Seeking Multi-thresholds directly from Support Vectors for Image Segmentation

Songcan Chen* Min Wang

Dept. of Computer Science & Engineering, Nanjing University of Aeronautics & Astronautics,

Nanjing 210016, China.

Abstract: Threshold selection is an important topic and also a critical preprocessing step for image analysis, pattern recognition and computer vision. In this letter, a novel automatic image thresholding approach only from the support vectors is proposed. It first fits the 1D histogram of a given image by support vector regression (SVR) to obtain all boundary support vectors and then sifts automatically so-needed (multi-)threshold values directly from the support vectors rather than the optimized extrema of the fitted histogram in which finding the extrema is in general difficult. The proposed approach is not only computationally efficient but also does not require a *priori* assumptions whatsoever are made about the image (type, features, contents, stochastic model, etc.). Such an algorithm is most useful for applications that are supposed to work with different (and possibly initially unknown) types of images. The experimental results demonstrate that the proposed approach can select the thresholds automatically and effectively, and the resulting images can preserve the main features of the components of the original images very well.

Index Terms: image segmentation, support vector regression, automatic thresholding, histograms, image processing.

1 Introduction

In many applications of image processing, abstractions of objects or features, which are used in high level tasks, are derived from images. For the purpose of abstraction, the pixels in an image have to be grouped into meaningful regions by a process called image segmentation.

In many cases, the gray levels of pixels belonging to the object are substantially different from the gray levels of the pixels belonging to the background. Thresholding then becomes a simple but effective tool to separate objects from the background. Its applications include document image analysis, where the goal is to extract printed characters [1, 2], logos, graphical content, or musical scores; map processing, where lines, legends, and characters are to be found [3]; scene processing, where a target is to be detected [4]; and quality inspection of materials [5, 6], where defective parts must be delineated.

---

* Corresponding author: Tel: +86-25-84892805; Fax: +86-25-84498069. E-mail: s.chen@nuaa.edu.cn (S.C. Chen); wm_wangmin@yahoo.com.cn (Min Wang)

Thresholding in its simplest form involves mapping all pixels above a threshold value to one gray value, say white, and the rest to another, say black. Since the result is an image with two gray vales, the process is called bilevel segmentation. When multiple threshold values are used, the result is a multilevel image, and the process is called multilevel segmentation. Bilevel segmentation is appropriate for some of the "classical" image processing applications such as the automatic image analysis of documents or industrial parts. But for the applications dealing with more complex scenes, automatic multilevel image segmentation methods have to be adopted.

Many automatic thresholding techniques use the histogram of a given image to select a good threshold. An image histogram is a frequency distribution of its gray levels. If, in an image, the objects have distinctly different gray values from the background, the histogram will exhibit two different peaks with a valley between them. The determination of a suitable threshold value, usually selected at the bottom of the valley between these two peaks, is a relatively simple. However, in many real-world images, this assumption is unrealistic. There have been a number of methods for threshold selection discussed in the literature, including those based on entropy [7-9], moment preservation [10], error minimization [11] and maximum likelihood [12]. One common characteristic of these existing methods is that the histogram is viewed as a mixture density function, and usually the problem of threshold determination is treated as a case of classification. In this paper, we propose a new threshold selection technique different from the existing approaches mentioned above. It first fits the 1D histogram of a given image by support vector regression (SVR) to obtain all boundary support vectors (BSV) and then sifts or selects automatically so-needed (multi-) threshold values directly from the BSVs rather than the extrema of the fitted histogram. Since there are more than one support vectors, this approach can consequently lead to multilevel thresholding.

This paper organizes as follows. In Section 2, the automatic threshold determination approach based on the functional regression of the histogram is described. Experimental results for the proposed approach and comparison with two other approaches are presented in Section 3. We compare the performance of our approach with two other approaches from the literature, namely Belkasim *et al*'s phase-based optimal image thresholding [13] and Chung *et al*'s fast adaptive PNN-based thresholding algorithm 1 [14]. Finally, Section 4 ends the paper with some future work.

2 Seeking multi-thresholds from support vectors

Before the detailed description of our approach, it is necessary to give a brief review of ε −SVR first.

2.1 SVR

Let the training set D be $\{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$, with input $\mathbf{x}_i \in \Re^n$ and output $y_i \in \Re$. In the ε −SVR, $\mathbf{x}$ is first mapped to $\mathbf{z} = \psi(\mathbf{x})$ in a Hilbert space $\mathcal{F}$ (with inner product $\langle \cdot, \cdot \rangle$) via a nonlinear map $\psi : \Re^n \to \mathcal{F}$. This space $\mathcal{F}$ is often called the *feature space* and its dimensionality is usually very high (sometimes infinite). Then, a linear function $f(\mathbf{x})=\langle\mathbf{w},\psi(\mathbf{x})\rangle+b$ is constructed in $\mathcal{F}$ such that it deviates least from the training data according to Vapnik's ε −insensitive loss function

$$|y-f(\mathbf{x})|_{\varepsilon}=\begin{cases}0, & \text{if } |y-f(\mathbf{x})|\le\varepsilon\\ |y-f(\mathbf{x})|-\varepsilon, & \text{otherwise}\end{cases}$$

while at the same time keeps as "flat" as possible (i.e., ||**w**|| is as small as possible). Mathematically, this means

$$\text{minimize}\ \frac{1}{2}\|\mathbf{w}\|^2+C\sum_{i=1}^{N}(\xi_i+\hat{\xi}_i)$$

$$\text{subject to}\begin{cases}y_i-f(\mathbf{x}_i)\le\varepsilon+\xi_i\\ f(\mathbf{x}_i)-y_i\le\varepsilon+\hat{\xi}_i, i=1,\ldots,N\\ \xi_i,\hat{\xi}_i\ge 0\end{cases}$$

where C is a user-defined constant. It is well known that the above problem can be transformed to the following quadratic programming (QP) problem:

$$\max\text{imize}\quad \sum_{i=1}^{N}y_i(\alpha_i-\hat{\alpha}_i)-\varepsilon\sum_{i=1}^{N}(\alpha_i+\hat{\alpha}_i)-\frac{1}{2}\sum_{i,j=1}^{N}(\alpha_i-\hat{\alpha}_i)(\alpha_j-\hat{\alpha}_j)K(\mathbf{x}_i,\mathbf{x}_j)$$

$$\text{subject to}\quad \sum_{i=1}^{N}(\alpha_i-\hat{\alpha}_i)=0 \quad\text{and}\quad 0\le\alpha_i,\hat{\alpha}_i\le C, i=1\cdots l$$

where $K(\cdot,\cdot)$ is a kernel function.

All $\mathbf{x}_i$'s corresponding to nonzero $(\alpha_i-\hat{\alpha}_i)$'s constitute the support vector set (SV set for short). And all $\mathbf{x}_i$'s on bound, i.e., satisfying α_i or $\hat{\alpha}_i$ $=C$, are members of the boundary support vector set (BSV set for short). Thus, the regressed function can be written as

$$y_i^{'}=\sum_{j=1}^{N}(\alpha_i-\hat{\alpha}_i)K(\mathbf{x}_i,\mathbf{x}_j)+b$$

2.2 Seeking multi-thresholds directly from support vectors --- A case analysis and algorithm

Next, we explain our approach by taking the 'bacteria' image (see Fig. 3(a)) as example.

Let each pixel of the image have gray level in [0, 1, 2, …, L-1] , and commonly L = 256. The number of pixels with gray level i is denoted by n_i, i=0, 1, 2, …, L-1, and the total number of pixels is denoted $N=n_0+n_1+\cdots+n_{L-1}$. Thus, the gray level histogram is defined as a probability distribution: $p(i) = n_i/N$, $p(i)\geqslant 0$, and $\sum_{i=0}^{L-1}p(i)=1$. Forming a histogram $P(x)$ of the image results in an ordered set of discrete values, $p(0)$, $p(1)$, …, $p(L$-1). Suppose those zero $p(j)$s' are deleted and we pack the remaining nonzero $p(k)$'s, say m nonzero $p(k)$'s, into an array with size m. We thus have a compact image histogram with the probability distribution <$p(i_0)$, $p(i_1)$, …, $p(i_{m-1})$> for $0\leqslant i_j\leqslant L-1$, j=0, 1, 2, …, m-1, and $p(i_j)\neq 0$. The total number of pixels is denoted by $N = n_{i_0} + n_{i_1} + \cdots + n_{i_{m-1}} = n_0^{'} + n_1^{'} + \cdots + n_{m-1}^{'}$, where $n_{i_j} = n_j^{'} = N \times p(i_j)$. According to the above definition, the compact image histograms for the 'bacteria' and 'Lena' image are shown in Fig. 1-2(a), respectively.

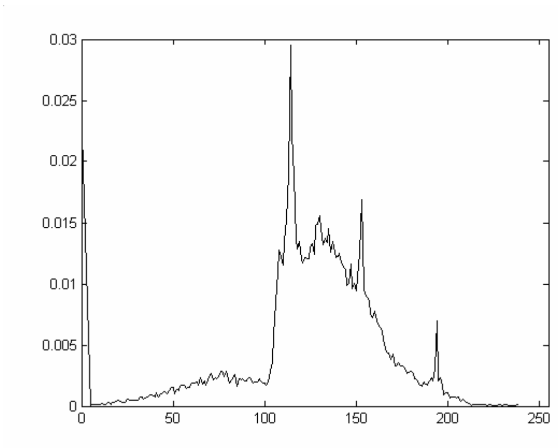
Our aim is to approximate the set of discrete values by some fitting method such that the p_j, j=0, 1, 2, …, m-1 can be replaced by q_j, j=0, 1, 2, …, m-1. In this paper, we choose the SVR as the fitting tool rather than the traditional histogram smooth technique. An advantage of doing so is that the generated support vectors on boundary can provide us good candidate thresholds, consequently, it is not necessary for us to employ some complicated optimization technique to find so-needed thresholds from the extrema of the fitted or regression histogram. Besides, the SVR has much better noise-tolerance than the traditional smooth one. For the 'bacteria' and 'Lena' image, the regression histograms are shown in Fig. 1-2(b) respectively, and the BSVs are labeled with '+'.

Next we will seek out optimum threshold values directly from the BSV set. We calculate first order derivative of the regression histogram with respect to each support vector (SV) in the BSV. From this derivative we can obtain threshold values by identifying the gray levels or SVs within the areas where negative to positive transition is occurring.

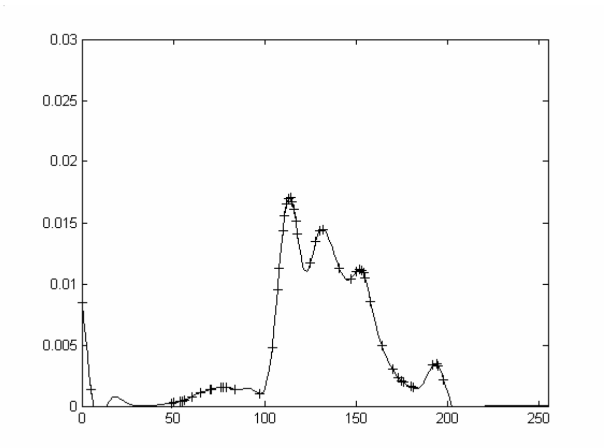
According to the above description, the proposed algorithm is summarized below:

---

1. Form a compact histogram form a given input image;
2. Use the ε −SVR to approximate the histogram and let BSV be the set consisted of all support vectors on boundary;
3. Seek thresholds from the BSV set to ensure them to lie in the areas where negative to positive transition of first order derivative is occurring;
4. Use the so-obtained thresholds to segment the input image.

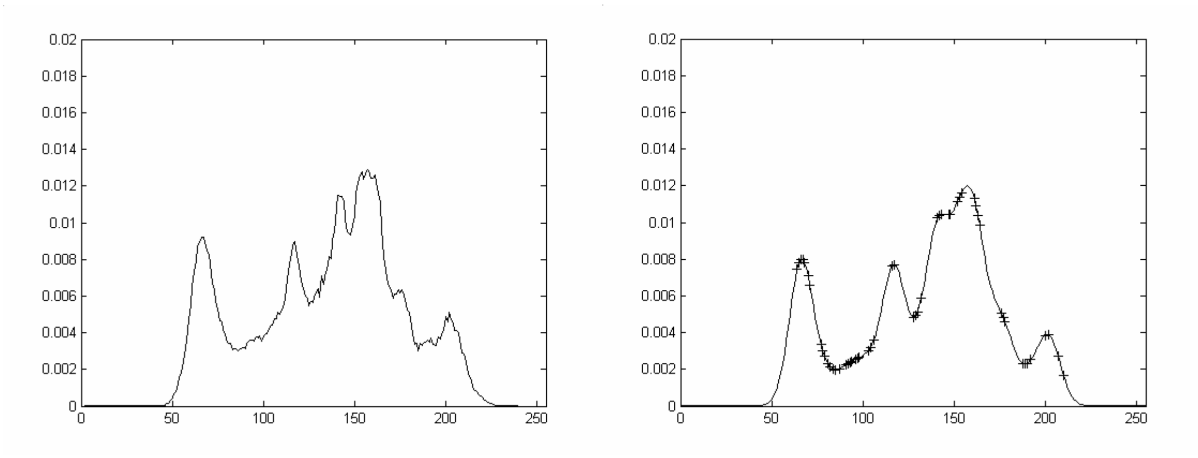


(a)



(b)

Fig. 1. (a) Compact histogram of the ‘bacteria’ image (b) SVR histogram and BSVs (labeled by “+”).



(a) (b)

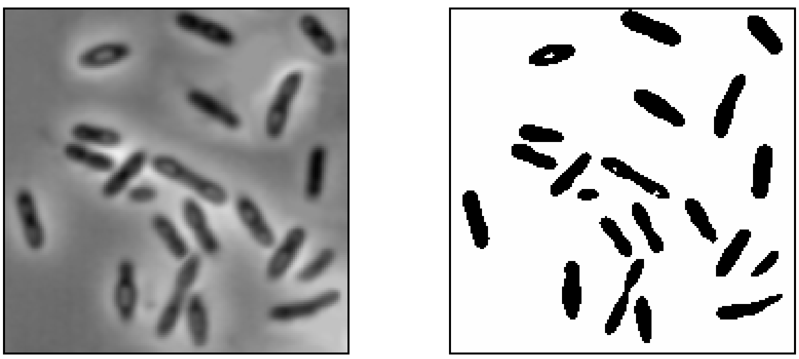
Fig. 2. (a) Compact histogram of the ‘Lena’ image (b) SVR histogram and BSVs (labeled by “+”)

3 Experiments

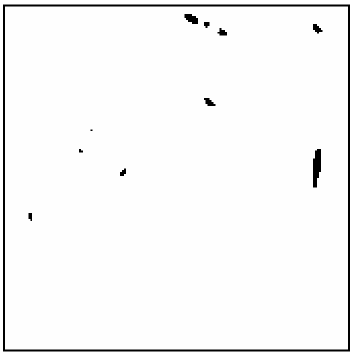
In this section, two types of experiments are carried out to evaluate the performance of the proposed algorithm. The first experiment is used to evaluate the bi-thresholded images by using our approach. The second one is used to justify the performance of multi-thresholding.

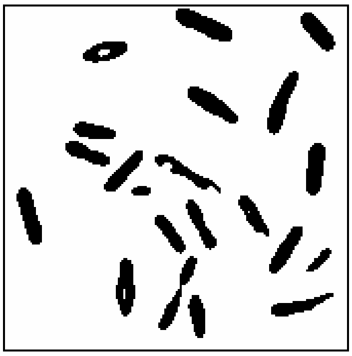
3.1 Bilevel thresholding

The threshold values for Fig. 3-4(a) were computed by using Belkasim *et al*’s phase-based optimal image thresholding [13], Chung *et al*’s fast adaptive PNN-based thresholding algorithm 1 [14] and our approach based on histogram regression, respectively. We choose such two approaches as a comparison because they are more recent and effective approaches to the problem. The thresholded images are shown in Fig. 3-4(b-d). Obviously, for such images, both the phase based approach and PNN-based approach perform visually poor compared to our approach. Our approach can select the thresholds automatically and effectively, and the resulting images can preserve the main features of the components of the original images very well.



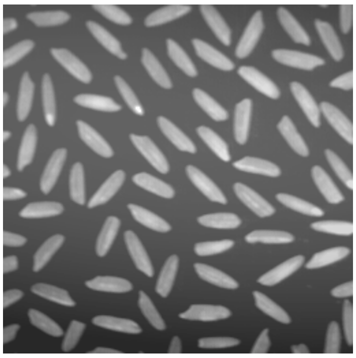
(a) (b)

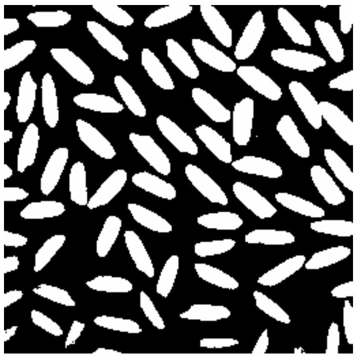




(c) (d)

Fig. 3 (a) Original 'bacteria' image; (b) our thresholded image; (c) the thresholded image by phase-based approach; (d) the thresholded image by PNN-based approach





(a) (b)

(c) (d)

Fig. 4 (a) Original 'rice' image; (b) our thresholded image; (c) the thresholded image by phase-based approach; (d) the thresholded image by PNN-based approach

3.2 **Multi-level thresholding**

In the second experiment, we show the result of multi-thresholding using our approach. Because the phase-based approach is only applicable for bi-thresholding, we only make a comparison for the thresholded images between our approach and the PNN-based approach. For convenience, we only consider the case of selecting two threshold values although the same principle is also applicable for selecting more threshold values in terms of need.

Since the criterion used in each algorithm is different, the determined thresholds are somewhat different from each other. Here, the main features of the thresholded images are examined for both approaches. For the thresholded image of Lena, we mainly compare five features [14]: (1) the nose, (2) the lip, (3) the cheek, (4) the stumps, and (5) the shoulder. After comparing the two related thresholded images as shown in Figs. 5(b)-(c) with Fig. 5(a), the above five features comparison is illustrated in Table 1. Here, for each feature, we use “fair” or “good” to grade its feature-preserving capability. From Table 1, for Lena, it is observed that the thresholded image of our approach has the relatively better quality exhibition.

(a) (b) (c)

Fig. 5 (a) Original ‘lena’ image; (b) our approach’s thresholded image (at 85, 148); (c) the thresholded image by PNN-based approach (at 100, 139)

Table 1. Feature-preserving comparison of the thresholded image for Lena

	Ours	[14]’s
(1) Nose	Good	Fair
(2) Lip	Good	Good
(3) Cheek	Fair	Good
(4) Stumps	Good	Fair
(5) Shoulder	Fair	Fair

For the thresholded image of the F-16 plane, we mainly compare seven features [14]: (1) the entrance with shape □, (2) the F-16 mark, (3) the star signature, (4) the text “U.S.AIR FORCE”, (5) the belly, (6) the cloud, and (7) the ID number 01568. After comparing the two related thresholded images as shown in Figs. 6(b)-(c) with Fig. 6(a), the above seven features comparison is illustrated in Table 2. From Table 2, for plane, it is observed that the thresholded image of our approach has the relatively better quality exhibition.

(a) (b) (c)

Fig. 6 (a) Original 'plane' image; (b) our approach's thresholded image (at 105, 138); (c) the thresholded image by PNN-based approach (at 76, 154)

Table 2. Feature-preserving comparison of the thresholded image for plane

	Ours	[14]'s
(1) Entrance with shape □	Fair	Fair
(2) F-16 mark	Good	Good
(3) Star signature	Good	Good
(4) Text "U.S.AIR FORCE"	Good	Good
(5) Belly	Good	Fair
(6) Cloud	Good	Good
(7) ID # 01568	Good	Fair

Combining the above feature-preserving comparison, it comes to a conclusion that the thresholded images of our proposed approach have an encouraging feature-preserving capability.

4 Conclusions and future works

This paper introduces an automatic threshold determination approach that is based on the support vector regression of the image histogram function. Experimental results obtained here demonstrate that our approach performs better than both the phase-based approach and PNN-based approach.

We need to point out that the results presented here only a first step in the exploration of SVR's application to image segmentation. The scheme is both novel and computationally efficient due to 1) SVR approximation just for a very small scale (≤255) of data points and 2) selecting thresholds only from the BSVs whose distribution is sparse and thus its size is also very small. However, there are still a multitude of open questions await exploration. For example, the selection of optimum threshold values from support vectors needs further investigation. It is our hope that this paper will generate sufficient interest and entice other researchers to join our effort in advancing the frontiers of this new endeavor.

Acknowledgements

We would like to thank Peng Zhou and Daohong Li for their experiments of the fast adaptive PNN-based thresholding algorithm 1 and the phase-based optimal image thresholding respectively. And we also give thanks to National Science Foundations of China and of Jiangsu under Grant Nos. 60271017 and BK2002092, Jiangsu Natural Science Key Project, Jiangsu "QingLan" Project Foundation and the Returnee's Foundation of China Scholarship Council for partial supports respectively.

References

1. M. Kamel and A. Zhao, "Extraction of binary character/graphics images from grayscale document images," *Graph. Models Image Process.*, vol. 55, no. 3, pp. 203-217, 1993.
2. T. Abak, U. Baris, and B. Sankur, "The performance of thresholding algorithms for optical character recognition," In: *Proc. Intl. Conf. Document Anal. Recog. ICDAR'97*, 1997, pp. 697-700.
3. O. D. Trier and A. K. Jain, "Goal-directed evaluation of binarization methods," *IEEE Trans. Pattern Anal. Mach. Intell.,*vol.17, pp. 1191-1201, 1995.
4. B. Bhanu, "Automatic target recognition: state of the art survey," *IEEE Trans. Aerosp. Electron. Syst.,* vol. 22, pp. 364-379, 1986.
5. M. Sezgin and R. Tasaltin, "A new dichotomization technique to multilevel thresholding devoted to inspection applications," *Pattern Recogn. Lett.,* 21, pp. 151-161, 2000.
6. M. Sezgin and B. Sankur, "Comparison of thresholding methods for non-destructive testing applications," In: *IEEE ICIP'2001, Intl. Conf. Image Process.*, 2001, pp. 764-767.
7. J. N. Kapur, P. K. Sahoo and A. K. C. Wong, "A new method for gray-level picture thresholding using the entropy of the histogram," *Comput. Vis. Graph. Image Process.*, vol. 29, pp. 273-285, 1985.
8. T. Pun, "A new method for gray-level picture thresholding using the entropy of the histogram," *Signal Process.*, vol. 2, pp. 223-237, 1980.
9. A. D. Brink, "Thresholding of digital images using two dimensional entropies," *Pattern Recog*., vol. 25, pp. 803-808, 1992.
10. W. Tsai, "Moment-preserving thresholding: a new approach," *Comput. Vis. Graph. Image Process*., vol. 29, pp. 377-393, 1985.
11. J. Kittler and J. Illingworth, "Minimum error thresholding," *Pattern Recog.*, vol. 19, pp. 41-47, 1986.
12. T. Kurita, N. Otsu and N. Abdelmalik, "Maximum likelihood thresholding based on population mixture models," *Pattern Recog*., vol. 25, pp. 1231-1240, 1992.
13. S. Belkasim, A. Ghazal and O. A. Basir, "Phase-based optimal image thresholding," *Digital Signal Processing*, vol. 13, pp. 636-655, 2003.
14. K.-L. Chung, W.-Y.Chen, "Fast adaptive PNN-based thresholding algorithms," Pattern Recog., 36, pp. 2793-2804, 2003.