MATCHING TEMPLATE

clc; clear;

img = rgb2gray(imread('C:\Users\Swathika\Downloads\images (2).jpg'));

template = rgb2gray(imread('C:\Users\Swathika\Downloads\images (2).jpg'));

if any(size(template) > size(img)), template = imresize(template, min(size(img)./size(template))\*0.9); end

[h,w] = size(template);

methods = {'normxcorr2','ssd','sad','correlation','normalized\_correlation','normalized\_difference'};

figure('Name','Template Matching','NumberTitle','off');

for i = 1:6

method = methods{i};

if strcmp(method,'normxcorr2')

result = normxcorr2(template,img);

[score, idx] = max(result(:));

[ypeak, xpeak] = ind2sub(size(result), idx);

top\_left = [xpeak-w, ypeak-h];

else

result = zeros(size(img)-[h,w]+1);

T = double(template); Tn = (T - mean(T(:))) / std(T(:));

for y = 1:size(result,1)

for x = 1:size(result,2)

P = double(img(y:y+h-1, x:x+w-1));

result(y,x) = sum( ...

strcmp(method,'ssd').\*(P-T).^2 + ...

strcmp(method,'sad').\*abs(P-T) + ...

strcmp(method,'correlation').\*(P.\*T) + ...

strcmp(method,'normalized\_correlation').\*((P-mean(P(:)))./std(P(:))).\*Tn + ...

strcmp(method,'normalized\_difference').\*((P-T).^2)/sum(T(:).^2), ...

'all');

end

end

if ismember(method, {'ssd','sad','normalized\_difference'})

[score, idx] = min(result(:));

else

[score, idx] = max(result(:));

end

[y,x] = ind2sub(size(result), idx); top\_left = [x,y];

end

subplot(6,3,(i-1)\*3+1); imagesc(result); colormap gray; axis off;

subplot(6,3,(i-1)\*3+2); text(0.5,0.5,{method,sprintf('Score: %.4f',score)}, ...

'HorizontalAlignment','center','FontSize',12,'BackgroundColor','yellow'); axis off;

subplot(6,3,(i-1)\*3+3); imshow(img); hold on;

rectangle('Position',[top\_left(1), top\_left(2), w, h],'EdgeColor','r','LineWidth',2); hold off;

end

HARRIS CORNER

% Harris Corner Detection - Simplified Version

filename = "C:\Users\Swathika\Downloads\2i3TK603pRdrEiN7mnoSa6EurvY-mobile.jpg";

img = imread(filename);

figure, imshow(img), title('Original Image');

gray = double(rgb2gray(img));

cornerStrength = cornermetric(gray, 'Harris');

cornerStrength = imdilate(cornerStrength, strel('disk', 1));

threshold = 0.01 \* max(cornerStrength(:));

cornerMask = cornerStrength > threshold;

cornerImg = img;

cornerImg(repmat(cornerMask, [1,1,3])) = 0;

cornerImg(:,:,1) = cornerImg(:,:,1) + uint8(255 \* cornerMask);

figure, imshow(cornerImg), title('Harris Corner Detection');

BRUTE FORCE

img1 = imread('C:\Users\Swathika\Downloads\images (4).jpg');

img2 = imread('C:\Users\Swathika\Downloads\download (5).jpg');

if size(img1,3)==3, img1=rgb2gray(img1); end

if size(img2,3)==3, img2=rgb2gray(img2); end

disp(['MATLAB Version: ', version]);

[pts1, pts2] = deal(detectSIFTFeatures(img1), detectSIFTFeatures(img2));

[features1, validPts1] = extractFeatures(img1, pts1);

[features2, validPts2] = extractFeatures(img2, pts2);

indexPairs = matchFeatures(features1, features2, 'MatchThreshold', 100, 'MaxRatio', 0.7);

matchedPts1 = validPts1(indexPairs(:,1));

matchedPts2 = validPts2(indexPairs(:,2));

figure; showMatchedFeatures(img1, img2, matchedPts1, matchedPts2, 'montage');

title('SIFT Feature Matching using Lowe''s Ratio Test');