Handwriting Recognition Using B-Spline Curve

Khoi Nguyen-Tan and Nguyen Nguyen-Hoang

DATIC Lab., DaNang University of Technology
ntkhoi@dut.udn.vn, hoangnguyenbkit@gmail.com

Abstract. This paper aims at presenting novel approach for curve matching and character recognition such as printed writing, handwriting, signatures, etc. based on B-Spline curve. The advantages of the B-Spline that are continuous curve representation and affine invariant, and the robustness. The recognition process is composed of two main steps: sample training and recognition. The computer must be trained with data from bitmap image file. The next step is pre-processing input data from the binary image and finding its skeleton. The reconstruction of a B-Spline curve representing the sample character is applied to find out the control points. Then the sample B-spline curve of each character is stored in a database. For the test character, it has the same process with the sample character. The matching is done by computing the Euclidean distance between the control points of test curve with those of all sample characters to recognize the character. The experimental results show the performance of the proposed algorithm.

Keywords: B-Spline curve, matching curve, handwriting, Optical character, recognition, reconstruction.

1 Introduction

Optical character recognition is a subject that attracts the attention of many researchers but comprehensive solutions for this problem have not been found yet. The key principle of optical character recognition is the description of characters; the result of recognition will be captured by the comparison of these characteristics [1-6]. Speed and accuracy of recognition system depend greatly on the approach, and description of character features. Character recognition is getting more and more useful in daily life for various purposes. During the last few years, researchers have made great efforts on off-line signature recognition. There are many rather effective recognition methods proposed, such as neural network, support vector machine, etc.., and other methods [15, 16]. Character recognition is a form of pattern recognition process. In reality, it is very difficult to achieve 100% accuracy. Many researchers have been done on many types of characters by using different approaches. However, these methods have difficulty recognizing the samples that are larger than the previously studied samples in term of size; the time of training systems are quite large in number; and it is difficult to reconstruct characters.

Major of these researches are about signature verification, however some of them are about character identification. In [1] proposed a wavelet-based offline signature

P.C. Vinh et al. (Eds.): ICCASA 2012, LNICST 109, pp. 335–346, 2013.

verification system that exist within different signatures of the same class and verify whether a signature is a forgery or not. [2] present a signature verification system using Discrete Radon Transform and Dynamic Programming. The author in [3] have used Radon Transform and Hidden Markov Model (HMM) for offline signature verification. Features are extracted by Radon Transform and fed to a HMM classifier.

In [11], the authors compared different statistical methods by using a feature extraction preprocessing, to carry out the recognition of signatures. In [12], the performance of a signature recognition system based on support vector machines (SVM) has been compared with a traditional classification technique, multi-layer perceptrons (MLP). Experimental results show that the performance of SVM is higher than MLP. The common way used for character recognition would be the use of artificial neural networks and feature extraction methods [5]. In [9,10] has proposed an algorithm is based on constructing and comparing B-spline curves of object boundaries.

In this paper, we present a method by using B-splines representation for optical character recognition. The main idea is each character presenting as a curve and its characteristics are described by the control points of B-Spline curve. We combines the advantages of B-spline that are continuous curve representation [7,8] and the robustness of dissimilar matching with respect to noise and affine transformation. It avoids the need for other matching algorithms that have to use the resampled points on the curve. The proposed algorithm has been tested by matching similar shapes from a prototype database. Our experimental results demonstrate that we can achieve good recognizing results.

The rest of this paper is organized as follows: In section 2, overview related works. Section 3 addresses the main steps of our algorithm. In section 4, reconstruction of B-spline curves based on inverse subdivision method are described. Finally, section 5 proposed method for dissimilarity calculations between curves method are described and experimental results are given to demonstrate the usefulness and quality of the approach.

2 Character Recognition Using B-Spline Curve

This section discusses the mathematical formulation of the problem of aligning two curves. We first consider the case of aligning two curve segments and then use this to align two closed curves.

During the process of writing or signing, all characters are formed by certain curves; each word or letter can be considered as a curve. In geometric modeling, B-spline stands as one of the most efficient curve representation. With the property of easy local controlling being invariant under affine transformations such as translation allowed, scales, rotations, B-Spline curve is suitable to simulate handwriting, printed texts or signatures of people with different spellings. The B-spline curve is not uniquely described by a set of control points. The construction of B-Spline curve with each character results in the problem of recognizing the matching set of control points of them.

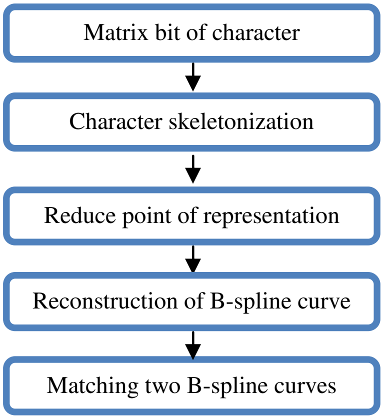


Fig. 1. Steps of handwriting recognition

The main steps of the algorithm are as follows:

1) Scanning and transformation of the document: A printed document is scanned by scanner and can be saved in a bitmap file format. The next, this bitmap file is converted it to monochrome binary image. Now the image is ready to be processed.
2) Segmentation into lines and segments lines into characters: the partitioning of lines from the total document based on thresholding histogram values gathered horizontally. Then characters are segmented from each line by using values of vertical histogram. The next step will process these individual character segmented.
3) Binary the character bitmaps, or transform the image of characters into a bit matrix (convention that black pixel is *0* and white pixels is *1*).
4) Finding the skeleton of the characters segmented.
5) Transformation of the bit matrix into the list of co-ordinations
6) Reduction of point number that represents the character skeletonization, retains only main medial axis of a character.
7) Reconstructing B-spline curve from pixel points of the character skeletonization. This process to find the set of control points of B-spline curve.
8) Matching the control points of test B-spline curve and sample B-spline curve to recognize the character corresponding.

These steps are explicitly presented in Figure 1.

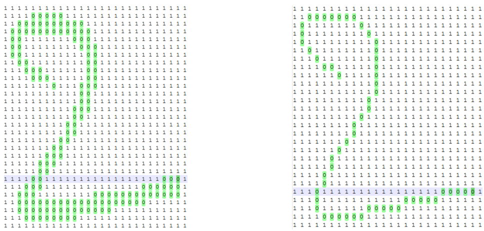
2.1 Skeletonization

The objective of skeletonization is to find the medial axis of a character. Skeletonization is the process of thinning of a pattern as many pixels as possible without affecting the general shape of the pattern. After pixels have been thinned, the pattern should still be recognized.

The process of character segmentation is made thin to a unit pixel thickness. There are two main methods of skeletonization:

1) Skeletonization based on smoothing: Considering all the pixels of the object, if it satisfies certain delete conditions, it will be deleted. The skeletonization process is repeated until there is no more pixels which can be deleted.
2) Skeletonization not based on smoothing: Considering at any point of the object, if there are many points of edge which have the same shortest distance to that point; it lies on the median axis. The set of points lying on the median axis of the object forms the skeleton of the character.

We chose the skeletonization based on smoothing algorithm and Hilditch's algorithms [17] for these methods are robust and easy to install. The Figure 2 illustrates a result of skeletonization algorithm for a bitmap character. This algorithm deletes the extra pixels which do not belong to the backbone of the character.



a) Before skeletonization *b) After skeletonization*

Fig. 2. Result of skeletonization

2.2 Reducing the Number of Data Points

After the skeletonization of character segmented, we obtain a series of consecutive points. The purpose of this step is to minimize the number of point performing the curve and reduce storage space and it is more convenient to find the controlling vector of the characters and to match later.

The curve C(u) consists of n points in the plane $(x_1, y_1),(x_2, y_2)\ldots(x_n, y_n)$. The problem is to remove a number of points on the curve so that a new curve $(x_{i1}, y_{i1}),(x_{i2}, y_{i2})\ldots(x_{im}, y_{im})$ are "nearly identical" to the original curve. Algorithms Bandwidth [17] is used as follow:

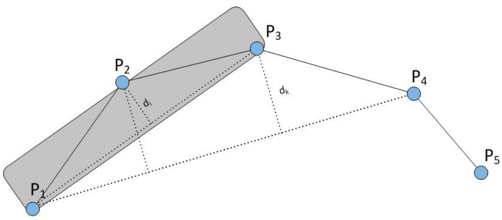


Fig. 3. Result of bandwidth algorithm

1) Identify the first point on the curve and see it as a key point (P_1). The point (P_3) is considered as the variable point. The mid-point (P_2) of the key point and the variable point is intermediate point. The distance between these points is optional.
2) Calculate the distance from the intermediate point to the line connecting the key point and the variable point.
3) Check the distance found. If that distance is smaller than a given threshold θ, the intermediate points can be removed. Otherwise, the key point moves to the intermediate point. In this paper, we chose the value of threshold is *0.4*.
4) The process is repeated, the intermediate point is transferred to the variable point and the next point of the variable one is appointed the new variable point.

The example of this process result is shown in the following Figure 2.

The skeleton hence obtained must have the following properties: as thin as possible, connected, centered. Therefore the result of skeletonization character will be taken as input to the B-spline curve approximation step.

The B-spline curve is considered as an appropriate model for character recognition. In the next section, we present the B-spline mathematical formula and the B-spline curve reconstruction algorithm.

3 B-Spline Curve Approximation

In the following we represent the mathematical formulation of the reconstructing B-spline curve. We shall assume a B-Spline curve *C(u)* of degree *p* which is defined by *n* control points $P = \left\{ P_0, P_1, \ldots, P_{n-1} \mid P_i \in \Re^k \right\}$. Let a vector known as the knot vector be defined by $\mathbf{U} = \left\{ u_0, u_1, \ldots, u_{n+p} \right\}$, where ***U*** is a nondecreasing sequence that satisfies $u_i \in [0,1]$ and $u_i \leq u_{i+1}$. Then, the B-Spline curve is defined as [7,8]:

$$C(u) = \sum_{i=0}^{n} P_i . N_{i,p}(u)$$

The B-Spline basis functions $N_{i,k}$ are defined as [5]:

$$N_{i,0}(t) = \begin{cases} 1 & u_i \leq t \leq u_{i-1} \\ 0 & otherwise \end{cases}$$

$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u)$$

The shapes of the $N_{i,k}$ basis functions are determined entirely by the relative spacing between the knot vector U. B-Splines consist of sections of polynomial curves connected at points called knots. The knot vector defines how the polynomial pieces are blended together with the proper smoothness and determines where and how the control points affect the B-spline curve. If duplication happens at the other knots, we can generate a curve with sharp turns or even discontinuities.

The problem is reconstructing smooth curve from an discrete point data set.

We present in this section a new B-Spline curve reconstruction method based on the non-uniform inverse subdivision scheme (NUISS) [13,14]. As the subdivision scheme is invertible, one can restore recursively all the previous coarser polygon by using the inverse subdivision scheme. Consequently, it also permits to reconstruct the control polygon of a limit parametric curve issued from a nonuniform subdivision curve. After each step of the inverse subdivision, we retrieve curve elements of the previous subdivision curve. The knot intervals are already determined by retaking the information of input polygonal curve.

We propose the formulas to compute the inverse subdivision points based on the formulas of non-uniform cubic B-Spline curve subdivision in [13, 14]. The inverse vertex point P_i^k of polygonal curve P^k (Figure 4) is computed based on the points of polygonal curve P^{k+1} by the following formulas [13,14]:

$$P_i^k = 2P_{2i-1}^{k+1} - \frac{d_i P_{2i-2}^{k+1} + d_{i-1} P_{2i}^{k+1}}{d_{i-1} + d_i} \qquad \textit{for } i = 1 \ldots n-1$$

$$P_i^k = \frac{2(d_{i-1} + d_i + d_{i+1}) P_{2i}^{k+1} - (d_i + 2d_{i-1}) P_{i+1}^k}{d_i + 2d_{i+1}} \quad \textit{for } i = 1 \ldots n-2$$

$$P_{i+1}^k = \frac{2(d_{i-1} + d_i + d_{i+1}) P_{2i}^{k+1} - (d_i + 2d_{i+1}) P_i^k}{d_i + 2d_{i-1}} \quad \textit{for } i = n-2$$

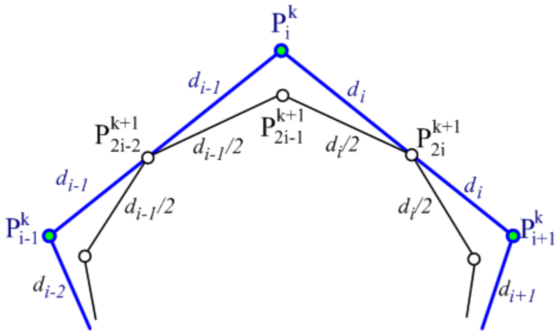


Fig. 4. The relation of inverse points P^k with the points P^{k+1}

For our reconstruction NUISS method, we must determine the knot interval vectors $U = \{u_0, \ldots, u_{n+p}\}$. As these vectors are computed from knot vectors, we propose to define the knot vector from the input initial polygonal curve. The knot vector greatly affects the shape and parameterization of a B-spline curve. Fundamentally three types of knot vector are used: uniform, open uniform and non-uniform. Different parametric methods have been proposed. We only consider the most widely-used parametric chord length method [5].

Suppose that we are given a set of points {P_i} *(i = 0,..., n-1)*, we will find the parametric values $\{u_i \mid 0 \le u_i \le 1, i = 0,\ldots,n-1\}$ associated with these points {P_i}. Therefore, we use open uniform knot vector and it is defined by the following formula [7,8]:

$$\bar{u}_0 = 0, \qquad \bar{u}_{n-1} = 1$$
$$\bar{u}_i = \bar{u}_{i-1} + \frac{|P_i - P_{i-1}|}{\sum_{j=1}^{n-1} |P_i - P_{i-1}|} \qquad \forall i \in [1,\ldots,n-2]$$

with $|P_i - P_{i-1}|$ is the length between two consecutive points.

Our goal is to create knot interval vectors for an initial polygonal curve to implement the NUISS scheme, and also to reconstruct the non-uniform B-Spline curve. The knot interval vectors for the initial polygon curve *P* can be created by using the parametric chord length method. From the initial polygonal curve *P* with initial knot interval vectors, we can recreate the coarse polygonal curve *Q* by our NUISS scheme. This coarse curve Q using as a control polygon of the non-uniform B-Spline curve C(u), the knot vectors of which are created from the corresponding knot interval vectors of the polygonal curve. It creates an approximate B-Spline curve reconstruction. The main advantage of our algorithm NUISS is that it can also give us good results for verification purposes and this approach completely avoids the parameterization problem. As the inverse subdivision can be stopped after each step, different approximation curve can be obtained. An interpolation curve is obtained by locally interpolation fitting the given initial data points.

3.1 Matching Using B-Spline Curve

In this step each test character will be recognized and will be identified to a predefined test character. For our algorithm, we made the test character and sample character have the same size. Therefore, it is easily to reconstruct the B-spline curves representing of these characters. All reconstructed B-spline curves have the same degree and number of control points for the matching.

We use the Euclidean distance to evaluate the dissimilarity between two control points set In mathematics. The Euclidean distance or Euclidean metric is the "ordinary" distance between two control points corresponding of two B-spline curves.

With two set of control points $P = (P_1, P_2, \ldots, P_n)$ and $H = (H_1, H_2, \ldots, H_n)$, in N-dimensional Euclidean space, the distance between points P_i and H_i is the length of the line segment connecting them $\overline{P_i H_i}$. Therefore, the Euclidean distance is given by:

$$E(p,q) = \sqrt{\sum_{k=1}^{n} (p_i - q_i)^2}$$

In the exception case, if the length of two control-point vectors are different, suppose that P has the length n, Q has the length m, with $m>n$, then we need to add $k=m-n$ control points to reconstruct the new B-spline curve for two vectors have the same size.

4 Experimental Results

The test character will be matched with all sample characters. The result of recognition is the studied sample which has the minimal error E with the tested sample.

4.1 Printed Text Recognition

In this experimental, we present an example of recognition a printed text document. Figure 5 shows the training of a test character of number eight in one ten number characters. Curve matching is achieved by computing the Euclidean distances. Also we have tested on different sizes of characters. With the test number character is presented in 14 point Arial font, the accuracy of the recognition process reach 100%. With the 19 point Arial font of size 18, the recognition accuracy is 98%, and with the same Arial font but size 10, the recognition accuracy is 90%.

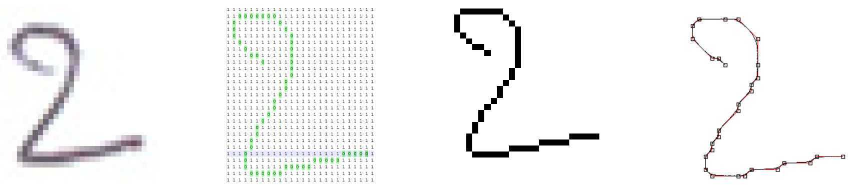
The recognition result of the large test character is better than that of the small test character. The reason is when using the small size; the character skeletons *should also have a corresponding small size*. Then, the *reconstruction of B-spline curve* from a small point is not well approximated with the initial data points.

a. Input image *b. Matrix of bits* *c. Skeleton image* *d. B-spline curve*

Fig. 5. The process of recognition of eight font Arial

4.2 Handwriting Recognition

This type of recognition is a function that allows to write onscreen in a small panel, and to recognize characters and other symbols written by hand in natural handwriting. Figures 6(a-d) illustrate our surface reconstruction method applied to the skidoo model. We also present other examples in Figure 3. In order to compare the fitting error across different models, we uniformly scale the data points P to fit within a unit cube.



a. Input image *Matrix of bits* *c. Skeleton image* *d. B-Spline curve of character*

Fig. 6. Process of recognition of handwritten

Let the machine learn handwritten digit that has the image size is *40×40px*. The result is as follow (Figure 6).

By computing the Euclidean distances between the sample curves from database and the test B-spline curve reconstructed from the character image, we are able to recognize the character by locating the minimum distance in the table of distances (Table 1).

Table 1. Matching result between sample tests and trained one (handwritten) with the Euclid measure

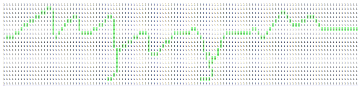
Sample trained	1 2 3 4 5 6 7 8 9 0									
Sample test	1 2 3 4 5 6 7 8 9 0									
Train / Recog.	0	1	2	3	4	5	6	7	8	9
0	**63.73**	114.98	123.23	97.78	122.36	130.28	172.9	161.54	132.29	84.34
1	82.81	**51.75**	81.5	61.23	62.31	71.29	55.54	68.18	107.71	84.83
2	96.04	88.82	**43.69**	129.41	107.33	98.79	155.68	134.38	171.39	178.17
3	81.75	111.58	88.86	**72.91**	110.92	138.25	132.78	87.63	147.84	108.88
4	119.96	76.96	173.73	158.01	**55.56**	116.11	167.26	94.88	185.07	202.79
5	144.68	199.79	139.73	124.12	179.73	**66.34**	145.69	169.94	226.97	191.51
6	103.31	173.64	106.95	80.67	150.66	120.46	**65.29**	146.15	122.56	86.24
7	156.03	206.05	166.55	103.29	136.72	157.35	172.52	**95.33**	189.87	152.28
8	80.09	136.72	89.5	75.34	120.41	119.12	104.45	117.33	**40.84**	113.59
9	130.19	157.74	208.57	104.95	195.96	202.68	199.57	150.13	179.47	**91.11**

Table 1 shows the experimental results where after eight tests, average recognition accuracy of 95 % is obtained. The blue cell illustrated the minimum Euclidean distance in each row of the table. We have set the reference set font as Times New Roman type with size 20.

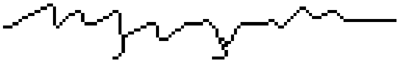
4.3 Signature Recognition

The signature identification and verification is very important in security and resource access control. Signature recognition examines a person when he signs her name. Many documents such as forms and bank checks necessitate the signing of a signature.

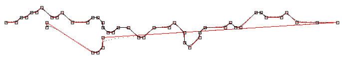
a. Input image



b. Matrix of skeleton



c. Image of skeleton



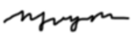
d. Sign approximation with a B-Spline curve

Fig. 7. Process of recognition of sign

We have to fit a B-Spline curve with sample from a signature image of a person. The sample B-spline curve stored in a database. By computing the differences between the test B-spline curves and all sample curves from database using Euclidean distance, we are able to identify the person by locating the minimum value in all distances.

Let the program learns three sample signatures of three different people (named Long, Nguyen, Yen). The steps of recognition process are illustrated in the Figure 8.

Table 2. Result of matching between sample tests and trained one (signature) with the measure Euclid

Train / Recognize			
	**1494.12**	2243.58	9708.1
	3153.64	**1236.71**	4333.74
	11365.59	10132.86	**7066.92**
	Long	*Nguyen*	*Yen*

In the Table 2, the blue cell is the minimum value in each row. The results in Table 1 and Table 2 show that the method of using B-Spline curves to recognize the same sized characters as well as the signature have high percentage of accuracy. Especially, the larger the tested samples are, the more accurate of the recognition is, because the specific deviation and the size are directly proportional. However, when the recognition system recognizes small samples, a specific deviation decreases, it will the resulting in the confusion of the recognition among samples.

5 Conclusions and Further Work

In this paper we deal with the problem of matching and recognizing the optical characters including printed text, handwriting and signature which are modeled as B-splines. The essence of this recognition is comparing the test characters with the sample trained ones. First, the computer must be trained with sample data from bitmap image file. The next step is pre-processing input data including creating the binary image and finding its skeleton. Then the reconstruction of B-Spline curve from skeleton is applied to find out the control points of the character and save them into a sample. The recognition process has the same process. The last step is computation the Euclidean distance between the control points of the test character and the trained samples. The return result is the trained samples that have the smallest Euclidean distance with the control points of the sample character.

The most important part of this recognition method is all the reconstructed B-spline curves from characters have the same degree and number of control points. It makes the identification process easier and less time taken one. Thus, the curve matching error is reduced. The organization of sample data storage is also very important impact on the rate and precision of recognition process. We have run the algorithm to many styles of characters and get high accuracy on those styles. The experimental results showed the robustness and accuracy of the proposed method in B-spline curve matching. The results of recognition characters can be applied in many areas such as text translator, intelligent word recognition, internet search engines…

Further work is improving methods and deploying applications for text recognition. Also parallel approaches may be explored to get the performance of text document recognition process.

References

1. Deng, P.S., Liao, H.–Y.M., Ho, C.W., Tyan, H.–R.: Wavelet–based Off–line Signature Verification. Computer Vision and Image Understanding 76(3), 173–190 (1997)
2. Herbst, B., Coetzer, H.: On An Offline Signature Verification System. In: 9th Annual South African Workshop on Pattern Recognition, pp. 39–43 (1998)
3. Coetzer, J., Herbst, B.M., Du Preez, J.A.: Offline Signature Verification Using the Discrete Radon Transform and a Hidden Markov Model. Eurasip Journal on Applied Signal Processing 4, 559–571 (2004)
4. Basu, J.K., Bhattacharyya, D., Kim, T.-H.: Use of Artificial Neural Network in Pattern Recognition. International Journal of Software Engineering and Its Applications 4(2) (2010)
5. Izakian, H., et al.: Multi-Font Farsi/Arabic Isolated Character Recognition Using Chain Codes. World Academy of Science, Engineering and Technology 43, 67–70 (2008)
6. Pal, U., Dutta, S.: Segmentation of Bangla Unconstrained Handwritten Text. In: ICDAR, 7th, pp. 1128–1132 (2003)
7. Piegl, L., Tiller, W.: The NURBS Book, 2nd edn. Springer (1997)
8. Rogers, D.F.: An Introduction to NURBS With Historical Perspective (2001)
9. Tirandaz, H., Nasrabadi, A., Haddadnia, J.: Curve Matching and Character Recognition by Using B-Spline Curves. IACSIT International Journal of Engineering and Technology 3(2) (2011)
10. Tirandaz, H., Nasrabadi, A., Haddadnia, J.: Curve Matching and Character Recognition by Using B-Spline Curves. IACSIT International Journal of Engineering and Technology 3(2) (2011)

11. Riba, J.R., et al.: Method for invariant signature classification. In: Proceedings of 15th International Conference on Pattern Recognition, vol. 2, pp. 953–956 (2000)
12. Frias-Martinez, E., Sanchez, A., Velez, J.: Support vector machines versus multi-layer perceptrons for efficient off-line signature. Engineering Applications of Artificial Intelligence 19, 693–704 (2006)
13. Nguyen-Tan, K., Le, C.: A method for describing planar cam profile function from a set of digital points. In: Proceedings of the 10th Global Congress on Manufacturing and Management, Thailand, pp. 145–150 (2010)
14. Nguyen-Tan, K., Raffin, R., Daniel, M., Le, C.: B-spline surface reconstruction by inverse subdivisions. In: Proceedings of the 7th International Conference on Computing and Communication Technologies, pp. 336–339 (2009)
15. Mokhtarian, F., Mackworth, A.K.: Scale-Based Description and Recognition of Planar Curves and Two dimensional Shapes. IEEE Trans. PAMI. PAMI-8, 34–43 (1986)
16. Ozdil, M.A., Vural, F.T.Y.: Optical character recognition without segmentation. In: Proceedings of the Fourth International Conference on Document Analysis and Recognition, vol. 2, pp. 483–486. IEEE Computer Society (1997)
17. Cheriet, M., Kharma, N., Liu, C.-L., Suen, C.Y.: Character recognition systems A guide for Students and Practioners. Wiley (2007)