1. 使用matlab写一个函数 img=generateFigure（imgW,imgH）,其作用为产生一副彩色图像，图像中用红色显示[0，2\*pi]的正弦波，用绿色显示[0，2\*pi]的余弦波，蓝色显示[0，2\*pi]的y=x^2图像。

程序主要代码：

function [img] = generateFigure(imgH ,imgW )

img = zeros(imgH,imgW,3);

img = uint8(img);

img(:,:,1)=255;

img(:,:,2)=255;

img(:,:,3)=255;

x=0:2\*pi/(imgW-1):2\*pi;

y1=sin(x);

y2=cos(x);

y3=x.^2;

x=int32(x/2/pi\*imgW);

y1=int32(imgH/40\*39-round(y1\*imgH/40));

y2=int32(imgH/40\*39-round(y2\*imgH/40));

y3=int32(imgH/40\*39-round(y3\*imgH/40));

for i=1:imgW

if x(i)==0

x(i)=x(i)+1;

end

if y1(i)>0 && y2(i)<=imgH

img(y1(i),x(i),2)=0;

img(y1(i),x(i),3)=0;

end

if y2(i)>0 && y2(i)<=imgH

img(y2(i),x(i),3)=0;

img(y2(i),x(i),1)=0;

end

if y3(i)>0 && y3(i)<=imgH

img(y3(i),x(i),1)=0;

img(y3(i),x(i),2)=0;

end

end

img(:,1,:)=0;

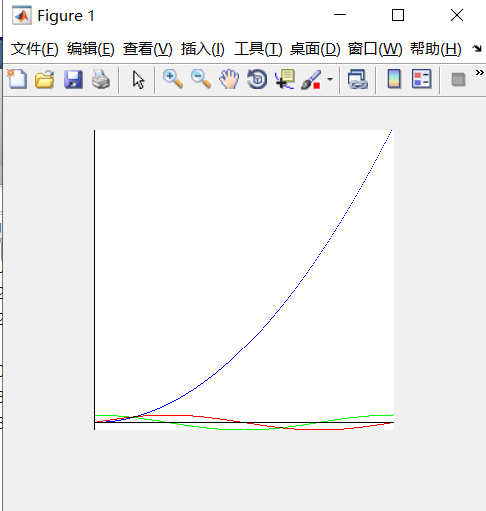
img(round(imgH/40\*39),:,:)=0;

imshow(img);

End

命令行中输入generateFigure（300，300）得

结果：



1. 不使用for循环，实现bilinear interpolation,测试图片为test.png，把原图片放大二倍。

主要代码：

function [ ZI ] = imblizoom( I,zmf )

if ~exist('I','var') || isempty(I)

    error('输入图像 I未定义或为空！');

end

if ~exist('zmf','var') || isempty(zmf) || numel(zmf) ~= 1

     error('位移矢量 zmf未定义或为空或 zmf中的元素超过2！');

end

if isstr(I)

    [I,M] = imread(I);

end

if zmf <= 0

     error('缩放倍数 zmf的值应该大于0！');

end

%% Step2 通过原始图像和缩放因子得到新图像的大小，并创建新图像。

[IH,IW,ID] = size(I);

ZIH = round(IH\*zmf); % 计算缩放后的图像高度，最近取整

ZIW = round(IW\*zmf); % 计算缩放后的图像宽度，最近取整

ZI = zeros(ZIH,ZIW,ID); % 创建新图像

%% Step3 扩展矩阵I边缘

IT = zeros(IH+2,IW+2,ID);

IT(2:IH+1,2:IW+1,:) = I;

IT(1,2:IW+1,:)=I(1,:,:);IT(IH+2,2:IW+1,:)=I(IH,:,:);

IT(2:IH+1,1,:)=I(:,1,:);IT(2:IH+1,IW+2,:)=I(:,IW,:);

IT(1,1,:) = I(1,1,:);IT(1,IW+2,:) = I(1,IW,:);

IT(IH+2,1,:) = I(IH,1,:);IT(IH+2,IW+2,:) = I(IH,IW,:);

%% Step4 由新图像的某个像素（zi，zj）映射到原始图像(ii，jj)处，并插值。

for zj = 1:ZIW         % 对图像进行按列逐元素扫描

    for zi = 1:ZIH

        ii = (zi-1)/zmf; jj = (zj-1)/zmf;

        i = floor(ii); j = floor(jj); % 向下取整

        u = ii - i; v = jj - j;

        i = i + 1; j = j + 1;

        ZI(zi,zj,:) = (1-u)\*(1-v)\*IT(i,j,:) +(1-u)\*v\*IT(i,j+1,:)...

                    + u\*(1-v)\*IT(i+1,j,:) +u\*v\*IT(i+1,j+1,:);

    end

end

ZI = uint8(ZI);

%% 以图像的形式显示同现矩阵P

figure

imshow(I,M);

axis on

title(['原图像（大小： ',num2str(IH),'\*',num2str(IW),'\*',num2str(ID),')']);

figure

imshow(ZI,M);

axis on

title(['缩放后的图像（大小： ',num2str(ZIH),'\*',num2str(ZIW),'\*',num2str(ID)',')']);

end