AFFINE-INVARIANT FOURIER DESCRIPTORS AND THEIR APPLICATION IN A NUMBER PLATE RECOGNIZING SYSTEM

Nguyen Thanh Hai*, Pham The Long, Nguyen Cong Dinh

Le Qui Don University

Received 28 March 2008

ABSTRACT

This paper introduces a method for describing partial shapes that are affine-invariant. The proposed method facilitates the extraction of descriptors, which is effective for opened-curves and invariant to different viewpoints. This method helps to improve the performance and quality of real world object image recognizing systems. The proposed method was tested in a number plate recognition system.

1. INTRODUCTION

The problems of object recognition in poor view conditions remain one of the great challenges for researchers in the field of image processing and recognition. In many real-world applications, digital image objects are very difficult to separate due to a number of different reasons such as: objects stick together or overlap with each other, objects are hidden some of their parts by the geographical location, objects are embedded in a complex background... In such cases, it is not reasonable to solve the problems by the traditional approaches [6, 7, 9, 11, 12] which usually solve the problem in two stages: separating objects then recognizing the separated objects. In order to get meaningful results in solving such problems, it is usually required a method of describing and recognizing to some parts of objects in the images [2, 8, 13].

The Fourier transform has always been a powerful tool for shape representation of digital image objects [6, 7]. Fourier descriptors are rather simple to implement, and involve less computations by either using fast Fourier transform (FFT) or using truncated Fourier transform algorithms. The resultant Fourier descriptors are also compact, which makes matching or recognizing process relatively simple. In [1, 2], the authors proposed an effective method for using Fourier transform to describe partial shapes through opened-curves. The proposed method gives us the descriptors that are invariant to some geometrical transforms such as translation, rotation, and scale (geometrical similarity invariant).

In some real-world and commercial image processing systems, digital images of objects may be taken from different view points (for example, taken by some cameras located at different positions). In such cases, the images of an object may not be identical to each other but rather being related through some affine transformations since the viewpoint transformations can be described by affine transformations [11]. Consequently, it is essentially meaningful to have methods for describing partial shapes that are invariant to affine transformations. In [12], Sait S.

* Corresponding author e-mail: nthhai@gmail.com

and Mustafa U. proposed a method for affine invariant fitting of curves using Fourier descriptors but it is only effective for closed-curves (in [1] we have shown that the traditional Fourier descriptors are not suitable for opened-curves).

In this paper, the authors introduce a method for describing opened-curves (use to represent partial shapes) and that descriptors are invariant to affine transformations. The proposed method is applied to build a system of vehicle plate number recognition.

The rest of the paper is organized as follows. In Section II, the construction of affine-invariant Fourier descriptors for opened-curves are given with some experimental results. A real world system of car plate number recognition is introduced in Section III. The conclusions and future works are drawn in Section IV.

2. CONSTRUCTION OF AFFINE-INVARIANT FOURIER DESCRIPTORS FOR OPENED-CURVES

2.1 Some definitions of curves and their representation methods

Definition 1 [1]**.** [L^{ε} -neighbour]: L^{ε} -neighbor set of a point $(x,y)\in R^2$ is:

$$L^{\varepsilon}=\left\{(u,v)\in R^2 \mid (u,v)\neq(x,y),\sqrt{(u-x)^2+(v-y)^2}\leq\varepsilon\right\} \tag{1}$$

Definition 2 [1]. [Curve, closed-curve, opened-curve]:

Curve: A curve (L^{ε} -curve) is a finite sequence of points in 2D plane $g_0, g_1, ..., g_n$ such that g_i and g_{i+1} are L^{ε} -neighbours of each other. The curve is denoted as $g_0 g_1 ... g_n$. We call, in a curve $g_0 g_1 ... g_n$, g_0 is the starting point and g_n is the ending point.

Closed-curve, opened-curve: If the starting point and the ending point of a L^{ε} -curve $g_0 g_1 ... g_n$ are L^{ε} -neighbour points then the curve is said to be a closed-curve, denoted as $[g_0 g_1 ... g_n]$. Otherwise, the L^{ε} -curve is an opened-curve, denoted as $(g_0 g_1 ... g_n)$.

A closed-curve may contain some opened-curves and conversely, an opened-curve may contain some closed-curves.

2.2 Affine-invariant curve normalization

In order to have affine-invariant descriptors for opened-curves, the opened-curve is first normalized through some steps of orthogonalization [5] and then a step of rotation for affine-invariant. After that, the Fourier transform is applied to the curve for obtaining a symmetrical representation of the normalized curve and for producing some significant Fourier coefficients that construct the descriptors for the original opened-curve (More details could be seen in [2]).

Suppose that $\mathbf{T}=(t_0 t_1 ... t_N)$ is an opened-curve, the following steps could be done to normalize **T**:

a) Orthogonalization procedure

For each curve **T**, the (*p*,*q*)-order moments are calculated as:

$$m_{pq}(\mathbf{T}) = \frac{1}{N+1}\sum_{i=0}^{N} x_i^p y_i^q \tag{2}$$

Without loss of generality, it is assumed from here that **T** does not represent a line segment, so that $m_{20}(\mathbf{T}) \neq 0$ and $m_{02}(\mathbf{T}) \neq 0$. In such a case, normalization steps involving division by these quantities may be omitted. The orthogonalization procedure is similar to [5] that include the following steps:

Step 1. The center of gravity of the curve is normalized so as it coincides with the origin:

$$\mathbf{x}_1 = \mathbf{x} - m_{10}(\mathbf{T}),\ \mathbf{y}_1 = \mathbf{y} - m_{01}(\mathbf{T}),\ \mathbf{T}_1 = [\mathbf{x}_1\ \mathbf{y}_1]^T \tag{3}$$

where $\mu_x = m_{10}(\mathbf{T})$, $\mu_y = m_{01}(\mathbf{T})$.

Here, for simplicity the addition or subtraction of a scalar from a vector (or from a row vector of a matrix) denotes addition or subtraction of all its elements.

Step 2. The curve is scaled horizontally and vertically so that its second-order moments become equal to one:

$$\mathbf{x}_2 = \sigma_x \mathbf{x}_1,\ \mathbf{y}_2 = \sigma_y \mathbf{y}_1,\ \mathbf{T}_2 = [\mathbf{x}_2\ \mathbf{y}_2]^T \tag{4}$$

where $\sigma_x = 1\big/\sqrt{m_{20}(\mathbf{T}_1)}, \sigma_y = 1\big/\sqrt{m_{02}(\mathbf{T}_1)}$.

Step 3. The curve is rotated counterclockwise by $\theta_0 = \pi/4$:

$$\mathbf{T}_3 = \mathbf{R}_{\pi/4}\mathbf{T}_2 = \frac{1}{\sqrt{2}}\begin{bmatrix} \mathbf{x}_2 - \mathbf{y}_2 \\ \mathbf{x}_2 + \mathbf{y}_2 \end{bmatrix} \tag{5}$$

where $\mathbf{R}_\theta$ is a 2×2 matrix corresponding to a counterclockwise rotation by θ radians.

Step 4. Finally, the curve is scaled again, just as in step 2:

$$\mathbf{x}_4 = \tau_x \mathbf{x}_3,\ \mathbf{y}_4 = \tau_y \mathbf{y}_3,\ \mathbf{T}_4 = [\mathbf{x}_4\ \mathbf{y}_4]^T \tag{6}$$

where $\tau_x = 1\big/\sqrt{m_{20}(\mathbf{T}_3)}, \tau_y = 1\big/\sqrt{m_{02}(\mathbf{T}_3)}$.

In brief, the normalized curve $n_a(\mathbf{T}) \equiv \mathbf{T}_4$ can also be written as

$$n_a(\mathbf{T}) = \mathbf{N}(\mathbf{T})(\mathbf{T} - \mu(\mathbf{T})) = \frac{1}{\sqrt{2}}\begin{bmatrix} \tau_x & 0 \\ 0 & \tau_y \end{bmatrix}\begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}\begin{bmatrix} \sigma_x & 0 \\ 0 & \sigma_y \end{bmatrix}\left(\begin{bmatrix} \mathbf{x} - \mu_x \\ \mathbf{y} - \mu_y \end{bmatrix}\right) \tag{7}$$

where $\mu(\mathbf{T}) = [m_{10}(\mathbf{T})\ m_{01}(\mathbf{T})]^T$ and $\mathbf{N}(\mathbf{T})$ denotes the 2×2 *normalization matrix* of **T**.

b) The property of invariance to translation, scaling, and skewing

Let us now consider two curves **T** and **T'** related through an affine transformation:

$$\mathbf{T}' = \mathbf{AT} + \mathbf{t} = \begin{bmatrix} \mathbf{x}' \\ \mathbf{y}' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}\begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \tag{8}$$

where matrix **A** is assumed to be of full rank (in the degenerate case of det**A** = 0, any 2-D data set is mapped onto a line and normalization is impossible).

Proposition 1 [5]**.** *If two curves* **T** and **T'** *are related through an affine transformation, there* *exists an orthogonal* 2×2 *matrix* **Q** *such that:*

$$n_a(\mathbf{T}') = \mathbf{Q}n_a(\mathbf{T}) \tag{9}$$

Thus, the above normalization procedure reduces affine transformations to orthogonal transformations that may contain only rotation and/or reflection (depending on whether det**Q** = 1 or det**Q** = −1), and therefore normalized curves are invariant to translation, scaling, and skew transformations. So, the transformations are affine invariant completely after the steps of rotation and reflection normalization.

c) Invariance to reflection

The horizontal and vertical reflection is normalized according to the third-order moments of $\mathbf{z}_1$:

$$\mathbf{x}_5 = v_x\mathbf{x}_4,\ \mathbf{y}_5 = v_y\mathbf{y}_4,\ \mathbf{T}_5 = [\mathbf{x}_5\ \mathbf{y}_5]^T \tag{10}$$

where $v_x = \text{sgn}m_{12}(\mathbf{T}_4)$, $v_y = \text{sgn}m_{21}(\mathbf{T}_4)$, sgn denotes the signum function.

Denotes $n_{av}(\mathbf{T}) \equiv \mathbf{T}_5$. From proposition 1 we have the following corollary:

Corollary 1: *If two curves* **T** and **T'** *are related through an affine transformation then* $n_{av}(\mathbf{T}')$ *is an image of* $n_{av}(\mathbf{T})$ *by a rotation* [5].

2.3 Affine-invariant Fourier descriptors for opened-curves

We define $n_{av}(\mathbf{T}) = (g_0 g_1 ... g_N)$. In order to build descriptors for opened-curve **T** using Fourier transform, according to [1], first a closed-curve **G** that is a symmetrical representation of **T** through an axial symmetry in which the symmetrical axis is $g_0 g_N$ is created.

Let g_{2N-k} ($k = \overline{0, N}$) is the symmetrical point of g_k by the line $g_0 g_N$. Then, it can be shown that $\mathbf{G} = [g_0 g_1 ... g_N g_{N+1} ... g_{2N-1}]$ is a closed-curve (Figure 1c). We say that **G** is a closed-representation (affine-invariant) of opened-curve **T**. The line $g_0 g_N$ is called the symmetrical axis of the curve **G**.

Remarks: (*i*) $g_{2N} \equiv g_0$; (*ii*) g_k, g_l are symmetrical by $g_0 g_N$ if $k + l = 2N$.

Denotes: $g_k = (x_k, y_k)$, $z_k = x_k + iy_k$; $k = \overline{0, 2N-1}$, i – is complex unit, $i^2 = -1$.

Let $\boldsymbol{\xi} = \{\xi_0, \xi_1, ..., \xi_{2N-1}\}$ is a complex representation of **G** [1]. Apply the Fourier tranform to elements of $\boldsymbol{\xi}$ and get out the Fourier cofficients that calculated as follows:

$$F_m = \sum_{k=0}^{2N-1} \xi_k e^{-i2\pi km/2N}, m = 0,1,...,2N-1 \tag{11}$$

Then, **G** can be described by the set of Fourier cofficients:

$$F = \{F_0, F_1, F_2, ..., F_{2N-1}\} \tag{12}$$

Elements of $\boldsymbol{\xi}$ can be received from F through the inverse Fourier transform:

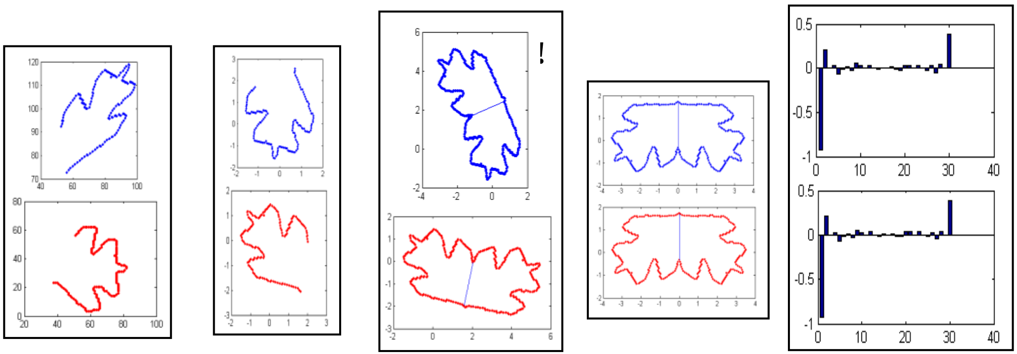
$$\xi_k = \frac{1}{2N}\sum_{m=0}^{2N-1} F_m e^{i2\pi km/2N}, k = 0,1,\dots,2N-1 \tag{13}$$

Denotes Real(λ), Imag(λ) respectively are the real and imaginary parts of a complex number λ. The following proposition allows to reduce the numbers of extracting Fourier descriptors.

Proposition 2 [1]: If $g_0 g_N$ is in horizontal axis then $\text{Imag}(F_m) = 0, \forall m = \overline{0,2N-1}$. Dually, if $g_0 g_N$ is in vertical axis then $\text{Real}(F_m) = 0, \forall m = \overline{0,2N-1}$.

Subsequently, the following operators are performed to get completely affine-invariant to any affine transformation:

- *Invariant to translation*: $F_0 := 0$.
- *Invariant to reflection*: $F_n := F_n \cdot K_q$ where $K_q = e^{i\varphi}$ is a rotate coefficient (φ is the angle formed by the line $g_0 g_N$ and vertical axis).



a) Opened-curves (from differrent viewpoints)

b) Orthogonalization

c) Symmetric

d) Rotation invariant

f) Fourier descriptors

Fig. 1: *Results of affine-invariant curve normalization and Fourier descriptors .*

In the Fourier transform, the most meaning cofficients are those at the beginning and the end of the Fourier cofficients sequence *F*. On the other hand, the symmetrical axis of **G** lies on vertical axis by the normalization**.** This and proposition 2 allow us select a set *H* that contains descriptors for opened-curve **T** as follows:

$$H(\mathbf{T}) = \{\text{Imag}(F_1),\dots,\text{Imag}(F_M),\text{Imag}(F_{2N-M}),\dots,\text{Imag}(F_{2N-1})\} \tag{14}$$

Remark: The number of elements of *H* is 2*M*, it is independent of *N*.

From consequence 1 and the properties of axial symmetry, it is trivial that:

Proposition 3. Let *H*(**T**) be defined as (14), then *H*(T) is completely invariant to any affine transformation on **T**.

Definition 3. [Partial Fourier set, Fourier features vector]

The set $H(\mathbf{T})$ defined as in (14) is called a partial Fourier affine invariant set of the opened-curve **T**.

Define $H(\mathbf{T}) = \{h_1^T, h_2^T, ..., h_K^T\}$. Vector $h^T = (h_k^T, k = 1,2,.., K)$ is called a Fourier feature vector (affine-invariant) of the opened-curve **T**.

Definition 4. [Matching measure between two curves]

Suppose that $\mathbf{T}$ and $\boldsymbol{\tau}$ are opened-curves that have feature vectors h^T and h^τ respectively. There following measures could be used for estimating the degree of matching between $\mathbf{T}$ and $\boldsymbol{\tau}$.

- Normalized Euclidean distance measure:

$$\varepsilon_{Euclide}(\mathbf{T}, \boldsymbol{\tau}) = \frac{\left\| h^T - h^\tau \right\|}{\left\| h^T \right\| + \left\| h^\tau \right\|} = \frac{\sqrt{\sum_{k=1}^{2M} \left(h_k^T - h_k^\tau \right)^2}}{\sqrt{\sum_{k=1}^{2M} \left(h_k^T \right)^2} + \sqrt{\sum_{k=1}^{2M} \left(h_k^\tau \right)^2}} \tag{15}$$

- Maximal distance measure:

$$\varepsilon_{\max}(\mathbf{T}, \boldsymbol{\tau}) = \max \left\{ \left| h_k^T - h_k^\tau \right|, k = \overline{1,2M} \right\} \tag{16}$$

- Hausdorff distance measure:

$$\varepsilon_{Hausdorff}(\mathbf{T}, \boldsymbol{\tau}) = \max \left\{ \min \left\{ \left| h_i^T - h_j^\tau \right|, i = \overline{1,2M} \right\}, j = \overline{1,2M} \right\} \tag{17}$$

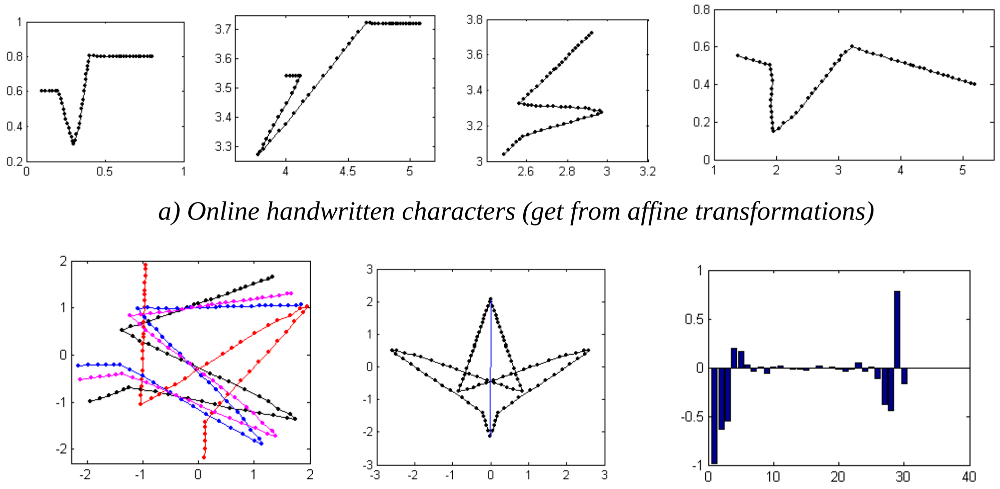
The smaller the above measures the more similar curves $\mathbf{T}$ and $\boldsymbol{\tau}$ are in respect to affine-invariance.

2.4. Some illustrative examples and tests

a) An example of online handwritten characters

In the first example, we sampled from an online-stroke to get an opened-curve (the square root symbol on the left most of figure 2a), then it was applied with some affine transformations in formula (8) to get other opened-curves (figure 2a). It can be seen that other affine opened-curves still represent the square root symbol. Figure 2b) illustrates the affine-invariant extracting features that are the results of our proposed method (four symbols in figure 2a have the same extracting features).

Figure 3a) shows two digital images of digit 5 that from one car plate and that they were taken from different viewpoints (see section III for more details). These images were then processed to produce the described opened-curves (Figure 3b). Then some affine-invariant features were extracted from the received opened-curves by the proposed method (the extracting features are shown in Figure 3c). The extracting features of two images are plotted in the same coordinates (Figure 3d), this shows that the two extracting feature sets are very similar (where the normalized Euclidean distance matching measure between them are 0.0494). The affine-invariant features and geometrical similar invariant features (that are extracted according to [1]) are shown in Figure 4. The results show that the affine-invariant feature extracting method proposed in this paper is more effective than the geometrical similar invariant feature extracting method proposed in [1], in which the comparison are carried out in this car plate number recognizing context.



a) Online handwritten characters (get from affine transformations)

b) Normalization and extracting features

Fig. 2: *Example about online handwritten characters.*

b) Example of characters on a car plate

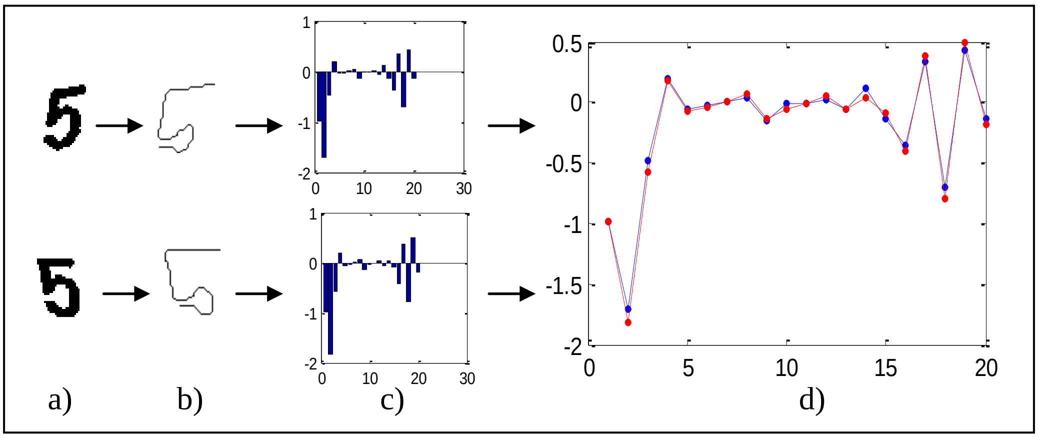


Fig. 3: *Tests on car plate character images .*

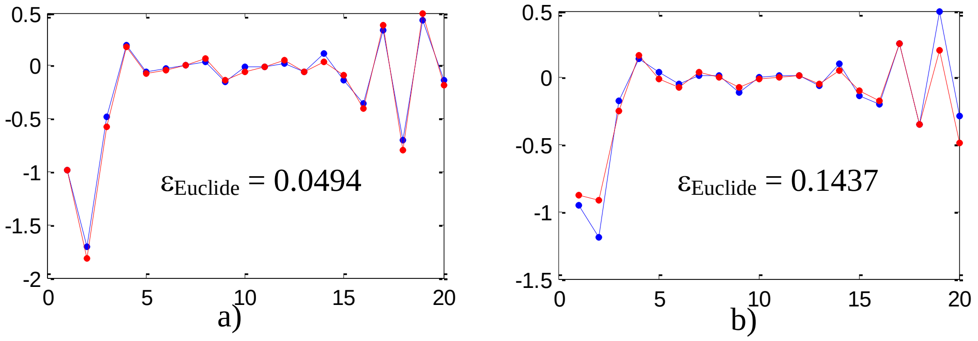


Fig. 4: *Graph of extracting features: a) affine invariant, b) similar invariant.*

c) Compare the different matching for some car plate character images

In this test, we got some character images in the same car plate from two different viewpoints, then sampled to get opened-curves and extract the affine-invariant and similar-invariant [1] features. The matching measures between extracting features are calculated as fomulars (15), (16) and presented the results into tables 1 and 2. The numerical results in the tables 1, 2 show that with respect to the same character object, the affine-invariant matching measures is much smaller than geometrical similar-invariant matching measures. This means that the affine-invariant features are better than the geometrical similar-invariant features in the context of car plate number recognition. Furthermore, tables 1 and 2 also show that by the affine-invariant extracting the matching measures of different character images are much bigger than those of the similar character images. This is a promising pretext for solving confidential problems in pattern recognition.

Table 1: *Matching measures using normalized Euclidean distances.*

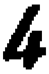
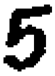
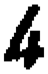
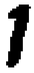
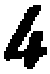
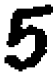
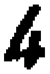
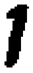
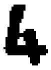
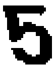
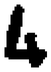
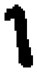
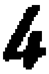
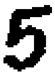
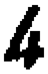
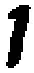
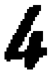
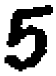
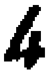
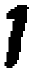
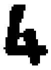
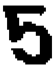
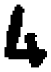
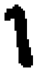
	Affine-invariant				Geometrical similar-invariant			
								
	0.0367	0.6014	0.0420	0.9010	0.0803	0.4465	0.0861	0.7221
	0.5808	0.0494	0.5714	0.7991	0.4730	0.1437	0.4618	0.6873
	0.0291	0.5959	0.0182	0.8961	0.0820	0.4438	0.0837	0.7167
	0.8921	0.8008	0.8912	0.0824	0.7241	0.5777	0.7246	0.0827

Table 2. *Matching measures using mmaximal distances.*

	Affine-invariant				Geometrical similar-invariant			
								
	0.0323	1.5264	0.0338	1.3181	0.1464	0.6108	0.1416	1.1142
	1.4388	0.1157	1.4012	1.7829	0.8227	0.2819	0.7751	1.3739
	0.0452	1.5093	0.0211	1.3229	0.1623	0.5950	0.1575	1.0984
	1.3289	1.9407	1.3285	0.1218	1.2339	1.0786	1.2292	0.1233

3. BUILDING A CAR PLATE NUMBER RECOGNITION SYSTEM

In this section, a real world system of car plate number recognition is introduced. It was built to manage cars go in and out of an office. There were some cameras (corresponding to some paths) conected to a computer. The system receives images from cameras, analyses images to know the moment that a car going in or out, processes and recognizes the car plate number, and make reasonable decisions for registered cars (office's cars, cars of people who are working in that office...) or unregistered cars (customer's cars...). Figure 6 is the interface of this system. The operation diagram of the system is shown in figure 5.

Analysing some basical functions in the diagram.

- *Detecting in/out cars:* There were some solutions for detecting a car by using some hardwares such as magnetic rings, sensors... In this system, we used a software solution for detecting a car through some image processing operators (using some detected technology and calculating the size of moved objects in images [4]). The proposed solution contributed to not only decrease hardware costs but also warranty and maintenance costs.

- *Separate characters in the car plate:* The characters can be separated from the image that contains a plate through a reasonable threshold and calculating some received binary features of binary image objects. Those technologies are discussed detail in [3].

- *Representation character images by opened-curves*: The curves for describing binary image objects can be received from some technologies of detecting their boundaries, boundary appropxiates and some feature points in the curves (that have extreme curvatures) [2].

- *Recognition*: We use a nearest neighbor method (a simple method) for recognition. The features of character images in car plates are collected from real cars. They form a learning set: $\{p_1, t_1\}, \{p_1, t_2\}, \ldots \{p_Q, t_Q\}$, where, p_k is affine-invariant feature extracting vector, t_k is a corresponding target. In order to recognize an input *p* (an affine-invariant feature extracting vector of character image needs to recognize), we do the follows matching fomular:

$$\|p - p_k\| = \min\left\{\|p - p_i\|, i = \overline{1, Q}\right\}$$

$\Rightarrow$ *p* is character t_k, where $\|.\|$ is any matching measure from fomulars (15)÷(17).

A remarkable point in this built system is that the cameras are located at different positions because of actual space conditions so the car images are taken from different viewpoints. Figures 7a, b, c are images of the same car that are taken from three moment correspoding three different cameras. We can see that the car plate character images have different shapes in which they do not geometrical similar each other but they are related through an affine transformation. The proposed method in section II of this paper has already used very effectively in this system.

The performace of exact recognition in this system is more than 96% on the total cars go in and out the office every day. The number of elements of learning set is 360 corresponding 10 digit characters and 26 upper-case letters A, B, C,..., Z (average about 10 for a character) that were collected from cars go through this system in some days.

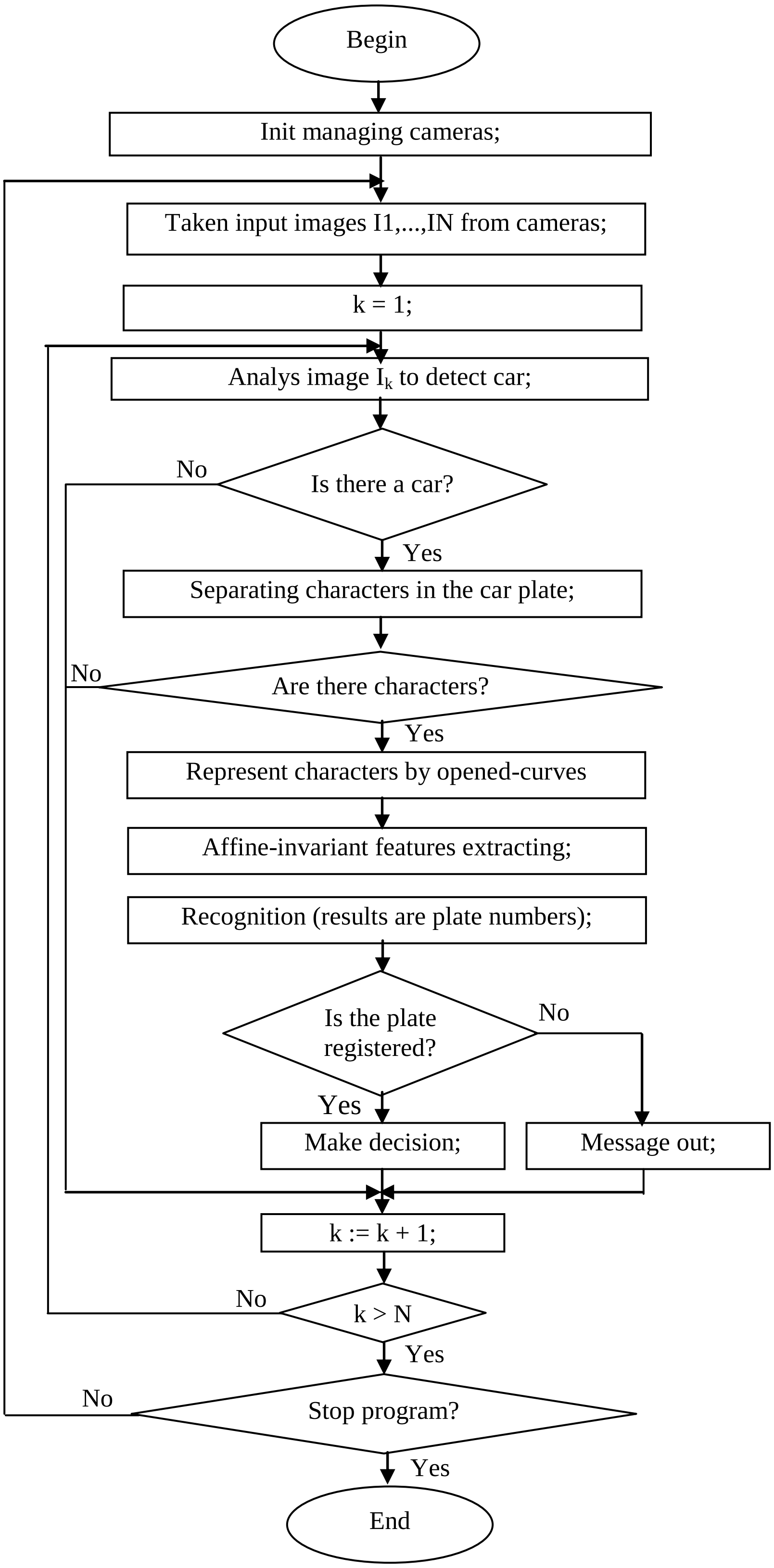
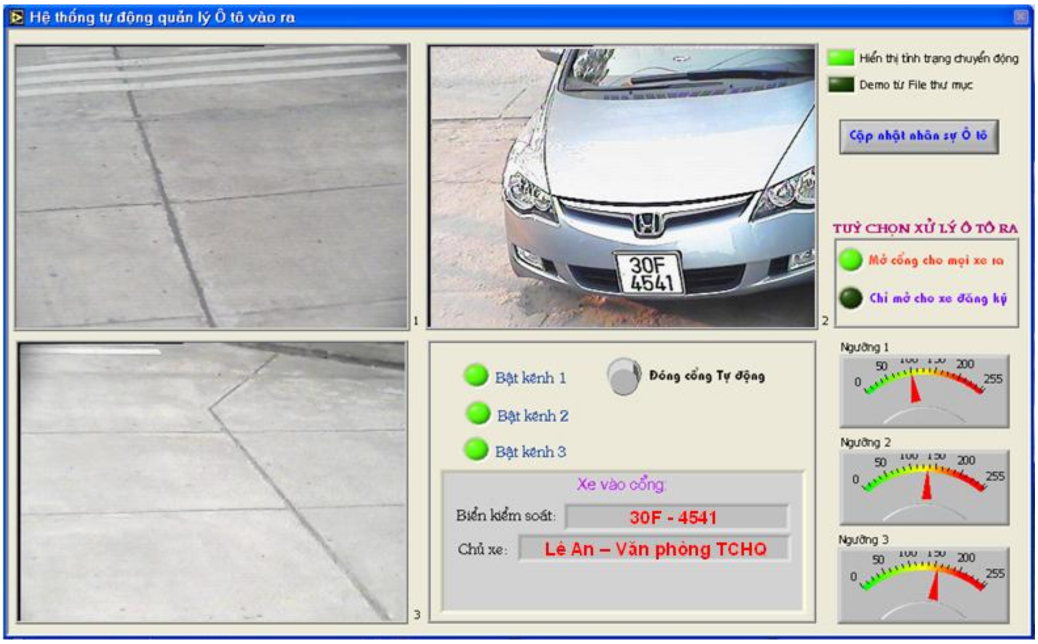

Begin
Init managing cameras;
Taken input images I1,...,IN from cameras;
k = 1;
Analys image I_k to detect car;
Is there a car?
No
Yes
Separating characters in the car plate;
Are there characters?
No
Yes
Represent characters by opened-curves
Affine-invariant features extracting;
Recognition (results are plate numbers);
Is the plate registered?
No
Yes
Make decision;
Message out;
k := k + 1;
k > N
No
Yes
Stop program?
No
Yes
End

Fig. 5. *Diagram of system of multi-kernels management.*

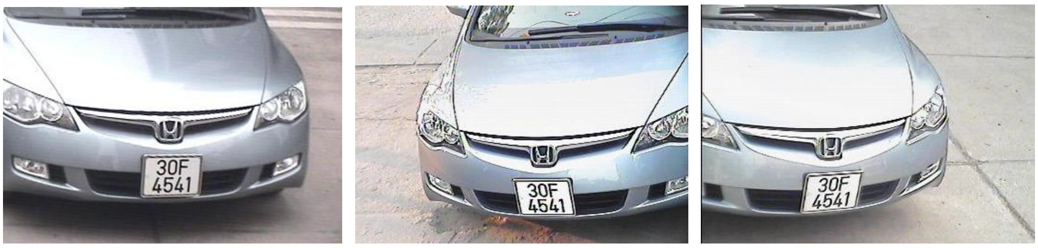


Xe vào cổng

Biển kiểm soát: 30F - 4541

Chủ xe: Lê An – Văn phòng TCHQ

***Fig. 6:** Interface of the system.*



a) Image taken from kernel 1 *b) Image taken from kernel 2* *c) Image taken from kernel 3*

***Fig. 7:** Images taken from diffrent viewpoints.*

4. CONCLUSION AND FUTURE WORK

The paper proposed a method for describing opened-curves corresponding to partial shapes that are invariant to affine transformations. The method allows to normalize many affine deformations of an object image shape into a shape to get out the same feature extracting vector. In many cases of pattern recognition, this allows to decrease the number of elements of the learning set and improve the peformance. The proposed method was used in a car plate number recognizing system with managing multi-kernels.

Besides, the proposed method could be integrated with some other artificial intelligence technologies such as neural network or fuzzy logics. The results of this papers can be developed for solving other difficult problems such as online handwritting recognition, signature recognition, partial recognition of 2D, 3D objects... These will be our future work .

REFERENCES

1. Nguyen Thanh Hai, Pham The Long, and Nguyen Cong Dinh (2007), A method of building Fourier descriptors for partial shapes and some properties, Submitted to Journal of Computer Science and Cybernetics. (in press) (in Vietnamese).
2. Nguyen Thanh Hai, Pham The Long, and Nguyen Cong Dinh (2007), Building applications of matching and recognizing object images based on the Fourier descriptors for partial shapes, Submitted to Journal of Science and Technology. (in press) (in Vietnamese).
3. Nguyen Thanh Hai, Nguyen Huy Hoang, and Pham The Long (2007), On a neural-image system for applications of processing and image object recognition, Submitted to the special issue of Telecommunications and Information Technology, Posts and Telecommunications Journal (in press) (in Vietnamese).
4. Nguyen Thanh Hai and Nguyen Cong Dinh (2006), Building a moved target tracking system integrated with image recognizing technique and applying in supervising and traffic monitoring, Proceeding ICT'rda III of the Third National Symposium on Research, Development and Application of Information and Communication Technology (ICT.rda'06), Hanoi-5/2006, p. 286-295. (in Vietnamese).
5. Yannis Avrithis, Yiannis Xirouhakis, and Stefanos Kollias (2001), Affine-invariant curve normalization for object shape representation, classification, and retrieval, Machine Vision and Applications, 13: 80-94, @Springer-Verlag.
6. Zhang D.S. and Lu G. (2002), A comparative study of Fourier descriptors for shape representation and retrieval, Proceedings of the Fifth Asian Conference on Computer Vision (ACCV02), Melbourne, Australia, January 22-25, 2002, pp. 646-651.
7. Zhang D.S. and Lu G. (2004), Review of shape representation and description techniques, Pattern Recognition 37(2004) p.1-19.
8. Sabera E., Xu Y., and Tekalp A.M. (2005), Partial shape recognition by sub-matrix matching for partial matching guided image labeling, Pattern Recognition 38: 1560-1573.
9. Kuhl F.P. and Giardina C.R. (1982) Elliptic Fourier features of a closed contour. Comp Graphics Image Processing. 18: 236-258.
10. Grigorescu C. and Petkov N. (2003), Distance Sets for Shape Filters and Shape Recognition, IEEE transactions on image processing, Vol. 12, No. 10, October 2003.
11. Sait S. and Mustafa U. (2006), Geometric Invariant Curve and Surface Normalization, Springer-Verlag Berlin Heidelberg, p. 445-456.
12. Sait S. and Mustafa U. (2005), Affine invariant fitting of algebraic curves using Fourier descriptors, Springer-Verlag London Limited, p. 72-83.
13. Petrakis E., Diplaros A., and Milios E. (2002), Matching and Retrieval of Distorted and Occluded Shapes Using Dynamic Programming, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.24 (11), pp.1501-1516, Nov. 2002.