Giorgio Mendoza
CS549-E24-E01
Dr. Alexandros Lioulemes

Week 3 Assignment

The task involves implementing the initial stages of the Scale-Invariant Feature Transform algorithm in Python. Specifically, it includes the scale-space extrema detection, where a scale-space is constructed using Gaussian blurring to identify potential keypoints through a Difference of Gaussian pyramid. These keypoints are crucial as they represent invariant features across different views of the same object.

Additionally, the task includes accurate keypoint localization, which refines the positions of these keypoints to subpixel accuracy using a Taylor expansion of the scale-space function. This step ensures the robustness of the detected features, which is essential for subsequent tasks such as image matching and object recognition.

This implementation is guided by David Lowe's seminal paper on SIFT, which provides detailed methodologies for each part of the algorithm. Insights from MATLAB implementations also help in effectively translating the algorithm into Python. The focus of this assignment is solely on these first two foundational steps, which are critical for identifying and refining feature points in images.

Scale-Space Extrema Detection:

The scale-space extrema detection is implemented as part of the SIFT algorithm. This process involves generating Gaussian and DoG pyramids from the input image. Each level of the Gaussian pyramid is created by successively blurring the original image with a Gaussian filter, and each level of the DoG pyramid is computed as the difference between two successive Gaussian-blurred images. This difference highlights regions of the image that differ in intensity from their surroundings, which are potential keypoints.

After constructing the DoG pyramid, the find_scale_space_extrema function searches for local maxima and minima across scale and space. This is done by comparing each pixel in the DoG images to its neighbors in the current and adjacent scales. Keypoints are initially identified by their contrast and then refined to subpixel accuracy using the refine_keypoint_location function, which employs a Taylor expansion to the detected extrema. The refinement step was important for achieving a better keypoint localization as shown in the images below.

Accurate keypoint localization:

The accurate keypoint localization is achieved through subpixel refinement, using the refine_keypoint_location function. This process starts after initial keypoint detection from the scale-space extrema in the Difference of Gaussian (DoG) images.

The function calculates first and second derivatives at each keypoint to form the Jacobian and Hessian matrices. These derivatives assess the image intensity's change around the keypoint, helping identify the exact position with maximum contrast.

The First Derivatives (dx, dy, ds) indicate the rate of change across spatial and scale dimensions. While the Second Derivatives (dxx, dyy, dss, etc.) measure the curvature of intensity changes, important for determining the nature of the extremum.
The Hessian matrix is used to compute an offset by solving the linear equation formed with the Jacobian, precisely adjusting the keypoint's position. If the Hessian is singular, implying instability or edge-like features, the keypoint is discarded. This refinement ensures keypoints are positioned at significant intensity changes, enhancing feature reliability for further steps in the SIFT process.

The image shown was generated using basic SIFT processes but without the Taylor expansion series, which refines keypoint localization to subpixel accuracy. Without this refinement, the keypoints are positioned based solely on the initial Difference of Gaussian detection, leading to less precise localization. Essentially, this image represents keypoints identified before applying the precise localization adjustments offered by the Taylor expansion.

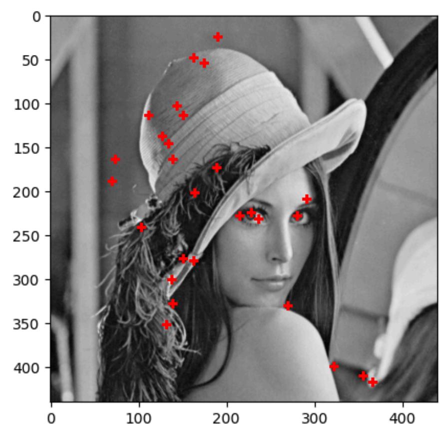

Figure 1. R = 10, contrast Threshold = 0.00025, Num octaves = 2 S = 1
Number of keypoints detected: 28

Figure 2 utilizes the Taylor expansion series for enhanced keypoint localization, as detailed in the paper "Distinctive Image Features from Scale-Invariant Keypoints." This technique refines each keypoint's position to subpixel accuracy, improving the precision of feature detection. Additionally, the contrast threshold was slightly increased to detect more keypoints, leading to a total of 74 keypoints, as depicted in the image.

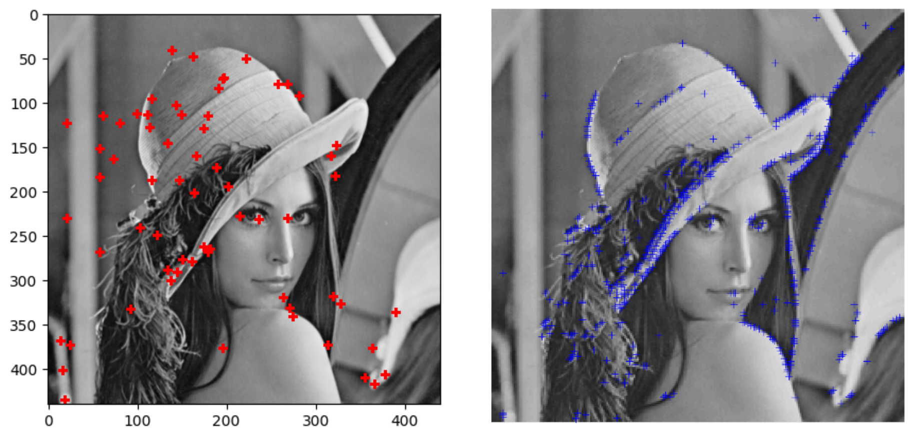

Figure 2. R = 10, contrast Threshold = 0.00022, Num octaves = 2, S = 1
Number of keypoints detected: 74

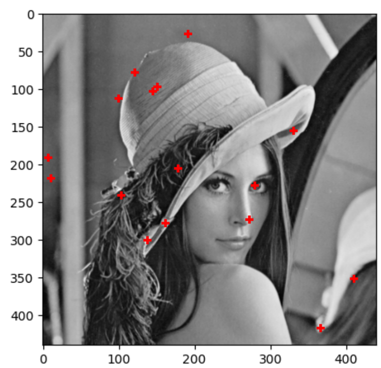

Figure 3. R = 10, contrast Threshold = 0.00026, Num octaves = 3 S = 2
Number of keypoints detected: 16

Increasing the number of octaves and intervals per octave was also tested, but even fewer points were detected. This reduction could be due to the increased sensitivity to noise and image variations at higher octaves. As the scale increases, smaller and less significant features might be smoothed out, resulting in fewer detected keypoints that meet the contrast threshold and edge response criteria. This could demonstrate a trade-off between detecting fine details at lower octaves and capturing larger, more significant features at higher octaves.

Conclusion:

The optimal results for this task were achieved with the settings detailed in Figure 2. Lowering the contrast threshold increased the number of detected keypoints, predominantly clustering them near the top left corner, potentially due to variations in image brightness or noise in that area.
Increasing the contrast would result in less points but some areas of interest would also be lost.
Various areas of interest were identified if compared with the reference image. For example, the contour of the hat, the eyes, some parts of the hair and the hat’s brim and the chin. Additionally, some areas of less interest were identified such as the top left corner and the bottom right corner in the reference picture.

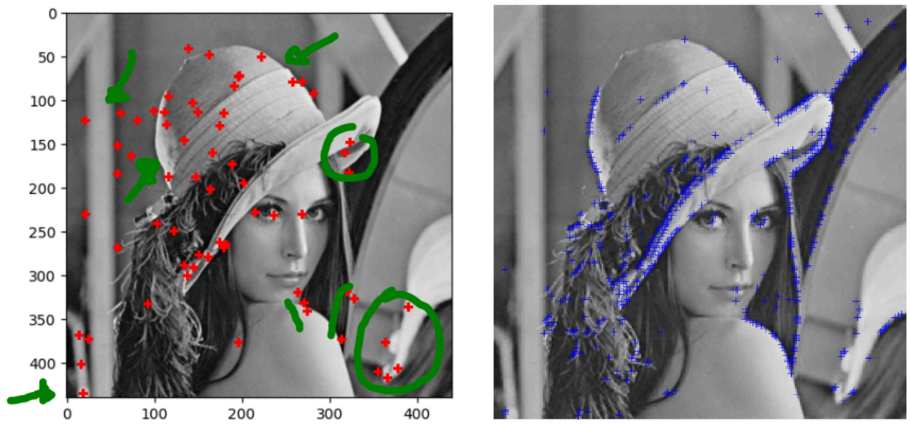

Figure 4. Final Comparison