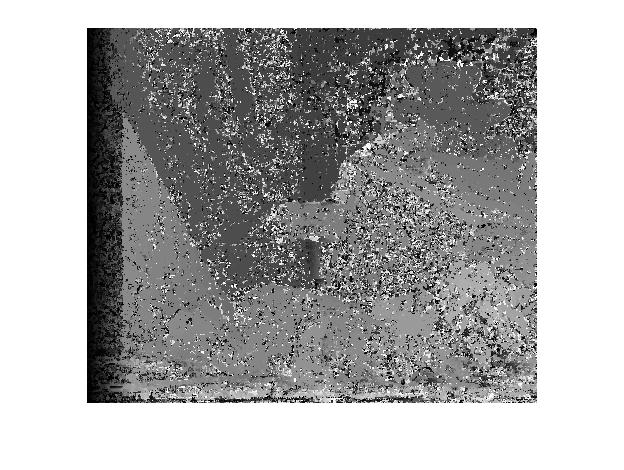
# Homework2

CS532: Homework Assignment 2

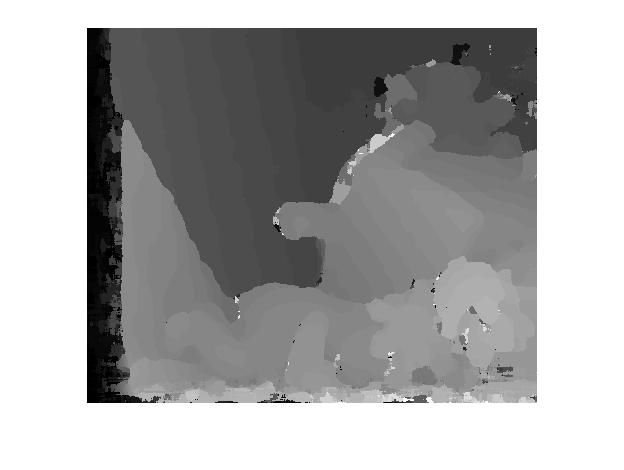
Problem1:

The disparity maps:

For the 3x3 window , the error rate is 48.09% and for the 15x15 window , the error rate is decrease to 22%.



3X3 SAD Window and the error rate is 48.09%



15X15 SAD Window. Error rate = 22%.

Problem 2

The value of PKRN is C2/C1, so when the c2 is equal to c1 that means the number is equal to 1 .These value need to be ignored in this evaluation. Then, using the 50% most confident pixels to compute the error rate and found the error rate is 39.05%.Beside the number of pixels that have been kept is 76880.

Source Code:

%Tianpei Luo @cs532 homework2

left=imread('teddyL.pgm');

right=imread('teddyR.pgm');

disp=imread('disp2.pgm');

%get the size of the image

height=size(left,1);

width=size(left,2);

a=[height,width];

display(a);

imshow(left);

%figure,imshow(right);

%figure,imshow(disp);

%implement the rank transform

rank = 5;

rkleftt = zeros(size(left));

rkrightt =zeros(size(right));

half\_rk = (rank-1)/2;

for i=1:height

for j=1:width

%set lims to avoid out of boundary

lims = [max(1,i-half\_rk);min(height,i+half\_rk)

max(1,j-half\_rk);min(width,j+half\_rk)];

%catch the pixel from image in the window size

lwindow=double(left(lims(1):lims(2),lims(3):lims(4)));

%compare with the origin pixel and sum the intensity

rkleftt(i,j) = sum(sum(lwindow<left(i,j)));

rwindow=double(right(lims(1):lims(2),lims(3):lims(4)));

rkrightt(i,j) = sum(sum(rwindow<right(i,j)));

end

end

%implement the SAD stereo matching

window = 3;

%window = 15;

disparities = zeros(size(left));

pkrn = zeros(size(left));

su=zeros(size(left));

c1=zeros(size(left));

c2=zeros(size(left));

half\_win=(window-1)/2;

count=0;

for i = 1:height

for j = 1:width

suMin=rank\*rank\*255;

c1(i,j)=rank\*rank\*255;

c2(i,j)= c1(i,j)+1;

for d = 0:64

su(i,j)=0;

for q =max(1,i-half\_win):min(height,i+half\_win)

for p=max(1,j-half\_win):min(width,j+half\_win)

pixLeft = rkleftt(q,p);

if p-d<=0

pixRight=0;

else

pixRight=rkrightt(q,p-d);

end

su(i,j)= su(i,j)+abs(pixLeft-pixRight);

end

end

if su(i,j) < suMin

suMin = su(i,j);

disparities(i,j) = d;

end

%find out the global minimum of the cost curve c1;

if su(i,j)<c1(i,j)

cur = c1(i,j);

c1(i,j)=su(i,j);

c2(i,j)=cur;

else

if su(i,j) <=c2(i,j)

c2(i,j)= su(i,j);

end

end

end

pkrn(i,j)=c2(i,j)/c1(i,j);

if pkrn(i,j)==1.0000

count=count+1;

end

end

end

figure,imshow(disparities);

%coumputing errors

disparities = double(disparities);

figure,imshow(disparities,[0 63]);

%caculate the error rate by comparing with the ground true image

ground\_t = imread('disp2.pgm');

ground\_t = double(ground\_t)./4;

err = sum(sum(abs(ground\_t-disparities)>1))/(height\*width);

display(err);

%PKRN error

% ignored in this evaluation

display(count);

% find the number that is below the top 50%most confidence pixels

pkrn\_sort=pkrn(:);

pkrn\_sort=sort(pkrn\_sort);

pkrn\_sort(1:count)=[];

display(length(pkrn\_sort));

middle=pkrn\_sort(floor(length(pkrn\_sort)/2));

display(middle);

count\_1=0;

count\_2=0;

for i=1:height

for j=1:width

if pkrn(i,j)> middle

count\_1=count\_1+1;

if abs(disparities(i,j)-ground\_t(i,j))>1

count\_2=count\_2+1;

end

end

end

end

pkrn\_err\_rate=count\_2/count\_1;

display(count\_1);

display(pkrn\_err\_rate);