Interpolation of Non-Linear Retinex Type Algorithms

Doron Shaked
HP Laboratories Israel
HPL-2006-179
December 4, 2006*

image processing, retinex-type algorithms, image interpolation

In this paper we propose a method to speed up Retinex-type algorithms, consisting of a computationally intensive non-linear illumination estimation module followed by a relatively simple manipulation module. Speed up is obtained by way of computing the illumination on a sub-sampled image. The challenge is to interpolate a piece-wise smooth low resolution image. We present and analyze the trade off between two types of interpolation methods. On one hand, regular illumination interpolation, which preserves the Retinex-type output quality, however may result in artifacts. On the other hand a detail preserving interpolation which removes artifacts, however may compromise the output quality.

* Internal Accession Date Only
Published in the proceedings of Electronic Imaging 2007, 28 January – 1 February 2007, San Jose, CA, USA
Approved for External Publication

Interpolation for non-linear Retinex-type algorithms

Doron Shaked

ABSTRACT

In this paper we propose a method to speed up Retinex-type algorithms, consisting of a computationally intensive non-linear illumination estimation module followed by a relatively simple manipulation module. Speed up is obtained by way of computing the illumination on a sub-sampled image. The challenge is to interpolate a piece-wise smooth low resolution image. We present and analyze the trade off between two types of interpolation methods. On one hand, regular illumination interpolation, which preserves the Retinex-type output quality, however may result in artifacts. On the other hand a detail preserving interpolation which removes artifacts, however may compromise the output quality.

Keywords: Image Enhancement, Retinex-type Algorithms, Image Interpolation Schemes

1. INTRODUCTION

Many traditional and state-of-art algorithms for both dynamic-range compression and illumination correction are what we call ‘Retinex type’ algorithms. Examples of Retinex-type algorithms appear the following references [1-23]. They are characterized by a two-module structure depicted in Figure 1. An ‘illuminant’ estimation module computes a smooth version of the input image. Usually, it is an either linear or non-linear low-pass filter of the input image. An ‘illumination’ manipulation module usually subtracts part of the smooth version from the input image.

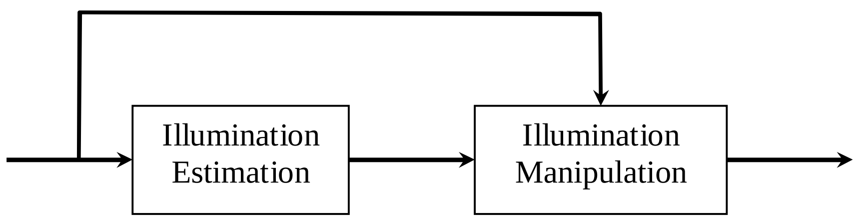

Figure 1: A block diagram of a generic Retinex type algorithm.

There are many variants on that theme:

- Some algorithms do not relate themselves to the Retinex theory, and will not refer to Retinex or illumination (e.g. [4,20]). Others refer to Retinex theory, and to a ‘mask’ (which is usually the inverse of the illumination). Nevertheless, if they are one of the following (or similar) variants of Figure 1, we consider them as Retinex type algorithms.
- Retinex type algorithms are usually performed in the Log domain.
- Classic algorithms use linear space invariant low pass filters [12,13] or partial differential equations [1,7,23] for the first module. Variants include:
  - Slowly varying envelopes, i.e. local envelops instead of local averages [5,6,9,10,15].
  - Non-linear low passes resulting in piece-wise smooth averages or envelopes, which might change abruptly whenever the input changes abruptly [2,3,11,17,18,19,22].
- In the second module, the illumination might be subtracted in part [9,22], e.g. subtract half of the illumination from the input image. Alternative manipulation methods may be used, which reduce more or less of the illumination image as a function of the corresponding illumination value.
- The algorithm may be applied to monochrome or color images. In the later case it might be applied to all planes, or only to the illumination (e.g. Y) plane (see e.g. [10]).
- In some algorithms both modules are performed in an iterative filtering and subtraction scheme [5,6,12,15,19]. In others they are interleaved in a scale-space [2,8,10,11,22,23].
- Recently Retinex type algorithms are being applied to gamut mapping [11,15].

One of the main drawbacks of Retinex type algorithms is the high computational complexity they require. The main complexity is the low-pass module, especially since for good performance those algorithms should preferably use large support filters. Several computationally efficient methods have been suggested for linear and non-linear versions of the illumination estimation module. However, most of them have a computationally intensive part that is, at best, linear in the input complexity. Namely, the computationally intensive loop of most of the algorithms will work at least once for every input pixel.

The only exceptions to the above are algorithms whose computationally intensive operation produces a smooth function (either an average or envelope). For those algorithms there have been (a surprisingly small number of) implementations [3,9,16], which rely on the fact that smooth images can be computed on a sparse grid and interpolated to the fine grid, as depicted in Figure 2. Thus, those algorithms reduce the complexity of the computationally intensive part to sub-linear, with an additional linear-complexity down sampling and interpolation (relatively fast algorithms).

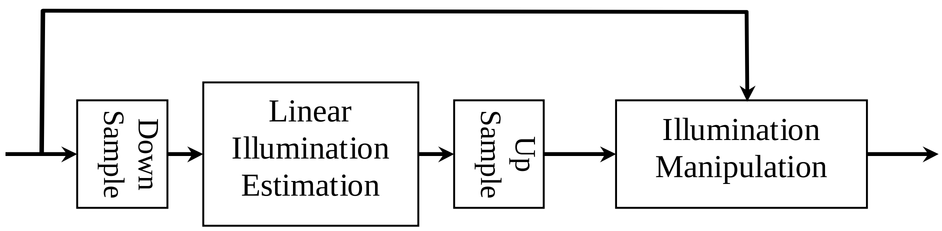

Figure 2: A block diagram of down sampled Retinex-type algorithms.

State of art Retinex type algorithms use non-linear illumination estimation engines rather than linear convolutions or smooth envelope engines. Non-linear illumination engines produce piecewise smooth illumination images, which pose a problem for the down sampled scheme, since they do not interpolate well.

Kiesel et al. [9] improve images based on an illumination image (or improvement field – as they call it) obtained from a sub-sampled version of the input image. A coarse description of their algorithm corresponds to Figure 2. The improvement field they use is a smooth envelope illumination obtained via [5]. They did not relate to non-linear or piece-wise smooth improvement fields, or to problems their method might incur if it were to be applied for such fields. Non-linear illumination correction algorithms we are ware of [2,11,18,19,22] do not report any sort of sub-sampling. Durand and Dorsey [3] are an exception to the above.

Durand and Dorsey [3] perform sub-sampling and interpolation for speed-up in a non-linear Retinex-type algorithm. However, their algorithm uses the peculiarities of the particular non linear algorithm (Bilateral Filtering) they implemented, to avoid the need to interpolate a low-resolution piece-wise smooth illumination image. Instead, they interpolate a set of smooth low-resolution intermediate images, and use the high-resolution input to select corresponding output pixels from the resulting set of high-resolution images. In addition, the method in [3] pays dearly for the benefit of sub-sampling, namely, the computationally intensive (sub-sampled) part is repeated multiple times, once for each of the intermediate images.

In this paper we analyze the sub-sampling approach in the non-linear settings. We suggest criteria and methods that avoid artifacts when non-linear illumination estimation images are interpolated, thereby enabling a significant speed-up of Retinex type algorithms.

2. SUB-SAMPLED NON-LINEAR ILLUMINATION ESTIMATION

The block diagram we relate to appears in Figure 3. In the following, we detail the possible ways one may implement the down sampling and up sampling blocks, and the way the preferred implementation depends on the type of non-linearity in the illumination estimation module, and the sampling rate.

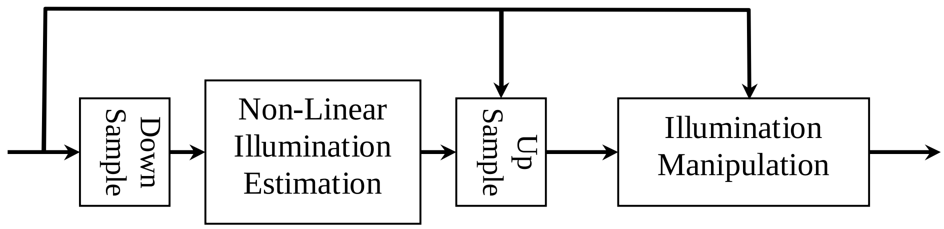

Figure 3: A block diagram for a non-linear sub-sampled Retinex-type

2.1 Down Sampling

Down sampling is a standard algorithm in image and signal processing, where one samples a filtered version of the high-resolution image. In the non-linear setting, the filter should be the appropriate non-linear filter with a support corresponding to the required sub-sampling. If for example, the Retinex-type non-linear illumination estimation averages a large support neighborhood with weights corresponding to neighbor's differences from the center pixel, the same weighting scheme should be applied in the down sampling filter (usually, a much smaller support filter). In case the illumination is an envelope rather than a local average, down sampling may be a sampling of a local maximum of the high-resolution image. Alternatively, for large sampling factors one may choose to take a combination. For example sample by a factor of s1 using averaging (local-maximum) followed by a sub-sampling with a factor of s2 using local-maximum (averaging), for an overall sampling factor of S=s1*s2.
Using local maximum sampling might increase hallo artifacts as discussed in the next section.

2.2 Up Sampling

Up sampling is a standard algorithm in image and signal processing, where new samples are interpolated between low-resolution input samples. Standard methods include neatest neighbor, bi-linear, and bi-cubic interpolations. Our case is a little different, where low resolution images contain edges, and those have to be interpolates such that edges in the high-resolution output do not mismatch edges in the original image (inaccurately placed details will result in artifacts). The solution is to either make sure the interpolation does not add details, or else use the input image as a guide to detail placement.
Figure 4 details the two options. In the illumination interpolation option of Figure 4a, high-resolution illumination values are interpolated from low-resolution values using e.g. bilinear interpolation (any other interpolation method that does not try to estimate or create new details will do). In case the estimated illumination is an envelope rather than an average, then due to rare cases in which the interpolated illumination will be below the input, the envelope constraint should be enforced explicitly as a maximum operation.

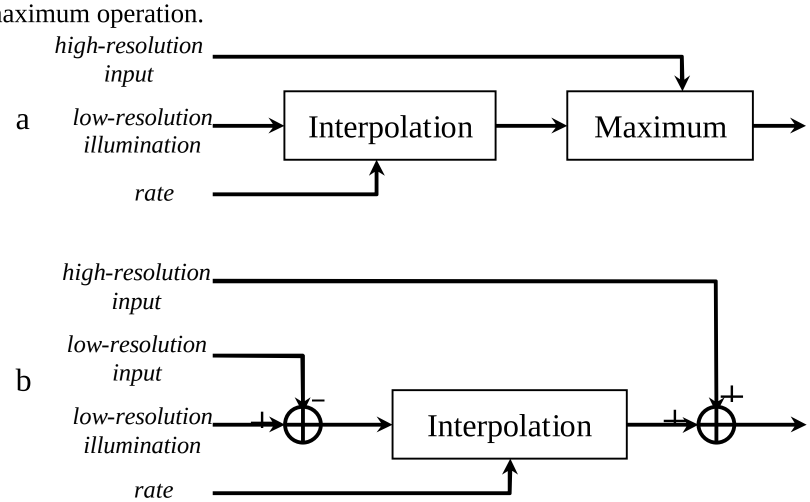

Figure 4: Block diagrams of two alternative up-sampling methods for Figure 3: a) Illumination interpolation, and b) Difference interpolation.

In the difference interpolation option of Figure 4b, a high-resolution difference (between the illumination and the input) is interpolated from low-resolution values (same interpolation methods as above). The difference is consequently added to the high-resolution input.

Figure 5 is a schematic illustration of signals through the two interpolation systems. High-resolution input, and low-resolution envelope are depicted as black curves. Low-resolution samples are depicted as dots, and interpolated to form the piece-wise constant black curve. Linear interpolation, which might result from the algorithm described in Figure 4a is depicted in Figure 5a. The light blue curve is the interpolation, and the dark blue curve is the result after clipping. Note that curves, which are in practice overlapping, have been slightly displaced for clarity. Difference interpolation, which might result from the algorithm described in Figure 4b is depicted as an orange curve in Figure 5b.

Note that the cyan interpolation curve of Figure 5a has no new details, whereas the orange interpolated curve of Figure 5b has many of the input's details (in good alignment with the input).

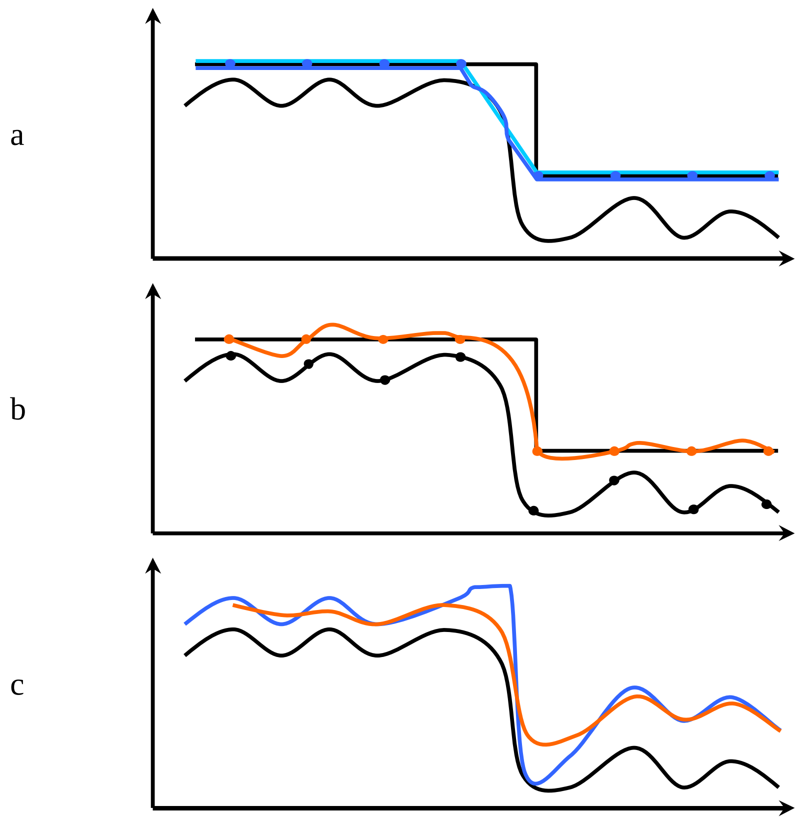

Figure 5: Schematic illustration of: (a) Illumination interpolation, (b) Difference interpolation, and (c) corresponding effect on the Retinex outcome

Consider now the illumination manipulation module of Figures 1-3. All version of this module are similar to a difference between the input and the illumination. Thus, outputs of Retinex-type algorithms using each of the above interpolation methods will have the inverse properties with respect to their interpolated illumination. This is depicted schematically in Figure 5c, where the blue curve illustrating the output of an illumination interpolation is considerably sharper than the orange curve illustrating the output of the difference interpolation.

A concrete example of a sub-sampled non-linear Retinex based on [18] is given in Figure 6. Figure 6a, presents an input image, and a zoomed-in part of it. Figure 6b presents parts of a 1:5 illumination-interpolation and corresponding Retinex correction. Figure 6c presents parts of a 1:5 difference interpolation and corresponding Retinex correction. Note that the sharper Retinex output corresponds to the blurred interpolated illumination and vise versa.

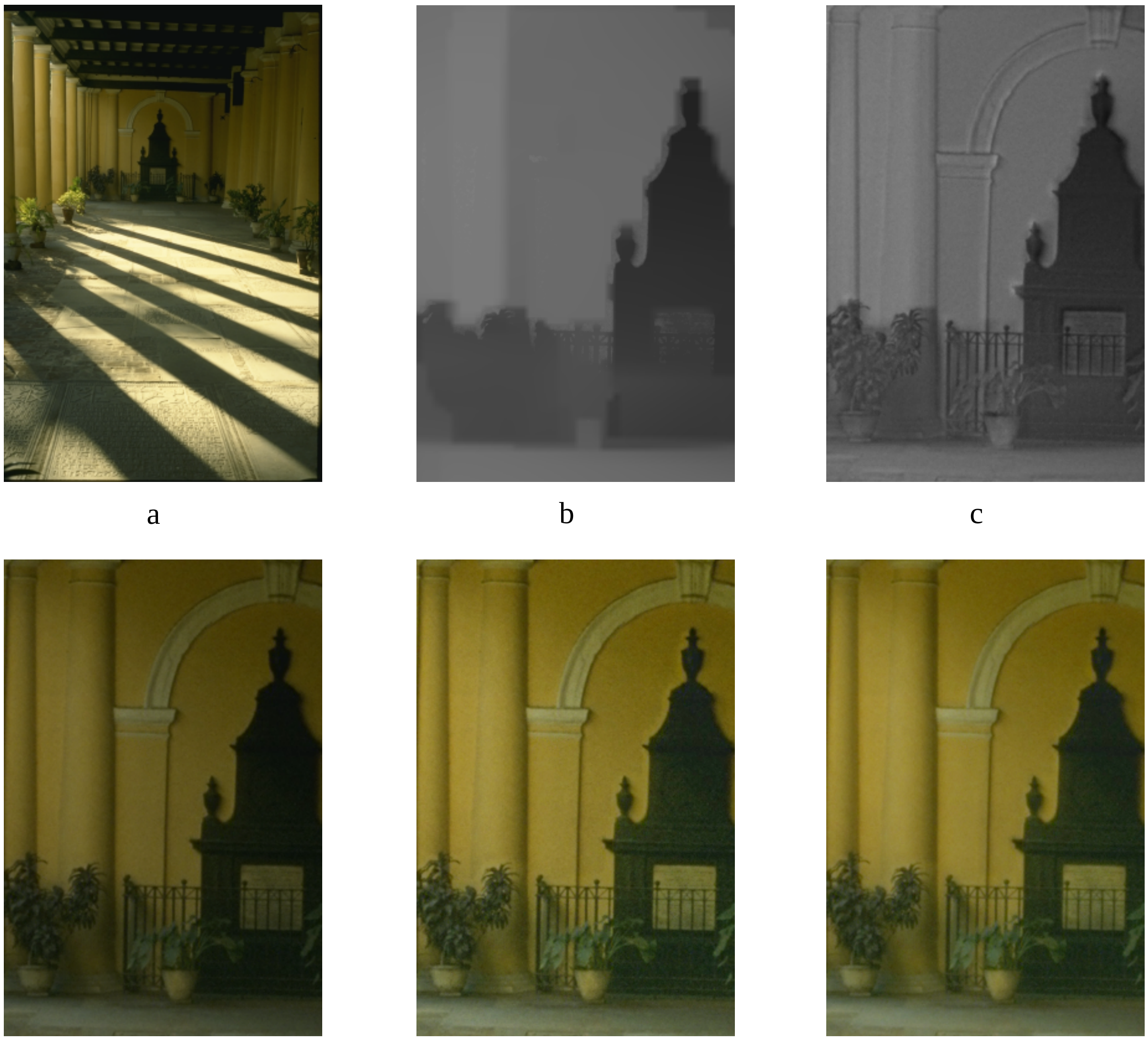

Figure 6: (a) Input image {original and part}, (b) Illumination interpolation {illumination and Retinex correction}, (c) Difference interpolation {illumination and Retinex correction}.

2.3 Up Sampling Artifact Balance

Choice of the interpolation depends on imaging intent and required interpolation rate. While usually sharp images are desirable, in some applications over sharpening is a problem. Furthermore, for large sampling rate misplaced details become halos rather than sharpening effects. In all these cases one should balance the blurring artifacts of difference interpolation against the sharpening / halo artifacts of illumination interpolation. Alternatives include:

- Average the high-res illumination images from illumination and difference interpolations.
- Cascade illumination and difference interpolations, such that the total interpolation rate is the required interpolation.

These alternatives are depicted in Figure 7. Note that for the two alternatives to perform comparable up scaling we need $R0 = RI * RD$. Also, since $WI + WD = 1$, the weights Wi play a similar role to rate ratio $Ri/R0$ in the corresponding schemes.

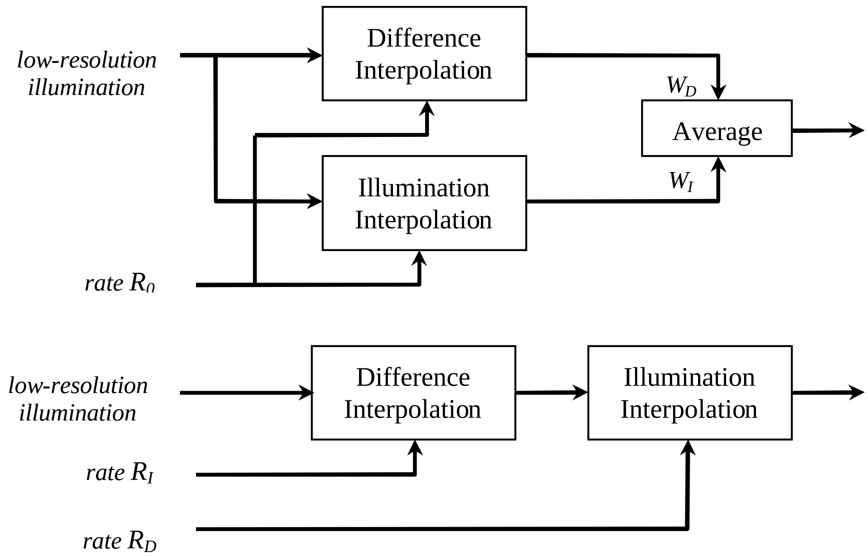

Figure 7: Block diagrams of two alternative hybrid up-sampling methods.

Figure 8 depicts the result of the first hybrid alternative with WI = WD = 0.5, compared against each of the basic interpolation methods.

a b c

Figure 8: Retinex correction for (a) Illumination interpolation, (b) Difference interpolation, and (c) equal weight average of the above.

A different way to combine the two interpolation methods may be motivated by the fact that different artifacts are significant in different locations in the image. For example, halos are more visible near strong edges for which one side is in the mid-tones (high and low tones have less distinctive halos). This is evident in Figure 6b, where the halos on the dark altarpiece are less noticeable than those on the beige wall. Also at texture regions, halos are much less visible, as e.g. in the case of the textured dark fence against the wall versus the edge between the altar-piece and the wall. In such cases it might be preferable to modify the rates Ri or the weights Wi locally, according to image properties.

REFERENCES

[1] Blake, “Boundary Conditions of Lightness Computation in Mondrian World”, Computer Vision Graphics and Image Processing, Vol. 32, pp. 314-327, 1985.

[2] J. M. DiCarlo, and B. A. Wandell, “Rendering High Dynamic Range Images”, in Proc. SPIE Vol. 3965, pp. 392-401, 2000.

[3] F. Durand, and J. Dorsey, “Fast Bilateral Filtering for the Display of High Dynamic Range Images”, in proc. SIGGRAPH02, Los Angeles, 2002.

[4] O. D. Faugeras, “Digital Image Color Processing Within the Framework of a Human Visual System”, IEEE Trans. on ASSP, Vol. 27, pp. 380-393, 1979.

[5] J. Frankle, and J. McCann, “Method and Apparatus for Lightness Imaging”, US Patent # 4,384,336, 1983.

[6] Funt, F. Ciurea, and J. McCann, “Retinex in Matlab”, in Proc. of IS&T 8th Color Imaging Conference, pp. 112-121, 2000.

[7] K. P. Horn, “Determining Lightness from an Image”, Computer Graphics and Image Processing, Vol, 3. pp. 277-299, 1974.

[8] J. Jobson, Z. Rahman, and G. A. Woodell, “A Multiscale Retinex for Bridging the Gap Between Color Images and the Human Observation of Scenes”, IEEE Trans. on Image Proc., Vol. 6, 1997.

[9] K. C. Kiesel, and W. R. Wray, “Reconstruction of Images”, US Patent # 4,649,568, 1897.

[10] R. Kimmel, M. Elad, D. Shaked, R. Keshet, and I. Sobel, “A Variational Framework for Retinex”, Int. Journal of Computer Vision, Vol. 52, pp. 7-23, 2003.

[11] R. Kimmel, D. Shaked, M. Elad, and I. Sobel, “Space Sensitive Color Gamut Mapping: A Variational Approach”, IEEE Trans. Image Processing, Vol. 14, pp. 796-803, 2005.

[12] H. Land, “Recent Advances in the Retinex Theory and Some Implications for Cortical Computations: Color Vision and the Natural Image”, in Proc. National Academy of Science USA, Vol. 80, pp. 5163-5169, 1983.

[13] H. Land, “An Alternative Technique for the Computation of the Designator in the Retinex Theory of Color Vision”, Proc. National Academy of Science USA, Vol. 83, pp. 3078-3080, 1986.

[14] E. H. Land, and J. J. McCann, “Lightness and the Retinex Theory”, J. Optical Soc. of America A, Vol. 61, pp. 1-11, 1971.

[15] J. McCann, “Lessons Learned from Mondrians Applied to Real Images and Color Gamuts”, in Proc. IS&T/SID 7th Color Imaging Conference, pp. 1-8, 1999.

[16] J. McCann, “Special Session: Retinex at 40”, in Proc. of SPIE Vol. 4662, Jan. 2002.

[17] L. Meylan, and S. Susstrunk, “High Dynamic Range Image Rendering Using a Retinex-Based Adaptive Filter, IEEE Trans. on Image Processing, Vol. 15, pp. 2820-2830, 2006.

[18] D. Shaked, and R. Keshet, “Robust Recursive Retinex Operators for Fast Retinex”, Hewlett Packard Technical Report HPL-2002-74, March 2002.

[19] R. Sobol, "Improving the Retinex algorithm for rendering wide dynamic range photographs" in Proc. of SPIE Vol. 4662, Jan. 2002.

[20] T. G. Stockham Jr, “Image Processing in the Context of a Visual Model”, Proc. of the IEEE, Vol. 60, 828--842, 1972.

[21] J. Tumblin, J. K. Hodgins, B. Guenter, “Two Methods for Display of High Contrast Images”, ACM Trans. on Graphics, Vol. 18, pp. 56-94, January 1999.

[22] J. Tumblin, and G. Turk, “Low Curvature Image Simplifiers (LCIS): A Boundary Hierarchy for Detail-Preserving Contrast Reduction”, in SIGGRAPH, pp. 83-90, 1999.

[23] D. Terzopoulos, “Image Analysis Using Multigrid Relaxation Methods”, IEEE Trans. on PAMI, Vol. 8, 129-139, 1986.