A Fast Thresholded Linear Convolution Representation of Morphological Operations

Branislav Kisačanin and Dan Schonfeld

Abstract—**In this correspondence, we present a fast thresholded linear convolution representation of morphological operations. The thresholded linear convolution representation of dilation and erosion is first proposed. A comparison of the efficiency of the direct implementation of morphological operations and the thresholded linear convolution representation of morphological operations is subsequently conducted.**

I. Introduction

In recent years, mathematical morphology has emerged as a powerful new tool for image processing [1]–[5]. An important example of morphological operations is provided by the dilation and erosion. It has been shown that the dilation and erosion can be used for the representation of all morphological operations [1].

The practical utility of mathematical morphology is often dependent on the existence of efficient methods for the implementation of morphological operations. An efficient implementation of the dilation and erosion is thus fundamental to the efficient implementation of morphological operations.

A tremendous effort has been directed at the efficient implementation of morphological operations [6]–[11]. Numerous approaches to the parallel implementation of morphological operations have been proposed [6]–[9]. Several methods for the decomposition of (large) structuring elements have also been proposed [10], [11]. Nonetheless, an improvement in the efficient implementation of morphological operations would significantly enhance its practical utility.

In this correspondence, we present a fast thresholded linear convolution representation of morphological operations. In Section II, the basic definitions of the dilation and erosion are introduced. In Section III, the thresholded linear convolution representation of dilation and erosion is proposed. In Section IV, a comparison of the efficiency of the direct implementation of morphological operations and the thresholded linear convolution representation of morphological operations is conducted. Finally, in Section V, we provide a brief summary and discussion of our results.

II. Mathematical Morphology

In this section, we introduce the basic definitions of the dilation and erosion. For a more comprehensive presentation of mathematical morphology, refer to [1]–[5].

In this presentation, we shall restrict ourselves to n-dimensional discrete and binary functions, i.e., $\{f(x) \in \{0,1\} : x \in \mathcal{Z}^n\}$. The *dilation* $[f \oplus s](x)$ of a function $f(x)$ by a structuring element $s(x)$ is given by

$$[f \oplus s](x) = \bigcup_{\{y \in \mathcal{Z}^n : s(y)=1\}} f(x-y); \tag{1}$$

whereas the *erosion* $[f \ominus s](x)$ of a function $f(x)$ by a structuring element $s(x)$ is given by

$$[f \ominus s](x) = \bigcap_{\{y \in \mathcal{Z}^n : s(y)=1\}} f(x-y). \tag{2}$$

It is important to notice that the dilation and erosion are equivalent to the Minkowski addition and subtraction, respectively [1]–[5].[1]

The *complement* $f^c(x)$ of a function $f(x)$ is given by $f^c(x) = 1 - f(x)$, for every $x \in \mathcal{Z}^n$.

Property (Duality):

$$[f \oplus s](x) = ([f^c \ominus s](x))^c; \tag{3}$$

and

$$[f \ominus s](x) = ([f^c \oplus s](x))^c. \tag{4}$$

From the duality property, we observe that the representation of the dilation is equivalent to the representation of the erosion (and vice versa).

III. Thresholded Linear Convolution

In this section, we propose the threshold linear convolution representation of dilation and erosion.

The *threshold* $h_\tau(\cdot)$ by $\tau \in \mathcal{R}$ is given by

$$h_\tau(x) = \begin{cases} 0, & x \le \tau \\ 1, & x > \tau; \end{cases} \tag{5}$$

for every $x \in \mathcal{R}$.

The *linear convolution* $[f_1 * f_2](x)$ of functions $f_1(x)$ and $f_2(x)$ is given by

$$[f_1 * f_2](x) = \sum_{y \in \mathcal{Z}^n} [f_1(x-y) \cdot f_2(y)] \tag{6}$$

for every $x \in \mathcal{Z}^n$ [12].

In the following proposition, we provide a thresholded linear convolution representation of the dilation.

Proposition 1:

$$[f \oplus s](x) = h_\tau([f * s](x)) \tag{7}$$

for every $0 \le \tau < 1$.

Proof: From (1), (5), and (6), we observe that

$$\begin{aligned} [f \oplus s](x) &= \bigcup_{\{y \in \mathcal{Z}^n : s(y)=1\}} f(x-y) \\ &= \bigcup_{\{y \in \mathcal{Z}^n : s(y)=1\}} [f(x-y) \cap s(y)] \\ &= \bigcup_{y \in \mathcal{Z}^n} [f(x-y) \cap s(y)] \\ &= \bigcup_{y \in \mathcal{Z}^n} [f(x-y) \cdot s(y)] \\ &= h_\tau\left(\sum_{y \in \mathcal{Z}^n} [f(x-y) \cdot s(y)]\right) \\ &= h_\tau([f * s](x)). \end{aligned}$$

This completes the proof. □

In the following proposition, we provide a thresholded linear convolution representation of the erosion.

[1] Often, the dilation (resp., erosion) is defined as the Minkowski addition (resp., subtraction) by the symmetric of the structuring element [1]–[5].

Manuscript received June 3, 1993; revised August 30, 1993. This work was supported by the Office of Naval Research Award N00014-91-J-1725. The associate editor coordinating the review of this paper and approving it for publication was Prof. Nikolas P. Galatsanos.

The authors are with the Signal and Image Research Laboratory, the Department of Electrical Engineering and Computer Science, University of Illinois, Chicago, IL 60607-7053.

IEEE Log Number 9400310.

Proposition 2:

$$[f \ominus s](x) = (h_\tau([f^c * s](x)))^c \tag{8}$$

for every $0 \leq \tau < 1$.

Proof: From (2), (5), and (6), we observe that

$$\begin{aligned}[f \ominus s](x) &= \bigcap_{\{y\in \mathcal{Z}^n : s(y)=1\}} f(x-y)\\ &= \left(\bigcup_{\{y\in \mathcal{Z}^n : s(y)=1\}} f^c(x-y)\right)^c\\ &= \left(\bigcup_{y\in \mathcal{Z}^n : s(y)=1} [f^c(x-y)\cap s(y)]\right)^c\\ &= \left(\bigcup_{y\in \mathcal{Z}^n} [f^c(x-y)\cap s(y)]\right)^c\\ &= \left(\bigcup_{y\in \mathcal{Z}^n} [f^c(x-y)\cdot s(y)]\right)^c\\ &= \left(h_\tau\left(\sum_{y\in \mathcal{Z}^n} [f^c(x-y)\cdot s(y)]\right)\right)^c\\ &= (h_\tau([f^c * s](x)))^c.\end{aligned}$$

This completes the proof. □

The thresholded linear convolution representation of dilation and erosion proposed will be used for the efficient implementation of morphological operations in the next section.

IV. Simulations

In this section, we conduct a comparison of the efficiency of the direct implementation of morphological operations and the threshold linear convolution representation of morphological operations.

The merit of the proposed thresholded linear convolution representation of morphological operations is contingent on the low computational complexity of the linear convolution [12]–[13]. A standard approach to the implementation of the linear convolution is based on the fast Fourier transform (FFT) [12]. Far more powerful methods to the implementation of the linear convolution have also been proposed, e.g., Winograd and Agarwal–Cooley algorithms [13]. For simplicity, however, we restrict our investigation of the computational complexity to the thresholded linear convolution representation of the dilation and erosion based on the FFT.

Let us consider a function of size N^n and a structuring element of size M^n. The direct implementation of the dilation and erosion requires $\mathcal{O}(N^n M^n)$ logical operations (see (1) and (2)), whereas the thresholded linear convolution representation of the dilation and erosion based on the FFT yields $\mathcal{O}(n(N+M)^n \log(N+M))$ multiplications [12].

A computer simulation comparison of the resulting performance of the direct implementation of morphological operations and the thresholded linear convolution representation of morphological operations based on the FFT is depicted in Fig. 1. In this comparison, we consider the cpu-time required for the dilation of a binary image of size 256×256 by a structuring element of size $2^k \times 2^k$, for $0 \leq k \leq 6$ (using a MATLAB 4.0 software package on a SPARC II workstation).

From the computer simulation, we observe that the direct implementation of morphological operations results in a large increase in cpu-time as the size of the structuring element increases, whereas the thresholded linear convolution representation of morphological operations based on the FFT yields an almost negligible increase

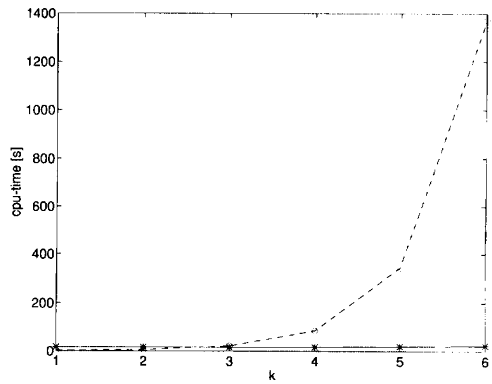

Fig. 1. Comparison of the CPU-time required for the dilation of a binary image of size 256×256 by a structuring element of size $2^k \times 2^k$, for $0 \leq k \leq 6$ (using a MATLAB 4.0 software package on a SPARC II workstation), for the direct implementation (dashed line) and the thresholded linear convolution representation based on the FFT (solid line).

in cpu-time as the size of the structuring element increases. Moreover, we also observe that the cpu-time required for morphological operations using large structuring elements is significantly lower for the thresholded linear convolution representation based on the FFT than the direct implementation. As a result, we conclude that the thresholded linear convolution representation of morphological operations based on the FFT is superior to the direct implementation of morphological operations.

V. Summary

In this correspondence, we presented a fast thresholded linear convolution representation of morphological operations. The thresholded linear convolution representation of dilation and erosion was first proposed. A comparison of the efficiency of the direct implementation of morphological operations and the thresholded linear convolution representation of morphological operations was subsequently conducted. As a result, we concluded that the thresholded linear convolution representation of morphological operations based on the FFT is superior to the direct implementation of morphological operations. The results presented in this correspondence can easily be extended to the thresholded linear convolution representation of morphological operations in the Euclidean (continuous) space. Optical methods are particularly efficient in computation of the linear convolution in the Euclidean (continuous) space [14]. Our approach can thus be used in the optical implementation of morphological operations.

References

[1] G. Matheron, *Random Sets and Integral Geometry.* New York: Wiley, 1975.

[2] J. Serra, *Image Analysis and Mathematical Morphology.* New York: Academic, 1982.

[3] ——, "Introduction to mathematical morphology," *Comput. Vision Graphics Image Processing,* vol. 35, pp. 283–305, 1986.

[4] R. M. Haralick, S. R. Sternberg, and X. Zhuang, "Image analysis using mathematical morphology," *IEEE Trans. Patt. Anal. Machine Intell.,* vol. 9, pp. 532–550, 1987.

[5] C. R. Giardina and E. R. Dougherty, *Morphological Methods in Image and Signal Processing.* Englewood Cliffs, NJ: Prentice-Hall, 1988.

[6] J. C. Klein and J. Serra, "The texture analyzer," *J. Microscopy,* vol. 95, pp. 349–356, 1972.

[7] V. Goetcherian, "From binary to gray tone image processing using fuzzy logic concepts," *Patt. Recogn.,* vol. 12, pp. 7–15, 1980.

[8] J. Pecht, "Speeding-up successive Minkowski operations with bit-plane computers," *Patt. Recogn. Lett.,* vol. 3, pp. 113–117, 1985.

[9] L. Abbott, R. M. Haralick, and X. Zhuang, "Pipeline architecture for morphologic image analysis," *Machine Vision Applications,* vol. 1, pp. 23–40, 1988.

[10] X. Zhuang and R. M. Haralick, "Morphologic structuring element decomposition," *Comput. Vision Graphics Image Processing,* vol. 35, pp. 370–382, 1986.

[11] C. H. Richardson and R. W. Schafer, "A lower bound for structuring element decompositions," *IEEE Trans. Patt. Anal. Machine Intell.,* vol. 13, pp. 365–369, 1991.

[12] D. E. Dudgeon and R. M. Mersereau, *Multidimensional Digital Signal Processing.* Englewood Cliffs, NJ: Prentice-Hall, 1984.

[13] R. E. Blahut, *Fast Algorithms for Digital Signal Processing.* Reading, MA: Addison-Wesley, 1985.

[14] J. W. Goodman, *Introduction to Fourier Optics.* New York: McGraw-Hill, 1968.