MAGPANTAY NINO JANDEL C.

BSCS-4B

CSST106

Perception and Computer Vision

Machine Problem No. 4

Task 1: Harris Corner Detection

Harris Corner Detection is a widely used algorithm in computer vision for identifying points of interest in images, known as corners. The process begins by loading a grayscale image, which simplifies the analysis by reducing the amount of information processed. The algorithm computes the intensity gradients of the image to identify changes in intensity, which are indicative of corners. It then applies a response function to determine the strength of each corner point, effectively highlighting areas of high curvature. By setting a threshold, only the most significant corners are retained for further analysis. The original image is then displayed alongside the processed image, where the detected corners are marked, typically in red. This dual visualization allows for a clear comparison between the original image and the locations of the identified corners, facilitating tasks like feature matching and image registration.

Task 2: HOG Feature Extraction

Histogram of Oriented Gradients (HOG) feature extraction is a powerful technique commonly used in object detection and image recognition. The process begins by loading an image, preferably of a person or an object of interest, and converting it to grayscale, as HOG relies on intensity rather than color. HOG features are then extracted by calculating the gradients of pixel intensities, which capture the structure and shape of objects within the image. This technique divides the image into smaller regions, or cells, computes histograms of gradient directions, and normalizes them to achieve robustness against changes in illumination. A visual representation of the HOG features is generated, allowing for a clear understanding of how features are distributed across the image. The original image and the HOG visualization are displayed side by side, making it easier to see how well the features correspond to the actual structure of the object. This method is particularly valuable in applications such as pedestrian detection and activity recognition, where understanding object shapes and orientations is crucial.

Task 3: ORB Feature Extraction and Matching

The ORB (Oriented FAST and Rotated BRIEF) algorithm is an efficient method for feature extraction and matching in computer vision applications. The task begins by loading two distinct images and utilizing the ORB algorithm to detect keypoints and compute their descriptors. ORB is designed to be fast and rotation invariant, making it suitable for real-time applications where computational efficiency is essential. Once the keypoints and descriptors are obtained, a FLANN (Fast Library for Approximate Nearest Neighbors) matcher is employed to match the descriptors between the two images. This matcher is optimized for high-dimensional data, which is characteristic of image descriptors, and effectively identifies corresponding keypoints across the images. The matched keypoints are visualized in a combined image, providing a clear representation of how the algorithm aligns similar features between the two images. This feature matching process is fundamental in applications such as image stitching, object recognition, and 3D reconstruction.

Task 4: SIFT and SURF Feature Extraction

SIFT (Scale-Invariant Feature Transform) and SURF (Speeded-Up Robust Features) are two popular algorithms used for extracting and describing keypoints in images, particularly in complex scenarios. In this task, two images are loaded, and both algorithms are applied to detect keypoints and compute their descriptors independently. SIFT is known for its robustness against scale and rotation, while SURF offers improved speed, making both techniques valuable for different applications. Once the keypoints and descriptors are extracted, the results are visualized by drawing the detected keypoints on the original images. This visualization enables a comparison of how each algorithm captures features, revealing differences in sensitivity to changes in scale, rotation, and lighting conditions. By analyzing the visualized keypoints, researchers can choose the most suitable method based on the specific requirements of their applications. Overall, the ability to extract and visualize keypoints is crucial for numerous computer vision tasks, including object recognition and scene matching.

Task 5: Feature Matching using Brute-Force Matcher

Feature matching using the Brute-Force Matcher is a fundamental step in many computer vision applications, enabling the identification of corresponding features across different images. This task begins by loading two images and extracting ORB descriptors from them, which are key to determining similarity between features. The Brute-Force Matcher, which compares each descriptor from one image with every descriptor from the other image, is then employed to find the best matches based on a specified distance metric. This method, while computationally intensive, ensures thorough matching by exhaustively searching through all possible pairs of descriptors. The matches are sorted to identify the best correspondences, enhancing the efficiency of subsequent processing steps. The matched keypoints are visualized in a combined image, highlighting the relationships between corresponding features in both images. This visualization not only aids in evaluating the quality of the matches but also serves as a foundational step for further tasks, such as image stitching or object tracking.

Task 6: Image Segmentation using Watershed Algorithm

The Watershed algorithm is a powerful technique for image segmentation that helps to delineate distinct regions within an image based on the intensity of pixel values. The process begins by loading an image, converting it to grayscale, and applying a binary threshold to separate the foreground from the background effectively. Morphological operations are used to clean up noise and refine the boundaries of the objects within the image. By identifying the sure background and foreground areas, the algorithm sets up markers for the Watershed algorithm to process. The Watershed transformation treats the intensity image as a topographic surface, effectively simulating water flooding to create distinct segments. The final segmented image highlights boundaries and distinct regions, providing a clear representation of the underlying structure. This segmentation technique is widely applied in fields such as medical imaging, object detection, and scene analysis, making it essential for various computer vision applications.