

Nama : Luthfi Aminulloh
NRP : 1020181013
Kelas : TE2018

Computer Vision & Image Processing

Task 4: Template Matching

Tugas 1. Mendeteksi jumlah orang pada gambar dengan template matching.



Listing Program:

```
clc;
clear;

%Membaca gambar
x=imread('2.jpg');
xg=rgb2gray(x);
xd=double(xg);
xdm=mean(mean(xd));
xd=xd-xdm;

%Membaca template
h1=imread('2f.jpg');
h2=rgb2gray(h1);
hd=double(h2);
h3=hd/sum(sum(hd));
hm=mean(mean(h3));
h=h3-hm;

%Proses filtering
yd=filter2(h,xd);
ymax=max(max(yd));
y1=yd*255/ymax;
y=uint8(y1);

%Proses thresholding
```



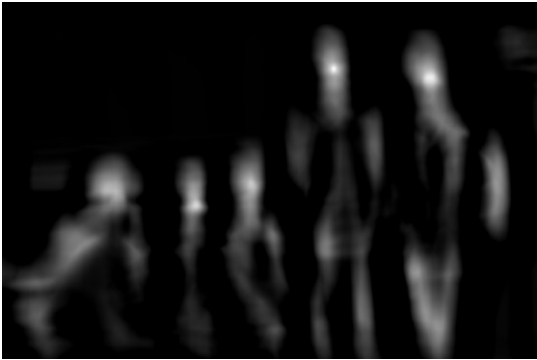
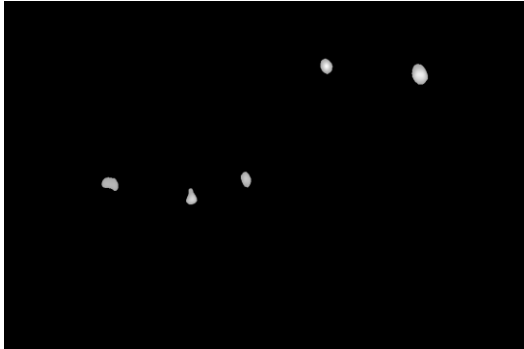
```

th=0.45;
y2=(yd>=th*ymax);
y3=yd.*y2;
y4=y3*255/ymax;
y5=uint8(y4);

%Mwnampilkan hasil
u=[xg y y5];
imshow(u);

```

Hasil Percobaan:

	
Original Image	Template
	
Hasil zero-mean filter	Zero-mean filter setelah threshold

Analisa:

Dari percobaan yang telah dilakukan dapat dianalisis bahwa dengan menggunakan metode zero-mean filter untuk mendeteksi jumlah orang dengan menggunakan template wajah berhasil dilakukan dengan mendapatkan 5 nilai yang melebihi threshold, dimana nilai threshold yang digunakan adalah sebesar 0.65, dimana nilai tersebut didapatkan secara tuning intuitively dari beberapa percobaan yang telah dilakukan.

Tugas 2. Mendeteksi jumlah orang pada gambar dengan template matching..



Listing Program:

```
clc; % Clear the command window.
close all; % Close all figures
imtool close all; % Close all imtool figures.
clear; % Erase all existing variables.
workspace; % Make sure the workspace panel is showing.
format long g;
format compact;
fontSize = 11;

rgbImage = imread('2.jpg'); % Original Image
% Get the dimensions of the image. numberOfColorBands should be = 3.
[rows, columns, numberOfColorBands] = size(rgbImage);
% Display the original color image.
subplot(2, 2, 1);
imshow(rgbImage, []);
axis on;
caption = sprintf('Original Color Image, %d rows by %d columns.', rows,
columns);
title(caption, 'FontSize', fontSize);
% Enlarge figure to full screen.
set(gcf, 'units','normalized','outerposition',[0, 0, 1, 1]);

% Let's get our template by extracting a small portion of the original
image.
templateWidth = 71
templateHeight = 49
smallSubImage = imread('2a.jpg'); % Template
% Get the dimensions of the image. numberOfColorBands should be = 3.
[rows, columns, numberOfColorBands] = size(smallSubImage);
subplot(2, 2, 2);
imshow(smallSubImage, []);
```

```

axis on;
caption = sprintf('Template Image to Search For, %d rows by %d columns.',
rows, columns);
title(caption, 'FontSize', fontSize);

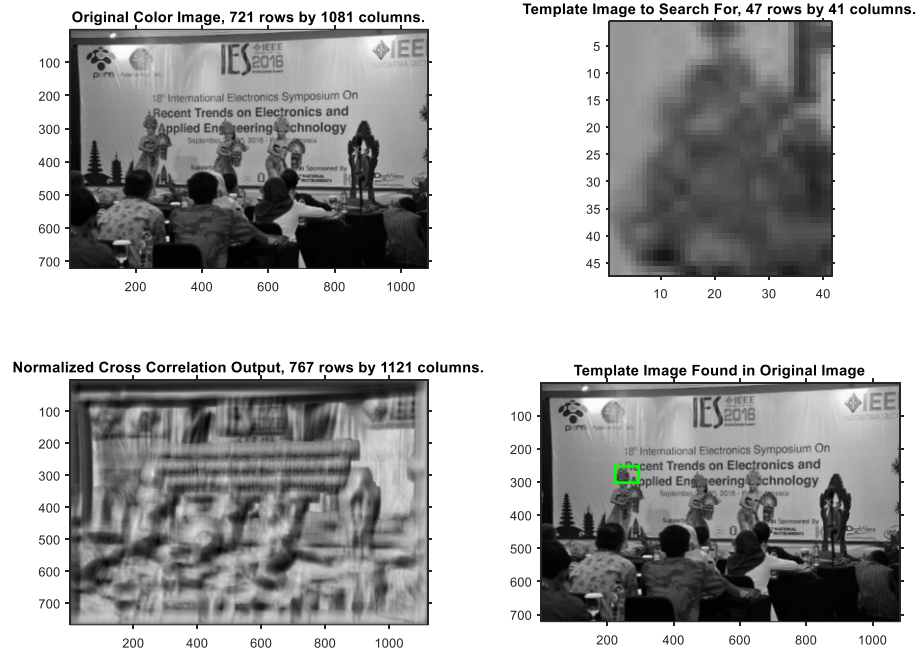
% Note: If you want, you can get the template from every color channel and
search for it in every color channel,
% then take the average of the found locations to get the overall best
location.
channelToCorrelate = 1; % Use the red channel.
correlationOutput = normxcorr2(smallSubImage(:, :, 1), rgbImage(:, :,
channelToCorrelate));
subplot(2, 2, 3);
imshow(correlationOutput, []);
axis on;
% Get the dimensions of the image. numberOfColorBands should be = 1.
[rows, columns, numberOfColorBands] = size(correlationOutput);
caption = sprintf('Normalized Cross Correlation Output, %d rows by %d
columns.', rows, columns);
title(caption, 'FontSize', fontSize);

% Find out where the normalized cross correlation image is brightest.
[maxCorrValue, maxIndex] = max(abs(correlationOutput(:)));
[yPeak, xPeak] = ind2sub(size(correlationOutput), maxIndex(1))
% Because cross correlation increases the size of the image,
% we need to shift back to find out where it would be in the original image.
corr_offset = [(xPeak-size(smallSubImage,2)) (yPeak-size(smallSubImage,1))]

% Plot it over the original image.
subplot(2, 2, 4); % Re-display image in lower right.
imshow(rgbImage);
axis on; % Show tick marks giving pixels
hold on; % Don't allow rectangle to blow away image.
% Calculate the rectangle for the template box. Rect = [xLeft, yTop,
widthInColumns, heightInRows]
boxRect = [corr_offset(1) corr_offset(2) templateWidth, templateHeight]
% Plot the box over the image.
rectangle('position', boxRect, 'edgecolor', 'g', 'linewidth', 2);
% Give a caption above the image.
title('Template Image Found in Original Image', 'FontSize', fontSize);

```

Hasil:



Analisa:

Pada percobaan tugas 2, mendeteksi orang menari dengan menggunakan metode pada tugas pertama tidak dapat dilakukan karena mengalami kegagalan dalam mendeteksi objek (template) dan banyak juga gambar lain selain penari yang terdeteksi sebagai penari sehingga metode pertama tidak dapat dilakukan, dan pada percobaan ini diterapkannya metode Normalized cross correlation dengan topi penari sebagai template, pada percobaan dapat dikenali nilai tertinggi dari kecocokan namun masih belum bisa mendeteksi ketiga penari dan dari analisa yang telah dilakukan bahwa untuk mendeteksi objek yang rumit diperlukan beberapa fitur untuk mendapatkan nilai keakuratan yang baik.