INDIAN INSTITUTE OF TECHNOLOGY, KHARAGPUR
DEPARTMENT OF COMPUTER SCICENCE AND ENGINEERING

Advanced Digital Image Processing and Computer Vision

Assignment-3

Report

Advisor: Prof. Jayanta Mukhopadhyay

IIT KHARAGPUR, MARCH 2024

Member List : Group C

Name	Roll No
Vikas Vijaykumar Bastewad	20CS10073
Pranil Dey	20CS30038

Part 1: Compute Pixel Coordinates

In this part of the assignment, we loaded the image "Sheephard_Iasi.jpg" and implemented a function to get the pixel coordinates of a selected point on the image. We displayed the image and waited for a mouse click event. Upon clicking on the image, the pixel coordinates of the selected point were printed to the console. This functionality allows users to interactively obtain the pixel coordinates of any point on the image.

Part 2: Compute Line Length

For this part, we extended the functionality to compute the length of a line segment defined by two selected points on the image. After loading the image, users were prompted to select two points by clicking on the image. Upon selecting the two points, the distance between them was calculated using the Euclidean distance formula. The distance in pixels was then printed to the console, and a line segment was drawn between the two selected points on the image for visualization.

Part 3: Identify Perpendicular Lines and Compute Transformed Dual Conic

In this part, we focused on analyzing a painting image to identify at least five different pairs of lines that are supposed to be perpendicular in the original painting. We employed edge detection and Hough Transform techniques to detect lines in the image. Then, we iterated through the detected lines to find pairs that are approximately perpendicular. For each perpendicular pair, we computed the transformed dual conic at infinity using geometric calculations. The resulting transformed dual conic at infinity was printed to the console.

Part 4: Compute Homography Matrices for Mapping to Target Rectangles

Here, we computed Homography matrices to map the painting image to target rectangles with specified aspect ratios. We defined the target aspect ratios as (2:3) and (3:4) and computed the Homography matrix for each target aspect ratio using the findHomography function from OpenCV. The image was warped using the computed Homography matrices to produce transformed images with the desired aspect ratios. Transformed images were displayed for visualization.

Part 5: Perform Affine Rectification

For affine rectification, we provided a transformation matrix manually to rectify distortions in the painting image. We defined the transformation matrix to correct skew by shearing the image horizontally. The shear factor was adjusted based on the skew correction needed. The warpAffine function from OpenCV was used to apply the transformation to the image. The original and rectified images were displayed for comparison.

Part 6: Perform Metric Rectification

In this part, we performed metric rectification on the painting image using a homography matrix. Corresponding points on the image were manually selected and used to compute the homography matrix. The warpPerspective function from OpenCV was employed to apply the homography transformation, resulting in a metrically rectified image. Additionally, the true aspect ratio of the transformed image was computed and printed to the console for analysis.

Conclusion

The image processing operations performed in this assignment demonstrate various techniques for analyzing and rectifying images. These operations enable tasks such as obtaining pixel coordinates, measuring distances, identifying geometric properties, and transforming images to correct distortions. By implementing these operations, we gain insights into the geometric properties of images and can manipulate them to achieve desired transformations for further analysis or visualization purposes.