# Content-based retrieval of line-drawing images using angular information of hough transform

Pasi Fränti[1], Alexey Mednonogov[2] and Heikki Kälviäinen[2]

[1] Department of Computer Science
University of Joensuu, Finland

[2] Department of Information Technology
Lappeenranta University of Technology, Finland

## ABSTRACT

We introduce a novel method for content-based retrieval of line-drawing images. The method is based on Hough transform, which is capable of extracting global linear features in the image. Images are indexed by generating feature vectors from the quantized accumulator matrices from the Hough transform. The matrix is first thresholded in order to preserve only the most significant values. The feature vector is then constructed by summing up the significant coefficients in the columns of each θ-value. Thus, only the angular information is used. The approach allows simple implementation of scale, translation and rotation invariant matching.

***Keywords:*** *Content-based retrieval, document image processing, indexing, Hough transform, line-drawings.*

## 1. INTRODUCTION

We consider database of complex engineering drawings, e.g. electrical circuits, cartographic maps, architectural and urban plans. We assume that the images are binary (black-and-black) images, and that they consist mainly of lines; this is a reasonable assumption for a variety of engineering drawings. Non-linear components such as symbols and complex curves are not taken into consideration in the assessment of the similarities.

A common approach for the image retrieval problem is to use indexing scheme in order to avoid full scale image matching [1]. There are two main approaches for indexing: *keywords* and *content-based retrieval*. Keywords have the advantage that the solutions for text retrieval [2] would be available for us. This approach, however, is impractical because we must design keywords and proper classification for the images in the database. Secondly, manual operations are needed for creating the keywords.

In content-based image retrieval, the images are indexed by generating a *feature vector* describing the content of the image. The feature vector is generated and stored at the same time the image is stored in database. The vector consists of global features describing the content of the image. The retrieval is performed by measuring the similarity of the feature vectors of the query image and the database images.

We introduce a novel indexing scheme for line-drawing images. The method is based on the *Hough transform* [3-6], which is applied for generating the feature vector. Hough transform is well suited for this task because it gives a global description of the spatial image content. It makes no assumptions on the image type and, in principle, it should be applicable to any type of bi-level images.

## 2. HT-BASED INDEXING

Retrieval is performed by asking a sample drawing as an input. Given the sample image, the task is to find all database images similar to the query image. A feature vector is generated from the query image, and this feature vector is used for the matching.

The selection of the features is the key part of the indexing scheme. The size of the feature vector should be small enough to be stored compactly and processed efficiently, but it should also representative so that then images can be differentiated from each other on the basis of their feature vectors.

We apply Hough transform for the indexing as it gives scale independent global description of the spatial image content. We will not use the full accumulator matrix because it can be too space consuming. Instead, we use only angular information and in this way, we can reduce the storage requirements of the indices and also limit the number of comparison operations needed in the matching.

### 2.1. Hough transform

In Hough Transform, global features are sought as sets of individual points over the whole image. In the simplest case the features are straight line segments. In the case of binary images, the line detection algorithm can be described as follows:

1. Create the set *D* of the black pixels in the image.
2. Transform each pixel in *D* into a parameteric curve in the parameter space.
3. Increment the cells in the accumulator matrix *A* determined by the parametric curve.
4. Detect local maxima in the accumulator array. Each local maximum may correspond to a parametric curve in the image space.
5. Extract the curve segments using the knowledge of the maximum positions.

In Step 2, the transformation can be $\rho = x\cdot\cos\theta + y\cdot\sin\theta$ where $(x, y)$ are the coordinates of the pixel to be transformed, and ρ and θ are the parameters of the corresponding line. Thus, every pixel $(x, y)$ can be seen as a curve in the $(\rho, \theta)$ parameter space where θ varies from the minimum to the maximum value, giving the corresponding ρ values. By transforming every point $(x, y)$ in the image into the parameter space, the line parameters can be found in the intersections of the parameterized curves in the accumulator matrix.

## 2.2. Generating feature vector

We denote the accumulator matrix by $A$, and each row in corresponds to one value of ρ, and each column to one value of θ. The procedure for generating the feature vector from the accumulator matrix is described in Fig. 1. First we extract only the most significant information by thresholding the matrix using a threshold value $T$ (step 1). Next, we shrink the thresholded accumulator matrix to one-dimensional θ-vector by summin up the remaining coefficients in each column (step 2). Thus, only the angular information of the matrix will be used. Finally, we normalize the feature vectors according to the mean value of the components of the vector (steps 3 and 4).

## 2.3. Translation and scale invariant matching

The usage of only the angular information (θ-vector) has several advantages. Firstly, the matching is independent of the spatial location of the lines and therefore the method is translation and scaling invariant by its nature. This is verified by the fact that every line has the same θ-value independent of scaling and translation of the image. The angular information can be sufficient for separating different image types. For example, drawing of buildings consists mainly of 45° and 90° angles.

We approximate the dissimilarity of the images by calculating the distance of their feature vectors. Let us assume that we got feature vector $D$ for database image and feature vector $S$ for sample query image (both of size $N$). The distance is calculated as:

$$d = \sum_{j=1}^{N} (D_j - S_j)^2 \tag{1}$$

Here $d$=0 coincides to absolutely similar images and $d$=$d_{max}$ coincides to images with no similarities found. An important advantage of the simplicity of the formula is its speed. This is highly desired property for searching from large image database.

1. Threshold the matrix:
$$A'_{ij} = \begin{cases} A_{ij}, & \text{if } A_{ij} > T \\ 0, & \text{if } A_{ij} \le T \end{cases} \quad \forall \; i = 1..M, \;\; j = 1..N$$
2. Calculate preliminary feature vector:
$$F_j^0 = \sum_{i=1}^{M} A'_{ij} \quad \forall \;\; j = 1..N$$
3. Calculate vector mean:
$$m = \frac{1}{N} \sum_{j=1}^{N} F_j^0$$
4. Normalize the feature vector:
$$F_j = \frac{F_j^0}{m} \quad \forall \quad j = 1..N$$

**Figure 1**: Algorithm for generating feature vector.

## 2.4. Rotation invariant matching

Rotation invariant matching of the method can be obtained with minor modifications at the cost of increasing running time if we consider the matching as a histogram matching problem. For example, the two feature vectors in Fig. 2 have the same shape but they have different location along the θ-axis. The best match can be found by rotating the feature vector of the sample query image, and calculating the distance in all positions.

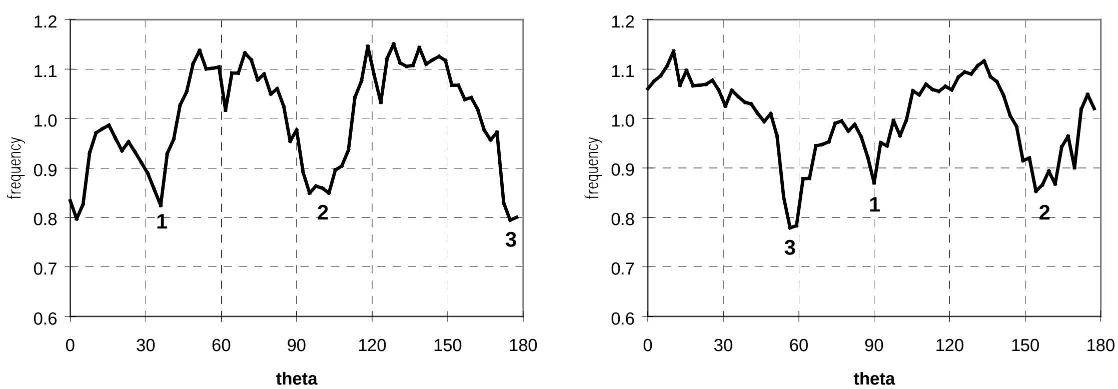


**Figure 2**: Graphical representation of the two feature vectors. The second feature vector is from the same image than the first one after the image is rotated right by 45°.

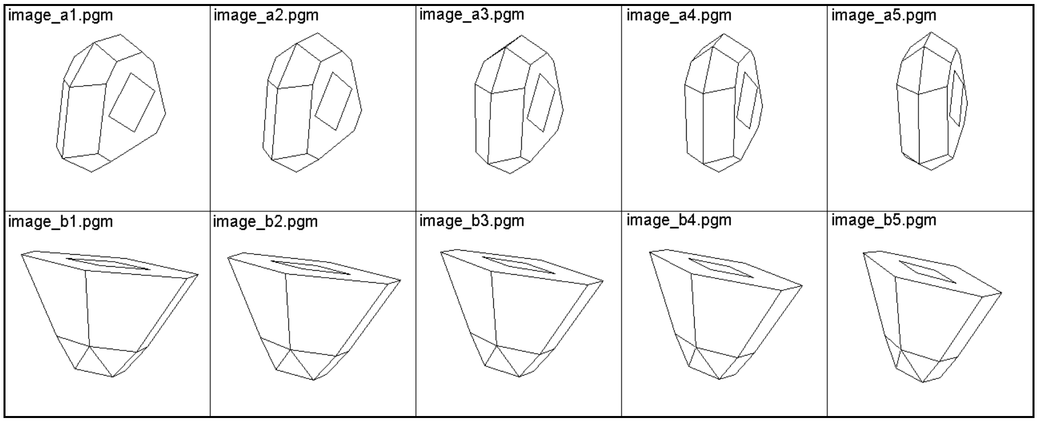


**Figure 3**: Test database of 10 images. The images have been generated by performing 3-D rotation for the two original images (*Image A* and *Image B*).

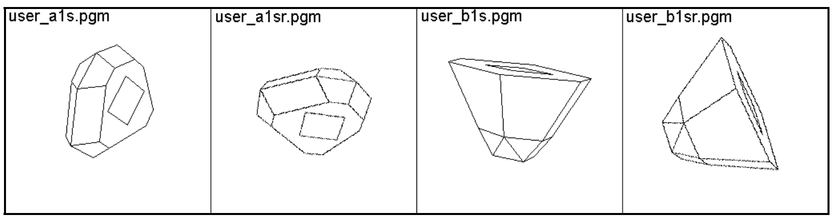


**Figure 4**: Four query images from left to right: scaled version of *Image A*, scaled and 2-D rotated version of *Image A*, scaled version of *Image B*, scaled and 2-D rotated version of *Image B*.

The distance of the feature vectors is defined as the minimum distance of all possible fittings:

$$d = \min_{k=1,2,\ldots,N} \sum_{j=1}^{N} (D_j - S^{(k)}{}_j)^2 \qquad (2)$$

where $S^{(k)}$ is a vector obtained by cyclic rotation of the feature vector $S$ by $k$ steps to the right. Thus, we use exhaustive search for finding the minimum distance. The drawback of the approach is that it requires $k$ times of computation in comparison to the previous variant.

## 3. EXPERIMENTS

We test the retrieval accuracy of the proposed method using a small image database of 10 images shown in Fig. 3. The database is generated by taking two different vector objects as the starting point. The objects were rotated by 5, 10, 15, 20 and 25 degrees around the *y*-axis and the corresponding images were stored in the database. The idea is that the rotation is performed in 3-D vector space and not in the 2-D image plane. The original images are denoted as image *A* and *B*, and their rotated versions as images *A1*, *A2*, *A3*, *A4*, *A5*, *B1*, *B2*, *B3*, *B4*, *B5*.

Next we generate four query images from the two original images, as shown in Fig. 4. The first query image (*a1s*) is a scaled version of the image *A*, and the second one (*a1sr*) is a scaled and rotated version of the same image. The third (*b1s*) and fourth (*b1sr*) images are generated from the image *B* respectively. The hypothesis is that the first two query images should match the database image *A1* since only scaling and 2-D rotation were performed. Slightly weaker matching should be made to the images *A2*, *A3*, *A4* and *A5*. Moreover, the images *a1s* and *a1sr* should not match to the images *B1*, *B2*, *B3*, *B4* and *B5*.

The results of the matching are show in Fig. 5 as the distance between the query image and the database image. The results verify the hypothesis as the image *A1* is the closest to both *a1s* and *a1sr*; and the image *B1* is the closest to *b1s* and *b1sr*, respectively. The results also show that the images *A2* and *A3* are rather close according to the measured distance values. The images *A4* and *A5*, on the other hand, do not match so well as their distance is already as far as that of the best matched images *B4* and *B5* in the different (wrong) test set. The results also show that matching is also rotation invariant as the results for *a1s* (*b1s*) and *a1sr* (*b1sr*) are similar to each other.

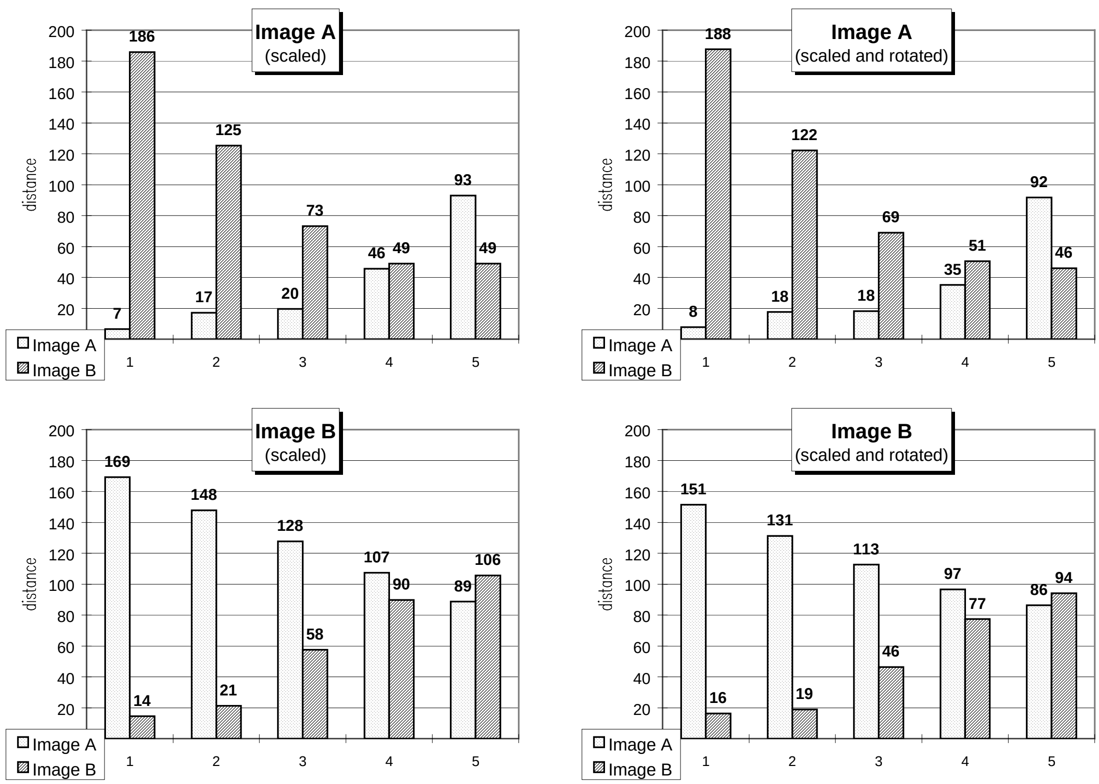


**Figure 5**: Matching results between the query images (*a1s*, *a1sr*, *b1s*, *b1sr*) and the databased images (*A*1-*A*5 and *B*1-*B*5). The results are given as the distances values according to (2).

## 4. CONCLUSIONS

We introduced a novel method for content-based retrieval of line-drawing images based on Hough transform. The method was shown to work well in practice. The advantages of the method are its speed, and the fact that it is invariant of scaling, translation and rotation. Extension to gray-scale and color image seems to be straightforward, too.

Further research is still required to understand how the proposed method could be efficiently applied to large image of more complex structures. Another limitation of the method is that it only answers the question of full image similarity. There are several applications where we need to search for small parts inside a large image. It is not yet clear how the proposed approach could be applied in such context. These points are the starting point or our future research.

## REFERENCES

1. W.-Y. Ma, H.J. Zhang, "Content-based image indexing and retrieval", in *Handbook of Multimedia Computing*, pp.227-254, CRC Press, 1999
2. I.H. Witten, A. Moffat and T.C. Bell, *Managing Gigabytes: Compressing and Indexing Documents and Images* (2nd edition). Van Nostrand Reinhold, New York (1999).
3. P.C.V. Hough, Methods and means for recognizing complex patterns. U.S. Patent 3,069,654, 1962
4. V.F. Leavers, Survey: Which Hough Transform. *CVGIP Image Understanding* 58 (2): 250-264, 1993.
5. J.R. Parker, *Algorithms for image processing and computer vision*. John Willey & Sons, 1996
6. H. Kälviäinen , P. Hirvonen, L. Xu, E. Oja, Probabililistic, non-probabilistic Hough transforms: overview and comparisons. *Image, Vision Computing* 13 (4), 239-251, May 1995.

## Authors:

***Pasi Fränti***, professor in the computer science dept., university of Joensuu, Finland.

Address: P.O. box 111, 80101 Joensuu, Finland.
E-mail: franti@cs.joensuu.fi

***Alexey Mednonogov***, M.Sc. student in the information technology dept., Lappeenranta university of technology, Finland.

Address: P.O. Box 20, FIN-53851 Lappeenranta, Finland.
E-mail: alexey.mednonogov@lut.fi

***Heikki Kälviäinen***, professor in the information technology dept., Lappeenranta university of technology, Finland.

Address: P.O. Box 20, FIN-53851 Lappeenranta, Finland.
E-mail: heikki.kalviainen@lut.fi