Geometric Hashing

Giorgos Tolias

Image, Video and Multimedia Systems Laboratory
National Technical University of Athens

December 2010
Outline

Geometric Hashing

Geometric Hashing with Local Affine Frames

Geometric Min-Hash

Feature Map Hashing
Outline

Geometric Hashing

Geometric Hashing with Local Affine Frames

Geometric Min-Hash

Feature Map Hashing
Scene features—including such things as points, linear and curvilinear segments, and corners—are accumulated during the feature extraction stage. The collection of features is represented by a set of dots, each dot representing a feature's location. Associated with each dot is a list of one or more attributes, depending on the feature's type.

Suppose we wish to perform recognition of patterns of point features in the presence of similarity transformations—that is, the point features may be translated, rotated, or scaled. (Geometric hashing can tackle other transformations, such as rigid and affine transformations, but similarity transformations are of moderate difficulty and effectively showcase the methodology.)

The left side of Figure 1 shows model $M_1$, which consists of five dots with position vectors $p_1, p_2, p_3, p_4, p_5$. We want to encode this dot information appropriately and store it into a table. This way, if the system detects this collection of dots in a scene, it could conclude that they belong to the model $M_1$.

If we assume for the moment that each dot has a unique, distinctive color, a potential albeit simplistic indexing scheme would use the color as the dot's index: an entry in the hash bin would include the identity of the model to which the dot belongs. In the recognition stage, the system would simply scan the dots, access the hash table using each dot's color, and increase the count of the models appearing in the accessed table bins. Models accumulating high counts have high probability to be present in the scene.

The computational complexity of such a scheme would be linear in the number of the scene dots. However, what happens in the least informative case, where dots belonging to a model have no attributes except for their geometric configuration? Is there a distinctive geometric "color"? Yes, the natural geometric "color" of a dot is the set of its coordinates, but coordinates depend on a reference frame. The question then becomes one of whether there is a natural reference frame for a model that will remain present under partial occlusion. One straightforward such choice.
Introduction

We need a method that allows direct access to only the relevant information—such as an indexing-based approach.

The model information is encoded in a pre-processing step and stored in a large memory, in this case a hash table. The contents of the hash table are independent of the scene and can thus be computed offline, not affecting the recognition time. Access to the memory is based on geometric information that is invariant of the object’s pose and computed directly from the scene.
Scene features—including such things as points, linear and curvilinear segments, and corners—are accumulated during the feature extraction stage. The collection of features is represented by a set of dots, each dot representing a feature's location. Associated with each dot is a list of one or more attributes, depending on the feature's type.

Suppose we wish to perform recognition of patterns of point features in the presence of similarity transformations—that is, the point features may be translated, rotated, or scaled. (Geometric hashing can tackle other transformations, such as rigid and affine transformations, but similarity transformations are of moderate difficulty and effectively showcase the methodology.) The left side of Figure 1 shows model $M_1$, which consists of five dots with position vectors $p_1, p_2, p_3, p_4, p_5$. We want to encode this dot information appropriately and store it into a table. This way, if the system detects this collection of dots in a scene, it could conclude that they belong to the model $M_1$.

If we assume for the moment that each dot has a unique, distinctive color, a potential albeit simplistic indexing scheme would use the color as the dot's index: an entry in the hash bin would include the identity of the model to which the dot belongs. In the recognition stage, the system would simply scan the dots, access the hash table using each dot's color, and increase the count of the models appearing in the accessed table bins. Models accumulating high counts have high probability to be present in the scene. The computational complexity of such a scheme would be linear in the number of the scene dots. However, what happens in the least informative case, where dots belonging to a model have no attributes except for their geometric configuration? Is there a distinctive geometric “color”? Yes, the natural geometric “color” of a dot is the set of its coordinates, but coordinates depend on a reference frame. The question then becomes one of whether there is a natural reference frame for a model that will remain present under partial occlusion. One straightforward such choice.

![Diagram](image-url)
Let’s take the pair of dots \( p_4, p_1 \) as an ordered basis to such a reference frame.

we scale the model \( M_1 \) so that the magnitude of \( \overrightarrow{p_4p_1} \) in the \( Ox y \) system equals 1. Suppose now that we place the midpoint between dots 4 and 1 at the origin of a coordinate system \( Ox y \) in such a way that the vector \( \overrightarrow{p_4p_1} \) has the direction of the positive \( x \) axis. The remaining three points of \( M_1 \) will land in three locations. Let’s record in a quantized hash table—in each of the three bins where the remaining points land—the fact that model \( M_1 \) with basis “(4, 1)” yields an entry in this bin.

Since our goal is to perform recognition under partial occlusion, we are not guaranteed that both basis points \( p_1 \) and \( p_4 \) will appear in each scene where model \( M_1 \) will be present. Consequently, we encode the model dot’s information in all possible ordered basis pairs.

Reference frame
Offline process

Preprocessing phase
For each model $m$ do the following:
1. Extract the model’s point features. Assume that $n$ such features are found.
2. For each ordered pair, or basis, of point features do the following:
   (a) Compute the coordinates $(u, v)$ of the remaining features in the coordinate frame defined by the basis.
   (b) After proper quantization, use the tuple $(u_q, v_q)$ as an index into a 2D hash table data structure and insert in the corresponding hash table bin the information $(m, (basis))$, namely the model number and the basis tuple used to determine $(u_q, v_q)$. 

Recognition phase
4. Appropriately quantize each such coordinate and access the appropriate hash table bin; for every entry found there, cast a vote for the model and the basis points. In the case where model points are missing from the image because they are occluded, recognition is still possible as long as a sufficient number of points hash to the correct basis.
5. Histogram all hash table entries that received one or more votes during step 4. Proceed to determine those entries that received more than a certain number, or threshold, of votes: Each such entry corresponds to a potential match.
6. For each potential match discovered in step 5, recover the transformation $T$ that results in the best least-squares match between all corresponding feature pairs.
7. Transform the features of the model according to the recovered transformation $T$ and verify them against the input image features. If the verification is successful it is sufficient to select as a basis tuple any basis (model, basis) combinations, then a subsequent docking in molecular biology (as discussed earlier).

Offline process
Online process

Recognition phase
When presented with an input image, do the following:
1. Extract the various points of interest. Assume that S is the set of the interest points found; let S be the cardinality of S.
2. Choose an arbitrary ordered pair, or basis, of interest points in the image.
3. Compute the coordinates of the remaining interest points in the coordinate system Oxy that the selected basis defines.
4. Appropriately quantize each such coordinate and access the appropriate hash table bin; for every entry found there, cast a vote for the model and the basis.
5. Histogram all hash table entries that received one or more votes during step 4. Proceed to determine those entries that received more than a certain number, or threshold, of votes: Each such entry corresponds to a potential match.
6. For each potential match discovered in step 5, recover the transformation T that results in the best least-squares match between all corresponding feature pairs.
7. Transform the features of the model according to the recovered transformation T and verify them against the input image features. If the verification fails, go back to step 2 and repeat the procedure using a different image basis pair.
Scene features—including such things as points, linear and curvilinear segments, and corners—are accumulated during the feature extraction stage. The collection of features is represented by a set of dots, each dot representing a feature’s location. Associated with each dot is a list of one or more attributes, depending on the feature’s type.

Suppose we wish to perform recognition of patterns of point features in the presence of similarity transformations—that is, the point features may be translated, rotated, or scaled. (Geometric hashing can tackle other transformations, such as rigid and affine transformations, but similarity transformations are of moderate difficulty and effectively showcase the methodology.) The left side of Figure 1 shows model $M_1$, which consists of five dots with position vectors $p_1, p_2, p_3, p_4, p_5$. We want to encode this dot information appropriately and store it into a table. This way, if the system detects this collection of dots in a scene, it could conclude that they belong to the model $M_1$.

If we assume for the moment that each dot has a unique, distinctive color, a potential albeit simplistic indexing scheme would use the color as the dot’s index: an entry in the hash bin would include the identity of the model to which the dot belongs. In the recognition stage, the system would simply scan the dots, access the hash table using each dot’s color, and increase the count of the models appearing in the accessed table bins. Models accumulating high counts have high probability to be present in the scene. The computational complexity of such a scheme would be linear in the number of the scene dots. However, what happens in the least informative case, where dots belonging to a model have no attributes except for their geometric configuration? Is there a distinctive geometric “color”? Yes, the natural geometric “color” of a dot is the set of its coordinates, but coordinates depend on a reference frame. The question then becomes one of whether there is a natural reference frame for a model that will remain present under partial occlusion. One straightforward such choice.

Figure 1. Determining the hash table entries when points 4 and 1 are used to define a basis. The models are allowed to undergo rotation, translation, and scaling. On the left of the figure, model $M_1$ comprises five points.

Figure 2. The locations of the hash table entries for model $M_1$. Each entry is labeled with the information “model $M_1$” and the basis pair $(i, j)$ used to generate the entry. The models are allowed to undergo rotation, translation, and scaling.
Voting example

In the diagram:
- The model points are represented in a 2D coordinate system.
- Each model point casts a vote in the hash table.
- In the end, all entries with one or more votes are counted in the histogram.
Invariant reference frames

1. *Translation in 2D*: The technique is applicable using a one-point basis, the point being viewed as the origin of the coordinate frame.

2. *Translation and rotation in 2D*: A two-point basis can be used, but one point with a direction (obtained, say, from an edge segment) provides enough information for a unique definition of a basis.


4. *Affine transformation in 2D*: A three-point basis defines an unambiguous reference frame.\(^3\)\(^5\)

5. *Projective transformation in 2D*: A four-point basis is needed to recover a projective transformation between two planes.
Outline

Geometric Hashing

Geometric Hashing with Local Affine Frames

Geometric Min-Hash

Feature Map Hashing
Geometric Hashing with Local Affine Frames

In many cases, the anisotropy factor of a LAF is not only dependent on the viewpoint, but mainly on the shape of the viewpoint. However, this approach is not affine-invariant and is very sensitive to scale changes.

In our approach, the following additional criterion is used: the angle is discretized into 25 bins and the distance into 16 bins. The remaining four dimensions are given by the coordinates of the 6D transformation. Let $L^0$ be the first image and $L_0$ the reference frame and the description of a distance from the origin of the reference frame in the RF.

For wide-baseline stereo matching, we propose six RF constructions, depicted in Fig. 3. Then, there are six hashing tables for each type of matching. For wide-baseline implementation, we have chosen the coordinates of the 6D normalization as description. Since the affine variant description.

Let $L^1$ be the first image and $L_0$ the reference frame. In our experiments set to the value of 4, are used for affine variant description.

Widespread range of off-plane rotation of the viewpoint, but mainly on the viewpoint. However, this approach is not affine-invariant and is very sensitive to scale changes.

In our approach, the following additional criterion is used: the angle is discretized into 25 bins and the distance into 16 bins. The remaining four dimensions are given by the coordinates of the 6D transformation. Let $L^0$ be the first image and $L_0$ the reference frame and the description of a distance from the origin of the reference frame in the RF.

In our experiments set to the value of 4, are used for affine variant description.

Wide range of off-plane rotation of the viewpoint, but mainly on the viewpoint. However, this approach is not affine-invariant and is very sensitive to scale changes.

In our approach, the following additional criterion is used: the angle is discretized into 25 bins and the distance into 16 bins. The remaining four dimensions are given by the coordinates of the 6D transformation. Let $L^0$ be the first image and $L_0$ the reference frame and the description of a distance from the origin of the reference frame in the RF.
Local affine frames

By a LAF we understand an ordered triplet of non-collinear points \( L = (x_1, x_2, x_3) \), where \( x_i = (x, y, 1)^\top \). Let \( L^I = (n_1, n_2, n_3) \), where \( n_1 = (1, 0, 1)^\top \), \( n_2 = (0, 0, 1)^\top \), and \( n_3 = (0, 1, 1)^\top \), be a canonical LAF. Let normalization \( N \) be an affine transformation that transforms LAF \( L \) to a canonical frame \( L^I = NL \). Let \( A \) be a matrix representing an affine transformation with the last row \((0, 0, 1)\); and let \( UDV^\top \) be a SVD decomposition of the upper left 2×2 submatrix of \( A \). Let \( D = \text{diag}(d_1, d_2) \), where \( d_1 \geq d_2 \). We define anisotropy factor of an affine transformation \( A \) as \( a(A) = d_1/d_2 \). The anisotropy factor of LAF is the anisotropy factor of its normalization.
**Normalization**

In many cases, the anisotropy factor of a descriptor is not only dependent on the viewpoint, but mainly on the shape of the descriptor itself (crops close to the image). How ever, this approach is only valid for a locally valid real image transformation, the descriptor as polar coordinates of the two remaining points from the description.

There is a number of possible representations of the 6D coordinates of the central point of the description.

The votes are counted in a sparse matrix, represented as a hashing table [5]. The collisions were handled by a secondary hashing function. Even if the same pairs of LAFs appear in an identical bin for more than a single construction of reference frame the vote is counted only once.

The angle is discretized into 25 bins and the distance into 16 bins. The remaining four dimensions are given by the polar coordinates of the two remaining points from the description.

The axis form an angle of almost 0 or 180 degrees. The collisions often cause failure in the first step of the matching process.

We assume a number of independent and identical LAF coordinates systems defined by such description points. The anisotropy factor of the transform is 1.3.

In many cases, the anisotropy factor of a descriptor is significantly higher than the anisotropy factor of the transform.

In our approach, the following additional criterion is used: The restrictions on the image transformation are of- ten made. Wide range of of-plane rotation of the viewpoint, but mainly on the the shape of the anisotropy factor are more sensitive to noise. Only those positional error). The circles are transformed to elliptical uncertainty regions in the normalized frame. A reference positional error). The circles are transformed to elliptical uncertainty regions of locations of points forming LA F associated with a RF construction, RF in Fig. 3. Then, there are six hashing tables for each type of stereo matching, we propose six RF constructions, depicted in Fig. 2. Normalization of a LA F is not only dependent on the viewpoint, but mainly on the the shape of the uncertainty regions of locations of points forming LA F.

We assume a number of independent and identical LAF coordinates systems defined by such description points. The anisotropy factor of the transform is 1.3.

In many cases, the anisotropy factor of a descriptor is significantly higher than the anisotropy factor of the transform.

In our approach, the following additional criterion is used: The restrictions on the image transformation are of- ten made. Wide range of of-plane rotation of the viewpoint, but mainly on the the shape of the anisotropy factor are more sensitive to noise. Only those positional error). The circles are transformed to elliptical uncertainty regions in the normalized frame. A reference positional error). The circles are transformed to elliptical uncertainty regions of locations of points forming LA F.

We assume a number of independent and identical LAF coordinates systems defined by such description points. The anisotropy factor of the transform is 1.3.

In many cases, the anisotropy factor of a descriptor is significantly higher than the anisotropy factor of the transform.

In our approach, the following additional criterion is used: The restrictions on the image transformation are of- ten made. Wide range of of-plane rotation of the viewpoint, but mainly on the the shape of the anisotropy factor are more sensitive to noise. Only those positional error). The circles are transformed to elliptical uncertainty regions in the normalized frame. A reference positional error). The circles are transformed to elliptical uncertainty regions of locations of points forming LA F.

We assume a number of independent and identical LAF coordinates systems defined by such description points. The anisotropy factor of the transform is 1.3.

In many cases, the anisotropy factor of a descriptor is significantly higher than the anisotropy factor of the transform.

In our approach, the following additional criterion is used: The restrictions on the image transformation are of- ten made. Wide range of of-plane rotation of the viewpoint, but mainly on the the shape of the anisotropy factor are more sensitive to noise. Only those positional error). The circles are transformed to elliptical uncertainty regions in the normalized frame. A reference positional error). The circles are transformed to elliptical uncertainty regions of locations of points forming LA F.
The affine invariant descriptor constructed in this paper is derived from the mutual position of two LAFs: one, called *reference frame* (RF), provides a coordinate system; the other, called *description frame* (DF)
Reference frames

We propose to use more than a single reference frame associated with a LAF. The restrictions on the image transformation can be then applied to the RFs. For wide-baseline stereo matching, we propose six RF constructions, depicted in Fig. 3. Then, there are six hashing tables for each type of RF construction, i.e. descriptors originated from the same type of RF construction can be matched. Not all six RFs are used for every frame, since the RFs with high value of the anisotropy factor are more sensitive to noise. Only those RFs having the anisotropy factor smaller than a threshold, in our experiments set to the value of 4, are used for affine invariant description.
Reference frames – example

4.00  1.33  2.16  3.42  2.16  3.42
Selection constraints

Since the affine transformation is only a locally valid approximation of the real image transformation, the description LAFs are selected in proximity of the reference frame (in our case measured as a distance from the origin of the reference frame in the RF coordinate system). If there were no other selection criteria, large RFs would use a large number of (even all) frames as description LAFs (leading to a computational explosion). In our approach, the following additional criterion is used: the reference frame and the description LAF are of similar scale.
Discretization

There is a number of possible representations of the 6D affine invariant descriptor. Let $L_1, L_2$ be two LAFs in the first image and $L'_1, L'_2$ in the second image. We express points of $L_2$ in a coordinate system derived from $L_1$. In our implementation, we have chosen the coordinates of the 6D descriptor as polar coordinates of points of the description LAF $L_2$. The first two dimensions are set as polar coordinates of the central point of the description LAF, having the origin of the coordinate system at the origin of the RF. The angle is discretized into 25 bins and the distance into 16 bins. The remaining four dimensions are given by the polar coordinates of the two remaining points from the description LAF. The origin of the polar coordinate system is located at the central point of the description LAF. The angles are discretized into 25 and the radii into 6 bins. This gives $9 \cdot 10^6$ possible values of the descriptor.
Matching example

Figure 1. Detecting a logo (central image). Successful detection in images 1-8 is highlighted by a quadrilateral. Note the differences in appearance. Bottom row: cut-outs from images 6, 7, 8, and 3.
Figure 5. The Graffiti experiment. Left image (a) with superimposed 88 LAFs that were detected in both images. Right image (b) with superimposed boundary of the transformed left image. (c) The anisotropy factor of an affine transformation locally approximating the ground truth homography of the left-to-right image mapping.

We next assess how much is gained if we exploit the a priori knowledge that illumination did not change. In this case, LAFs originating from MSER+ and MSER- were not allowed to match. The number of correctly matched correspondences increased to 80, which is close to the possible maximum of 88. The light solid curve plotted in Fig. 6 shows the fraction of inliers among $n$ top-ranked tentative correspondences (ranked by the number of votes) when MSER+ and MSER- were matched independently. Finally, we assess the benefit of defining multiple reference frames for each LAF. The dashed curve of Fig. 6 shows the fraction of inliers among $n$ top-ranked tentative correspondences if only a single reference frame is used; MSER+ and MSER- were joined in this experiment. Only 32 correct correspondences were among the tentative correspondences formed. The introduction of multiple reference frames thus doubled both the number and density of correct tentative correspondences.

Plot (Fig. 5c) shows the spatial distribution of the anisotropy factor of an affine transformation locally approximating the projective transformation of the images. The values are plotted for pixels of the left image that are mapped into (have an image in) the right image.

The ground truth homography $H$ was approximated in each point $x_0$ by a first-order approximation (affine transformation $A$)

$$A = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} x_0.$$  

The anisotropy factor $a(A(H; x_0))$ reaches 4.3 in the rightmost part of the image, while the threshold on the anisotropy factor of the normalization transformation $a(N)$ of coordinate frames was set to 4. Note, that this does not mean that LAFs from this region cannot be matched. It only means that coordinate frames with the anisotropy factor $a(N)$ below 1.075 have no chance of being matched.

Arbitrary background - text on a transparency. A text printed on a transparent foil was captured on two different backgrounds. Some of the letters are detected in both images. An affine-invariant measurement region larger than the detected letter includes the background, which is never the same in the corresponding parts of the two images. A measurement region that covers the letter (or its part) only, is not discriminative, as most of the letter appears many times in the text. On the other hand, groups of letters and their mutual positions provide enough information for the matching – the two images were successfully registered, see Fig. 7 (right).

Epipolar geometry estimation. This experiment demonstrates the performance of the proposed method on a non-planar scene. The correct epipolar geometry was recovered.
Outline

Geometric Hashing

Geometric Hashing with Local Affine Frames

Geometric Min-Hash

Feature Map Hashing
Abstract

We propose a novel hashing scheme for image retrieval, clustering and automatic object discovery. Unlike commonly used bag-of-words approaches, the spatial extent of image features is exploited in our method. The geometric information is used both to construct repeatable hash keys and to increase the discriminability of the description. Each hash key combines visual appearance (visual words) with semi-local geometric information.

Compared with the state-of-the-art min-Hash, the proposed method has both higher recall (probability of collision for hashes on the same object) and lower false positive rates (random collisions). The advantages of Geometric min-Hashing approach are most pronounced in the presence of viewpoint and scale change, significant occlusion or small physical overlap of the viewing fields. We demonstrate the power of the proposed method on small object discovery in a large unordered collection of images and on a large scale image clustering problem.

1. Introduction

Algorithms based on hashing techniques are the core of methods that have produced impressive results for a range of computer vision problems, like matching of point sets [14], object recognition [8], image retrieval [27], duplicate detection and clustering in large image collections [5].

In the paper, we propose a novel hashing scheme – the Geometric min-Hash (GmH). The advantages of the Geometric min-Hashing have high impact in problems involving significant occlusion or small physical overlap of the viewing fields. In such cases the difference in recall and precision reach orders of magnitude compared to min-Hash [7] algorithm. Moreover, GmH is less sensitive to scale changes. The advantages of the min-Hash, e.g. compact representation and robustness, are preserved. The potential of the method is demonstrated on small object discovery in a large unordered collection of images (see Fig. 1).

The min-Hash describes images by selecting independently visual words as global descriptors, with the property that the higher number of common features in two images, the higher the probability of having the same min-Hash. A single min-Hash is not sufficiently discriminative to support indexing and thus min-Hashes must be grouped into sketches for hashing. In order to find a pair of positive samples, the sketches are compared with the features extracted from the image.

Figure 1. An automatically discovered object in an unordered collection of 100k images. The lamp is visible in approximately 0.014% of the images and it covers on average about 0.28% of pixels of those images. Top: the 'seed' image pair found by the Geometric min-Hash. The three close-ups show the colliding sketch, its geometric support and the co-segmentations, respectively. Bottom: other detections (with close-ups) obtained by object retrieval using the seed pair. Note that all bounding boxes are discovered, not drawn by the user.
Motivation. Geometric min-Hashing improves on the min-Hash by achieving higher recall, higher precision (lower false positive rate) simultaneously. The algorithmic change is motivated by the following observations:

Observation 1 (on uniqueness): with large vocabularies (such as 1M visual words), visual words in images usually do not appear more than once. Selecting a visual word from an image is typically equivalent to selecting a feature (visual word with spatial location and extent).

Observation 2 (on repeatability): s-tuples of features localized in space and scale have (much) higher repeatability than random s-tuples.


## Algorithm

1. Select a set $F$ of image features that have a unique visual word within the image and have at least $v = 3$ features of similar scale in their neighbourhood. The set $F$ is used for all sketches.
2. Use a random min-Hash function to select a central feature (= visual word) from $F$.
3. Find a set of features $N$ that are (a) no closer than $d_{\text{min}}$ and no further than $d_{\text{max}}$ from the central feature, (b) the relative scale change to the central feature is not smaller than $c_{\text{min}}$ and not larger than $c_{\text{max}}$, and (c) no other feature with the same visual word satisfies the first two conditions.
4. Select secondary feature(s) using independent random min-Hash function(s) from features in $N$. 


Sketch example

The probability of two images having the same min-Hash is equal to their set overlap, that the probability of two sets having the same value of the function of the intersection and union of their set representations.

A min-Hash is a binary information (present or absent). A min-Hash is a function that assigns a number to each set of visual words since word frequency information is reduced into a words. This is a weaker representation than a bag of visual words in the sketch are selected from affine covariant neighbour-hoods (highlighted in the images) of the central feature using in-
ter dependent min-Hash functions. The secondary min-Hash selects features in the sketch are selected from affine covariant neighbour-
hoods (highlighted in the images) of the central feature using min-Hash from all unique features in an image that is assigned to a particular visual feature. In other words, that there is usually at most one observed that for large visual vocabularies, for most feature there is a one-to-one mapping from visual word to a age retrieval as a connected component of related images. A cluster of images is defined as a set of images with related repeated structures. However, statistically the observation probability of having the same sketches. On the other hand, probability of having the same sketches. On the other hand, the min-Hash function in common and hence have high dependency min-Hash functions.

To estimate the word overlap of two images, multiple in-
dependent min-Hash functions are used. The fraction of the min-Hash functions that assigns an identical value to the two sets gives an unbiased estimate of the similarity of the two images. To efficiently retrieve images with high simi-
larity that not a single image pair is retrieved from a cluster (even as low as 3%) probability of retrieval, the probabil-
ity of retrieval is increased by repeating the ran-
dom selection of the min-Hash functions that assigns an identical value to the two images. To efficiently retrieve images with high simi-
larity that not a single image pair is retrieved from a cluster (even as low as 3%) probability of retrieval, the probabil-

The recall of min-Hash is increased by repeating the ran-

dom selection of the min-Hash functions that assigns an identical value to the two images. To efficiently retrieve images with high simi-
larity that not a single image pair is retrieved from a cluster (even as low as 3%) probability of retrieval, the probabil-

The probability of a pair of images having at least one repeated structures. However, statistically the observation probability of having the same sketches. On the other hand, probability of having the same sketches. On the other hand, the min-Hash function in common and hence have high dependency min-Hash functions.

To estimate the word overlap of two images, multiple in-
dependent min-Hash functions are used. The fraction of the min-Hash functions that assigns an identical value to the two sets gives an unbiased estimate of the similarity of the two images. To efficiently retrieve images with high simi-
larity that not a single image pair is retrieved from a cluster (even as low as 3%) probability of retrieval, the probabil-

In min-Hash, images are treated as sets of visual words. We numbers with unique visual word is number of features per image is.

In min-Hash, the sketch is simply created as a ordered s-tuple of independent min-Hashes. In the proposed Geo-

k weighted set overlap. Let

\[ ovr \] is always refer to the weighted version.

In the sequel, when speaking about set or word overlap, we shown to give better results than the plain set overlap (1).

One possible choice, inspired by text retrieval, is the tf-idf weighting term frequency – inverse document frequency) weighting.

The weights are equally important. It was shown in [7] that the similarity...
Parameters

There are four parameters involved in the procedure. Two parameters $d_{\text{min}}$ and $d_{\text{max}}$ governing the minimal and maximal distance of the secondary feature(s) from the central feature. To preserve affine invariant selection, the distance is measured as a Mahalanobis distance using the covariance matrix of the central feature. The values of the parameters are set to $d_{\text{min}} = 0$ and $d_{\text{max}} = 3$ in our experiments. The other two parameters $c_{\text{min}}$ and $c_{\text{max}}$ give minimal and maximal relative scale change of the secondary feature with respect to the central one. In our experiments, we set $c_{\text{max}} = 1/c_{\text{min}} = \sqrt{2}$. 

Results
Geometric min-Hashing: Finding a (Thick) Needle in a Haystack

Ondřej Chum, Michal Perd'och and Jiří Matas

CMP, Dept. of Cybernetics, Faculty of Elec. Eng., Czech Technical University in Prague
chum@cmp.felk.cvut.cz

Abstract
We propose a novel hashing scheme for image retrieval, clustering and automatic object discovery. Unlike commonly used bag-of-words approaches, the spatial extent of image features is exploited in our method. The geometric information is used both to construct repeatable hash keys and to increase the discriminability of the description. Each hash key combines visual appearance (visual words) with semi-local geometric information.

Compared with the state-of-the-art min-Hash, the proposed method has both higher recall (probability of collision for hashes on the same object) and lower false positive rates (random collisions). The advantages of Geometric min-Hashing approach are most pronounced in the presence of viewpoint and scale change, significant occlusion or small physical overlap of the viewing fields. We demonstrate the power of the proposed method on small object discovery in a large unordered collection of images and on a large scale image clustering problem.

1. Introduction
Algorithms based on hashing techniques are the core of methods that have produced impressive results for a range of computer vision problems, like matching of point sets [14], object recognition [8], image retrieval [27], duplicate detection and clustering in large image collections [5]. In the paper, we propose a novel hashing scheme – the Geometric min-Hash (GmH). The advantages of the Geometric min-Hashing have high impact in problems involving significant occlusion or small physical overlap of the viewing fields. In such cases the difference in recall and precision reach orders of magnitude compared to min-Hash [7] algorithm. Moreover, GmH is less sensitive to scale changes. The advantages of the min-Hash, e.g. compact representation and robustness, are preserved. The potential of the method is demonstrated on small object discovery in a large unordered collection of images (see Fig. 1).

The min-Hash describes images by selecting independently visual words as global descriptors, with the property that the higher number of common features in two images, the higher the probability of having the same min-Hash. A single min-Hash is not sufficiently discriminative to support indexing and thus min-Hashes must be grouped into sketches for hashing. In order to find a pair of poses...
Outline

Geometric Hashing

Geometric Hashing with Local Affine Frames

Geometric Min-Hash

Feature Map Hashing
Feature Map Hashing

Avrithis Tolias Kalantidis – ACM Multimedia 2010
Feature Map Hashing: Sub-linear Indexing of Appearance and Global Geometry
Local patches

- Each local feature is associated with an image patch $L$, which also represents an affine transform.
- The rectified patch $R_0$ is transformed to the patch via $L$.
- The patch is rectified back to $R_0$ via $L^{-1}$. 

![Diagram](image-url)
Single correspondence hypothesis

- A patch correspondence $L \leftrightarrow R$
- The transformation from one patch to the other is $RL^{-1}$
- Each correspondence provides a transformation hypothesis.
- Transformation hypotheses are now $O(n)$ and we can compute them all [Philbin et al. 2007]
Feature set rectification

- Rectify both feature sets by transformations $L^{-1}$ and $R^{-1}$, then compare
- Extrapolate each local transform to the entire image frame
- Rectify the entire set of features in advance
Feature set rectification

- Rectify both feature sets by transformations $L^{-1}$ and $R^{-1}$, then compare
- Extrapolate each local transform to the entire image frame
- Rectify the entire set of features in advance
Spatial quantization

- Encode positions in polar coordinates \((\rho, \theta)\)
- Quantize positions in the rectified frames
- Define spatial codebook \(\mathcal{U} \subseteq \mathbb{R}^2\) with \(|\mathcal{U}| = k_{\rho} \times k_{\theta} = k_u\) bins
Spatial quantization

- Encode positions in polar coordinates \((\rho, \theta)\)
- Quantize positions in the rectified frames
- Define spatial codebook \(\mathcal{U} \subseteq \mathbb{R}^2\) with \(|\mathcal{U}| = k_\rho \times k_\theta = k_u\) bins

\[\tilde{\rho} = 1, \tilde{\theta} = 11\]

\[k_\rho = 5, k_\theta = 12\]
Feature maps

- An image is represented by a local feature set $P$
- Define the joint (visual-spatial) codebook $\mathcal{W} = \mathcal{V} \times \mathcal{U}$ with $|\mathcal{W}| = k_v k_u = k$ bins
- To construct a feature map we rectify a feature set and assign rectified features to spatial bins and visual words
- There is a different map for each origin; represent each image with a feature map collection $F_P$
- Can be seen as a local descriptor encoding the global feature set rectified in a local coordinate frame

$$f_P(\hat{x}) = h_{\mathcal{W}} (P(\hat{x}))$$
Feature maps

- An image is represented by a local feature set $P$
- Define the joint (visual-spatial) codebook $\mathcal{W} = \mathcal{V} \times \mathcal{U}$ with $|\mathcal{W}| = k_v k_u = k$ bins
- To construct a feature map we rectify a feature set and assign rectified features to spatial bins and visual words
- There is a different map for each origin; represent each image with a feature map collection $F_P$
- Can be seen as a local descriptor encoding the global feature set rectified in a local coordinate frame

$$f_P(\hat{x}) = h_{\mathcal{W}}(P(\hat{x}))$$

feature map of $P$ wrt origin $\hat{x}$
Feature maps

- An image is represented by a local feature set $P$
- Define the joint (visual-spatial) codebook $\mathcal{W} = \mathcal{V} \times \mathcal{U}$ with $|\mathcal{W}| = k_v k_u = k$ bins
- To construct a feature map we rectify a feature set and assign rectified features to spatial bins and visual words
- There is a different map for each origin; represent each image with a feature map collection $F_P$
- Can be seen as a local descriptor encoding the global feature set rectified in a local coordinate frame

$f_P(\hat{x}) = h_\mathcal{W} \ (P(\hat{x}))$

- feature map of $P$ wrt origin $\hat{x}$
- rectified feature set $P$ wrt origin $\hat{x}$
Feature maps

- An image is represented by a local feature set $P$
- Define the joint (visual-spatial) codebook $\mathcal{W} = \mathcal{V} \times \mathcal{U}$ with $|\mathcal{W}| = k_v k_u = k$ bins
- To construct a feature map we rectify a feature set and assign rectified features to spatial bins and visual words
- There is a different map for each origin; represent each image with a feature map collection $F_P$
- Can be seen as a local descriptor encoding the global feature set rectified in a local coordinate frame

$$f_P(\hat{x}) = h_{\mathcal{W}}(P(\hat{x}))$$

- feature map of $P$ wrt origin $\hat{x}$
- rectified feature set $P$ wrt origin $\hat{x}$
- histogram wrt joint codebook $\mathcal{W}$
Well aligned feature sets are likely to have maps with a high degree of overlap.
Well aligned feature sets are likely to have maps with a high degree of overlap.
Feature map similarity (FMS)

\[
S_F(P, Q) = \max_{v \in V(P,Q)} \max_{\hat{x} \in H_v(P)} \max_{\hat{y} \in H_v(Q)} f^T_P(\hat{x}) f_Q(\hat{y})
\]
Feature map similarity (FMS)

\[ S_F(P, Q) = \max_{v \in V(P, Q)} \max_{\hat{x} \in H_v(P)} \max_{\hat{y} \in H_v(Q)} f_P^T(\hat{x}) f_Q(\hat{y}) \]

feature map of image \( P \) wrt origin \( \hat{x} \)
Feature map similarity (FMS)

\[ S_F(P, Q) = \max_{v \in V(P, Q)} \max_{\hat{x} \in H_v(P)} \max_{\hat{y} \in H_v(Q)} f_P^T(\hat{x}) f_Q(\hat{y}) \]

feature map of image \( P \) wrt origin \( \hat{x} \)

feature map of image \( Q \) wrt origin \( \hat{y} \)
Feature map similarity (FMS)

\[
S_F(P, Q) = \max_{v \in V(P,Q)} \max_{\hat{x} \in H_v(P)} \max_{\hat{y} \in H_v(Q)} f_P^T(\hat{x}) \cdot f_Q(\hat{y})
\]

for all origins mapped to visual word \( v \)

feature map of image \( P \) wrt origin \( \hat{x} \)

feature map of image \( Q \) wrt origin \( \hat{y} \)
Feature map similarity (FMS)

for all visual words that $P, Q$ have in common

$$S_F(P, Q) = \max_{v \in V(P, Q)} \max_{\hat{x} \in H_v(P)} \max_{\hat{y} \in H_v(Q)} f_P^T(\hat{x}) f_Q(\hat{y})$$

feature map of image $P$ wrt origin $\hat{x}$

feature map of image $Q$ wrt origin $\hat{y}$
Feature map similarity - example

Inliers using fast spatial matching [FastSM - Philbin et al. ] (35 inliers)

Inliers using feature map similarity (32 inliers)
Distribution of $\rho$

- Non-linear transformation using Weibull CDF
- Estimation of parameters via maximum likelihood
- Bins equally populated when distribution w.r.t. $\rho$ is uniform
Memory savings – speed

Unique visual words
- Use as origins only features that map uniquely to visual words

Range parameter $\tau$
- Add constraints on spatial proximity via range parameter $\tau$
- $\tau \in [0, 1]$ controls the balance between local and global geometry

Origin selection
- Statistically measure which visual words get better aligned
- Select as origins only features mapped to those visual words
Memory savings – speed

Unique visual words
- Use as origins only features that map uniquely to visual words

Range parameter $\tau$
- Add constraints on spatial proximity via range parameter $\tau$
- $\tau \in [0, 1]$ controls the balance between local and global geometry

Origin selection
- Statistically measure which visual words get better aligned
- Select as origins only features mapped to those visual words
Memory savings – speed

Unique visual words
• Use as origins only features that map uniquely to visual words

Range parameter $\tau$
• Add constraints on spatial proximity via range parameter $\tau$
• $\tau \in [0, 1]$ controls the balance between local and global geometry

Origin selection
• Statistically measure which visual words get better aligned
• Select as origins only features mapped to those visual words
Memory savings – speed

Unique visual words
- Use as origins only features that map uniquely to visual words

Range parameter \( \tau \)
- Add constraints on spatial proximity via range parameter \( \tau \)
- \( \tau \in [0, 1] \) controls the balance between local and global geometry

Origin selection
- Statistically measure which visual words get better aligned
- Select as origins only features mapped to those visual words

\( \tau = 0.6 \)
Memory savings – speed

Unique visual words
• Use as origins only features that map uniquely to visual words

Range parameter $\tau$
• Add constraints on spatial proximity via range parameter $\tau$
• $\tau \in [0, 1]$ controls the balance between local and global geometry

Origin selection
• Statistically measure which visual words get better aligned
• Select as origins only features mapped to those visual words

\[ \tau = 0.7 \]
Memory savings – speed

Unique visual words

- Use as origins only features that map uniquely to visual words

Range parameter $\tau$

- Add constraints on spatial proximity via range parameter $\tau$
- $\tau \in [0, 1]$ controls the balance between local and global geometry

Origin selection

- Statistically measure which visual words get better aligned
- Select as origins only features mapped to those visual words

$\tau = 0.8$
Towards indexing

- FMS is a fast way of matching 2 images, but still not enough for indexing
- A feature map is an extremely sparse histogram; bin count typically takes values in \( \{0, 1\} \)
- Each feature map \( f \) is represented by set \( \bar{f} \subset \mathcal{W} \) of non-empty bins
Min-wise independent permutations

- The feature space is now $\mathbb{F} = \mathcal{P}(\mathcal{W})$, the powerset of $\mathcal{W}$
- $h : \mathbb{F} \rightarrow \mathcal{W}$, hash function mapping objects back to $\mathcal{W}$
- $\pi : \mathbb{F} \rightarrow \mathbb{F}$, a random permutation
- Given a feature map $\bar{f} \subset \mathcal{W}$: compute a hash value $h(\bar{f}) = \min\{\pi(\bar{f})\}$

$$\Pr[\min\{\pi(\bar{f})\} = \min\{\pi(\bar{g})\}] = \frac{|\bar{f} \cap \bar{g}|}{|\bar{f} \cup \bar{g}|} = J(\bar{f}, \bar{g})$$

- Two features maps are hashed to the same value with probability equal to their resemblance or Jaccard similarity coefficient
Map sketch

- Construct a set \( \Pi = \{ \pi_i : i = 1, \ldots, m \} \) of \( m \) independent random permutations
- Represent each feature map \( \bar{f} \) by map sketch \( f \in \mathcal{W}^m \),

\[
f = f(\bar{f}) = [\min\{\pi_1(\bar{f})\}, \ldots, \min\{\pi_m(\bar{f})\}]^T
\]

- Sketch similarity, count number of elements that sketches \( f, g \) have in common

\[
s_K(f, g) = m - \|f - g\|_0
\]
Feature map hashing (FMH)

• Map sketch collection $F$: set of all map sketches $f$ of an image

• Image similarity reduces to sketch similarity

$$S_M(F, G) = \max \max_{f \in F} s_K(f, g)$$

• Collisions may appear for several pairs of maps; sum all sketch similarities instead of keeping the best one

$$S_K(F, G) = \sum_{f \in F} \sum_{g \in G} s_K(f, g)$$
Matching maps

Multiple matching pairs of feature maps
Matching maps

Multiple matching pairs of feature maps
Matching maps

Multiple matching pairs of feature maps
Matching maps

Multiple matching pairs of feature maps
Matching maps

Multiple matching pairs of feature maps
Matching maps

Multiple matching pairs of feature maps
Indexing

Index construction

- Represent the entire dataset by triplet \((\hat{v}, w, \pi)\) (origin, sketch element, permutation)
- Use an inverted file for sub-linear search
- Memory requirements \(5\times\) a typical baseline system

Query

- Construct triplet \((\hat{v}, w, \pi)\) for query image
- Rank images with a voting process
- Re-estimate transformation parameters using LO-RANSAC
- Re-ranking is an order of magnitude faster than FastSM, because an initial estimate is already available