

论文题目：毕设模板

学生姓名：王鸿鑫

指导教师：郝运

摘 要

中文摘要另起一页，在这页开头向左对齐，用宋体，小四，加粗，标明：论文题目，学生姓名，指导教师。居中编排“摘要”二字（三号宋体），二字间距为两个字符。“摘要”二字下为摘要正文，每段开头空两字符，小四号。摘要正文内容下，空一行，左对齐，打印“关键词”三字（五号加黑），后接冒号，其后为关键词（五号宋体）。关键词由 3 ~ 5 个组成，每一关键词之间用分号隔开，最后一个关键词后无标点符号

关键词：中文

Title: moban

Name: wanghongxin

Supervisor: haoyun

ABSTRACT

The two main plagues of image restoration are oscillations and smoothing. Traditional image restoration techniques prevent parasitic oscillations by resorting to smooth regularization. Hence, singular image features and oscillatory textures cannot be restored. The usefulness of images obtained by smooth regularization is very limited. Regularized faces and characters lose features and are unrecognizably distorted. Oscillatory patterns and textures are not allowed by regularization methods. Smoothly regularized images are of no use in criminal/civil investigations and as court evidence.

The two main plagues of image restoration are oscillations and smoothing. Traditional image restoration techniques prevent parasitic oscillations by resorting to smooth regularization. Hence, singular image features and oscillatory textures cannot be restored. The usefulness of images obtained by smooth regularization is very limited. Regularized faces and characters lose features and are unrecognizably distorted. Oscillatory patterns and textures are not allowed by regularization methods. Smoothly regularized images are of no use in criminal/civil investigations and as court evidence.

KEY WORDS : stupid xjtu

目 录

1 绪论	4
1.1 参考文献	5
1.2 二级标题	5
1.3 三级标题	6
1.3.1 四级标题	6
1.3.2 四级标题	6
1.3.3 又是一个四级标题.....	6
2 理论分析	7
2.1 问题重述	7
2.2 问题分析	7
2.3 PDE 的推导	8
2.4 模型的数值实现	9
3 实验	12
3.1 实验 1	12
3.2 实验 2	13
4 结论与展望	15
附录 1 Matlab 程序	16
附录 2 还是那个 Matlab 程序	18
致谢	20

1 绪论

正文是学位论文的主体，第一章为绪论，最后一章为结论与展望。我们来测试一个上标引用。为什么要写这篇论文，要解决什么问题，主要观点是什么。对本论文研究主题范围内已有文献的评述（包括与课题相关的历史的回顾，资料来源、性质及运用情况等）。说明本论文所要解决的问题，所采用的研究手段、方式、方法。明确研究工作的界限和规模。概括论文的主要工作内容。

我们其实已经不知不觉地到达了终点。不幸的是，我们经过 20 多年辛苦跋涉所到达的这个终点，并不是什么美丽的“新世界”，而是中国历史上反复出现过的旧朝代的翻版。王亚南先生早在半个世纪之前就在它那部研究中国官僚体系的开山之作中提醒我们，中国源远流长的官僚政治体系有着神奇的亲和力，它可以与任何体制相互融合。它既可以与古老的小农经济，也可以与现代意义上的自由市场或者国家资本主义相结合；它既能够将儒释道汇于一炉，也能够将现代三民主义、法西斯主义杂糅并蓄之后，自成一体。而当今世界的这个主义，那个主义自然同样不在话下。这或许正是中国官僚政治自秦以降，能够绵绵延续 2000 年的关键原因。正所谓“2000 年之政，秦政也，皆大盗也。”



(a) 原始图像 (b) 噪声图像 ($\mu = 0, \sigma = 20$)

图 1 初始图像



(a) $\lambda = \frac{1}{20^2}$ (b) $\lambda = \frac{5}{20^2}$ (c) $\lambda = \frac{10}{20^2}$ (d) $\lambda = \frac{20}{20^2}$

图 2 处理的像

每章标题按一级标题编排，每节标题按二级标题编排，每小节标题按三级标题编排。

1.1 参考文献

当然参考文献可以直接写 `bibitem`，虽然费点功夫，但是好控制，各种格式可以自己随意改写。

本模板推荐使用 `BIBTeX`，样式文件为 `thubib.bst`，基本符合学校的参考文献格式（如专利等引用未加详细测试）。看看这个例子，关于书的 [?, ?, ?]，还有这些 [?, ?, ?]，关于杂志的 [?, ?, ?]，硕士论文 [?, ?]，博士论文 [?, ?]，标准文件 [?]，会议论文 [?, ?]，技术报告 [?]。中文参考文献 [?] 应增 `lang=``zh''` 字段，以便进行相应处理。另外，这个 `bst` 对中文文献 [?] 的支持并不是十全十美，如果有不如意的地方，请手动修改 `bbl` 文件。

表 1 模板文件。如果表格的标题很长，那么在表格索引中就会很不美观，所以要像 `chapter` 那样在前面用中括号写一个简短的标题。这个标题会出现在索引中。

文件名	描述
<code>thuthesis.ins</code>	<code>L^AT_EX</code> 安装文件， <code>docstrip</code> ^a
<code>thuthesis.dtx</code>	所有的一切都在这里面 ^b 。
<code>thuthesis.cls</code>	模板类文件。
<code>thuthesis.cfg</code>	模板配置文。 <code>cls</code> 和 <code>cfg</code> 由前两个文件生成。
<code>thubib.bst</code>	参考文献 <code>Bibtex</code> 样式文件。
<code>thutils.sty</code>	常用的包和命令写在这里，减轻主文件的负担。

^a表格中的脚注

^b再来一个

“章”、“节”、“小节”的编号统一为：1、1.1、1.1.1。

四级以后的标题和编号的编排原则为：下级标题的显目程度不超过上一级，不重复或混淆。

编号与题目之间空一格。

1.2 二级标题

积土成山，风雨兴焉；积水成渊，蛟龙生焉；积善成德，而神明自得，圣心备焉。故不积跬步，无以至千里；不积小流，无以成江海。骐驎一跃，不能十步；弩马十驾，功在不舍。锲而舍之，朽木不折；锲而不舍，金石可镂。蚓无爪牙之利，筋骨之强，上食埃土，下饮黄泉，用心一也。蟹六跪而二螯，非蛇鳝之穴无可寄托者，用心躁也。

1.3 三级标题

平林漠漠烟如织，寒山一带伤心碧。暝色入高楼，有人楼上愁。王阶空伫立，宿鸟归飞急。何处是归程，长亭更短亭。

1.3.1 四级标题

君不见，黄河之水天上来，奔流到海不复回。君不见，高堂明镜悲白发，朝如青丝暮成雪。人生得意须尽欢，莫使金樽空对月。天生我材必有用，千金散尽还复来。烹羊宰牛且为乐，会须一饮三百杯。岑夫子，丹丘生，将进酒，君莫停。与君歌一曲，请君为我侧耳听。钟鼓馔玉不足贵，但愿长醉不复醒。古来圣贤皆寂寞，惟有饮者留其名。陈王昔时宴平乐，斗酒十千恣欢谑。主人何为言少钱，径须沽取对君酌。五花马，千金裘，呼儿将出换美酒，与尔同销万古愁。

1.3.2 四级标题

一屠晚归，担中肉尽，止有剩骨。途中两狼，缀行甚远。

屠惧，投以骨。一狼得骨止，一狼仍从。复投之，后狼止而前狼又至。骨已尽矣，而两狼之并驱如故。

屠大窘，恐前后受其敌。顾野有麦场，场主积薪其中，苫蔽成丘。屠乃奔倚其下，弛担持刀。狼不敢前，眈眈相向。

少时，一狼径去，其一犬坐于前。久之，目似瞑，意暇甚。屠暴起，以刀劈狼首，又数刀毙之。方欲行，转视积薪后，一狼洞其中，意将隧入以攻其后也。身已半入，止露尻尾。屠自后断其股，亦毙之。乃悟前狼假寐，盖以诱敌。

狼亦黠矣，而顷刻两毙，禽兽之变诈几何哉？止增笑耳。

1.3.3 又是一个四级标题

孔子东游，见两小儿辩斗。问其故。

一儿曰：“我以日始出时去人近，而日中时远也。”

一儿以日初出远，而日中时近也。

一儿曰：“日初出大如车盖，及日中则如盘盂，此不为远者小而近者大乎？”

一儿曰：“日初出沧沧凉凉，及其日中如探汤，此不为近者热而远者凉乎？”

孔子不能决也。

两小儿笑曰：“孰为汝多知乎？”

2 理论分析

2.1 问题重述

我们首先引入线性算子

$$R: L^2(\Omega) \longrightarrow L^2(\Omega)$$

$$Ru(x, y) = \int_{\Omega} K(x - x^1, y - y^1) u(x^1, y^1) dx^1 dy^1$$

其中 $K(x, \epsilon, y, \tau)$ 为脉冲响应函数, 并且我们总假定线性算子 R 是有界的。

若已知 u 的模糊版本 g , 则由方程 $g = Ru$ 求 u 是一个典型的反问题, 称为图像去模糊问题或图像去卷积。这里我们考虑含加性噪声的图像 u_0 , 含加性噪声的模糊图像 u_0 满足方程

$$u_0 = Ru + \eta$$

其中: η 表示高斯白噪声, 即

$$\int_{\Omega} \eta(x) = 0, \int_{\Omega} \eta^2(x) = \sigma^2$$

综上所述, 这里考虑的基本问题是: 已知初值数据 u_0 , 算子 $R: L^2(\Omega) \longrightarrow L^2(\Omega)$ 是有界线性算子, 图像噪声方差 σ^2 , 根据含噪声模糊图像模型

$$u_0 = Ru + \eta$$

重建或者复原 u , 特别的, 当 R 为恒同算子时, 基本问题就称为去噪问题。

2.2 问题分析

以下我们考虑当 R 为恒同算子时的去噪问题, 即

$$u_0 = u + \eta \tag{1}$$

其中: η 表示高斯白噪声, 即

$$E\eta(x) = 0, E\eta^2(x) = \sigma^2 \tag{2}$$

由噪声式 (2) 的假设导致了两种约束

$$\iint_{\Omega} u(x, y) dx dy = \iint_{\Omega} u_0(x, y) dx dy \tag{3}$$

$$\frac{1}{|\Omega|} \iint_{\Omega} (u(x, y) - u_0(x, y))^2 dx dy = \sigma^2 \tag{4}$$

其中 $|\Omega|$ 为图像区域 Ω 的面积。

由 Rudin、Osher 和 Fatemi 提出的 TV 图像复原模型，是图像复原中最成功的方法。与最小二乘复原模型相比，TV 模型最主要的差别是用梯度的 L1 范数替代了它的 L2 范数，也就是最小化全变分（Rudin 认为，有噪声图像的全变分比无噪声图像的全变分明显大，最小化全变分可以消除噪声）。

令 $F_1(u) = \int_{\Omega} 2|\nabla u|^{\frac{1}{2}} dx dy$ ，由于 $F_1(u)$ 具有平移不变性，即 $F_1(u+c) = F_1(u)$ ， c 为任意常数，那么第一个约束实际上是自满足的。因此，通常我们只需要考虑第二个拟合约束。通过引入拉格朗日乘子 λ ，可以定义一个新的能量泛函：

$$J(u) = \int_{\Omega} 2|\nabla u(x, y)|^{\frac{1}{2}} dx dy + \frac{\lambda}{2} \int_{\Omega} |u - u_0|^2 dx dy \quad (5)$$

其中，参数 λ 对平衡去噪与平滑起重要作用。这是一个泛函求极值问题，即变分问题。

2.3 PDE 的推导

从 (5) 式中可以看出该能量泛函为

$$J(u) = \int_{\Omega} F(x, y, u, \frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}) dx dy \quad (6)$$

型泛函，其中

$$F = 2|\nabla u(x, y)|^{\frac{1}{2}} + \frac{\lambda}{2}|u - u_0|^2 \quad (7)$$

由泛函求极值的必要条件，有欧拉—拉格朗日方程（PDE）

$$F_H - \frac{\partial}{\partial x}\{F_p\} - \frac{\partial}{\partial y}\{F_q\} = 0 \quad (8)$$

其中 $p = \frac{\partial u}{\partial x}, q = \frac{\partial u}{\partial y}$ 。所以对于 (7) 式有

$$F_H = \lambda(u - u_0), F_p = \frac{\frac{\partial u}{\partial x}}{|\nabla u|^{\frac{3}{2}}}, F_q = \frac{\frac{\partial u}{\partial y}}{|\nabla u|^{\frac{3}{2}}} \quad (9)$$

将上式代入式 (8) 有：

$$\begin{aligned} \lambda(u - u_0) - \left\{ \frac{\partial}{\partial x} \left\{ \frac{\frac{\partial u}{\partial x}}{|\nabla u|^{\frac{3}{2}}} \right\} + \frac{\partial}{\partial y} \left\{ \frac{\frac{\partial u}{\partial y}}{|\nabla u|^{\frac{3}{2}}} \right\} \right\} &= 0 \\ \lambda(u - u_0) - \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y} \right) \cdot \left(\frac{\frac{\partial u}{\partial x}}{|\nabla u|^{\frac{3}{2}}}, \frac{\frac{\partial u}{\partial y}}{|\nabla u|^{\frac{3}{2}}} \right) &= 0 \end{aligned}$$

综上得到该模型的欧拉—拉格朗日方程（PDE）：

$$-\nabla \cdot (|\nabla u|^{-\frac{3}{2}} \nabla u) + \lambda(u - u_0) = 0 \quad (10)$$

同理，对于能量泛函

$$J(u) = \int_{\Omega} \frac{1}{p} |\nabla u(x, y)|^p dx dy + \frac{\lambda}{2} \int_{\Omega} |u - u_0|^2 dx dy, 0 < p < 2 \quad (11)$$

我们有对应的欧拉—拉格朗日方程（PDE）：

$$-\nabla \cdot (|\nabla u|^{p-2} \nabla u) + \lambda(u - u_0) = 0 \quad (12)$$

2.4 模型的数值实现

通过前面的分析知道，求解变分问题式 (5) 与求解偏微分方程 (10) 是等价的。我们在这里通过差分迭代法直接求解相关的平衡方程（稳定解）。

首先对图像进行等间隔采样，设采样步长 $h = 1$ 。设目标像素为 $u(i, j)$, $A = \{(i, j+1), (i-1, j), (i, j-1), (i+1, j)\}$ 为目标像素 $u(i, j)$ 的四个邻域点位置的集合，如下图所示。 e, n, w, s 为对应的四个半像素点，它们不能直接从数字图像中得到。由于 $|\nabla u|^{\frac{3}{2}}$ 出现在分母，为了避免它为零，我们引入一个小的正参数 ϵ ，使得 $|\nabla u|_{\epsilon} = \sqrt{\epsilon^2 + |\nabla u|^2}$ 。

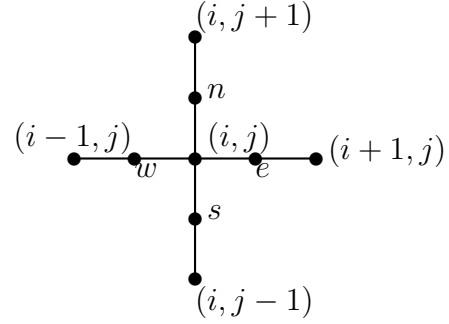


图 1: 目标像素 (i, j) 与它的邻域

只要保持 ϵ 足够小，便不会影响复原的性能。

为了对式 (10) 进行数值实现，我们先令

$$v = (v^1, v^2) = \frac{\nabla u}{|\nabla u|_{\epsilon}^{\frac{3}{2}}}$$

那么旋度 $\nabla \cdot$ 由差分表示为

$$\nabla \cdot v = \frac{\partial v^1}{\partial x} + \frac{\partial v^2}{\partial y} \approx (v_e^1 - v_w^1) + (v_n^2 - v_s^2) \quad (13)$$

下面给出 $v_e^1, v_w^1, v_n^2, v_s^2$ 的四个逼近：

$$v_e^1 = \frac{1}{|\nabla u_e|_{\epsilon}^{\frac{3}{2}}} \left[\frac{\partial u}{\partial x} \right]_e \approx \frac{1}{|\nabla u_e|_{\epsilon}^{\frac{3}{2}}} [u(i+1, j) - u(i, j)]$$

$$|\nabla u_e|_{\epsilon}^{\frac{3}{2}} \approx (\epsilon^2 + [u(i+1, j) - u(i, j)]^2 + \left[\frac{u(i+1, j+1) - u(i+1, j-1)}{2} \right]^2)^{\frac{3}{4}}$$

同理可得：

$$v_w^1 = \frac{1}{|\nabla u_w|_{\epsilon}^{\frac{3}{2}}} \left[\frac{\partial u}{\partial x} \right]_w \approx \frac{1}{|\nabla u_w|_{\epsilon}^{\frac{3}{2}}} [u(i, j) - u(i-1, j)]$$

$$v_n^2 = \frac{1}{|\nabla u_n|_{\epsilon}^{\frac{3}{2}}} \left[\frac{\partial u}{\partial y} \right]_n \approx \frac{1}{|\nabla u_n|_{\epsilon}^{\frac{3}{2}}} [u(i, j+1) - u(i, j)]$$

$$v_s^2 = \frac{1}{|\nabla u_s|_{\epsilon}^{\frac{3}{2}}} \left[\frac{\partial u}{\partial y} \right]_s \approx \frac{1}{|\nabla u_s|_{\epsilon}^{\frac{3}{2}}} [u(i, j) - u(i, j-1)]$$

所以有

$$-\nabla \cdot (|\nabla|^{-\frac{3}{2}} \nabla u) = -\nabla v = \sum_{q \in A} |\nabla u_q|_\epsilon^{-\frac{3}{2}} [u(i, j) - u(q)] \quad (14)$$

其中，我们定义

$$\text{若 } q = (i - 1, j), \text{ 则 } p = (i - \frac{1}{2}, j) \quad (15)$$

$$\text{若 } q = (i + 1, j), \text{ 则 } p = (i + \frac{1}{2}, j) \quad (16)$$

$$\text{若 } q = (i, j - 1), \text{ 则 } p = (i, j - \frac{1}{2}) \quad (17)$$

$$\text{若 } q = (i, j + 1), \text{ 则 } p = (i, j + \frac{1}{2}) \quad (18)$$

因此式 (10) 可转化为

$$\sum_{q \in A} |\nabla u_p|_\epsilon^{-\frac{3}{2}} [u(i, j) - u(q)] + \lambda(i, j) [u(i, j) - u_0(i, j)] = 0 \quad (19)$$

则可推出：

$$u(i, j) = \frac{\sum_{q \in A} |\nabla u_p|_\epsilon^{-\frac{3}{2}}}{\lambda(i, j) + \sum_{q \in A} |\nabla u_p|_\epsilon^{-\frac{3}{2}}} u(q) + \frac{\lambda(i, j)}{\lambda(i, j) + \sum_{q \in A} |\nabla u_p|_\epsilon^{-\frac{3}{2}}} u_0(i, j) \quad (20)$$

令

$$\omega_q = \frac{1}{|\nabla u_p|_\epsilon^{\frac{3}{2}}} \quad (21)$$

$$h_q = \frac{\omega_q}{\lambda(i, j) + \sum_{q \in A} \omega_q} \quad (22)$$

$$h_0 = \frac{\lambda(i, j)}{\lambda(i, j) + \sum_{q \in A} \omega_q} \quad (23)$$

那么式 (20) 变为

$$u(i, j) = \sum_{q \in A} h_q u(q) + h_0 u_0(i, j) \quad (24)$$

其中

$$h_0 + \sum_{q \in A} h_q = 1 \quad (25)$$

采用 Gauss-Seidel 迭代法，在每一步 n ：

$$u(i, j)^{(n)} = \sum_{q \in A} h_q^{(n-1)} u(q)^{(n-1)} + h_0^{(n-1)} u_0(i, j) \quad (26)$$

綜上有图像去噪的算法如下：

```
初始化  $u^{(0)}$ , 通常取  $u^{(0)} = u_0$  ;  
for  $n = 1, 2, \dots$   
  for  $i = 1, 2, \dots, M$   
    for  $j = 1, 2, \dots, N$   
      根据式 (21), (22), (23) 分别计算以  $(i, j)$  为中心点  
      的  $\omega_q, h_q, h_0$ , 然后利用式 (26) 计算  $u^{(n)}(i, j)$ ;  
    End  
  End  
End
```

3 实验

假设噪声为高斯白噪声，并已知其均值 0，标准差为 σ 。以下给出初始图像及噪声图像：



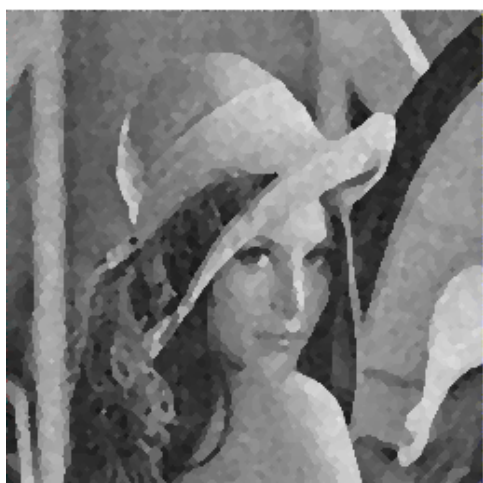
(a) 原始图像



(b) 噪声图像 ($\mu = 0, \sigma = 20$)

3.1 实验 1

分别取 $\lambda = \frac{1}{20^2}, \frac{10}{20^2}, \frac{5}{20^2}, \frac{20}{20^2}, \epsilon = 0.00001$ ，迭代 50 次，得到结果如下：



(c) $\lambda = \frac{1}{20^2}$



(d) $\lambda = \frac{5}{20^2}$

(e) $\lambda = \frac{10}{20^2}$ (f) $\lambda = \frac{20}{20^2}$

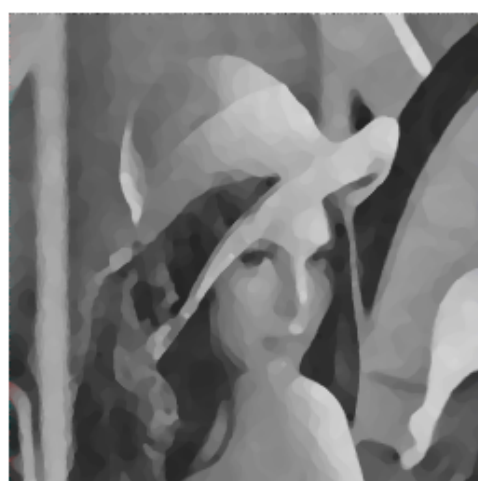
由上结果可以知道因为参数 λ 对平衡去噪与平滑起到重要作用，因此参数 λ 对于复原效果很重要。 λ 太小时，图像过度平滑以致边缘严重模糊；相反， λ 太大时，图像去噪效果不好。

3.2 实验 2

下面将本文算法与 TV 复原算法作比较，取 $\lambda = \frac{5}{20^2}$, $\epsilon = 0.00001$ ，迭代 50 次。以下是所得结果：



(g) 本文算法所得结果

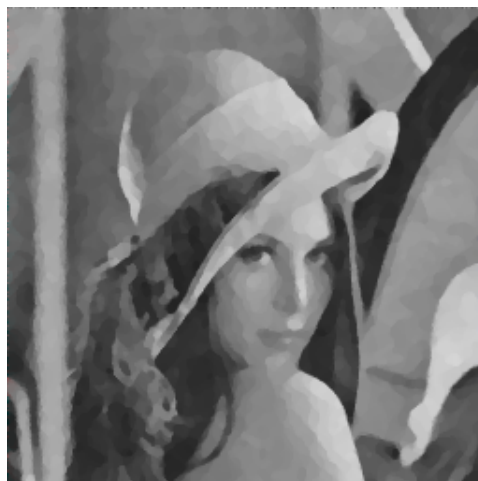


(h) TV 复原算法所得结果

取 $\lambda = \frac{10}{20^2}$, $\epsilon = 0.00001$ ，迭代 50 次。以下是所得结果：



(i) 本文算法所得结果



(j) TV 复原算法所得结果

从上结果可以看出在相同参数条件下，本文算法相对于 TV 复原算法对于边缘的保持较好。

4 结论与展望

该模板是一个开源项目，旨在提供符合西安交通大学有关部门要求的学位论文模板。您当前看到的文件是的示例排版文档。项目目前托管在 Google Code: 采用 Mercurial 管理源代码。访问项目的网站，了解更多信息。

如果你是本科生，请和系里联系以确定可以使用完成论文。研究生请和西安交大研究生院学位办 (周主任) 联系：

电话 82668899

办公地址 教学主楼 1311 室

电子邮件

附录 1 Matlab 程序

以下为 Matlab 程序

```
close all;
clc;
clear all;
lam=10/20^2;
eps=0.00001;
p=1/2;
I=imread('001.bmp');
J=I;
MASK=J;
[width,height] = size(MASK);
U=double(J);
V=double(J);
T=double(J);
n=1;
IterTimes=100;
while n<=IterTimes
for i=2:(width-1)
for j=2:(height-1)
gridUe2=(eps^2+(V(i+1,j)-V(i,j))^2+
(1.0/4)*(V(i+1,j+1)-V(i+1,j-1))^2)^((2-p)/2);
gridUw2=(eps^2+(V(i,j)-V(i-1,j))^2+
(1.0/4)*(V(i-1,j+1)-V(i-1,j-1))^2)^((2-p)/2);
gridUn2=(eps^2+(V(i,j)-V(i,j+1))^2+
(1.0/4)*(V(i+1,j+1)-V(i-1,j+1))^2)^((2-p)/2);
gridUs2=(eps^2+(V(i,j)-V(i,j-1))^2+
(1.0/4)*(V(i+1,j-1)-V(i-1,j-1))^2)^((2-p)/2);
we=1/gridUe2;
ww=1/gridUw2;
wn=1/gridUn2;
ws=1/gridUs2;
sum=we+ww+wn+ws;
h0=lam/(lam+sum);
U(i,j)=(ww*V(i-1,j)+we*V(i+1,j)+wn*V(i,j+1)+
ws*V(i,j-1))/(lam+sum)+h0*T(i,j);
```



```
V(i ,j)=U(i ,j );  
end  
end  
n=n+1;  
V=U;  
end  
V=U;  
figure ;  
imshow( uint8(V) );
```

附录 2 还是那个 Matlab 程序

以下为 Matlab 程序

```
close all;
clc;
clear all;
lam=10/20^2;
eps=0.00001;
p=1/2;
I=imread('001.bmp');
J=I;
MASK=J;
[width,height] = size(MASK);
U=double(J);
V=double(J);
T=double(J);
n=1;
IterTimes=100;
while n<=IterTimes
for i=2:(width-1)
for j=2:(height-1)
gridUe2=(eps^2+(V(i+1,j)-V(i,j))^2+
(1.0/4)*(V(i+1,j+1)-V(i+1,j-1))^2)^((2-p)/2);
gridUw2=(eps^2+(V(i,j)-V(i-1,j))^2+
(1.0/4)*(V(i-1,j+1)-V(i-1,j-1))^2)^((2-p)/2);
gridUn2=(eps^2+(V(i,j)-V(i,j+1))^2+
(1.0/4)*(V(i+1,j+1)-V(i-1,j+1))^2)^((2-p)/2);
gridUs2=(eps^2+(V(i,j)-V(i,j-1))^2+
(1.0/4)*(V(i+1,j-1)-V(i-1,j-1))^2)^((2-p)/2);
we=1/gridUe2;
ww=1/gridUw2;
wn=1/gridUn2;
ws=1/gridUs2;
sum=we+ww+wn+ws;
h0=lam/(lam+sum);
U(i,j)=(ww*V(i-1,j)+we*V(i+1,j)+wn*V(i,j+1)+
ws*V(i,j-1))/(lam+sum)+h0*T(i,j);
```

```
V(i ,j)=U(i ,j );  
end  
end  
n=n+1;  
V=U;  
end  
V=U;  
figure ;  
imshow( uint8(V) );
```

致 谢

衷心感谢西安交通大学数学与统计学院的各位老师。对于毕业设计（论文）的指导教师，对毕业设计（论文）提过有益的建议或给予过帮助的同学、同事与集体，都应在论文的结尾部分书面致谢，言辞应恳切、实事求是。应注明受何种基金支持（没有可不写）。