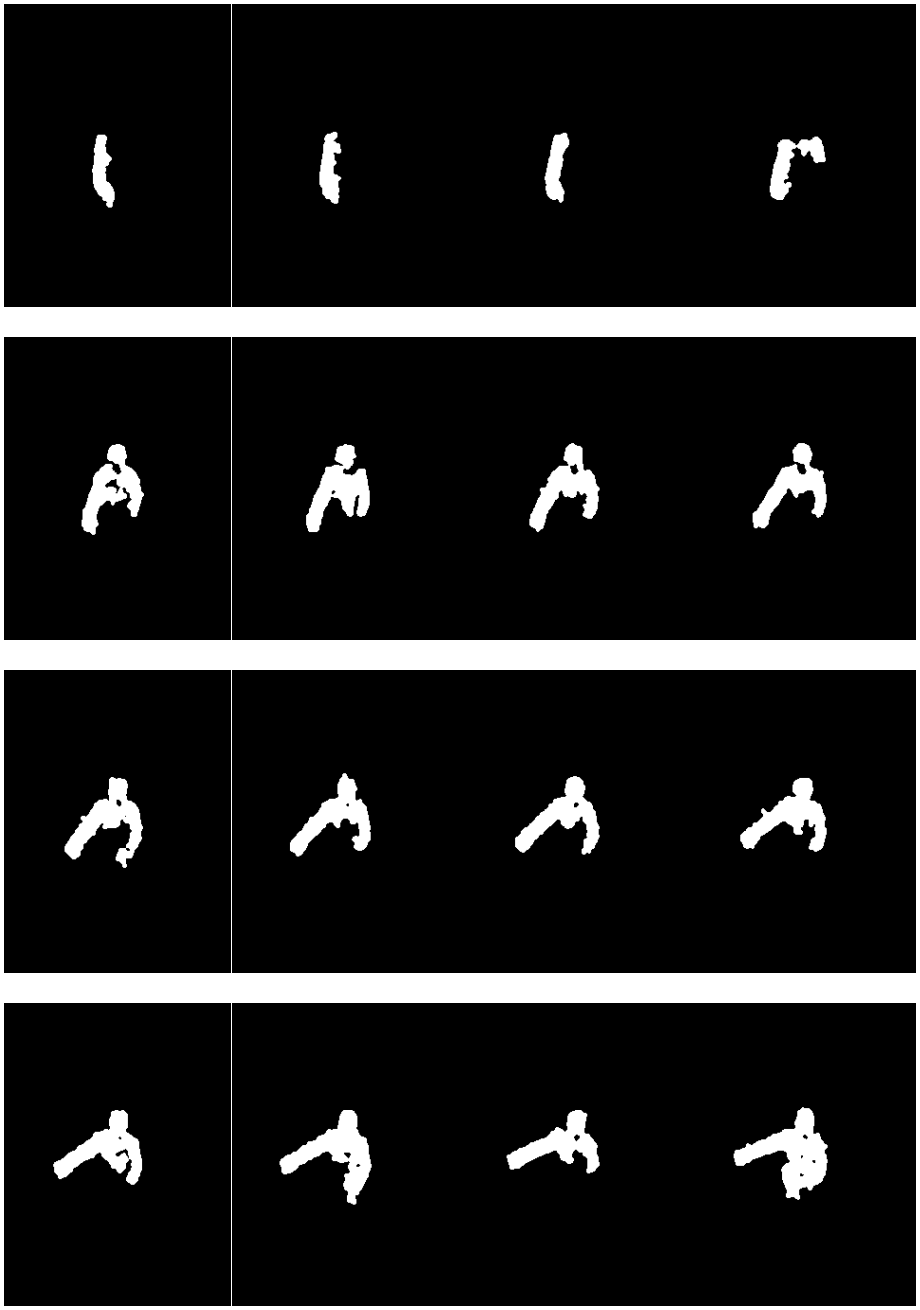
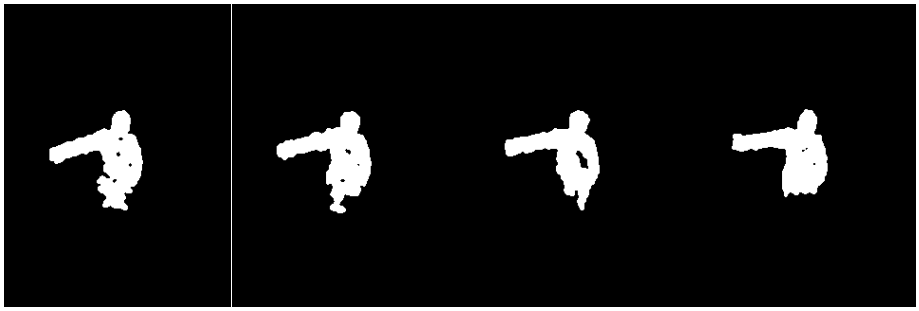
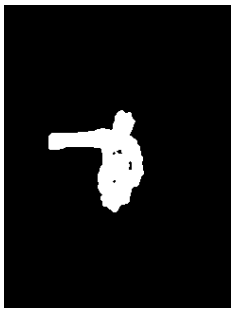
Yukun Duan
Computer Vision for HCI AU'22
Homework Assignment #5

1.
I tried lots of T (The rest of the images are shown in the folder) and here I use T = 4 for differencing. After differencing, I removed tiny regions, perform dilation, and finally smoothed through the median filter. Here is the resulting differencing image from Image 2 to image 22. Initially, the only thing in the image is the left hand because other body parts do not move too much. Later, the head and upper body are shown in the image because these parts are moving as her arm moves up.

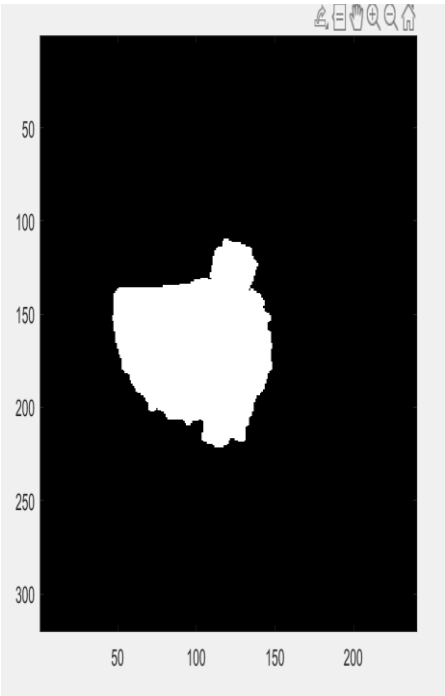




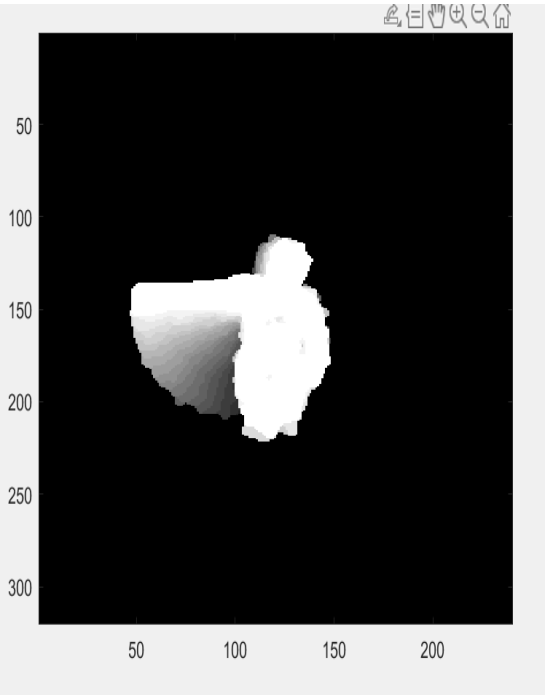


2.

Here is the MEI image:



MHI image:



Here is the seven similitude I calculated for MEI:

0.0822 0.0002 0.0008 0.0069 0.0907 -0.0035 -0.0076

MHI:

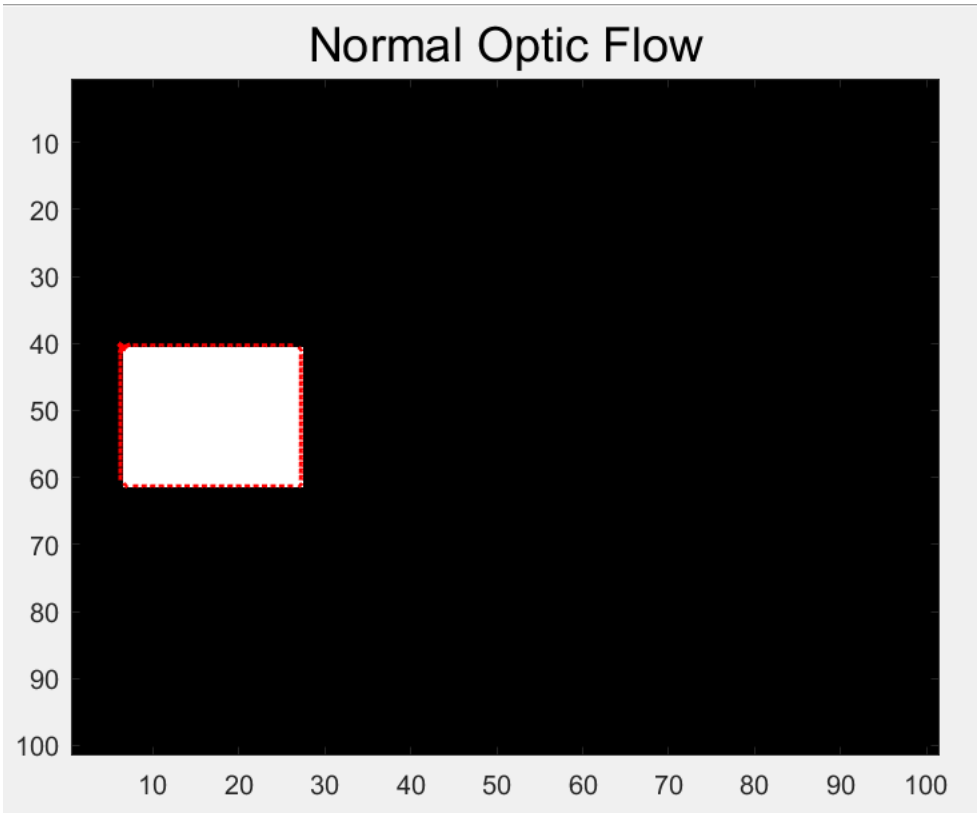
0.0970 0.0055 0.0094 0.0089 0.1092 -0.0047 -0.0178

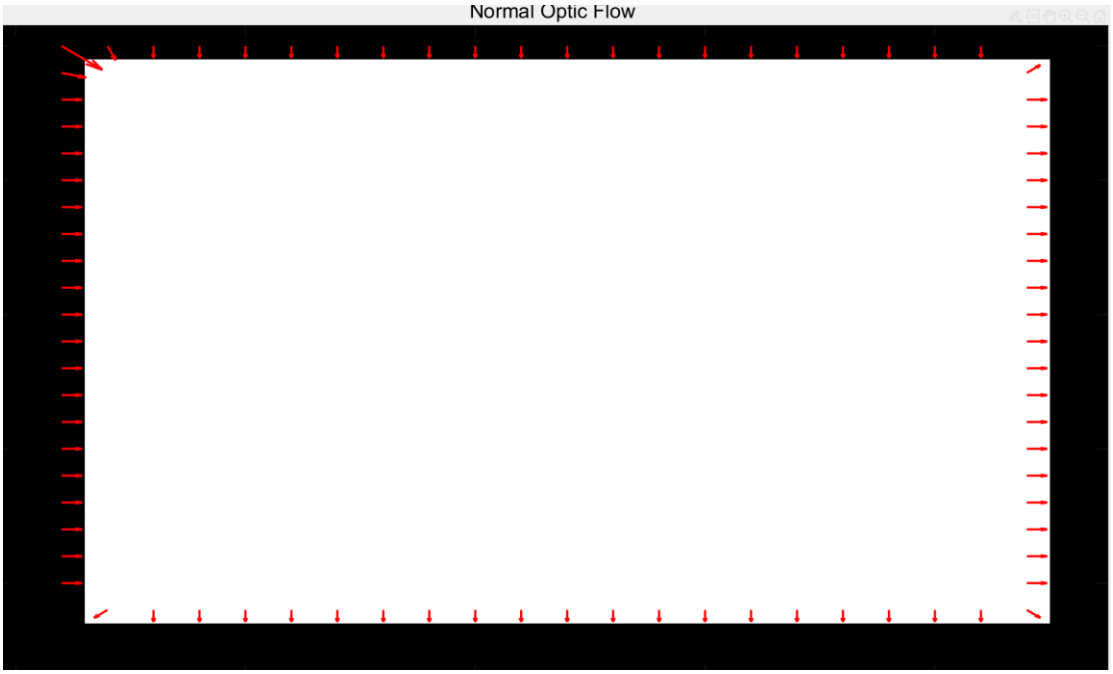
From MHI image, we can see the whole tracks of the arm, from bottom to top. The intensity is faded from top to bottom because the most recent image has the highest intensity, and

the intensity change indicates the direction in which she moves her arm. However, the MEI does not provide any information about direction.

3.

Here is the normal optic flow I get, the bottom one is the image I zoomed out:





The flow is exactly what I expected. The arrows on the x-axis are pointing towards the bottom

because the second picture is moved down by 1 pixel (if we only care about the vertical movement). The arrows on the y-axis are pointing towards the right-hand side because the second picture is moved towards the right by 1 pixel (here we only consider horizontal movement). The left top arrows and right bottom arrows are both pointing towards the left bottom side because these two points on the corner are moved towards the right and bottom. However, the point on the left bottom side and right top side are moving outwards because the direction is influenced by ft, while the value of the left bottom point on ft is positive (so -ft is negative), which directory switched sign of fx and fy (change fx to negative and fy to positive). Thus, the left bottom point is pointing towards the left bottom side. Similarly, the fx on the right top point is positive originally, and fy is negative. The ft is negative (since it's 0 on image2 and 1 on image1), which switched the sign of fx and fy as well. The vector is pointing to the right top side after the sign has been switched.

Attached Code:

```
% Yukun Duan
% CSE5524 - HW5
% 9/24/2022


%% Problem 1


T = 1;
for i = 2:22
    % get image absolute difference
    im = double(imread(sprintf('aerobic-%03d.bmp',i)));
    img2 = double(imread(sprintf('aerobic-%03d.bmp',i-1)));
    dif = abs(img2 - im);
    % Threshold, remove tiny regions, dilate, median filter
    dif(dif >= T) = 255;
    dif(dif < T) = 0;
    dif = bwareaopen(dif,150,8);
    dif = imdilate(dif, strel('square', 4));
    dif = medfilt2(dif);
    imwrite(dif, sprintf('T%d-%d.png',T,i))
    imshow(dif)
    pause;
end


T = 2;
for i = 2:22
    % get image absolute difference
    im = double(imread(sprintf('aerobic-%03d.bmp',i)));
```

```
    img2 = double(imread(sprintf('aerobic-%03d.bmp',i-1)));
    dif = abs(img2 - im);
    % Threshold, remove tiny regions, dilate, median filter
    dif(dif >= T) = 255;
    dif(dif < T) = 0;
    dif = bwareaopen(dif,150,8);
    dif = imdilate(dif, strel('square', 4));
    dif = medfilt2(dif);
    imwrite(dif, sprintf('T%d-%d.png',T,i))
    imshow(dif)
    pause;
end

T = 4;
for i = 2:22
    % get image absolute difference
    im = double(imread(sprintf('aerobic-%03d.bmp',i)));
    img2 = double(imread(sprintf('aerobic-%03d.bmp',i-1)));
    dif = abs(img2 - im);
    % Threshold, remove tiny regions, dilate, median filter
    dif(dif >= T) = 255;
    dif(dif < T) = 0;
    dif = bwareaopen(dif,150,8);
    dif = imdilate(dif, strel('square', 4));
    dif = medfilt2(dif);
    imwrite(dif, sprintf('T%d-%d.png',T,i))
    imshow(dif)
    pause;
end

T = 10;
for i = 2:22
    % get image absolute difference
    im = double(imread(sprintf('aerobic-%03d.bmp',i)));
    img2 = double(imread(sprintf('aerobic-%03d.bmp',i-1)));
    dif = abs(img2 - im);
    % Threshold, remove tiny regions, dilate, median filter
    dif(dif >= T) = 255;
    dif(dif < T) = 0;
    dif = bwareaopen(dif,150,8);
    dif = imdilate(dif, strel('square', 4));
    dif = medfilt2(dif);
    imwrite(dif, sprintf('T%d-%d.png',T,i))
    imshow(dif)
```

```
    pause;
end

T = 20;
for i = 2:22
    % get image absolute difference
    im = double(imread(sprintf('aerobic-%03d.bmp',i)));
    img2 = double(imread(sprintf('aerobic-%03d.bmp',i-1)));
    dif = abs(img2 - im);
    % Threshold, remove tiny regions, dilate, median filter
    dif(dif >= T) = 255;
    dif(dif < T) = 0;
    dif = bwareaopen(dif,150,8);
    dif = imdilate(dif, strel('square', 4));
    dif = medfilt2(dif);
    imwrite(dif, sprintf('T%d-%d.png',T,i))
    imshow(dif)
    pause;
end

T = 40;
for i = 2:22
    % get image absolute difference
    im = double(imread(sprintf('aerobic-%03d.bmp',i)));
    img2 = double(imread(sprintf('aerobic-%03d.bmp',i-1)));
    dif = abs(img2 - im);
    % Threshold, remove tiny regions, dilate, median filter
    dif(dif >= T) = 255;
    dif(dif < T) = 0;
    dif = bwareaopen(dif,150,8);
    dif = imdilate(dif, strel('square', 4));
    dif = medfilt2(dif);
    imwrite(dif, sprintf('T%d-%d.png',T,i))
    imshow(dif)
    pause;
end

T = 50;
for i = 2:22
    % get image absolute difference
    im = double(imread(sprintf('aerobic-%03d.bmp',i)));
    img2 = double(imread(sprintf('aerobic-%03d.bmp',i-1)));
    dif = abs(img2 - im);
    % Threshold, remove tiny regions, dilate, median filter
```

```
    dif(dif >= T) = 255;
    dif(dif < T) = 0;
    dif = bwareaopen(dif,150,8);
    dif = imdilate(dif, strel('square', 4));
    dif = medfilt2(dif);
    imwrite(dif, sprintf('T%d-%d.png',T,i))
    imshow(dif)
    pause;
end

%% Problem 2
mei = zeros([320 240]);
mhi = zeros([320 240]);

for i = 2:22
    im = imread(sprintf('T4-%d.png',i));
    mei = mei + im;
    mhi(im > 0) = i;
end

% normalize mei to binary
mei(mei>0)=1;
imagesc(mei)
colormap('gray')
pause;


% normalize mhi between 0 - 1
mhi = max(0, (mhi - 1)/21);
imagesc(mhi)
colormap('gray')

imwrite(mei, 'MEI.png')
imwrite(mhi, 'MHI.png')

% compute 7 similityde moments for MEI & MHI
disp(similitudeMoments(mei))
pause;
disp(similitudeMoments(mhi))
pause;
```

```
%% Problem 3
% Create the Image and place the box
image1 = double(zeros([101 101]));
image2 = double(zeros([101 101]));
box = double(ones([21 21]) * 255);
image1(40:60, 6:26) = box;
image2(41:61, 7:27) = box;
% sobel filter
hx = [-1 0 1; -2 0 2; -1 0 1]/8;
hy = [-1 -2 -1; 0 0 0;1 2 1]/8;
fx = imfilter(image2, hx);
fy = imfilter(image2, hy);
fxy = sqrt(fx.^2 + fy.^2);
fxy(fxy == 0) = 1; % prevent zero dividing
ft = image2 - image1;
fx = fx./ fxy;
fy = fy./ fxy;
ft = ft./ fxy;
fx = fx.* ft * -1;
fy = fy.* ft * -1;

xind = repmat(1:size(image2,2),size(image2,1),1); % col => x
yind = repmat((1:size(image2,1))', 1, size(image2,2)); % row => y

imagesc(image2)
colormap('gray')
hold on
quiver(xind,yind,fx, fy, 'color', [1 0 0], 'linewidth', 2)
set(gca,'Ydir','reverse')
title('Normal Optic Flow', 'fontsize', 18)
hold off

%% Similitude Calculation
function Nvals = similitudeMoments(boxIm)
    Nvals = [];
    % initialize matrix for row index, col index, x average and y average.
    xIndex = repmat(1:size(boxIm,2),size(boxIm,1),1); % col => x
    yIndex = repmat((1:size(boxIm,1))', 1, size(boxIm,2)); % row => y
    m00 = sum(boxIm, 'all');
    m10 = sum(xIndex.*boxIm, 'all');
    m01 = sum(yIndex.*boxIm, 'all');
    xbar = ones(size(boxIm)) * m10/m00;
    ybar = ones(size(boxIm)) * m01/m00;
    % iteratively calculate 7 similitude moments
```

```
    for i = 0:3
        for j = max(0,(2-i)):(3-i)
            % 2 <= (i+j) <= 3
            nij = sum(((xIndex - xbar).^i).*((yIndex - ybar).^j).*boxIm,
'all')/(m00.^((i+j)/2+1));
            img = ((xIndex - xbar).^i).*((yIndex-
ybar).^j).*boxIm/(m00.^((i+j)/2+1));
            Nvals = [Nvals, nij];
        end
    end
end
```