**Method Used for corner detection**  - Y=2

Hessian

1. Compute the Hessian matrix at each pixel of the image. The Hessian matrix describes the curvature of the image at that pixel.
2. Compute the determinant and trace of the Hessian matrix at each pixel. The determinant and trace values provide information about the shape of the local neighborhood around the pixel.
3. Threshold the determinant to identify the corners. Pixels with values greater than threshold are considered corners, while pixels with low values are discarded.
4. Perform non-maximum suppression to remove multiple responses from the same corner.
5. Repeat the process at multiple scales to detect corners at different levels of detail in the image.
6. Using OpenCV we then show the images along with corners on them

**Matching using proximity and ssd method –**

We match the detected corners in both images by computing the SSD (sum of squared differences) between corresponding patches around each pair of corners. If the SSD is below a certain threshold and the ratio between the SSD values of the best and second-best matches is below a certain ratio, then the pair of corners is considered a match. Finally, a list of matched corners in both images is returned.

The ssd score is computed using ssd match function and naïve matching implements the remaining algorithm. Using OpenCV we then show the matching in the images.

**Stitching explanation –**

The function takes two images and an affine matrix as input. It first transforms the second image using the matrix and concatenates it with the first image to create a panorama. It then applies a mask to crop the unnecessary parts of the panorama and returns the resulting image.

To run the code simply run the code in vs-code , ensure that you have all the libraries that need e.g.- OpenCV etc. as specified in requirements file also ensure that you change the directory while running the code.

Link for output images – [Output images](https://csciitd-my.sharepoint.com/:f:/g/personal/mcs222067_iitd_ac_in/EgCLRrbq4dJGvNuKLsW7fzgBcOww_-qZ8bE3OWipfGSzxw?e=xrwZkV)