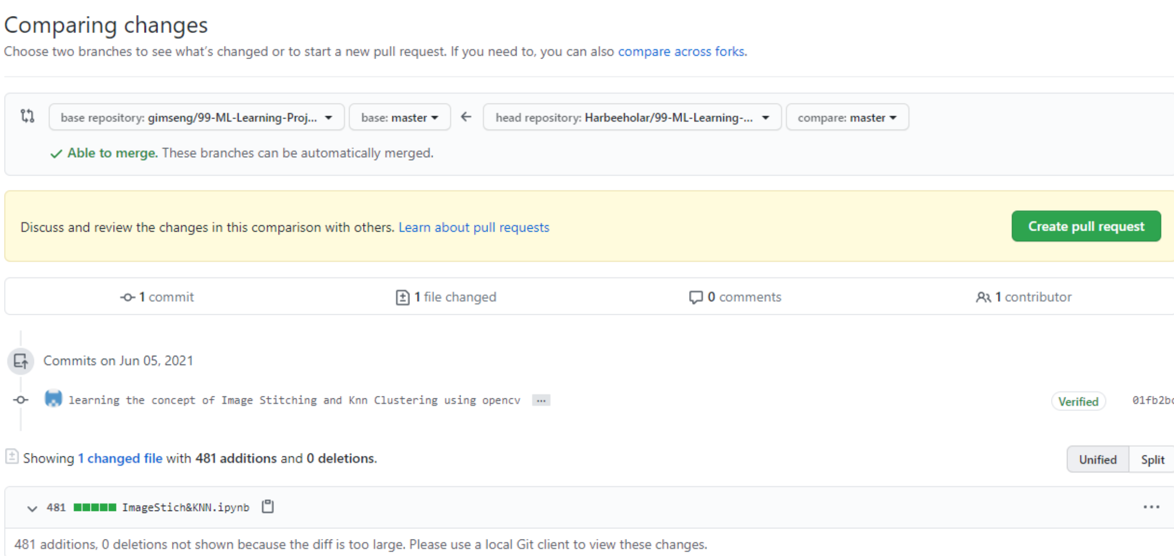
NAME: Abiola Godo

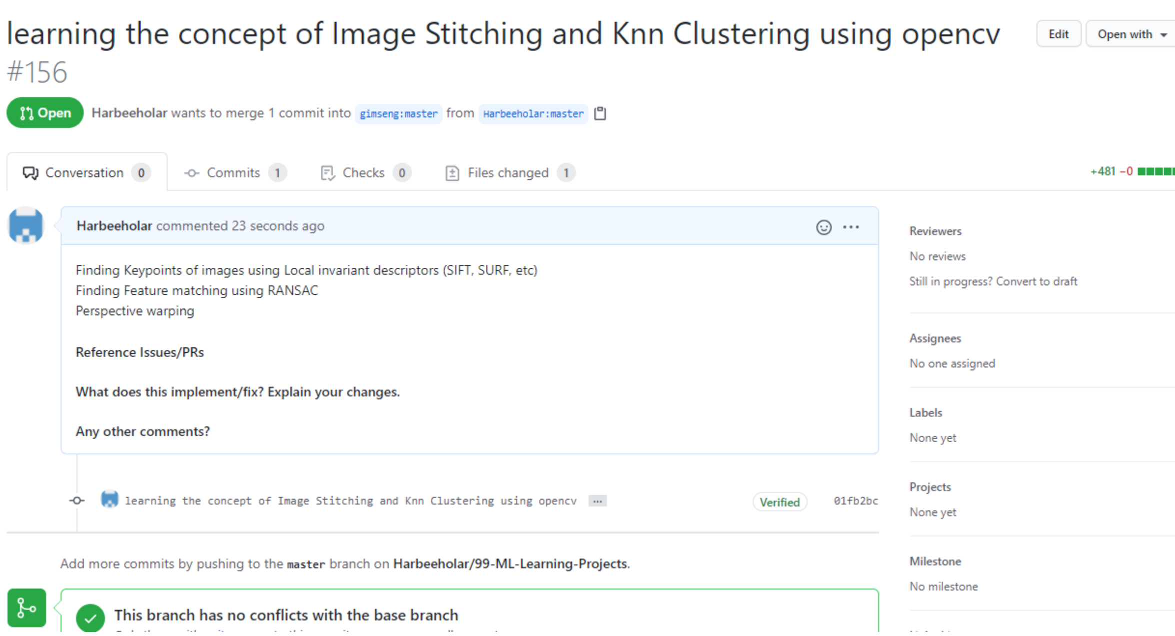
CLASS: CPSC 362 Assignment 2

A write up describing the process for making changes in this project: development setup, pre-checkin checks, code reviews (by who), description testing (automated testing?), deployment.

<u>The Link to the pull request/commit</u>

https://github.com/gimseng/99-ML-Learning-Projects/pull/156





Development Setup

Google Colab/ Jupyter Notebook

Pre-checking checks

Working Repository is Machine learning projects. The new project added is related to image processing.

Other prechecking project checks are

General-Purpose Machine Learning

Computer Vision

Natural Language Processing

Bayesian

Misc/Mix Models

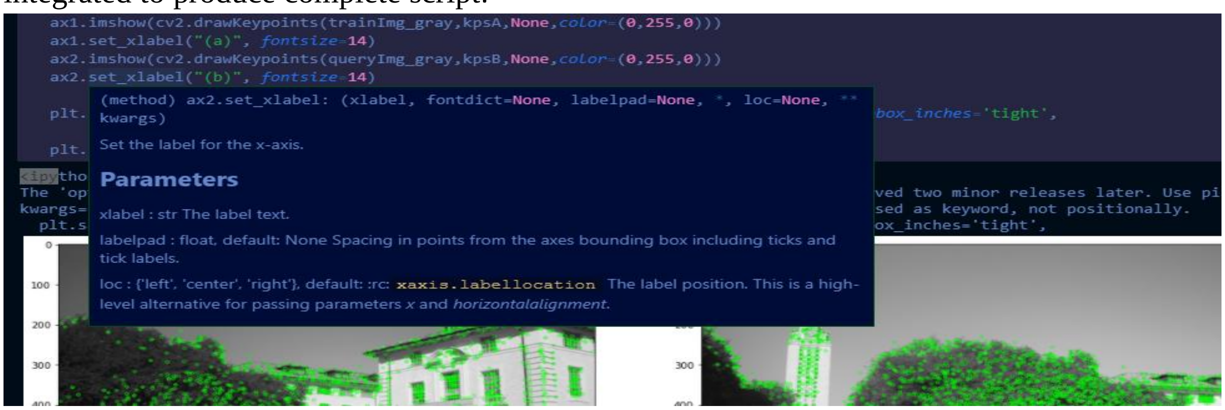
Project lies with Computer Vision

Code Review

Pending by reviewers

Automated testing

Testing is performed by unit tests. Python script is created using small modules which is integrated to produce complete script.



```
    elif feature_matching == 'knn':
        matches = matchKeyPointsKNN(featuresA, featuresB, ratio=0.75, method=feature_extractor)
        img3 = cv2.drawMatches(trainImg,kpsA,queryImg,kpsB,np.random.choice(matches,100),
                               None,flags=cv2.DrawMatchesFlags_NOT_DRAW_SINGLE_POINTS)


    plt.imshow(img3)
    plt.axis('off')
    plt.savefig("./output/" + feature_matching + "_matching_img_"+str(imageId)+'.jpeg', bbox_inches='tight',
                dpi=300, optimize=True, format='jpeg')
    plt.show()
```

```
Using: bf feature matcher
Raw matches (Brute force): 1826
```

```
            return None
```

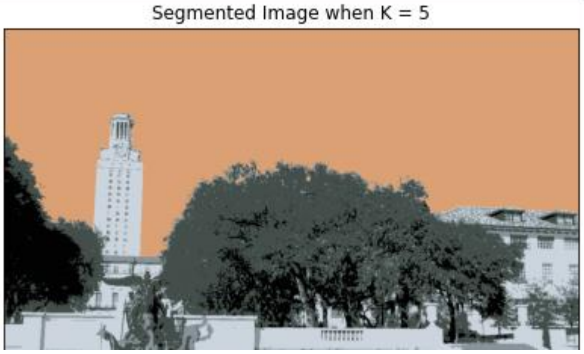
```
M = getHomography(kpsA, kpsB, featuresA, featuresB, matches, reprojThresh=4)
if M is None:
    print("Error!")
(matches, H, status) = M
print(H)
```

```
[[ 7.64643420e-01  3.61878762e-02  4.46938787e+02]
 [-1.35482276e-01  9.10796305e-01  7.63090370e+01]
 [-2.11444932e-04 -3.33670861e-05  1.00000000e+00]]
```

```
ret,label,center=cv2.kmeans(vectorized,K,None,criteria,attempts,cv2.KMEANS_PP_CENTERS)
center = np.uint8(center)
res = center[label.flatten()]
result_image = res.reshape((img.shape))
figure_size = 15
plt.figure(figsize=(figure_size,figure_size))

plt.subplot(1,2,1),plt.imshow(result_image)
plt.title('Segmented Image when K = %i' % K), plt.xticks([]), plt.yticks([])
plt.show()
```

Segmented Image when K = 5



A write up describing the project at a high level, what is it, what does it do? What impact does your change have?

Impact: Stitched images are used to produce images that have Large field of views with more information capacity. This helps in many important applications like ranging from Autonomous Driving with ability to handle complex occlusions. Furthermore, Image clustering can help to find objects in field of view.

Project

1. Finding Keypoints of images using Local invariant descriptors (SIFT, SURF, etc)
2. Finding Feature matching using RANSAC
3. Perspective warping
4. KNN clustering

A write up describing the use-cases and requirements covered by your commit using at least two different requirements models covered in class

Two requirements project works image stitching algo and KNN cluttering algo

Use case 1: ANALYSIS PATTERNS

Pattern Name	Image Stitched Algorithm
Intent	Creating a two-image stitched with feature descriptors
Motivation	• Finding Key points of images using local invariant descriptors (SIFT, SURF, etc) • Finding Feature matching using RANSAC • Perspective warping
Primary actor	Colab/jupyter notebook and Developer

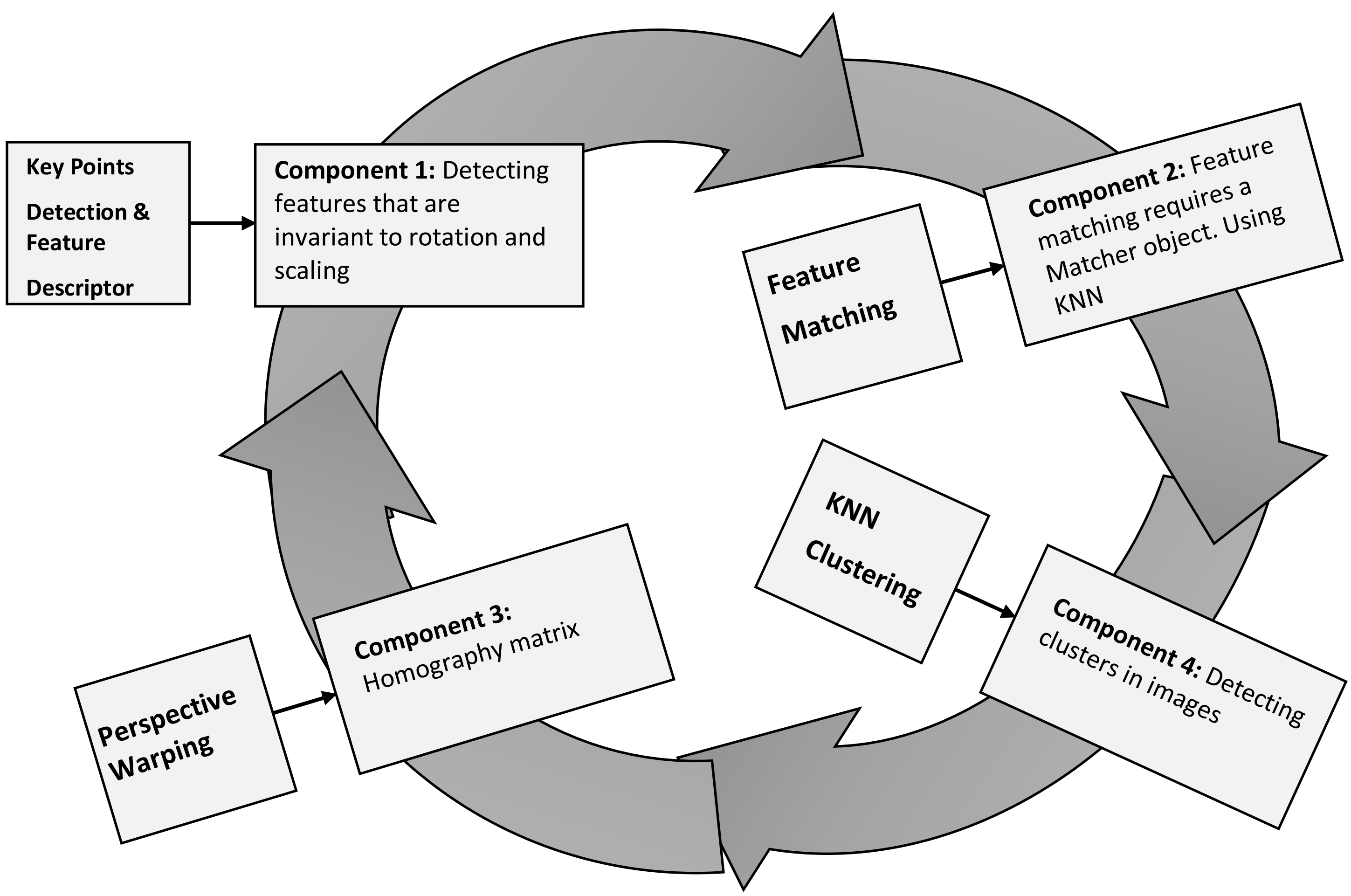
Use case 2:

Pattern	Image Stitch Algorithm
Intent	Creating a KNN clustering

Motivation	• Vectorized image • Defining number of clustering • Using OpenCV KNN function • Subplot image
Primary Actor	Developer and Colab/jupyter notebook

Prototyping Model

We have designed our algorithm in form of four increments which eventually form our project.



Generic Process Model