Efficient Polynomial Implementation of Several Multithresholding Methods for Gray-Level Image Segmentation

David Menotti[1](✉), Laurent Najman[2], and Arnaldo de A. Araújo[3]

[1] Federal University of Paraná, Curitiba, PR, Brazil
menotti@inf.ufpr.br
[2] Groupe ESIEE Paris, Université Paris-Est, Noisy-le-Grand, France
[3] Federal University of Minas Gerais, Belo Horizonte, MG, Brazil

Abstract. Multithresholding consists of segmenting a histogram of image in classes by using thresholds. Many researchers avoid the exponential space problem of possible thresholds combinations of a given criteria function. In this work, we present a polynomial easy-to-implement dynamic programming algorithm to find the exact optimum thresholds of three well-known criteria functions for multithresholding: the maximum of histogram between class-variance (Otsu's method); the maximum histogram entropy (Kapur *et al.*'s method), and minimum histogram error (Kittler and Illingworth's method). The algorithm, that has been used to optimum quantization, has $O((K-1)L^2)$ time complexity, where K and L stand for the number of desired classes and the number of gray levels in the image, respectively. Experiments showed that the exact optimum thresholds for gray-level image segmentation can be found in less than 160 milliseconds in a Pentium 4-2GHz, in whatever the number of classes.

Keywords: Segmentation · Multithresholding · Dynamic programming

1 Introduction

Threshold selection is the simpler image segmentation method [6]. This method consists of selecting a threshold value in a histogram of image, which separates the objects from the background. In many applications of image processing and pattern recognition, the gray-level of pixels belonging to the object are substantitally different from those belonging to the background. Hundreds works were developed based on this assumption [6].

A class of threshold selection methods consists of finding the threshold value on histogram which optimizes a given function. In the following, we cite three well-known methods of this class. Otsu [5] proposed to maximize the separability of the resultant thresholded histogram classes, by using the between-class variance criterion associated with them. Kapur *et al.*[1] proposed a method based on information theory concepts to find a suitable threshold set such that the maximum entropy can be obtained by the segmented histogram classes. Based on

A. Pardo and J. Kittler (Eds.): CIARP 2015, LNCS 9423, pp. 350–357, 2015.
DOI: 10.1007/978-3-319-25751-8_42

assumption that the probability distributions of gray-level objects in an image are Gaussianly distributed, Kittler and Illingworth [2] proposed the minimum error thresholding method. The optimum threshold is the one which minimizes the error rate of the resultant thresholded histogram classes with the desired mixtures of Gaussian distributions. In the remainder of this work, due to space constrain and simplicity, we will only refer to Otsu, Kapur and Kittler methods.

Multiple threshold selection, or simply multithresholding, consisting of segmenting a histogram of image in classes by using thresholds. Several threshold selection methods can be directly extended to multiple threshold selection, such as the three ones cited earlier. Nevertheless finding the optimum thresholds for these extended multithresholding methods become computationally very expensive. One need to test all possible thresholds combinations, which increase exponentially as the number of desired classes increase. For instance, to find the seven optimum thresholds (which segment an image in eight classes) of a given criteria function of an image with 256 gray levels, we should test $256!/(256-7)!$ possible thresholds combinations. If we take a computer that can test 1 billion of threshold combinations by second, it shall take about two years to test all possible combinations. In Liao *et al.* [3] and Wu *et al.* [7]'s works, the time complexity order of the search algorithm to find the optimum thresholds for Otsu, Kapur and Kittler methods was decreased. However it still continue exponential.

In this paper, we present a polynomial easy-to-implement dynamic programming algorithm to find the optimum thresholds for these methods, which has $O((K-1)L^2)$ time complexity, where K and L stand for the number of desired classes and the number of gray levels in the image, respectively. Note that there are in the literature faster algorithms than ours [4], $O(KL)$, but their implementation are not straightforward.

The rest of this paper is organized as follows. Basic definitions are presented in Section 2. The criteria functions of three multithresholding methods are described in Section 3. Section 4 presents the polynomial dynamic programming (DP) algorithm and shows an implementation for it. Experiments concerning the run time of the algorithm for the three methods and a segmentation example are shown in Section 5. Finally, conclusions are pointed out in Section 6.

2 Basic Definitions

2.1 Images and Histograms

Let $\mathbb{N}$ denote the set of natural numbers. Let X be a subset of points $(x, y) \in \mathbb{N}^2$, such that $0 \leq x < m$, and $0 \leq y < n$, where m and n denote the dimensions of X. Let $|Y|$ denote the the cardinality of a set $Y \subseteq \mathbb{N}^2$. Note that $|X| = m \times n$. A mapping I, from X to Z_L, where $Z_L = \{0, ..., L-1\}$, is called an *image*. In applications, L is typically 256. For a point $(x, y) \in X$, $l = I(x, y)$ is called the *level* of the point (x, y) on I.

Let X_l be a subset of X, such that for all $(x, y) \in X_l \subseteq X$, we have $I(x, y) = l$. Let $H(l)$ be the *absolute frequency* of level l in image I, i.e. $H(l) = |X_l|$. Note that $H(l) = 0$ if there is no $(x, y) \in X$ such that $I(x, y) = l$. Let P_l denote the probability of one pixel having the gray-level l, i.e. $P_l = H(l)/(m \times n)$. Note that $\sum_{l=0}^{L-1} H(l) = m \times n$, and $\sum_{l=0}^{L-1} P_l = 1$. The mapping H from the levels of image I to its absolute frequency levels, i.e. $H : Z_L \rightarrow \mathbb{N}$, is called the *histogram* of image I.

2.2 Thresholding

Usually a multithresholding histogram-based method uses $K-1$ thresholds (i.e. $T = \{t_0, t_1, ..., t_{K-2}\}$) to partition the histogram of an image in K classes. That is, the histogram is partitioned in classes $C_k = \{H(i) | s_k \leq i \leq f_k\}$ with $0 \leq k < K$, where s_k and f_k stand for the starting and the final histogram class boundaries, respectively. They are defined as: $s_k = 0$ if $k = 0$, and $s_k = t_{k-1} + 1$ otherwise; $f_k = L - 1$ if $k = K - 1$, and $f_k = t_k$ otherwise.

2.3 Histogram Statistics

Now let us present definitions that will help us to describe the thresholding methods and the DP algorithm. Let I be an image with L levels and let H be the corresponding histogram. We define the ath-order histogram statistics of the histogram class $H(p, q)$ as

$$S^a(p, q) = \sum_{l=p}^{q} l^a \times H(l). \tag{1}$$

By using this definition, let us define the cumulative probability and the mean of the histogram class $H(p, q)$ as

$$\omega(p, q) = S^0(p, q)/(m \times n), \tag{2}$$

$$\mu(p, q) = S^1(p, q)/S^0(p, q), \tag{3}$$

respectively.

Let us introduce one more definition, which will help us to simplify further explanations, the ath-order histogram statistics error of the histogram class $H(p, q)$ regarding to the level b, i.e.

$$E^a(p, q, b) = \sum_{l=p}^{q} |l - b|^a \times H(l). \tag{4}$$

Finally, we set the $H(p, q)$ histogram class variance as

$$\begin{aligned} \sigma^2(p, q) &= \left(\sum_{l=p}^{q} |l - \mu(p, q)|^2 \times P_l \right) / \omega(p, q) \\ &= E^2(p, q, \mu(p, q))/S^0(p, q). \end{aligned} \tag{5}$$

Equations 1 and 4 (and consequently the other ones defined in this section) can be computed with $O(L^2)$ complexity for all possible $0 \leq p \leq q < L$ if we use a recursive definition as

$$\alpha(p, q) = \begin{cases} \beta & \text{if } p = q, \\ \beta + \alpha(p, q - 1) & \text{otherwise.} \end{cases} \tag{6}$$

where β stands for the last computation at the equation, namely, when $l = q$.

3 Thresholding Criterion Functions

In this section, we describe three well-known criteria functions ($\psi(T)$) of multithresholding methods for gray-level image segmentation: ($\psi_{BCV}(T)$) the between class variance [5], ($\psi_{Ent}(T)$) the maximum entropy [1], and ($\psi_{ME}(T)$) the minimum error [2] criteria. These functions can be represented and decomposed as histogram class factors, i.e. $\varphi(s_k, f_k)$.

The optimum threshold set T^* is obtained as the argument value which optimizes a given criteria function, i.e.

$$T^* = \arg \operatorname*{opt}_{0 \leq T < L} \psi(T). \tag{7}$$

where opt is a extremum operator depending on criteria function. That is, max for Otsu and Kittler methods, and min for Kapur method.

3.1 The Between-Class Variance

In Otsu's work [5], three possible discriminant criterion functions based on ratios of the within-class, the between-class and that of total variance are presented. All of these are equivalent to each other for the evaluation of the optimum thresholding process. Otsu suggested to maximize the between-class variance because it is the simplest one for computation. The between-class variance criteria function can be computed as

$$\psi_{BCV}(T) = \sum\nolimits_{k=0}^{K-1} \varphi(s_k, f_k), \tag{8}$$

where the histogram class factor $\varphi(s_k, f_k)$ is

$$\varphi(s_k, f_k) = \omega(s_k, f_k)[\mu(s_k, f_k) - \mu(0, L-1)]^2. \tag{9}$$

3.2 The Maximum Entropy

In the maximum entropy-based thresholding, the optimum threshold is obtained by applying information theory. That is, the optimal threshold set (T_{Ent}) is the one that maximizing the information content of the histogram image. As derived from Kapur *et al.*'s work [1], the original gray-level distribution of the image is divided into a number of classes of probability distributions in the multilevel thresholding case. Then the entropies associated with these distributions are computed as

$$\varphi(s_k, f_k) = -\sum_{l=s_k}^{f_k} \frac{P_l}{\omega(s_k, f_k)} \log \frac{P_l}{\omega(s_k, f_k)}. \tag{10}$$

And, then the criteria function is defined as

$$\psi_{Ent}(T) = \sum\nolimits_{k=0}^{K-1} \varphi(s_k, f_k). \tag{11}$$

3.3 The Minimum Error

In the concept of minimum error thresholding, the gray-level histogram of the image is thought of as an estimate of the probability density function $p(l)$ of the mixture distribution, comprising the gray-level of several classes (i.e. objects and background). It is assumed that each of these class distributions $p(l|k)$ of the mixture follows a normal distribution, with a class standard deviation of σ_k, a class mean of μ_k, and a priori probability of ω_k; hence, the histogram can be approximated as:

$$p(l) = \sum_{k=0}^{K-1} \frac{\omega(s_k, f_k)}{\sigma^2(s_k, f_k)\sqrt{2\pi}} \exp^{-(l-\mu(s_k,f_k))^2/2\sigma^2(s_k,f_k)} \tag{12}$$

The optimum threshold (T_{ME}) can be determined by solving an resultant quadratic equation with respect to l [2]. However, the parameters $\omega(s_k, f_k)$, $\mu(s_k, f_k)$, and $\sigma^2(s_k, f_k)$ of the mixture density function $p(l)$ associated with the image are unknown. In order to overcome the difficulties of estimating the unknown parameters, Kittler and Illingworth [2] presented a criterion function $\psi_{ME}(T)$, which is given by:

$$\psi_{ME}(T) = 1 + 2\sum_{k=0}^{K-1} \varphi(s_k, f_k) \tag{13}$$

where

$$\varphi(s_k, f_k) = \omega(s_k, f_k)[\log(\sigma(s_k, f_k)) + \log(\omega(s_k, f_k))] \tag{14}$$

4 Dynamic Programming Algorithm

Thanks to decomposition of the criteria functions in histogram class factors $\varphi(s_k, f_k)$ among other mathematical properties we can apply the polynomial DP algorithm to find the optimum thresholds for image segmentation which is described in this section. A proof of correctness of the algorithm is not show due to space constrain.

Let $\psi(p, q)$ be an optimum criteria function conceived on the $q+1$-first gray-levels of the histogram of image when segmented by p-first optimum thresholds $(t_0, t_1, ..., t_{p-1})$, which can be written as the sum of the histogram class factors φ's, i.e.

$$\psi(p, q) = \sum_{k=0}^{p} \varphi(s_k, f_k^{p,q}), \tag{15}$$

where $f_k^{p,q} = q$ if $k = p$, and $f_k^{p,q} = t_k$ otherwise.

Let us define $\psi(p, q)$ as a recurrence equation as follows. Initially, we are interested to know the optimum criteria conceived on the $q+1$-first gray-level of the histogram of image when no threshold is used to segmented the histogram, i.e. $\psi(0, q)$. It can be directly computed as the histogram class factor $\varphi(0, q)$, i.e

$$\psi(0, q) = \varphi(0, q) \tag{16}$$

Algorithm 1. Computing $\psi(K-1, L-1)$

Data: $\varphi(p, q)$ - histogram class contribution
Result: ψ - criteria function
Result: DPT - optimum thresholds matrix

1 **for** $q = 0 \ldots L-1$ **do** $\psi(0, q) \leftarrow \varphi(0, q)$;
2 **for** $p = 1 \ldots K-1$ **do**
3     $\psi(p, p) \leftarrow \psi(p-1, p-1) + \varphi(p, p)$;
4     $DPT(p, p) \leftarrow p-1$;
5   **for** $q = p+1 \ldots L-K+p$ **do**
6         $\psi(p, q) \leftarrow -\infty$;
7   **for** $l = p-1 \ldots q-1$ **do**
8   **if** $(\psi(p, q) > \psi(p-1, l) + \varphi(l+1, q))$ **then**
9                 $\psi(p, q) \leftarrow \psi(p-1, l) + \varphi(l+1, q)$;
10                 $DPT(p, q) \leftarrow l$;

Once we have the solution computed up to the level $p-1$, i.e. $\psi(p-1, q)$, for all possible q, we are interested in computing the $\psi(p, q)$. It can be computed as the optimum (e.g. most or least depending on criteria function) term composed by the optimum criteria function conceived on the $l-1$-first gray-levels of the histogram of image being segmented by $p-1$ thresholds ($\psi(p-1, l)$) and the histogram class factor $\varphi(l+1, q)$, i.e.

$$\psi(p, q) = \underset{p-1 \leq l < q}{\mathrm{opt}} \left(\psi(p-1, l) + \varphi(l+1, q)\right) \tag{17}$$

for all $p \leq q$, with $0 \leq p < K$, and $0 \leq q < L$, and where opt stands for max or min depending on criteria function.

To recover the thresholds that yield the optimum criteria function, we have to store the thresholds obtained from the dynamic programming algorithm, for future backwards searching, as follows

$$DPT(p, q) = \begin{cases} p-1, \text{ if } p = q, \\ \arg \mathrm{opt}_{p-1 \leq l < q} \left(\psi(p-1, l) + \varphi(l+1, q)\right), \text{ otherwise,} \end{cases} \tag{18}$$

for all $p \leq q$, with $0 \leq p < K$, and $0 \leq q < L-1$. From $DPT(K-1, L-1)$ we can find the $K-1$ optimum thresholds, i.e. $T^* = \{t_0^*, t_1^*, ..., t_{K-2}^*\}$, as

$$t_{n-1}^* = DPT(n, X_n) \tag{19}$$

where, $X_n = L-1$ if $n = K-1$, and $X_n = t_n^*$ otherwise.

Algorithm 1 shows an implementation for the dynamic programming algorithm. It is composed of two main loops, where the second one is enchained by other two loops, which gives the cubic time complexity of the algorithm, i.e. $O((K-1)L^2)$. Note that the Algorithm 1 maximize the criteria function. If minimizitation is required (as for Kittler method), we have only to modify two lines. To use $+\infty$ in line 6, and $<$ in the comparison in line 8.

Fig. 1. From left to right, the classic lena image (512 × 512 pixels), its segmented images in eight classes (using seven thresholds) by Otsu (61 87 111 131 148 167 191), Kapur (60 87 112 136 159 180 202) and Kittler (64 89 115 135 149 163 184) methods, respectively.

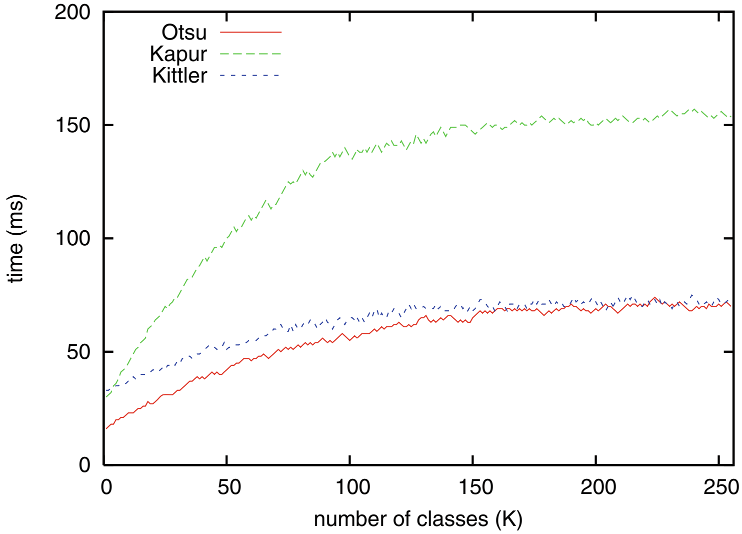

Fig. 2. Run time curves for Otsu, Kapur, and Kittler methods using lena image (512x512 pixels), by varing K from 1 up to 256.

5 Experiments

In this section, we show an experiment concerning the run time of the dynamic programming algorithm. We ran the algorithm using as input the classical lena image (with 512 × 512 pixels) for the three presented multithrehsolding methods. Fig. 2 shows the run time curves obtained from each method (using the DP algorithm) by varying the number of desired classes K from 1 up to 256. We can observe a significant difference among the response time of Kittler and Otsu methods regarding to Kapur method. This fact is due to an overhead on consistency verification of histogram class factors, i.e. φ, for Kapur method. By analyzing the upper-bound $O((K-1)L^2)$ and the curves shown in Fig. 2, we can say that the polynomial dynamic programming algorithm works (sub)linearly regarding to the number of classes K. In practice, our algorithm segments an image with 512×512 pixels in less than 160 milliseconds on a machine with architecture Pentium 4 - 2GHz, in whatever the number of classes.

Fig. 1 shows an example of segmentation: image of lena and its segmentation in eight (8) classes for Otsu, Kapur and Kittler methods. The optimum thresholds are shown in parenthesis. These images were segmented each one in less than 50 milliseconds in same architecture described above.

6 Conclusion

In this work, we presented and tested a polynomial algorithm to find the exact optimum threshold for three well-know methods: Otsu [5], Kapur [1], and Kittler [2]. This algorithm can be used to find the optimum thresholds of several other multithresholding methods. In a forthcoming work, we will establish the necessary and sufficient conditions which the criterion of a multithresholding method must respect to be solved by such polynomial algorithm. As future works, we want to join to the presented dynamic programming algorithm a divide-and-conquer strategy. This new algorithm to find the optimum thresholds will have $O(KN \log N)$ time complexity.

Acknowledgments. We thank UFPR, UFMG, FAPEMIG, CAPES, and CNPq for the support.

References

1. Kapur, J., Sahoo, P., Wong, A.: A new method for gray-level picture thresholding using the entropy of the histogram. Computer Vision, Graphics and Image Processing **29**, 273–285 (1985)
2. Kittler, J., Illingworth, J.: Minimum error thresholding. Pattern Recognition **1**(19), 41–47 (1986)
3. Liao, P.S., Chen, T.S., Chung, P.C.: A fast algorithm for multilevel thresholding. Journal of Information Science and Egineering **17**, 713–727 (2001)
4. Luessi, M., Eichmann, M., Schuster, G.: New results on efficient optimal multilevel image thresholding. In: IEEE Int. Conf. on Image Processing, pp. 773–776 (2006)
5. Otsu, N.: A threshold selection method from grey-level histograms. IEEE Transactions on Systems, Man and Cybernetics **9**(1), 41–47 (1979)
6. Sezgin, M., Sankur, B.: Survey over image thresholdintg techniques and quantitative performance evaluation. Journal of Electronic Imaging **13**(1), 146–165 (2004)
7. Wu, B.F., Chen, Y.L., Chiu, C.C.: Efficient implementation of several multilevel thresholding algorithms using a combinatorial scheme. International Journal of Computer and Applications **28**(3), 259–269 (2006)