An Investigation of Methods for Improving Scale Invariant Feature Transform (SIFT)

Shruti Appiah, Mira Sleiman
University of Waterloo

Abstract

Content Based Image Retrieval (CBIR) is a novel method for extracting like images from a large image repository, given a single input image. Google's VisualRank uses CBIR in conjunction with PageRank, a graph centrality-based ranking algorithm. This paper explores how Google's Image Search algorithm detects local features using Scale-Invariant Feature Transform and demonstrates a method for improving the efficiency and accuracy of the SIFT, i.e. keypoint localization, through a case study.

Background

A traditional Google Search for images returns relevant images based on a comparison of their attributes, metadata, tags, and authority score[1][1]. The images are sorted using Google's PageRank which relies solely on the text associated with images, so none of the image contents themselves are used for searching or ranking. Researchers at Google and Georgia Institute of Technology identified an opportunity to instead use the visual content of images as a basis for searching for and determining relevance of images. Content-Based Image Retrieval (CBIR) has since been applied in a variety of fields such as art, intellectual property rights, visual content censorship, and Reverse Image Search.

Relevance of Linear Algebra to the Analysis

The entire image database of the world wide web can be modelled as a graph. It forms a scale-free network, given that it is complex, interconnected, and follows power laws. Consider a graph G with vertices/nodes V and edges/connections E.

$$G\left(V, E\right)$$

While searching for an image, Google's VisualRank algorithm collects a subset of images by comparing the text-based cues such as metadata, tags, attributes etc. Images belonging to this subset are then compared with the original input image to determine their corresponding similarities. Locality Sensitive Hashing (LSH) is a type of Nearest Neighbour Search algorithm that offers a relatively quick way to obtain image-to-image similarity matrices[1]. It performs dimensionality reduction on the original subset by hashing each image and classifying them into buckets containing similar items. The Scale Invariant Feature Transform (SIFT) algorithm is employed to find the visual similarity between two images[2]. Once the similarity matrices for all images has been obtained, a visual similarity graph is created. This graph has the most relevant and highly ranked images at the centre and has the less relevant images along the periphery of the graph. The PageRank algorithm runs on the graph to determine the *centrality* of its constituent images. Centrality is a measure of a node's influence in a graph[1]. Keeping all factors constant, a node close to an influential node has a higher rank than those further away [1]. Eigenvector Centrality can be described as:

$$EVcentrality(i) \propto \sum_{j} A_{ij} EVcentrality(j)$$

Where A_{ij} is an element of the adjacency matrix[3]. Therefore, A_{ij} = 1 if *i* and *j* share an edge; otherwise, A_{ij} = 0. Google's PageRank uses an improved version of Katz Centrality, which is based on Eigenvector Centrality [3]. All of them compute the sum of the geodesic distances from the node under consideration to all the others. Consider an input image *u* being compared to an image on the web, *v*. Images with a high centrality are given a higher ranking by VisualRank. VisualRank is defined as:

[1] The authority score of a node is proportional to the number of edges linking to it.

$$VR = S^* \times VR$$

S^* is the similarity matrix of the image, where $S_{u,v}$ measures the visual similarity between image *u* and *v*[1]. The S* matrix is multiplied repeatedly with VR to obtain the dominant eigenvector of the S* matrix.

Before images can be analyzed, they are first transformed into a collection of vectors. Key points, that usually lie in high-contrast regions of the image, are recorded and stored in a database[4]. Sometimes however, unstable points are falsely recorded as significant. In order to reject low contrast points which can be sensitive to noise, the algorithm performs keypoint localization. This interpolates the keypoint's nearby data to determine the point's exact location by using Taylor expansion of the scale-space function, D(x,y,σ) shifted so that the origin is at the key point. This expansion is shown below:

$$D(x) = D + \frac{\delta\ D^T}{\delta\ x} + \frac{1}{2} x^T \frac{\delta^2 D}{\delta\ x^2} x$$

Where D and its derivatives are evaluated at the sample point and $x=(x,y,\sigma)^T$ is the offset from the point[4]. In addition, the key points' location can be found by taking the derivative of this function with respect to x and setting it to zero. This yields,

$$\hat{x} = -\frac{\delta^2 D^{-1}}{\delta x^2} \frac{\delta D}{\delta x}$$

If the offset $\hat{x}$ is larger than 0.5 in any dimension, it implies that the key point lies closer to another sample point. The sample point should then be changed and the interpolation should be performed about the new point.

In addition, another way to improve the feature detection accuracy is through eliminating unstable or insignificant keypoints. These include low contrast and edge points which can be identified and removed through Harris Corner Detection[5]. This algorithm detects patches of the image which cause a large variation when moved. An ideal patch can be described as having a maximum shift when subjected to any kind of change. Such patches probably contain a corner. The presence of an edge, a corner, or no interesting feature can be determined based on the eigenvalues of the Harris matrix.

Case Study and Analysis

The SIFT algorithm accommodates manipulations on the image such as scaling, orientation or pose, noise, and illumination. The SIFT algorithm begins by finding image keypoints and storing them as feature vectors. These are points of high-contrast that fall along the edges of objects and thus can be used to describe the different features in an image. These vectors can be used to compare and match the features of two images. This is done by calculating the Euclidean distance between the corresponding keypoints in the two images. From this, the ratio of the distance from the closest neighbour to the distance from the second closest neighbour is obtained. Larger distance ratios mean that the features are dissimilar, and therefore such images are eliminated.

To account for variations in the orientation and position of objects in the image, a Hough Transform is performed. Hough transforms are used to detect the shapes present in the image using their representative equations. Similar feature clusters are identified and their commonly voted object interpretation is taken as being the truth. This sets a basis for feature extraction from the image. To verify scale and orientation, a process involving the least squares approach is then performed for the parameters of the affine transformation thus relating the model to the image. The affine transformation of a model point $[x,y]^T$ to an image point $[u,v]^T$ is shown as:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

1. The authority score of a node is proportional to the number of edges linking to it.

Where model translation is $[t_x, t_y]^T$ and the m_1, m_2, m_3, and m_4 parameters represent the affine rotation, stretch, and scale. The solution of the system of linear equations can then be given in terms of:

$$x = (A^TA)^{-1}A^Tb$$

where,

$$(A^TA)^{-1}A^T$$

is the pseudoinverse of matrix A.

Mathematical Solution

The SIFT keypoint detection algorithm identified and plotted the keypoints of high contrast in the image. For searching by image, these keypoints would be compared to the keypoints in other images through the ratios of their Euclidean distances in order to determine if the objects within the image are the same. For a given image, certain parameters were also gradually varied one by one while keeping all others constant to determine the mathematical sensitivity of the system.

Results and Interpretation

The SIFT algorithm performs keypoint detection and forms a feature vector of the image consisting of the key points. Since a significant amount of the detected keypoints do not mark the edges of objects, and thus do not contribute to feature extraction, they are removed through keypoint localization. The resulting feature vector is much smaller and less noisy. It was observed that the keypoints detected successfully in a high contrast image were able to identify features present in the image.

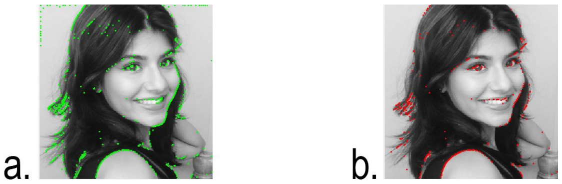

Figure 1: Comparing Keypoint Detection Before and After Keypoint Localization. a. Before Keypoint Localization. b. After Keypoint Localization

It can also be seen that as the contrast of the image is decreased, the algorithm cannot make meaningful detections. After decreasing the opacity, the increasing inaccuracy in keypoint detection leads the algorithm to find fewer points that lead to feature detection.

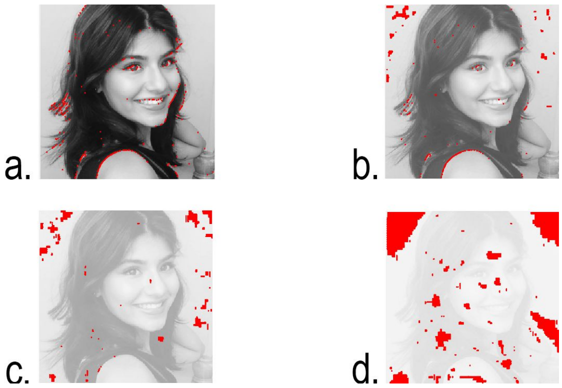

Figure 2: Comparing Point Detections with Various Opacities. a. High Opacity Leading to Accurate Detections (with a 1.56 s processing time). b. Lower Opacity Leading to Less Accurate Point Detections (with a 3.8 s processing time). c. Much Lower Opacity Leading to Even Less Accurate Detections (with a 5.05 s processing time). d. Lowest Opacity Causing Inaccurate Detections (with a 34.8 s processing time).

Conclusion

Using images as a basis for searching through Google content is a fascinating process that applies a lot of underlying concepts of Linear Algebra. Overall, through the investigation of the SIFT algorithm, it was evident that varying the image contrast drastically improved the performance of keypoint localization, thus improving the accuracy and efficiency of feature detection.

References

[1] Jing, Y., & Baluja, S. (2008). VisualRank: Applying PageRank to Large-Scale Image Search. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, 30*(11). Retrieved from http://www.kevinjing.com/jing_pami.pdf

[2] Jing, Y., & Baluja, S. (2008). PageRank for Product Image Search. Retrieved from: http://www.www2008.org/papers/pdf/p307-jingA.pdf

[3] Arratia, A. & Ferrer-i-Cancho, R. (2016) Centrality. Universitat Polit`ecnica de Catalunya, Retrieved from: https://www.cs.upc.edu/~CSN/slides/06centrality.pdf

[4] Lowe, D. Object Recognition from Local Scale-Invariant Features. University of British Columbia. Retrieved from: http://www-inst.eecs.berkeley.edu/~cs294-6/fa06/papers/LoweD_Object%20recognition%20from%20local%20scale-invariant%20features.pdf

[5] Harris, C. and Stephens, M. (1998) A Combined Corner And Edge Detector. Plessey Research Roke Manor. Retrieved from: http://www.bmva.org/bmvc/1988/avc-88-023.pdf

[1] The authority score of a node is proportional to the number of edges linking to it.