y)}$$





## 5.6.1 基于对数图像处理模型的图像增强

### 其他图像增强方法

#### ■ LIP基本同态函数

正变换 
$$\varphi(f) = -M \cdot \ln \left( 1 - \frac{f(x, y)}{M} \right)$$

反变换 
$$\varphi^{-1}(f) = M \cdot \left( 1 - e^{\left( \frac{-f(x, y)}{M} \right)} \right)$$

- 通过正变换，灰度色调函数 $f(x, y)$ 的取值范围由 $[0, M]$ 扩大到 $[-\infty, M]$ 。
- 通过正、逆变换，使LIP的运算空间和输入图像的实数空间公式同构。



## 5.6.1 基于对数图像处理模型的图像增强

### 其他图像增强方法

#### (2) 基于对数图像处理模型的增强

$$f'(x, y) = \alpha \otimes A(x, y) \oplus \beta [f(x, y) \ominus A(x, y)]$$

- $f(x, y)$  是原图的灰度色调函数;
- $f'(x, y)$  是增强后图像的灰度色调函数;
- $A(x, y)$  为  $n \times n$  邻域的灰度色调函数的平均值;
- $\alpha$  用来调整对比度;
- $\beta$  用来调整锐化效果



## 5.6.1 基于对数图像处理模型的图像增强

### 其他图像增强方法

将 $f(x,y)$ 进行基本正态函数的正变换，且令：

$$\bar{f}(x, y) = 1 - f(x, y)/M$$

化简后的增强表达式

$$\ln \bar{f}'(x, y) = \alpha \cdot \ln \bar{A}(x, y) + \beta \cdot [\ln \bar{f}(x, y) - \ln \bar{A}(x, y)]$$

$$\ln \bar{A}(x, y) = \frac{1}{n \times n} \sum_{k=x-\frac{n}{2}}^{x+\frac{n}{2}} \sum_{l=y-\frac{n}{2}}^{y+\frac{n}{2}} \ln \bar{f}(x, y)$$



## 5.6.1 基于对数图像处理模型的图像增强

### 其他图像增强方法

#### ■ 算法步骤

- 求灰度色调函数；
- 进行基本正态函数的正变换；
- 求以点 $(x,y)$  中心的 $n \times n$ 邻域的平均值；
- 进行基于对数图像处理模型的增强处理；
- 进行基本正态函数的反变换。

## 5.6.1 基于对数图像处理模型的图像增强

### 其他图像增强方法



(a) 原始图像



(b) LIP增强后图像



## 5.6.2 图像去雾增强

### 其他图像增强方法

由于受雾的影响，拍摄的图像往往出现模糊不清、对比度下降、色彩失真等质量退化现象。

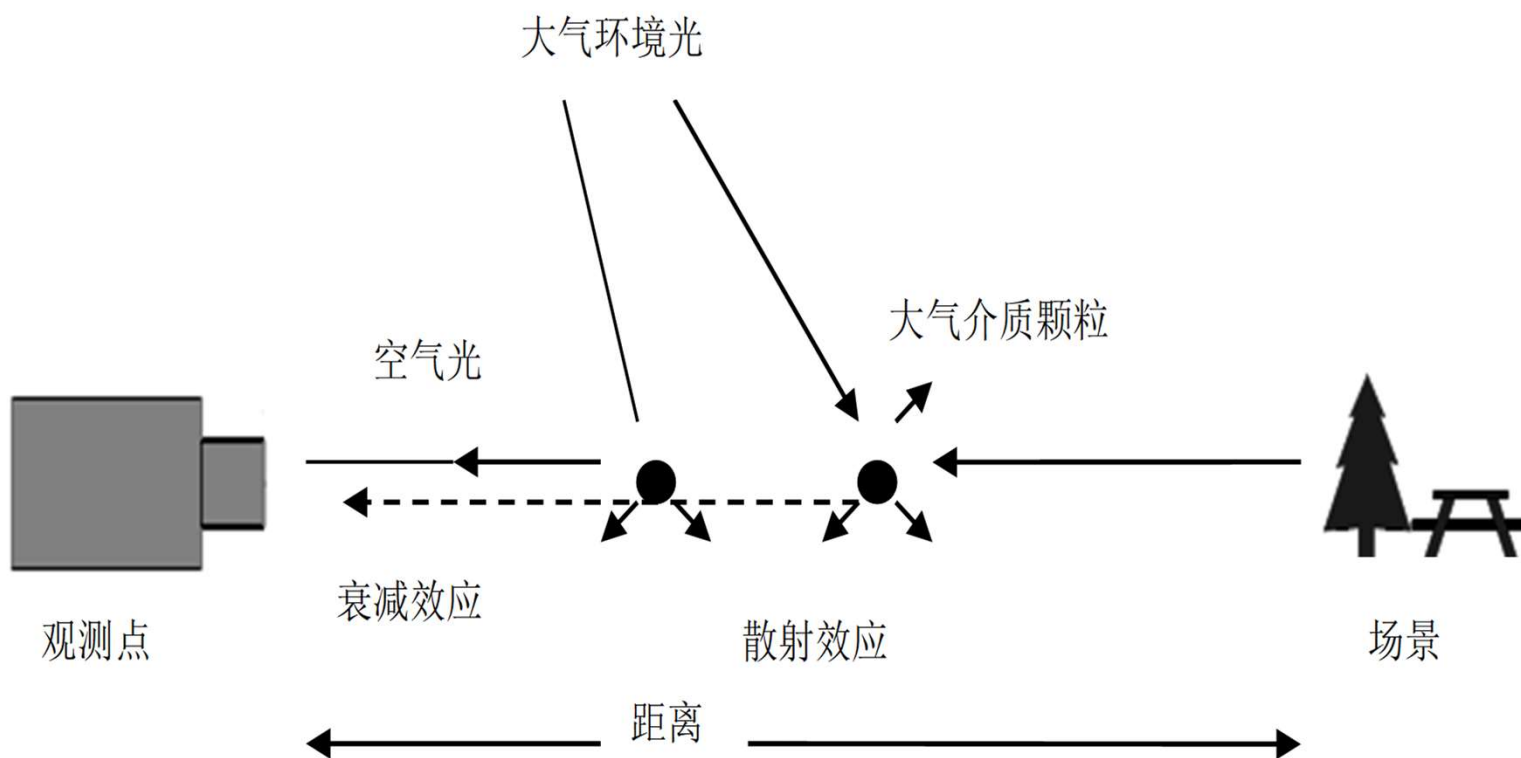
图像去雾方法主要分为两类：

- 基于非物理模型的去雾方法，是不考虑雾导致图像退化的成因，只通过实现对比度增强方法来达到去雾目的。
- 基于物理模型的去雾方法，考虑雾导致图像退化成因，进行数学建模，补偿退化过程造成的失真，恢复无雾图像。

## 5.6.2 图像去雾增强

### 其他图像增强方法

#### (1) 图像去雾物理模型







## 5.6.2 图像去雾增强

### 其他图像增强方法

令  $E_d(d, \lambda)$  --- 衰减效应,

$E_s(d, \lambda)$  --- 散射效应

$$E_d(d, \lambda) = J(x) \cdot e^{-\beta(\lambda) \cdot d(x)}$$

$$E_s(d, \lambda) = A \left( 1 - e^{-\beta(\lambda) \cdot d(x)} \right)$$

$$I(x) = E_d(d, \lambda) + E_s(d, \lambda)$$



$$I(x) = J(x) \cdot e^{-\beta \cdot d(x)} + A \left( 1 - e^{-\beta \cdot d(x)} \right)$$

图像去雾  
的物理模型

$J(x)$  为场景目标直接反射光成像的亮度。  
 $A$  为大气光。 $\beta$  为空气的散射系数



## 5.6.2 图像去雾增强

### 其他图像增强方法

#### (2) 基于暗原色先验的去雾方法

He Kaiming等人通过对户外大量清晰无雾的自然图像观察统计得到：

在绝大多数非天空的局部区域内，某一些像素在RGB三色通道中至少有一个通道的像素颜色值比较低。



## 5.6.2 图像去雾增强

### 其他图像增强方法

#### ■ 暗原色先验

若给出一幅清晰无雾图像  $J$ ，暗通道  $J^{dark}$  为：

$$J^{dark}(i,j) = \min_{C \in \{r,g,b\}} \left( \min_{y \in \Omega(i,j)} (J^C(y)) \right)$$

$\Omega(i,j)$  表示以点  $(i,j)$  为中心的一个局部区域

对于非天空清晰无雾图像  $J$ ， $J^{dark}$  值非常低，趋近于 0，那么  $J^{dark}$  称为  $J$  的暗颜色，并且把以上观察得出的经验性规律称为 暗原色先验。



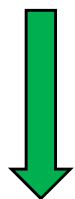
## 5.6.2 图像去雾增强

### 其他图像增强方法

#### ■ 图像去雾的物理模型简化

定义 $t(x)$  为介质传播函数，表示透射率

$$t(x) = e^{-\beta \cdot d(x)} \quad 0 \leq t(x) \leq 1$$



基于图像去雾的物理模型

$$I(x) = J(x) \cdot t(x) + A(1 - t(x))$$

暗原色先验去雾的目的就是希望能够通过雾天图像 $I(x)$  恢复无雾图像 $J(x)$ 。



## 5.6.2 图像去雾增强

### 其他图像增强方法

#### ■ 估计大气光 $A$ ---计算图像的暗原色最亮区域

##### □ 算法步骤

- ✓ 划分雾天图像的局部块，求暗原色图；
- ✓ 寻找图像中最不透明区域；
- ✓ 求该区域的像素的亮度平均 $I_{mean\_dc}$ ；
- ✓ 设置大气光 $A$ 的估算最大值 $A_{max}=240$ ，  
计算  $\min(A_{max}, I_{mean\_dc})$  作为大气光 $A$ 的估计值。

## 5.6.2 图像去雾增强

### 其他图像增强方法

#### ■ 估计大气光 $A$ ---计算图像的暗原色最亮区域

图像 $I$ 亮度值最大区域



由算法估计大气光 $A$

暗原色图最亮区域



- ✓ 暗原色图中最亮点作为估计的大气光的值。
- ✓ 估计的大气光并不一定为图像中亮度最大值。



## 5.6.2 图像去雾增强

### 其他图像增强方法

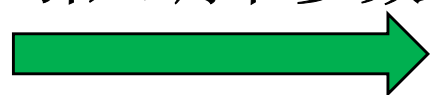
#### ■ 估计透射率 $t(x)$

$$\min_{C \in \{r, g, b\}} \left( \min_{x \in \Omega(i, j)} \left( \frac{I^C(x)}{A} \right) \right) = t(x) \cdot \min_{C \in \{r, g, b\}} \left( \min_{x \in \Omega(i, j)} \left( \frac{J^C(x)}{A} \right) \right) + (1 - t(x))$$

因为  $\min_{C \in \{r, g, b\}} \left( \min_{x \in \Omega(i, j)} \left( \frac{J^C(x)}{A} \right) \right) \rightarrow 0 \quad A > 0$

→  $t(x) = 1 - \min_{C \in \{r, g, b\}} \left( \min_{x \in \Omega(i, j)} \left( \frac{I^C(x)}{A} \right) \right)$

引入调节参数



$$\omega \in (0, 1)$$

$$t(x) = 1 - \omega \cdot \min_{C \in \{r, g, b\}} \left( \min_{x \in \Omega(i, j)} \left( \frac{I^C(x)}{A} \right) \right)$$





## 5.6.2 图像去雾增强

### 其他图像增强方法

#### ■ 获得无雾图像

$$J^c(x) = A + \frac{I^c(x) - A}{t(x)}$$



## 5.6.2 图像去雾增强

### 其他图像增强方法



(a) 原图



(b) 暗原色先验去雾效果



## 5.6.2 图像去雾增强

### 其他图像增强方法

#### (3) 基于暗原色先验的低照度图像增强

将低照度图像反转后，其图像表征及直方图表征与雾天图像具有很高的相似性。

##### ■ 算法步骤

- 求得输入低照度图像的反转图像

$$R^C(x) = 255 - I^C(x)$$

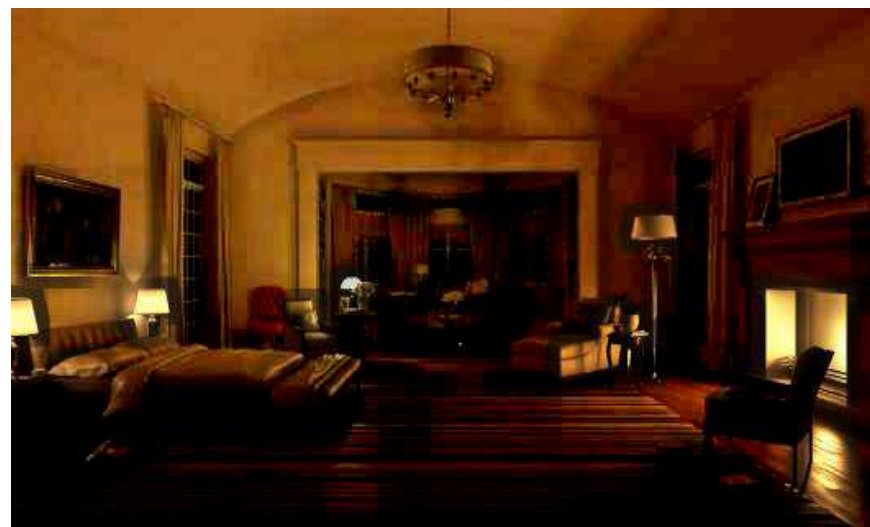
- 进行暗原色先验去雾，获得无“雾”图像
- 反转无“雾”图像，获得低照度增强图像

## 5.6.2 图像去雾增强

### 其他图像增强方法



(a) 低照度图像



(b) 暗原色先验增强结果