A GENERIC 2D SHARPNESS ENHANCEMENT ALGORITHM FOR LUMINANCE SIGNALS

Egbert G.T. Jaspers and Peter H.N. de With

Philips Research Labs, Eindhoven, The Netherlands.

Abstract - *In this paper we attempt to generalize a sharpness enhancement technique for TV applications. Basically, the enhancement is accomplished by adding overshoot to luminance edges. However, the optimal amount of overshoot added for a high image quality depends on the local image statistics. For this purpose, four properties of the video signal are analysed locally by separate units and depending on this analysis, we regulate the amount of sharpness enhancement to be provided. Due to these additional controls, the system is robust with respect to varying image statistics and yields a high performance.*

Keywords: *television, image processing, enhancement, sharpness, peaking, luminance, signal processing.*

1. Introduction

In the consumer electronics area, the display technologies of television and computers are going to merge, due to the advent of new features such as internet, games and video conferencing. Since the spatial resolution of synthetic images of the PC outperforms that of natural video scenes, the desire for improved quality in TV images will increase. The subjective attribute sharpness, which is determined by the human visual system, is one of the most important factors in the perception of image quality. The conventional techniques derived from generic image processing applications to improve the image sharpness are not suitable for broadcasted TV material, because they do not consider transmission impairments, whereas the conventional algorithms from the TV world are not applicable for synthetic images. In this paper, we study a generic sharpness improvement that can be applied for TV images containing both natural scenes and graphically generated pictorial information.

We have confined ourselves to a model for sharpness improvement, in which an overshoot is added to the edges around objects in the image [1]. A literature study revealed that peaking, crispening, sharpness unmasking and statistical differencing are all techniques based on adding overshoot [1] [2] [3]. It has been shown experimentally that from the above-cited possibilities, peaking gives a high subjective sharpness improvement and it yields a simple solution from the signal processing point of view. In this paper we have extended this technique with an advanced adaptive control which uses the local image content, for combating various signal deteriorations such as transmission impairments and aliasing artifacts, thereby enabling the application in a much wider range of video signals. Spatially variant enhancement has already been subject of investigation for a.o. contrast enhancement [4] [5] [6].

In peaking, the added overshoot is determined by a 2-D high-pass filtering of the input signal, which can be formulated mathematically as:

$$G(j,k) = F(j,k) + \sum_{x=-w}^{w} \sum_{y=-w}^{w} a_{x,y} F(j+x, k+y), \qquad (1)$$

where $F(j, k)$ denotes the input signal and the weight factors $a_{x,y}$ represent the coefficients of the high-pass filter with order w. In this model, the filtered signal acts as an approximation of the second derivative of the original image. With the aforementioned model, a generic sharpness improvement algorithm can be found by locally controlling the amount of overshoot added, i.e. making the addition adaptive to the local picture contents and the transmission impairments measured in the same area. We have concentrated on the following properties for adaptive control of the sharpness enhancement:

- local intensity level and related noise visibility;
- noise level contained by the signal;
- local sharpness of the input signal;
- aliasing prevention, where alias results from non-linear processing such as clipping.

2. Local intensity level and related noise visibility.

The Human Visual System (HVS) is capable of perceiving a large range of intensities, but not simultaneously within an image [7]. Rather, the HVS handles this large variation by adapting itself to its overall sensitivity, a phenomenon known as brightness accommodation. The "simultaneously" perceived subjective brightness B is for a given brightness accommodation level B_b a continuous function of the intensity. B increases as a function of the intensity and is upper-bounded by peak-white and lower-bounded by pure black. The exact functional dependence of the intensity is unknown, but a mathematical approximation derived from experiments can be:

$$B = K_1 \cdot \{\mathrm{erf}(K_2(I - B_b)) + 1\}, \qquad (2)$$

with K_1 and K_2 constants and B_b the brightness accommodation level (see Figure 1). Let us now consider the influence of noise on the perceived brightness. If the intensity level of an object in the image exceeds the accommodation level, noise with a negative amplitude is perceived more visible than noise with a positive amplitude at that position. This phenomenon is shown in Figure 1. To perceive the same amount of "positive"

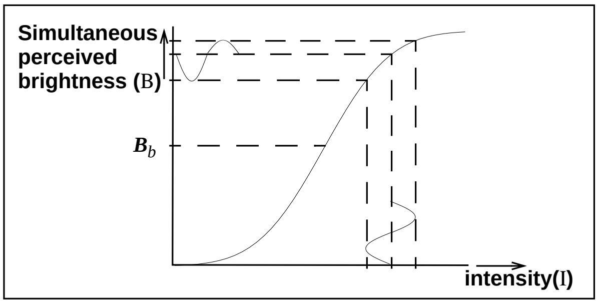

Figure 1. *Simultaneously perceived brightness as a function of the intensity.*

and "negative" noise, the noise with positive amplitude has to be larger than the noise with negative amplitude. Consequently, overshoot resulting from the 2-D high-pass peaking filter has to be larger than the undershoot from this filter for high-intensity regions in the image. For low-intensity regions in the image, an opposite statement applies. Since the accommodation level also depends on intensity contributions from other sources than the display (sun light, light spots, etc.), it is assumed to be half the intensity range of the video signal. By compensating the amount of overshoot in high- and low-brightness areas, the perceived amount of overshoot is equal for all intensity levels. Experimental evaluations show that the subjective noise sensitivity decreases when the output of the peaking filter is corrected with a suppression gain $k_1(j, k)$ according to:

$$k_1(j,k) = \begin{cases} \dfrac{F(j,k)}{256}, & H(j,k) > 0, \\ \dfrac{255 - F(j,k)}{256}, & H(j,k) \le 0, \end{cases} \tag{3}$$

where $F(j, k)$ is the input intensity level (0...255) and $H(j, k)$ stands for the output of the peaking filter.

3. Local sharpness of the input signal

Occasionally, generated overshoot can be rather large, which may result in conspicuously visible sharpness or even annoying impressions of the image. For steep luminance transitions, the output of the high-pass filter has generally a high amplitude. This results in a considerable amount of -undesirable- extra overshoot at already steep edges. To prevent this, the local steepness of luminance transitions are measured. The steepness measurement is based on an approximation of the maximum derivative of the signal in all spatial directions. The approximation is based on determining the dynamic range of the signal within a small surrounding of the desired position. The steepness can be described mathematically as:

$$D(j,k) = F_{max}(j,k) - F_{min}(j,k)\text{, with} \tag{4}$$

$$F_{max}(j,k) = \max\{F(j-n, k-m) \mid n, m \in \langle -1, 0, 1 \rangle\},$$

$$F_{min}(j,k) = \min\{F(j-n, k-m) \mid n, m \in \langle -1, 0, 1 \rangle\}.$$

The correction gain $k_2(j, k)$ to control the sharpness on sample basis depending on the local steepness, is evidently inversely proportional to the dynamic range $D(j, k)$ according to:

$$k_2(j,k) = \begin{cases} 0, & \hat{k}_2(j,k) < 0 \\ 1, & \hat{k}_2(j,k) \ge 1 \\ \hat{k}_2(j,k) = 1.25 - 0.01 \cdot D(j,k) & \end{cases} . \tag{5}$$

It can be seen that $k_2(j, k)$ suppresses the sharpness enhancement for large edges (dynamic ranges). Relation (5) was found after conducting numerous simulation experiments.

4. Noise contained by the signal (adaptive coring)

Sharpness enhancement amplifies the noise level and thus decreases the signal-to-noise ratio (SNR). Since the output of the enhancement filter is proportional to the local detail in the image, and the noise is equally distributed over the image, the SNR decreases in low-detail areas, so that noise in these image parts is mostly annoying. By suppressing the sharpness enhancement for low-detail areas, the subjective image quality can be improved. If the output of the peaking filter is large, it may be assumed that an edge is detected. This is true when the signal power exceeds that of the noise power, thus if the SNR is larger than 0 dB. If the filter output is small, noise can be a significant part of the signal contents. Summarizing, the SNR can be preserved by suppressing the sharpness enhancement at positions in the image where the information content of the video signal is small. A correction gain $k_3(j, k)$ at the output of the peaking filter results in the following output SNR.

$$SNR_o = \frac{\mathrm{E}\left\{[S(j,k) + k_3(j,k) \cdot P\{S(j,k)\}]^2\right\}}{\mathrm{E}\left\{[N(j,k) + k_3(j,k) \cdot P\{N(j,k)\}]^2\right\}}, \tag{6}$$

where $S(j, k)$ denotes the video signal, $N(j, k)$ stands for the noise signal and $P\{\}$ is the peaking filter operation. Regularly, the noise signal is equally distributed over the image for a given noise level. When the term $P\{S(j, k)\}$ is small and considering formula (6), the SNR can only be improved by decreasing $k_3(j, k)$. For other values of the filter output, $k_3(j, k)$ equals unity. The previously described behaviour of $k_3(j, k)$ can be formulated by:

$$k_3(j,k) = \begin{cases} 0, & \hat{k}_3(j,k) < 0 \\ 1, & \hat{k}_3(j,k) \ge 1 \\ \hat{k}_3(j,k) = -0.25 + 0.05 \cdot |H(j,k)| & \end{cases} , \tag{7}$$

where $H(j, k)$ denotes the output of the peaking filter. This function, also known as coring, has a small transition to let $k_3(j, k)$ increase gradually from 0 to 1. However, this solution has a disadvantage. The adaptation of

$k_3(j, k)$ in this model is independent of $P\{N(j, k)\}$. When this noise power is small, $k_3(j, k)$ does not have to depend on $H(j, k)$ as described in (7) to obtain a good SNR. A high SNR can also be obtained if $k_3(j, k)$ is decreased less for smaller values of $H(j, k)$. This problem is solved by giving $H(j, k)$ a penalty E depending on the noise level. Consequently, $\hat{k}_3(j, k)$ becomes:

$$\hat{k}_3(j, k) = -0.25 + 0.05 \cdot (|H(j, k)| - E), \qquad (8)$$

The noise level is determined using Equation (4). It was experimentally found that the number of times $D(j, k)$ does not exceed a certain threshold τ within one field or frame, is inversely proportional to the amount of noise. This property can be formulated mathematically as:

$$M = \sum_{j=1}^{L_l} \sum_{k=1}^{L_f} U[D(j, k) - \tau], \qquad (9)$$

where $U[]$ denotes the unit-step function, L_l the number of pixels per video line and L_f being the number of video lines per field or frame. Experimental results have shown that τ=10 is a satisfying threshold value. The measured value of M is translated to the penalty E according to:

$$E = 50 - \left(\frac{M \cdot 2^{10}}{L_l \cdot L_f}\right). \qquad (10)$$

Evidently, the value of M is normalized to the image size. The noise measurement as discussed above is not very accurate, but it proved to be sufficient for our application. When the image is noiseless (≥45 dB), the penalty decreases below zero and no coring will be applied. The suppression gain $k_3(j, k)$ is depicted in Figure 2. It can be seen that the coring function shifts to the left when the image is noiseless and it shifts to the right if more noise is measured in Equation (9).

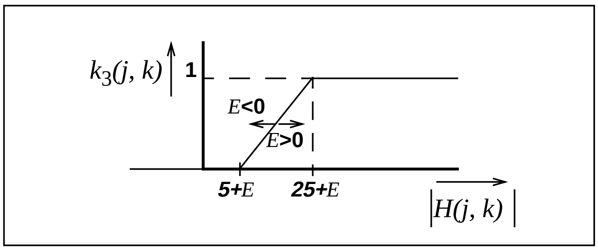

Figure 2. *$k_3(j,k)$ as function of the peaking filter output.*

5. Aliasing prevention from non-linear processing

Clipping of the signal for preventing pixel range excursions may cause artifacts. Clipping is a non-linear operation which creates additional cross-frequencies, which may be partially fold back into the signal spectrum, causing undesired aliasing components. The only way to solve this problem is to prevent the occurrence of clipping.

The clipping problem occurs mostly locally in a number of samples on or close to a signal transient. If, in accordance with the previous sections, a suppression factor $k_4(j, k)$ would be defined, the gain factor would show significant corrections for the clipped samples only and little for the surrounding samples. Consequently, the parameter $k_4(j, k)$ would portray a rather discontinuous behaviour at locations in the image where clipping occurs, thereby resulting in aliasing effects. Alternatively, if $k_4(j, k)$ would vary smoothly as a function of the sample position, the system would become locally linear and aliasing would not be visible. A smooth behaviour of $k_4(j, k)$ is guaranteed if it is considered and controlled on an area basis (e.g. rectangular blocks), instead of a sample basis. As a result, the image is divided into a raster of blocks, each of them being represented by a single $k_4(j, k)$ value. Hence, the "subsampled" suppression gain of $k_4(j, k)$ is defined as:

$$K_4(J, K) = k_4(nJ + 0.5n, mK + 0.5m), \qquad (11)$$

where n is the horizontal size of a block in pixels and m the vertical size of a block expressed in lines. Experimental results have shown that a block size of 32 x 32 is a good choice.

Let us now consider a workable definition of $K_4(J, K)$ and $k_4(j, k)$. It was found that the potential number of clippings that would occur in the actual block area is related to the smoothed version of $k_4(j, k)$ by the following reasoning. If $k_4(j, k)$ shows a strong local decrease in the correction gain in a particular area, then because of its smoothness, a significant number of samples will be affected. If the decrease is small, then the amount of samples affected is also small. Hence, the value of the correction is proportional to the number of clippings in that area. Since this relation was found to be more or less linear, $K_4(j, k)$ is defined as a value proportional to the number of potential clippings. The counted number of potential clippings, called $N_C(J, K)$, is determined by

$$N_C(J, K) = \sum_{j=32J-16}^{32J+16} \sum_{k=32K-16}^{32K+16} (U[-S_o + 0] + U[S_o - 255])$$

$$\text{with} \quad S_o = H(j, k) + F(j, k). \qquad (12)$$

Subsequently, $N_C(J, K)$ is converted to $K_4(J, K)$ with a linear transfer function. This function equals

$$K_4(J, K) = \begin{cases} 0, & \hat{K}_4(J, K) < 0 \\ 1, & \hat{K}_4(J, K) \geq 1 \\ \hat{K}_4(J, K) = 1.3 - \dfrac{N_C(J, K)}{170} & \end{cases}. \qquad (13)$$

The division of an image into a raster of blocks is depicted in Figure 3. To reduce the amount of line memories, $K_4(J, K)$ is determined, stored and provided in the next field or frame. Prior to carrying out the actual gain correction, $K_4(J, K)$ has to be upsampled to prevent blocking artifacts. Upsampling is performed by a biline-

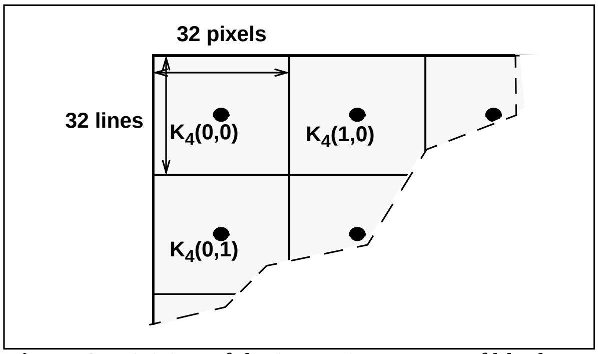

Figure 3. *Division of the image in a raster of blocks.*

ar interpolation technique. This technique can be accomplished with low cost by using three one-dimensional linear interpolations as shown in Figure 4.

Lets now consider the bilinear interpolation in more detail. The first vertical interpolation is done according to:

$$a_1 = K_4(J, K),$$
$$a_2 = K_4(J, K+1) - K_4(J, K), \text{ and} \quad (14)$$
$$k_4(nJ + 0.5n, mK + 0.5m + \Delta K) = a_1 + a_2 \cdot \Delta K.$$

Similarly, the second vertical interpolation equals

$$b_1 = K_4(J+1, K),$$
$$b_2 = K_4(J+1, K+1) - K_4(J+1, K), \text{ and} \quad (15)$$
$$k_4(nJ + 1.5n, mK + 0.5m + \Delta K) = b_1 + b_2 \cdot \Delta K.$$

Finally, the results of Equation (14) and 15 are used for a horizontal interpolation, leading to

$$c_1 = k_4(nJ + 0.5n, mK + 0.5m + \Delta K),$$
$$c_2 = k_4(nJ + 1.5n, mK + 0.5m + \Delta K)$$
$$- \; k_4(nJ + 0.5n, mK + 0.5m + \Delta K) \text{ and} \quad (16)$$
$$k_4(j, k) = k_4(nJ + 1.5n + \Delta J, mk + 0.5m + \Delta K)$$
$$= c_1 + c_2 \cdot \Delta L.$$

The value of $k_4(j, k)$ is calculated sequentially by first letting j run from 1 to L_l and subsequently letting k run from 1 to L_f. Due to the sequential index augmentation, the calculation can be made recursive. Rewriting Equation (16) as a function of the $K_4(J, K)$ and writing $k_4(j, k)$ as a function of $k_4(j\text{-}1, k)$ leads to:

$$k_4(0, 0) = K_4(0, 0)$$
$$k_4(j, k) = k_4(j-1, k) + c_2 \quad (17)$$
$$= k_4(j-1, k) + b_1 - a_1 + (b_2 - a_2) \cdot \Delta K$$

For each new block, the terms a_1, a_2, b_1 and b_2 are recomputed. It can be noticed that one multiplication and a few additions are sufficient to perform each iteration. Vertical recursion is also possible at the cost of a video line memory, in order to restore $k_4(0, k\text{-}1)$ till $k_4(L_l, k\text{-}1)$.

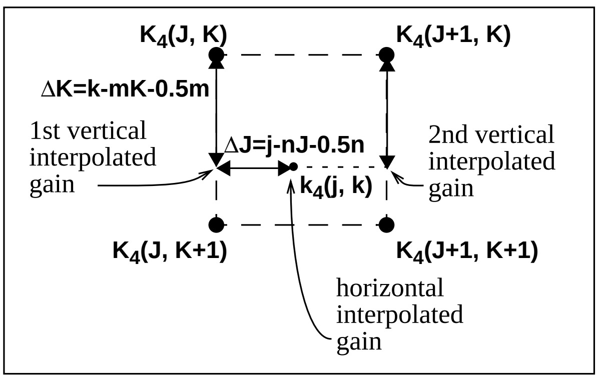

Figure 4. *Bilinear interpolation of $K_4(J, K)$ values.*

Experimental results showed that the problems due to clipping were still not solved completely. Aliasing was not only produced by the two spatial dimensions, but also by the temporal dimension. Apparently, $k_4(j, k)$ also contains discontinuities in the temporal domain. The aliasing produced by the time component is perceived as field flicker. This problem was solved by filtering $K_4(J, K)$ with a two-tap recursive filter as a function of time. As was mentioned before, $K_4(J, K)$ is determined, stored and subsequently provided in the next field or frame. The temporal filter updates the stored $K_4(J, K)$ instead of replacing it (see Figure 5).

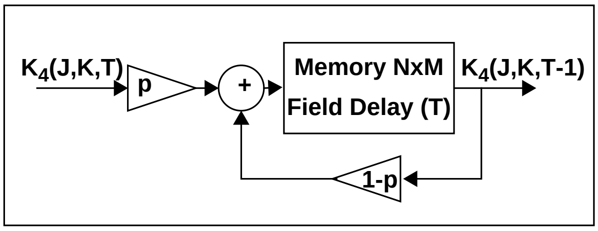

Figure 5. *Temporal filtering of the gain factor to suppress temporial aliasing.*

The block size influences both the image quality and the implementation as follows.

- Spatial aliasing: too small blocks lead to a inconsistent $k_4(j, k)$ and too large blocks lead to loss of local adaptivity.
- Required memory resources to restore $K_4(J, K)$.
- Errors in the temporal moment of $k_4(j, k)$: due to the temporal filtering and the motion in the images, small blocks will lead to larger position errors in $k_4(j, k)$.

Although sharpness enhancement will locally decrease in the areas where much clipping would occur, aliasing is perceived as more annoying than the local sharpness loss.

6. The system including all controls

In the overall sharpness enhancement model, the overshoot at edges in the peaking processing is suppressed by a special control parameter $k_o(j, k)$, which is determined by all aforementioned artifacts and considerations. The overall block diagram of the algorithm is depicted in Figure 6. In the complete system, the formulation of the peaking becomes as follows:

$$G(j, k) = F(j, k) + k_o(j, k) \cdot H(j, k)\text{, with} \qquad (18)$$

$$H(j, k) = \sum_{x=-w}^{w} \sum_{y=-w}^{w} a_{x, y} \cdot F(j + x, k + y)\ , \qquad (19)$$

where $k_o(j, k)$ is bounded between $0 \le k_o(j, k) \le 1$. When one of the individual contributions $k_i(j, k)$ for i = 1...4, portrays a large occurence of the corresponding artifact, the correction of the added overshoot will be large too, leading to a small $k_o(j, k)$. In this model, the smallest correction factor represents the most annoying artifact. When the gain of the smallest artifact measure is applied to suppress the total enhancement, the remaining artifacts will be suppressed as well. We have found that this decision criterion yields a good performance, although it is a highly non-linear operation.

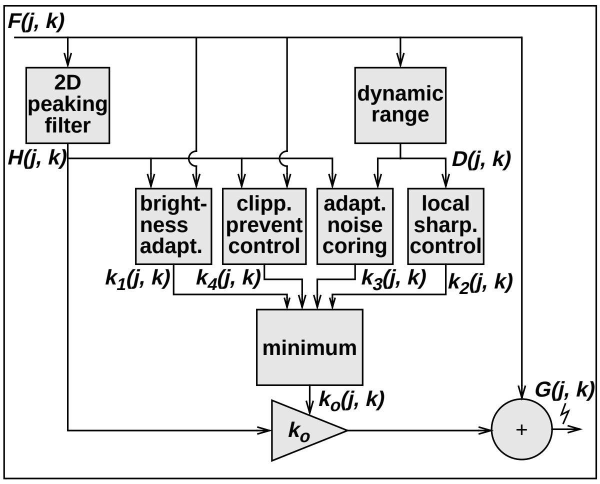

Figure 6. *Block diagram of the generic 2D peaking.*

7. Results and conclusions

We have presented a sharpness enhancement technique for TV applications based on adding overshoot to luminance transitions in the image. A key feature of the system is that the amount of overshoot added depends on four correction gain parameters. These parameters result from analysing different properties of the local image signal, i.e. intensity level, noise level, local sharpness of the input and potential clipping artifacts. The concept enables a stable sharpness improvement for a large variety of TV scenes.

The algorithm has been evaluated by computer simulations and the results of the adaptive peaking together with its special artifact control outperforms clearly the existing algorithms implemented in current TV sets. This is explained by the application of a real 2-D filter kernel and by the must more advanced image analysis to control added enhancement. The algorithm indeed proves to be robust with respect to varying image statistics, while the performance is very good. Even for high-quality video signals, a remarkable improvement is obtained (see Figure 7 in which the original is digital YUV). The appropriate suppression of overshoot in textured areas with large signal amplitudes or areas containing significant noise energy, is working quite effectively an subjectively improves the image quality.

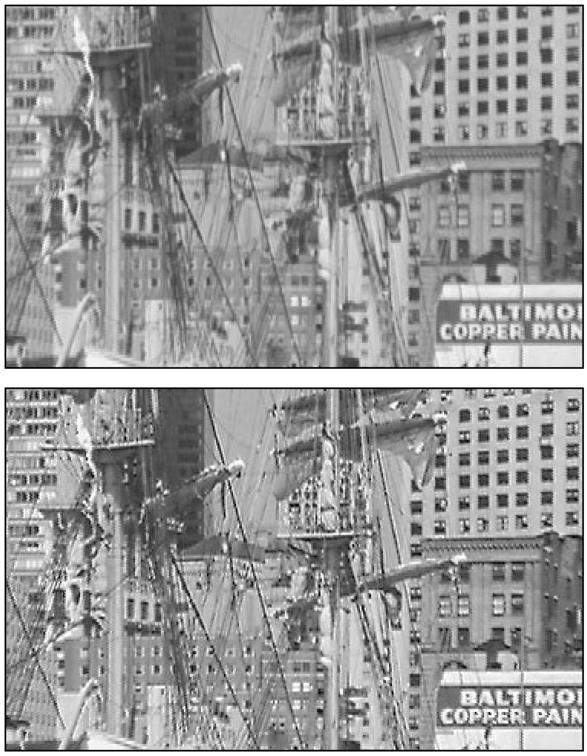

Figure 7. *The original 'Baltimore' image (upper picture) and the sharpness enhanced version of this image (lower picture).*

8. References

[1] A. Rosenfeld and A.C. Kak, *Digital Picture Processing*, 2nd. ed., Vol. 1, New York: Academic Press, 1982, pp. 237-250.

[2] W.K. Pratt, *Digital Image Processing*, 2nd. ed., John Wiley & Sons, Inc., New York, 1991.

[3] R.H. Wallis, " An Approach for the Variant Restoration", Proc. Symp. Current Math. Problems in Image Science, Monterey, USA (CA), Nov. 1976.

[4] J.D. Fahnestock and R.A. Schowengerdt, "Spatially variant contrast enhancement using local range modification", Optical Eng., Vol. 22, No. 3, pp. 378-381, 1983.

[5] Chen and V.-H. Tsai, "Moment-Preserving Sharpening", Int. Journ. Comp. Vis., Graphics, and Image Proc., Academ. Press, Vol. 41, pp. 1-13, 1988.

[6] T.-L. Ji, M.K. Sundareshan, and H. Roehrig, "Adaptive Image Contrast Enhancement Based on Human Visual Properties", IEEE Trans. Medical Imaging, Vol. 13, No. 4, pp. 573-586, Dec. 1994.

[7] R.C. Gonzalez and R.E. Woods, *Digital Image Processing*, Addison-Weslay Publishing Company Inc., June 1992, Chapter 2.