HACETTEPE UNIVERSITY FALL 21/22

BBM 413

ASSIGNMENT III

Name: Abdullah Atahan

Surname: Türk

Student ID: 21827943

Topics:

Genarating panorama images with feature matching and image blending

T.A: Yunus Can Bilge

Explain the each step of the algorithm. Why can SIFT features be used for? What is the purpose of the RANSAC algorithm? Why are they used for?

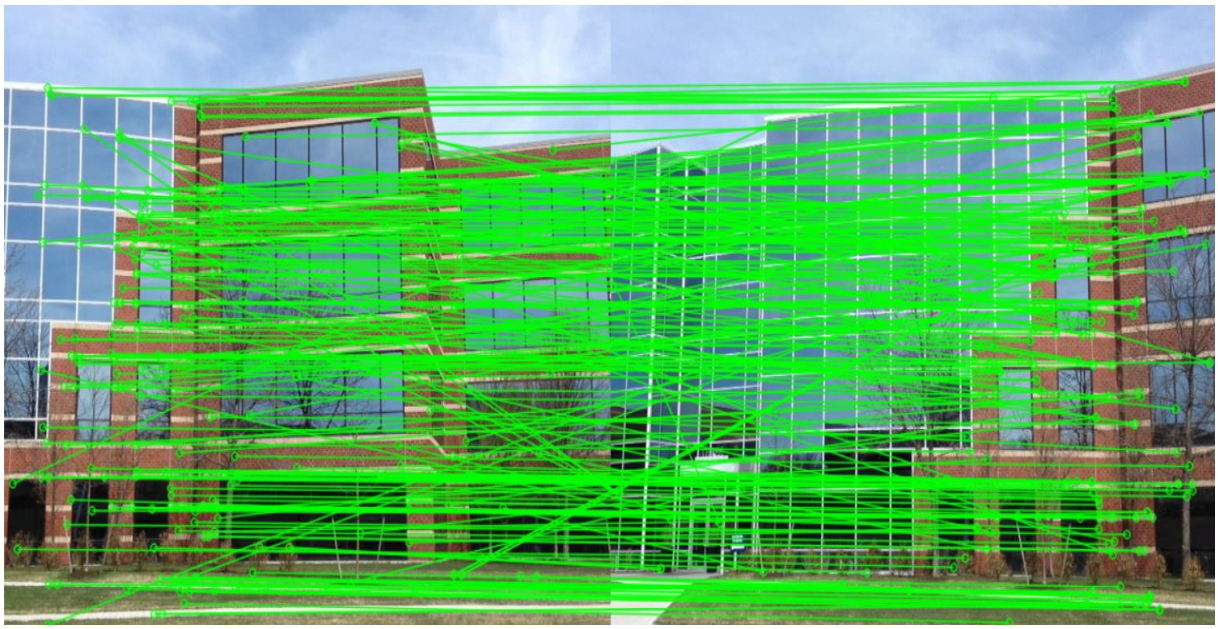
When we try to stitch images, our solution has to be robust even if the images have differences one or more aspects like scaling, angle, spatial position etc.For stitching the images, we need to imply some main steps like keypoint and descripton detection, finding local invariant descriptors with using SIFT, feature matching and finding the homography.

The first step that we would do should be extract key points and features of interests. When detecting keypoints, using a method which is invariant to rotation and scaling is very important. There are many methods we can use like SIFT, SURF, ORB etc. but I prefered to use SIFT. SIFT uses Difference of Gaussians (DoD). It apply DoD on differently scaled versions of the same image. It also uses the neighboring pixel information to find and refine key points and corresponding descriptors.

With using SIFT_create() and detectAndCompute(), we can detect the keypoints and descriptors. But the important thing is, before doing that, we need to convert the images to grayscale.

After found the key points, we need to do feature matching. Because some key points might be wrong or misleading. we should compare the sets of features and stick with the pairs that show more similarity. Fort his part, we can use BFMatcher() and knnMatch() methods from cv2. BF Matcher computes the Euclidean distance between two points. On the other hand, knn Matcher Instead of returning the single best match for a given feature, it returns the k

best matches, I defined the k as 2. I also did a ratio test with using a for loop. if the distance between two key points is within a certain ratio, I kept it, otherwise, I threw them away. I choose the ratio values as 0.8 because when I choose it as 0.8 I get the best results.



Now, we need to find the homography. To do that we can use more than one way but some methods like linear regression but it is too sensitive to outliers. So it might fail if the number of the outliers are too much. On the other hand RANSAC only fits the model on the subset of points identified as the inliers. That's why I used RANSAC. It solves this problem by estimating parameters only using a subset of inliers in the data. In the code I used findHomography() and cv2.RANSAC method.

After doing all those step we need to warp one of the images to a common plane. We should apply a perspective transformation to one of the images. The perspective transform may combine one or more operations like rotation, scale, translation, or shear. The idea is to transform one of the images so that both images merge as one (First I stitched second and third images and then stitch the first image with them). I used warpPerspective() method for this. It takes an image and the homography as input. Then, it warps the source image to the destination based on the homography.

Can you use different methods to merge images? What are the advantages of the blending compared to other methods?

Of course there are more than one methods to merge images. Image blending involves executing the adjustments figured out in the calibration stage, combined with remapping of the images to an output projection. Colors are adjusted between images to compensate for exposure differences. Images are blended together and seam line adjustment is done to minimize the visibility of seams between images.

I found some algorithms other than the image blending for merging images. For example "Merging in spatial domain" is another blending algorithm. Its algorithm described as:

"Set the value for number of images/layers 'n' Align the images or layers so that number of samples is same. Normalize the image values i(x,y) to 0-1 While all pixels(x,y) in the image are not seen do ADD corresponding intensities of all n images."

There is another algorithm named "Merging in frequency domain" for merging:

"Set the value for number of images/layers 'n' Align the images or layers so that number of samples is same. Normalize the image values i(x,y) to 0-1 Identify the highest frequency (Max.Frequency) from frequency spectrum of the input image. Set threshold T=Max.Frequency * x Where 0<x<1 While all pixels(x,y) in the images are not seen do ADD corresponding density of frequencies of all n images."

Image Blending combines images very softly, this is one the advantages of the image blending. If we compare the results of image blending and the other 2 algorithms that I found on the net, we can see that image blending has better results. Also image blending is a simple and fast algorithm.

How can you improve the results?

We can use methods more robust and faster than SIFT like SURF or FAST(Features from accelerated segment test). Altough SIFT is very common, as I see SURF and FAST are really robust and fast detectors. Especially FAST is very robust and fast. Here is some differences and advantages of SURF:

" SURF approximates the DoG with box filters. Instead of Gaussian averaging the image, squares are used for approximation since the convolution with square is much faster if the integral image is used. Also this can be done in parallel for different scales. The SURF uses a BLOB detector which is based on the Hessian matrix to find the points of interest. For orientation assignment, it uses wavelet responses in both horizontal and vertical directions by applying adequate Gaussian weights. For feature description also SURF uses the wavelet responses. A neighborhood around the key point is selected and divided into subregions and then for each subregion the wavelet responses are taken and represented to get SURF feature descriptor. The sign of Laplacian which is already computed in the detection is used

for underlying interest points. The sign of the Laplacian distinguishes bright blobs on dark backgrounds from the reverse case. In case of matching the features are compared only if they have same type of contrast (based on sign) which allows faster matching .”

Also we can use more complex image blending methods but I couldn’t find many info about them. But obviously there are much more better methods. Programs like Adobe Photoshop uses very complex and advanced methods for stitching and image blending.

Change the parameter for the image pyramid level and comment about the effects of it.

Due to my researches and observations, increasing the pyramid level increases the brightness of the image. Because of we increase the levels, we do the process on a smaller pyramid first, so this would give us a better image.

Change the parameter for bluring parameter and comment about the effects of it.

We need to be careful when we set the blurring parameter. For example if you increase blurring you can get much better and smoother images because the pyramids gets more blurred and the feature points becomes more prominent next to its neighbors. But if you increase it too much, the result would be bad because you can because if blurring is too much, we may have difficulty detecting keypoints.

RESOURCES:

1. https://towardsdatascience.com/image-panorama-stitching-with-opencv-2402bde6b46c
2. https://stackoverflow.com/questions/11172408/surf-vs-sift-is-surf-really-faster
3. https://medium.com/@shehan.a.perera/a-comparison-of-sift-surf-and-orb-333d64bcaaea
4. https://www.pyimagesearch.com/2018/12/17/image-stitching-with-opencv-and-python/
5. https://www.pyimagesearch.com/2016/01/11/opencv-panorama-stitching/