ASSIGNMENT 2 REPORT

ASSIGNMENT 2(A)

Q1. You are given a scene image containing logo, and a gallery containing reference logos for 10 business brands. Find out which business brand is present in the scene. Try out three different approaches and compare them.

(a) Submit your implementation. File name of your python file should be logoMatch.py and it should take scene image, logo image and approach name as input and either show the matched region or say not enough match points found.

(b) Show your results qualitatively in the report and write down your observation.

Ans:

a)implemented in google colab

b)Algorithm:-

1. First approach:-SIFT
2. Second approach:- ORB
3. Third approach:-Template Matching
4. Additional approach :- EaseOcr

1. SIFT (SCALE INVARIANT FEATURE TRANSFORMATION) DETECTOR BASED MATCHING.

   SIFT (Scale-Invariant Feature Transform) is a popular feature detection and matching algorithm used for image recognition tasks. Here are the steps for using SIFT-based matching:

   Detect keypoints and extract descriptors from the logo image and the scene image using the SIFT algorithm.
   Match the descriptors from both images using a matcher algorithm, such as FLANN (Fast Library for Approximate Nearest Neighbors) matcher.
   Apply a ratio test on the matches to eliminate false positives.
   Count the number of good matches and if the count is above a certain threshold, declare a match.

2. ORB (ORIENTED FAST AND ROTATED BRIEF) METHOD BASED MATCHING.

   In this approach, we can use feature-based matching algorithms Oriented FAST and Rotated BRIEF (ORB) to detect and match keypoints between the scene image and reference logos. We can then use the number of matched keypoints to determine the similarity between the two images. The brand logo with the highest number of matched keypoints can be considered the most likely match.

3. TEMPLATE MATCHING

   In this approach, we can use correlation-based template matching methods such as normalized cross-correlation (NCC) to match a template of the reference logo with different regions of the scene image. The region with the highest correlation score can be considered the most likely match.

4. EASE OCR METHOD
   In this method ,when there will be text involved in the logo of the Main image ,then only text will be extracted.and in that case this method will work best .

RESULTS AND COMPARISION:

1. Result of SIFT DETECTOR.
2. Comparing Levis shop logo with every logo brands.
   /content/sample_data/Example1/logos/spar.jpg
   Points matched are 1
   /content/sample_data/Example1/logos/tacobell.jpg
   Points matched are 1
   /content/sample_data/Example1/logos/umbro.jpg
   Points matched are 0
   /content/sample_data/Example1/logos/hp.jpg
   Points matched are 9
   /content/sample_data/Example1/logos/kfc.jpg
   Points matched are 0
   /content/sample_data/Example1/logos/tommyhilfiger.jpg
   Points matched are 0
   /content/sample_data/Example1/logos/lg.jpg
   Points matched are 1
   /content/sample_data/Example1/logos/shell.jpg
   Points matched are 2
   /content/sample_data/Example1/logos/levis.jpg
   Points matched are 21
   /content/sample_data/Example1/logos/nescafe.jpg
   Points matched are 0

Matched Result is LEVIS:

Here we have seen from the above result is that 21 points are matched which is maximum in the case of levis logo. Hence , correct result gets predicted

Comparing Starbucks shop logo with every logo brands. .
/content/sample_data/Example2/logos/motorola.jpg
Points matched are 3
/content/sample_data/Example2/logos/honda.jpg
Points matched are 0
/content/sample_data/Example2/logos/starbucks.jpg
Points matched are 35
/content/sample_data/Example2/logos/pepsi.jpg
Points matched are 1
/content/sample_data/Example2/logos/warnerbros.jpg
Points matched are 0
/content/sample_data/Example2/logos/toyota.jpg
Points matched are 4
/content/sample_data/Example2/logos/puma.jpg
Points matched are 1
/content/sample_data/Example2/logos/hp.jpg
Points matched are 7
/content/sample_data/Example2/logos/rolex.jpg
Points matched are 0
/content/sample_data/Example2/logos/lg.jpg
Points matched are 2
Matched result is STARBUCKS

Here we have seen from the above result is that 35 points are matched which is maximum in the case of Starbucks logo. Hence , correct result gets predicted

2. Result of ORB Detector:
Comparing Levis shop logo with every logo brands.:
/content/sample_data/Example1/logos/spar.jpg
Points matched are 13
/content/sample_data/Example1/logos/tacobell.jpg
Points matched are 22
/content/sample_data/Example1/logos/umbro.jpg
Points matched are 16
/content/sample_data/Example1/logos/hp.jpg
Points matched are 3
/content/sample_data/Example1/logos/kfc.jpg
Points matched are 34
/content/sample_data/Example1/logos/tommyhilfiger.jpg
Points matched are 22
/content/sample_data/Example1/logos/lg.jpg
Points matched are 8
/content/sample_data/Example1/logos/shell.jpg
Points matched are 33
/content/sample_data/Example1/logos/levis.jpg
Points matched are 15
/content/sample_data/Example1/logos/nescafe.jpg
Points matched are 36

Here,
Maximum points are 36 for the logo NESCAFE

Here we have seen from the above result is that 36 points are matched which is maximum in the case of Nescafe logo. Hence , incorrect result gets predicted.
Or we can say ‘there was not enough points available’

Comparing Starbucks shop logo with every logo brands.

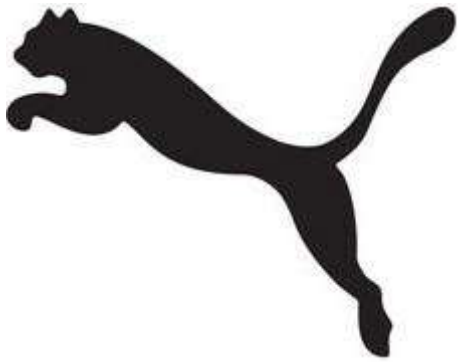
/content/sample_data/Example2/logos/motorola.jpg
Points matched are 6
/content/sample_data/Example2/logos/honda.jpg
Points matched are 3
/content/sample_data/Example2/logos/starbucks.jpg
Points matched are 34
/content/sample_data/Example2/logos/pepsi.jpg
Points matched are 7
/content/sample_data/Example2/logos/warnerbros.jpg
Points matched are 9
/content/sample_data/Example2/logos/toyota.jpg
Points matched are 5
/content/sample_data/Example2/logos/puma.jpg
Points matched are 3
/content/sample_data/Example2/logos/hp.jpg
Points matched are 3
/content/sample_data/Example2/logos/rolex.jpg
Points matched are 3
/content/sample_data/Example2/logos/lg.jpg
Points matched are 2

Here,
Maximum points are 34 for the logo LG

3 .Results of Template Matching
Example 1:
/content/sample_data/Example1/logos/spar.jpg
2551
/content/sample_data/Example1/logos/tacobell.jpg
4041
/content/sample_data/Example1/logos/umbro.jpg
0
/content/sample_data/Example1/logos/hp.jpg
0
/content/sample_data/Example1/logos/kfc.jpg
367
/content/sample_data/Example1/logos/tommyhilfiger.jpg
2
/content/sample_data/Example1/logos/lg.jpg
157
/content/sample_data/Example1/logos/shell.jpg
0
/content/sample_data/Example1/logos/levis.jpg
57
/content/sample_data/Example1/logos/nescafe.jpg
2060
Here we see that Max matched point is of Tacobells :4041

Example :2
/content/sample_data/Example2/logos/motorola.jpg
718
/content/sample_data/Example2/logos/honda.jpg
0
/content/sample_data/Example2/logos/starbucks.jpg
2
/content/sample_data/Example2/logos/pepsi.jpg
0
/content/sample_data/Example2/logos/warnerbros.jpg
0
/content/sample_data/Example2/logos/toyota.jpg

0
/content/sample_data/Example2/logos/puma.jpg
18034
/content/sample_data/Example2/logos/hp.jpg
0
/content/sample_data/Example2/logos/rolex.jpg
8397
/content/sample_data/Example2/logos/lg.jpg
6665



4. AdditionalMethod EaseOCR Result
[([[153, 235], [473, 235], [473, 360], [153, 360]], 'Levis', 0.9965880896935464), ([[562, 372], [604, 372], [604, 398], [562, 398]], 'O501', 0.4031730592250824), ([[1237, 559], [1280, 559], [1280, 615], [1237, 615]], 'HU', 0.12167566641780296)]
[([[21, 17], [205, 17], [205, 79], [21, 79]], 'Levis', 0.7854220372776649)]
3
Levis
LOGO HAS BEEN FOUND MATCHING TO THE SCENE WITH BRAND NAMED AS:Levis

COMPARISONS:
1.SIFT DETECTOR:Works Best in both the examples
Pros:
Feature-based matching algorithms are robust to changes in image scale, orientation, and lighting conditions.
SIFT and ORB are widely used in computer vision and have well-established implementations in popular libraries such as OpenCV.
Cons:
Feature-based matching can be sensitive to changes in the scene such as occlusions or perspective distortions.
The number of matched keypoints may not always accurately reflect the similarity between two images.
2.ORB DETECTOR:Works good in 2nd Example but incorrect in 1st example
3.TEMPLATE MATCHING:Works worst in both the examples.
Pros:
Template matching is a simple and intuitive approach that requires little to no parameter tuning.

Correlation-based methods are computationally efficient and can be implemented using simple matrix operations.
Cons:
Template matching is sensitive to changes in image scale, orientation, and lighting conditions.
Template matching may fail if there are other objects or patterns in the scene that are similar to the reference logo.
4.EaseOCR: good with LEVIS logo as it contains text and not in starbucks as it has not text in its logo.
Colab link:
https://drive.google.com/file/d/105E-nhYnWHrF65NK8UQSfXLLNcmTCRJa/view?usp=sharing

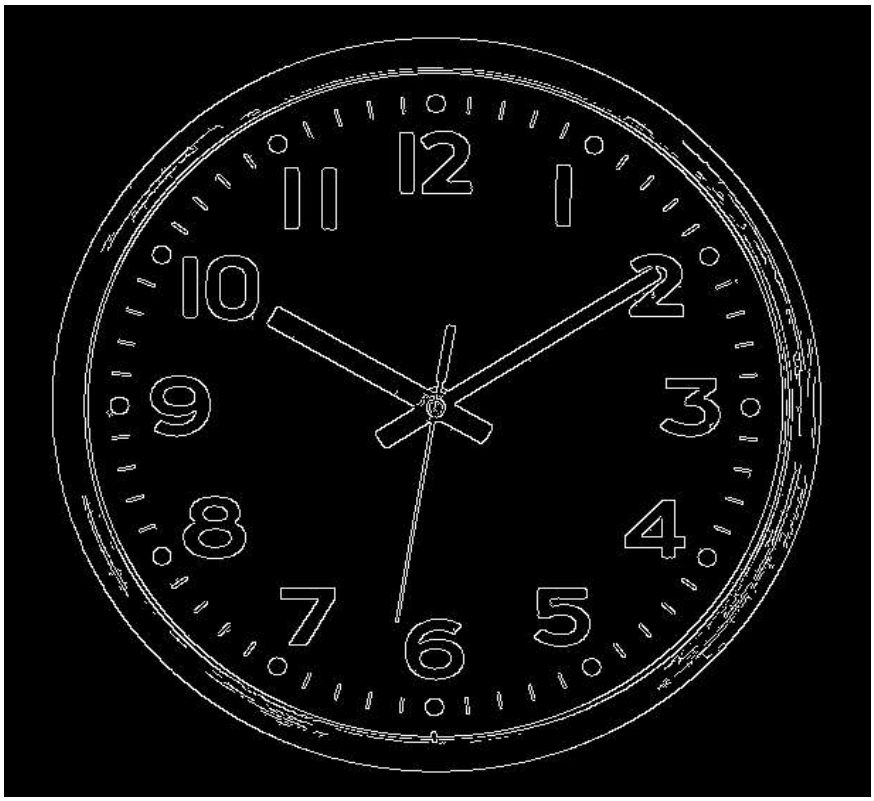
Q2.Implement Hough Transform for line detection from scratch. Compare the result of openCV implementation vs your implementation (both speed and performance wise) on a picture of your choice. (a) Submit your implementation. File name of your python file should be HoughTrans.py. (b) Show your results qualitatively in the report and write down your observations.

Ans:
a)

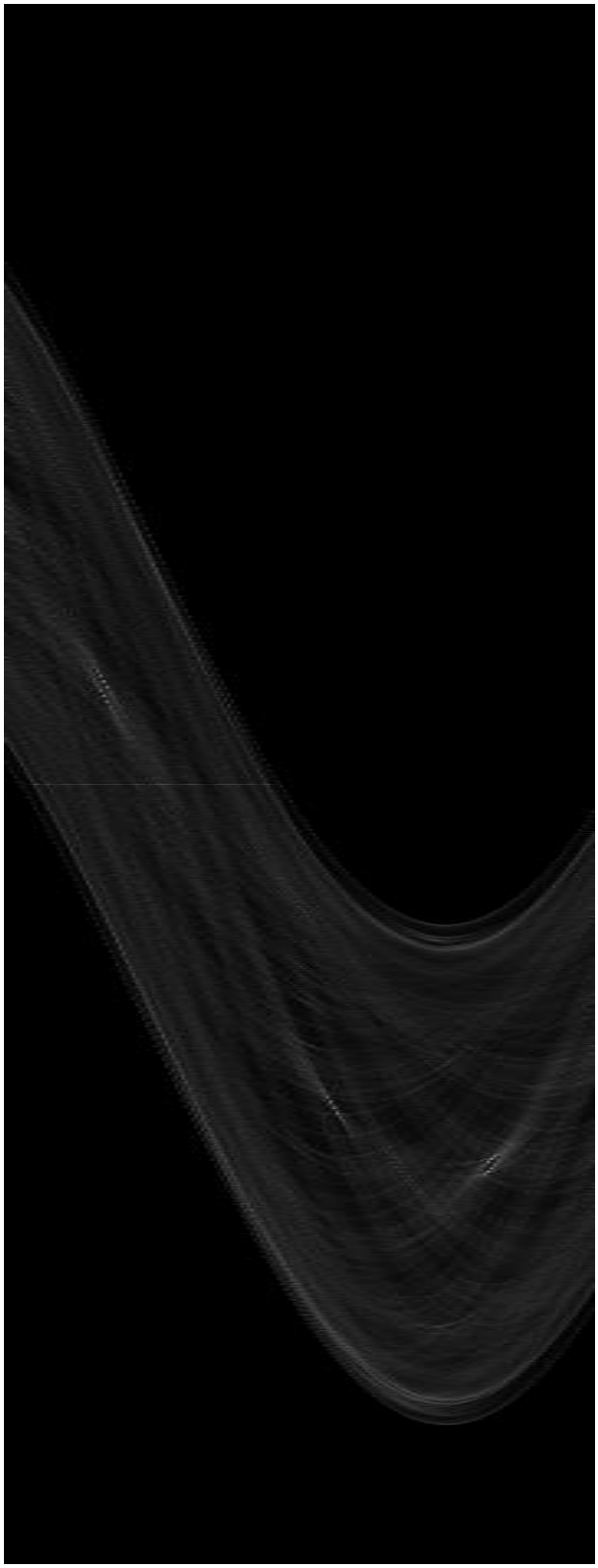
1. Load the image and find the gray image of it
2. Find the Canny edge of the image using canny edge detector of the opencv library.
3. Next define the function HoughTrans which takes input as edges.
4. Inside it define the parameters rho ,theta and accumulator in it and return it .
5. Next call that function and store all the parameters in the list.
6. After that use idx and angle in the loop from the range theta.
7. Define the range of theta and max and min value of rhos. Using Hough trans Peak function.
8. And convert the values back to cartesian coordinate values using rho and theta.
9. Now Plot the lines in the figure such that all the slope of all the points lies in the common line of the image.
10. Next Call the Real HoughTransform function of opencv library
11. Plot the lines in the same image
12. Compared the result of both image.

b)
Results
Inputted Image:

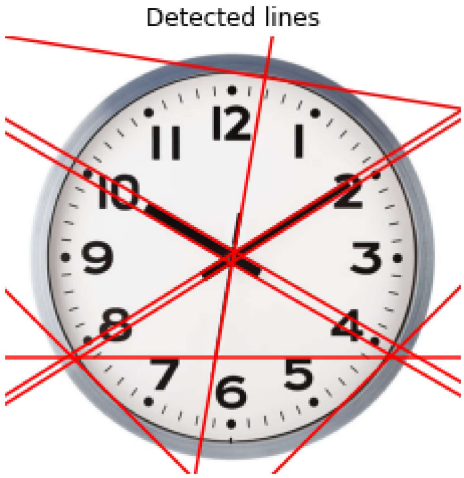
Edged Image:



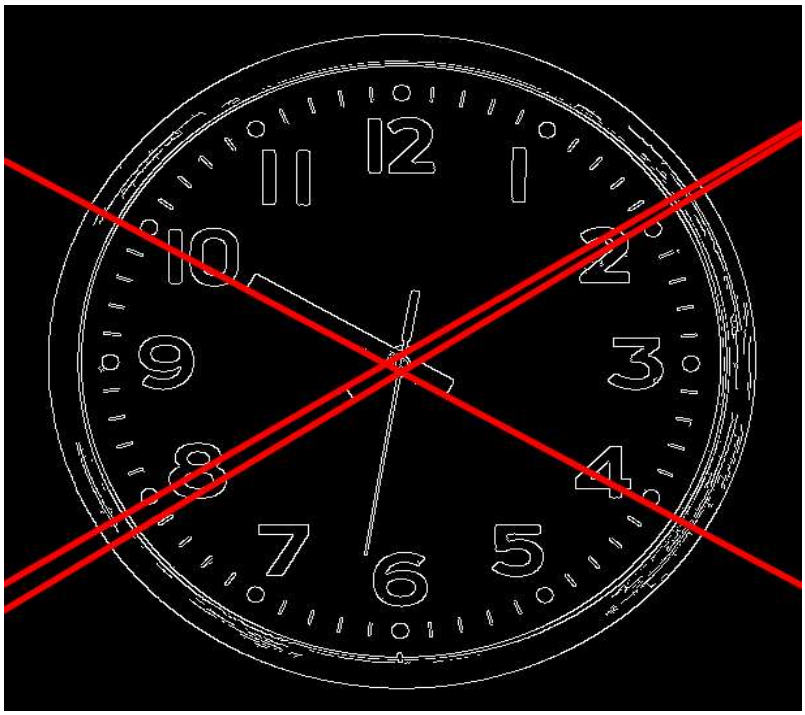
Plotted the Hough Space values of rho and theta



Output image:



Results using Hough Transform Function of OpenCV For the limited no. of the iterations



Observation:
Here We have seen that , In the scratch part some of the random lines are shown but mostly are accurately on the lines of the image. In the function's output we have drawn the limited lines and hence we see that the line obtained are accurate to the length of the clock's hand.

ColabLink:https://colab.research.google.com/drive/1EZHNDugr2YibY3G-IG3P9wrGO1MHCsrE?usp=sharing

Q.3 A manufacturing company in Bangalore, came up with the following problem statement: They have one reference design image for a part of equipment and a probe image of either faulty or perfect. You need to identify faulty image and show the defective region. (a) Submit your implementation. File name of your python file should be intelligentMatch.py, it should take two images (reference and probe) and outputs: faulty or perfect and if faulty it shows the region because of which it comes to conclusion that is faulty. (b) Show your results qualitatively in the report and write down your observation

Ans:

a)

The problem statement involves identifying whether the probe image is faulty or perfect and if faulty, identifying the defective region. This can be achieved by following the steps below:

Load the reference and probe images using an image processing library such as OpenCV.
Preprocess the images to enhance their features and reduce noise. This can be done using techniques such as smoothing, thresholding, and edge detection.
Extract features from the images that can be used for comparison. This can be achieved using techniques such as feature extraction or descriptor extraction,Homography and Warp Perspective.
Compare the features of the reference and probe images by taking the warp perspective of the gray image and after that taking the absolute difference of the to determine their similarity. This can be done using techniques SIFT DETECTOR and then binarizing the image.
If the images are similar, then the probe image is perfect. Otherwise, it is faulty. If the probe image is faulty, then the defective region can be identified by comparing the differences between the reference and probe images.

b)

Results:

We have taken a reference image and one Faulty image.

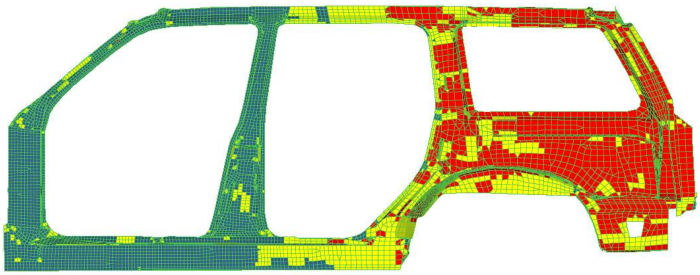


Fig:Faulty Image

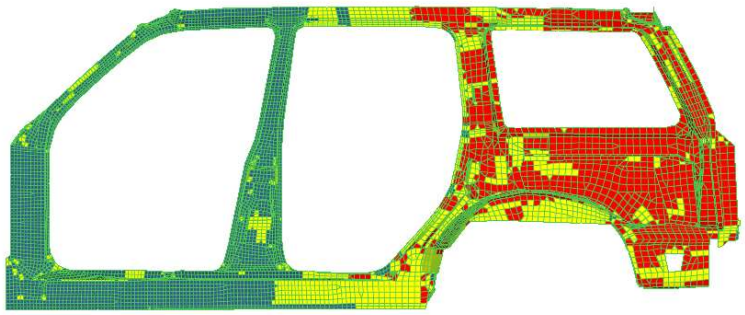
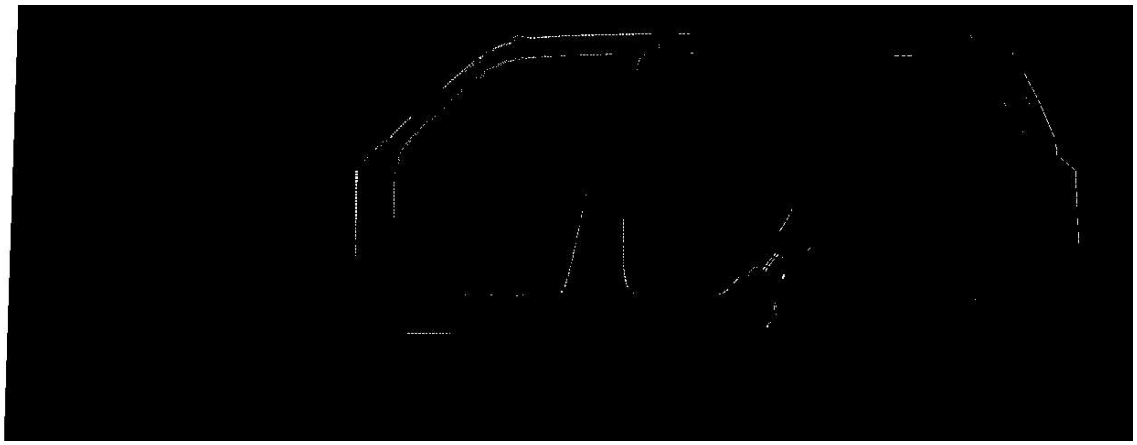
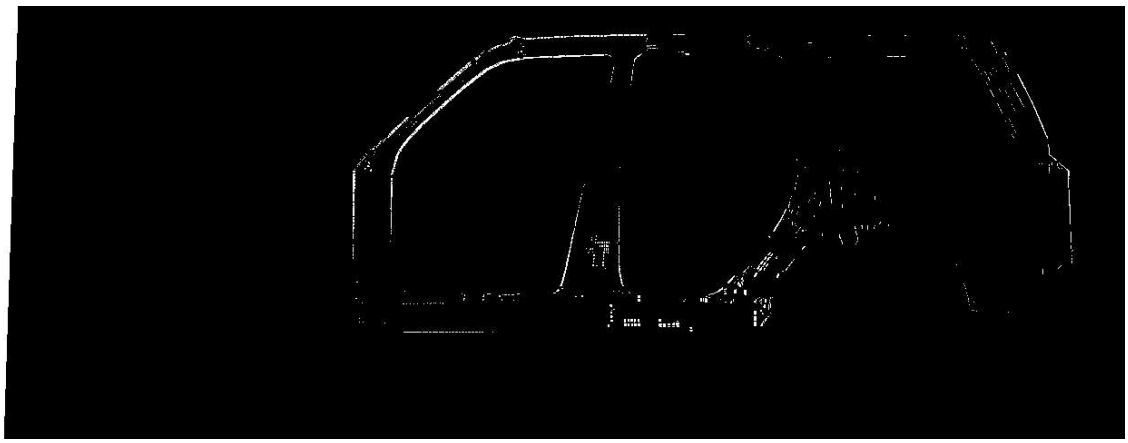


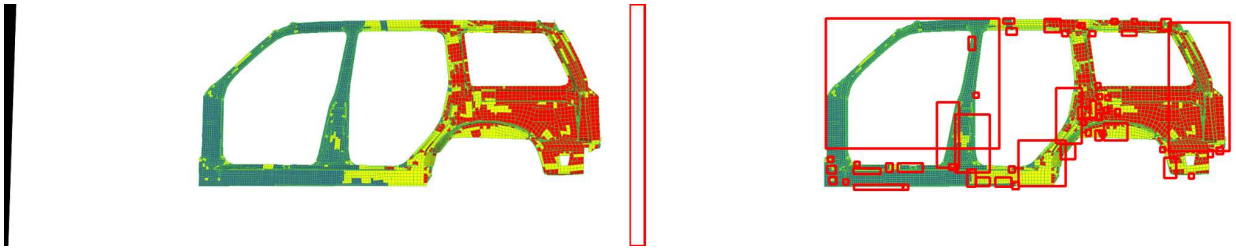
Fig:Reference image

Absolute Difference And Binarised image of Diference of image :





Faulty Part of the image in the Red BOX:



Observations:

The success of the implementation depends on the accuracy of the feature extraction and comparison techniques used. Also, the quality of the reference image and probe image can significantly affect the results. Therefore, it is important to ensure that the images are of high quality and have sufficient resolution for accurate comparison.

Colab link:
https://colab.research.google.com/drive/1yVfFuLHq81cx5kZUG2CFc0GDI5-e9dcl?usp=sharing

ASS 2B Report:

Ans1:

A homography matrix is a 3x3 transformation matrix that relates the transformation between two images of the same scene seen from different points of view to a planar scene. This is also known as a projective transformation or a perspective transformation in which just using that matrix we transformed the image in one plane to the other plane.

In computer vision and image processing, homography matrices are used for image registration, image stitching and object recognition tasks and augmented reality task.. Homography matrices can describe various types of image transformations, such as translation, rotation, scaling, and warping.

In a homography matrix, the first two columns represent rotation and scale transformations, while the third column represents translation transformations.
The last row of the matrix is usually set to [0 0 1] to ensure that the matrix is invertible.

The homography matrix is calculated according to the corresponding points between the two images and also it requires the usage of least square method .. The calculation of the homography matrix requires at least four points of similarity. Match points are selected manually or detected automatically using feature detection algorithms. The homography matrix is calculated by solving a system of linear equations using linear algebra techniques.
Once the homography matrix is calculated, it can be used to warp an image on the same plane as another image, which is useful in image registration and stitching applications. It can also be used to project 3D objects onto a 2D plane, which is useful in augmented reality applications.

The steps to calculate Homography matrix are:-
Correspondence Points Selection: Choose a set of corresponding points between the two images that you want to match. These points must be manually selected in both images by an expert user or using feature detection algorithms.

Homogeneous Coordinate Representation: Represent the corresponding points as homogeneous coordinates. Homogeneous coordinates are used to represent points in 2D or 3D space by adding an extra dimension to the vector. This extra dimension is typically set to 1.

Constructing the Homogeneous System of Equations: we have used the homogeneous coordinates to make a system of linear equations. minimum of four point must be used from the correspondences are required to solve for the Homography Matrix.

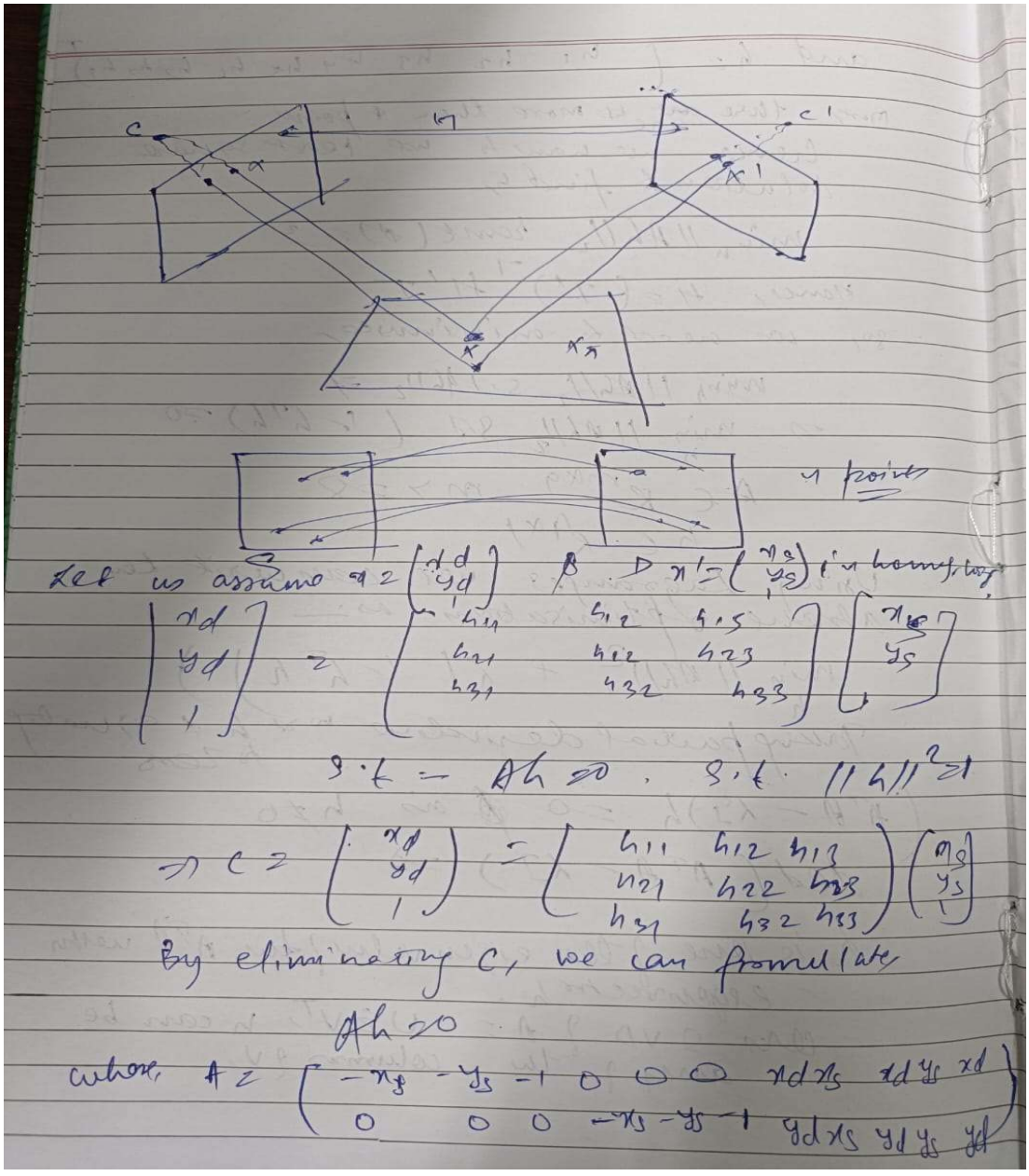
Solving the System of Equations: After making the system of equations we need to solve the homogeneous system of equations using a linear algebra solver, such as Singular Value

Decomposition (SVD) /Gaussian Elimination. This will result in a vector h containing the nine elements of the Homography Matrix H.

Normalizing the Homography Matrix:After that we need to normalize the Homography Matrix by dividing all its elements by h33, to ensure that the last element of the matrix is equal to 1.

Reshaping the Homography Matrix:we have to reshape the normalized Homography Matrix into a 3x3 matrix, and the matrix is ready to use.

We can get a more clear concept by seeing the example and mathematical formulations:-
OBTAINING H (HOMOGRAPHY MATRIX)-



and $h_2 \left(h_1 \; h_2 \; h_3 \; h_4 \; h_5 \; h_6 \; h_7 \; h_8 \; h_9 \right)^T$

more there are more than 4 points.

Hence, we have to use least square solution to find h,

$\min_h \|Ah\|_2$ such $(A) = 3$

Hence, $H \in (T^*)^{-1} H' T$

so, we need to minimise,

$\min_h \|Ah\|_2$ s.t $\|h\|_2 = 1$

$\Rightarrow \min_h \|Ah\|_2$ s.t $(1 - h^T h) = 0$

$A \in R^{m \times 9}$ $m >= 8$

$h \in R^{9 \times 1}$

Using Lagrange we can write the above optimisation as:—

$$\min_h \|Ah\|_2^2 + \lambda(1 - h^T h)$$

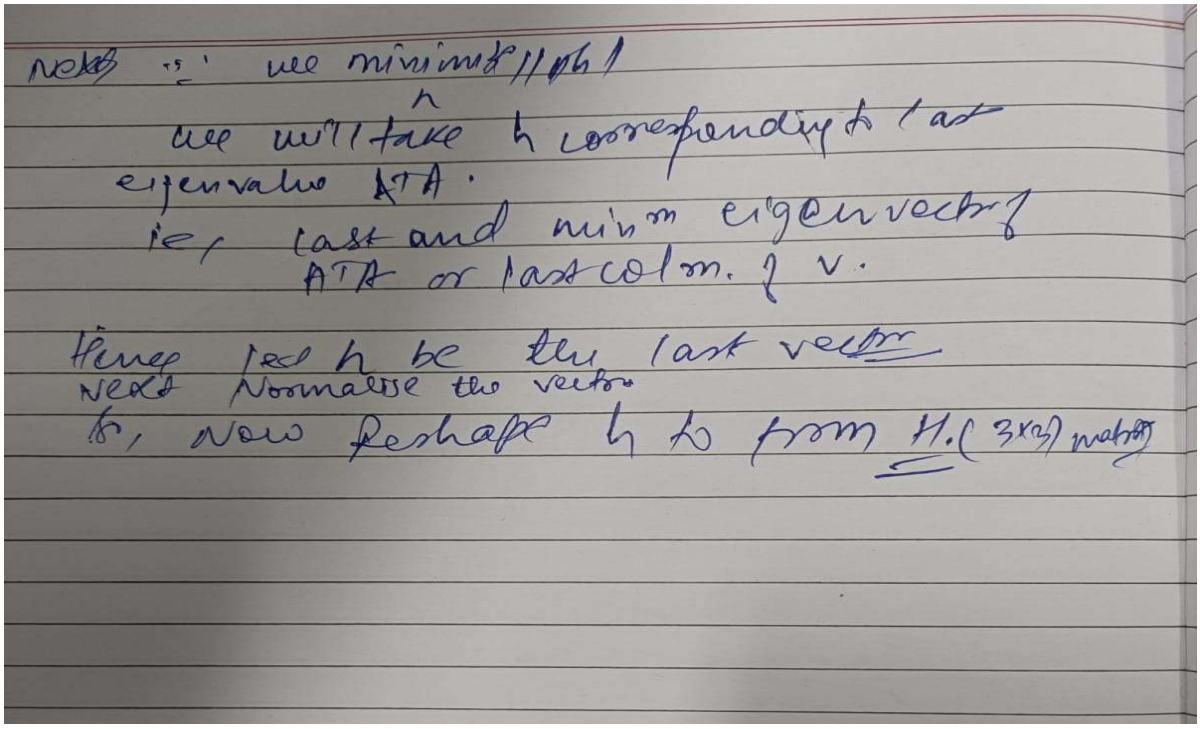
Taking partial derivative w.r.t h & equating to zero

$$(A^T A - \lambda I) h = 0 \quad \text{as } h \neq 0$$

$$\det(A^T A - \lambda I) = 0$$

λ is one of the eigen value of the $A^T A$ with eigenvector h.

As SVD of $A = U \Sigma V^T$, h can be any of the columns of V.



Ans2.

Stereo Matching:

Stereo matching is the process of finding corresponding points between two or more images taken from different viewpoints of the same scene, also known as stereo images. It is a fundamental problem in computer vision and is used for applications such as 3D reconstruction, object recognition, and depth estimation.

Stereo matching works by finding corresponding pixels or features in each of the stereo images. This is typically done by comparing the intensity or color values of the corresponding pixels or features in each image. The goal is to find the displacement or disparity between the corresponding pixels or features, which can be used to estimate the depth or distance between the camera and the scene.

The process of stereo matching involves the following steps:

Image Acquisition: Two images of the same scene are captured from two different viewpoints.

Image Rectification: The images are rectified to ensure that corresponding epipolar lines are parallel to each other. This simplifies the stereo matching process by reducing the search space.

Feature Extraction: Salient features such as edges, corners, or blobs are extracted from the rectified images.

Correspondence Search: Corresponding features in the two images are matched by comparing their descriptors or by computing similarity metrics such as sum of squared differences (SSD) or normalized cross-correlation (NCC).

Disparity Calculation: The disparity or depth information of each corresponding feature is calculated using triangulation, where the distance between the two camera viewpoints and the position of the feature in each image are used to calculate its 3D position.

Disparity Refinement: The calculated disparities may contain errors or outliers, which can be reduced or eliminated by applying post-processing techniques such as smoothing, filtering, or outlier rejection.
Stereo matching can be challenging due to various factors such as occlusions, lighting changes, and noise in the stereo images. To overcome these challenges, stereo matching algorithms often use various techniques such as filtering, regularization, and post-processing to improve the accuracy of the disparity map.

APPLICATIONS OF STEREO MATCHING IS STATED BELOW AS:
Stereo matching has various applications, such as 3D reconstruction, object recognition, and robotic navigation and its application like moving the robots in an unknown area. It is used in autonomous vehicles, robotics, and augmented reality (AR) systems.

Q3.Write down steps to stitch images to create the panorama. Use the three Taj Mahal Images provided with this assignment to create one panorama. Show panorama into one
Ans3.
Here are the steps to stitch images to create a panorama using OpenCV homography and warp projection:

1. Load the images to be stitched using OpenCV's imread() function.
2. Detect key points and extract feature descriptors using an algorithm such as SIFT, SURF or ORB.
3. Match the key points between the images using a descriptor matching algorithm such as Brute-Force, FLANN or BFMatcher.
4. Use a RANSAC-based algorithm such as findHomography() to compute the homography matrix that maps the points from one image to the other.
5. Use the homography matrix to warp one image onto the other using warpPerspective() function.
6. Combine the warped images into a single panorama using a blending technique such as multi-band blending or feathering.

a)Here we have initially taken input all the three images and after that preprocess the image like resizing and rotating and after that converted the image to the black and white and after

that used the ORB descriptor to find the keypoints and descriptor of all the three images and after that created the BFmatcher object and then match the keypoints using the Knn of the objects created . and after that applied the ratio test and performed the operations with img1 and img2 and also img2 and img3 .Next find the homography between img1 ,img2 and img2,img3 .After that applied the warp projection using the Homography results of M1.M2 of the Homography point.

b)Results:
Input 3 image which we want to stitch :



Output image:

Colab link:
https://colab.research.google.com/drive/1l80iCyj8xyuX0oTjnStqW8QJeT_RnfZ7?usp=sharing

References of Ass2a:
https://docs.opencv.org/4.x/dc/dc3/tutorial_py_matcher.html#:~:text=FLANN%20based%20Matcher,than%20BFMatcher%20for%20large%20datasets.
https://docs.opencv.org/3.4/d9/db0/tutorial_hough_lines.html
https://docs.opencv.org/4.x/d4/dc6/tutorial_py_template_matching.html
https://pypi.org/project/easyocr/
https://www.pythonpool.com/cv2-findhomography/
https://docs.opencv.org/4.x/dc/dc3/tutorial_py_matcher.html

References of Ass2b:
https://towardsdatascience.com/estimating-a-homography-matrix-522c70ec4b2c
https://docs.opencv.org/4.x/dc/dc3/tutorial_py_matcher.html#:~:text=FLANN%20based%20Matcher,than%20BFMatcher%20for%20large%20datasets.
https://towardsdatascience.com/image-panorama-stitching-with-opencv-2402bde6b46c
https://docs.opencv.org/4.x/dc/dc3/tutorial_py_matcher.html
https://python.plainenglish.io/opencv-image-stitching-second-part-388784ccd1a
https://www.pythonpool.com/cv2-findhomography/
https://theailearner.com/tag/cv2-warpperspective/