TP5: Interest Points and Object Recognition

Loic Simon Chaohui Wang Panagiotis Koutsourakis

23 Mai 2007

1 Theory

1.1 Harris Corner Detector

The Harris corner detector was proposed by Harris in 1988 [1]. It is one of most popular interest point detectors due to its invariance to rotation, scale, illumination variation and image noise [2]. This detector is based on the local auto-correlation function of a signal, which measures the local changes of the signal with patches shifted by a small amount in different directions.

Given a shift $(\Delta x, \Delta y)$ and a point (x, y), the auto-correlation function is defined as:

$$E(x,y) = \sum_{\Omega(x,y)} G_{(x,y)}(x_i, y_i)[I(x_i, y_i) - I(x_i + \Delta x, y_i + \Delta y)]^2 \tag{1}$$

where $I(\cdot,\cdot)$ denotes the image function, (x_i, y_i) the points in the window $\Omega(x, y)$ which is centered on (x, y), and $G_{(x,y)}(\cdot,\cdot)$ a Gaussian kernel function also centered on (x, y).

If the displacement is rather small, the shifted image can be approximated by a Taylor expansion truncated to the first order terms. Thus we get the formula below:

$$I(x_i + \Delta x, y_i + \Delta y) \approx I(x_i, y_i) + \begin{bmatrix} I_x(x_i, y_i) & I_y(x_i, y_i) \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \tag{2}$$

where $I_x(\cdot,\cdot)$ and $I_y(\cdot,\cdot)$ denote respectively the partial derivatives in the x and y direction.

And then, by substituting Eq. 2 into Eq. 1, we can get:

$$\begin{aligned} E(x,y) &= \sum_{\Omega(x,y)} G_{(x,y)}(x_i, y_i)(\begin{bmatrix} I_x(x_i, y_i) & I_y(x_i, y_i) \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix})^2 \\ &= \begin{bmatrix} \Delta x & \Delta y \end{bmatrix} M(x,y) \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \end{aligned} \tag{3}$$

where $M(x, y)$ is as:

$$M(x,y) = \begin{bmatrix} \sum_{\Omega(x,y)} G_{(x,y)}(x_i, y_i) I_x^2(x_i, y_i) & \sum_{\Omega(x,y)} G_{(x,y)}(x_i, y_i) I_x(x_i, y_i) I_y(x_i, y_i) \\ \sum_{\Omega(x,y)} G_{(x,y)}(x_i, y_i) I_x(x_i, y_i) I_y(x_i, y_i) & \sum_{\Omega(x,y)} G_{(x,y)}(x_i, y_i) I_y^2(x_i, y_i) \end{bmatrix} \tag{4}$$

Note that the matrix $M(x, y)$ captures the intensity structure of the local neighborhood. Let λ_1, λ_2 be the eigenvalues of matrix $M(x, y)$. The eigenvalues form a invariant description with respect to rotation. For each pixel (x, y) , there are three principal cases depending on the eigenvalues:

1. If both λ_1, λ_2 are small, the local auto-correlation function approximatively vanishes in any direction. We consider thus that the small image region around the pixel (x, y) is of approximately constant intensity.

2. If one eigenvalue is high and the other is low, the local auto-correlation function is ridge shaped, which means that a local shift along the ridge causes little change in $M(x, y)$ and significant change in the orthogonal direction. This indicates that an edge passes (x, y);

3. If both eigenvalues are high, the local auto-correlation function is sharply peaked, which means that a shift in any direction will result in a significant increase of $M(x, y)$. This indicates that there is a corner on (x, y).

In order to avoid the calculation of the two eigenvalues, we use instead the response function R defined in formula 5 to distinguish the three cases above.

$$R(x, y) = \det(M(x, y)) - k \cdot (\mathrm{trace}(M(x, y)))^2 \tag{5}$$

Note that $k = 0.04$ is a empirical value which is usually used in the literature. We can verify that $\lambda_1 \cdot \lambda_2 = \det(M)$ and $\lambda_1 + \lambda_2 = \mathrm{trace}(M)$. Thus we can consider that $R(x, y)$ is negative (with relatively big absolute value) for a point on the edge, close to 0 for a point in the homogeneous zone, and relatively big positive value for a point on the corner. Therefore, a point is considered as a corner if $R \geq s$ where $s > 0$ is a certain fixed threshold predefined by the user.

1.2 K-means Algorithm

The K-means algorithm is an common classification algorithm which is usually used to partition the given data which is composed of N unlabeled objects $\{x_1, x_2, \ldots, x_N\}$ into K clusters $\{G_1, G_2, \ldots, G_K\}$ $(K < N)$. The goal of this algorithm is to minimize the squared error function:

$$V = \sum_{k=1}^{K} \sum_{x_n \in G_k} \|x_n - \mu_k\|^2 \tag{6}$$

where μ_k $(k \in \{1, 2, \ldots, K\})$ is the center(mean) of the cluster k.

The idea of K-means algorithm is to minimize the squared error V iteratively by choosing the partition of data and by updating the center of the clusters, which can be summarized as the following steps:

1. Initialize the centers of K groups, i.e.: μ_k $(k \in \{1, 2, \ldots, K\})$

2. Partition the N objects into K groups using the labeling method as:

$$l_n = \arg\min_k \|x_n - \mu_k\|^2 \tag{7}$$

   where l_n represents the index of the group to which x_n should belong.

3. Update the center for each group:

$$\mu_k = \frac{1}{|G_k|} \sum_{i \in G_k} x_i, \quad k \in \{1, 2, \ldots, K\} \tag{8}$$

4. Repeat the steps 2 and 3 until the error V can't descend any more. In practice, we can detect this by observing if the centers of those groups (or the labelings of all the objects) don't change between two consecutive iterations.

We can use K-means algorithm to partition the detected interest points into several coherent groups. The classification will be done using the descriptors of these points. A simple choice of descriptor consists in using the intensity of those points around a considered point. We can also use those descriptors more complex like SIFT, which is based on a histogram of orientation of the gradients in a patch centered on the considered point.

2 Computation

2.1 Corner Detection

Complete the function *harrisCorners* which takes an image *src* as input argument and the parameter k in formula 5 is fixed as 0.04. This function calculates the value of R (c.f.: formula 5) for each point of the input image and stocks them in a matrix *harris*, which is a CImg object.

2.2 Classification of descriptors using K-means

Complete the function *kMeans* which has four input arguments:

1. *points*: a list of points. In our exercise, these points are the Harris corners detected in the last exercise
2. *pointsAssignment*: the initialization of the labeling of each point
3. *centers*: the initialization of the list of all the clusters' centers
4. *centerN*: the number of clusters

The objective of this function is to update *pointsAssignment* and *centers* using the K-means algorithm introduced in Section 1.2.

References

[1] C. Harris and M.J. Stephens. A combined corner and edge detector. In Alvey Vision Conference, pages 147-152, 1988.

[2] C. Schmid, R. Mohr, and C. Bauckhage. Evaluation of interest point detectors. International Journal of Computer Vision, 37(2):151-172, June 2000.