

# IMAGE MATCHING UNDER GENERALIZED HOUGH TRANSFORM\*

Qiang Li and Bo Zhang

*State Key Laboratory of Intelligent Technology and System, Computer Science & Technology Department, Tsinghua University, Beijing 100084, China*

## ABSTRACT

The paper analyzes the techniques of image template matching, and gives a matching framework based on generalized Hough transform, which is applicable to lots of current methods. With the framework's guidance, extracting image features with specific characteristics and devising feature matching algorithms become easier and more efficient. We propose a new fast matching algorithm as evidence. It is robust for the linear transformation of image grey value and image noise. Its time complexity reduced to  $O(\text{size of search image})$ , the theoretic lower limit for image matching.

## KEYWORDS

Image processing; Generalized Hough transform; Template matching; Algorithm time complexity

## 1. INTRODUCTION

Image matching is a crucial technique in image process tasks for scene fusion taken at different times or spectrums, or object geometric localization in the variety backgrounds. It is widely used in computer vision for object tracking, remote sensing processing, scene navigation, and medical image analysis etc. Due to the diversity of images acquisition manner and application areas, image matching has gotten much recent attention to research. More than 1000 papers published in the last 10 years were indexed by the databases of the Institute of Scientific Information(ISI) on this topic (Zitová and Flusser, 2003). It is hardly to design a theoretical framework to analysis and compare all or most of image matching techniques.

Usually, image matching technique can be divided into two major categories: area-based method and feature-based method. This partition originates from the stage of feature extraction, and retains in the next stage of feature matching. Area-based method uses the pixel intensity directly to compute the similarity measure between a small template and the observation window sliding on the whole image, by the quantities of statistical correlation, Fourier cross-power spectrum or mutual information. Feature-based method uses the higher level structure-features including more visual contents, such as the descriptions and properties of dominant points, lines, edges and regions. Thus this method is applicable to the case that images contain sufficient distinctive and easily detectable feature objects.

Hough transform was introduced for image segmentation (Hough 1962; Illingworth and Kittler 1988). The underlying principle of the Hough transform is to map the data of image space into the parametric space, search the optimal parameters so fitting the features in the image space. The classical Hough transform identifies lines in the image, but the generalized Hough transform (Ballard 1981) extends the principle to identifying positions of arbitrary shapes, and now it becomes an powerful and robust parameter estimation technique in computer vision and image analysis (Zhang 1997).

In this paper, we suggest an image matching framework based on generalized Hough transform, which can localize the components of image matching techniques, analysis the common thoughts and structures. With the framework's guidance, extracting image features with specific characteristics and devising feature matching algorithms become easier and more efficient. We propose a new fast matching algorithm as

---

\* This work is supported by National Natural Science Foundation of China (60321002 and 60135010) and by the Chinese National Key Foundation Research & Development Plan (2004CB318108).

evidence. It is robust for the linear transformation of image grey value and image noise. Its time complexity reduced to  $O(\text{size of search image})$ , the theoretic limit for exact registration.

The paper is organized as follows. Section 2 describes the image matching framework, based it then gives the analysis of the current matching manners. In Section 3 we propose a new fast template matching algorithm by the framework guidance. Finally, Section 4 concludes the work and offers the outlook for the future.

## 2. GENERALIZED HOUGH TRANSFORM FRAMEWOKR FOR IMAGE MATCHING

The philosophy of generalized Hough transform is a voting mechanism or clustering. Parameter space is appropriately quantized into bins. For each selected feature, all possible parameters are evaluated and respective bins in parameter space are accumulated, just like one-to-more voting. After that, a cluster approach is used in these bins to estimate the optimal parameters. Bayesian generalized Hough transform (Ji 2001) takes account of the uncertainty in the detection and selection of features, and uses the Bayesian inference to compute the contribution of each feature to the parametric accumulator (bin). Shekhar et al(1999) indicated that the finial accumulating results is equal to the cross-correlation of the Hough-transformed distribution of attributes of image features. Chang et. al(1997), Zana and Klein(1999), González-Linares(2003) etc. have used this method to estimate the transformation parameters according features between two images.

Actually, given a small image namely template  $T$  and a big searched image  $S$ , the template matching is to find and locate the optimal sub-window of the searched image, which is most similar to the template under certain similarity measure. The position of sub-window on the searched image is referred to as *reference point*. This can be regarded as each feature offers contribution to some extent for the sub-window which the feature falls into. This can be stated formally as:

$$C[l] = c(f[k], l | k \in w(l)) \quad (1)$$

where  $C[l]$  represents the similarity measure of the sub-window with reference point  $l$ .  $w(l)$  represents the extent of sub-window  $l$ . Function  $c(\bullet)$  composites all features  $f[k]$  of which the position  $k$  falls into the sub-window extent  $w(l)$ . This can be simplified by:

$$C[l] = \sum_{k \in w(l)} h(f[k], l) \quad (2)$$

where the function  $c(\bullet)$  in (1) usually used to be an additive model, or other operators e.g. product no matter. Note the second argument of  $h(\bullet)$  provides the feature's relative position within the sub-window, that is the same feature can be utilized by different sub-window for different devotion. And this representation preserves the geometric structures.

For area-based image matching methods, pixel gray value is the lowest-level feature used in comparison. Thus we can write the normalized cross-correlation method by

$$NCC(l) = \sum_{k \in w(l)} \frac{N}{\sigma_{w(l)} \sigma_T} (f(k) - \mu_{w(l)})(t(k-l) - \mu_T)$$

where  $\mu, \sigma$  represent mean and standard deviation,  $t(k-l)$  the corresponding pixel gray value of the template to pixel gray value  $f(k)$  of the sub-window,  $N$  the total number of pixels in the template. Other area-based methods e.g. sum of absolute difference method or sum of squared differences method are easily derived the similar forms satisfying (2). Exploiting the coarse-to-fine resolution strategy (the pyramidal approach) to enhance efficiency is intrinsically searching on higher-level feature or coarser granularity and can be written in the form like (2).

Towards feature-based image matching methods, high-level features e.g. angle points detected within searched image can be compared with each angle point detected within template, to increase posteriori probability for every possible matching. This is especially useful to multimodal images matching, by which the same object may has different gray value distribution taken from different sensors.

Now using the idea of generalized Hough transform, image matching procedure can be described as follows:

1. detect and extract features from searched image.
2. for each feature ( $k$ ), evaluate each possible matching position, accumulates the result to corresponding position accumulator.
3. search or cluster the accumulator array, eliminate illuminating noise and other distorted condition, give the all acceptable matching localizations.

Using cascade-like method (Viola and Jones 2001; Schneiderman 2004), above procedure can be iterated so using simple feature rapidly rejects most dissimilar regions in searched image. Features composition can choose the semi-naïve Bayesian test to achieve similar-content matching:

$$C[l] = \sum_{k \in w(l)} \log \frac{P(h(f[k], l) | \omega_1)}{P(h(f[k], l) | \omega_2)} > \lambda \quad (3)$$

where  $\omega_1, \omega_2$  are the similar and dissimilar classes respectively.  $\lambda$  is the acceptable threshold.

### 3. A NEW FAST MATCHING ALGORITHM

Under the guidance of the matching framework based on generalized Hough transform, we present a new fast image matching algorithm. It originates from the fractal-coding image retrieval (Zhang et al. 1996; Wang et al. 2001). Block-constrained fractal coding can be seemed as local self-similar of images. Extracted this self-similarities as feature, images can be indexed and retrieved based visual contents. This paper utilizes the concepts of Range-block and Domain-block in fractal coding, devises a new feature coding directly based on gray value. Intrinsically, it is a description of local distribution of image gray value. Its feature extracting computation is very easy. Only equality-comparison and accumulation is needed during the matching procedure. Theoretical analysis and experiment results will prove the algorithm has a time complexity near the lower-bound and is robust to luminance linear transform and image noise.

#### 3.1 Feature Extraction

For extracting feature robust to noise and luminance variance, we need define the coarser granularity feature using more pixels. Therefore image is first divided into small non-overlapping blocks with pixel size  $k \times k$ , called *R-block*.  $k$  can be chosen according to image resolution. With side length  $H$ , an image has  $[H^2/k^2]$  R-blocks.

**Definition 1** One R-block (e.g.  $R_5$  in Figure 1) and its 8 neighboring R-blocks consist of the *R-block neighborhood*. Divide a R-block neighborhood into 4 parts,  $D_1, D_2, D_3, D_4$  (as Figure 1), each is referred as a *R-block's D-neighborhood*, having

$$D_1 = R_1 \cup R_2 \cup R_4 \cup R_5, \quad D_2 = R_4 \cup R_5 \cup R_7 \cup R_8, \quad D_3 = R_5 \cup R_6 \cup R_8 \cup R_9, \quad D_4 = R_2 \cup R_3 \cup R_5 \cup R_6.$$

For four R-blocks within the D-neighborhood  $D_j$ , define one counter-clockwise order illustrated in Figure 2. Sorting the sum of pixel gray values of these four R-block, there is  $4! = 24$  possible results.

**Definition 2** According to inner orders of four D-neighborhoods of R-block  $R_i$ , we can represent  $R_i$  overall feature as a single value with a range of  $24^4$  possible values, written as  $F(R_i)$ .

Obviously,  $F(R_i)$  describe the order-relation of the gray value distribution local among  $R_i$  and its eight R-block neighbors. It is robust for luminance linear variation and random noise.

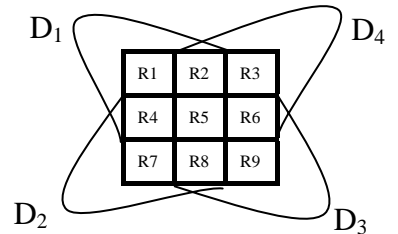


Figure1. Feature definition of R-block

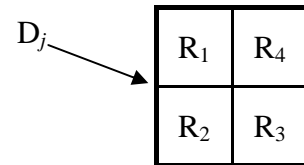
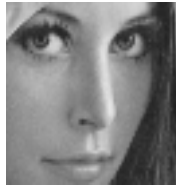


Figure 2. Inner order of the D-neighborhood  $D_j$

### 3.2 Feature Matching

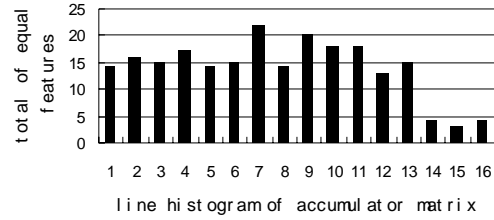
Divide the searched image  $S$  into blocks with the same size of template  $T$ , called as *constraint-blocks*. Feature matching procedure will restrict to each constraint-block in order to achieve higher efficiency. After the subdivision, if there are reminders on the rightmost (or the nethermost) of the searched image  $S$ , add one column (or one line) constrain-block from the right edge (or bottom) of  $S$  so that the subdivision completely covers  $S$ . Assuming the side lengths of  $S$  and  $T$  are  $M$  and  $N$  respectively, there are  $\lceil M/N \rceil^2$  constraint-block on  $S$ .

For each constraint-block  $C^{i,j}$  of  $S$ , where  $(i,j)$  is its position coordinate on  $S$ , we get a  $[N/k]$  dimension square matrix with R-block features as elements. Set the accumulators for feature matching as a  $[N/k]$  dimension square matrix too. Compared each R-block  $R_i$  feature in  $C^{i,j}$  with all R-block features extracted from template  $T$ , if they are equal, then increase the corresponding accumulator with  $R_i$  having the same matrix coordinate. This means increasing the possibility of any constraint-block which the R-block  $R_i$  belongs to.



(a) template  $T$  (b) a constraint block in  $S$   
Figure 3. Processing of feature comparing result

Table 1 Example of clustering line histogram of accumulator matrix



After the comparing stage, directly clustering the accumulator matrix we can get the matching region (although maybe a partial matching region due to the size limit of the constraint-block) if it exists. Therefore only the constraint-block whose total number of equal features after comparing is biggest or is above some threshold need cluster. The clustering method can be implemented efficiently by filtering two 1-D histograms along the line- and column-direction of the accumulator matrix. For example, in figure 3, the constraint-block in (b) partial matches the template in (a), so as the line histogram of the accumulator matrix of feature comparison is showed in Table 1. It is easy to detect the clustering line-edge.

Some techniques are very important to let the matching algorithm more efficient and effect. First, utilizing quick-sort algorithm to sort two feature sets of constrained-block  $C^{i,j}$  and template  $T$ . It will lead to the order-comparing. Time complexity of feature sorting and comparison will reduce from  $O(N^4)$  to  $O(N^2 \log(N))$ . Cross-correlation and other area-based matching methods cannot utilize sorting, because they need arithmetic operations, not equality comparing.

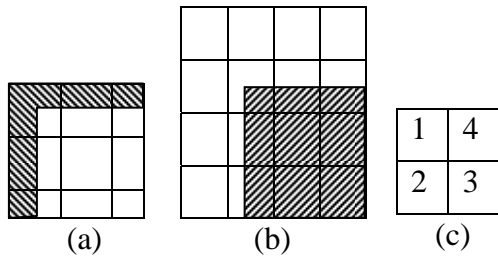


Figure 4. Aligning template position  
(a)template (b)search image (c) template after cut

Secondly, to achieve the pixel-level matching precision, we need align the template position to searched image. As show in Figure 4, the shade in Figure 4(b) matches with the template in Figure 4(a). However after R-block subdivision in searched image and template (illustrated with the grid in Figure 4(a) & (b)), there exist not common-content R-block between searched image and template.

Cut the shade from the template in Figure 4(a), the remainder can divide into four R-blocks showed in Figure 4(c), has aligned with bottom-rightmost four R-blocks of searched image in Figure 4(b). Cut  $u$  lines and  $v$  columns R-blocks from the leftmost and topmost of template, where  $0 \leq u, v \leq k$ , getting the cut-

template  $T^{u,v}$ . Obviously, the cost of pixel-level localization is doing more  $k^2-1$  comparison with searched image. But this also makes the cluster edge more distinguishable.

Whole matching algorithm's time complexity concentrates on stage of feature comparing. Time complexities of feature extraction on template and searched image are  $O(k^2N^2)$ ,  $O(M^2)$  respectively; clustering on the comparison result need only  $O(N)$ . The time complexity of feature comparison is

$$O\left(\left\lceil \frac{M}{N} \right\rceil^2 \left\lceil \left( \frac{N}{k} \right)^2 \log \frac{N}{k} + k^2 \left( \frac{N}{k} \right)^2 \right\rceil\right) = O\left(M^2 \frac{\log N}{k^2} + M^2\right).$$

It means for the total number  $\lceil M/N \rceil^2$  of constrained-blocks on searched image, each has  $(N/k)^2$  features. Sorting each feature set need  $O((N/k)^2 \log(N/k))$ . Orderly comparison each feature set with  $k^2$  of cut-templates costs  $O(k^2(N/k)^2)$ . Usually, the side length  $N$  of template  $T$  is lesser than 100, so  $\log N < k^2$ . Therefore the time complexity of the new algorithm can achieve  $O(M^2)$ , the theoretical lower bound.

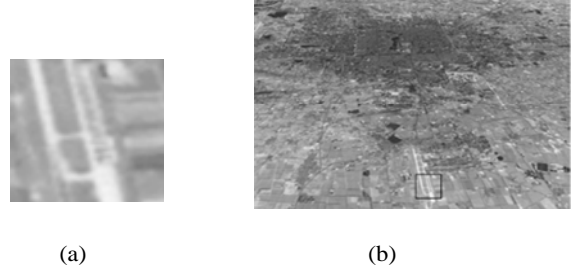
### 3.3 Experiments

We compare our method with two area-based methods: the widely used normalized cross-correlation (NCC) and computing-simplest sum of absolute difference (SAD), use the real satellite photograph and adding noise portrait photo as examples.

Figure 5 illustrates a satellite remote sensing photo as the searched image with pixel size  $800 \times 600$ . Choose one  $64 \times 64$  sub-window locating on (425,500) as the template. The algorithm computes that the constraint-block with reference point (448,512) has the maximum equal features with template  $T^{0,3}$ . After clustering, the upper-rightmost 11 lines and 8 columns belong to the matching region. Thereafter the global matching reference point (425,500) can be figured out.

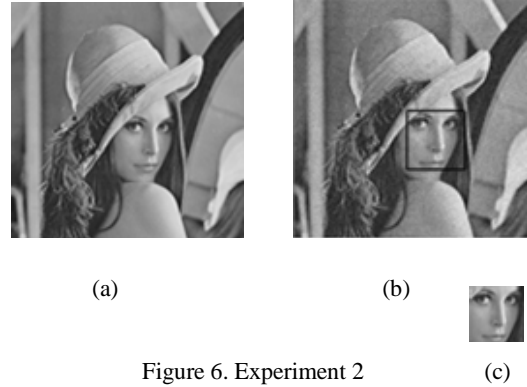
Figure 6 shows the Gaussian white noise is added to the  $256 \times 256$  original portrait photo Figure 6(a) as the searched image Figure 6(b), which has evident visual degradation. Cut a  $64 \times 64$  sub-image from Figure 6(a) as template. The new algorithm is robust to compute the correct matching location.

Table 2 lists the different running result of our algorithm, ABS and NCC methods on Pentium III 900Mhz with  $k = 4$ . Comparative with current correlation methods, our new algorithm improves two orders of magnitude in running time.



(a) template (b) search image zoom out six times

Figure 5. Experiment 1



(a) original image (b) search image added noise (c) template cut from the origin image

Figure 6. Experiment 2

Table 2 Comparison of computing times of three matching algorithms

	Experiment 1. 800×600 satellite image (millisecond)	Experiment 2. 256×256 portrait photo (millisecond)
our new algorithm	40.1	5.0
ABS	12107.4	1151.7
NCC	26658.3	2543.6

## 4. CONCLUSION

The main contributions of this paper are: first, a new image matching framework based generalized Hough transform has been presented. This is robust parametric estimation and clustering method even though image is distorted by noise and discretization. By selecting various feature and feature composition function, it is applicable for various applications. Secondly, as an argumentation, a fast grey value-based matching method is described. It extracts a robust local grey value feature, utilizes the geometry position information to vote the possible matching region, use clustering to get the resulting pixel-precision location. Theoretical analysis and several examples with real images have proved the robustness and efficiency of the new algorithm.

In future works, invariant features can be introduced into this method, such as angles and edge descriptors or robust local textures. After direct comparing feature equality with those of deformable object template, clustering procedure can detect the matching region edge in arbitrary shape. Another application is automatically compute the matching regions with arbitrary shape edge between two images even without object template. These should be invariant for geometrical transformation and be robust to image noise.

## REFERENCES

- Ballard, D.H., 1981. Generalizing the Hough Transform to Detect Arbitrary Shapes. *In Patter Recognition*. Vol. 13, No. 2, pp. 111-122.
- Chang, S-H. et al., 1997. Fast Algorithm for Point Pattern Matching: Invariant to Translations, Rotations and Scale Changes. *In Pattern Recognition*. Vol.30, No. 2, pp. 311-320.
- González-Linares, J.M. et al., 2003. An Efficient 2D Deformable Objects Detection and Location Algorithm. *In Pattern Recognition*. Vol. 36, No. 11, pp. 2543-2556.
- Hough, P.V., 1962. Method and Means for Recognizing Complex Patterns. *In U.S. Patent 3069654*.
- Ji, Q., 2001. Error Propagation for the Hough Transform. *In Pattern Recognition Letters*. Vol. 22, No. 6-7, pp. 813-823.
- Illingworth J. and Kittler J, 1988. A survey of the Hough Transform. *In Computer vision, graphics, and image processing*, Vol 44, No 1, pp 87-116.
- Schneiderman H., 2004. Feature-Centric Evaluation for Efficient Cascaded Object Detection. *Proceedings of IEEE Conference on Computer Vision and Patter Recognition (CVPR2004)*. Washington, D.C., USA, Vol. 2, pp. 29-36.
- Shekhar, C. et al., 1999, Multisensor Image Registration by Feature Consensus. *In Pattern Recognition*. Vol. 32, No. 1, pp. 39-52.
- Viola, P. and Jones, J.M., 2001, Rapid Object Detection Using a Boosted Cascade of Simple Features. *Proceedings of IEEE Conference on Computer Vision and Patter Recognition (CVPR2001)*. Kauai Marriott, Hawaii , USA, Vol. 1, pp. 511-518.
- Wang Z. et al., 2000. Content-based image retrieval using block-constrained fractal coding and nona-tree decomposition. *In IEE Proceedings - Vision, Image and Signal Processing*. Vol. 147, No. 1, pp. 9~15.
- Zana, F. and Klein, J.C., 1999. A Multimodal Registration Algorithm of Eye Fundus Images Using Vessels Detection and Hough Transform. *In IEEE Transactions on Medical Imaging*. Vol. 18, No. 3. pp. 419-428.
- Zhang A. et al., 1996. A fractal-based clustering approach in large visual database systems. *In The International Journal on Multimedia Tools and Applications*, Vol. 3, No. 3, pp. 225~244.
- Zhang Z., 1997. Parameter Estimation Techniques: A Tutorial with Application to Conic Fitting. *In Image and Vision Computing*. Vol. 15, No. 1, pp. 59-76.
- Zitová, B. and Flusser J, 2003. Image Registration Methods: A Survey. *In Image and Vision Computing*, Vol. 21, No. 11, pp 977-1000.