

Programming Assignment 4

NikhilTank: 13110070

1. Capture 10 overlapping photographs of a static scene which is far from your camera. Stitch them to make a panorama/ mosaic of the scene to depict extended field of view in a single photograph.



Images: Sardar Sarovar Dam's Reservoir

Note: SIFT & SURF operators are removed from OpenCV 3.x, due to [patent issues](#). So to complete the assignment MatLab was used as the programming platform.

```

%Matlab Code for stitching 10 images in a panorama
clc
clear

%% Load Image
imageLocation=fullfile('C:', 'Users', 'NikhilT', 'Desktop', 'assignment4', 'SSD', {'(00).jpg', '(01).jpg', '(02).jpg', '(03).jpg', '(04).jpg', '(05).jpg', '(06).jpg', '(07).jpg', '(08).jpg', '(09).jpg'});
imgSet = imageSet(imageLocation);
I = read(imgSet, 1);

%% Initiate
grayimg = rgb2gray(I);
points = detectSURFFeatures(grayimg);
[features, points] = extractFeatures(grayimg, points);
%2D affine transformation of image count size
tforms(imgSet.Count) = projective2d(eye(3));

%we want some way to find corelation between 2 set of images and same of
%all 10 image
for n= 2:imgSet.Count

    %store the I(n-1) previous image features and points data to new variable
    featuresPrev=features;
    pointsPrev=points;
    %read new image in I(n) variable (over-write)
    I = read(imgSet, n);
    grayimg = rgb2gray(I);
    points = detectSURFFeatures(grayimg);
    [features, points] = extractFeatures(grayimg, points);

    %correspondences between I(n) and I(n-1).
    indexPairs = matchFeatures(features, featuresPrev, 'Unique', true);
    matchedPoints = points(indexPairs(:,1), :);
    matchedPointsPrev = pointsPrev(indexPairs(:,2), :);

    %finding geometric-transformation for 2 image pair
    tforms(n) = estimateGeometricTransform(matchedPoints, matchedPointsPrev,...
        'projective', 'Confidence', 99.9, 'MaxNumTrials', 1500);
    %recurvively finding the tranformation operation for given set of images
    %we need to multiply the tforms of prev and new (as learned in Comp Graphic
    course)
    tforms(n).T = tforms(n-1).T * tforms(n).T;
    c=n
end

imageSize = size(I); % all the images are the same size

% Compute the output limits for each transform
for i = 1:numel(tforms)
    [xlim(i,:), ylim(i,:)] = outputLimits(tforms(i), [1 imageSize(2)], [1
    imageSize(1)]);
end

%averaging x limit and y limit
avgXLim = mean(xlim, 2);
[~, idx] = sort(avgXLim);

```

```

centerIdx = floor((numel(tforms)+1)/2);
centerImageIdx = idx(centerIdx);

Tinv = invert(tforms(centerImageIdx));
for i = 1:numel(tforms)
    c=i+13
    tforms(i).T = Tinv.T * tforms(i).T;
end

%% Initiating the panorama
for i = 1:numel(tforms)
    c=i+25
    [xlim(i,:), ylim(i,:)] = outputLimits(tforms(i), [1 imageSize(2)], [1
imageSize(1)]);
end

% minimum and maximum output limits
xMin = min([1; xlim(:)]);
xMax = max([imageSize(2); xlim(:)]);

yMin = min([1; ylim(:)]);
yMax = max([imageSize(1); ylim(:)]);

% Width and height of panorama taken using min max values of the dimensions
wid = round(xMax - xMin);
ht = round(yMax - yMin);

% Initialize the "empty" panorama.
Panorama = zeros([ht wid 3], 'like', I);

%% Render the panorama data
render = vision.AlphaBlender('Operation', 'Binary mask', ...
    'MaskSource', 'Input port');

% Create a 2-D spatial reference object defining the size of the panorama.
xLim = [xMin xMax];
yLim = [yMin yMax];
panoramaView = imref2d([ht wid], xLim, yLim);

% Create the panorama.
for i = 1:imgSet.Count
    c=i+38
    I = read(imgSet, i);

    % Transform I into the panorama.
    warpedImage = imwarp(I, tforms(i), 'OutputView', panoramaView);

    % Overlay the warpedImage onto the panorama.
    Panorama = step(render, Panorama, warpedImage, warpedImage(:,:,1));
end
%% image
imwrite(Panorama, 'P.jpg');
% refered code:http://www.mathworks.com/examples/matlab-computer-vision/725-feature-based-panoramic-image-stitching

```



Figure: P1.JPG (taking first 5 images)



Figure: P1.JPG (taking last 5 images)



Figure: P.JPG (taking all 10 images)



Figure: P(cropped).JPG (Manually cropped)

Reference:

http://tobw.net/index.php?cat_id=2&project=Panorama+Stitching+Demo+in+Matlab

<http://in.mathworks.com/help/vision/examples/feature-based-panoramic-image-stitching.html>

<http://www.mathworks.com/examples/matlab-computer-vision/725-feature-based-panoramic-image-stitching>