A New Algorithm for Weighted Order Statistics Operations

A. Gasteratos and I. Andreadis

Abstract— **A new algorithm suitable for the comptation of weighted order statistics operations is presented in this letter. The algorithm is based on local histogram and a successive approximation technique. The successive approximation technique ensures that the result of the weighted order statistics operation is computed in a fixed number of steps, equal to the pixel value resolution. Comparative experimental results are also included. It is shown that the proposed algorithm is faster than the quick sort algorithm for weighted order statistics operations.**

Index Terms—**Morphological operations, nonlinear filters.**

I. Introduction

Nonlinear filters are a large family of filters used in signal and image processing [1]. They include well-known filters such as rank-order and morphological filters. Rank-order filters exhibit excellent robustness properties and provide solutions in many cases where linear filters are inappropriate. The data window elements are sorted in ascending order and the output of the rank-order filter with rank k is the kth element (kth-order statistic). The output of a weighted rank-order filter of rank k with seights $\{w_1, w_2, \cdots w_N\}$ is kth-order statistic of the set $\{w_1 \diamondsuit x_1, w_2 \diamondsuit x_2, \cdots w_N \diamondsuit x_N\}$, where w_i is a natural number and $w_i \diamondsuit x_i$ denotes w_i times repetition of x_i.

Algorithms suitable for software implementation are the tree sorts, shell sorts, and quick sorts [2]. Several VLSI designs for median filters are studied in [3], where it is stated that for a relatively small pixel value resolution the threshold decomposition technique [4] is probably the best choice. However, implementation of filters capable of handling high-resolution numbers is not practical, since the complexity increases exponentially with both the data resolution and data window size. A bit-sliced algorithm for generalized rank-order filtering, suitable for hardware implementation has been proposed in [5]. A fast algorithm for median filtering (running median), based on local histogram, has been presented in [6].

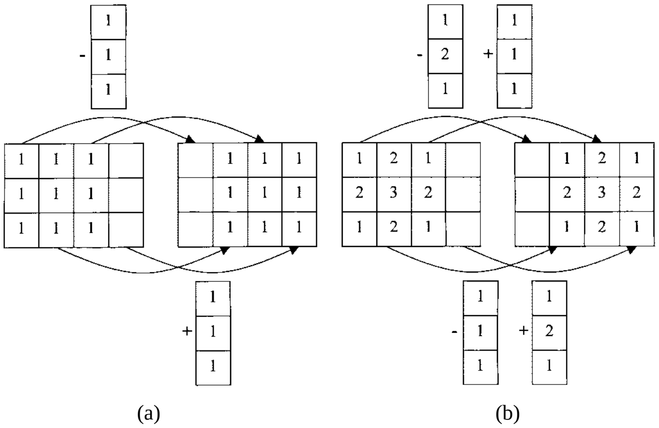

Fig. 1. (a) Demonstration of the running property of the Huang *et al.* algorithm. (b) Adaptation of the running property to a weighted case.

In this work, a new algorithm capable of performing weighted rank-order filtering is presented. The proposed algorithm is based on local histogram and a successive approximation technique, and it provides an output result in a fixed number of steps, which is equal to the image pixel value resolution. Using the proposed technique any weighted rank-order filter can be computed. Also, soft morphological operations can ve computed, provided that the required additions/subtractions of the image pixels with the structuring element pixels have been executed before applying the algorithm. Comparative experimental results show that for small image data windows, the proposed algorithm is slightly faster than the quick sort algorithm for weighted order statistics filtering. For larger windows the new algorithm is a great improvement in speed.

II. Algorithm Description

A method to compute an order statistic is by summing the values in the histogram until the desired order statistic is reached [1]. Instead of adding the local histogram values serially, a successive approximation technique is proposed. This ensures that the result is traced in a fixed number of steps. The number of steps is equal to the number b of the bits per pixel. According to the proposed successive approximation technique the N pixel values are compared to 2^{b-1}. Pixel values, which are greater than, less than or equal to that value, are marked with labels GT, LT, and EQ, respectively. GT, LT, and EQ are either zero or one. Pixel labels are then multiplied

Manuscript received October 14, 1998. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. G. Ramponi.

The authors are with the Laboratory of Electronics, Section of Electronics and Information System Technology, Department of Electrical and Computer Engineering, Democritus University of Thrace, GR-671 00 Xanthi, Greece (e-mail: iandread@demokritos.cc.duth.gr).

Publisher Item Identifier S 1070-9908(99)02459-1.

1070–9908/99$10.00 © 1999 IEEE

1	1	1	1	1	1	1	1	1
1	2	2	2	2	2	2	2	1
1	2	3	3	3	3	3	2	1
1	2	3	3	3	3	3	2	1
1	2	3	3	3	3	3	2	1
1	2	3	3	3	3	3	2	1
1	2	3	3	3	3	3	2	1
1	2	2	2	2	2	2	2	1
1	1	1	1	1	1	1	1	1

(a)

1	1	1	1	1	1	1
1	2	2	2	2	2	1
1	2	3	3	3	2	1
1	2	3	3	3	2	1
1	2	3	3	3	2	1
1	2	2	2	2	2	1
1	1	1	1	1	1	1

(b)

1	1	1	1	1
1	2	2	2	1
1	2	3	2	1
1	2	2	2	1
1	1	1	1	1

(c)

1	2	1
2	3	2
1	2	1

(d)

Fig. 2. Weight arrays used for: (a) 9 × 9, (b) 7 × 7, (c) 5 × 5, and (d) 3 × 3 weighted median filtering, respectively.

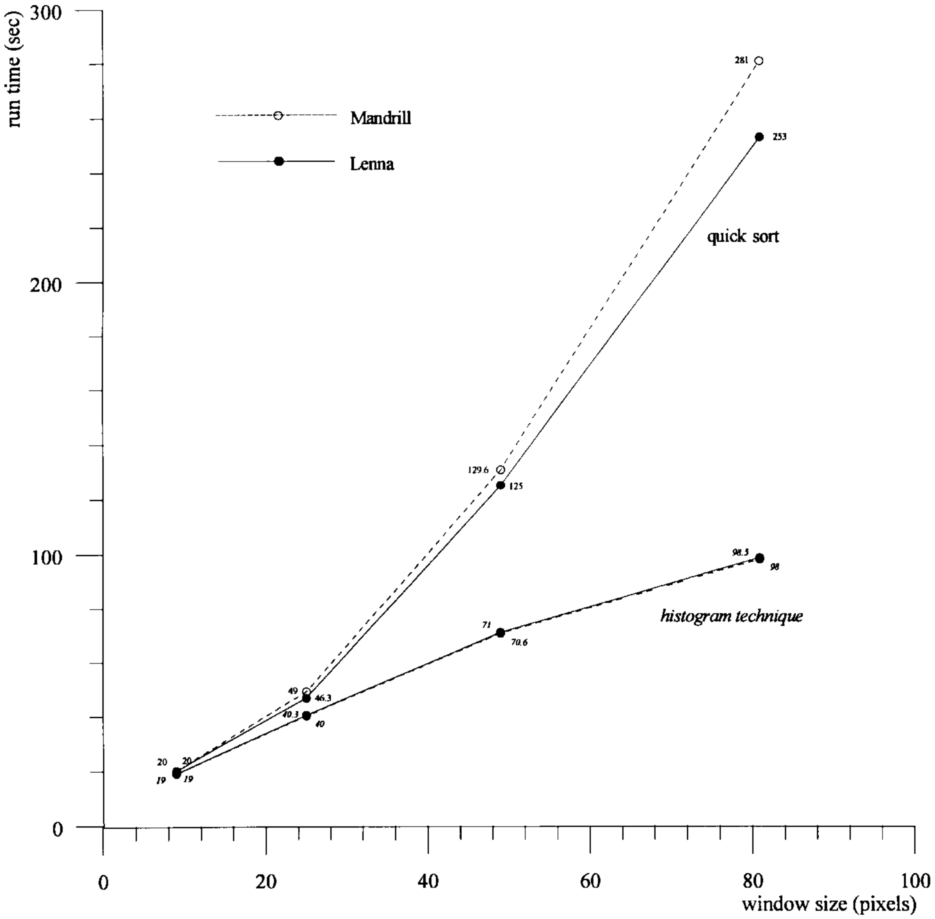

Fig. 3. Computer time per image plotted against window size.

by the corresponding pixel weight (w_j). The sum of LT's and EQ's determines whether the kth-order statistic is greater than, less than or equal to 2^{b-1}.

1) If the sum $\Sigma_{j=1}^{N} w_j(EQ_j + LT_j)$ is greater than or equal to k and the sum $\Sigma_{j=1}^{N} w_j LT_j$ is less than k, then the kth-order statistic is 2^{b-1}.
2) If the sum $\Sigma_{j=1}^{N} s_j LT_j$ is greater than or equal to k, then the kth-order statistic is less than 2^{b-1} and, therefore, 2^{b-2} should be subtracted from 2^{b-1}.
3) Otherwise, the kth-order statistic is greater than 2^{b-1} and, therefore, 2^{b-2} should be added to 2^{b-1}.

If the kth-order statistic is not traced according to 1), then the comparison of each pixel value is made with the subtraction or the addition result of 2) or 3), respectively. The process is repeated until the required weighted order statistic is computed. In the ith step 2^{b-1-i} is either subtracted or added. Thus, the kth-order statistic is computed in b steps.

Notations and the pseudocode of the algorithm is presented as follows.

Notations:

N	Number of pixels.
b	Pixel value resolution (bits).
$im_1, im_2, \cdots im_N$	Image pixels.
$w_1, w_2, \cdots w_N$	Corresponding weights.
k	Sought order statistic.
temp	Temporal result.
o	Output pixel.

```
initial
$o = 0$
temp $= 2^{b-1}$
begin
for $i = 1$ to $b$ do
    begin
    compare($im_1, im_2, \cdots im_N$: temp)
    {if $im_j =$ temp then $EQ_j = 1$ else $EQ_j = 0$
    if $im_j <$ temp then $LT_j = 1$ else $LT_j = 0$}
    if $\left(\sum_{j=1}^{N} w_j(EQ_j + LT_j) \geq k\right)$ AND
       $\left(\sum_{j=1}^{N} w_j LT_j < k\right)$
    then $o \leftarrow$temp
    elsif $\sum_{j=1}^{N} w_j LT_j \geq k$
    then temp$\leftarrow$temp$-2^{b-1-i}$
    else temp$\leftarrow$temp$+2^{b-1-i}$
    end
end.
```

A module utilizing standard comparators for comparisons, adders/subtractors for additions/subtractions and multiplexers for "if" operations can be used to implement this technique in hardware. Also, there are two ways to realize the algorithm. The first is through a loop, which feeds the temp signal back to the input b times. The second way is to use b successive modules to process the data in a pipeline fashion. The latter implementation is more hardware demanding, but results into a faster hardware structure.

The above described algorithm requires a fixed number of steps equal to b. Furthermore, the number of steps grows linearly according to the pixel value resolution $(O(b))$. The main advantage of this algorithm is that it can directly compute weighted rank-order operations. This means that there is no need to reconstruct the local histogram according to the weights of the image pixel window. The classic running median algorithm [6] is quite difficult to be adapted in the case of weighted order statistics filtering. Fig. 1(a) demonstrates the running property of the latter algorithm. The numbers in the windows are the weights. In the case of Fig. 1(b), when the window moves from left to right, not only the leftmost and the rightmost columns should be deleted and added, respectively, but also the intermediate columns should be updated, so that the appropriate weights are taken into consideration. In general, the local histogram must be completely reconfigured, thus the advantage of the running property is not valid.

III. Experimental Results

The algorithm has been implemented in C (on a PC 486/66 with a commercial frame grabber) and applied to several images. Images are of 256 × 256-pixel spatial resolution and 8-b pixel value resolution. Experiments have been performed using 9 × 9, 7 × 7, 5 × 5, and 3 × 3-image data windows and the weight arrays shown in Fig. 2(a)–(d), respectivley. As a comparison, the weighted median quick sort algorithm described in [7] has been implemented and applied to the same images. Typical images considered here are Lenna and mandrill. The computational times for each image and algorithm have been measured and plotted in Fig. 3. As it can be seen with a 3 × 3-image data window both methods perform about the same, but with window sizes 5 × 5 and larger the proposed technique is a great improvement in speed.

IV. Conclusions

In this letter, a new algorithm suitable for the computation of weighted order statistics operations, including standard morphological and soft morphological operations, has been presented. This provides an output result in a fixed number of steps, which is linearly related to image pixel value resolution. It has been shown that for 5 × 5 and larger image data windows this algorithm outperforms the existing quick sort algorithm for weighted order statistics filtering.

References

[1] E. D. Dougherty and J. Astola, *Introduction to Nonlinear Image Processing*. Bellingham, WA: SPIE, 1994.

[2] M. Juhola, J. Katajainen, and T. Raita, "Comparison of algorithms for standard median filtering," *IEEE Trans. Signal Processing*, vol. 39, pp. 204–208, Jan. 1991.

[3] D. S. Richards, "VLSI median filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-38, pp. 145–153, Jan. 1990.

[4] J. P. Fitch, E. J. Coyle, and N. C. Gallagher, Jr., "Threshold decomposition of multidimensional ranked order operations," *IEEE Trans. Circuits Syst.*, vol. CAS-32, pp. 445–450, May 1985.

[5] J. P. Fitch, "Software and VLSI algorithm for generalized ranked order filtering," *IEEE Trans. Circuits Syst.*, vol. CAS-34, pp. 553–559, May 1987.

[6] T. S. Huang, G. J. Yang, and G. Y. Tang, "A fast two-dimensional median filtering algorithm," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-27, pp. 13–18, Feb. 1979.

[7] I. Pitas, *Digital Image Processing Algorithms*. Englewood Cliffs, NJ: Prentice-Hall, 1995.