

第七届数学中国数学建模网络挑战赛

地址：数学中国数学建模网络挑战赛组委会
电话：0471-4969085

邮编：010021

网址：www.tzmcm.cn
Email：2014@tzmcm.cn

第七届“认证杯”数学中国

数学建模网络挑战赛

承 诺 书

我们仔细阅读了第七届“认证杯”数学中国数学建模网络挑战赛的竞赛规则。

我们完全明白，在竞赛开始后参赛队员不能以任何方式（包括电话、电子邮件、网上咨询等）与队外的任何人（包括指导教师）研究、讨论与赛题有关的问题。

我们知道，抄袭别人的成果是违反竞赛规则的，如果引用别人的成果或其他公开的资料（包括网上查到的资料），必须按照规定的参考文献的表述方式在正文引用处和参考文献中明确列出。

我们郑重承诺，严格遵守竞赛规则，以保证竞赛的公正、公平性。如有违反竞赛规则的行为，我们接受相应处理结果。

我们允许数学中国网站(www.madio.net)公布论文，以供网友之间学习交流，数学中国网站以非商业目的的论文交流不需要提前取得我们的同意。

我们的参赛队号为：

参赛队员（签名）：

队员 1：唐天全

队员 2：李栖楠

队员 3：李凡

参赛队教练员（签名）： 余红伟

参赛队伍组别(中学|专科|本科|研究生)： 本科

第七届数学中国数学建模网络挑战赛

地址：数学中国数学建模网络挑战赛组委会
电话：0471-4969085

邮编：010021

网址：www.tzmcm.cn
Email：2014@tzmcm.cn

第七届“认证杯”数学中国

数学建模网络挑战赛

编号专用页

参赛队伍的参赛队号：（请各个参赛队提前填写好）：

1083

竞赛统一编号（由竞赛组委会送至评委团前编号）：

竞赛评阅编号（由竞赛评委团评阅前进行编号）：

第七届数学中国数学建模网络挑战赛

地址：数学中国数学建模网络挑战赛组委会
电话：0471-4969085

邮编：010021

网址：www.tzmcm.cn
Email：2014@tzmcm.cn

2014 年第七届“认证杯”数学中国 数学建模网络挑战赛第二阶段论文

题 目 基于双线性插值及边缘矢量化的位图放大法

关 键 词 双线性插值 双三次插值 轮廓矢量化 平均梯度

傅立叶幅度 图像放大

摘 要：

随着图形技术的发展，由像素点组合而成的位图已经无法满足人们对于高放大倍数时图片清晰度的要求，在很多情况下，人们需要的图形均是以位图的放大形式呈现的，而受到位图放大后图像模糊或马赛克现象的限制，图像的不失真放大技术已成为当今图像处理技术的关键之一。

本文针对位图放大后图像质量下降的问题，建立了三种不同的数学模型，设计了相应的放大位图算法，以大小为 682×1024 的灰度图像作为目标位图分别进行 4×4 倍放大测试。

首先，针对传统的双线性插值图像放大算法存在的边缘模糊问题，建立了基于双线性插值法的位图放大模型，提出一种改进算法：先计算插值点的双线性插值和最近邻点插值，然后以 4 个邻点的灰度方差构造权重，将二种插值进行加权融合获得最终插值结果。该算法结果的均方误差为 23.56，平均梯度为 1.6503，程序耗时 1.2314s。模型既考虑到插值点与邻点之间距离关系，又考虑到邻点的灰度分布特性，有效地提高了放大图像的质量。实验结果验证了算法的有效性。

其次，本文建立了基于双三次插值法的位图放大模型。采用双三次插值实现图像放大具有较高的图像质量，但运算量很大。在分析常用插值算法的基础上，本文提出一种基于区域的双三次插值算法。该算法避免进行图像分割，通过被插值点四邻域像素的均值来划分图像的平坦区域和纹理细节复杂区域，采用不同的插值算法进行计算。实验结果表明，其均方误差为 23.98，平均梯度为 1.9173，程序耗时 2.6234s，与传统的双三次插值算法相比，该算法在保持放大后图像质量的同时，运算量降低 10% 以上，具有一定的实用价值。

最后，本文在第一阶段位图矢量化问题的基础上，和模型二相结合，建立了结合图像轮廓矢量化与双三次插值的综合位图放大模型。该算法将图像分割为简单图像区域与复杂图像区域，使用 Bezier 曲线拟合简单图像部分的轮廓，从而进行不失真地矢量放大，使用双三次插值法对复杂图像部分进行插值计算。实验结果表明，此方法的均方误差为 22.36，平均梯度为 2.1021，程序耗时 3.0123s，且改善放大图像的质量较好。

为定量分析验证三种模型的放大效果，本文采用均方误差、平均梯度和傅立叶幅度作为评价指标，相关结果数据见表 2、图 11 及图 12。结果表明，双线性插值法对放大图像的质量改善效果最差，双三次插值法次之，将图像轮廓矢量化与双三次插值相结合的综合位图放大法最佳，但是综合位图放大法也存在计算量大、程序耗时长的问题。

本文从实际出发，利用多种模型和算法解决了位图放大失真的问题，提出的定量指标能够精确分析放大图像的效果，促进了图像处理技术的研究与发展，对实际生产生活中位图放大的应用具有借鉴意义。

第七届数学中国数学建模网络挑战赛

地址：数学中国数学建模网络挑战赛组委会

电话：0471-4969085

邮编：010021

网址：www.tzmcm.cn

Email：2014@tzmcm.cn

参赛队号： 1083

参赛密码 _____ (由组委会填写)

所选题目： B 题

英文摘要（选填）

With the development of graphics technology, bitmap made by pixel has been unable to satisfy people's requirement of high magnification. In many cases, people need the megascopic bitmap. However, enlarging the bitmap would make the image fuzzy or even arise Mosaic phenomenon. Therefore, the undistorted technology to enlarge the image has become one of the key technology of image processing.

Considering the decline of the enlarged image quality of bitmap, this paper established 3 kinds of mathematical models and designed the corresponding bitmap algorithms. We regard a gray image with size of 682×1024 as the target bitmap, and enlarge it by 4×4 separately.

Firstly, in views of the traditional bilinear interpolation algorithm problems about the edge of the fuzzy image, this paper propose an improved algorithm. Compute the interpolation points of bilinear interpolation and the nearest neighbor interpolation, then structure the weight by using the gray variance of four neighboring points, and the final result by weight fusion can be obtained. The algorithm results show that the MSE is 23.56, the average gradient is 1.6503, and 1.2314s are required to run the result. This model consider both the relationship with the distance between adjacent points of interpolation points and the gray level distribution characteristics of adjacent points. Therefore, it improves the quality of the magnified image effectively. Moreover, the experimental results verity the effectiveness of the algorithm.

Secondly, this paper established the bitmap amplification model based on double three interpolation. By analyzing the common interpolation algorithm, this paper proposes region double three interpolation algorithm. This algorithm avoids image segmentation and uses different interpolation algorithm to clarify regions. The experimental results showed that the MSE is 23.98, the average gradient is 1.9173, and it takes 2.6234s. Compared with the traditional double three interpolation algorithm, this algorithm can keep the enlarged image quality, also reduce the computational complexity by more than 10%.

Finally, a model combining the vector quantization of bitmap in phase 1 with model 2 is established. The algorithm divides an image into simple area and complex area. Bezier curve fitting is used for the outline of simple area, while double three interpolation method for complex image. The experimental results show that the MSE is 22.36, the average gradient is 2.1021, and it takes 3.0123s. Moreover, the quality of the magnified image used this algorithm is the best.

To analyze the amplification effect of the three kind of models quantitatively, this paper uses the mean square error(MSE), the average gradient and Fourier amplitude as evaluation index. The results data are shown in table 2, figure 11, figure12 and figure13. The results show that the effect of the bilinear interpolation method to enlarge the image is the worst, and the method of combining the image contour vector quantization and double three interpolation comprehensive bitmap amplification is the best. However, there are large amount of calculation in the combining model, and it takes the longest time to obtain the image.

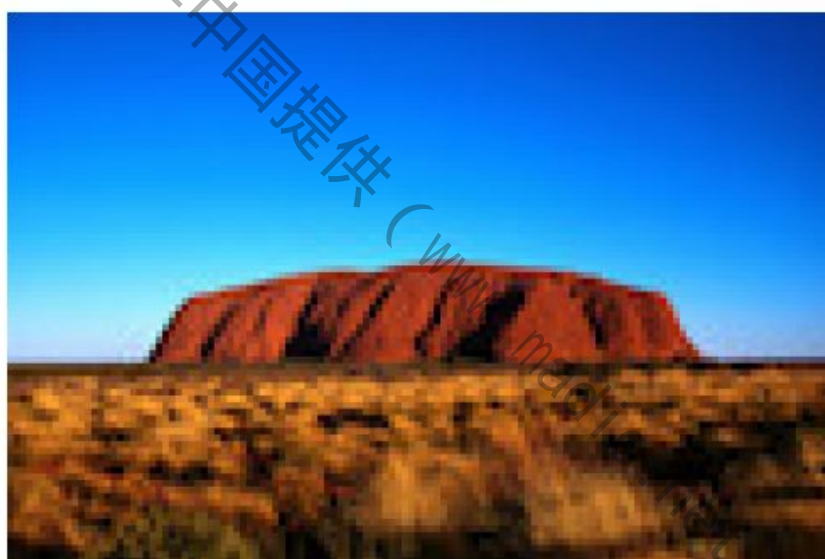
一、问题重述

图形（或图像）在计算机里主要有两种存储和表示方法。矢量图是使用点、直线或多边形等基于数学方程的几何对象来描述图形，位图则使用像素来描述图像。一般来说，照片等相对杂乱的图像使用位图格式较为合适，矢量图则多用于工程制图、标志、字体等场合。矢量图可以任意放缩，图形不会有任何改变。而位图一旦放大后会产生较为明显的模糊，线条也会出现锯齿边缘等现象。见图一。



图一：栅格化示意图

位图在放大时，图像质量常会有所下降，如容易产生较为明显的模糊或马赛克等现象（见图二）。请你建立合理的数学模型，来设计一个放大位图的算法，使图像在被放大后仍能尽量保持较好的图像质量。



图二：低分辨率的位图经放大后可能产生的马赛克效果

二、问题分析

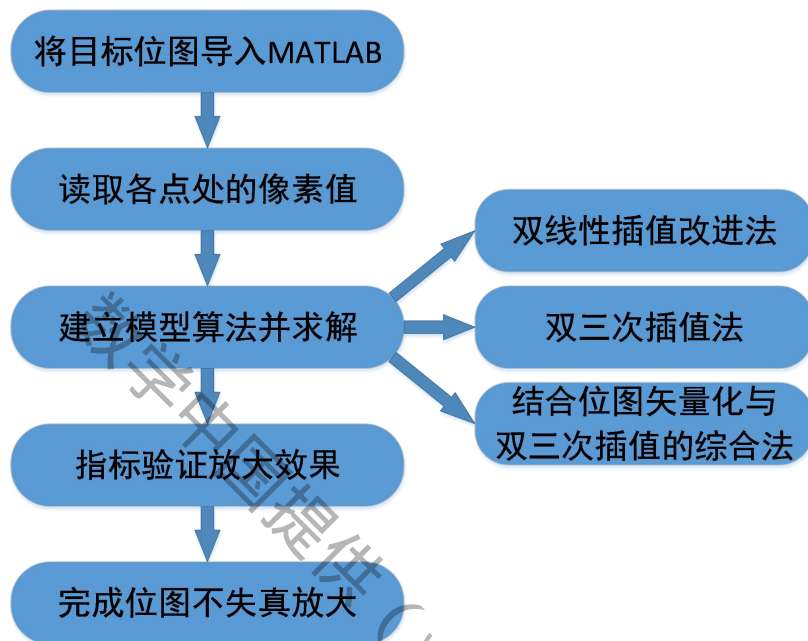
2.1 问题背景

图像，作为生产生活中随处可见的元素，在人们的社会生活及科研生产活动中起到了基本的表达与展示的作用。图像信息是人类认识世界的重要知识来源，随着科学技术的发展，人们所获取的图像形式的信息将会变得越来越多，而图像信息处理技术的发展也更为深入、广泛和迅速。目前，图像信息处理技术已广泛的应用于许多社会领域，如工业、农业、国防军事、社会公安、科研、生物医药、通信邮电等等。图像理论与技术涉及众多的学科，如数学、物理学、信号处理、控制论、模式识别、人工智能、生物学等等，它是一门兼具交叉性和开放性的学科。^[1-3]因此，图像处理已经成为一门引人注目，前景广阔的新型学科，并正在向更高更深层次发展^[4]。其中，数字图像处理技术的发展

涉及信息科学计算机科学数学物理学以及生物学等学科因此数理及相关的边缘学科对图像处理科学的发展有越来越大的影响^[5]。

2.2 问题分析

本题的要求是设计位图放大算法,使图像在被放大后仍能尽量保持较好的图像质量。根据要求,本文建立了三种不同的数学模型,通过设计对应的算法分别对问题进行了求解,流程图如下。



图三：问题分析流程图

首先,建立基于双线性插值法的放大模型。双线性插值法是较为普遍的一种算法,在此基础上,本文对模型进行改进,通过对图像的边缘细节区域和平滑区域分别采用最近邻点插值法和双线性插值法,将两者的各自优点充分发挥,达到位图放大的目的。其次,建立基于双三次插值法的放大模型。由于双线性插值的改进算法精度不够高,此模型不但考虑了4个直接相邻点的灰度影响,还利用了带采样像素点周围16个点的灰度值进行三次插值,充分考虑各邻点间灰度值变化率的影响,通过目标位图像素值的复杂矩阵运算,提高图像放大算法的精度,从而可以更好地改善放大图像的质量。最后,将第一阶段的轮廓矢量化和双三次插值法相结合,建立综合放大模型。考虑到第二阶段要处理的位图是内部图像较为复杂的彩色图像,针对目标位图简单图像部分和复杂图像部分,本模型分别采用第一阶段提出的Bezier曲线拟合法对位图轮廓进行矢量化和双三次插值法,将第一阶段的问题和第二阶段有机地结合起来,综合得到较为理想的位图放大效果。

在定性判断位图放大效果的基础上,本文采用均方误差、平均梯度和傅里叶幅度作为定量评价指标,进一步理论地分析各种模型对位图放大的实际效果。

三、问题假设及符号说明

3.1 问题假设

(1) 假设文件中像素点的值等于图象的灰度值;

- (2) 假设所处理的位图都是低频图像，亦或噪点比较小的图像；
 (3) 假设图像简单区域主要轮廓的灰度值和噪点灰度值有较大区别；

3.2 符号说明

符号名称	符号意义
$g(x, y)$	插值后坐标 (x, y) 处的灰度值
$f(x, y)$	坐标 (x, y) 处的灰度值
$[x]$	不大于 x 的整数
$[y]$	不大于 y 的整数
σ	标准差
w_i	加权系数
Var	方差
I	图像像素的值
\vec{V}_i	特征多边形顶点的位置矢量
$J_{n,i}$	伯恩斯坦基函数

四、基于双线性插值的位图放大模型

图像放大是图像处理和应用中的一种常见的操作，也是一些其他更复杂处理的基础，其主要目的是提高放大后图像的分辨率，以满足人们的视觉享受或实际应用要求。插值是实现图像放大的最常用的方法，也是图像超分辨率重建的一个关键步骤^[13]。常见的位图放大算法有最邻近插值法、双线性插值法等等，由于最邻近插值法原理简单且放大后图片的效果不佳，因此本文不加以讨论。为了保证输出的放大位图的质量，使图像模糊和马赛克现象尽量减少，本文分别通过建立基于双线性插值法、双三次插值法、综合位图矢量化和双三次插值法的三种数学模型分别对位图放大的过程及结果进行分析，以示例图片的原图为例进行放大处理，并对多种算法模型进行了比较。

4.1 基于双线性插值的位图放大模型

近年来为了获取更好的放大效果，人们将各种数学工具应用到图像放大领域，提出了基于小波变换、偏微分方程和变分泛函等^[9-11]的新型图像放大方法，这些新型方法能够较好地保持放大图像的边缘特征和细节信息，但是算法复杂，计算量大，不利于图像

参赛队号#1083

的实时放大处理。双线性插值法具有算法简单、易于实现、计算量小的特点，但该算法只考虑到插值点与邻点之间距离关系，存在放大图像边缘模糊的问题。本文在分析的基础上，提出了一种改进算法，采用数据融合将线性插值法和最近邻点插值法相结合，有效地提高了放大图像边缘的清晰度。

4.1.1 最近邻点插值和双线性插值

● 最近邻点插值

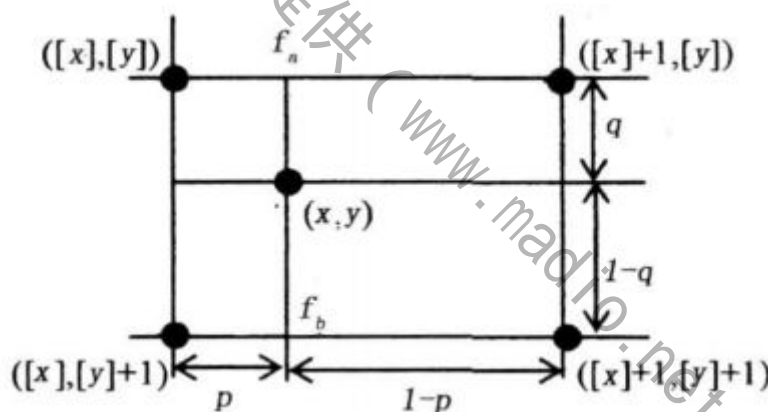
最近邻点插值是一种最简单的插值方法，它取插值点的4个邻点中距离最近的邻点灰度值作为该点的灰度值。设插值点 (x, y) 到周边4个邻点 $f_i(x, y)$ ($i=1,2,3,4$)的距离为 d_i ($i=1,2,3,4$)，则

$$g(x, y) = f_j(x, y), d_j = \min\{d_1, d_2, d_3, d_4\}, j=1,2,3,4 \quad (1)$$

最近邻点插值法存在马赛克和锯齿现象，单独使用时放大图像的效果差。但它是一种非线性插值方法，具有高通滤波保护边缘的特性。

● 双线性插值

双线性插值法用插值点周围4个邻点的灰度值加权内插作为该点的灰度值，它可分解为二次一维线性插值，见下图。



图四：双线性插值法

先进行第一次线性插值计算 f_a, f_b 值。

$$f_b = f([x], [y]+1) + (f([x]+1, [y]+1) - f([x], [y]+1)) \times p \quad (2)$$

再由 f_a, f_b 进行第二次线性插值计算 (x, y) 处插值结果 $g(x, y)$

$$\begin{aligned} g(x, y) &= f_a + (f_b - f_a) \times q = \\ &= (1-p)(1-q) \cdot f([x], [y]) + p(1-q)f([x]+1, [y]) + \\ &= p(1-q)f([x]+1, [y]) + q(1-p) \cdot f([x], [y]+1) + pqf([x]+1, [y]+1) \end{aligned} \quad (3)$$

式中， $g(x, y)$ 为插值后坐标 (x, y) 处的灰度值；

$f(x, y)$ 为插值前坐标 (x, y) 处的灰度值；

$[x], [y]$ 分别为不大于 x, y 的整数。

由上式可知，插值点的灰度值是由周围 4 个邻点的灰度值加权获得，权重由插值点与邻点的距离比决定。因此，双线性插值法是一种低通滤波器，会使图像中的高频边缘细节成分丢失，造成插值后图像的边缘模糊。

4.1.2 改进算法

最近邻点插值法具有高通滤波特性，在边缘细节区域采用最近邻点插值法有利于保护边缘。而双线性插值法适用于图像中像素之间的相关性较强、灰度变化缓慢、灰度的变化与距离近似呈线性关系的平滑区域，不适用于边沿细节区域。因此本文对图像的边缘细节区域和平滑区域分别采用最近邻点插值法和双线性插值法。另外，由于平滑区域与边缘细节区域在实际操作中不易严格区分，本文采用数据融合技术^[12]提出了一种基于以上插值法的改进算法。

首先，计算插值点的双线性和最近邻点二种插值结果，再计算插值点的 4 个邻点像素的灰度方差，由灰度方差构造权重系数，将二种插值进行加权融合，获得最终插值结果。邻点的方差越小，表示该邻域越平滑，双线性插值的权重增大，最近邻点插值的权重减小。该算法既考虑到插值点与邻点之间距离关系，又考虑到邻点的灰度分布特性，可有效提高放大图像的质量，减少放大图像模糊或者马赛克现象的产生。

首先，根据式 (2) 和式 (3) 计算插值点 (x, y) 的双线性插值 $g_i(x, y)$ 和最近邻点插值 $g_n(x, y)$ ，并计算插值点周边 4 个邻点 $f_i(x, y) (i=1, 2, 3, 4)$ 的标准差

$$\sigma = \sqrt{\frac{1}{4} \sum_{i=1}^4 [f_i(x, y) - f(x, y)]^2} \quad (4)$$

式中，

$$f(x, y) = \frac{1}{4} \sum_{i=1}^4 f_i(x, y) \quad (5)$$

再对 $g_i(x, y)$ 和 $g_n(x, y)$ 进行加权融合，得到最终插值结果

$$g(x, y) = w_i \times g_i(x, y) + w_n \times g_n(x, y) \quad (6)$$

式中， w_i 和 w_n 为加权系数，计算式如下：

$$\begin{cases} w_i = \frac{1 + e^{-ku}}{2} \\ w_n = 1 - w_i \end{cases} \quad (7)$$

式中， k 为常数，取值范围为 0~002，调节 k 值可改变算法的插值性能，在实际工程应用中，针对不同图像选择合适 k 值使放大效果达到最佳。

4.2 模型的求解与结果分析

4.2.1 模型的求解

根据上文，通过 MATLAB 对模型进行求解，具体步骤如下。

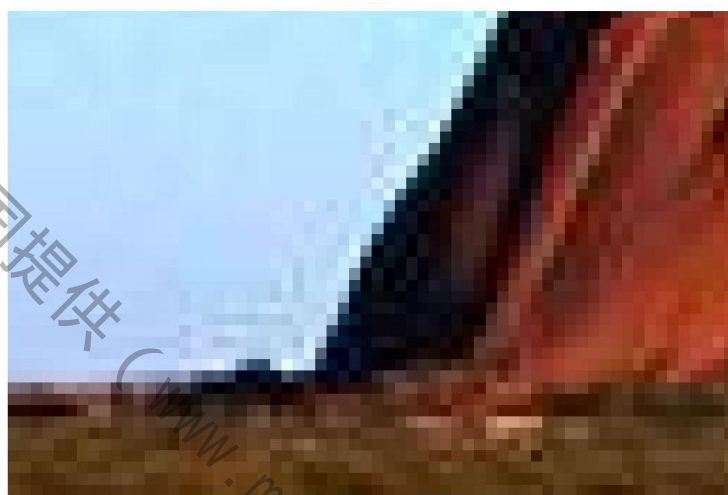
- Step 1: 将示例位图导入 MATLAB，并读取位图各像素点的灰度值；
- Step 2: 根据式 (2) 和式 (3) 计算插值点 (x, y) 的双线性插值 $g_l(x, y)$ 和最近邻点插值 $g_n(x, y)$ ；
- Step 3: 计算插值点周边 4 个邻点 $f_i(x, y)(i=1,2,3,4)$ 的标准差；
- Step 4: 对 $g_l(x, y)$ 和 $g_n(x, y)$ 进行加权融合，得到最终插值结果；
- Step 5: 根据插值结果，选择合适的常数 k 实现位图的放大。

4.2.2 结果分析

选用大小为 682×1024 的灰度图像作为原始对象，进行 4×4 倍放大测试，结果如下。



(a) 原图全景图



(b) 原图局部图



(c) 双线性插值法全景图



(d) 双线性插值法局部图

图五：图像放大效果对比图

由上图对比可以看出，原始位图放大后的图像质量较差，尤其是局部图像内的边缘锯齿化和马赛克现象明显，图像较为模糊；通过双线性插值法对位图进行放大处理后的

参赛队号#1083

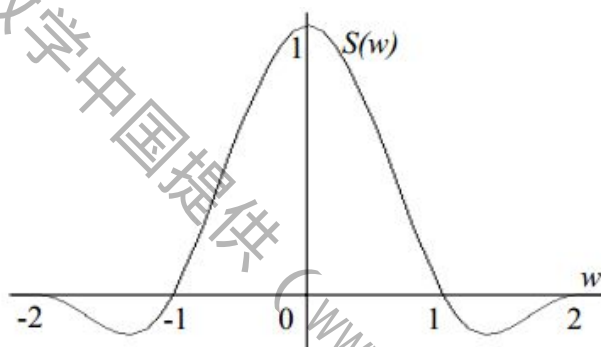
图像质量较好，图像的轮廓较为连贯、且锯齿化和马赛克的现象没有原图放大后的明显，图像颜色过渡平缓、图像较为清晰。因此，通过双线性插值法的位图放大方法可以有效地避免放大图像模糊或马赛克现象。

五、基于双三次插值法的位图放大模型

5.1 基于双三次插值法的位图放大模型

双线性插值法仅考虑浮点坐标 $(x+u, y+v)$ 4 个直接邻点对它的影响，而双三插值法则考虑该点周围的 16 个邻点。虽然双三次插值法的位图放大效果较好，但运算量远大于前者。因此，本文主要针对其特点进行改进，提出了基于区域的双三次插值算法。

双三次插值又称立方卷积插值^[14]。三次卷积插值是一种更加复杂的插值方式。该算法利用待采样点周围 16 个点的灰度值作三次插值，不仅考虑到 4 个直接相邻点的灰度影响，而且考虑到各邻点间灰度值变化率的影响。三次运算可以得到更接近高分辨率图像的放大效果，但也导致了运算量的急剧增加。这种算法需要选取插值基函数来拟合数据，其最常用的插值基函数如下图。



图六：双三插值基函数

其表达式如下：

$$S(w) = \begin{cases} 1 - 2|w|^2 + |w|^3 & |w| < 1 \\ 4 - 8|w| + 5|w|^2 - |w|^3 & 1 \leq |w| < 2 \\ 0 & |w| \geq 2 \end{cases} \quad (8)$$

目标坐标的像素值可由如下的公式得到：

$$f(i+u, j+v) = ABC \quad (9)$$

其中，A、B、C 均为矩阵，其形式如下：

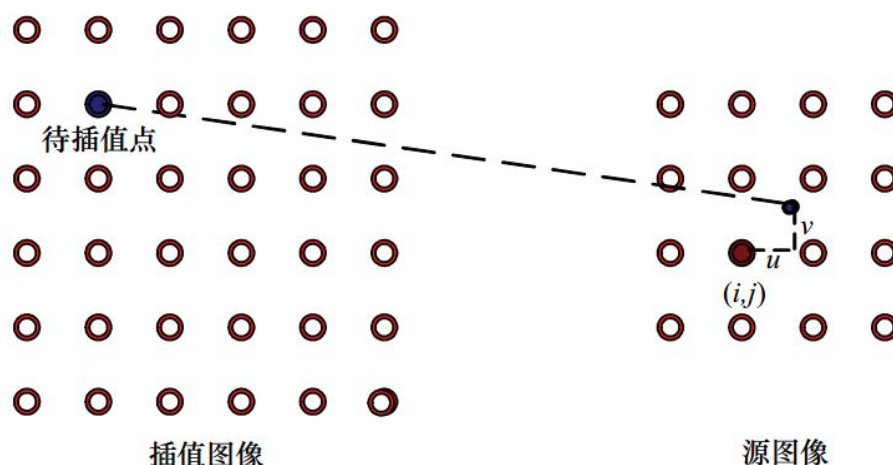
$$A = [S(1+u) \quad S(u) \quad S(1-u) \quad S(2-u)] \quad (10)$$

$$B = \begin{bmatrix} f(i-1, j-2) & f(i, j-2) & f(i+1, j-2) & f(i+2, j-2) \\ f(i-1, j-1) & f(i, j-1) & f(i+1, j-1) & f(i+2, j-1) \\ f(i-1, j) & f(i, j) & f(i+1, j) & f(i+2, j) \\ f(i-1, j+1) & f(i, j+1) & f(i+1, j+1) & f(i+2, j+1) \end{bmatrix} \quad (11)$$

$$C = [S(1+v) \quad S(v) \quad S(1-v) \quad S(2-v)]^T \quad (12)$$

参赛队号#1083

其中， $f(i, j)$ 表示源图像 (i, j) 处像素点的灰度值，如下图所示。



图七：双三次插值映射示意图

5.2 模型求解与结果分析

5.2.1 模型的求解

根据上文，通过 MATLAB 对模型进行求解，具体步骤如下。

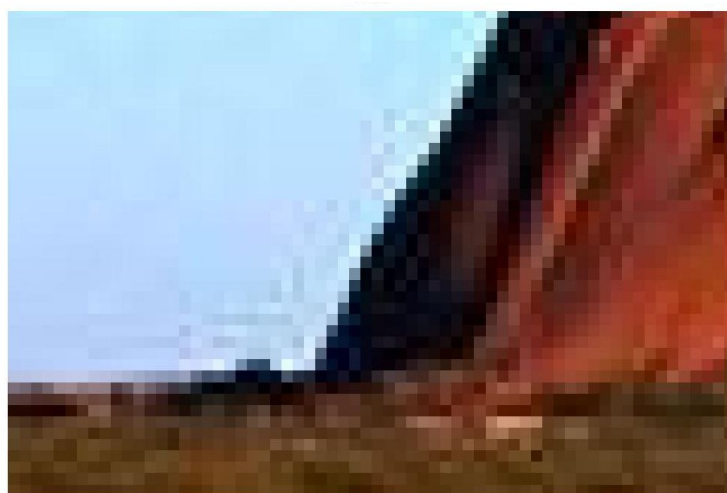
- Step 1: 将示例位图导入 MATLAB，并读取位图各像素点的灰度值；
- Step 2: 根据式 (2) 和式 (3) 计算插值点 (x, y) 的双线性插值 $g_l(x, y)$ 和最近邻点插值 $g_n(x, y)$ ；
- Step 3: 计算插值点周边 16 个邻点 $f_i(x, y) (i=1, 2, 3, 4)$ 的标准差；
- Step 4: 对 $g_l(x, y)$ 和 $g_n(x, y)$ 进行加权融合，得到最终插值结果；
- Step 5: 根据插值结果，选择合适的常数 k 实现位图的放大。

5.2.2 结果分析

选用大小为 682×1024 的灰度图像作为原始对象，进行 4×4 倍放大测试，结果如下。



(a) 原图全景图



(b) 原图局部图



(c) 双三次插值法全景图



(d) 双三次插值法局部图

图八：图像放大效果对比图

由上图对比可以看出，原始位图放大后的图像质量较差，尤其是局部图像内的边缘锯齿化和马赛克现象明显，图像较为模糊；通过双三次插值法对位图进行放大处理后的图像质量较好，图像的轮廓较为连贯、且锯齿化和马赛克的现象没有原图放大后的明显，图像颜色过渡平缓、图像较为清晰、噪点较少。因此，通过双三次插值法的位图放大方法可以有效地避免放大图像模糊或马赛克现象。

由图七、图八和表 1 对比可以看出，双三次插值法的位图放大图较双线性插值法的均方误差跟小、平均梯度更大，即图像更清晰、噪点更少、马赛克现象更少，因此，双三次插值法和双线性插值法对放大图像的质量都有相应的改善，其中，双三次插值法改善的效果更佳。

表 1：图像放大效果定量评价数据

指 标	算 法	双二次插值	双三次插值
均方误差		23.56	23.98
平均梯度		1.6503	1.9173

六、基于位图矢量化和双三次插值法的放大模型

6.1 综合位图矢量化和双三次插值法的放大模型

在第一阶段的问题中，我们通过提取简单位图的边缘轮廓实现了位图的矢量化。由于矢量图可以任意不失真地放大，所以在解决本阶段实现位图不失真放大的问题时，本文将待放大位图分为简单图像部分和复杂图像部分。对于简单部分，在第一阶段问题的基础上，本文通过最大熵法确定二值化的阈值，并在提取图像轮廓特征点的前提条件下，建立了基于 Bezier 曲线的拟合模型进行轮廓矢量化，进而实现放大；对于难以量化的复杂部分，本文采用双三次插值法直接对图像实现放大。

● 简单图像部分的 Bezier 曲线拟合法

一条 n 次 Bezier 曲线^[6]由 $n+1$ 个控制点决定，依次连接这 $n+1$ 个控制点即得到特征

多边形。

由 Bezier 曲线定义可得

$$\vec{r}(n) = \sum_{i=0}^n J_{n,i}(u) \vec{V}_i \quad (13)$$

式中， n 为 Bezier 曲线次数；

i 为特征多边形顶点的序号， $0 \leq i \leq n$ ；

u 为参数， $0 \leq u \leq 1$ ；

\vec{V}_i 为特征多边形顶点的位置矢量；

$J_{n,i}$ 为伯恩斯坦基函数。

$$J_{n,i}(u) = C_n^i u^i (1-u)^{n-i} \quad (14)$$

其中， C_n^i 为组合数，

$$C_n^i = \frac{n!}{i!(n-i)!} \quad (15)$$

为表述简便，可将以上公式表示成矩阵的形式，即

$$\vec{r}(n) = [t^n, t^{n-1}, \dots, t, 1] Mn [\vec{V}_0, \vec{V}_1, \dots, \vec{V}_n]^T \quad (16)$$

其中， Mn 是 n 阶方阵，元素 b_{ij} 为

$$b_{ij} = \begin{cases} (-1)^{n-i-j} C_n^{n-i} C_n^j & i+j \leq n+1 \\ 0 & \text{else} \end{cases} \quad (17)$$

已知控制点的位置矢量是使用 Bezier 曲线的前提，由于已知 $N+1$ 个型值点分别为 $P_j (j=0,1,2,\dots,N)$ ，可用逆算法求解出控制点，即

$$\sum_{i=0}^n J_{n,i}(u_j) \vec{V}_i = \vec{P}_j \quad (18)$$

写成矩阵的形式即为

$$\vec{r}(u) = (t^n, t^{n-1}, \dots, t, 1) \begin{bmatrix} (-1)^n C_n^n C_n^0 & (-1)^{n-1} C_n^n C_n^1 & \dots & (-1) C_n^n C_n^n \\ (-1)^{n-1} C_n^{n-1} C_n^0 & (-1)^{n-2} C_n^{n-1} C_n^1 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ (-1)^0 C_n^0 C_n^0 & 0 & \dots & 0 \end{bmatrix} \begin{bmatrix} \vec{V}_0 \\ \vec{V}_1 \\ \dots \\ \vec{V}_n \end{bmatrix} \quad (19)$$

该方程有 $n+1$ 个未知数， $N+1$ 个方程，若 $n=N$ ，则方程有唯一解；若 $n>N$ ，则方程有无穷多解，即控制点可随意选取；若 $n<N$ ，则为矛盾方程组，需要用最小二乘法求解。

参赛队号#1083

根据上述分析可知，特征点的个数一般小于 4 个，而且求得的 Bezier 曲线要与两侧的曲线光滑拼接，故令 $n = N + 2$ ，将多余的两个方程用于边界条件的求解，从而使得方程有唯一解。

边界条件：设与 $\bar{r}(u)$ 相邻的两条曲线如下，且 $u \in [0, 1]$

$$\bar{s}(u) = \sum_{i=0}^m J_{m,i}(u) \bar{P}_i \quad (20)$$

$$\bar{w}(u) = \sum_{i=0}^l J_{l,i}(u) \bar{Q}_i \quad (21)$$

①位置连续条件：

$$\begin{cases} \bar{r}(0) = \bar{P}_1 \\ \bar{r}(1) = \bar{Q}_0 \end{cases} \quad (22)$$

②斜率连续条件

$$\begin{cases} \bar{s}'(1) = \alpha \bar{r}'(0) \\ \bar{r}'(1) = \beta \bar{w}'(0) \end{cases} \quad (23)$$

其中， α, β 为参数。当 $\alpha \neq \beta \neq 1$ 时，即切矢共线，但模长不同，可满足数学上可到的条件；当 $\alpha = \beta = 1$ 时，拼接时切矢量相等，即曲率相等，此时拟合效果最好。

● 复杂图像部分的双三次插值法

由于第二阶段要处理的位图是将内部图像较为复杂的彩色图像，因此在实际操作中，将整幅图像矢量化并得到相应轮廓方程、再进行放大的方法是不可行的。对于简单的图像轮廓，可以采用上述 Bezier 曲线拟合的位图矢量化模型进行处理，但对于位图中的其他复杂区域，则可采用上文提到的双三次插值方法进行放大。基于双三次插值方法的位图放大模型在第五部分已经做过介绍，因此这里不再赘述。

6.2 模型求解与结果分析

6.2.1 模型的求解

根据上文，通过 MATLAB 对模型进行求解，具体步骤如下。

- Step 1: 将位图主要轮廓提取，并通过曲率确定分段点；
- Step 2: 对每两个分段点之间用 Bezier 曲线拟合得到曲线方程；
- Step 3: 得到主要轮廓的矢量方程，将之赋予相应位图轮廓之上，得到半矢量半像素图片；
- Step 4: 图片的矢量边界放大指定倍数，同时将像素点由矢量边界为起点进行双三次线性插值放大相同倍数，得到放大后的半矢量半像素图片；
- Step 5: 对比半矢量半像素放大图片和由双三次线性插值放大图片的均方误差和平均梯度；

6.2.2 结果分析

参赛队号#1083

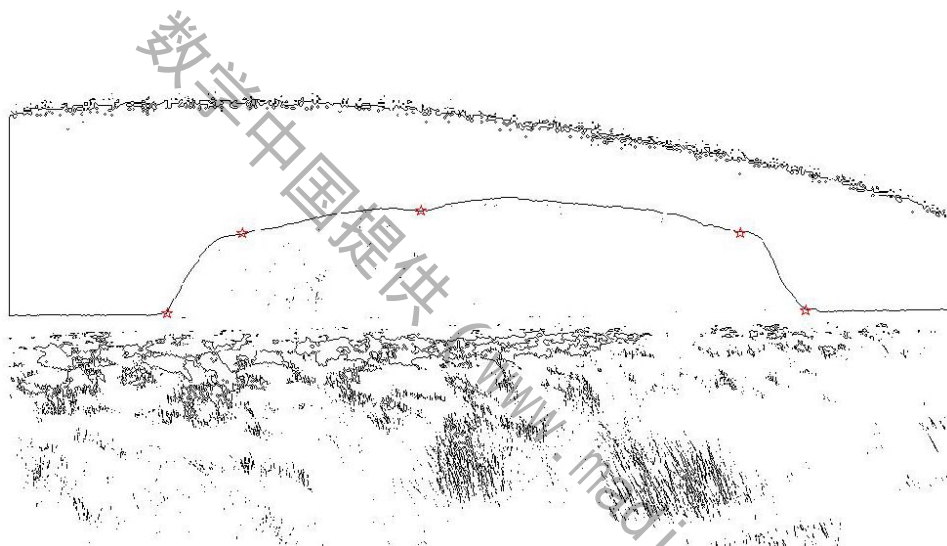
选用大小为 682×1024 的灰度图像作为原始对象，进行 4×4 倍放大测试，结果如下。

(a) 原始位图

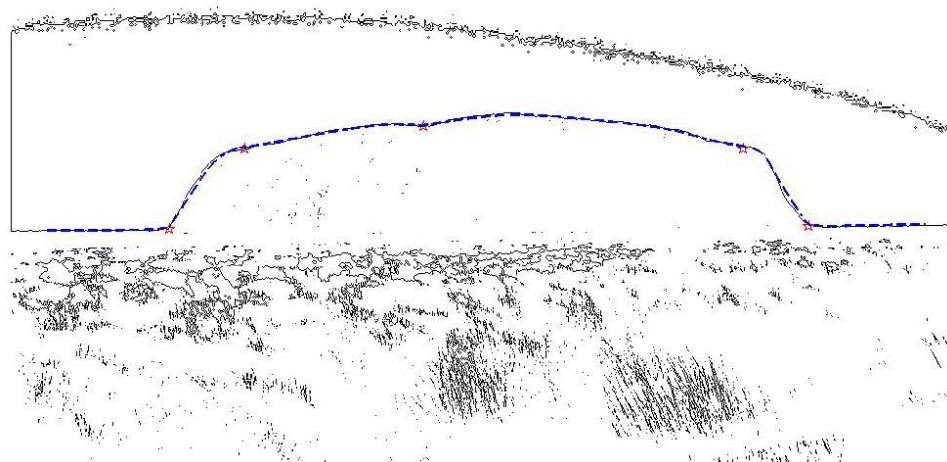


(b) 本模型放大效果图

☆ 曲线拟合分段点



(c) 简区域分段点的分布图

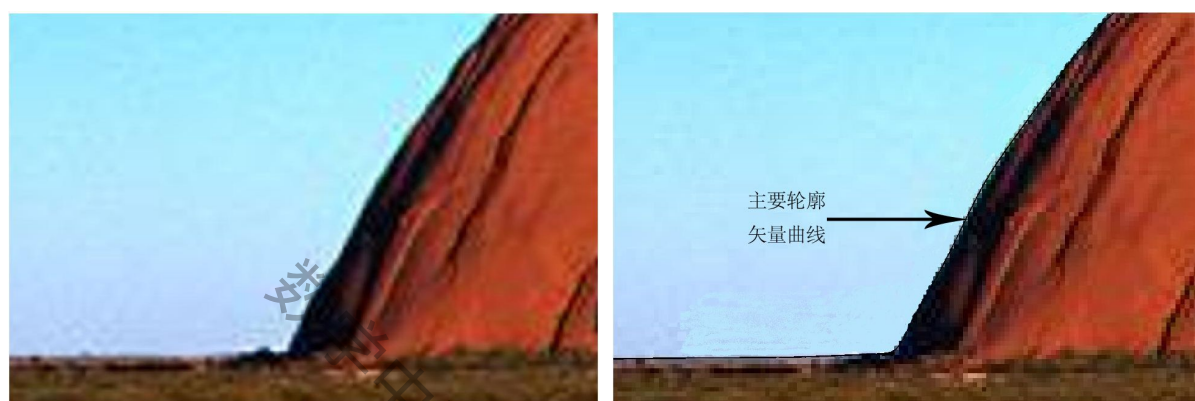
☆ 曲线拟合分段点
--- Bezier拟合曲线示意

(d) Bezier 曲线拟合示意图

图九：图像放大示意图

由上图可以看出，通过对位图简单部分分段点的提取和轮廓的矢量化、对复杂部分双三次插值的计算和处理，得到的最终放大效果图质量较好。其轮廓分明、颜色过渡平缓、锯齿化或马赛克的现象较少、图象清晰，因此，结合位图矢量化和双三次插值法的综合位图放大模型可以有效避免放大图像失真和图像模糊、乃至马赛克现象的产生。

为了更直观地看出本模型与仅基于双三次插值法位图放大模型的区别，本文分别给出了二者局部放大图像的效果图。



(a) 仅使用双三次插值法的放大效果局部图

(b) 使用本模型的放大效果局部图

图十：两种模型放大效果局部对比图

由上图对比可以看出，结合位图矢量化与双三次插值法的位图放大法比单使用双三次插值法的位图放大法所得到的图像效果更加清晰，即图像轮廓更明朗、马赛克现象更少，因此，本模型使用的综合法较基于双三次插值法的模型更适用于位图的放大处理，所得图像的质量更佳。

七、模型结果的分析与比较

7.1 放大效果图的评价指标

为了定性的给出以上基于三种插值方法的模型的结果、测试模型的有效性，本文将以上埃特金插值方法、改进的双线性插值法和双三次插值法进行放大效果的对比。选用大小为 682×1024 的灰度图像作为对象，进行 4×4 倍放大测试。为定量的评价位图的放大效果，本文采用均方误差和平均梯度作为定量评价指标。

● 均方误差

均方误差是衡量“平均误差”的一种较方便的方法^[7]，可以评价数据的变化程度。在位图放大过程中，可以用以下公式来评判其优劣程度：

$$\frac{1}{M \times N} \sum_{i=1}^M \sum_{j=1}^N (F(i, j) - G(i, j))^2 \quad (24)$$

式中， F 和 G 分别为原图和放大图像；

M 和 N 分别是图像的长和宽。

均方误差反映放大图像与原图的逼近程度，均方误差越小表示放大图像越逼近原图，放大效果越好。

参赛队号#1083

本文采用三种插值算法进行位图放大，故比较多个模型的精度，可以将均方误差(MSE)进行分解，得到均方误差分解公式。模型公式如下：

$$\begin{aligned} MSE &= E[(y_{t+k} - \hat{y}_{t+k,t})^2] \\ &= \text{var}(y_{t+k} - \hat{y}_{t+k,t}) + (E[y_{t+k}] - E[\hat{y}_{t+k,t}])^2 \end{aligned} \quad (25)$$

将上式进行扩展，上式等价于

$$MSE = \text{var}(y_{t+k}) + \text{var}(\hat{y}_{t+k,t}) - 2\text{cov}(y_{t+k}, \hat{y}_{t+k,t}) + (E[y_{t+k}] - E[\hat{y}_{t+k,t}])^2 \quad (26)$$

这一结果清楚地表明均方误差仅取决于原图序列和插值后序列联合分布二阶矩的结构。

● 平均梯度

平均梯度指图像的边界或影线两侧附近灰度有明显差异，即灰度变化率大。这种变化率的大小可用来表示图像清晰度。它反映了图像微小细节反差变化的速率，即图像多维方向上密度变化的速率，表征图像的相对清晰程度。平均梯度即图像的清晰度，反映图像对细节对比的表达能力，计算公式为

图像梯度：
$$G(x, y) = dx_i + dy_i \quad (27)$$

$$dx(i, j) = I(i+1, j) - I(i, j) \quad (28)$$

$$dy(i, j) = I(i, j+1) - I(i, j) \quad (29)$$

其中， I 是图像像素的值（如：RGB 值）；

(i, j) 为像素的坐标。

图像梯度一般也可以用中值差分，即

$$dx(i, j) = [I(i+1, j) - I(i, j)]/2 \quad (30)$$

$$dy(i, j) = [I(i, j+1) - I(i, j)]/2 \quad (31)$$

这里本文引用了平均梯度来表示图片的放大效果。平均梯度越大，说明图像的层次越多，也就越清晰。其定义为

$$\bar{g} = \frac{1}{(M-1)(N-1)} \times \sum_{i=1}^{M-1} \sum_{j=1}^{N-1} \sqrt{\frac{\Delta I_x^2 + \Delta I_y^2}{2}} \quad (32)$$

式中，

$$\Delta I_x = F(i, j) - F(i+1, j) \quad (33)$$

$$\Delta I_y = F(i, j) - F(i, j+1) \quad (34)$$

7.2 三种模型效果评价分析

参赛队号#1083

由上文可知，基于双线性插值法的位图放大模型、基于双三次插值法的位图放大模型、结合位图轮廓矢量化和双三次插值法的位图放大模型均可在一定程度上保证放大图像的质量、减少图像模糊或马赛克现象的产生，其各自放大效果图见下图十一、图十二。



(a) 原始位图



(b) 双线性插值法放大效果图

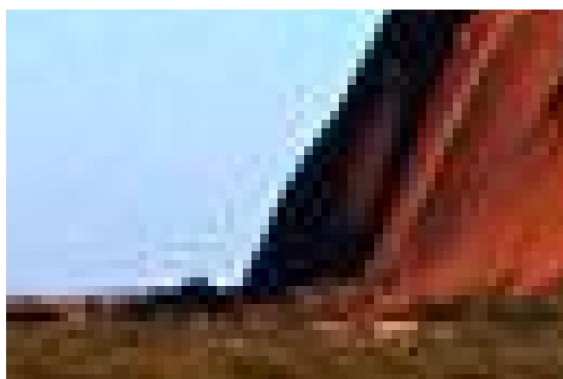


(c) 双三次插值法放大效果图



(d) 综合矢量化与双三次插值法放大效果图

图十一：图像放大效果整体对比图



(a) 原始位图局部图



(b) 双线性插值法放大效果局部图

参赛队号#1083



(c) 双三次插值法放大效果局部图



(d) 综合矢量化与双三次插值法放大效果局部图

图十二：图像放大效果局部对比图

通过对比各个模型算法下放大图像的效果图可以看出，相比于其他两种算法，双线性插值法的放大效果最差，图像出现马赛克和锯齿现象，图中岩石的边缘出现失真的现象；双三次插值法放大效果优于双线性插值法，图像柔和，但也存在边缘不太清晰的现象；而结合图像矢量化和双三次插值的综合方法对图片放大的效果最佳，图像轮廓清晰、颜色过度柔和，其边缘区清晰度最优，也几乎没有出现锯齿或者马赛克的现象。

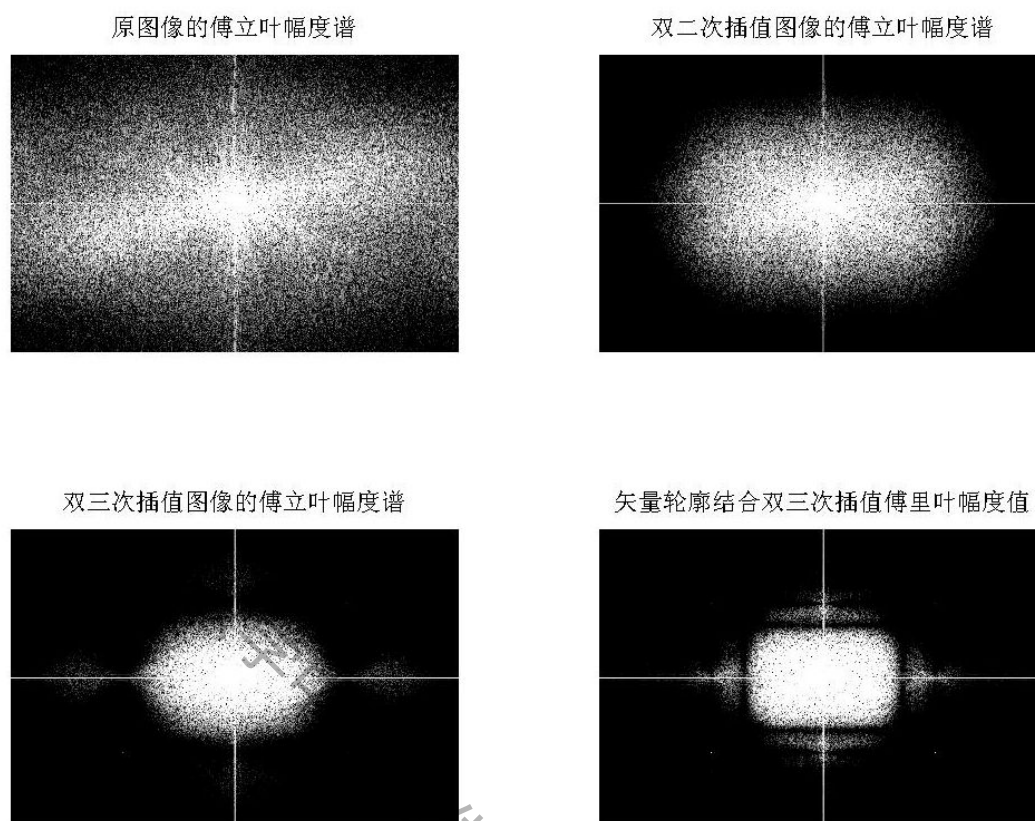
为评价验证各个模型放大位图的最终效果，本文计算每个模型放大图像的均方误差和平均梯度，通过对比数据得出不失真放大位图的最佳模型。其中，各个模型放大图像的相关指标如下表。

表 2：图像放大效果定量评价数据

指 标	双二次插值	双三次插值	轮廓矢量化结合双三次插值
均方误差	23.56	23.98	22.36
平均梯度	1.6503	1.9173	2.1021
(放大 4 倍)程序 耗时/s	1.2314	2.6234	3.0123

从定量评价数据上看，双二次插值法的均方误差为 23.56，是三种算法中最大的，说明放大图像与原图的差距较大，其平均梯度最小为 1.6503，说明图像中微小细节的反差与纹理变化的特征明显，即使程序有计算量较小、程序耗时少的优点，但放大的效果最差；双三次插值法的均方误差为 23.98，小于双二次插值法的均方误差，同时平均梯度也相应变大，因此基于双三次插值法的放大效果优于双二次插值法，放大的图像清晰度较高、质量好，但相应的程序耗时增多；结合位图轮廓矢量化与双三次插值的综合算法计算量最大，耗时也最多，为 3.0123 秒，但其均方误差最低，为 22.36，其平均梯度也高于其他两种算法，说明此方法放大的图片清晰度最高、质量最好，即图片放大的效果最佳。实验结果证明，此方法对于改善放大图像的质量最有效。

为进一步验证模型的结果，本文做出各个模型放大后的傅立叶幅度图，见下图。图像的频率作为图像中灰度变化剧烈程度的指标，也是灰度在平面空间上的梯度。因此，傅立叶幅度用于表征图像的光滑程度。其中，点的离散程度代表图像的光滑程度，点分布得越离散、代表图像的马赛克区域越多，即图像的光滑程度越低。



图十三：图像放大效果傅立叶幅度图

从傅立叶幅度值对比图可以看出，原图像的傅立叶幅度谱点的离散程度最大，即图像的马赛克区域最多，图像的光滑程度最低；双线性插值法放大的图像离散程度高于原始图像，即此方法可以减少放大后的马赛克现象、提高图片放大质量；双三次插值图像的傅立叶幅度图中，各点分布较前者要集中，即图像的光滑程度高于前者；而结合位图轮廓矢量化与双三次插值的综合方法放大的图像中，各点分布得最集中、离散程度最低，说明图像的马赛克区域最少，即图像的光滑程度最高、图像质量最好。这与上文中从指标数据方面分析放大效果的结论相一致。

综上，双线性插值法、双三次插值法和结合位图矢量化和双三次插值的综合法都可以在一定程度上改善放大位图的质量，其中，结合位图矢量化和双三次插值的综合法效果最佳、双三次插值法次之、双线性插值法改善效果最弱。

八、模型评价

6.1 模型优点

①本文建立了三个模型针对位图放大失真的问题进行求解，通过数据和效果图比较分析各个模型的结果，分别总结出每个模型的优缺点，说服力强，且筛选出的最优模型既可保证放大位图的质量，又具有较高的可行性，对实际的位图放大研究具有借鉴意义；

②模型二将双线性插值法与最近邻点法相结合，既考虑到插值点与邻点之间距离关系，又考虑到邻点像素的灰度分布特性，创新性强。实验结果从直观视觉和定量评价上验证了本文算法的有效性；

参赛队号#1083

③双线性插值算法和双三次插值算法二者间为一个循序渐进的过程，建立第三个综合模型时，本文有效地利用了模型一、二得出的结论。总体而言，模型间具有较高的连贯性和相关性，论文的思路层层递进、由浅入深；

④本文给出了多个定量评价位图放大效果的指标（均方误差、平均梯度、傅立叶幅度），通过多项指标分析对比各个模型算法的实际放大效果，对模型具有总结意义、科学性强。

6.2 模型缺点

①模型三的位图放大效果最好，能够精确的不失真放大，但其运算量大、程序耗时最多，因此，此种方法仅在对目标放大位图质量要求高的情况下使用，普适性较弱；

②本文将低频图像，亦或噪点比较小的图像作为研究对象，提出的三种模型算法对高频图像的放大效果还有待于研究和完善。

七、模型优化与推广

通过对比三种算法模型的结果，可以发现，综合位图矢量化及双三次插值法的模型的效果对位图的放大是最佳的。因此，本文重点讨论基于位图矢量化的双三次插值算法模型的优化与推广。

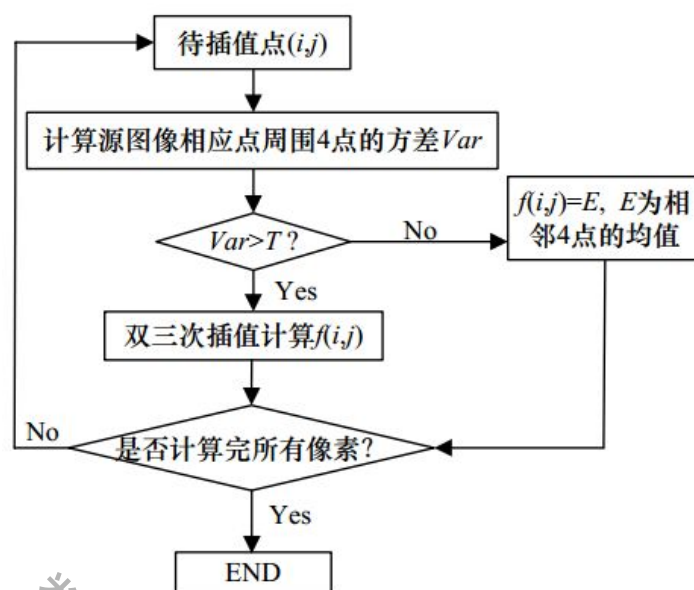
在简单图像矢量化放大的同时，双三次插值算法对其余复杂图像的像素均进行同样的大运算量插值。基于减小运算量、减少运算时间这一考虑，对双三次插值算法进行如下优化。

针对双三次插值存在的问题，首先考虑到的就是把复杂图像划分成不同的区域，对不同的区域采用不同的插值算法，以达到在保持放大后图像质量的同时减少运算量的目的。一般的划分区域的算法需要对图像进行分割，如文献^[15-16]。考虑到图像分割比较复杂，而且没有适用于所有图像的统一算法。本文避免首先进行图像分割，而是先计算待插值点在源图像中的对应点的四邻域点像素值的方差 Var ，如果方差小于设定的阈值 T ，则直接取这4个像素值的均值 E 作为待插值点的像素值；反之，仍按双三次插值计算。方差 Var 的计算公式如下：

$$Var = (E - f_{11})^2 + (E - f_{12})^2 + (E - f_{21})^2 + (E - f_{22})^2 \quad (35)$$

其中， f_{11} 、 f_{12} 、 f_{21} 、 f_{22} 分别为当前待插值点在源图像的对应点邻域4点的像素值， E 为此4点像素值的均值。由式(35)可知，这种改进的方法比原双三次插值的计算量要小得多。

方差 Var 反映了该像素点所处四邻域的平坦程度。对于不同的图像，很难确定统一的阈值标准，而且在不同场合也往往需要调整阈值，以满足实际应用要求。根据实验经验值，当选取阈值 $T = 20$ 时，运算量的降低和图像质量的保持相对合理，在具体应用时可以根据需要在20上下调整阈值，以得到最佳的折中效果。本文图像均选用 $T = 20$ 进行实验。算法流程如下图所示。



图十四：算法流程

本文在简单图像矢量化放大的基础上，提出的基于区域的双三次插值算法，避免进行图像分割，在保持图像质量的同时，比双三次插值算法的运算量在一定程度上有所降低。在对放大后图像质量和实时性均要求较高的情况下，本文算法具有一定的实用价值。另外，如何自适应地选择阈值 T ，更好地在运算速度和放大图像质量之间进行折中，是下一步要研究的内容。

八、参考文献

- [1]霍宏涛. 数字图像处理. 北京: 北京理工大学出版社, 2002: 1~50
- [2]朱学芳, 智文广. 计算机图像处理导论. 北京: 科学技术文献出版社, 2003: 1~100
- [3]孙即祥. 图像处理. 北京: 科学出版社, 2004: 1~150
- [4]宋建中. 图像处理智能化的发展趋势[J]. 中国光学, 2011, 4(5): 431-440.
- [5]景敏. 数字图像处理技术的应用与发展[J]. 科技信息, 2010, 27: 448.
- [6]计算机辅助几何造型技术[M]. 科学出版社, 2009.
- [7]http://baike.baidu.com/link?url=8AGO5Q3guBoXnh0_9LarOjWKM9w5gS2JltZkzDaGj2Fg5bgisnnL2HU8EjEnWnyi1bcYlKnePS_f4K-ji7S6Xq
- [8]贾玉珍, 靳冰, 刘琮, 等. BMP 文件结构的信息隐藏方法与实现[J]. 江西理工大学学报, 2009, 30(1): 42-44.
- [9]程玉宝, 徐海萍. 基于小波重构和灰度分段的红外图像放大增强[J]. 红外技术, 2008, 30(10): 567-570.
- [10]朱宁, 吴静, 王忠谦. 图像放大的偏微分方程方法[J]. 计算机辅助设计与图形学学报, 2005, 17(9): 1941-1945.
- [11]冯象初, 姜东焕, 徐光宝. 基于变分和小波变换的图像放大算法[J]. 计算机学报, 2008, 31(2): 340-345

参赛队号#1083

- [12]龚昌来, 杨冬涛. 一种改进的双线性插值图像放大算法[J]. 激光与红外, 2009, 39(8): 899-901.
- [13]陈小蕾, 张俊, 吴乐南. 多帧超分辨率重建中的图像插值[J]. 信号处理, 2004, 20(3): 268-271.
- [14]王会鹏, 周利莉, 张杰. 一种基于区域的双三次图像插值算法[J]. 计算机工程, 2010, 36(19).
- [15]符祥, 郭宝龙. 区域指导的自适应图像插值算法[J]. 光电子激光, 2008, 19(2): 233-236.
- [16] 李将云. 图像处理中的插值和缩放若干技术研究[D]. 杭州: 浙江大学, 2002.

九、附录

1.

%主函数

in=['ttq2.jpg'];

k=input('放大倍数 k=');

A=imread(in); %原图

figure;imshow(A);

title('原图')

%B=imresize(A,k,'nearest'); %最大内间

%figure;imshow(B);

%title('埃特金插值')

%ig=rgb2gray(B); %调用 avegrad 求平均梯度

%avegrad(ig)

BB=imresize(A,k,'bilinear'); %双二次插值

figure;imshow(BB);

title('双二次插值')

ig=rgb2gray(BB); %调用 avegrad 求平均梯度

```
avegrad(ig)
```

```
BBB=imresize(A,k,'bicubic'); %双三次插值
figure;imshow(BBB);
title('双三次插值')
ig=rgb2gray(BBB); %调用 avegrad 求平均梯度
avegrad(ig)
```

```
%jiehesuanfa %(另一个函数接口) 矢量话边界结合双三次插值算法
```

```
figure;subplot(221);imshow(A);
title('原图')
subplot(222);imshow(BB);
title('双二次插值')
subplot(223);imshow(BBB);
title('双三次插值')
subplot(224);imshow(B);
title('矢量边界及双三次内插算法')
```

2.

```
%求解平均梯度，用于评价放大算法的好坏；
```

```
function AVEGRAD=avegrad(img)
```

```
%%%%%% this function is used to calculate theaverage gradient of an image.
```

```
%%%%%%平均梯度可敏感地反映图像对微小细节反差表达的能力，可用来评价图像的模糊程度
```

```
%%%%%%在图像中，某一方向的灰度级变化率大，它的梯度也就大。因此，可以用平均梯度值来衡量图像的清晰度，还同时反映出图像中微小细节反差和纹理变换特征。
```

```
%img=imread(img);
%img=rgb2gray(img);
img=double(img);
```

参赛队号#1083

```

[M,N]=size(img);
gradval=zeros(M,N); %%%% save the gradient value of single pixel
diffX=zeros(M,N); %%%% save the differential value of X orient
diffY=zeros(M,N); %%%% save the differential value of Y orient
tempX=zeros(M,N);
tempY=zeros(M,N);
tempX(1:M,1:(N-1))=img(1:M,2:N);
tempY(1:(M-1),1:N)=img(2:M,1:N);
diffX=tempX-img;
diffY=tempY-img;
diffX(1:M,N)=0; %%%% the boundary set to 0
diffY(M,1:N)=0;
diffX=diffX.*diffX;
diffY=diffY.*diffY;
AVEGRAD=(diffX+diffY)/2;
AVEGRAD=sum(sum(sqrt(AVEGRAD)));
AVEGRAD=AVEGRAD/((M-1)*(N-1));

```

3.

%提起图像主要轮廓方程的程序，

```

I=imread('ttq2.jpg');
J=im2bw(I,0.5);
I2=bwperim(J);
figure;imshow(1-I2)
hold on
pp =[ 172.1662  254.6259  448.0315  794.3624  864.8280
      431.4561  344.4985  320.5102  344.4985  428.4575];
plot(pp(1,:),pp(2,:),'rp');    %分割点

p=ginput;    %选取控制点

```

参赛队号#1083

```

p=p';
[a,b]=size(p);
t=0:0.005:1;

for i=1:b-3    %曲线拟合
    x=p(1,i)*(1/6)*(-t.^3+3*t.^2-3*t+1)+p(1,i+1)*(1/6)*(3*t.^3-6*t.^2+4)...
    +p(1,i+2)*(1/6)*(-3*t.^3+3*t.^2+3*t+1)+p(1,i+3)*(1/6)*t.^3;
    y=p(2,i)*(1/6)*(-t.^3+3*t.^2-3*t+1)+p(2,i+1)*(1/6)*(3*t.^3-6*t.^2+4)...
    +p(2,i+2)*(1/6)*(-3*t.^3+3*t.^2+3*t+1)+p(2,i+3)*(1/6)*t.^3;
    plot(x,y,'--','linewidth',1);

end

legend('曲线拟合分段点')
legend('曲线拟合分段点','Bezier 拟合曲线示意')
%plot(p(1,:),p(2,:), 'linewidth',1);

%hold off;

```

4.

```

function []=chahzi(str,k)
%??str??input??image??
%??k???zoo??in??or??out??k??times???clear??all???clc???k=2;??
I=imread(str);
%??if??size(I,3)~=1??
%??I=rgb2gray(I);??
%??end??
figure,imshow(I);
title('原图');I=double(I);
[rows cols tongdao]=size(I);x_new=rows*k;
y_new=cols*k;%缩放至 k 倍。???
I_new=zeros(x_new,y_new,tongdao);

```

参赛队号#1083

```

for rgb=1:tongdao
    for i=1:x_new
        for j=1:y_new
            x=i/k;a=floor(x);
            y=j/k;b=floor(y);%%??    双    线    性    插    值    算
            法????????????????????????????????????????
            if a>0&&b>0&a<rows&&b<cols
                cxb=I(a+1,b,rgb)*(x-a)+I(a,b,rgb)*(1+a-x);
                cxb1=I(a+1,b+1,rgb)*(x-a)+I(a,b+1,rgb)*(1+a-x);
                I_new(i,j,rgb)=round(cxb1*(y-b)+cxb*(1+b-y));
            end
        end
    end
end
figure,imshow(uint8(I_new))
%size(uint8(I_new))
%title('result??image');

```

```

%imresize method
%I = imread('yuan.jpg');
%J = imresize(I, 5);
%figure, imshow(I), figure, imshow(J);

```

5.

clear;

clc;

ff=imread('ttq2.jpg'); %读取图像到 ff

[mm,nn]=size(ff); %将图像隔行隔列抽取元素，得到缩小一半的图像 f

m=mm/2;n=nn/2;

for i=1:m

参赛队号#1083

```

for j=1:n
    f(i,j)=ff(2*i,2*j);
end
end
k=3; %设置放大倍数
%将待插值图像矩阵前后各扩展两行两列,共扩展四行四列
a=f(1,:); %取矩阵 f 的第一行的全部元素
c=f(m,:); %取矩阵 f 的第 m 行的全部元素
b=[f(1,1),f(1,1),f(:,1)',f(m,1),f(m,1)];d=[f(1,n),f(1,n),f(:,n)',f(m,n),f(m,n)];
a1=[a;a;c;c]; %a1 为 f 上扩展两行,下扩展两列
a1'; %a1 转置
b1=[b;b;a1';d]; %b1 为 a1 转置后上下各扩展两行
f=b1';f1=double(f); %b1 再转置,成为 f 的周围扩展两圈
for i=1:k*m; %利用双三次插值公式对新图象所有像素赋值
    u=rem(i,k)/k; i1=floor(i/k)+2; %i 能否被 k 整除,整除时 u=0, floor 指朝负无穷方向舍入
    A=[sw(1+u) sw(u) sw(1-u) sw(2-u)]; %矩阵 A
    for j=1:k*n;
        v=rem(j,k)/k;j1=floor(j/k)+2;
        C=[sw(1+v);sw(v);sw(1-v);sw(2-v)]; %矩阵 C
        B=[f1(i1-1,j1-1) f1(i1-1,j1) f1(i1-1,j1+1) f1(i1-1,j1+2)
            f1(i1,j1-1) f1(i1,j1) f1(i1,j1+1) f1(i1,j1+2)
            f1(i1+1,j1-1) f1(i1+1,j1) f1(i1+1,j1+1) f1(i1+1,j1+2)
            f1(i1+2,j1-1) f1(i1+2,j1) f1(i1+2,j1+1) f1(i1+2,j1+2)]; %矩阵 B
        g1(i,j)=(A*B*C);
    end
end
end
g=uint8(g1);
imshow(f); title('缩小一半的图像'); %显示缩小的图像
figure,imshow(ff);title('原图像'); %显示原图像
figure,imshow(g);title('双三次插值放大后的图像'); %显示插值后的图像

```

6.

%双三次插值具体实现

clc,clear;

fff=imread('ttq2.jpg');

%YIQ = rgb2ntsc(fff);%保存彩图数据

ff = rgb2gray(fff);%转化为灰度图像

[mm,nn,td]=size(fff);

%将图像隔行隔列抽取元素，得到缩小的图像 f

m=mm/2;n=nn/2;

f = zeros(m,n);

for i=1:m

for j=1:n

f(i,j)=ff(2*i,2*j);

end

end

k=4; %设置放大倍数

bijiao1 = imresize(f,k,'bilinear');%双线性插值结果比较

bijiao = uint8(bijiao1);

bioo=imresize(f,2,'bicubic');

bioo=uint8(bioo);

a=f(1,:);

c=f(m,:); %将待插值图像矩阵前后各扩展两行两列,共扩展四行四列

b=[f(1,1),f(1,1),f(:,1)',f(m,1),f(m,1)];

d=[f(1,n),f(1,n),f(:,n)',f(m,n),f(m,n)];

参赛队号#1083

```

a1=[a;a;f;c;c];
b1=[b;b;a1';d;d];
ffff=b1';
f1=double(ffff);
g1 = zeros(k*m,k*n);
for i=1:k*m                                %利用双三次插值公式对新图象所有像素赋值
    u=rem(i,k)/k; i1=floor(i/k)+2;
    A=[sw(1+u) sw(u) sw(1-u) sw(2-u)];
    for j=1:k*n
        v=rem(j,k)/k;
        j1=floor(j/k)+2;
        C=[sw(1+v);sw(v);sw(1-v);sw(2-v)];
        B=[f1(i1-1,j1-1) f1(i1-1,j1) f1(i1-1,j1+1) f1(i1-1,j1+2)
            f1(i1,j1-1)   f1(i1,j1)   f1(i1,j1+1)   f1(i1,j1+2)
            f1(i1+1,j1-1) f1(i1+1,j1) f1(i1+1,j1+1) f1(i1+1,j1+2)
            f1(i1+2,j1-1) f1(i1+2,j1) f1(i1+2,j1+1) f1(i1+2,j1+2)];
        g1(i,j)=(A*B*C);
    end
end
g=uint8(g1);
imshow(uint8(f));title('缩小的图像');          %显示缩小的图像

figure,imshow(ff);title('原图');                %显示原图像

figure,imshow(g);title('双三次插值放大的图像'); %显示插值后的图像
figure,imshow(bijiao);title('双线性插值放大结果'); %显示插值后的图像
mse=0;

ff=double(ff);

g=double(g);

```

参赛队号#1083

```

ff2=fftshift(fft2(ff));    %计算原图像和插值图像的傅立叶幅度谱
g1=fftshift(fft2(bijiao));
g2=fftshift(fft2(g));
g3=fftshift(fft2(bioo));
figure,subplot(2,2,1),imshow(log(abs(ff2)),[8,10]);
title('原图像的傅立叶幅度谱');
subplot(2,2,3),imshow(log(abs(g1)),[8,10]);
title('双三次插值图像的傅立叶幅度谱');
subplot(2,2,4),imshow(log(abs(g2)),[8,10]);
title('矢量轮廓结合双三次插值傅里叶幅度值');
subplot(2,2,2),imshow(log(abs(g3)),[8,10]);
title('双二次插值图像的傅立叶幅度谱');

```

7.

%sanci 函数的调用子函数;

```

function A=sw(w1)
w=abs(w1);
if w<1&&w>=0
    A=1-2*w^2+w^3;
elseif w>=1&&w<2
    A=4-8*w+5*w^2-w^3;
else
    A=0;
end

```