A General Framework for Image Retrieval using Reinforcement Learning

S. Srisuk[1], R. Fooprateepsiri[3], M. Petrou[2], S. Waraklang[3] and K. Sunat[1]
Department of Computer Engineering[1], Department of Information Technology[3],
Mahanakorn University of Technology, Bangkok, Thailand 10530
University of Surrey[2], Guildford, United Kingdom GU2 7XH
s.srisuk@mut.ac.th

Abstract

In this paper, we present a general framework for image retrieval that is invariant to translation, rotation, scaling, local texture distortion and local geometrical deformation of an image. We first solve the problems in image database retrieval by transforming the image space into the Trace transform space. The Trace transform captures the key characteristics of an object by suppressing variations of an image, while maintaining discriminability. A robust feature from the Trace transform can be thought of as texture representations which can be used in an image retrieval. In addition, the redundant features are ignored by using only the flagged line in the Trace transform with the help of weighted Trace transform (WTT). The optimal parameter of the algorithm is discovered using reinforcement learning for which the within-class variance is minimized. We obtain a high indexing rate of 98.72% on 353 images.

Keywords: Trace transform, invariant features, image retrieval, reinforcement learning

1 Introduction

Image retrieval system requires abilities to search similar images in a possibly large database by using the contents of a queried image provided by user. In traditional method, the images in database have been queried by textual information. This is hard to achieve due to the different between meaning of the textual indexed in the database and the interpretation from the user. Several systems have been therefore proposed to overcome such problem by using contents from an image directly, e.g. texture [1, 2], shape [5, 6] and color [7] information. Gevers *et al.* [7] proposed a method for combining color and shape invariant information for indexing and retrieving images. The image representations were color and shape invariant histograms which were then combined as a single feature. Image retrieval based on color invariants provides very high retrieval accuracy whereas shape invariants yield poor discriminative power. There are however limited of the method proposed in [7] that it can not be used in grey-scale images. Muller *et al.* [6] described an image retrieval system which enables user to search a grey-scale image by presenting simple sketches. An image was represented by a Hidden Markov Model (HMM) by which it has been modified in order to obtain the features invariant to rotation and scaling. They presented a good result on their own hand tool data set. Torres-Mendez *et al.* [4] presented a new method for object recognition under translation, rotation and scaling conditions that has been applied to character recognition. A new coding of an object was proposed which describes its topological characteristics. The algorithm was tested on character recognition in which the test images were different in 17 sizes and 14 rotations without addressing the problem of local geometrical distortion of such characters. They achieved 98% correct recognition on their own data set. In [1], Petrou and her colleague presented a new tool for image processing, the Trace transform, which can be used to construct robust features invariant to rotation, translation and scaling. In this paper, we present a robust method for improving the performance of the traditional Trace transform. We aim at querying images under translation, rotation, scaling, local texture distortion, and local geometrical deformation (affine transformation).

The organization of this paper is as follows. Section 2 introduces a method for tracing line on an image. Section 3 presents a method to construct the weighted Trace transform. A new distance measure is presented in section 4. Section 5 describes a method for searching the optimal values of WTT using reinforcement learning with the training algorithm in section 6. Section 7 presents our experimental results. Finally, we conclude in section 8.

2 The Trace Transform

The Trace transform [1] is a new tool for image processing. To produce the Trace transform one computes a functional along tracing lines of an image. Each line is characterized by two parameters, namely its distance p from the centre of the axes and the orientation ϕ the normal to the line has with respect to the reference direction. In addition, we define parameter t along the line with its origin at the foot of the normal. The definitions of these three parameters are shown in Fig. 1. The image is transformed to another image with the Trace transform which is a 2-D function

$$g(\phi, p) = T(F(\phi, p, t)), \tag{1}$$

where $F(\phi, p, t)$ stands for the values of the image function along the chosen line. Parameter t is eliminated after taking the trace functional. The result is therefore a 2-D function of parameters ϕ and p and can be interpreted as another *image*. The resultant Trace transform depends on the functional we used. Let us denote $t_i \in t$ the sampling points along a tracing line defined by ϕ and p. Let us also denote by n the number of points along the tracing line. n may be varied depending on the length of the tracing line. The trace functionals used in our experiments are shown in table 1. The denomination $\text{median}_x\{x, w\}$ means the weighted median of sequence x with weights in the sequence w.

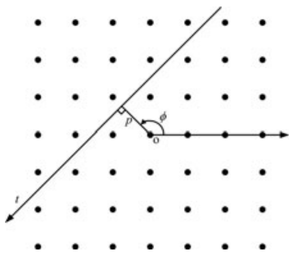

Fig. 1. Tracing line on an image with ϕ, p, and t.

In an image retrieval, one may have the problem of indexing an object that is rotated, scaled and translated with respect to the objects in the database. We show an original object and its rotated versions in Fig. 2. It can be seen that the Trace transform is shifted left or right by the corresponding amount of rotation. One can apply a matching algorithm on each column between an original and rotated versions of the Trace transform, i.e. along the ϕ direction, then shift the columns according to the best matching position to the Trace transform of the original object. The Trace transform of the rotated object can then be reconstructed to match the Trace transform of the original object. The procedure is indicated in Fig. 3. It is therefore seen that the rotation in image space is equivalent to the translation in the Trace transform space. Thus, it is easy to use a matching algorithm between the Trace transform of an original object and that of its distorted version to infer the rotation angle.

3 The weighted Trace Transform

The Trace transform allows us the define a new way for object representation. Fig. 4 shows the Trace transforms for

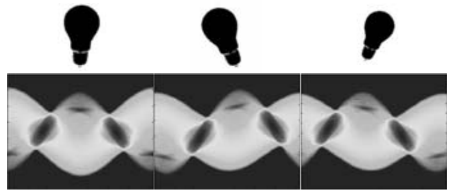

Fig. 2. An original object and its versions distorted by rotation. The corresponding Trace transform for each object version is shown in the bottom row.

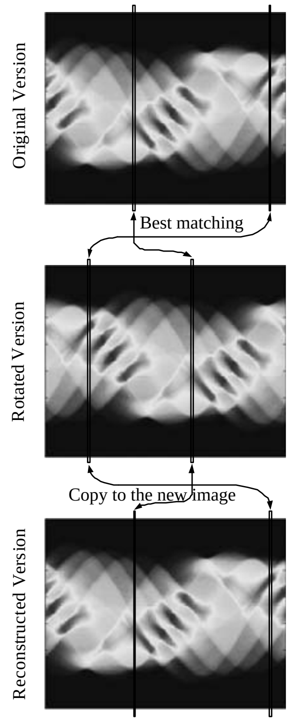

Fig. 3. The reconstruction of the Trace transform.

2 different fishes in the database and for 3 different images of each fish. We observe that there are subtle differences between the Trace transforms of different fishes that make them distinct, while the Trace transform of the same fish seems to retain its structure over the three different images.

Every point in the Trace representation of an image represents a tracing line. Here, we shall describe a method for weighting each tracing line according to the role it plays in recognizing the object. We need to find the persistence of the features locally in the Trace transform for each object. So, by selecting the features in the Trace transform which persist for a particular object, even when their local texture and local geometric change (e.g., the texture on the fish, a fin, etc.), we are identifying those scanning lines that are most important in the definition of the object. We refer this method as the weighted Trace transform. Let us define

Table 1. Examples of the Trace functionals T

No.	Trace Functionals	Details
1	$T(f(t)) = \int_0^{\infty} f(t)dt$	
2	$T(f(t)) = \left[\int_0^{\infty} \|f(t)\|^p \, dt\right]^q$	$p = 0.5, q = 1/p$
3	$T(f(t)) = \text{median}_t\{f(t), \|f(t)\|\}$	
4	$T(f(t)) = \int_{n+1}^{2n} \left\| \frac{d}{dt}\mathcal{M}(\mathbf{F}(t)) \right\| dt$	$\mathcal{M}$ is a median filtering operator, $\mathbf{F}(t) = [f(t) \; f(t) \; f(t)]$ so that the length of vector $\mathbf{F}(t)$ is $3n$
5	$T(f(t)) = \text{median}_{t^*}\{f(t^*), \|f(t^*)\|^{1/2}\}$	$f(t^*) = [f(t_{c*}) \; f(t_{c*+1}) \; \dots \; f(t_n)]$, $l = 1, 2, \dots, n$, $c = \text{median}_l\{l, f(t)\}$, $c*$ signifies the nearest integers of c
6	$T(f(t)) = \left\| \int_{c*+1}^{\infty} e^{ik\log(r)} r^p f(t)dt \right\|$	$p = 0.5$, $k = 4$, $r = \|l - c\|$, $l = 1, 2, \dots, n$, $c = \text{median}_l\{l, \|f(t)\|^{1/2}\}$, $c*$ signifies the nearest integers of c

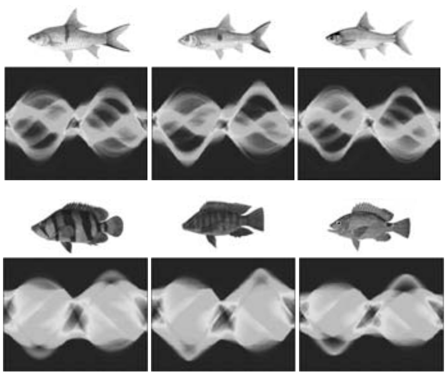

Fig. 4. An example result of the Trace transform for 2 different categories of fish. The corresponding Trace transforms for each fish are shown in the second row and fourth row, respectively. From left to right, the Trace transforms for training images 1 to 3 are shown in each column.

parameters for weighted Trace transform as follows: F_{ij} is training image j of an object i, g_{ij}^k the Trace transform for training image j of an object i using trace functional k, D_l^{ik} the lth absolute difference of the Trace transforms for an object i and functional k. $W^{ik}(\phi, p)$ the weighted Trace transform for an object i and functional k. Let us assume that we have 3 training images for each object, the weighted Trace transform can be described step by step as follows.

Step 1. Compute the Trace transform, g_{ij}^k, for object i, training image j and functional k.

$$g_{ij}^k(\phi, p) = T_k(F_{ij}(\phi, p, t)), \tag{2}$$

where T_k is the kth Trace functional.

Step 2. Calculate the differences of the Trace transforms for an object i and functional k.

$$\begin{aligned} D_1^{ik} &= \left|g_{i1}^k - g_{i2}^k\right|, \\ D_2^{ik} &= \left|g_{i1}^k - g_{i3}^k\right|, \\ D_3^{ik} &= \left|g_{i2}^k - g_{i3}^k\right|. \end{aligned} \tag{3}$$

step 3. Compute the weight matrix for the Trace transform for an object i, and functional k.

$$W^{ik}(\phi, p) = \begin{cases} 1, & \text{if } \sum_l \rho(D_l^{ik}(\phi, p)) = 0, \\ 0, & \text{Otherwise.} \end{cases} \tag{4}$$

where

$$\rho(x) = \begin{cases} 0, & \text{if } x \le \nu, \\ 1, & \text{Otherwise,} \end{cases} \tag{5}$$

and ν is some threshold. In other words, the weight matrix flags only those scanning lines which in all three images produced values for the Trace transform that differ from each other only up to a certain tolerance ν. When constructing features from the Trace transform we shall use only the flagged lines.

4 Distance measure

Let us explain how test claims are performed for weighted Trace transform. In the training phase, we use only 3 images for each object in order to search its optimal weight matrix. We then use the weight matrix for each object to identify the significant tracing line. In the test phase, the test image is compared with the images in the training set for all objects, and then the most m similar images are chosen as the reply by which the most similar image is maximized the matching confidence. Let us denote by $G_r \in \mathcal{T}$ the Trace transform for training images, $G_t \in \mathcal{N}$ the Trace transform for test images, where $\mathcal{T}$ is the set of Trace transform for all training images, $\mathcal{N}$ the set of Trace transform for all test images, with $\mathcal{T} \cap \mathcal{N} = \emptyset$. We obtain the most m similar images between test images and those images in the training set by

$$\mathcal{D} = \arg \max_{G_r \in \mathcal{T}}^{m} \{D(G_r, G_t)\}, \tag{6}$$

with
$D(G_r, G_t) =$

$$\exp\left(\frac{1}{n_\kappa}\sum_{\phi,p} W(\phi,p)\cdot \min_{\phi_t} |g_r^i(\phi,p) - g_t(\phi_t,p)|\right)^{-1}, \tag{7}$$

where g_r^i is the i^{th} Trace transform of images in training set, g_t the Trace transform for test image, and n_κ the total number of flagged line in weighted Trace transform. The superscript m in (6) indicates that we choose the m most maximum matching between the test image and the images in training set. We measure the minimum distance between $g_r^i(\phi, p)$ and $g_t(\phi_t, p)$ by which the Trace transform of the test image is scanned along the ϕ_t direction. This is the same procedure as indicated in Fig. 3. The weight $W(\phi, p)$ is computed using (4). $D(G_r, G_t)$ returns the highest value 1 when g_r^i and g_t are exactly the same. It is clear that the m most similar images are returned as the results in which the replied images are the nearest ones of the test object.

5 Reinforcement Learning

When we apply the weighted Trace transform to solve the problems of image retrieval, we use algorithms which rely on threshold ν (see equation (5)). The quality of the obtained result depends strongly on the values of these parameters. For WTT, we need to find the persistent features by varying parameter ν so that the within-class variance is minimized. We propose here to use reinforcement learning [3] so that the system learns the optimal values of weight computation parameter. Each parameter ν is represented by a set of Bernoulli quasilinear units, and the output of each unit is binary. Fig 5(a) depicts a Bernoulli quasilinear unit.

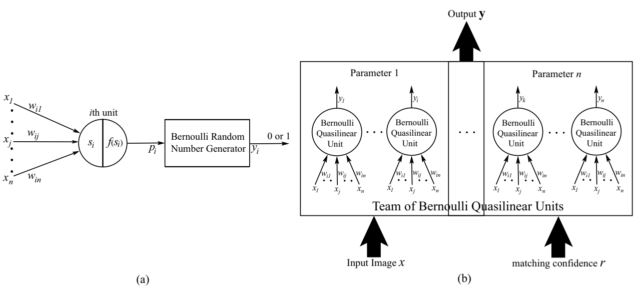

Fig. 5. Reinforcement learning. (a) a Bernoulli quasilinear unit. (b) Team of Bernoulli quasilinear units for learning parameters.

For any Bernoulli quasilinear unit, the probability that it produces a 1 on any particular trial given the value of the weight matrix W is

$$\Pr\{y_i = 1|W\} = p_i = f(s_i) = \frac{1}{1+e^{-s_i}}, \tag{8}$$

where $s_i = \sum_j w_{ij}x_j$ is the weighted summation of input values x_j to that unit. The weights w_{ij} are adjusted according to

$$\Delta w_{ij}^{(t)} = \alpha[(r^{(t)} - \overline{r}^{(t-1)})(y_i^{(t)} - \overline{y}_i^{(t-1)})]x_j - \delta w_{ij}^{(t)}, \tag{9}$$

where α is a learning rate, and δ a weight decay rate. Factor $(r^{(t)} - \overline{r}^{(t-1)})$ is called the reinforcement factor and $(y_i^{(t)} - \overline{y}_i^{(t-1)})$ the eligibility of the weight w_{ij}. $\overline{r}^{(t)}$ is the exponentially weighted average, or trace, of prior reinforcement values $\overline{r}^{(t)} = \gamma\overline{r}^{(t-1)} + (1-\gamma)r^{(t)}$, with $\overline{r}^{(0)} = 0$. $\overline{y}_i^{(t)}$ is an average of past values of y_i computed by the same exponential weighting scheme used for $\overline{r}$. Moreover, the "biased" RL is used here to force parameter ν to change slowly, when the confidence level of matching has exceeded the "bias" threshold τ_{b}. Let us suppose that parameter ν is represented by n Bernoulli units with m significant bits. The m most significant bits are forced to change by

$$y_i = \begin{cases} 1, & \text{if } p_i > 0.5, \\ 0, & \text{otherwise.} \end{cases} \tag{10}$$

6 The Training Algorithm

In this section, we present a closed-loop object recognition algorithm as shown in Fig. 6. Let us suppose that we have three training images. We pick one image for each object from the training set and thus create the reference set. The remaining two images for each object are defined as the tuning set because they will help us to tune the system to the best values of the weight computation parameter ν. The goal of our approach as shown in Fig. 6 is therefore to find the optimal parameter ν, for each object, for which the confidence level of matching between the reference and tuning sets is maximized.

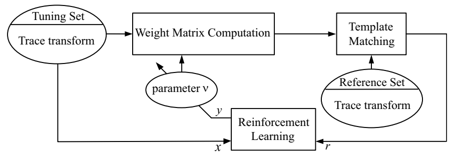

Fig. 6. Closed-loop object recognition system for weighted Trace transform.

Our approach is described by Algorithm 1, where N is the maximum number of iterations, τ_{r} a user-defined threshold, and τ_{b} the 'biased' RL threshold. The parameter values used in all our experiments were chosen as follows: $\alpha = 0.9$, $\delta = 0.01$, $\gamma = 0.9$, $\tau_b = 0.85$, and $\tau_r = 0.92$, respectively.

Algorithm 1: The closed-loop object recognition training algorithm for WTT

1. Initialise randomly weights w_{ij} ($w_{ij} \in [-1, 1]$)
2. Initialise matching confidence r^k to 0 (i.e. $r^k = 0, \forall k$)
3. For each image k in the tuning set do
   (a) Input tuning image x^k to the RL algorithm
   (b) Compute weight matrix with current parameter ν, i.e. perform the following test:
   if $r^k \geq \tau_{\mathrm{b}}$, obtain weight matrix parameter ν for the most significant bits m by (10); update the outputs of the remaining $n - m$ units using the Bernoulli probability distribution in (8)
   Otherwise ($r^k < \tau_{\mathrm{b}}$), acquire weight matrix parameter ν by (8)
   (c) Compute template matching using distance measure against reference set by (7)
   (d) Update each weight w_{ij} using r^k as reinforcement for the RL algorithm by (9)
4. Find the minimum matching confidence $\zeta = \min_k r^k$
5. Repeat step 3 until the number of iterations is equal to N or $\zeta \geq \tau_{\mathrm{r}}$

7 Experimental Results

We describe an object database we used in this paper and then present an object indexing results under translation, rotation, scaling and local texture and geometric distortions. Fig. 7 shows some examples of the objects in our database consisting of 353 images. For rotation and scaling problems, the objects for testing were generated by applying a random scaling and rotation factors to the objects, which was distributed within $[0.5, 1.5]$ and $[0, 360]$ degrees, respectively. The positions of the object were also randomly placed in the image which was used to test the translation problem. The extreme cases are the local texture and geometric distortions of the objects such as the different textures of the fish, the types of fin, the different claws of a crab, etc. Such problems are more difficult when the distorted object is also generated by applying scaling and rotation factors in a random way for which the distorted objects are used to test the rotation, scaling and local distortion problems.

Fig. 8 shows the Trace transform for the same category of crab with different texture and claw. It can be seen as the problem of local texture and geometric distortions of the objects. The second row of Fig. 8 shows the Trace transform for trace functional 1, the third row for trace functional 2, and so on. From the inspection of the Fig. 8, the different

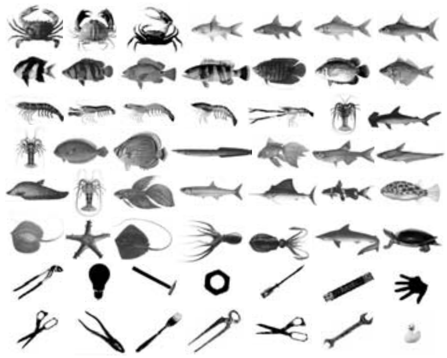

Fig. 7. Some examples of the objects in database.

images of the same category of crab were produced as the same image in the Trace transform space. This is a robust feature derived from the Trace transform which can be used to recognize objects under nonlinear distortion. It should be pointed out that the results from all trace functionals may be thought of as texture representations which can be used as a robust feature for object indexing. We however use a trace functional for a specific object. On the other hand, the choice of the trace functionals we used for a particular object depends on the classification ability in which the between-class variance is maximized with respect to the minimized within-class variance. In addition, by using the technique we proposed, WTT+RL, the redundant features are ignored in the computational process. This helps us maximize the matching confidence between objects of the same category, even when their local texture or local geometric changes.

The parameters of the Trace transform were chosen as follows: the sampling parameter $p = 250$, i.e. number of tracing lines in each direction, and parameter $\phi = 360$, i.e. we traced the line in all directions. These parameters were kept constant for all calculations of images. Parameter p was kept constant for which the scaling problem can be decreased, i.e. the resolution of the 'image' in the Trace transform space is similar for different sizes of the objects.

Fig. 9 shows the queried object and the best 5 corresponding images in the database. The first column is the queried object and the following columns are the similar images arranged in order of similarity. The similarity measure in (6) as described in section 4 was used to find the best matching in the training set. It should be noted that the queried object is the distorted version of the original object by which it was randomly generated with respect to translation, rotation and scaling. Some images of the same category are different by local texture information and local geometrical distortion, for example, the fish, crab, etc. The first 3 rows in Fig. 9 are the examples of the queried objects under local texture and geometrical distortions that

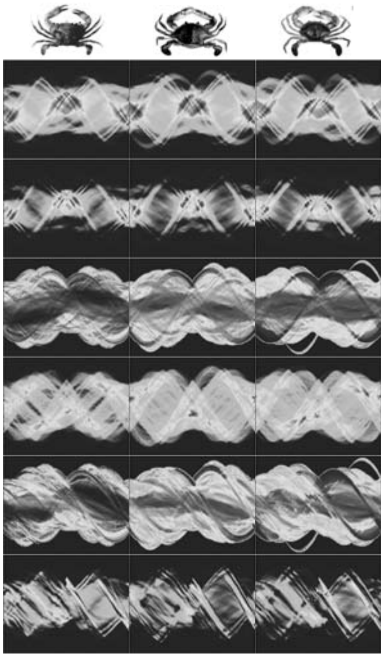

Fig. 8. The Trace transform for the same category of crab computed from 6 different trace functionals from table 1. The second row shows the Trace transform for trace functional 1, the third row for trace functional 2, and so on.

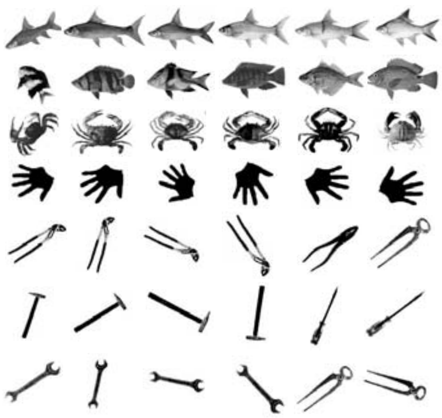

Fig. 9. Queried objects (first column) and the most similar images in training set (following columns) arranged in order of similarity.

has been shifted, rotated and scaled in some random way. The remaining rows are queried objects under translation, rotation and scaling. We can summarize from the Fig. 9 that our proposed method performs very well and that the images can be retrieved without knowing rotation angle of the desired object which is one of the important properties of image database retrieval system. Table 2 compares the performance of image retrieval using WTT and WTT with RL in which the corrected objects were included among the best 5 response images.

Table 2. Performance comparison of image retrieval.

Methods	Indexing Rates (%)
WTT	96.78
WTT+RL	98.72

*** WTT-weighted Trace transform, RL-reinforcement learning

8 Summary and Conclusions

We have presented a new approach for indexing distorted objects in order to solve the problems in application of image retrieval. This includes the problems of rotation, translation, scaling, local texture distortion and local geometrical deformation. We solve these problems by transforming the image space into the Trace transform space. A robust feature from the Trace transform can be thought of as texture representations which can be used in an image retrieval. The Trace transform captures the key characteristics of an object by suppressing variations of the image, while maintaining discriminability. In addition, the redundant features are ignored by using only the flagged line in the Trace transform with the help of WTT and reinforcement learning. We obtained a high indexing rate of 98.72% on 353 images.

References

[1] A. Kadyrov and M. Petrou. The Trace Transform and Its Applications. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 23, No. 8, pp. 811-828, August 2001.

[2] C. Wolf, J.M. Jolion, W. Kropatsch and H. Bischof. Content based Image Retrieval using Interest Points and Texture Features. *In Proc. of International Conference on Pattern Recognition*, pp. 4234-4237, 2000.

[3] J. Peng and B. Bhanu. Closed-Loop Object Recognition using Reinforcement Learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 20, No. 2, pp. 139-154, Feb 1998.

[4] L. A. Torres-Mendez, J. C. Ruiz-Suarez, L. E. Sucar and G. Gomez. Translation, Rotation, and Scale-Invariant Object Recognition. *IEEE Trans. on Systems, Man, and Cybernatics-Part C: Applications and Reviews*, Vol. 30, No. 1, pp. 125-130, 2000.

[5] R. Distasi, M. Nappi, M. Tucci and S. Vitulano. CONTEXT: A Technique for Image Retrieval Integrating CONtour and TEXTure Information. *In Proc. of International Conference on Image Analysis and Processing*, pp. 224-229, 2001.

[6] S. Muller, S. Eickeler and G. Rigoll. Image Database Retrieval of Rotated Objects by User Sketch. *In Proc. of IEEE Workshop on Content-Based Access of Image and Video Database*, pp. 40-44, 1998.

[7] T. Gevers and A. W.M. Smeulders. PicToSeek: Combining Color and Shape Invariant Features for Image Retrieval. *IEEE Trans. on Image Processing*, Vol. 9, No. 1, pp. 102-119, 2000.