# Optical Character Recognition for Model-based Object Recognition Applications

Qing Chen and Emil M. Petriu
*University of Ottawa, Ottawa, ON, Canada*
*E-mail: {qingchen, petriu}@site.uottawa.ca*

## Abstract

*This paper discusses the performance of Fourier descriptors and Hu's seven moment invariants for an Optical Character Recognition (OCR) engine developed for 3D model-based object recognition applications.*

## 1. Introduction

One of the most frequent tasks in computer vision and image processing is the recognition of an image or an object in the image. Among these tasks, Optical character recognition (OCR) is a popular research topic. OCR aims at enabling computers to recognize optical symbols without human intervention. This is accomplished by searching a match between the features extracted from the given symbol's image and the library of image models. Ideally, we would like the features to be distinct for different symbol images so that the computer can extract the correct model from the library without confusion. Meanwhile, we also want the features to be robust enough so that they will not be affected by viewing transformations, noises, resolution variations and other factors. Fig. 1 shows the basic process of an OCR system. The applications of OCR have been found in 3D object recognition, automated guided vehicles (AGV), digital libraries, invoice and receipt processing, and recently on personal digital assistants (PDA). OCR includes essential problems of pattern recognition, which are common to other related topics such as image retrieval systems.

Various OCR algorithms have been proposed to achieve better performances. These algorithms include template matching, image signatures, image geometric features, and shape-based image invariants [1]. Among these algorithms, the shape-based image invariants are of particular interests as they have the invariance property which can improve the recognition performance even the images have undergone various image transformations such as translation, scaling and rotation. There are two types of shape-based image invariants: boundary-based and region-based [2]. The boundary-based image invariants focus on the properties contained in the image's contour. The most common boundary-based image invariants include Fourier descriptors and chain codes. The region-based image invariants take the whole image area as the research object. Region-based image invariants include various moment-based invariants such as Hu's seven moment invariants, Zernike moments and complex moments.

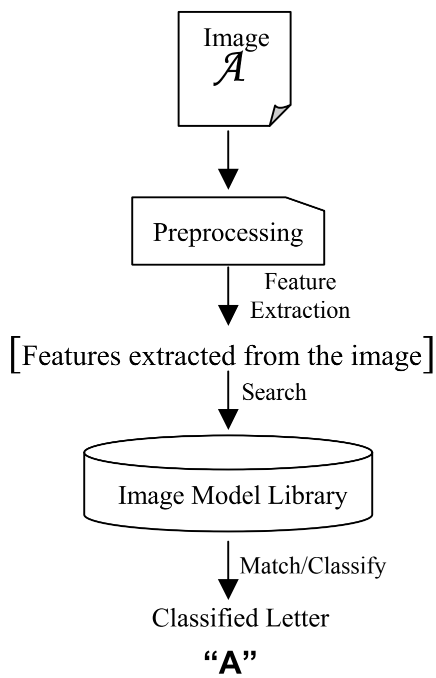


Fig. 1. The basic process of an OCR system.

## 2. OCR in 3D Model-based Object Recognition

The OCR can play a very important role in 3D model-based object recognition systems as Fig. 2 shows. There are two basic operations in 3D model-based object recognition: identification and location [3]. Identification determines the identity of the imaged objects by the comparison of the image data with a database of models. Location determines the location of the object in space. A practical and rapid method for visual recognition of 3D objects is the use of surface encoding with Pseudo Random Binary Array (PRBA) and feature matching with

0-7803-81 08-4/03/$17.00 0 2003 IEEE 

the model database [4]. After the object is visually recognized, its position parameters can be further refined by the supplementary geometric measurements by a pose engine with the data provided by tactile sensors.

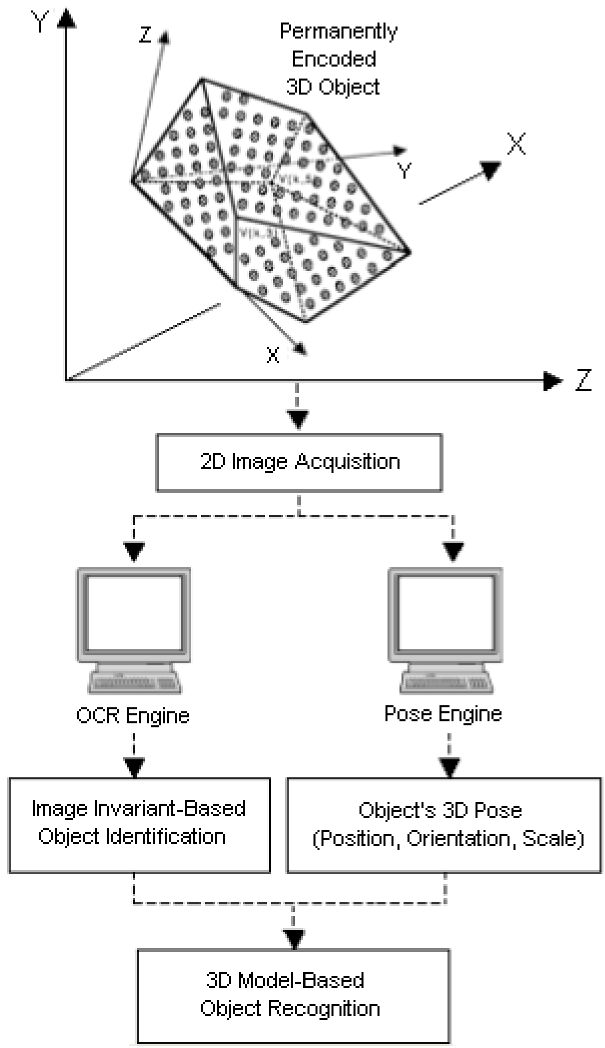


Fig. 2. The OCR engine in a 3D model-based object recognition system.

A pseudo-random binary array {A(i,j) | i= 0,1,2,...,n1-1; j= 0,1,...,n2-1} can be obtained by folding a (2n-1) -term pseudo-random binary sequence {S(p)| p= 0,1,..., 2n-2} generated by a modulo-two n-bit shift register under the following constraints: 2n-1 =2k1.k2-1, n1=2k1-1, n2=(2n-1) /n1 , where *n1* and *n2* must be relatively prime.

According to the PRBA "window property" any nonzero binary pattern seen through a *k1*-by-*k2* window sliding over the array is unique and may fully identify the window's absolute coordinates *(i,j)* within the PRBA. The "pseudo-random/natural" code conversion is implemented as a memory stored table. This encoding has the notable advantage of using only one bit per grid step, independent of the desired grid resolution.

The PRBA code are Braille-like embossed on the object surface. For an efficient pattern recognition, the particular shape of the binary symbols were selected in such a way to meet the following conditions:

- there is enough information at the symbol level to provide an immediate indication of the grid orientation;
- the symbol recognition procedure is invariant to position and orientation changes;
- the symbols have a certain peculiarity so that other objects in the scene will not be mistaken for encoding symbols.

One key problem of symbols recognition is that the appearance of the symbols depends on imaging conditions like viewpoint and perspective angles. If we can find image features that do not change with imaging conditions, the problem would be solved.

The image invariants can be described as functions of some image measurables that do not change under a certain class of transformations [3]. With different values yielded by image invariants for all the shapes of interest, we can use vectors to index a database of models. The OCR engine based on image invariants supports direct, feature-based model indexing, and therefore well-suited to identify the specific subsets of the PRBA codes embossed on the object's surfaces. However, one basic limitation of image invariants is that they are only invariant to a certain class of image transformations. Defining useful invariants for all image transformations is not easy at all. For the purpose of our introductory discussion, we consider only scaling, translation and rotation/orientation of the geometric image transformations.

After the encoded symbols are recognized, it is possible to immediately recover the identity, position and orientation of the imaged object by consulting a database which stores the PRBA mapping relationship.

The paper will investigate the Fourier descriptors and Hu's seven image invariants, which are the most popular boundary-based and region-based image invariants.

## 3. Fourier Descriptors

Fourier descriptors are boundary-based image invariants which were introduced by Zahn and Roskies [5], Persoon and Fu [6], Wallace [7], Arbeter *et al* [8] for description of the shape of a closed, planar figure.

Given a close shape in the 2D Cartesian plane, the boundary *s* can be traced and re-sampled according to, say, the counterclockwise direction, to obtain an uniformly distributed *K* points as shown in Fig. 3. Each point's coordinates can be expressed as $(x_0, y_0)$, $(x_1, y_1)$, $(x_2, y_2)$,…, $(x_{K-1}, x_{K-1})$. These coordinates can be expressed in the form $x(k) = x_k$ and $y(k) = y_k$ . Under this condition, the boundary can be expressed as a sequence of complex numbers [10]:

$$s(k) = x(k) + j\,y(k)\,,\quad k = 0, 1, 2, \ldots, K\text{-}1$$

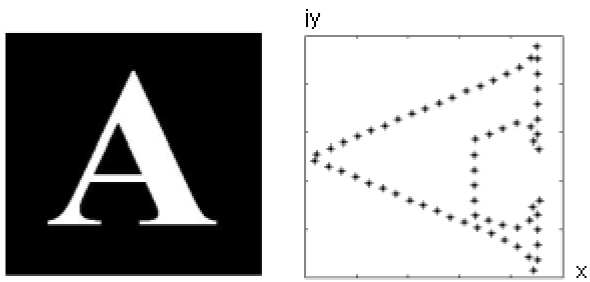


Fig. 3. A re-sampled boundary and its representation as a complex sequence.

That means the *x*-axis is treated as the real axis and the *y*-axis as the imaginary axis of a sequence of complex numbers. The coefficients of Discrete Fourier Transform (DFT) of this complex sequence $z(u)$ is:

$$z(u) = \frac{1}{K}\sum_{k=0}^{K-1} s(k) e^{-j2\pi uk/K}\;,\; u = 0, 1, \ldots, K\text{-}1$$

The complex coefficients $z(u)$ are called the Fourier descriptors of the boundary.

Fourier Descriptors are not directly insensitive to image transformations such as scaling, translation, and rotation, but the changes in these parameters can be related to simple operations on the Fourier descriptors as follows:

Scaling: $z_s(u) = \alpha\, z(u)$

Translation: $z_t(u) = z(u) + \Delta_{xy}\delta(u)$

Rotation: $z_r(u) = z(u)e^{j\theta}$

Starting point: $z_p(u) = z(u)e^{-j2\pi u\, k0\,/\,K}$

The symbol $\Delta_{xy} = \Delta x + j\Delta y$. $\delta(u)$ is an impulse function which only has non-zero values at the origin and zero values everywhere else.

We can see that the magnitudes of $z(u)$ are invariant to rotations because:

$$|z_r(u)| = |z(u)e^{j\theta}| = |z(u)|\cdot|e^{j\theta}| = |z(u)|\cdot 1 = |z(u)|$$

The translation consists of adding a consistent displacement to all sample coordinates of the boundary. The translation has no effect on the descriptors except for $u=0$ because of the impulse function $\delta(u)$. The first component of the Fourier descriptors depends only on the position of the shape. It is not useful in describing the shape and can be discarded.

The invariance to scale changes can be achieved by forming the ratio of two coefficients so that we can get rid of the parameter $\alpha$.

The magnitudes of $z(u)$ are also invariant to the change of starting point because:

$$|z_p(u)| = |z(u)e^{-j2\pi k_0 u}| = |z(u)|\cdot|e^{-j2\pi k_0 u}| = |z(u)|\cdot 1 = |z(u)|$$

By summarizing above analysis, we can use:

$$c(u-2) = \frac{|z(u)|}{|z(1)|}\,,\quad u = 2, 3, \ldots, K\text{-}1$$

to keep the Fourier descriptors invariant to rotation, translation and scaling at the same time.

## 3. Hu's Seven Moment Invariants

Fourier descriptors only explore the contour information; they cannot capture the interior content of the shape. On the other hand, this method cannot deal with disjoint shapes where single closed boundary may not be available; therefore, they have limited applications. For region-based invariants, all of the pixels of the image are taken into account to represent the shape. Because region-based invariants combine information of an entire image region rather than exploiting information just at the boundary points, they can capture more information regarding the image.

Moment-based invariants are the most common region-based image invariants which have been used as pattern features in many applications. Hu first introduced a set of moment-based invariants using nonlinear combinations of regular moments in 1961 [9]. Hu's seven moment invariants have the desirable properties of being invariant under image translation, scaling, and rotation.

Regular moments are defined as:

$$m_{pq} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x^p y^q f(x,y)\,dxdy \quad p,\, q = 0, 1, 2, \ldots$$

where $m_{pq}$ is the (p+q)th order moment of the continuous image function $f(x, y)$.

The central moments of f(x, y) are defined as:

$$\mu_{pq} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} (x-\bar{x})^p (y-\bar{y})^q f(x,y)\,dxdy$$

where $\bar{x} = m_{10} / m_{00}$ and $\bar{y} = m_{01} / m_{00}$, which are the centroid of the image.

For digital images the integrals are replaced by summations and $m_{pq}$ becomes:

$$m_{pq} = \sum_x \sum_y x^p y^q f(x, y)$$

Then the he central moments are changed to:

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q f(x, y)$$

The central moments are computed using the centroid of the image, which is equivalent to the regular moments of an image whose center has been shifted to coincide with its centroid; therefore the central moments are invariant to image translations.

In image transformations, scale changes are caused by:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} S_x & 0 \\ 0 & S_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

To obtain scale invariance, we let $f'(x', y')$ represent the image $f(x, y)$ after scaling the image by $S_x = S_y = \alpha$., so $f'(x', y') = f'(\alpha x, \alpha y) = f(x, y)$, and $x' = \alpha x$, $y = \alpha y$, then we have:

$$m'_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x'^p \, y'^q \, f'(x', y') dx' dy'$$

$$= \alpha^{p+q+2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q f(x, y) dxdy$$

$$= \alpha^{p+q+2} m_{pq}$$

Similarly,

$$\mu'_{pq} = \alpha^{p+q+2} \mu_{pq}, \; \mu'_{00} = \alpha^2 \mu_{00}$$

We can define normalized central moments as:

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\gamma}}, \quad \gamma = (p+q+2)/2, \; p+q = 2, 3, \ldots$$

$\eta_{pq}$ is invariant to changes of scale because:

$$\eta'_{pq} = \frac{\mu'_{pq}}{\mu'^{\gamma}_{00}} = \frac{\alpha^{p+q+2} \mu_{pq}}{\alpha^{2\gamma} \mu_{00}^{\gamma}} = \frac{\mu_{pq}}{\mu_{00}^{\gamma}} = \eta_{pq}$$

Normalized central moments are invariant to image translation and scaling.

Hu's seven moment invariants bsed on normalized central moments are defined as:

$$\phi_1 = \eta_{20} + \eta_{02}$$

$$\phi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2$$

$$\phi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2$$

$$\phi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2$$

$$\phi_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2]$$

$$+ (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03}) \, [3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]$$

$$\phi_6 = (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] +$$

$$4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03})$$

$$\phi_7 = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] +$$

$$(3\eta_{21} - \eta_{30})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]$$

Hu's seven moment invariants are invariant to image transformations including translation, scaling and rotation. As they take every image pixel into account, the computation cost will be much higher than Fourier descriptors.

## 4. The OCR Engine

To investigate the overall performance of Fourier descriptors and Hu's seven moment invariants, an OCR engine is implemented with MATLAB scripts. With this OCR engine, we can collect image features and implement the image recognition. Fig. 4 is a screenshot of the OCR engine.

This OCR engine includes two basic functions: (i) feature extraction, and (ii) image recognition, as illustrated in Fig. 5.

The feature extraction is basically a training process. After a training image is loaded, the user needs to input the capital letter corresponding to the image. By clicking the corresponding "Get" buttons, the Fourier descriptors or Hu's seven moment invariants of this image can be computed and collected. The computed results will be saved to a model library in the format of MATLAB ".mat" file by clicking the "Data" menu and selecting "Save". The data is in the format of feature vectors which can be indexed for each trained shape.

After loading the model library, the user can open an image file and implement the recognition. All of the images are in "JEPG" format in our experiment. By clicking the corresponding recognition button, the OCR engine can compute the features of the current image and compares the result with the feature vectors stored in the model library to make the classification. The comparison is implemented with the Euclidean distance between every feature vector in the model library and the feature vector just computed from the current image.

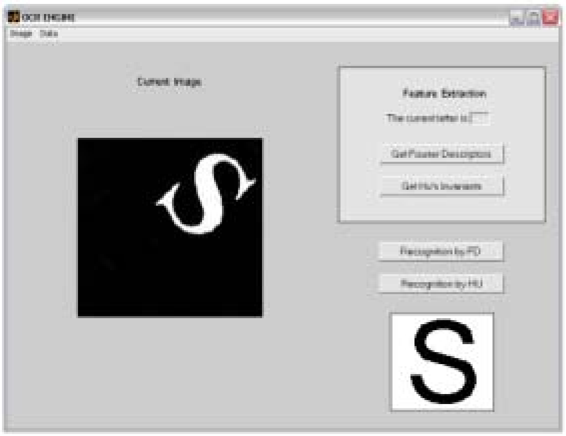

Fig. 4. A screen shot of the OCR engine.

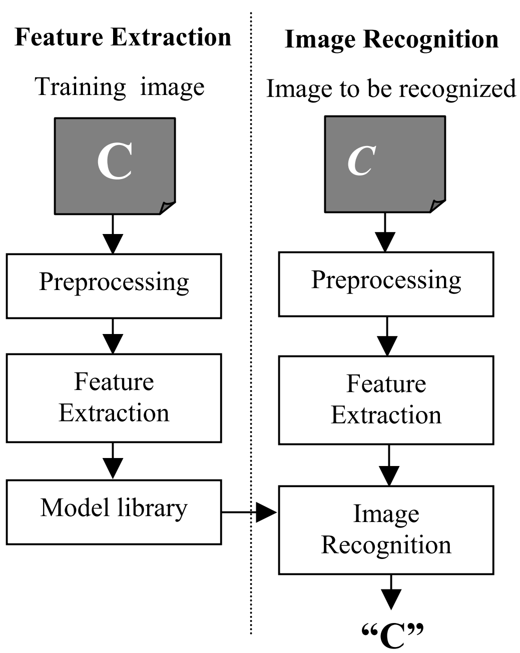


Fig. 5. The working process of the OCR engine.

## 5. Experiment Results

The image samples we used in the experiment are gray level images of 26 capital English letters from "A" to "Z". The image samples spatial resolutions range from 16×16 to 512×512 with the step of the square of 2.

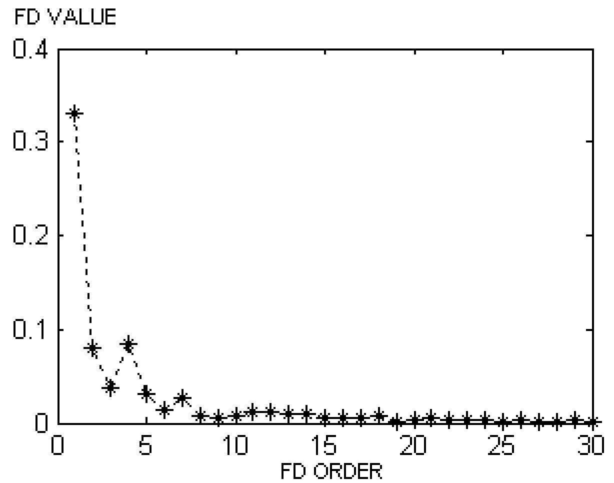


Fig. 6. The Fourier descriptors' distribution of the image "A" with spatial resolution of 512×512.

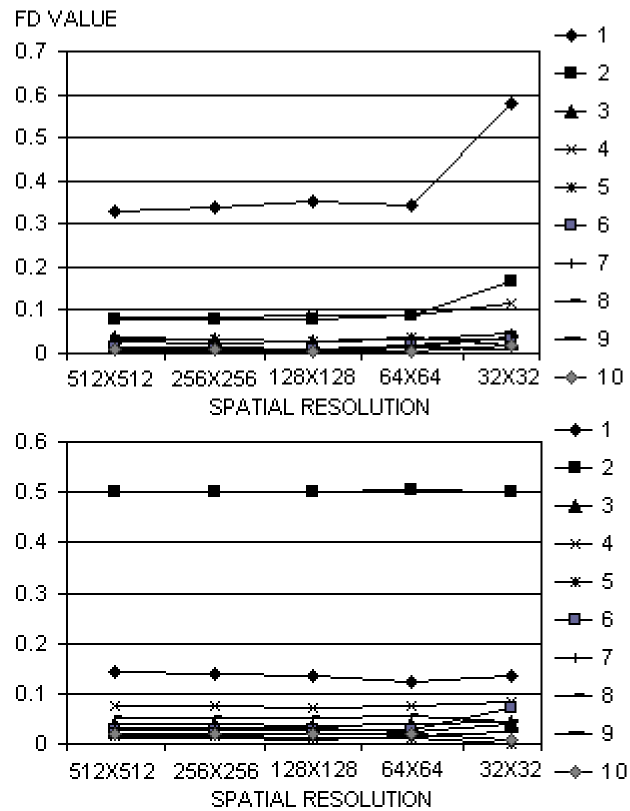


Fig. 7. The Fourier descriptors of image "A" and "Z" with spatial resolutions from 512×512 to 32×32.

In our experiment, we tested all of the image samples from "A" to "Z" with resolutions from 16×16 to 512×512 as well as the rotated images with the resolution of 128×128. In this article, we will only give a randomly selected example to show the result.

Fig. 6 plots the Fourier series of the image "A" with resolution of 512×512. From the figure we see the most significant values of the series are at the first 10 descriptors. The values of the rest descriptors are very close to zero. Based on this, we only collect the first 10 elements of the series to compose the feature vectors. Fig. 7 shows the Fourier series of the image "A" and "Z" against different resolutions. From the diagram we see that the resolution of 64×64 is apparently a threshold for Fourier descriptors. The recognition rate keeps at 100% from 512×512 to 64×64, it reduces to 53.8% at the resolution of 32×32.

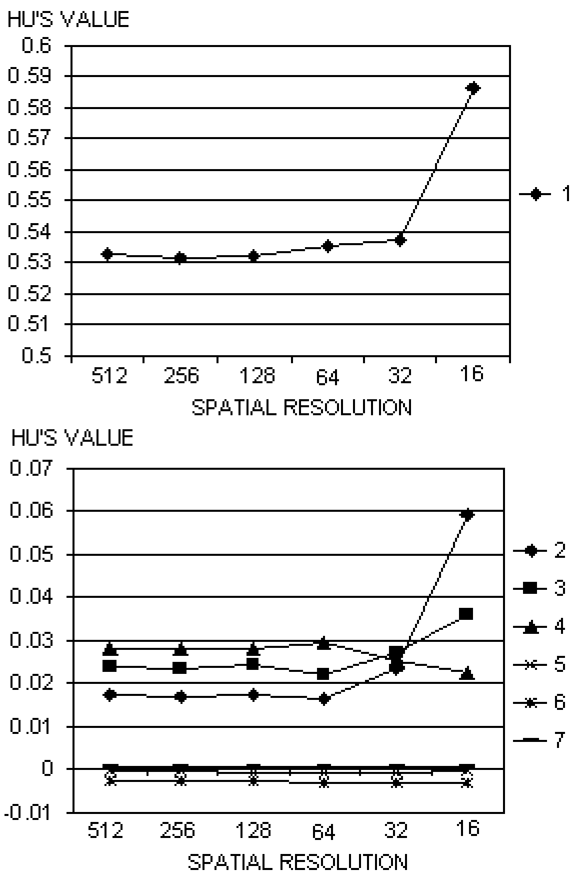


Fig. 8. The Hu's moment invariants of the image "C" with spatial resolutions from 512×512 to 16×16.

Fig.8 plots Hu's seven moment invariants of image "C" against different resolutions from 512×512 to 32×32. We see that the resolution of 128×128 is the threshold for Hu's seven moment invariants. From the resolution of 512×512 to 128×128, the values are kept in a comparatively stable situation and they crossed together at the resolution of 64×64 and the recognition rate also drops to 84.6% from 100%.

## 6. Conclusions

In this paper we studied Fourier descriptors and Hu's seven moment invariants for 3D model-based object recognition applications. Both algorithms have the invariance property against image transformations including scaling, translation and rotation. However, a resolution threshold exists for each method. For Fourier descriptors, with feature vectors composed of the 10 most significant Fourier descriptors, the resolution threshold is 64×64. For Hu's seven moment invariants, the resolution threshold is 128×128. With the same computer configuration, the average computation time of Hu's seven moment invariants is about twice longer than Fourier descriptors. A generic OCR Engine is developed which can extract features from images, build up the image model library, and perform image recognitions.

## References


[1] O. D. Trier, A. K. Jain, and T. Taxt, "Feature extraction methods for character recognition – a survey," *Pattern Recognition*, Vol.29, No.4, pp. 641-662, 1996.

[2] D. Zhang and G. Lu, "A comparative study of three region shape descriptors," *DICTA2002: Digital Image Computing Techniques and Applications*, pp. 21-22, Melbourne Australia, January 2002.

[3] E. Trucco and A. Verri, *Introductory Techniques for 3-D Computer Vision*, Prentice Hall, 1998.

[4] S. K. Yeung, W. S. McMath, E. M. Petriu and N. Trif, "Three dimensional object recognition using integrated robotic vision and tactile sensing," *Proc. IEEE/RSJ International Workshop on Intelligent Robots and Systems IROS'91*, pp.1370-1373, Osaka, Japan, November 3-5, 1991.

[5] C. T. Zahn and R. Z. Roskies, "Fourier descriptors for plane closed curves," *IEEE Trans. Computers*, C-21 (1), pp. 269-281, 1972.

[6] E. Persoon and K. S. Fu, "Shape discrimination using Fourier descriptors," *IEEE Trans. Systems, Man Cybernetics*, Vol.7, No.2, pp.170-179, 1977.

[7] T. P. Wallace and P. A. Wintz, "An efficient three-dimensional aircraft recognition algorithm using normalized Fourier descriptors," *Computer Graphics and Image Processing*, 13, pp. 99-126, 1980.

[8] K. Arbter, W. E. Snyder, H. Burkhardt, and G. Hirzinger, "Application of affine-invariant Fourier descriptors to recognition of 3-D objects," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.12, pp. 640-647, July 1990.

[9] M. K. Hu, "Visual pattern recognition by moment inariants," *IRE Trans. Inf. Theory,* Vol.IT-8, pp.179-187, February 1962.