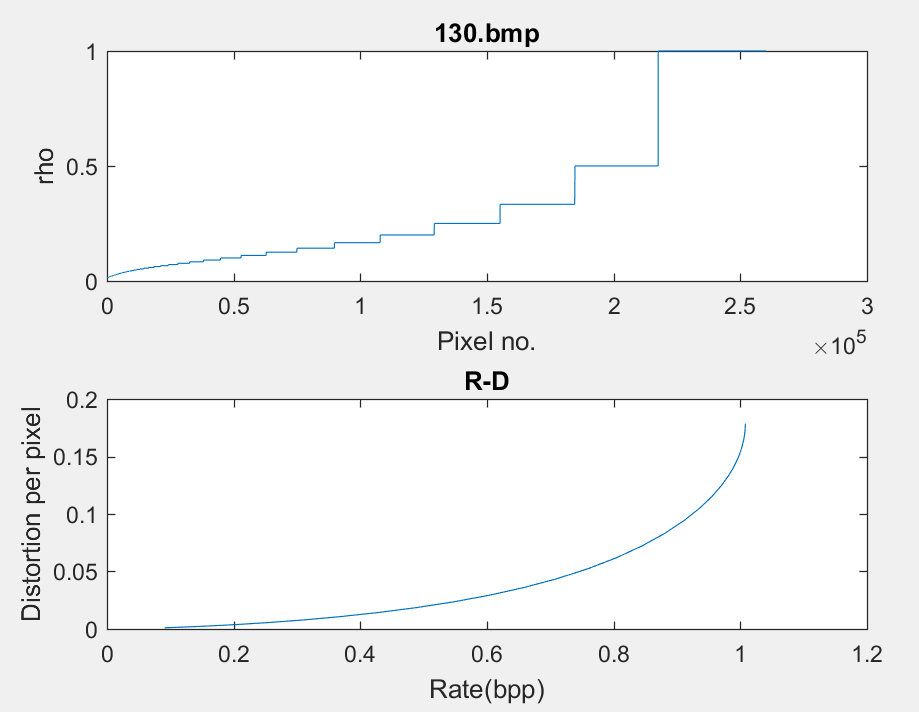
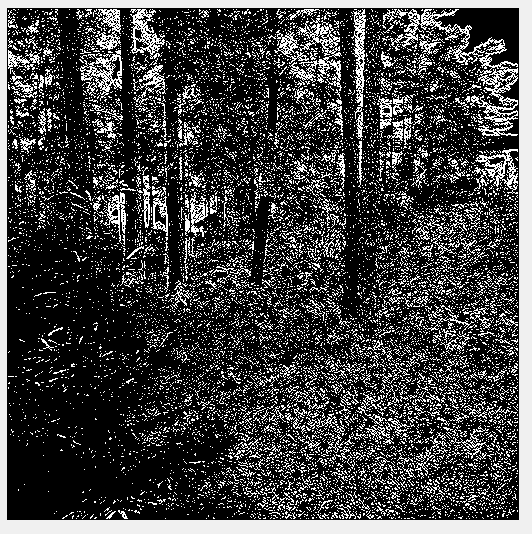
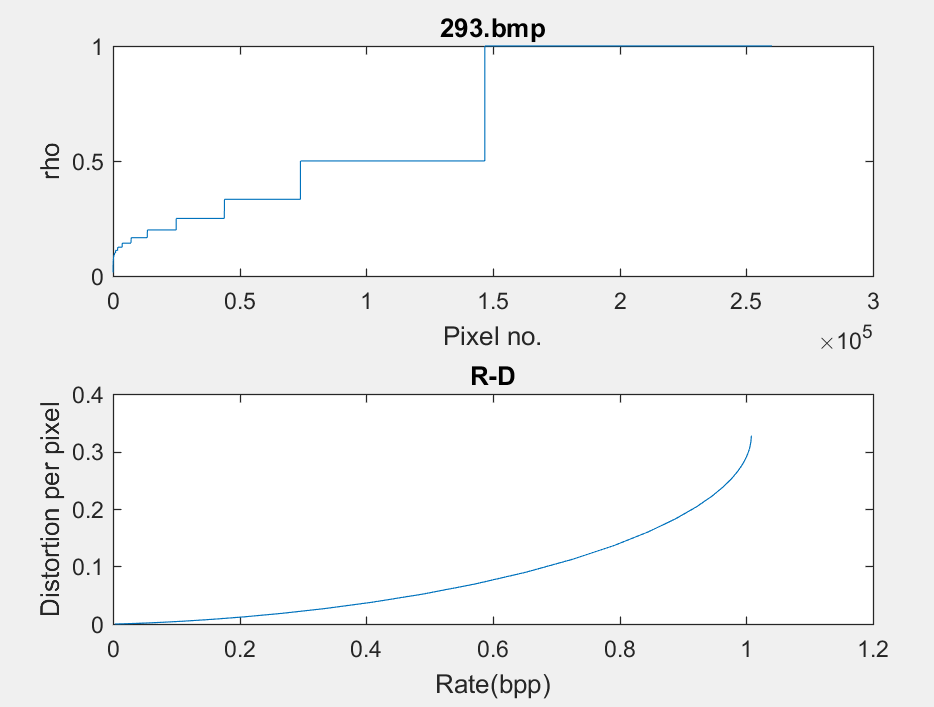
Xin Wen









3.1 130.bmp alpha is around 0.75 293.bmp, alpha is around 0.48

3.2 For the forest graph, the texture is complex and modifications are hard to be noticed. In the cost view, the surrounding pixel are more likely to be different, which make the pixels have a relatively higher Rij and lower ρij. But for the ocean graph, the sky area and water area are relaively similar, which makes it have higher ρij in embedding

3.3 130.bmp d is around 0.019 293.bmp, d is around 0.05

3.4 The choosen pixels are in the figures

MATLAB Code

function [ cost ] = RD( fileName )

img = imread(fileName);

[m, n] = size(img);

R = zeros(m, n); % R and cost

cost = zeros(m,n);

for i = 2:m-1

for j = 2:n-1

cur = img(i, j);

R(i, j) = 1/4 \*(abs(cur-img(i, j-1)) + abs(cur-img(i, j+1)) + abs(cur-img(i-1, j)) + abs(cur-img(i+1, j)));

end

end

cost = 1 ./ (1 + R);

cost\_temp = cost(2:m-1, 2:n-1);

cost\_temp = sort(reshape(cost\_temp, (m-2)\*(n-2), 1));

subplot(2,1,1);

plot(cost\_temp); % plot costs

xlabel('Pixel no.');

ylabel('rho');

title(strcat(fileName));

lambda = zeros(51, 1); % initializing

alpha = zeros(51, 1);

d = zeros(51, 1);

for i = 1:51

lambda(i) = 1.2^(-31 + i);

l = lambda(i);

e = exp(-l .\* cost);

alpha(i) = 1./((m-2)\*(n-2)) \* sum(sum(h(1./(1+e))));

d(i) = 1./((m-2)\*(n-2)) \* sum(sum(cost.\*(e./(1+e))));

end

subplot(2,1,2);

plot(alpha, d); % plot R-D

xlabel('Rate(bpp)');

ylabel('Distortion per pixel');

title('R-D');

lambda = 1.2^(-31+31);

prob = zeros(m, n);

for i = 2:m-1

for j = 2:n-1

miu = -1/lambda;

prob(i, j) = 1 / (1 + exp(-cost(i,j)/miu));

end

end

prob\_temp = sort(reshape(prob(2:m-1, 2:n-1), (m-2)\*(n-2),1 ));

threshold = prob\_temp((m-2)\*(n-2)-50000); % select the largest 50000 pixel with highest probability

newimg = zeros(m, n);

for i = 1:m

for j = 1:n

if prob(i, j) >= threshold

newimg(i, j) = 255;

end

end

end

figure(2);

imshow(newimg);

end