Features and Image Matching

Summarization:

Objectives: What is features and how can we utilize it to solve problems. Different methods to detect features. How to match multiple features.

1. What is features

A feature is a piece of information which is relevant for solving the computational task related to a certain application. Features may be specific structures in the image such as points, edges or objects. Features may also be the result of a general neighborhood operation or feature detection applied to the image. The features can be classified into two main categories:

- The features that are in specific locations of the images, such as mountain peaks, building corners, doorways, or interestingly shaped patches of snow. These kinds of localized features are often called **keypoint features** (or even corners) and are often described by the appearance of patches of pixels surrounding the point location.
- The features that can be matched based on their orientation and local appearance (edge profiles) are called **edges** and they can also be good indicators of object boundaries and occlusion events in the image sequence.

2. What it use for ?

   Features are used for:

   • Image alignment (e.g., panoramic mosaics)
   • Object recognition
   • 3D reconstruction (e.g., stereo)
   • Motion tracking
   • Indexing and content-based retrieval

- Robot navigation

Method of detection:

A. Harris Corner Detector

This method using corner as a interest point. Corner are regions in the image with large variation in intensity in all the directions. To find the corner point, they applied a small 3x3 window around each pixel in the image. Then they measures the amount of change in pixel values after shifting the windows a little bit in any direction.

Let us define the change function E(u,v) as the **sum** of all the sum squared differences (SSD), where u,v are the x,y coordinates of every pixel in our 3 x 3 window and I is the intensity value of the pixel. The features in the image are all pixels that have large values of E(u,v), as defined by some threshold.

$$E(u,v) = \sum_{x,y} w(x,y)[I(x+u,y+v) - I(x,y)]^2$$

We have to maximize this function $E(u,v)$ for corner detection. Applying Taylor Expansion to the above equation, we get the final equation as:

$$E(u,v) \approx [u \quad v] \left(\sum \begin{bmatrix} {I_x}^2 & I_x I_y \\ I_x I_y & {I_y}^2 \end{bmatrix} \right) \begin{bmatrix} u \\ v \end{bmatrix}$$

Let M be the summed-matrix:

$$M = \sum w(x,y) \begin{bmatrix} {I_x}^2 & I_x I_y \\ I_x I_y & {I_y}^2 \end{bmatrix}$$

We have:

$$E(u,v) \approx [u \quad v]\, M \begin{bmatrix} u \\ v \end{bmatrix}$$

Then using the eqution below to dertermine if it is a corner or not:

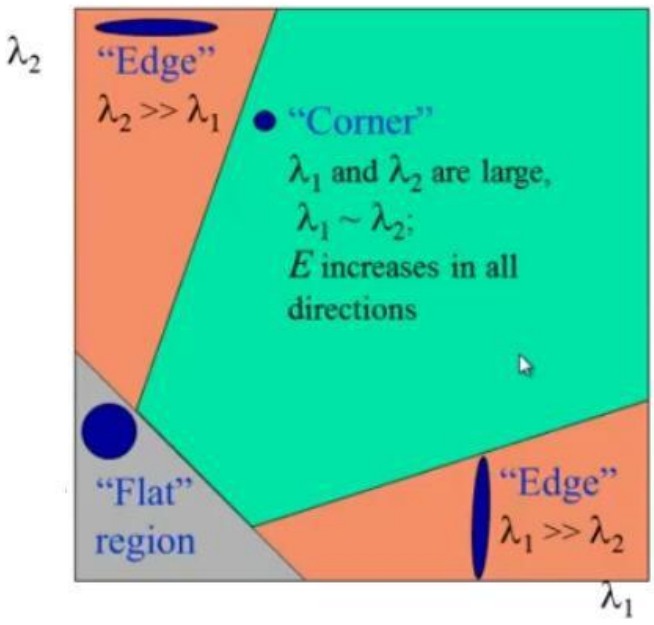
$$R = \det M - k(trace\, M)^2$$

$$\det M = \lambda_1 \lambda_2$$

$$trace\, M = \lambda_1 + \lambda_2$$

So the magnitudes of these eigenvalues decide whether a region is a corner, an edge, or flat.

- When $|R|$ is small, which happens when $\lambda 1$ and $\lambda 2$ are small, the region is flat.
- When $R<0$, which happens when $\lambda 1 >> \lambda 2$ or vice versa, the region is edge.
- When R is large, which happens when $\lambda 1$ and $\lambda 2$ are large and $\lambda 1 \sim \lambda 2$, the region is a corner.



B. Scale Invariant Feature Transform

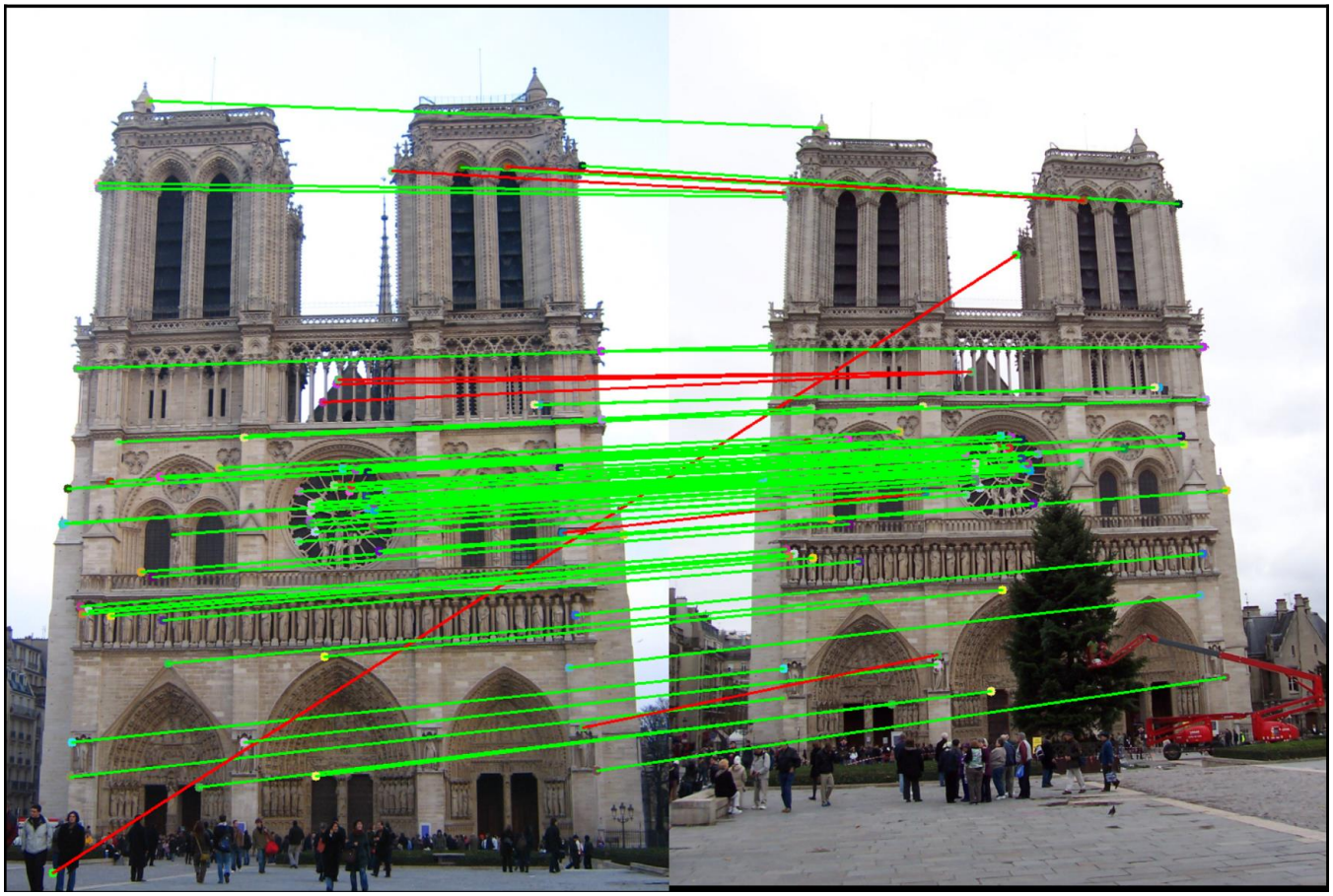
This is an advanced technique to use when matching feature. Other method like haris can only be useful when the images a similar in nature(same scale, orientation,…). When the images is different in scales and rotations, then we have to use this.

The algorithm follow a number of steps:

- Scale-space peak selection: Potential location for finding features.
- Keypoint Localization: Accurately locating the feature keypoints.
- Orientation Assignment: Assigning orientation to keypoints.
- Keypoint descriptor: Describing the keypoints as a high dimensional vector.
- Keypoint Matching

C. Image matching

Features matching or generally image matching, a part of many computer vision applications such as image registration, camera calibration and object recognition, is the task of establishing correspondences between two images of the same scene/object. A common approach to image matching consists of detecting a set of interest points each associated with image descriptors from image data. Once the features and their descriptors have been extracted from two or more images, the next step is to establish some preliminary feature matches between these images.



For this report we will be forcusing on using Brute-Force Matcher as this is quite simple. It takes the descriptor of one feature in first set and is matched with all other features in second set using some distance calculation. And the closest one is returned.