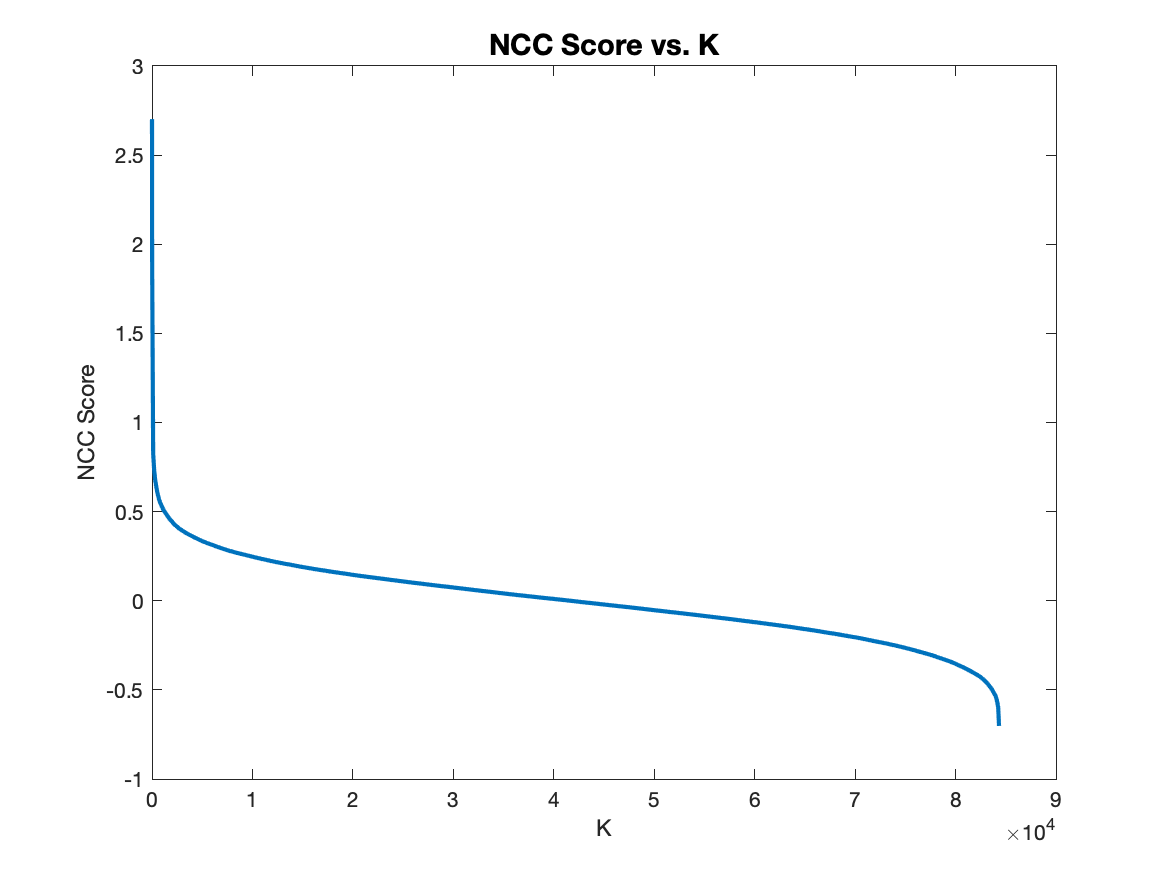
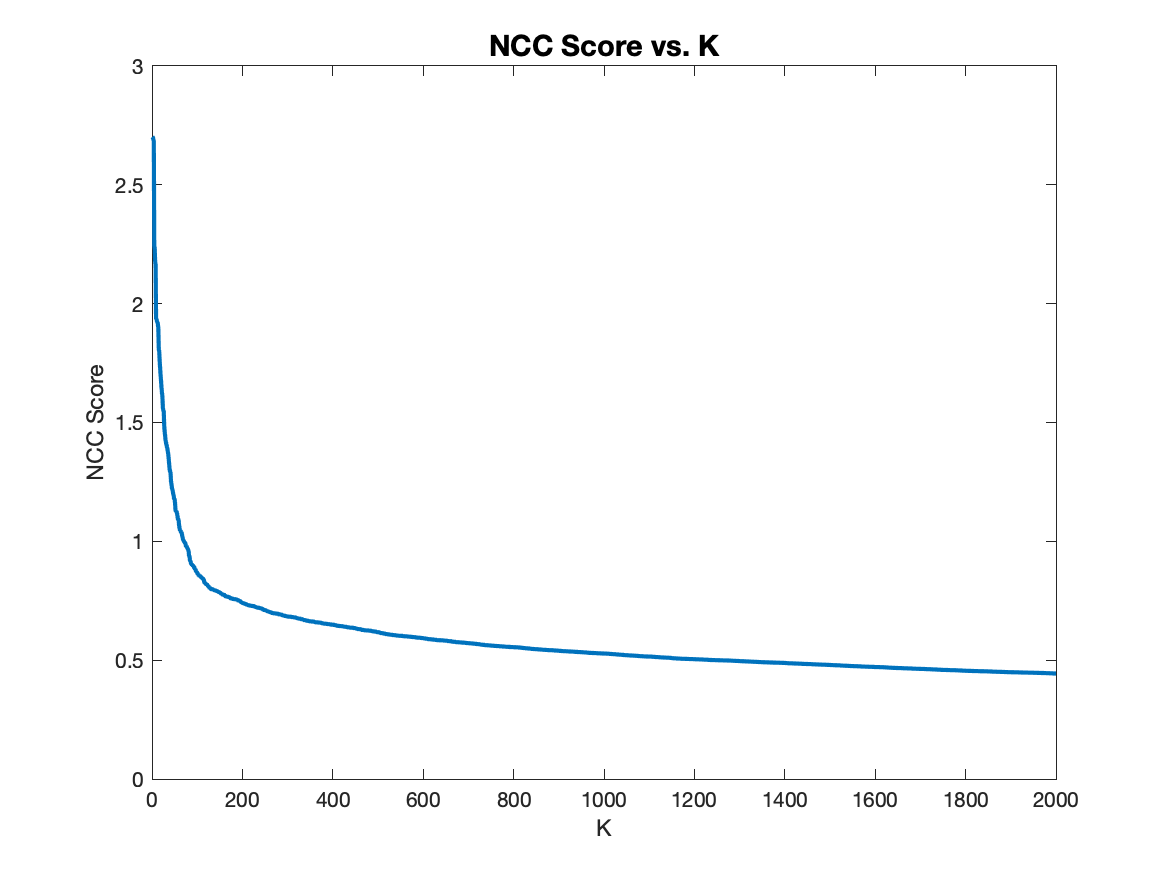
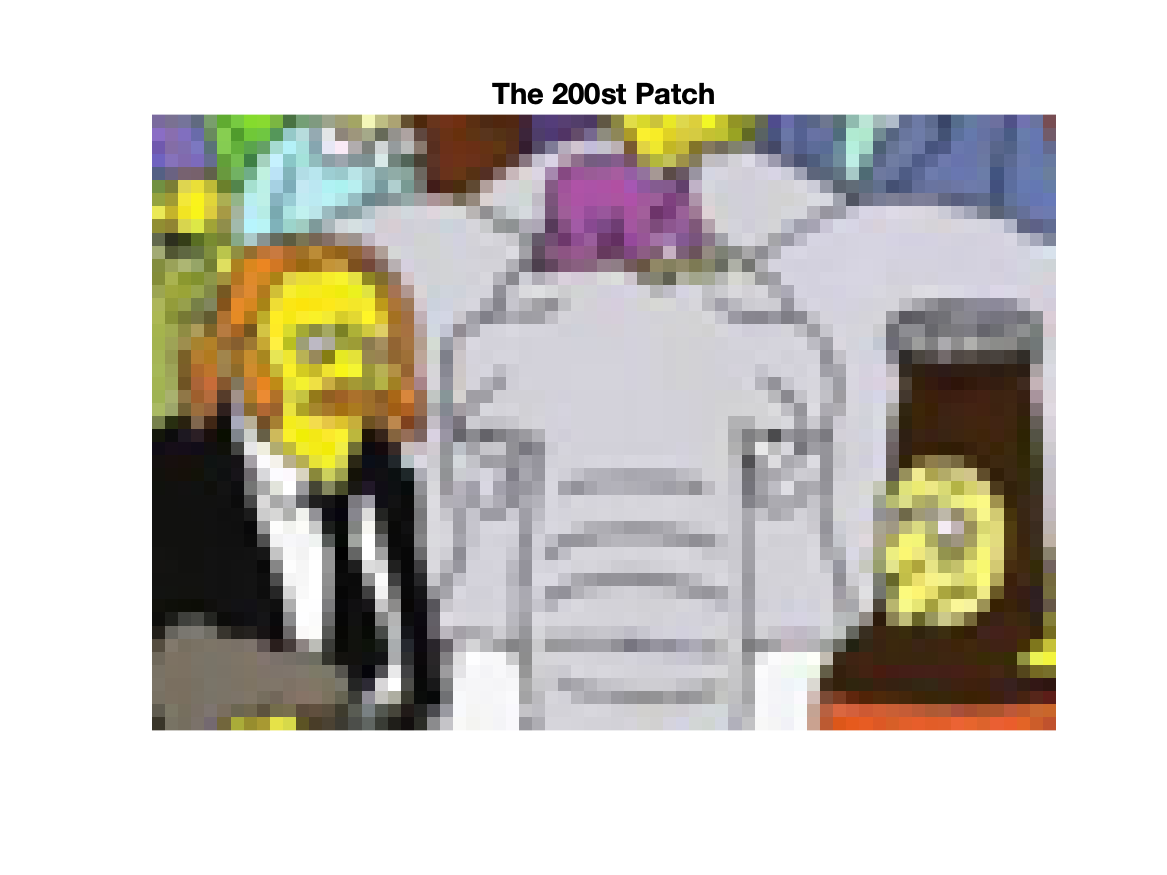
CSE 5524 HW7

Author: Yi Zhao

Professor: Jim Davis

1. The NCC list is calculated based on the image given on the slides. Here is the plot of NCC vs. K values (zoomed on the left), and the corresponding results are shown below. The center of the first patch is [106 106]. The result looks reasonable for k equals 1 to 10. The 50th, 100th, 200th result caught the elephant, but the center is a little bit far away from the origin in template. The 500th result failed to catch the correct spot. The plot indicates that the most accurate match occurs in K < 200. The value of NCC drops significantly from 1-150 (approximately), and afterwards the slope becomes smooth and flat for K > 400.





% Author: Yi Zhao

% CSE 5524, HW7

% 10/16/2019

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Problem 1

img = double(imread('./data/search.png'));

template = double(imread('./data/template.png'));

ncc = imageIteration(img, template);

for i = [1 2 5 10 50 100 200 500]

dispCropImg(img, ncc(i,2:3), [47 69])

title(sprintf('The %dst Patch',i),'FontSize',14)

saveas(gcf,sprintf('./output/patch\_%d.png',i))

pause;

end

disp(ncc(1,:))

displayPlot(reshape(ncc(:,1), 1, []), size(ncc, 1))

% displayPlot(reshape(ncc(:,1), 1, []), 2000)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Helper Function

% return the value of NCC of two eqo\_sized image.

function ncc = calculateNCC(origin, template)

p\_mean = mean(origin, [1 2]);

t\_mean = mean(template, [1 2]);

p\_sigma = std(origin, 0, [1 2]);

t\_sigma = std(template, 0, [1 2]);

arr = ((origin - p\_mean).\*(template - t\_mean))./(p\_sigma.\*t\_sigma);

ncc = sum(sum(arr, [1 2])/(size(template,1)\*size(template,2)-1), 'all');

end

% Calculate the sorted list of NCC given image and template

function nccList = imageIteration(img, template)

ncc = zeros(size(img) - size(template) + 1);

halfr = floor(size(template,1)/2);

halfc = floor(size(template,2)/2);

for r = halfr+1:size(img, 1)-halfr

for c = halfc+1:size(img, 2)-halfc

crop = img(r-halfr:r+halfr, c-halfc:c+halfc, :);

ncc(r-halfr,c-halfc) = calculateNCC(crop, template);

end

end

yind = double(repmat((halfr+1:size(img, 1)-halfr)', 1, size(ncc,2)));

xind = double(repmat(halfc+1:size(img, 2)-halfc,size(ncc,1),1));

disp(size(yind))

disp(size(xind))

disp(size(ncc))

nccList = cat(3,ncc,yind,xind);

nccList = reshape(nccList, [], 3);

nccList = sortrows(nccList, 1, 'descend');

end

% Display the cropped image given coordinate (r, c) and result size (r, c)

function dispCropImg(img, coordinate, size)

r = coordinate(1); % row of center pixel

c = coordinate(2); % col of center pixel

halfr = floor(size(1)/2);

halfc = floor(size(2)/2);

imshow(img(r-halfr:r+halfr, c-halfc:c+halfc, :)/255, 'InitialMagnification','fit');

end

% Plot the NCC score where arr = [y1 y2 y3 ...]

function displayPlot(arr, len)

x = 1:1:len;

y = arr(1:len);

plot(x,y,'Linewidth',2)

xlabel('K')

ylabel('NCC Score')

title('NCC Score vs. K','FontSize',14)

saveas(gcf,"./output/NCC\_Score.png");

end