基于**Harris**角点的图像拼接技术（**MATLAB**源代码）

clc

clear all

% 读入图片

pic1=imread('lena1.jpg');

pic2=imread('lena2.jpg');

% Harris 角点检测

points1=myHarris(pic1);

points2=myHarris(pic2);

% 画出 Harris 角点

figure(1)

drawHarrisCorner(pic1,points1,pic2,points2);

% 角点特征描述

des1=myHarrisCornerDescription(pic1,points1);

des2=myHarrisCornerDescription(pic2,points2);

% 角点粗匹配

matchs=myMatch(des1,des2);

% 获取各自出匹配角点位置

matchedPoints1=points1(matchs(:,1),:);

matchedPoints2=points2(matchs(:,2),:);

% 粗匹配角点连线

figure(2)

drawLinedCorner(pic1,matchedPoints1,pic2,matchedPoints2);

% 角点精匹配

[newLoc1,newLoc2]=pointsSelect(matchedPoints1,matchedPoints2);

% 精匹配角点连线

figure(3)

drawLinedCorner(pic1,newLoc1,pic2,newLoc2);

% 图像拼接

im=picMatched(pic1,newLoc1,pic2,newLoc2);

% 显示拼接图像

figure(4)

imshow(im);

set(gcf,'Color','w');

function points=myHarris(pic)

% 功能：寻找 Harris 角点

% 输入：RGB 图像或 gray 图像

% 输出：角点所在的行、纵的 N×2 矩阵

if length(size(pic))==3

pic=rgb2gray(pic);

end

pic=double(pic);

hx=[-1 0 1];

Ix=filter2(hx,pic);

hy=[-1;0;1];

Iy=filter2(hy,pic);

Ix2=Ix.\*Ix;

Iy2=Iy.\*Iy;

Ixy=Ix.\*Iy;

h=fspecial('gaussian',[7 7],2);

Ix2=filter2(h,Ix2);

Iy2=filter2(h,Iy2);

Ixy=filter2(h,Ixy);

[heigth,width]=size(pic);

alpha=0.06;

R=zeros(heigth,width);

for i=1:heigth

for j=1:width

M=[Ix2(i,j) Ixy(i,j);Ixy(i,j) Iy2(i,j)];

R(i,j)=det(M)-alpha\*(trace(M)^2);

end

end

Rmax=max(max(R));

pMap=zeros(heigth,width);

for i=2:heigth-1

for j=2:width-1

if R(i,j)>0.01\*Rmax

tm=R(i-1:i+1,j-1:j+1);

tm(2,2)=0;

if R(i,j)>tm

pMap(i,j)=1;

end

end

end

end

[row,col]=find(pMap==1);

points=[row,col];

function drawHarrisCorner(pic1,points1,pic2,points2)

% 功能：画出 Harris 角点的连接

% 输入：

% pic1、pic2：待拼接的图像

% points1、points2：Harris 角点位置

X1=points1(:,2);

Y1=points1(:,1);

X2=points2(:,2);

Y2=points2(:,1);

dif=size(pic1,2);

imshowpair(pic1,pic2,'montage');

hold on

plot(X1,Y1,'b\*');

plot(X2+dif,Y2,'b\*');

set(gcf,'Color','w');

function des=myHarrisCornerDescription(pic,points)

% 功能：Harris 角点特征描述

% 输入：

% pic：原图像

% points：角点位置

% 输出：

% des：8×N 的角点特征描述矩阵

if length(size(pic))==3

pic=rgb2gray(pic);

end

len=length(points);

des=zeros(8,len);

for k=1:len

p=points(k,:);

pc=pic(p(1),p(2));

des(1,k)=pic(p(1)-1,p(2)-1)-pc;

des(2,k)=pic(p(1),p(2)-1)-pc;

des(3,k)=pic(p(1)+1,p(2)-1)-pc;

des(4,k)=pic(p(1)+1,p(2))-pc;

des(5,k)=pic(p(1)+1,p(2)+1)-pc;

des(6,k)=pic(p(1),p(2)+1)-pc;

des(7,k)=pic(p(1)-1,p(2)+1)-pc;

des(8,k)=pic(p(1)-1,p(2))-pc;

des(:,k)=des(:,k)/sum(des(:,k));

end

function matchs=myMatch(des1,des2)

% 功能：特征点双向匹配

% 输入：

% des1、des2：特征点描述信息构成的矩阵

% 输出：

% matchs：匹配的特征点对应关系

len1=length(des1);

len2=length(des2);

match1=zeros(len1,2);

cor1=zeros(1,len2);

for i=1:len1

d1=des1(:,i);

for j=1:len2

d2=des2(:,j);

cor1(j)=(d1'\*d2)/sqrt((d1'\*d1)\*(d2'\*d2));

end

[~,indx]=max(cor1);

match1(i,:)=[i,indx];

end

match2=zeros(len2,2);

cor2=zeros(1,len1);

for i=1:len2

d2=des2(:,i);

for j=1:len1

d1=des1(:,j);

cor2(j)=(d1'\*d2)/sqrt((d1'\*d1)\*(d2'\*d2));

end

[~,indx]=max(cor2);

match2(i,:)=[indx,i];

end

matchs=[];

for i=1:length(match1)

for j=1:length(match2)

if match1(i,:)==match2(j,:)

matchs=[matchs;match1(i,:)];

end

end

end

function drawLinedCorner(pic1,loc1,pic2,loc2)

% 功能：画出匹配角点的连接

% 输入：

% pic1、pic2：待拼接的图像

% loc1、loc2：匹配角点位置

X1=loc1(:,2);

Y1=loc1(:,1);

X2=loc2(:,2);

Y2=loc2(:,1);

dif=size(pic1,2);

imshowpair(pic1,pic2,'montage');

hold on

for k=1:length(X1)

plot(X1(k),Y1(k),'b\*');

plot(X2(k)+dif,Y2(k),'b\*');

line([X1(k),X2(k)+dif],[Y1(k),Y2(k)],'Color','r');

end

set(gcf,'Color','w');

function [newLoc1,newLoc2]=pointsSelect(loc1,loc2)

% 功能：筛选匹配特征点对，获取高精度的控制点

% 输入：

% loc1、loc2：粗匹配特征点位置

% 输出：

% newLoc1、newLoc2：精匹配控制点位置

slope=(loc2(:,1)-loc1(:,1))./(loc2(:,2)-loc1(:,2));

for k=1:3

slope=slope-mean(slope);

len=length(slope);

t=sort(abs(slope));

thresh=t(round(0.5\*len));

ind=abs(slope)<=thresh;

slope=slope(ind);

loc1=loc1(ind,:);

loc2=loc2(ind,:);

end

newLoc1=loc1;

newLoc2=loc2;

function im=picMatched(pic1,newLoc1,pic2,newLoc2)

% 功能：获取拼接之后的图片

% 输入：

% pic1、pic2：待拼接图片

% newLoc1、newLoc2：变换控制点矩阵

% 输出：

% im：拼接后的图片

if length(size(pic1))==2

pic1=cat(3,pic1,pic1,pic1);

end

if length(size(pic2))==2

pic2=cat(3,pic2,pic2,pic2);

end

SZ=2000;

X1=newLoc1(:,2);

Y1=newLoc1(:,1);

X2=newLoc2(:,2);

Y2=newLoc2(:,1);

sel=randperm(length(newLoc1),3);

x=X2(sel)';

y=Y2(sel)';

X=X1(sel)';

Y=Y1(sel)';

U=[x;y;ones(1,3)];

V=[X;Y;ones(1,3)];

T=V/U;

cntrX=SZ/2;

cntrY=SZ/2;

im=zeros(SZ,SZ,3);

for i=1:size(pic2,1)

for j=1:size(pic2,2)

tmp=T\*[j;i;1];

nx=round(tmp(1))+cntrX;

ny=round(tmp(2))+cntrY;

if nx>=1 && nx<=SZ && ny>=1 && ny<=SZ

im(ny,nx,:)=pic2(i,j,:);

end

end

end

im=imresize(im,1,'bicubic');

tpic1=zeros(SZ,SZ,3);

tpic1(1+cntrY:size(pic1,1)+cntrY,1+cntrX:size(pic1,2)+cntrX,:)=pic1;

re=rgb2gray(uint8(im))-rgb2gray(uint8(tpic1));

for k=1:3

ta=im(:,:,k);

tb=tpic1(:,:,k);

ta(re==0)=tb(re==0);

im(:,:,k)=ta;

end

clear ta tb re tpic1

im=getPicture(im,SZ);

im=uint8(im);

if length(size(pic1))==2

im=rgb2gray(im);

end

function im=getPicture(pic,SZ)

% 功能：获取图像有用区域

% 输入：

% pic：拼接图像

% SZ：预定图像尺寸

% 输出：

% im：有用区域图像

if length(size(pic))==2

pic=cat(3,pic,pic,pic);

end

k=1;

while k<SZ

if any(any(pic(k,:,:)))

break

end

k=k+1;

end

ceil=k; %上边界

k=SZ;

while k>0

if any(any(pic(k,:,:)))

break

end

k=k-1;

end

bottom=k; %下边界

k=1;

while k<SZ

if any(any(pic(:,k,:)))

break

end



k=k+1;

end

left=k; %左边界

k=SZ;

while k>0

if any(any(pic(:,k,:)))

break

end

k=k-1;

end

right=k; %右边界

%%获取图像

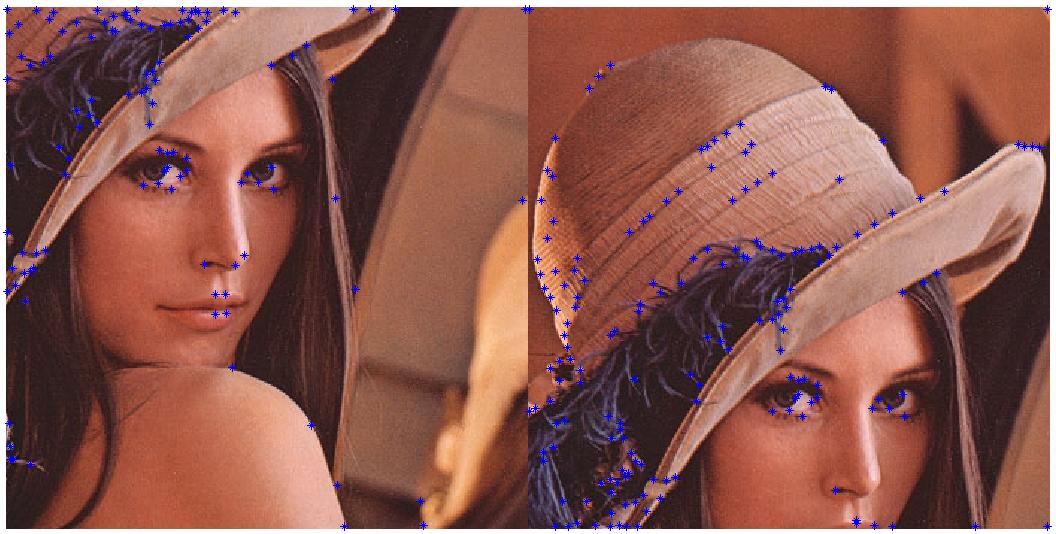
im=pic(ceil:bottom,left:right,:);

运行结果：

A 组实验图

拼接图像 A 组

检测出的 Harris 角点



粗匹配的角点对连线

去除无匹配之后的角点对连线

