

# 数学软件课程报告(2)

孟丽媛

July 29, 2020

## 摘要

报告将讨论基于各向异性扩散的图像降噪算法的发展过程<sup>1</sup>。对其发展历程中的重要模型做出概述, 对其中较为典型的模型的原理和特点做出更为详细的说明, 通过基于Matlab的仿真实验测试模型的降噪效果。

## 1 引言

在图像处理中, 各向同性扩散在消除噪声时无法很好的保留图像的细节信息。针对各向同性扩散带来的缺点, 根据图像原有信息, 使扩散在图像边缘处减少, 从而保护图像细节。用梯度算子作为边缘检测, 构造一个扩散系数关于原图像梯度递减的函数, 由此得到的各向异性扩散的图像降噪模型很好的平衡了图像降噪与细节保持之间的矛盾, 成为了图像降噪算法研究的一个重要方向。

## 2 各向异性扩散模型的发展过程

1990年Perona和Malik<sup>[1]</sup>提出了一种非线性扩散方程<sup>[1]</sup>。

J.weicker等人对扩散系数做了改进<sup>[2]</sup>, 将其改进为一个矩阵, 将扩散分解为沿图像等灰度线切向和法向两个分量, 优化了P-M模型。

Catté证明了<sup>[3]</sup>中提出的模型的解存在唯一<sup>[4]</sup>, L.Alvarez等人指出Catté\_PM模型当方差趋于0时, 解不稳定, 因此提出了等度轮廓线曲率演化方程<sup>[5]</sup>。

Rudin, Osher提出用BV空间的半范数-全变差来作为正则化, 提出TV正则化模型<sup>[6]</sup>。

低阶模型的共性问题会产生阶梯效应, You, Kaveh引入了四阶偏微分方程<sup>[7]</sup>, 用分段斜面来近似边界, 使处理后的图像更加自然, 但该模型处理后的图像边缘保持较差。

Jian Ling等提出各向异性中值扩散滤波<sup>[8]</sup>, 该算法在每次扩散后进行中值滤波, 在有效降噪的同时, 保持图像边缘。为使扩散过程对参数设置更具鲁棒性, 引入Biweight范数作为扩散系数<sup>[9]</sup>。

Gilboa等提出了正逆各向异性扩散模型, 在消除图像加性噪声的同时, 有效增强图像细节信息<sup>[10]</sup>。

## 3 各向异性扩散模型

### 3.1 经典P-M模型<sup>[1]</sup>

P-M各向异性扩散模型:

$$I_t = \nabla(g(|DI|)) \cdot \nabla I \quad (1)$$

其中, 扩散系数 $g$ 有两种形式:

$$g(*) = \frac{1}{1 + (\frac{*}{K})^2} \quad (2)$$

$$g(*) = e^{-(\frac{*}{K})^2} \quad (3)$$

各向异性扩散的基本思想是求解初值为原始图像的偏微分扩散方程。在P-M模型中, 由引入的系数函数 $g$ 来辨别图像边缘, 由领域中像素点的加权平均来模糊噪声, 多次迭代直至降噪效果符合要求。扩散系数 $g$ 使得在原图像梯度值较大的部分, 模型执行弱的平滑, 以保留边缘信息; 在原图像梯度值较小的部分, 模型执行强的平滑, 以实现图像降噪。

经典P-M模型克服了基于各向同性扩散模型的边缘保持的问题, 但仍存在一些问题, 由于P-M模型的扩散系数依赖于图像的梯度变化, 当噪声较强时, 模型反而会使噪声增强。此外, 当迭代次数增加时, 图像在平滑区域的梯度趋于零, 降噪后的图像显示出明显的阶梯效应。

### 3.2 Catté\_PM模型<sup>[4]</sup>

在经典P-M模型中系数函数 $g$ 辨别图像边缘依赖于待处理图像的梯度模, 也是因为这一原因经典P-M模型对

<sup>1</sup> 仅关注高斯噪声和脉冲噪声的图像降噪, 不包括近年来针对斑点噪声去除提出的模型。

强噪声的降噪效果不好。在Catté\_PM模型中用高斯平滑<sup>2</sup>后的梯度模代替原图像的梯度模。对于受高斯噪声污染的图像，高斯平滑后的图像梯度模可以更准确地区分图像边缘而不受噪声影响。Catté\_PM模型可以表示为：

$$I_t = \nabla(g(|\nabla(G_\sigma * I)|) \cdot \nabla I) \quad (4)$$

其中：

$$G_\sigma(*) = C\sigma^{-\frac{1}{2}}e^{-\frac{*^2}{4\sigma}} \quad (5)$$

是方差为 $\sigma$ 的高斯函数，\*表示卷积。

Catté\_PM模型对受加性高斯噪声污染的图像有很好的降噪效果。，因此被认为是一种较为成功的PM模型改进方法。但方差的选取对于此模型降噪效果有决定性的影响，且它的值较难确定，这是Catté\_PM模型的不足之处，同时此模型对其他统计特性的噪声处理能力下降。

### 3.3 F\_PDE模型<sup>[7]</sup>

为克服低阶扩散方程的阶梯效应，2000年You和Kaveh提出了基于四阶偏微分方程的各向异性扩散模型<sup>3</sup>。低阶模型在平滑迭代一定次数后，相对平滑的区域收敛为具有固定亮度的区块，使图像变为由不同亮度的区域组成，显示出明显的阶梯效应。You和Kaveh引入的F\_PDE模型用拉普拉斯绝对值来度量边缘区域，使得从视觉体验上，图像无阶梯效应。

F\_PDE模型可表示为：

$$I_t = \nabla(-\nabla^2(g(|\nabla^2 I|) \cdot \nabla^2 I)) \quad (6)$$

其中 $\nabla^2 I$ 为拉普拉斯算子。

[11]中用梯度模代替拉普拉斯绝对值作为边缘检测算子，改进的模型具有较快的收敛速度，改善了去噪效果，但模糊了图像边缘。[12]提出了各向异性的四阶降噪算法，在图像等灰线的法向和切向进行不同程度的扩散，更好的保留了图像的细节，但该模型会产生梯度伪影。

### 3.4 FABD模型<sup>[10]</sup>

2002年Gilboa等提出了正逆各向异性扩散模型<sup>4</sup>。Gilboa等指出如P-M模型等的正扩散主要是通过减缓在图像奇异值（如边缘）处的扩散力度来保留这些

奇异值，而如图像增强和锐化等的应用，需要增强这些奇异值或复原被噪声淹没的奇异值，这个过程可视为正扩散的逆过程，即逆扩散。为了在消除图像加性噪声的同时更有效地增强和锐化图像细节，Gilboe等提出了将正逆扩散相结合。

FABD扩散模型：

$$I_t = \nabla(g(|DI|) \cdot \nabla I) \quad (7)$$

$$g(*) = \frac{1}{1 + (\frac{*}{K_f})^n} - \frac{\alpha}{1 + (\frac{* - K_b}{\omega})^{2m}} \quad (8)$$

其中，Eq.8 等号右边第一项为正扩散项，第二项为逆扩散项； $K_f$ 表示正扩散门限， $K_b$ 和 $\omega$ 表示逆扩散门限； $\alpha$ 表示正逆扩散比例；n, m表示正逆扩散强度。

此模型使得对较低梯度值的平滑区域，通过正扩散降噪；当梯度值较大的区域，正扩散消失，通过逆扩散增强细节信息；当梯度值非常大时，逆扩散也被抑制以此防止增强噪声。

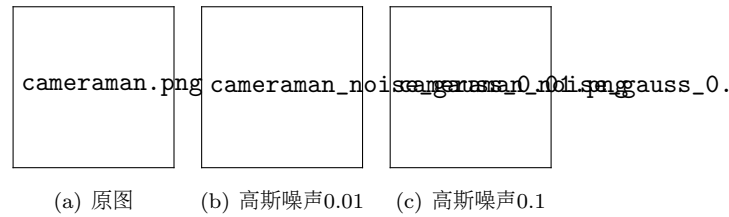
当噪声程度较大时，该模型性能下降；另外该模型中经验参数过多且难以自适应确定，限制了该模型的应用范围。

## 4 计算结果

### 4.1 评价指标

评价指标采用信噪比(SNR)、峰值信噪比(PSNR)和平均结构相似度(MSSIM)<sup>[18]</sup>。

### 4.2 cameraman灰度图像测试结果



	噪声原图	P-M模型	改进的P-M模型
SNR	-0.73	4.73	4.73
PSNR	62.84	68.29	68.29
MSSIM	62.36	63.38	63.37

## 5 实验结果

直观上对比P-M模型与改进的P-M模型的降噪效果，

<sup>2</sup> 高斯平滑是一种二维卷积运算，类似于均值滤波器，但使用的是不同的内核，标准差越大，越接近于均值滤波。

<sup>3</sup> fourth-order partial differential equations, 简称F\_PDE, 也称YK模型。

<sup>4</sup> forward-and-backward diffusion, FABD

二者并无明显差别；对比表中SNR, PSNR, MSSIM三个指标，二者几乎完全相同。[1]中新提出的模型相较于P-M模型没有实质性的改进，原文中的实验结果需要归功于将各向异性扩散扩展到8个梯度方向这一改进。

## 6 结论

通过数学软件课程一周的学习，对于Matlab/Octave的功能有了更多的了解。

我觉得数学软件相较于C或者Python等编程语言，更为集成一些，许多常用的计算功能都有已经封装好的函数，对于从事较为顶层的开发的工作者，这使得他们不用了解该计算过程的具体原理和算法，就可以直接调用计算时间少，占用内存少的高效的计算方法。在这一份课程报告中，图像与高斯函数的卷积运算就是这样一个例子，通过直接调用Matlab的imfilter函数可以便捷地实现模型的要求，不需要耗费额外的时间精力去寻找高效的卷积算法，为实验验证带来了极大便利。

此外，我也觉得想要用好数学软件，搞清楚封装好的函数背后的原理和算法也是很有益的。在这一份课程报告中，MSSIM指标函数是这样一个例子，此函数来自Matlab Central File Exchange社区，由作者Mohammad Moinul Islam提供。调用此函数时发现运行时间很长，查看函数的源代码后发现作者的原函数中有一些变量对于本次实验是用处不大的，可以设为常数；循环的步长虽然使得精度更高，但却让运行时间变长很多，而其提高的精度在本次实验中并不明显，可以修改步长在损失一定精度的代价下减少运行时间。当理解了被调用函数的原理和算法时，就可以对原函数进行一些微调，以使被调用函数更符合自己的实验要求。

最后，我觉得Matlab的矩阵计算功能很好用，在处理矩阵运算时，不需要让自己在人工计算矩阵的思维和计算机计算矩阵的思维之间来回切换，提高了工作效率。特别是在完成这一份课程报告中图像处理任务时，矩阵计算功能显得尤为方便。

## 参考文献

- [1] Scale-space and edge detection using anisotropic diffusion. Perona P, Malik J. IEEE Transactions on Pattern Analysis and Machine Intelligence . 1990
- [2] Weickert J. Anisotropic Diffusion In Image Processing[J]. B.g.teubner Stuttgart,1998, 16(1):272.
- [3] Alvarez L, Lions P L, Morel J M. Image Selective Smoothing And Edge-Detection By Nonlinear Diffusion. 2[J]. Siam Journal on Numerical Analysis, 1992, 29(1):845- 866.
- [4] CATTE F, LIONS P, MOREL J, et al. Image selectivesmoothing and edge detection by nonlinear diffusion[J].SIAM Journal on Numerical Analysis, 1992, 29:182-193.
- [5] Alvarez L, Lions P L, Morel J M. Image Selective Smoothing and EdgeDetection by Nonlinear Diffusion. II[J]. Siam Journal on Numerical Analysis,1992, 29(1 ):845-866.
- [6] Rudin L I, Osher S, Fatemi E. Nonlinear total variation based noise removalalgorithms [J]. Physica D Nonlinear Phenomena, 1992, 60(1- -4):259-268.
- [7] YOU Y, KAVEH M. Fourth-order partial differential equa-tions for noise removal[J].IEEE Transactions on ImageProcessing, 2000, 9 (10) :1723-1730.
- [8] LING J, BOVIK A C. Smoothing low-SNR molecular images via anisotropic median-diffusion[J].IEEE Transac-tions on Medical Imaging, 2002, 21 (4) :377-384.
- [9] BLACK M J, SAPIRO G, MARIMONT D H, et al. Ro-bust anisotropic diffusion[J].IEEE Transactions on ImageProcessing, 1998, 7 (3) :412-432.
- [10] GILBOA G, SOCHEN N, ZEEVI Y Y. Forward-and-backward diffusion processes for adaptive image enhancement and denoising[J].IEEE Transactions on ImageProcessing, 2002, 11 (7) :689-703.
- [11] Hajiaboli M R. A Self-governing Fourth-order Non-linear Diffusion Filter for Image Noise Removal[J]. Ipsj Transactions on Computer Vision Applications, 2010, 2:94-103.
- [12] Hajiaboli M R. An Anisotropic Fourth-Order Diffusion Filter for Image NoiseRemoval[J]. International Journal of Computer Vision, 2011, 92(2):177-191.
- [13] 余锦华,汪源源.基于各向异性扩散的图像降噪算法综述[J].电子测量与仪器学报,2011,25(02):105-116.
- [14] 王艳. 基于各向异性扩散的图像降噪算法的研究与优化[D].中北大学,2014.
- [15] 周浩. 基于偏微分方程的图像降噪算法研究[D].江苏科技大学,2010.
- [16] 陈建军. 基于偏微分方程的图像降噪和图像恢复研究[D].,2009.

- [17] 吴登辉. 一类基于偏微分方程的图像去噪方法研究[D].南京信息工程大学,2017.
- [18] 杨成佳.图像去噪及其效果评估若干问题研究[D].吉林大学,2016.

# 附录1

## main函数

```
I = imread('cameraman.jpg'); K = autoK(I);
l=0.01; In=imnoise(I,gaussian,0,l);
I = double(I)/256; In = double(In)/256;
Ina=In; dt = 0.05; k=1000; %迭代次数
imshow(In,[])
%打印图片和SNR,PSNR,MSSIM值,代码中文件名包含下划线等符号,以文字
代替
for tt = 1:k
    %1. 计算四个方向的梯度
    Gn = [In(1,:,:) ; In(1:end-1,:,:) ] - In; % N - O
    Gs = [In(2:end,:,:) ; In(end,:,:) ] - In; % S - O
    Ge = [In(:,2:end,:) In(:,end,:)] - In; % E - O
    Gw = [In(:,1,:) In(:,1:end-1,:)] - In; % W - O
    %2. 计算扩散稀疏
    Cn = 1./(1 + (Gn/K).^2);
    Cs = 1./(1 + (Gs/K).^2);
    Ce = 1./(1 + (Ge/K).^2);
    Cw = 1./(1 + (Gw/K).^2);
    %3. 一阶时间格式: 求解PDE(扩散)
    diff = (Cn.*Gn + Cs.*Gs + Ce.*Ge + Cw.*Gw);
    In = In + dt*diff; % 这是一个显欧拉格式
    Cna = 1./log(exp(1) + (Gn/K).^2);
    Csa = 1./log(exp(1) + (Gs/K).^2);
    Cea = 1./log(exp(1) + (Ge/K).^2);
    Cwa = 1./log(exp(1) + (Gw/K).^2);
    %3. 一阶时间格式: 求解PDE(扩散)
    diffa = (Cna.*Gn + Csa.*Gs + Cea.*Ge + Cwa.*Gw);
    ;
    Ina = Ina + dt*diffa; % 这是一个显欧拉格式
end
imshow(In,[])%...
imshow(Ina,[])%...
```

## aotoK函数

```
function K = autoK(I)%自动生成参数K的图像I为噪声图像
[row,col,nchannel] = size(I); K = 0;
if nchannel == 1 %gray image
    [gradx,grady] = imgradient(I);
    %gradx是图像水平方向梯度, grady是图像竖直方向梯度
    gradI = (gradx.^2 + grady.^2).^0.5;
    %gradI为每一像素点处梯度向量的2-范数
    K = 1.4826*mean(mean(abs(gradI-mean(mean(gradI)))))
    %mean(mean(gradI))为I中所有像素点的梯度向量2-范数的均值
    %mean(mean(abs(...)))代表了图像I中各像素点组成的集合的离散程度
    %K值太小时去噪效果不好, K值太大时边缘保持变差
    %所以图像I中各像素点组成的集合的离散程度可以表示K值大小
else %color image
    for i=1:3
        [gradx,grady] = imgradient(I(:,:,i));
        gradI = (gradx.^2 + grady.^2).^0.5;
        K = K + 1.4826*mean(mean(abs(gradI - mean(mean(gradI)))))
    end
    K = K/3;
    %在三通道彩色图像中, 以各像素点通道在每一组成的集合的离散程度的均值表示K值大小
end
end
```

## SNR指标函数

```
function snr=SNR(I,In)
% 计算图像信噪比函数, 其中I: 原图像In: 带噪声图像
[~,~,nchannel] = size(I);
if nchannel==1 %gray image
```

```
    ps=sum(sum((I-mean(mean(I))).^2));
    pn=sum(sum((I-In).^2));
else %color image
    ps=0;pn=0;
    for i=1:3
        ps=ps+sum(sum((I(:,:,i)-mean(mean(I(:,:,i)))).^2));
        pn=pn+sum(sum((I(:,:,i)-In(:,:,i)).^2));
    end
    ps=ps/3;pn=pn/3;
end
snr=10*log10(ps/pn);
end
```

## PSNR指标函数

```
function psnr = PSNR(I, In)
% 计算图像峰值信噪比函数, 其中I: 原图像In: 带噪声图像
[n,m,nchannel] = size(I);
if nchannel==1 %gray image
    mse=sum(sum((I-In).^2))/(n*m);
else %color image
    mse=0;
    for i=1:3
        mse=mse+sum(sum((I(:,:,i)-In(:,:,i)).^2))/(n*m);
    end
    mse=mse/3;
end
psnr=10*log10(255^2/mse);
end
```

## MSSIM指标函数

```
function mssim=MSSIM(I,In)
% I: 原图像In: 带噪声图像
[m,n,p]= size(I); c1= 0.0001; c2= 0.0009; win=max(m/2,n/2);
%win原本为调用函数时传递的参数, 表示MSSIM在SSIM基础上取均值时, 视窗选取的大小
%在本实验中为方便起见, 且不失MSSIM指标准确性,将其改为在函数内部定义
sz= floor(win/2); mer= [];
for k= 1:p
    for i= sz+1:10:m-sz
        for j= sz+1:10:n-sz
            %原函数循环步长为1, 考虑时间原因, 这里改为10
            six= I(i-sz:i+sz,j-sz:j+sz,k);
            %图像I第k维在当前视窗下的矩阵
            six= six(:); %将six拉成一维向量
            siy= In(i-sz:i+sz,j-sz:j+sz,k);
            %图像In第k维在当前视窗下的矩阵
            siy= siy(:); %将siy拉成一维向量
            meux= mean(six); meuy= mean(siy); %取均值
            sigx= std(six,0); sigy= std(siy,0); %取标准偏差
            sigxy= sum((six-meux).*(siy-meuy))/(numel(six)-1);
            er= ((2*meux*meuy+c1)*(2*sigxy+c2))/((meux^2+meuy^2+c1)*(sigx^2+sigy^2+c2));
            mer= [mer er];
        end
    end
end
mssim= sum(mer)/(numel(mer));
%此函数来自Matlab Central File Exchange, 由Mohammad Moinul Islam提供
%改变了变量命名以使各去噪指标函数编程风格一致
end
```