Cairo university

Faculty of Engineering

Systems and Biomedical Department

Computer Vision (SBE 3230)

Assignment 3

Name	B.N	Task
Eman Abdelazeem	**12**	**SIFT**
Hassnaa Hossam	**20**	**Feature matching, UI**
Abdelrahman Alaa	**36**	**Harris Operator**
Farha Elsayed	**3**	**Harris Operator, UI**

Supervised by: Dr. Ahmed Badawy

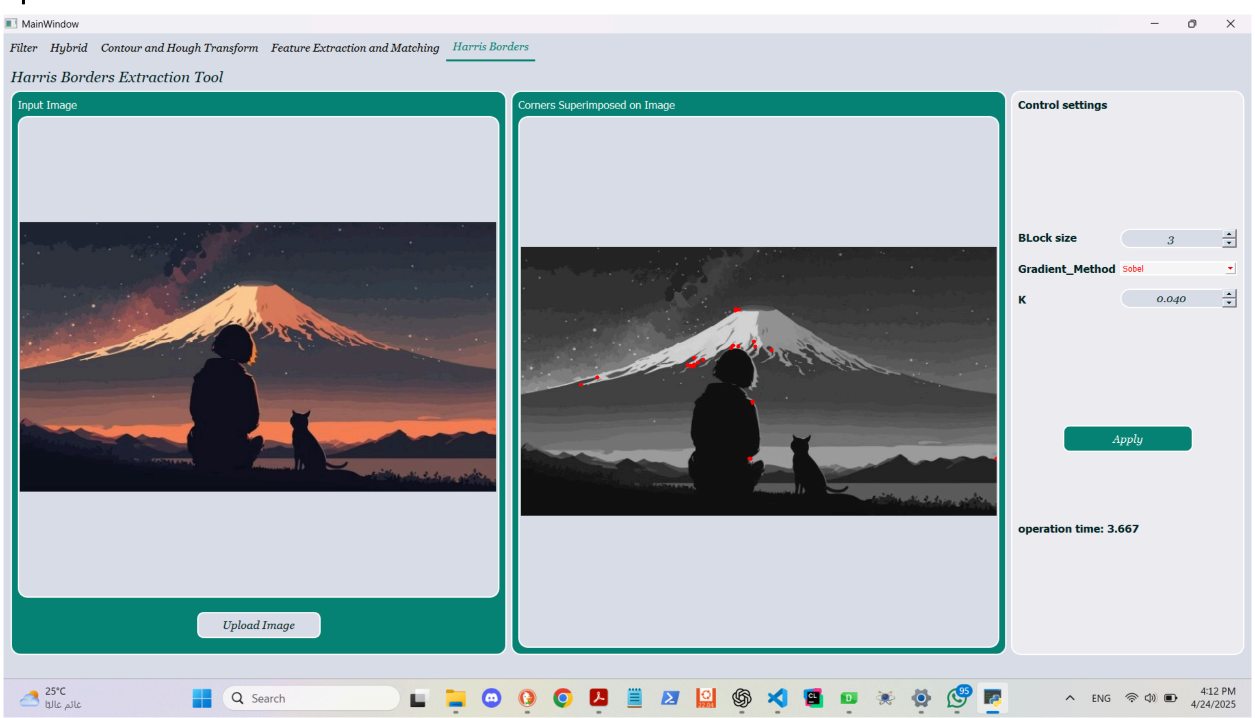
Submitted to: Eng. Yara wael &

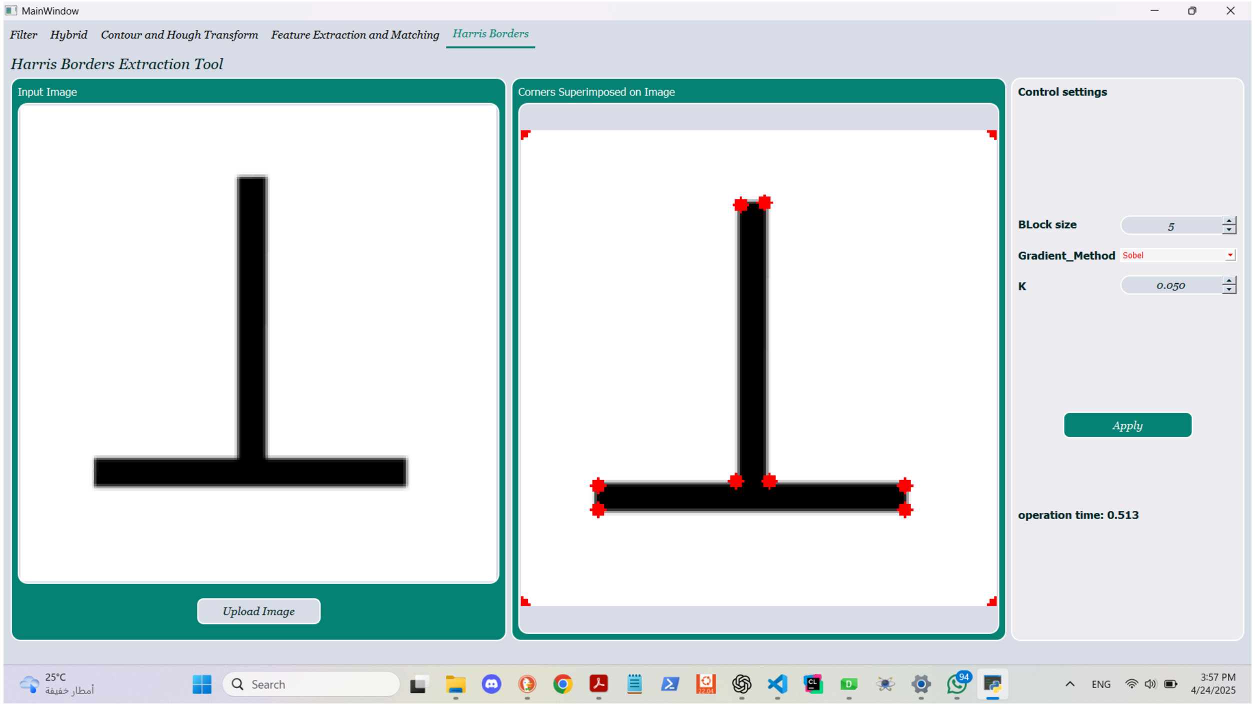
Eng. Omar Dawah

1.Harris Operator:

Harris' algorithm is based on one of the principles of mechanics called 2nd moment of Inertia. It is a measure of the object resistance to bending, moment, torque, torsion or rotation in general. This measure is calculated relative to an axis of rotation called the 2nd inertia moment axis. For a general object in the 2D space the 2nd moment of inertia is calculated around the x-axis denoted as Ixx, around the y-axis denoted as Iyy. Another measure for the 2nd inertia moment is the Ixy or the product of inertia that is a masseur of how well the shape is symmetric relative to the x and y axes.

When it comes to image processing the mechanics concept is applied to the shapes in images with some modifications. Harris Algorithm is the projection of the 2nd inertia moment on the image processing domain. Ixx here is calculated as Gx^2, Iyy = Gy^2 and Ixy = GxGy. These values are plugged in the harriss structure tensor matrix. It is a 2x2 matrix with Ixx, Ixy, Ixy and Iyy as the entries. The determinant will be hence defined as Det = (Ixx*Iyy) - (Ixy)^2. Trace = Ixx + Iyy. We then use the formula: HR = Det - k*Trace^2. K is an empirical variable called the Harris free parameter. HR is the response function that is able to classify regions into, flat regions, edges and corners. Here are some result from using the Harris operator in corner detection:





2. SIFT (Scale-Invariant Feature Transform):

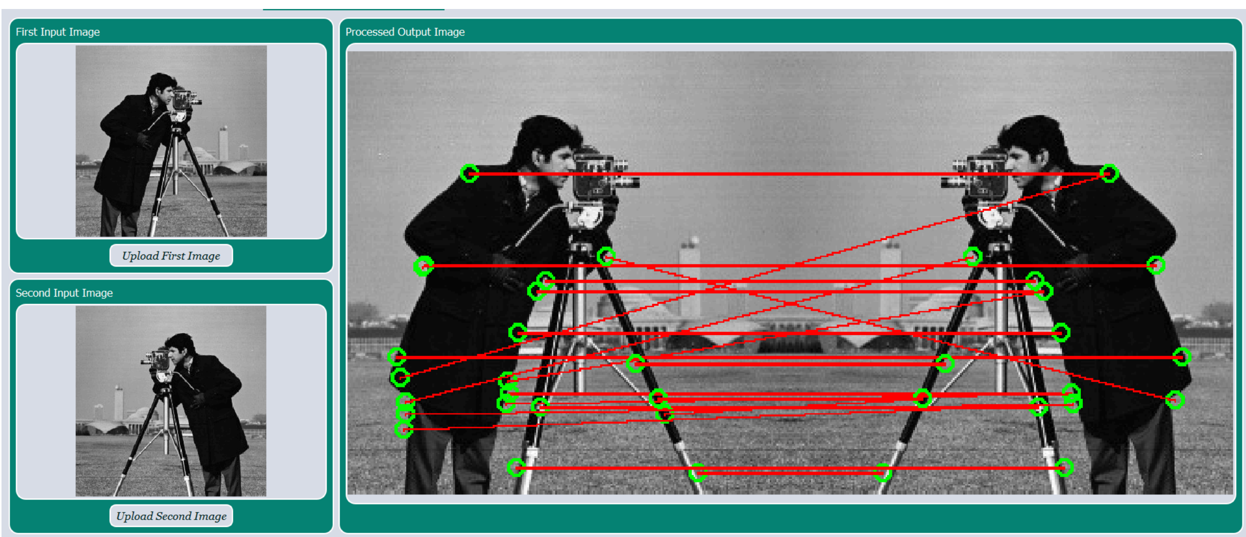
SIFT (Scale-Invariant Feature Transform) is used to find and describe important points in an image. These points stay reliable even if the image is resized, rotated, or slightly changed in lighting or angle. SIFT works by looking at the image at different scales and directions, then creates a unique description for each point so it can be matched with similar points in other images.

Steps:

1. Create a base image with applied Gaussian blur.
2. Calculate the number of octaves needed based on image dimensions.
3. Generate a set of Gaussian kernels for blurring at different scales.
4. Apply Gaussian filters to create a Gaussian scale space.
5. Generate Difference-of-Gaussian (DoG) images to detect edges and blobs.
6. Detect extrema in scale space that correspond to potential keypoints.
7. Remove duplicate or unstable keypoints.
8. Map the key points back to the original image size.

9. Generate descriptors for each keypoint, capturing orientation and local gradient features.

Results:



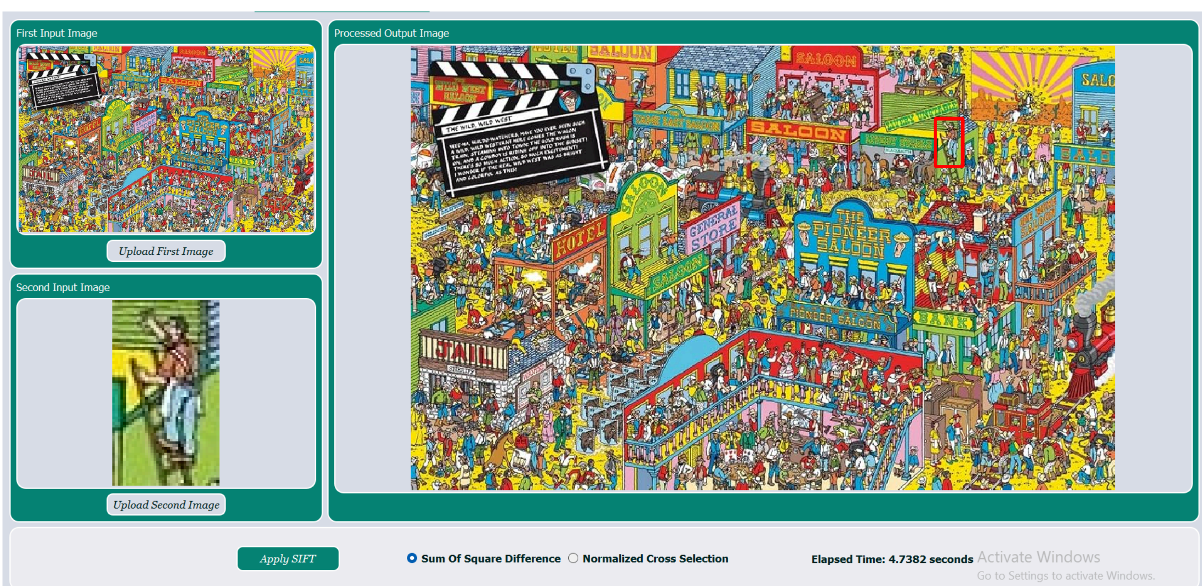
3. Feature Matching

1. Sum of Squared Differences (SSD):

The SSD method computes the squared differences between the template and a sliding window over the input image. The location with the smallest SSD value is considered the best match.

Steps:

1. **Image Preprocessing:**
   - Convert both the input image and the template to grayscale.
   - Create a copy of the original image (in color) for visualization.
2. **Validation:**
   - Ensure the template is smaller than the input image.
3. **Sliding Window Search:**
   - Slide the template over the input image in steps (stride).
   - At each position, calculate the SSD between the template and the current patch.
   - Keep track of the position with the minimum SSD.
4. **Visualization:**
   - Draw a red rectangle on the original image at the location of the best match.



2. Normalized Cross-Correlation (NCC):

NCC measures the similarity between the template and patches in the image. Unlike SSD, NCC is normalized, making it invariant to linear changes in brightness and contrast.

Steps:

1. **Image Preprocessing:**
    - Convert both images to grayscale and then to float32 for precise calculation.
    - Create a color copy of the input image.
2. **Template Normalization:**
    - Normalize the template by subtracting its mean and dividing by its standard deviation.
3. **Sliding Window Search:**
    - Slide the template over the image in steps.
    - Normalize each image patch and compute the NCC value.
    - Keep track of the position with the highest NCC.
4. **Visualization:**
    - Draw a blue rectangle on the original image at the location of the best match.