**Assignment #2 Report**

**Computer Vision**

| No | Name | ID |
| --- | --- | --- |
| 1 | عبدالرحمن السيد جاد السيد | 19015894 |
| 2 | عبدالعزيز محمد عبدالعزيز محمد | 19015941 |
| 3 | عمر خيرت محمد ابو ضيف | 19016063 |

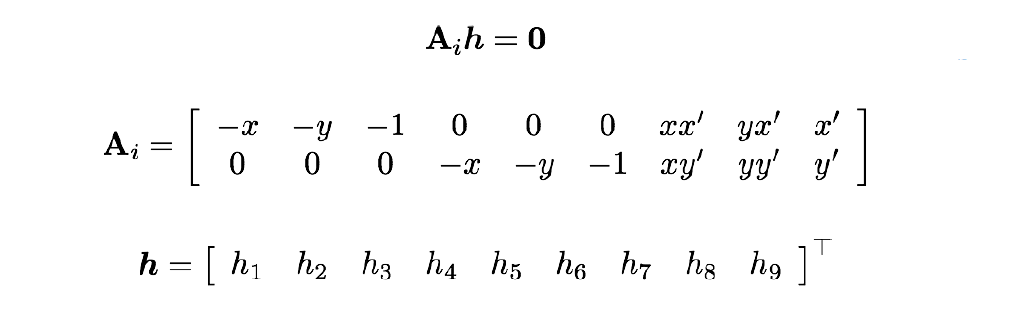
Google colab link: [here](https://colab.research.google.com/drive/1SBxA_tnvjzg3STob3GDWMqTcu0impYDc?usp=sharing)

**Part 1: Augmented Reality with Planar Homographies**

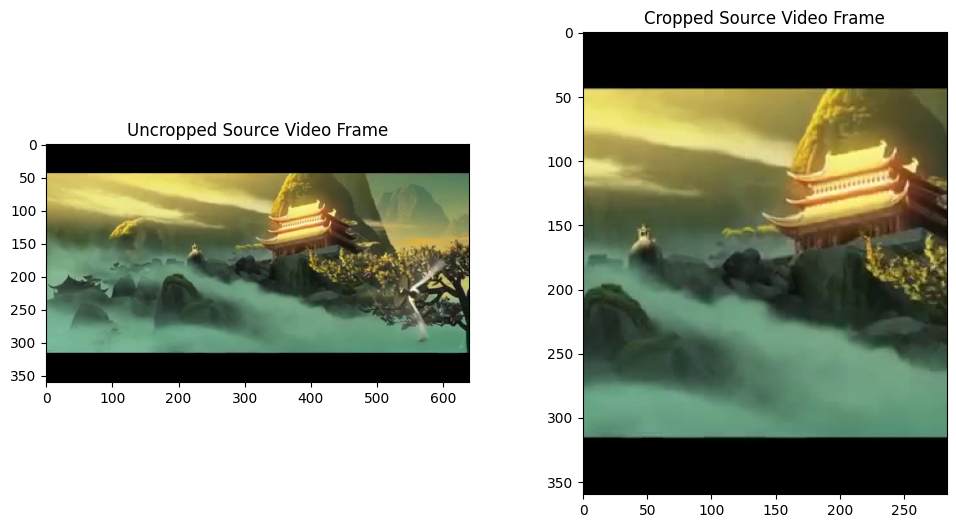
The goal of the assignment is to implement an Augmented Reality (AR) application using planar homographies. The specific task is to overlay each frame of "source.mov" onto the book in "book.mov,". The final output should be a new video consisting of frames with the desired overlay effect.

The process steps are the following:

1. Library needed: cv2, numpy, pyplot from matplot
2. Read “cv\_cover.jpg” as the book reference in grayscale, “book.mov” and “source.mov”.
3. Get correspondences between CV cover image and “book.mov” frame:
4. Get keypoints using SIFT descriptor for each.
5. Match keypoints using KNN with size 2.
6. Apply ratio check between the best 2 matches and filter at ratio = 0.65
7. Get the best 50 match.
8. Apply RANSAC to get inlier matches with considering inlier matches that has error after transformation less than 5 and max number of iterations is 2000.A collage of books

   Description automatically generated
9. Compute the homography parameters by solving the following system of equations:
10. Map point with the homography matrix as follows:A number of lines with numbers and symbols

    Description automatically generated with medium confidence
11. Get the book corner using the CV cover image width and height and the homography matrix:A group of books on a blue surface

    Description automatically generated
12. Crop “source.mov” video frame so it has the same aspect ratio of the book.
13. Overlay the 2 frames:
14. Compute the perspective matrix between the cropped frame coordinates and the book corners.
15. Wrap the cropped frame and mask it.A group of books on a blue surface

    Description automatically generated
16. Write the output frame in “output.mov” and repeat the steps from step 2 for the next frame.

Output video: [here](https://drive.google.com/file/d/1OV-MXfGmziC1XSmPzNS7j8VPJZ_Tf9L7/view?usp=sharing)

**Part 2: Image Mosaics with Bonus**

The objective of this assignment section is to develop an image stitcher that utilizes image warping and homographies to automatically generate an image mosaic. The primary scenario involves two input images that are intended to be combined into a mosaic. The implementation focuses on warping one image into the plane of the second image and displaying the combined views, maintaining the original input appearance in the final mosaic.

The process steps are the following:

1. Read the images.
2. Get correspondence same as part 1.A comparison of a house and trees

   Description automatically generated
3. Compute the homography matrix same as part 1.

A number on a white background

Description automatically generated

1. The recovered homography matrix is used to warp the first image to the perspective of the second image return a new image that is the warp of the input image using H . Since the transformed coordinates will typically be sub-pixel values, this is solved by rounding the resulting coordinatesA building with a tower in the background

   Description automatically generated
2. The elimination of holes in the warped image involves an inverse warp. This process entails mapping the coordinates of the holes from the warped image back into the original image before the warping. Subsequently, a bounding box is computed in the source image. All points within this destination bounding box are then sampled from the corresponding coordinates in the source image using bilinear interpolation.A statue in front of a building

   Description automatically generated
3. After transforming the source image to align with the destination image's frame of reference, we can generate a composite image displaying the mosaic. Generate a new image with sufficient size to accommodate both registered views. Overlay one view onto the other, leaving black regions wherever there is no available data.A fountain in front of a building

   Description automatically generated
4. **Bonus part**: Instead of stitching only two images, the process involves merging three overlapping images. Initially, the first two images are stitched together using established procedures. Subsequently, correspondences are identified between the output of stitching the initial two images and the third image, ensuring the correct order of stitching. Finally, the output from the initial stitching is combined with the third image to create the ultimate composite output.A black and white photo of a city

   Description automatically generated