**Programming Assignment**

**Image and Video Processing**

**Week 5 – Video Basics**

Problem 1

%frameDiff.m

%Subtracts every frame from previous frame and compiles all these

%differential frames into a new video

%Read source video into an object and initialize source and

%target video structs

vidObj = VideoReader('Radiohead.mp4');

nFrames = vidObj.NumberOfFrames;

vidH = vidObj.Height;

vidW = vidObj.Width;

mov(1:nFrames) = struct('cdata',zeros(vidH,vidW,3,'uint8'),'colormap',[]);

movDiff(1:nFrames) = struct('cdata',zeros(vidH,vidW,3,'uint8'),'colormap',[]);

%Write to source struct of frames

disp('Reading...');

for k = 5000:6500

mov(k).cdata = read(vidObj,k);

end

%Initialize target video file

outVid = VideoWriter('darkLotus4.avi');

outVid.FrameRate = vidObj.FrameRate;

open(outVid);

dim = size(mov(100).cdata);

index = 1;

disp('Writing...');

for k = 5000:6500

%Compute absolute difference between itself and subsequent frame

temp = abs(mov(k+1).cdata - mov(k).cdata);

%Thresholding values in each frame

for i = 1:dim(1)

for j = 1:dim(2)

if min(temp(i,j,:)) > 20

temp(i,j,:) = [255 255 255];

end

end

end

%Assigning frame to target struct and write to target video file

movDiff(index).cdata = temp;

writeVideo(outVid,movDiff(index).cdata);

index = index + 1;

end

close(outVid);

****

****

Source Video: https://www.youtube.com/watch?v=cfOa1a8hYP8

Video demo: [https://vimeo.com/157920432](https://vimeo.com/157920432" \t "_blank)

(rendered using the above code with RGB specific thresholding)

Problem 2

function [ xm, ym, matchblock] = computeEBMA(template,img,x0,y0,Rx,Ry)

% A function that jumps in increments of Rx and Ry across a given image and

% superimposes the template to find out best match.

%Compute dimensions

[H, W] = size(img);

[bH, bW] = size(template);

%Initialize values

maxError = bH\*bW\*255;

xm = x0; ym = y0;

matchblock = 0;

for k = (max(1,x0-Rx):min(W-bW,x0+Rx))

for l = (max(1,y0-Ry):min(H-bH,y0+Ry))

%Initialize a candidate block with the same size as the template

candidate = img(l:l+bH-1,k:k+bW-1);

error = sum(sum(abs(template-candidate)));%Absolute Difference

%Minimize error with every iteration

if error < maxError

xm = k;

ym = l;

matchblock = candidate;

maxError = error;

end

end

end

end



Figure : Frame 2357



Figure : Matchblock



Figure : Frame 2358



Figure : Matchblock