# Reverse Image Search Techniques

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

We will be presenting the method for content-based image retrieval (CBIR) query technique that involves providing the CBIR system with a sample image that it will then base its search upon. In terms of information retrieval, the sample image is what formulates a search query.

A combination of the following techniques. Companies like google rank the results using a proprietary algorithm. The following describe some of the approaches to achieve this goal:

**Feature Detection (Image fingerprinting to look for exact match):**

Three commonly used feature detection algorithms for matching image deformation such as blur, rotation, scale, and illumination change.

These are SIFT, PCA-SIFT and SURF.

**SIFT** - **Scale-Invariant Feature Transform (SIFT)**

SIFT keypoints of objects are first extracted from a set of reference images and stored in a database. An object is recognized in a new image by individually comparing each feature from the new image to this database and finding candidate matching features based on Euclidean distance of their feature vectors. From the full set of matches, subsets of keypoints that agree on the object and its location, scale, and orientation in the new image are identified to filter out good matches.

The determination of consistent clusters is performed rapidly by using an efficient hash table implementation of the generalized Hough transform. Each cluster of 3 or more features that agree on an object and its pose is then subject to further detailed model verification and subsequently outliers are discarded. **Discarding of outliers is an important feature of this method.**

Finally, the probability that a set of features indicates the presence of an object is computed, given the accuracy of fit and number of probable false matches. Object matches that pass all these tests can be identified as correct with high confidence.

**The key stages are:** Scale-invariant feature detection, Feature matching and indexing, Cluster identification by Hough transform voting, Model verification by linear least squares, Outlier detection

**Working (taken from Object Recognition from Local Scale-Invariant Features** by **David G. Lowe):**

This approach transforms an image into a large collection of local feature vectors, each of which is invariant to image translation, scaling, and rotation, and partially invariant to illumination changes and afﬁne or 3D projection.

The scale-invariant features are efﬁciently identiﬁed by using a staged ﬁltering approach. The ﬁrst stage identiﬁes key locations in scale space by looking for locations that are maxima or minima of a difference-of-Gaussian function. Each point is used to generate a feature vector that describes the local image region sampled relative to its scale-space coordinate frame. The features achieve partial invariance to local variations, such as afﬁne or 3D projections, by blurring image gradient locations. The resulting feature vectors are called SIFT keys which are in turn used in a nearest-neighbor approach to indexing for identifying candidate object models. Collections of keys that agree on a potential model pose are ﬁrst identiﬁed through a **Hough transform hash table**, and then through a **least-squares ﬁt** to a ﬁnal estimate of model parameters. When at least **3 keys** agree on the model parameters with low residual, there is strong evidence for the presence of the object. Since there may be dozens of SIFT keys in the image of a typical object, it is possible to have substantial levels of occlusion in the image and yet retain high levels of reliability. The current object models are represented as 2D locations of SIFT keys that can undergo afﬁne projection. Sufﬁcient variation in feature location is allowed to recognize perspective projection of planar shapes at up to a 60-degree rotation away from the camera or to allow up to a 20-degree rotation of a 3D object.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**SURF:** **Speeded Up Robust Features (SURF)**

It is a patented local feature detector and descriptor. It can be used for tasks such as object recognition, image registration, classification or 3D reconstruction. The standard version of SURF is many times faster than SIFT and claimed by its authors to be more robust against different image transformations than SIFT.

To detect interest points, SURF uses an **integer approximation** of the determinant of Hessian blob detector, which can be computed with 3 integer operations using a precomputed **integral image**. Its feature descriptor is based on the sum of the **Haar wavelet response** around the point of interest. These can also be computed with the aid of the integral image.

The SURF algorithm is based on the same principles and steps as SIFT, but details in each step are different. The algorithm has three main parts: interest point detection, local neighborhood description and matching.

**Working (Speeded-Up Robust Features (SURF)** by **Herbert Bay, Andreas Ess, Tinne Tuytelaars and Luc Van Gool)**

The search for discrete image point correspondences can be divided into three main steps.

First, ‘**interest points**’ are selected at distinctive locations in the image, such as corners, blobs, and T-junctions. The most valuable property of an interest point detector is its repeatability which is the reliability of a detector for finding the same physical interest points under different viewing conditions.

Next, the neighborhood of every interest point is represented by a **feature vector**. This descriptor has to be  
distinctive and at the same time robust to noise, detection displacements and geometric and photometric deformations.

Finally, the descriptor vectors are **matched** between different images. The matching is based on a distance between the vectors, e.g. the Mahalanobis or Euclidean distance. The dimension of the descriptor has a direct impact on the time this takes, and less dimensions are desirable for fast interest point matching.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**PCA-SIFT: Principal Components Analysis applied to SIFT descriptors**

PCA is a standard technique for dimensionality reduction, which is well-suited to represent the keypoint patches and enables us to linearly-project high-dimensional samples into a low-dimensional feature space.

In other words, **PCA-SIFT uses PCA instead of histogram** to normalize gradient patch. The feature vector is significantly smaller than the standard SIFT feature vector, and it can be used with the same matching algorithms. PCA-SIFT, like SIFT, also used Euclidean distance to determine whether the two vectors correspond to the same keypoint in different images. In PCA-SIFT, the input vector is created by concatenation of the horizontal and vertical gradient maps for the **41x41 patch** centered to the keypoint, which has **2x39x39=3042** elements. According to PCA-SIFT, fewer components requires less storage and will be resulting to a faster matching, they choose the dimensionality of the feature space, n = 20. This results to significant space (and hence memory usage) benefits.

-International Journal of Image Processing Vol.3, Issue 4

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*There is no perfect algorithm, so the choice depends mainly on the application, available hardware and what kind of trade-offs the application can tolerate.*