Fast algorithms for computing image local statistics in windows of arbitrary shape and weights

Leonid Bilevich*[a] and Leonid Yaroslavsky**[a]
[a]Department of Physical Electronics, Faculty of Engineering,
Tel Aviv University, 69978, Tel Aviv, Israel[1]

ABSTRACT

Computing image local statistics is required in many image processing applications such as local adaptive image restoration, enhancement, segmentation, target location and tracking, to name a few. These computations must be carried out in sliding window of a certain shape and weights. Generally, it is a time consuming operation with per-pixel computational complexity of the order of the window size, which hampers real-time applications. For acceleration of computations, recursive computational algorithms are used. However, such algorithms are available only for windows of certain specific forms, such as rectangle and octagon, with uniform weights. We present a general framework of fast parallel and recursive computation of image local statistics in sliding window of almost arbitrary shape and weights with "per-pixel" computational complexity that is substantially of lower order than the window size. As an illustration of this framework, we describe methods for computing image local moments such as local mean and variance, image local histograms and local order statistics (in particular, minimum, maximum, median), image local ranks, image local DFT, DCT, DcST spectra in polygon-shaped windows as well as in windows with non-uniform weights, such as Sine lobe, Hann, Hamming and Blackman windows.

1. INTRODUCTION

Computing image local statistics, such as local means, variance, general local moments, local order statistics, ranks, histograms, spectra, etc., in sliding window is frequently required in image processing. Generally, for arbitrarily shaped window of $WinSz$ pixels, the "per-pixel" computational complexity of this process is $O(WinSz)$ or even, for spectra, $O(WinSz \log WinSz)$. Even for moderate window sizes, this complexity might be formidably large, especially in real-time processing applications. Substantial reduction of the computational complexity is possible with the use of recursive computation methods which utilize information common to consecutive overlapping windows and compute local statistics for the current window position by means of an appropriate modification of the results obtained for the previous window position.

The problem of fast computing image local statistics in sliding window has attracted much attention over all more than thirty years digital image processing. Quite well known are recursive algorithms for computing local statistics such as mean, histogram and median, order statistics, spectra, e.g. DFT, DCT, DcST in the windows of a rectangular shape. Recursive computing local mean in a rectangle window was described in Refs. [1,2]. Recursive algorithm for computing local median was introduced in Refs. [3,4]. In Ref. [5] recursive algorithm for computing local ranks was presented. Recursive computing of DCT in sliding window was shown in Refs. [6,7]. The recursive algorithm for local filtration was reported in Ref. [8].

However, in many cases rectangular windows are far from being optimal and windows of other shape are required. More recently, a number of recursive algorithms for computing image local statistics in the window of non-rectangular shape were suggested. Specifically, in Ref. [9] mean and variance in octagonal window, in Ref. [10] recursive algorithms for computing image signal moments in diamond, hexagon, general polygon windows, in Ref. [11] a method for recursive computing image local mean and in Ref. [12] histogram, median and order statistics in window of arbitrary shape with uniform weights were introduced.

---

[1] *bilevich@eng.tau.ac.il; http://www.eng.tau.ac.il/~bilevich/
**yaro@eng.tau.ac.il; http://www.eng.tau.ac.il/~yaro/

In this paper, we suggest unification and extension of these algorithms and present a general solution for recursive computing local statistics in windows of virtually arbitrary shape and weights based on the general method of parallel and recursive digital filtering introduced in Ref. [13].

2. IMAGE LOCAL STATISTICS

In order to simplify formulations, we will use in what follows 1-D denotations. We will define scanning window for measurements of the local statistics as a window of $2N_w+1$ pixels with weight coefficients $\{w_n\}$, $n=-N_w,\ldots,0,\ldots,N_w$; $\sum_{n=-N_w}^{N_w} w_n = 1$, and define local statistics of a signal $\{a_k\}$ within the window in its k-th position on the sampling grid as following:

$$LS_k = \sum_{n=-N_w}^{N_w} w_n S(a_{k-n}), \tag{1}$$

where $S(a_{k-n})$ is a statistics (a statistical attribute) for the signal sample a_{k-n}. According to this definition, computing local statistics is equivalent to digital filtering of signal $S(a_{k-n})$ with a digital filter with point-spread function defined by the window weight coefficients $\{w_n\}$.

Special cases of image local statistical parameters are local signal moments, histograms and spectra and their derivatives. For instance, in computing local moment $\mathbf{M}_k^P$ of order P of a signal $\{a_k\}$, $S(a_{k-n}) = (a_{k-n})^P$ and:

$$\mathbf{M}_k^P = \sum_{n=-N_w}^{N_w} w_n (a_{k-n})^P . \tag{2}$$

In computing the local weighted histogram, each pixel within the window contributes to the bin of the histogram, which corresponds to its gray levels, with weight defined by the pixel position in the window:

$$\mathbf{H}_k(q) = \sum_{n=-N_w}^{N_w} w_n \delta\{q - a_{k-n}\}, \tag{3}$$

where $\delta(.)$ is Kronecker delta-function ($\delta(0)=1$, $\delta(\neq 0)=0$). Local histograms and closely related local variational rows and local order statistics are used in rank filtering for signal/image denoising, smoothing, enhancement, extraction of object details and their boundaries. Customary, local histograms are computed for windows with uniform weights $\{w_n\}=1$, however non-uniform weights provide better flexibility to all histogram-based algorithms.

The local signal spectrum $\boldsymbol{\alpha}_r^{(k)}$ with respect to the basis $\{\psi_r(n)\}$ within the window in its k-th position on the sampling grid is defined as:

$$\boldsymbol{\alpha}_k^{\psi}(r) \equiv \sum_{n=-N_w}^{N_w} w_n a_{k-n} \psi_r(n). \tag{4}$$

A variety of types of local spectra exist. Most important in applications are spectra in Discrete Fourier and Discrete Cosine Transforms (DFT and DCT) computed over windows with uniform weight. Other examples are Walsh and Haar spectra. Applications of spectral analysis in scanning window include local adaptive signal/image restoration (denoising, deblurring, resampling, blind restoration, image enhancement), differentiating, integrating, target location and optical flow (Ref. [14]).

3. SCANNING MODES AND THE PRINCIPLE OF RECURSIVE COMPUTATIONS

Recursive computations assume a certain arrangement of image data in computer memory and a certain method of scanning image data. According to a common convention, we assume that images are defined on a rectangular sampling grid. On this grid, the following scanning modes are possible: *progressive row-wise—column-wise* scanning mode, *zig-zag row-wise—column-wise* mode and *diagonal-* 45° *—diagonal-* 135° mode (Figure 1).

In the *progressive row-wise—column-wise* scanning mode, all rows are scanned one after another from left to right, and pixels are accessed in the row-wise order.

In the *zig-zag row-wise—column-wise* mode, all rows are scanned in a one continuous scan: even rows are scanned from left to right and odd rows are scanned from right to left.

In the *zig-zag diagonal-* 45° *—diagonal-* 135° mode, all pixels are accessed in the diagonal- 45° order.

The first two scanning modes are suited for rectangular windows, while the third scanning mode is suited for non-rectangular windows.

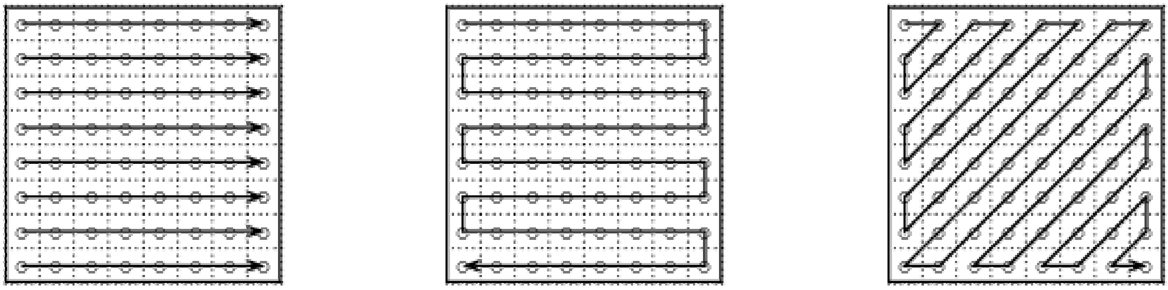

Figure 1. The progressive row-wise—column-wise scanning mode (left), the zig-zag row-wise—column-wise scanning mode (center) and the zig-zag diagonal- 45° —diagonal- 135° scanning mode (right).

In the process of image scanning with a window, the window position is associated with the position of the window central pixel and, in each window position, some pixels, which will be called incoming pixels, are entering the window and some, which will be called outcoming ones, are leaving the window; the rest of window pixels are remaining in the window. Incoming and outcoming pixels form what we will call "update structures". The principle of the recursive computation consists in performing computations not over all window pixels but only over "update structures" and using the computation results for updating the result for entire window obtained on the window previous position or several previous positions.

In Figure 2, examples of "update structures" for different window shapes are presented for row-wise image scanning mode. For instance, for a rectangular window and for the diamond window the "update structures" are pairs of window sides. The "update structures" lay on rows or columns of the rectangular sampling grid in the case of rectangular window and on diagonals of the rectangular grid in the case of diamond window. The "update structures" are mutually independent and therefore can be processed in parallel.

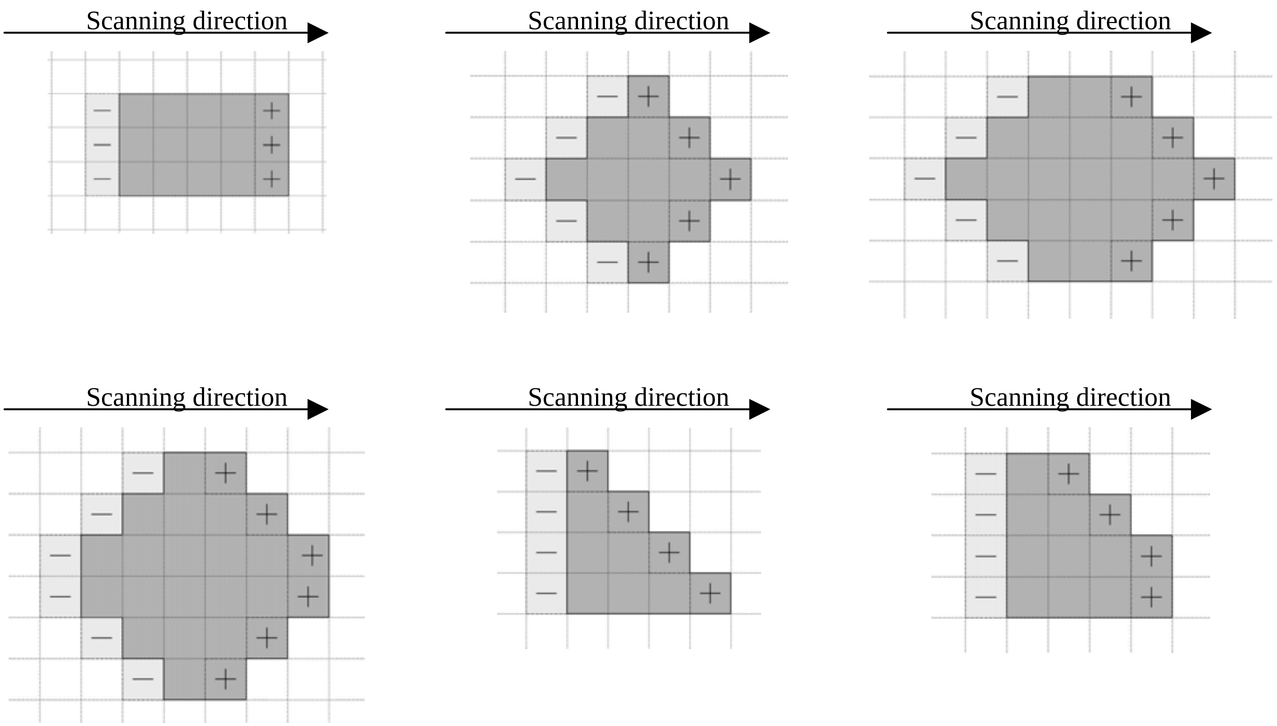

Figure 2. The “update structures” for windows of different geometrical shapes in the rectangular system of coordinates. First row: rectangle, diamond and hexagon. Second row: octagon, diamond sector and octagon sector.

4. PARALLELIZATION

As a general way to fast recursive computing of local statistics in windows of arbitrary shape and weights, parallelization of recursive computations can be used by means of splitting a computational task into several independent sub-tasks that can be performed in a recursive way, using, in each new window position, calculation results obtained in the previous window positions. This can be achieved by expansion of the window function over recursive bases.

Let window function $\{w_n\}$ can be approximated, with a desired accuracy, by expanding it into a series over a system of basis functions $\{\varphi_r(n), r = 0,1,2,\ldots,R-1;\ R \le 2N_w + 1\}$:

$$w_n \cong \sum_{r=0}^{R-1} \omega_r \varphi_r(n). \qquad (5)$$

The local statistics LS_k can then be approximated as

$$LS_k \cong \sum_{n=-N_w}^{N_w} S(a_{k-n}) \sum_{r=0}^{R-1} \omega_r \varphi_r(n) = \sum_{r=0}^{R-1} \omega_r \sum_{n=-N_w}^{N_w} \varphi_r(n) S(a_{k-n}) = \sum_{r=0}^{R-1} \omega_r \tilde{S}_r(k), \qquad (6)$$

where

$$\tilde{S}_r(k) = \sum_{n=-N_w}^{N_w} \varphi_r(n) S(a_{k-n}). \qquad (7)$$

Eq. (6) describes computing local statistics LS_k in R "***parallel filtering***" branches through sub-statistics $\tilde{S}_r(k)$ and summing up the sub-statistics with weights $\{\omega_r\}$.

Let us find conditions under which sub-statistics $\tilde{S}_r(k)$ can be computed recursively. In the simplest first order recursivity, computation result in k-th window position is computed using the result obtained in the preceding $k-1$ position. A link between $\{\tilde{S}_r(k)\}$ and $\{\tilde{S}_r(k-1)\}$ can be established as following:

$$\tilde{S}_r(k) = \sum_{n=-N_w}^{N_w} \varphi_r(n) S(a_{k-n}) = \sum_{n=-N_w}^{N_w} \varphi_r(n) S\left(a_{k-1-(n-1)}\right) = \sum_{m=-N_w-1}^{N_w-1} \varphi_r(m+1) S(a_{k-1-m}) =$$
$$\sum_{m=-N_w}^{N_w} \varphi_r(m+1) S(a_{k-1-m}) + \varphi_r(-N_w) S\left(a_{k+N_w}\right) - \varphi_r(N_w+1) S\left(a_{k-1-N_w}\right). \quad (8)$$

Choose basis functions $\varphi_r(m)$ such that

$$\varphi_r(m+1) = \varphi_r^{(0)} \varphi_r(m). \quad (9)$$

Then obtain:

$$\tilde{S}_r(k) = \varphi_r^{(0)} \sum_{m=-N_w}^{N_w} \varphi_r(m) S(a_{k-1-m}) + \varphi_r(-N_w) S\left(a_{k+N_w}\right) - \varphi_r^{(0)} \varphi_r(N_w) S\left(a_{k-1-N_w}\right) =$$
$$\varphi_r^{(0)} \tilde{S}_r(k-1) + \varphi_r(-N_w) S\left(a_{k+N_w}\right) - \varphi_r^{(0)} \varphi_r(N_w) S\left(a_{k-1-N_w}\right). \quad (10)$$

Eq. (10) signifies that for multiplicative basis functions $\{\varphi_r(n)\}$ that satisfy Eq. (9) sub-statistics $\tilde{S}_r(k)$ can be computed recursively with the recursivity on one preceding step.

Eq. (9) satisfied by the class of the recursive basis functions is that of exponential functions:

$$\varphi_r(m) = \left[\varphi_r^{(0)}\right]^m = \exp\left(i 2\pi p_0 \frac{rm}{M}\right); \quad M = 2N_w + 1, \quad (11)$$

where p_0 is a free parameter.

When the basis of Eq. (11) with $p_0 = 1$ is used for each of the R "***parallel filtering***" recursive filters the Eq. (7) performs signal discrete Fourier analysis in the window of $M = 2N_w + 1$ samples at the frequency r. The resultant spectral coefficients are multiplied element-wise by the window function spectral coefficients and summed up in order to obtain the required local statistics. The basis of Eq. (11) with $p_0 = 1/2$ corresponds to local DCT/DcST analysis. Note that local moments in Sine lobe window:

$$w_n = \sin\left[\frac{\pi(n+1/2)}{N_w}\right], \quad (12)$$

in Hann ("hanning", raised cosine, sine squared) window:

$$w_n = \frac{1}{2}\left\{1 - \cos\left[\frac{2\pi(n+1/2)}{N_w}\right]\right\}, \quad (13)$$

in Hamming window:

$$w_n = 0.54 - 0.46\cos\left[\frac{2\pi(n+1/2)}{N_w}\right] \quad (14)$$

and in Blackman window:

$$w_n = 0.42 - 0.5\cos\left[\frac{2\pi(n+1/2)}{N_w}\right] + 0.08\cos\left[\frac{4\pi(n+1/2)}{N_w}\right] \quad (15)$$

(where $n = 0,\ldots,N_w - 1$) are special cases of window DFT/DCT decomposition with the use of only corresponding first terms of the decomposition.

An important special case is the linearly independent basis of rectangular functions

$$\varphi_r(m) = \mathrm{rect}\left(\frac{m + N_w^r}{2N_w^r + 1}\right),\ N_w^r \le N_w, \quad (16)$$

which can be obtained from Eq. (11) when $p_0 = 0$. The latter corresponds to recursive computation of signal ***local mean***:

$$\bar{a}_k = \frac{1}{2N_w^r + 1}\sum_{n=-N_w^r}^{N_w^r} a_{k-n} = \bar{a}_{k-1} + \frac{a_{k+N_w^r} - a_{k-1-N_w^r}}{2N_w^r + 1}. \quad (17)$$

The basis of Eq. (16) leads to the multiple windows (composite window) method, when a given scanning window is decomposed onto several non-overlapping or overlapping sub-windows, or “building-blocks”, with uniform weights that allow recursive computations. The processing is performed in parallel on the sub-windows and then the results of sub-window computations are combined by addition with the decomposition coefficients. A simple example of composite window built from overlapping “building-blocks” is a combination of overlapping square and diamond (Figure 3). This window is an approximation of an octagon window, it is nearly isotropic and has “soft” edges. Other examples of composite windows are rings and combinations of sectors (Figure 3). The rings are obtained by subtraction of small window (usually of rectangular or octagonal shape) from the large window of the same type circumscribing the small one (usually two windows share a common center). The combinations of sectors are obtained by choosing the form of a sector window (usually of diamond sector or octagon sector shape) and combining several sectors of the same type and size but different orientation together to form parts of a diamond or an octagon.

In the section that follows we detail special cases of computing local statistics using decomposition of window into in multiple sub-windows.

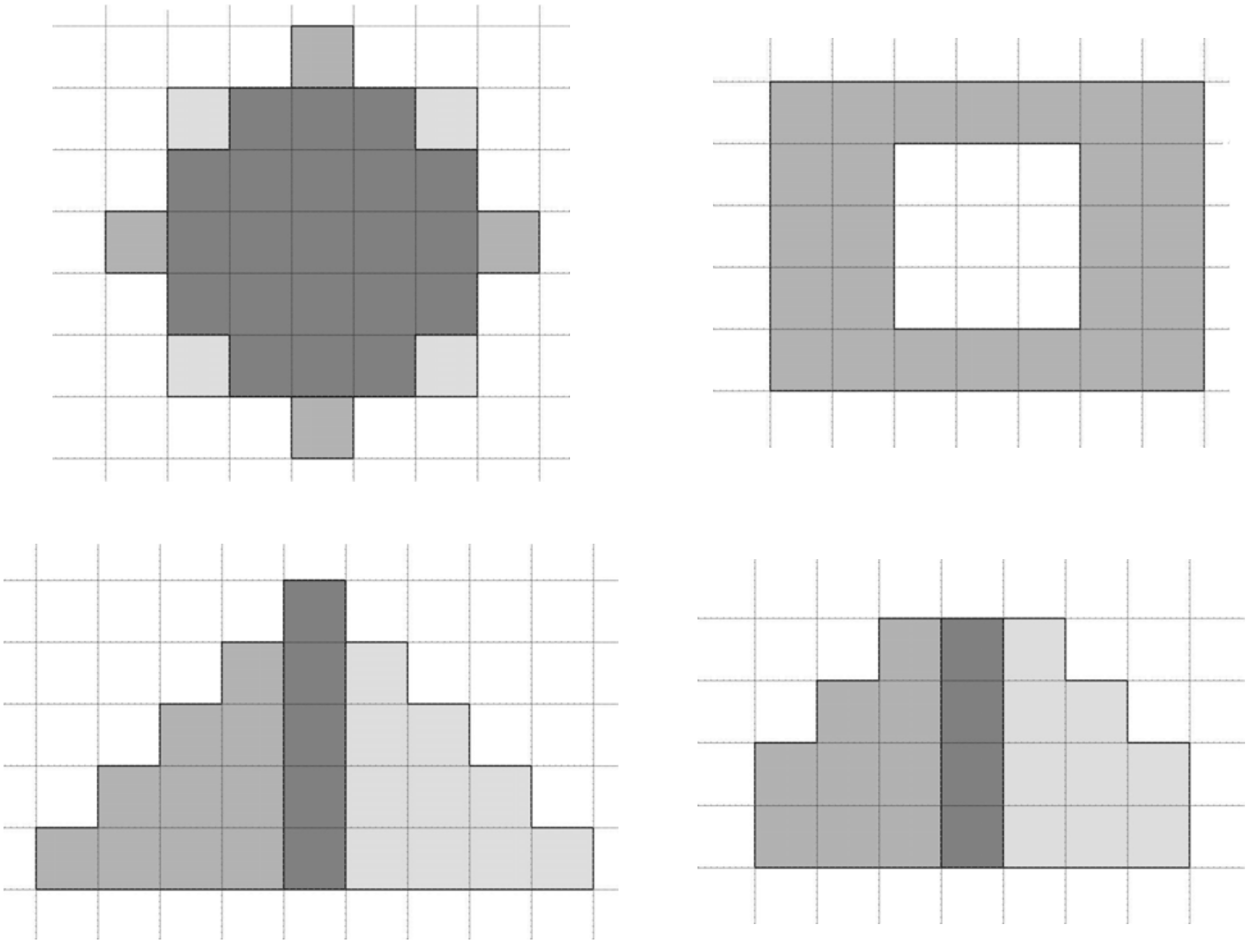

Figure 3. Examples of composite windows. First row: combination of square and diamond; rectangular ring. Second row: combination of diamond sectors; combination of octagon sectors.

5. PARALLEL AND RECURSIVE COMPUTATION OF LOCAL STATISTICS THROUGH COMBINATIONS OF MULTIPLE WINDOWS WITH BINARY WEIGHTS

In multiple window (composite window) method it is assumed that R "building-block" windows $\{W_r\}$ $(r = 0,\ldots,R-1)$ form an outlining window

$$\overline{W} \equiv \bigcup_r W_r \,, \tag{18}$$

and a constant weight w_r is assigned to r -th "building-block" (sub-window) W_r . Parallel and recursive computation of local moments by means of combining results of computations in multiple recursive windows is based on the following reasoning.

The P -th local moment over the sub-window W_r is given by:

$$\mathbf{M}_k^P(W_r) = \sum_{n \in W_r} [a_{k-n}]^P . \qquad (19)$$

It can be computed over the outline window $\overline{W}$, using the indicator function of the window W_r as:

$$\mathbf{M}_k^P(W_r) = \sum_{n \in \overline{W}} \delta(n \in W_r)[a_{k-n}]^P . \qquad (20)$$

The weighted sum of moments over sub-windows $\sum_{r=0}^{R-1} w_r \mathbf{M}_k^P(W_r)$ gives the local weighted P-th signal moment over the outline window $\overline{W}$:

$$\overline{\mathbf{M}}_k^P(\overline{W}) = \sum_{r=0}^{R-1} w_r \sum_{n \in \overline{W}} \delta(n \in W_r)[a_{k-n}]^P = \sum_{n \in \overline{W}} \left\{ \sum_{r=0}^{R-1} w_r \delta(n \in W_r) \right\} [a_{k-n}]^P = \sum_{n \in \overline{W}} \overline{w}_n [a_{k-n}]^P , \qquad (21)$$

with weights

$$\overline{w}_n = \sum_{r=0}^{R-1} w_r \delta(n \in W_r). \qquad (22)$$

Local weighted histogram over the outline window $\overline{W}$ can also be found as a weighted sum of local histograms over "building-block" sub-windows $\{W_r\}$ as following. The histogram over r-th window W_r is given by:

$$\mathbf{H}_k^{(W_r)}\{q\} \equiv \sum_{n \in W_r} \delta\{q - a_{k-n}\}. \qquad (23)$$

It can be computed from the histogram over the outline window $\overline{W}$, using the indicator function $\delta(n \in W_r)$ of the window W_r:

$$\mathbf{H}_k^{(W_r)}\{q\} = \sum_{n \in \overline{W}} \delta(n \in W_r) \delta\{q - a_{k-n}\}. \qquad (24)$$

The weighted sum of histograms over a combination of windows $\sum_{r=0}^{R-1} w_r \mathbf{H}_k^{(W_r)}\{q\}$ is the weighted histogram over the outline window $\overline{W}$:

$$\overline{\mathbf{H}}_k^{\overline{W}}\{q\} = \sum_{r=0}^{R-1} w_r \sum_{n \in \overline{W}} \delta(n \in W_r) \delta\{q - a_{k-n}\} = \sum_{n \in \overline{W}} \left\{ \sum_{r=0}^{R-1} w_r \delta(n \in W_r) \right\} \delta\{q - a_{k-n}\} = \sum_{n \in \overline{W}} \overline{w}_n \delta\{q - a_{k-n}\}, \qquad (25)$$

with weights of pixels in the window equal to the weighted sum of indicator functions over the windows:

$$\overline{w}_n = \sum_{r=0}^{R-1} w_r \delta(n \in W_r). \qquad (26)$$

Local weighted histograms can be used as bases for computing local weighted variational rows that are defined as cumulative sum of the weighted histogram, local ranks and local weighted order statistics, such as weighted median, and their derivatives, such as inter-quantile distances and alike. Some specific order statistics such as local minima and local maxima over composite windows can be found directly from recursively computed local minima and maxima over the window "building-blocks".

6. CONCLUSIONS

We briefly reviewed known methods of efficient recursive computation of image local statistics, such as local moments, histograms and order statistics, and local spectra in uniform windows of different geometrical shapes, and presented a general approach to recursive computation, in different ways of scanning image data, of local statistics in windows of virtually arbitrary shapes and weights. The approach exploits the idea of parallelization of computations by means of decomposition of given arbitrary window functions to a combination of either certain standard uniform windows, such as rectangular, diamond, octagon, diamond and octagon sectors, or of window functions that are basis functions of orthogonal transforms such as DFT, DCT that allow recursive computation. Different particular implementations of the approach to computing local image moments and their derivatives and local histograms and their derivatives are outlined. We believe that this opens new opportunities for real-time implementation of many image and video processing algorithms that are based on image local statistics.

REFERENCES

[1] Yaroslavsky, L. P., [*Digital Image Processing: Introduction*], Sov. Radio, Moscow, Russia (1979). English translation: Springer Verlag, Berlin, Germany, 1985.

[2] McDonnell, M. J., "Box-filtering techniques," *Comput. Graphics Image Process.* **17**(1), 65–70 (Sept. 1981).

[3] Huang, T. S., Yang, G. J., and Tang, G. Y., "A fast two-dimensional median filtering algorithm," *IEEE Trans. Acoust., Speech, Signal Processing* **27**(1), 13–18 (Feb. 1979).

[4] Garibotto, G. and Lambarelli, L., "Fast on-line implementation of two-dimensional median filtering," *Electron. Lett.* **15**(5), 24–25 (Mar. 1, 1979).

[5] Chaudhuri, B. B., "An efficient algorithm for running window pel gray level ranking in 2-D images," *Pattern Recognition Lett.* **11**(2), 77–80 (Feb. 1990).

[6] Yip, P. and Rao, K. R., "On the shift property of DCT's and DST's," *IEEE Trans. Acoust., Speech, Signal Processing* **35**(3), 404–406 (Mar. 1987).

[7] Xi, J. and Chicharo, J. F., "Computing running DCT's and DST's based on their second-order shift properties," *IEEE Trans. Circuits Syst. I* **47**(5), 779–783 (May 2000).

[8] Vitkus, R. Y. and Yaroslavsky, L. P., "Recursive algorithms for local adaptive linear filtration," in [*Mathematical Research, Computer Analysis of Images and Patterns*], Yaroslavsky, L. P., Rosenfeld, A., and Wilhelmi, W., eds., 34–39, Akademie Verlag, Berlin, Germany (1987). Band 40.

[9] Glasbey, C. A. and Jones, R., "Fast computation of moving average and related filters in octagonal windows," *Pattern Recognition Lett.* **18**(6), 555–565 (June 1997).

[10] Sun, C., "Moving average algorithms for diamond, hexagon, and general polygonal shaped window operations," *Pattern Recognition Lett.* **27**(6), 556–566 (Apr. 15, 2006).

[11] Ferrari, L. A. and Sklansky, J., "A fast recursive algorithm for binary-valued two-dimensional filters," *Comput. Vision Graphics Image Process.* **26**(3), 292–302 (June 1984).

[12] Van Droogenbroeck, M. and Talbot, H., "Fast computation of morphological operations with arbitrary structuring elements," *Pattern Recognition Lett.* **17**(14), 1451–1460 (Dec. 30, 1996).

[13] Yaroslavsky, L., "The possibility of parallel recursive organization of digital filters," *Telecommunications and Radio Engineering* **39**(5), 71–75 (May 1984).

[14] Yaroslavsky, L., [*Digital Holography and Digital Image Processing: Principles, Methods, Algorithms*], Kluwer Academic Publishers, Norwell, MA (2004).