A Variational Framework for Retinex

Ron Kimmel, Doron Shaked, Michael Elad, Renato Keshet,
Irwin Sobel
HP Laboratories Israel
HPL-1999-151 (R.1)
June 28th , 2001*

Retinex, variational calculus, quadratic programming

Retinex theory addresses the problem of separating the illumination from the reflectance in a given image and thereby compensating for non-uniform lighting. This is in general an ill-posed problem. In this paper we propose a variational model for the Retinex problem that unifies previous methods. Similar to previous algorithms, it assumes spatial smoothness of the illumination field. In addition, knowledge of the limited dynamic range of the reflectance is used as a constraint in the recovery process. A penalty term is also included, exploiting a-priori knowledge of the nature of the reflectance image. The proposed formulation adopts a Bayesian view point of the estimation problem, which leads to an algebraic regularization term, that contributes to better conditioning of the reconstruction problem.

Based on the proposed variational model, we show that the illumination estimation problem can be formulated as a Quadratic Programming optimization problem. An efficient multi-resolution algorithm is proposed. It exploits the spatial correlation in the reflectance and illumination images. Applications of the algorithm to various color images yield promising results.

* Internal Accession Date Only
Approved for External Publication
HP Labs Israel, Technion City, Haifa 32000, Israel

A Variational Framework for Retinex

Ron Kimmel[‡], **Doron Shaked, Michael Elad,**
Hewlett-Packard Laboratories – Israel
and Irwin Sobel
Hewlett-Packard Laboratories in Palo-Alto.

Abstract

Retinex theory addresses the problem of separating the illumination from the reflectance in a given image and thereby compensating for non-uniform lighting. This is in general an ill-posed problem. In this paper we propose a variational model for the Retinex problem that unifies previous methods. Similar to previous algorithms, it assumes spatial smoothness of the illumination field. In addition, knowledge of the limited dynamic range of the reflectance is used as a constraint in the recovery process. A penalty term is also included, exploiting a-priori knowledge of the nature of the reflectance image. The proposed formulation adopts a Bayesian view point of the estimation problem, which leads to an algebraic regularization term, that contributes to better conditioning of the reconstruction problem.

Based on the proposed variational model, we show that the illumination estimation problem can be formulated as a Quadratic Programming optimization problem. An efficient multi-resolution algorithm is proposed. It exploits the spatial correlation in the reflectance and illumination images. Applications of the algorithm to various color images yield promising results.

[‡]R. Kimmel is also affiliated with the Computer Science Dept., Technion, Haifa 32000, Israel.

1 Introduction

Retinex theory deals with compensation for illumination effects in images. The primary goal is to decompose a given image S into two different images, the reflectance image R, and the illumination image L, such that, at each point (x, y) in the image domain, $S(x, y) = R(x, y) \cdot L(x, y)$. The benefits of such a decomposition include the possibility of removing illumination effects of back/front lighting, enhancing shots that include spatially varying illumination such as images that contain indoor and outdoor zones, and correcting the colors in images by removing illumination induced color shifts.

Recovering the illumination from a given image is known to be a mathematically ill-posed problem, and algorithms proposed in the literature for its solution vary in their way of overcoming this limitation. The Retinex methodology was motivated by Land's landmark research of the human visual system [11]. Through his experiments it was shown that our visual system is able to practically recognize and match colors under a wide range of different illuminations, a property that is commonly referred to as the *Color Constancy Phenomenon*. As a matter of fact, Land's findings indicated that even when retinal sensory signals coming from different color patches under different illuminations are identical, subjects were able to name the surface reflectance color [11]. The ability to extract the illumination image is sufficient but not nessecary to achieve this property.

In this paper we define the Retinex reconstruction problem for gray-level images through physically motivated considerations. The proposed formulation is shown to be a mathematically well-posed problem. A variational expression is obtained by defining the optimal illumination as the solution of a Quadratic Programming (QP) optimization problem. It is shown that different previous algorithms are essentially solutions to similar variational problems. We introduce an efficient algorithm that exploits knowledge on QP solvers and the fact that the unknown illumination is spatially smooth. Our algorithm uses a multi-resolution reconstruction of the illumination with few relaxation iterations at each resolution layer.

We apply and compare the proposed algorithm in two color spaces. The first operates in the RGB space, in which each spectral channel is processed separately. The second is the HSV color space in which only the Value (V) channel is processed. Color corrections can be obtained as a by-product of the Retinex algorithm in the RGB space. In this case the resulting reflectance image usually appears to be over-enhanced. A relaxation algorithm for this effect is proposed. Tests on various images support the assumption that the results of the proposed formulation are similar to those of the human visual system.

This paper is organized as follows: In the next section we review several different Retinex algorithms. Some of those were motivated by assumptions based on the color constancy process in the human visual system. Section 3 presents the proposed formulation along with an efficient numerical algorithm for the illumination reconstruction. Uniqueness and convergence of the solution are also discussed in Section 3. Section 4 extends the proposed

formulation to color images. In Section 5 we apply the method to different images and demonstrate the algorithm's performances and the effects of its free parameters. Section 6 summarizes the paper and presents topics for future research.

2 Previous Work

The first Retinex algorithms proposed by Land et al., were of random walk type [12, 14]. Subsequent algorithms [8, 9, 13] used Homomorphic Filters [5, 21]. Yet another group of Retinex algorithms is based on solving a Poisson equation [2, 7, 20]. A recent Retinex algorithm by McCann and Sobel [17], is an iterative multi-resolution type of non-linear filter.

A first step taken by most algorithms is the conversion to the logarithmic domain by $s = \log S$, $l = \log L$, $r = \log R$, and thereby $s = l + r$. This step is motivated both mathematically, preferring additions over multiplications, and physiologically, referring to the sensitivity of our visual system [11]. The different Retinex algorithms usually have the same flow chart as shown in Figure 1, and the difference between them concentrates on the actual estimation of the illumination image. Let us review these different algorithms.

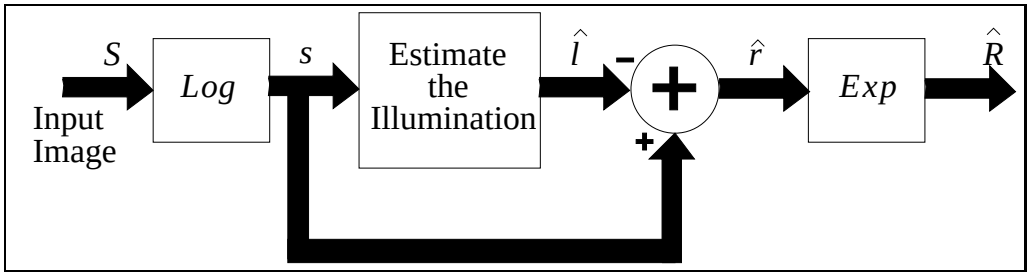

Figure 1: The general flow chart of Retinex algorithms

2.1 Random Walk Algorithms

A random walk is a discrete time random-process in which the 'next pixel position' is chosen randomly from the neighbors of the current pixel position. Random walk type Retinex algorithms are variants of the following basic formulation [4]: A large number of walkers are initiated at random locations of an input image s, adopting the gray-value of their initial position. An accumulator image A that has the same size as s is initialized to zero. As the walkers walk around, they update A by adding their values to each position they visit. Finally, the illumination image is obtained by normalizing the accumulator image, i.e., its value at each location divided by the number of walkers visited it.

By using many walkers with long paths, it is easily verified that each accumulator value

assimptotically converges to a Gaussian average of its neighbors, which is a low-pass filter of the image s [19]. A low-pass filter for the reconstruction of l from s was also proposed under the name of *Homomorphic Filtering*.

2.2 Homomorphic Filtering

Homomorphic Filtering type Retinex algorithms [5, 8, 9, 13, 21] share the following basic motivation: Assume the reflectance image corresponds to the sharp details in the image (i.e. edges), whereas the illumination image is expected to be spatially smooth, a reasonable guess for l is $\hat{l} = \mathrm{LP}\{s\}$, where LP is usually a convolution with a wide Gaussian kernel. This way, one actually applies the same process as the random walk algorithms by a single direct convolution.

2.3 Poisson Equation Solution

Following the above reasoning, since the illumination is expected to be spatially smooth, its derivative should be close to zero everywhere. On the other hand, by the assumption that the reflectance is piece-wise constant, its derivative is expected to vanish almost everywhere, and get high values along the edges. Thus, if we take the derivative of the sum $s = l + r$ and clip out the high derivative peaks, we can assume that the clipped derivative signal corresponds only to the illumination.

Poisson Equation type Retinex algorithms [2, 7, 20] rely on Land's Mondrian world model. The Mondrian model boils down to the above assumption on the reflectance as a piece-wise constant image. Applying the Laplacian, and the following clipping operation

$$\tau(\Delta s) = \begin{cases} \Delta s & \text{where } |\Delta s| < T \\ 0 & \text{otherwise,} \end{cases}$$

we get the following Poison equation

$$\Delta \hat{l} = \tau(\Delta s).$$

As to the solution of the resulting Poisson equation, Horn [7] suggested an iterative procedure which effectively inverts the Laplacian operator. Similar to the previous methods, a low-pass filter is applied in order to solve the above equation. Blake [2] introduced an improvement to Horn's method. He proposed to extract the discontinuities from the image gradient magnitude instead of the Laplacian and thereby came up with better boundary conditions that deal with less trivial scenarios along the image boundary.

2.4 McCann's Algorithm

Recently, McCann and Sobel [17] proposed an algorithm that can be equivalently written as follows: The illumination image $\hat{l}_0$ is initialized to be s, the original image. The algorithm

performs the following iterative procedure,

$$\hat{l}_{n+1} = \max\left\{\frac{\hat{l}_n + s}{2}, \frac{\hat{l}_n + D_n[\hat{l}_n]}{2}\right\}$$

where D_n is a translation operator, shifting the image by the n^{th} element of a sequence of spirally decaying translation vectors $\{d_n\}$, as shown in Figure 2. The size of the first displacement is set to be half the minimum between the image width and height.

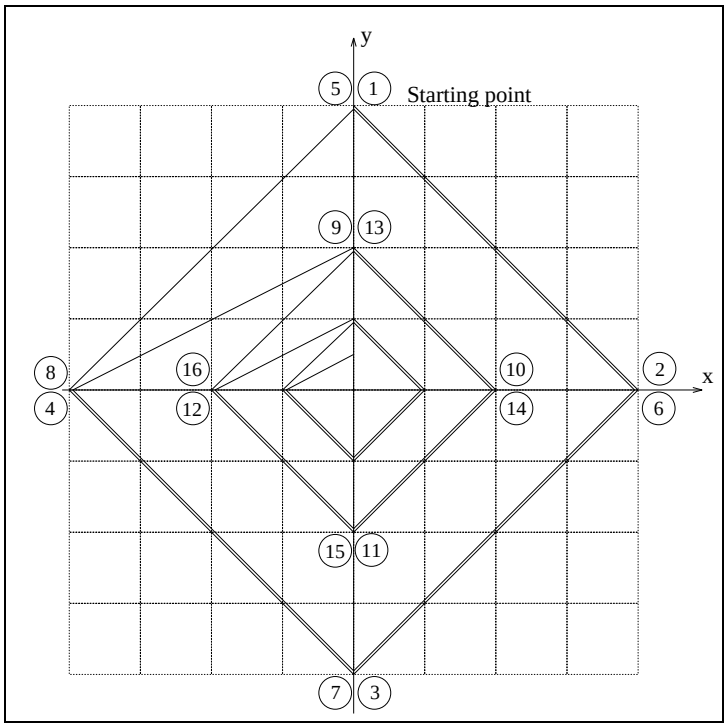

Figure 2: The sequence of displacement vectors for the D_n operator in McCann-Sobel algorithm.

Let us link this procedure to the previous methods. If we remove the max operation we get the simplified version

$$\hat{l}_{n+1} = \frac{\hat{l}_n + D_n[\hat{l}_n]}{2}.$$

This is a simple averaging operation that smoothes the image. Actually, it is possible to show that with the displacements shown in Figure 2, the effective smoothing kernel approaches a Gaussian.

The non-linear (max) operation inside the loop forces the illumination image to satisfy the constraint $\hat{l} \geq s$. This operation thus incorporates a physical property that previous methods

neglect. The physical nature of reflecting objects is such that they reflect only part of the incident light. Thus, the reflectance is restricted to the range $R \in [0, 1]$, and $L \geq S$, which implies $l \geq s$.

A multi-resolution version is also proposed in [17]. A Gaussian pyramid is constructed for the given image s. The algorithm starts at the coarsest level, and the size of the displacements for D_n are one pixel in each direction for each resolution. The multi-resolution version is significantly faster, yet produces lower quality results compared to the original version.

2.5 Summary of Previous Work

The discussion in this section suggests that the previous seemingly different algorithms are actually very similar. They are all based on the spatial smoothness assumption of the illumination l. All the above algorithms apply some sort of Gaussian smoothing to s in order to extract $\hat{l}$. Some methods add more assumptions about the reflectance, such as its limited range, or its Mondrian form. Eventually, 'skinning' the illumination from the given image yields the reflectance image, which is expected to be free of non-uniform illumination, have a reduced dynamic range, and present a sharper view of the scene.

3 The Variational Framework

3.1 Functional Definition

We start by listing the known information about the illumination image.

1. The first important assumption about the illumination is its spatial smoothness.
2. We also know that, since R is restricted to the unit interval, we can add the constraint $L \geq S$. Since the log function is monotone, we also have $l \geq s$.
3. By setting $l = \text{Const}$, where Const is any constant above the maximal value of s, we get a trivial solution that satisfies the two previous assumptions. We therefore add the assumption that the illumination image is close to the intensity image s, i.e., it minimizes a penalty term of the form $\text{dist}(l, s)$, e.g., the L_2 norm $(l - s)^2$.
4. The reflectance image $r = s - l$ can be assumed to have a high prior probability [3, 6, 10, 16]. One of the simplest prior functions used for natural images assigns high probability to spatially smooth images [10].
5. We can assume that the illumination continues smoothly as a constant beyond the image boundaries. This is an artificial assumption required for boundary conditions that would have minor effect on the final results.

Collecting all the above assumptions into one expression we get the following penalty functional

$$\begin{array}{rll} \text{Minimize:} & F[l] = \int_{\Omega} \left(|\nabla l|^2 + \alpha (l-s)^2 + \beta \left| \nabla (l-s) \right|^2 \right) dxdy & \\ \text{Subject to:} & l \geq s, \quad \text{and} \quad \langle \nabla l, \vec{n} \rangle = 0 \quad \text{on} \quad \partial\Omega, & (1) \end{array}$$

where Ω is the support of the image, $\partial\Omega$ its boundary, and $\vec{n}$ is the normal to the boundary. α and β are free non-negative real parameters. In the functional $F[l]$, the first penalty term ($|\nabla l|^2$) forces spatial smoothness on the illumination image. This choice of smoothness penalty is natural, if we keep in mind that minimizing $\int (|\nabla l|^2)\, dxdy$ translates into the Euler-Lagrange (EL) equation $\Delta l = 0$. Its steepest descent solution is a Gaussian smoothing operation with increasing variance of the initial condition. As mentioned in the previous section, several authors proposed Gaussian smoothing of s for the illumination reconstruction.

The second penalty term $(l-s)^2$ forces a proximity between l and s. The difference between these images is exactly r, which means that the norm of r should be small (i.e., R tends to Black). This term is weighted by the free parameter α. The main objective of this term is a regularization of the problem that makes it better conditioned. Notice that, in addition, we force the solution l to be $l \geq s$.

The third term represents a Bayesian penalty expression. It forces the reflectance image r to be a 'visually pleasing' image. This term forces r to be spatially smooth, and it is weighted by the free parameter β. Note that more complicated Bayesian expressions may be used allowing sharp edges, textures, $1/f$ behavior, etc. [3, 6, 10, 16]. As long as this expression is purely quadratic, the above minimization problem remains fairly simple.

The problem we have just defined has a Quadratic Programming (QP) form [1, 15]. The necessary and sufficient conditions for its minimization are obtained via the Euler-Lagrange equations

$$\forall (x,y) \in \Omega \quad \left\{ \begin{array}{c} \frac{\partial F[l]}{\partial l} = 0 = -\Delta l + \alpha (l-s) - \beta \Delta (l-s) \quad \text{and} \quad l > s \\ \\ \text{or} \\ \\ l = s \end{array} \right\} \qquad (2)$$

Note that the differential equation does not have to hold when $l = s$.

3.2 Numerical Solution

The minimization problem is QP with respect to the unknown image l. Many algorithms for solving such problems are known in the literature [1, 15]. In this paper we chose to focus on the Projected Normalized Steepest Descent (PNSD) algorithm, accelerated by a multi-resolution technique.

3.2.1 Projected Normalized Steepest Descent

The PNSD algorithm requires the application of a Normalized Steepest Descent (NSD) iteration that minimizes the functional $F[l]$, followed by a projection onto the constraints. A NSD iteration has the format:

$$l_j = l_{j-1} - \mu_{\text{NSD}} \cdot G,$$

where l_j and l_{j-1} are the illumination images at step j and $j-1$, respectively, G is the gradient of $F[l]$, and μ_{NSD} is the optimal line-search step size. In our case, Equation (2), the gradient of $F[l]$ is given by:

$$G = -\Delta l_{j-1} + (\alpha - \beta\Delta)\,(l_{j-1} - s)\,,$$

and μ_{NSD} is given by:

$$\mu_{\text{NSD}} = \frac{\int |G|^2}{\int (\alpha |G|^2 + (1+\beta)|\nabla G|^2)}$$

Observe that, by integration by parts, $\int |\nabla G|^2 = -\int G \Delta G$ up to boundary conditions.

An alternative approach is the Steepest Descent (SD) algorithm, where μ_{NSD} is replaced by a constant value μ_{SD}, such that:

$$\mu_{\text{SD}} \in \left(0,\ \frac{2}{\lambda_{max}\{-(1+\beta)\Delta + \alpha I\}}\right),$$

where $\lambda_{max}\{A\}$ refers to the greatest eigenvalue of the linear operator A. This alternative method saves computations at the expense of a slightly slower convergence.

Finally, projecting onto the constraint $l \geq s$ is done by $l_j = \max(l_j, s)$.

Notice that G can be calculated by:

$$G = G_A + \alpha\,(l_{j-1} - s) - \beta\,(G_A - G_B)\,,$$

where

$$\begin{aligned} G_A &\triangleq \Delta l_{j-1}, \\ G_B &\triangleq \Delta s_k. \end{aligned}$$

Similarly, μ_{NSD} is given by:

$$\mu_{\text{NSD}} = \frac{\mu_A}{\alpha\mu_A + (1+\beta)\mu_B},$$

where

$$\begin{aligned} \mu_A &\triangleq \int |G|^2, \\ \mu_B &\triangleq \int |\nabla G|^2. \end{aligned}$$

We approximate the Laplacian by a linear convolution with the kernel κ_{LAP}

$$\kappa_{\mathrm{LAP}} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix},$$

and the integrations are approximated by summations:

$$\begin{aligned} \int |G|^2 &\approx \sum_n \sum_m G[n,m]^2 \\ \int |\nabla G|^2 &= -\int G \Delta G \\ &\approx -\sum_n \sum_m G[n,m] \left(G * \kappa_{\mathrm{LAP}}\right)[n,m], \end{aligned}$$

where $G[m,n] = G(m\Delta x, n\Delta y)$ In order to accommodate the boundary conditions, as given in Equation (1), the above convolution is applied on an expanded version of the image G. Ths extension is done by replicating the first and last columns and rows. After the convolution, the additional rows and columns are removed.

3.2.2 Multi-Resolution

Although simple, the PNSD algorithm usually converges slowly [1, 15]. Instead of general acceleration schemes, we use the fact that the unknown image l is assumed to be smooth. Specifically, we apply a multi-resolution algorithm that starts by estimating a coarse resolution image l, expands it by interpolation and uses the result as an initialization for the next resolution layer. This way, few iterations at each resolution are enough for convergence.

3.2.3 The Algorithm

Summarizing the above, a proposed algorithm for the solution of Equation (1) involves the following steps,

1. **Input:** The input to the algorithm is an image s of size $[N, M]$, and two parameters α and β.
2. **Initialization:** Compute a Gaussian pyramid of the image s. This pyramid is constructed by smoothing the image with the kernel κ_{PYR}:

$$\kappa_{\text{PYR}} = \begin{bmatrix} \frac{1}{16} & \frac{1}{8} & \frac{1}{16} \\ \frac{1}{8} & \frac{1}{4} & \frac{1}{8} \\ \frac{1}{16} & \frac{1}{8} & \frac{1}{16} \end{bmatrix}$$

and decimating by 2:1 ratio. The process is repeated p times and produces a sequence of images $\{s_k\}_{k=1}^{p}$. The image s_1 is the original image s, and s_p is the one with the coarsest resolution in this pyramid. Define the numerical inner product

$$\langle G, F \rangle = \sum_{n=1}^{N} \sum_{m=1}^{M} G[n,m]F[n,m],$$

and the numerical Laplacian at the k^{th} resolution as $\Delta_k G = G * k_{LAP} 2^{-2(k-1)}$.

Set $k = p$, i.e., start at the coarsest resolution layer, and set the initial condition $l_0 = \max\{s_p\}$.

3. **Main Loop:** For the k^{th} resolution layer,

- **Calculate** $G_B \triangleq \Delta_k s_k$.
- **For** $j = 1, .., T_k$ **Do:**
  (a) Calculate gradient:

$$\begin{aligned} G_A &\triangleq \Delta_k l_{j-1}, \\ G &\leftarrow G_A + \alpha\left(l_{j-1} - s_k\right) - \beta\left(G_A - G_B\right). \end{aligned}$$

  (b) Calculate μ_{NSD}:

$$\begin{aligned} \mu_A &\triangleq \langle G, G \rangle, \\ \mu_B &\triangleq -\langle G, \Delta_k G \rangle, \\ \mu_{\text{NSD}} &\leftarrow \mu_A / \left(\alpha\mu_A + (1+\beta)\mu_B\right). \end{aligned}$$

  (c) Complete NSD iteration:

$$l_j \leftarrow l_{j-1} - \mu_{\text{NSD}} \cdot G,$$

  (d) Project onto the constraints

$$l_j = \max\left\{l_j, s_k\right\}.$$

- **End** j **Loop**;

The above loop solves the intermediate problem

$$\begin{aligned} \text{Minimize:} \quad & F_k[l] = \int_{\Omega_k} \left(|\nabla l|^2 + \alpha (l - s_k)^2 + \beta \, |\nabla (l - s_k)|^2 \right) dxdy \\ \text{Subject to:} \quad & l \geq s_k \qquad \text{and} \langle \nabla l, \vec{n} \rangle = 0 \quad \text{on } \partial\Omega, \end{aligned}$$

4. **Update the next resolution layer:**
If $k > 1$, the result l_{T_k} is up scaled (2:1 ratio) by pixel replication into the new l_0, the initialization for the next resolution $k-1$ layer. The resolution layer is updated $k = k - 1$, and the algorithm proceeds by going again to Step 3. If $k = 1$, the result l_{T_1} is the final output of the algorithm.

3.3 Relation to previous methods

Let us revisit the algorithms described in Section 2 and analyze them in light of the proposed formulation. First, by setting $\alpha = \beta = 0$, and removing the constraint $l \geq s$ we get the Homomorphic filtering, that was shown to be equivalent to the basic formulation of the random walk algorithms. Next, setting $\alpha = \beta = 0$, with the constraint $l \geq s$, one possible numerical relaxation scheme for the solution of the resulting problem is the McCann-Sobel algorithm.

The Poisson Equation approach seems to be unrelated directly to our formulation. However, if we set $\alpha(x, y) = \tau(\Delta s)$ and set the second distance term to $\int \alpha(x, y)(l - s)$, keeping the constraint $l \geq s$, we get that the optimal illumination should satisfy the equation

$$\Delta l = \tau(\Delta s), \tag{3}$$

subject to $l \geq s$, which is identical (up to the constraint) to Horn's formulation.

3.4 Uniqueness and Convergence

In this section we prove the uniqueness of the solution to Equation (1), and the convergence of the proposed numerical algorithm. The following theorem shows that the convexity of the problem guarantees existence and uniqueness of the solution.

Theorem: *The variational optimization problem* P, *given by*

$$\begin{aligned} \text{Minimize:} \quad & F[l] = \int_{\Omega} \left(|\nabla l|^2 + \alpha (l - s)^2 + \beta \, |\nabla (l - s)|^2 \right) dxdy \\ \text{Subject to:} \quad & l \geq s, \qquad \text{and} \quad \langle \nabla l, \vec{n} \rangle = 0 \quad \text{on} \quad \partial\Omega, \end{aligned}$$

with $\alpha > 0$ *and* $\beta \geq 0$, *has a unique solution.*

The proof is given in Appendix A.

Regarding the convergence of the numerical scheme, the core of the proposed algorithm is the Projected Normalized Steepest Descent (PNSD) algorithm, which is known to converge for convex optimization problems, such as our case [1, 15]. The pyramidal shell of the algorithm can be considered as an efficient method for creating a good initialization for the highest resolution layer stage. We found that few iterations at the finer resolution layer are sufficient for effective convergence.

4 Color Images

Thus far we dealt with a single channel. In this section, we apply our method to color images. When we process color images one option is to deal each color channel separately. We refer to channel-by-channel processing as 'RGB Retinex'. Treating the R, G, and B channels separately usually yields a color correction effect. For example, RGB Retinex on a reddish image is expected to modify the illumination in such a way that the red hue is removed so that the resulting image is brightened and corrected. Therefore, for some kinds of images, RGB Retinex actually improves the colors. In few other cases, such color correction can cause color artifacts that exaggerate color shifts, or reduce color saturation.

Another approach is to map the colors into a different color space, such as HSV, apply the Retinex correction only to the intensity layer, and then map back to the RGB domain. We refer to this method as the 'HSV Retinex'. Color shifts in such cases are less-likely. The advantage is that we have to process a single channel. The main drawback is that colors are no longer corrected with respect to the illumination hue.

5 Alternative Illumination Correction

The reflectance image obtained by the Retinex process is sometimes an over-enhanced image. This can be explained by the facts that *i)* the human visual system usually prefers some illumination in the image, and that *ii)* removal of all the illumination exposes noise that might exist in darker regions of the original image.

We propose adding a corrected version of the reconstructed illumination back to the reconstructed reflectance image. Figure 3 describes this operation. The proposed scheme computes the illumination image $L = \exp(l)$ from the intensity image $S = \exp(s)$, and the reflectance image $R = S/L$, as discussed in previous sections. Then, we 'tune up' the illumination image L by a Gamma Correction operation with a free parameter γ, obtain a new illumination image L', and multiply it by R, that gives the output image $S' = L' \cdot R$. The Gamma correction is performed by

$$L' = W \cdot \left[\frac{L}{W}\right]^{\frac{1}{\gamma}}, \tag{4}$$

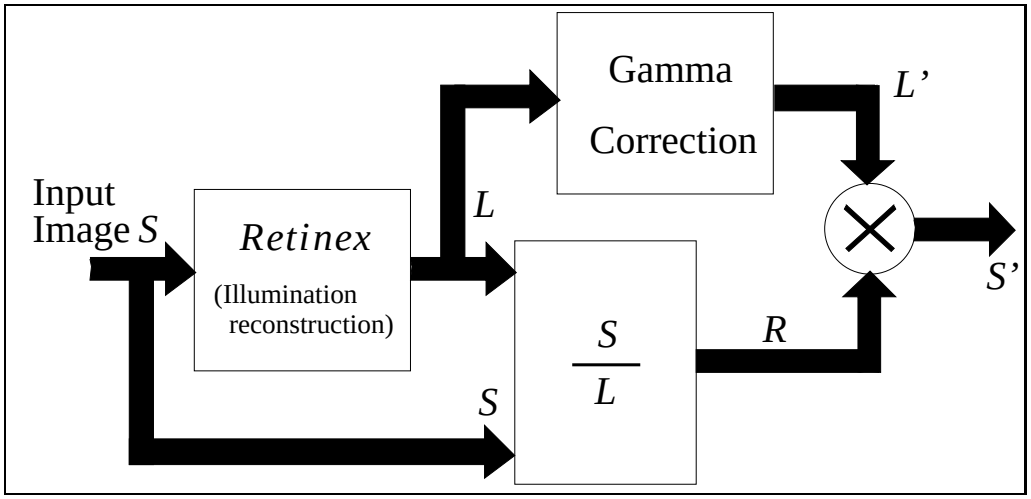

Figure 3: Returning part of the illumination to the reflectance image

where W is the White value (equal to 255 in 8-bit images).

The final result S' is given, therefore, by:

$$\begin{aligned} S' &= L' \cdot R = \frac{L'}{L} S \\ &= W \frac{(L/W)^{1/\gamma}}{L} S = \frac{S}{(L/W)^{1-1/\gamma}}. \end{aligned} \tag{5}$$

For $\gamma = 1$, the whole illumination is added back, and therefore $S' = S$. For $\gamma = \infty$, no illumination is returned, and we get $S' = R \cdot W$, which is the same reflectance image, R, as obtained by the original retinex, stretched to the interval $[0, W]$. The later case can also be considered as adding a maximum constant illumination W to the reflectance image R.

Adding part of the illumination to the final image can also be found in the homomorphic filtering approach. In [18, Chapter 10], the proposed linear filter for the illumination calculation in the log domain, removes high-pass spatial components of s, yet also attenuates the low-pass components by a factor of γ_i (where i stands for *illumination*). This is analog to a gamma correction of the illumination with $\gamma = \gamma_i$, since Equation (5) can be written in the form:

$$\frac{S'}{W} = \left(\frac{L}{W}\right)^{1/\gamma} \cdot R, \tag{6}$$

and therefore:

$$s' - w = \frac{1}{\gamma}(l - w) + r$$

$$= \frac{1}{\gamma}(\text{low-pass components}) + (\text{high-pass components}). \quad (7)$$

6 Results

In our experiments we applied the numerical algorithm of Section 3 to several test images. The results correspond to $\alpha = 0.0001$ and $\beta = 0.1$, unless indicated differently. Four resolution layers were used with $T_k = 2, 4, 8$, and 16 iterations at each layer, 2 iterations at the finest ($k = 1$) and 16 at the coarsest resolution ($k = 4$).

In the first test, we apply the RGB and the HSV Retinex algorithms to seven input images. The results are shown in Figures 4–10.

The second test (Figure 11) presents the influence of the β values on the reconstructed reflectance image.

In Figure 12 we test the effect of the number of iterations on the reflectance result. In all cases, the number of iterations was fixed to Const $\cdot 2^k$, where k is the resolution layer, with $2 \cdot$ Const iteration at the finest resolution. The number of iterations is determined by the constant Const $\in \{1, 2, 4, 8, 16\}$.

In Figure 13, we restore the illumination through Gamma correction and add it back to the reflectance image. We use each of the values $\gamma = \{1,\ 2,\ 6,\ 24,\ 96\}$ for each output image, respectively.

Finally, we demonstrate the convergence of the proposed numerical algorithm. Figure 14 shows the values of $F[l_n]$, the functional in Equation (1), as a function of the number of iterations. The results correspond to the image in Figure 9. The algorithm in this case runs on the R, G, and B layers separately, without the pyramidal structure, so that all l_n are computed at the same resolution. As can be seen, a rapid convergence is obtained for all the three channels.

We conclude that

1. As we see in Figures 4–10, both the RGB and the HSV Retinex algorithms provided the desired results. The reflectance images are indeed enhanced versions of the original one, although in some versions they are over-enhanced.

2. The illumination feedback through Gamma correction seems to improve both the RGB and the HSV Retinex results. However, they have different effects: In the RGB Retinex this process restores some of the colors, whereas in the HSV Retinex, the result is merely darker.

3. When we compare the RGB and the HSV Retinex algorithms it seems that the better method depends on the input image. Generally speaking, for images with colored

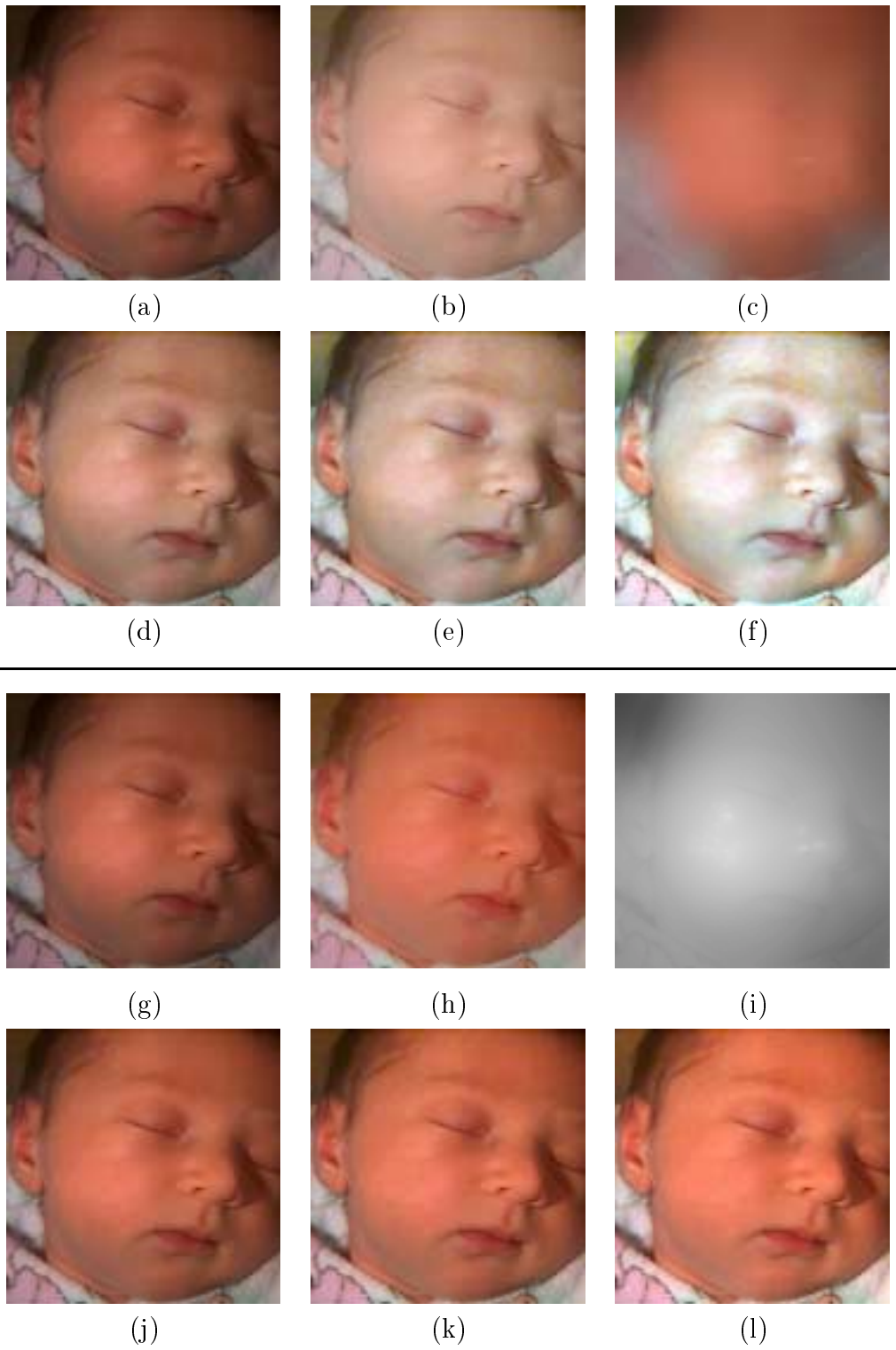

Figure 4: Example 1. Images (b)-(f) refer to processing of all RGB components separately, whereas images (h)-(l) correspond to processing only the V component of the HSV space. (a),(g) Original image, (b),(h) standard Gamma correction with $\gamma = 2.2$, (c),(i) estimated illumination image, (d),(j) proposed alg. with $\gamma = 2$, (e),(k) proposed alg. with $\gamma = 4$, (f),(l) reflectance image (proposed alg. with $\gamma = \infty$).

Figure 5: Example 2. Images (b)-(f) refer to processing of all RGB components separately, whereas images (h)-(l) correspond to processing only the V component of the HSV space. (a),(g) Original image, (b),(h) standard Gamma correction with $\gamma = 2.2$, (c),(i) estimated illumination image, (d),(j) proposed alg. with $\gamma = 2$, (e),(k) proposed alg. with $\gamma = 4$, (f),(l) reflectance image (proposed alg. with $\gamma = \infty$).

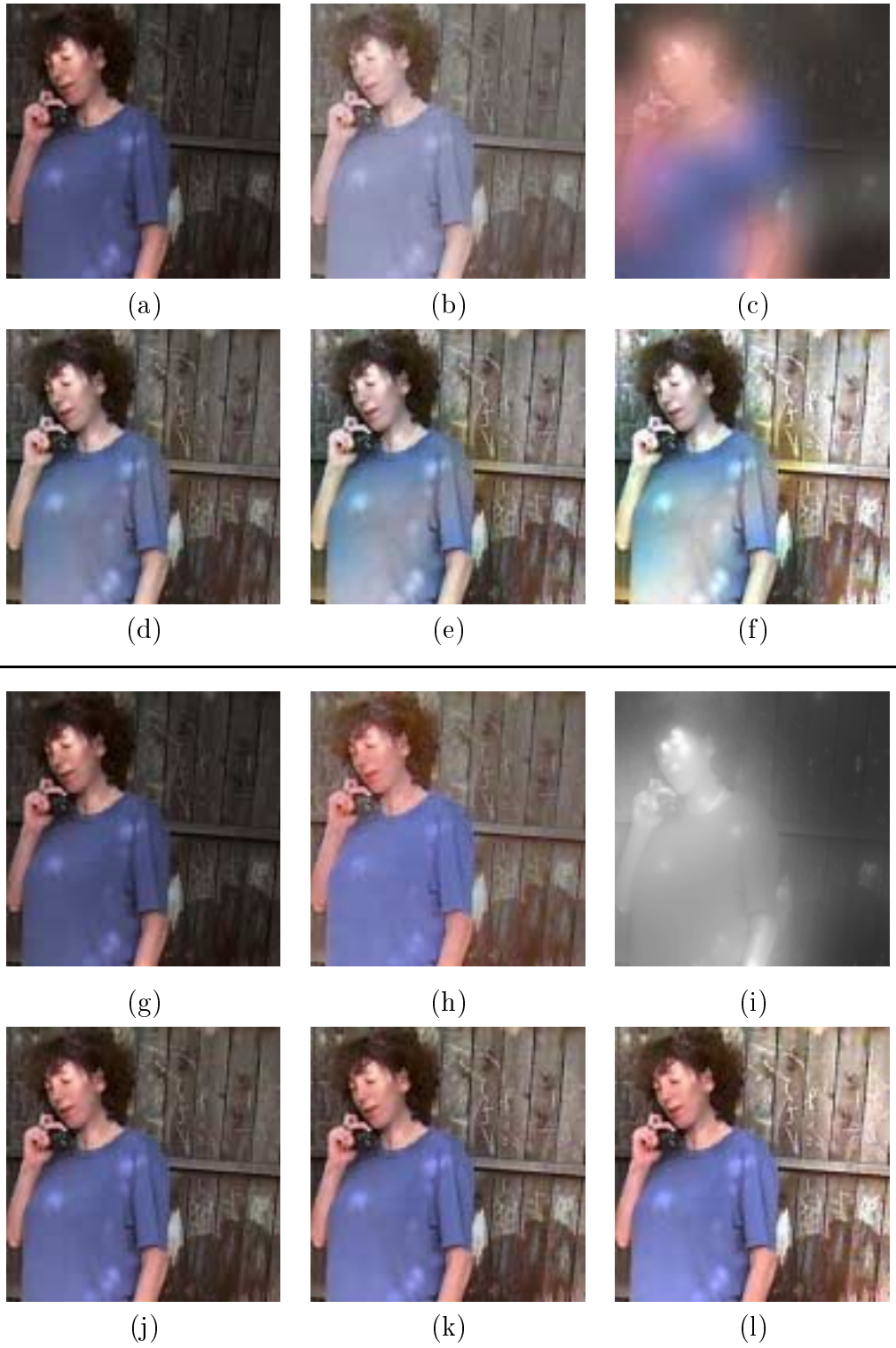

Figure 6: Example 3. Images (b)-(f) refer to processing of all RGB components separately, whereas images (h)-(l) correspond to processing only the V component of the HSV space. (a),(g) Original image, (b),(h) standard Gamma correction with $\gamma = 2.2$, (c),(i) estimated illumination image, (d),(j) proposed alg. with $\gamma = 2$, (e),(k) proposed alg. with $\gamma = 4$, (f),(l) reflectance image (proposed alg. with $\gamma = \infty$).

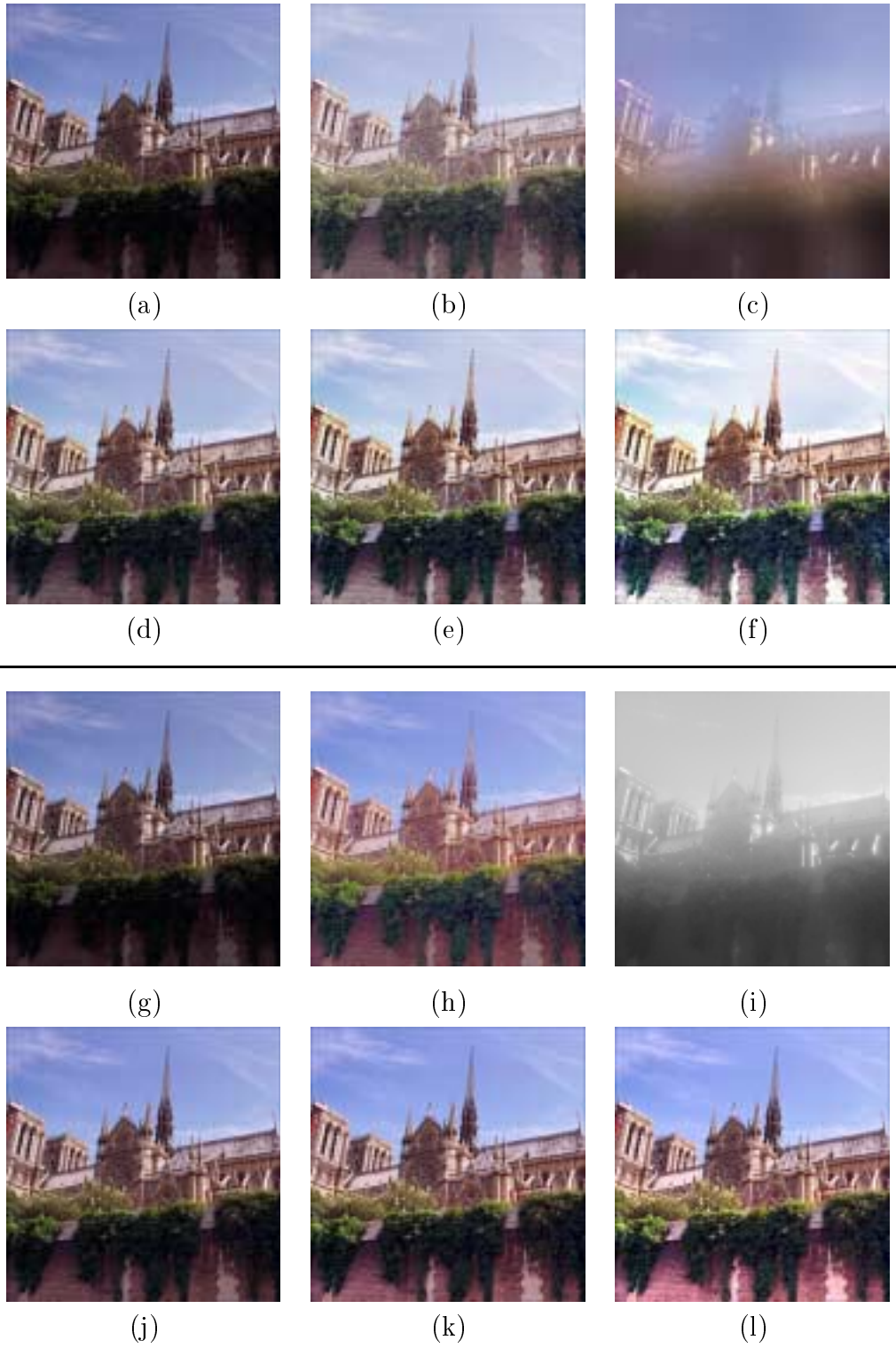

Figure 7: Example 4. Images (b)-(f) refer to processing of all RGB components separately, whereas images (h)-(l) correspond to processing only the V component of the HSV space. (a),(g) Original image, (b),(h) standard Gamma correction with $\gamma = 2.2$, (c),(i) estimated illumination image, (d),(j) proposed alg. with $\gamma = 2$, (e),(k) proposed alg. with $\gamma = 4$, (f),(l) reflectance image (proposed alg. with $\gamma = \infty$).

Figure 8: Example 5. Images (b)-(f) refer to processing of all RGB components separately, whereas images (h)-(l) correspond to processing only the V component of the HSV space. (a),(g) Original image, (b),(h) standard Gamma correction with $\gamma = 2.2$, (c),(i) estimated illumination image, (d),(j) proposed alg. with $\gamma = 2$, (e),(k) proposed alg. with $\gamma = 4$, (f),(l) reflectance image (proposed alg. with $\gamma = \infty$).

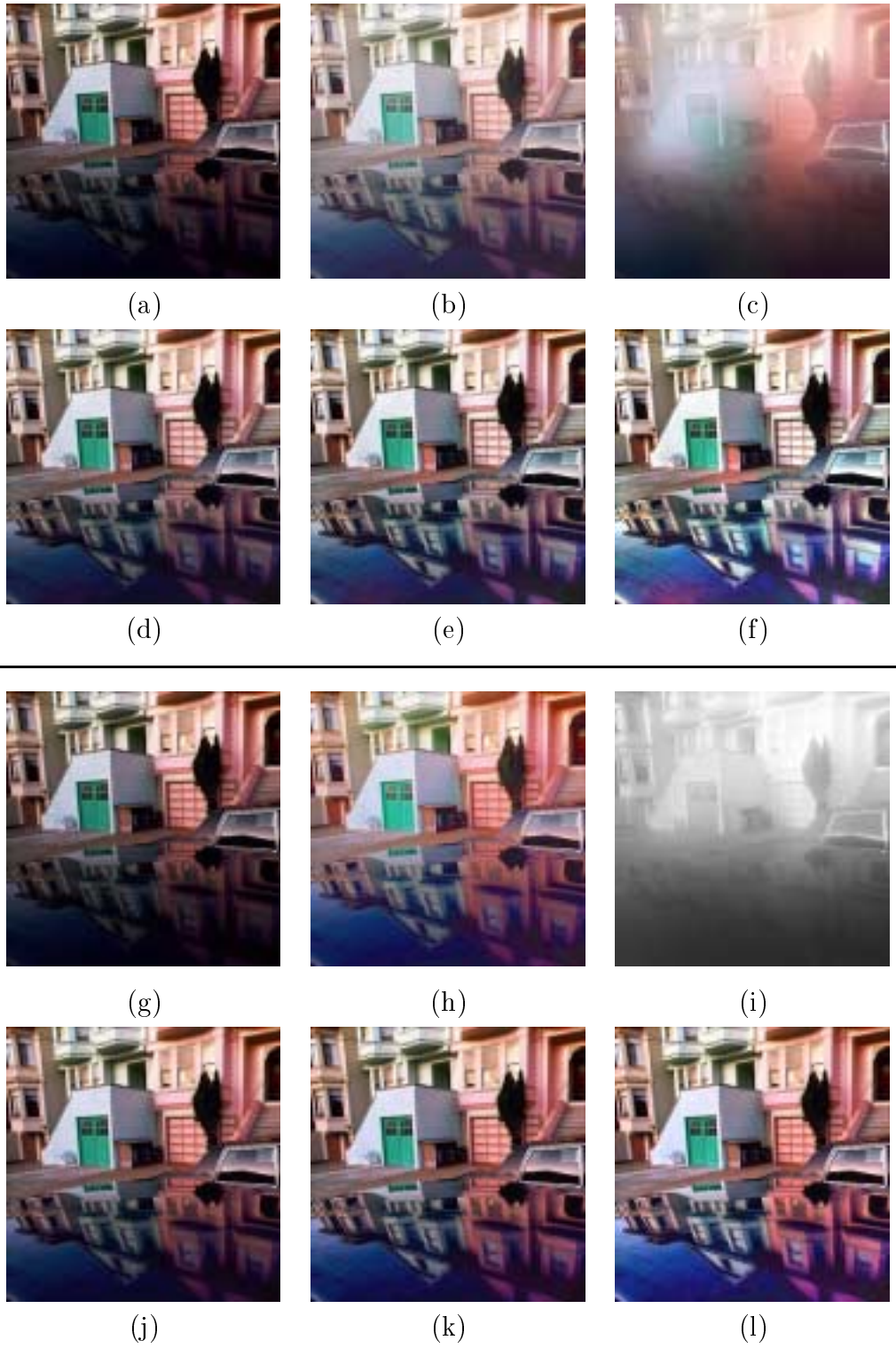

Figure 9: Example 6. Images (b)-(f) refer to processing of all RGB components separately, whereas images (h)-(l) correspond to processing only the V component of the HSV space. (a),(g) Original image, (b),(h) standard Gamma correction with $\gamma = 2.2$, (c),(i) estimated illumination image, (d),(j) proposed alg. with $\gamma = 2$, (e),(k) proposed alg. with $\gamma = 4$, (f),(l) reflectance image (proposed alg. with $\gamma = \infty$).

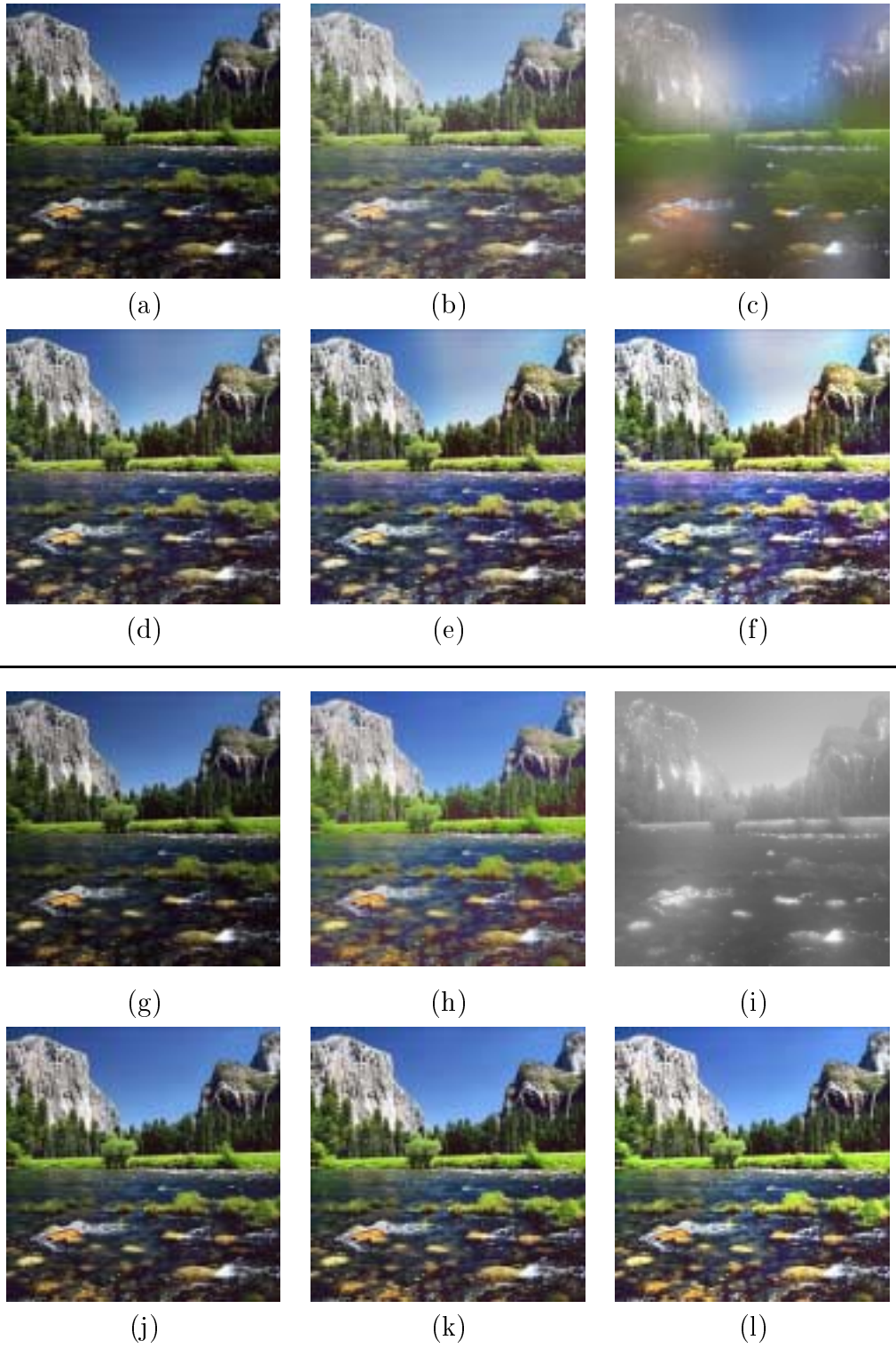

Figure 10: Example 7. Images (b)-(f) refer to processing of all RGB components separately, whereas images (h)-(l) correspond to processing only the V component of the HSV space. (a),(g) Original image, (b),(h) standard Gamma correction with $\gamma = 2.2$, (c),(i) estimated illumination image, (d),(j) proposed alg. with $\gamma = 2$, (e),(k) proposed alg. with $\gamma = 4$, (f),(l) reflectance image (proposed alg. with $\gamma = \infty$).

Figure 11: The influence of β. (a),(g) Original image, (b),(h) β = 1e-5, (c),(i) β = 1e-3, (d),(j) β = 1e-1, (e),(k) $\beta = 1$, (f),(l) $\beta = 10$. Images (b)-(f) refer to processing of all RGB components separately, whereas images (h)-(l) correspond to processing of V component of HSV space only.

Figure 12: The influence of varying the number of iterations T. (a),(g) Original image, (b),(h) $T = 2$, (c),(i) $T = 4$, (d),(j) $T = 8$, (e),(k) $T = 16$, (f),(l) $T = 32$. Images (b)-(f) refer to processing of all RGB components separately, whereas images (h)-(l) correspond to processing of V component of HSV space only.

(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j) (k) (l)

Figure 13: The influence of parameter γ (amount of illumination return). (a),(g) Original image, (b),(h) $\gamma = 1$, (c),(i) $\gamma = 2$, (d),(j) $\gamma = 6$, (e),(k) $\gamma = 24$, (f),(l) $\gamma = 96$. Images (b)-(f) refer to processing of all RGB components separately, whereas images (h)-(l) correspond to processing of V component of HSV space only.

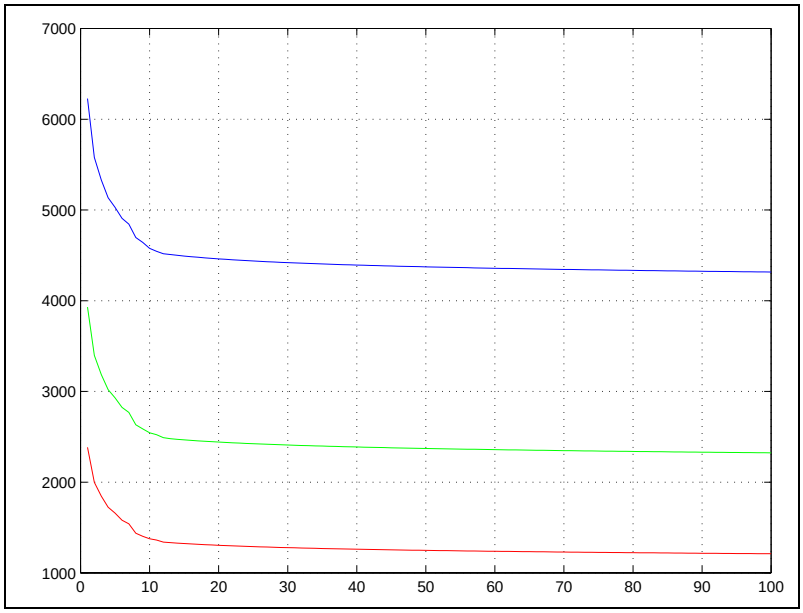

Figure 14: The functional value $F[l_n]$ as a function of the number of iterations, for the layers R (red), G (green) , and B (blue)

illumination, the RGB usually performs better, whereas for images with a milder illumination hue, the HSV is somewhat better.

4. The Retinex approach obviously performs better than a simple Gamma correction. The latter indeed improves the overall illumination of the image, but also decreases details contrast and flattens the objects. Retinex, on the other hand, usually increases both detail contrast and depth sensation in the image, as well as improving the overall illumination.

5. The proposed algorithm is robust to the choice of its parameters β, γ, and the number of iterations.

6. The proposed numerical method converges very fast to its steady-state solution which is also the minimizer of the defined functional in Equation (1).

7 Concluding Remarks

In this paper we surveyed several algorithms for image illumination correction and dynamic range compensation, based on a common motivation known as the Retinex theory. We have shown that in spite of their different formulations, these algorithms can be derived from the same variational principle.

We introduced a comprehensive Retinex analysis, motivated by the different Retinex algorithms. Our variational approach provides solid mathematical foundation, that yields efficient and robust numerical solutions.

We introduced a fast multi-resolution solution to the corresponding variational problem, resulting in an algorithm whose computational complexity amounts to less than 11 convolutions of the full size image with a 3x3 kernel plus a few addition algebraic operations per pixel. The advantages of the proposed algorithm are:

1. Computational efficiency.
2. Image quality comparable to state of the art methods, and in some cases better results.
3. Parameter robustness. It was shown that for a wide range of the involved parameters, the output quality is practically the same.

As part of the proposed Retinex enhancement algorithm, we proposed a new method to control the overall brightness of the image. Traditionally, after removing a non-uniform illumination via the Retinex, standard point operations like the γ-correction are typically required. According to the proposed method, the overall illumination correction is coupled with the non-uniform illumination removal. Instead of removing the illumination from the original image, the illumination is corrected via a standard point operation like the γ-correction, and returned to the reflectance image. Thus, dark regions in the image which have been poorly illuminated are better illuminated, as if the actual illumination conditions in the image are improved.

For color images, two modes of operations are possible, namely, performing Retinex on every color layer, or alternatively only on the luminance component. We compared these two Retinex modes. The later is naturally preferable in cases where the computational complexity is critical. However the former (RGB Retinex) enables color illumination compensation, e.g. removing Yellowish lighting in indoor images. For better results some of the reconstructed illumination is piped back through γ correction.

A Uniqueness of The Solution

Theorem: *The variational optimization problem* P, *given by*

$$\begin{aligned} \text{Minimize:} \quad & F[l] = \int_\Omega \left(|\nabla l|^2 + \alpha (l-s)^2 + \beta \, |\nabla (l-s)|^2 \right) dxdy \\ \text{Subject to:} \quad & l \geq s, \qquad \text{and} \quad \langle \nabla l, \vec{n} \rangle = 0 \quad \text{on} \quad \partial\Omega, \end{aligned}$$

with $\alpha > 0$ *and* $\beta \geq 0$, *has a unique solution.*

Proof: First, we show that the functional $F[l]$ is purely convex. The Hessian of the quadratic functional $F[l]$ is given by

$$\frac{\partial^2 F[l]}{\partial l^2} = -\left(1+\beta\right)\Delta + \alpha I,$$

where I is the identity operator. The multiplication of the Laplacian operator by the negative value $-(1+\beta) < -1$ yields a positive semi-definite operator $-\left(1+\beta\right)\Delta \geq 0$. Since $\alpha > 0$, $\alpha I > 0$, i.e., it is positive definite. Therefore, the Hessian is also a positive definite operator. Thereby, the functional $F[l]$ is a strictly convex functional [1, 15]. If $\alpha = 0$, the Hessian is semi-positive definite, and the convexity of $F[l]$ is not strict.

Define the set $C = \{l \mid l \geq s \ \text{ and } \ \langle \nabla l, \vec{n} \rangle = 0 \ \text{ on } \ \partial\Omega\}$ such that the constraints of P are equivalent to requiring $l \in C$. For every $l_1,\ l_2 \in C$, $\forall \theta \in [0,1]$, we have $\theta l_1 + (1-\theta) l_2 \in C$, or in other words, C is a convex set. This is true since C is the intersection of two convex sets (one per each original constraint).

Let us denote the minimum of the functional $F[l]$ as $\hat{l}_{opt}$. This solution is unique since $F[l]$ is strictly convex. If $\hat{l}_{opt} \in C$ than $\hat{l}_{opt}$ is the solution of P, and therefore we get a unique solution as the theorem claims.

On the other hand, if $\hat{l}_{opt} \notin C$, the solution to P is obtained on the boundary of the constraint set $C = \{l \mid l \geq s\}$. We prove this property by contradiction. Assume that the solution is given as $l_0 \in \text{Interior}\{C\}$. Define $l_1 = (1-\theta) l_0 + \theta \hat{l}_{opt}$ for $\theta \in (0,1)$. Due to the convexity of $F[l]$, it is clear that $F[l_1] < (1-\theta) F[l_0] + \theta F[\hat{l}_{opt}] < F[l_0]$. Since $l_0 \in C$, for θ sufficiently close to zero it can be guaranteed that $l_1 \in C$ as well. This way we get l_1 as a better solution, which contradicts our assumption. Thus, the solution for P is obtained on the boundary of C.

Let us now assume that two solutions are possible, and prove that this assumption leads to a contradiction. The two optimal solutions l_1 and l_2 must satisfy the following set of conditions

1. The solutions should be feasible: $l_1,\ l_2 \in C$.

2. Based on the previous results, the solutions should be on the boundary of C: $l_1,\ l_2 \notin$ Interior$\{C\}$.

3. The functional value of the two solutions should be the same: $F[l_1] = F[l_2]$.

4. The solutions are optimal: $\forall l \in C,\quad F[l] > F[l_1]$.

5. The solutions should not be equal to $\hat{l}_{opt}$, i.e., $F[l]$: $l_1,\ l_2 \neq \hat{l}_{opt}$.

Since C is convex, $\forall \theta \in (0,1),\quad l_0 = (1-\theta)l_1 + \theta l_2 \in C$. Moreover, by the strict convexity of $F[l]$, we have that $F[l_0] = F[(1-\theta)l_1 + \theta l_2] < (1-\theta)F[l_1] + \theta F[l_2] = F[l_1]$ and again, we got a better solution l_0. This contradicts the previous assumptions, and therefore, there is a unique solution to P. Q.E.D.

B References

[1] D. P. Bertsekas, *Non-Linear Programming*, Athena Scientific, Belmont, Massachusetts, 1995.

[2] A. Blake, "Boundary Conditions of Lightness Computation in Mondrian World", *Computer Vision Graphics and Image Processing*, Vol. 32, pp. 314–327, 1985.

[3] A. Blake and A. Zisserman, *Visual Reconstruction*, The MIT Press, Cambridge, Massachusetts 1987.

[4] D. H. Brainard, and B. Wandell, "Analysis of the Retinex Theory of Color Vision", *J. Opt. Soc. Am. A*, Vol. 3, pp. 1651–1661, 1986.

[5] O. D. Faugferas, "Digital Image Color Processing Within the Framework of a Human Visual System", *IEEE Trans. on ASSP*, Vol. 27, pp. 380–393, 1979.

[6] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distribution, and the Bayesian restoration of images", *IEEE Trans. on Pattern Analysis ans Machine*, Vol. 6, pp. 721-741, 1984.

[7] B. K. P. Horn, "Determining Lightness from an Image", *Computer Graphics and Image Processing*, Vol, 3. pp. 277–299, 1974.

[8] D. J. Jobson, Z. Rahman, and G. A. Woodell, "Properties and Performance of the Center/Surround Retinex", *IEEE Trans. on Image Proc.*, Vol. 6, pp. 451–462, 1997.

[9] D. 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.

[10] R. L. Lagendijk and J. Biemond, *Iterative Identification and Restoration of Images*, Kluwer Academic Publishing, Boston, Massachusetts, 1991.

[11] E. H. Land, "The Retinex Theory of Color Vision", *Sci. Amer.*, Vol. 237, pp. 108-128, 1977.

[12] E. H. Land, "Recent Advances in the Retinex Theory and Some Implications for Cortical Computations: Color Vision and the Natural Image", *Proc. Nat. Acad. Sci. USA*, Vol. 80, pp. 5163–5169, 1983.

[13] E. H. Land, "An Alternative Technique for the Computation of the Designator in the Retinex Theory of Color Vision", *Proc. Nat. Acad. Sci. USA*, Vol. 83, pp. 3078–3080, 1986.

[14] E. H. Land, and J. J. McCann, "Lightness and the Retinex Theory", *J. Opt. Soc. Am.*, Vol. 61, pp. 1–11, 1971.

[15] D. G. Luenberger, *Linear and Non-Linear Programming*, Addison-Wesley Pub., Menlo-Park, California, Second Edition, 1987.

[16] J. Marroquin, J. Mitter and T. Poggio, "Probabilistic solution for ill-posed problems in computational vision", *J. of the American Statistical Assos.*, Vol. 82, pp. 76-89, 1987.

[17] J. J. McCann, and I. Sobel, "Experiments with Retinex", HPL Color Summit, 1998.

[18] A. V. Oppenheim and R. W. Schafer, *Digital Signal Processing*, Prentice Hall: New Jersey, 1975.

[19] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, third Edition, pp. 345–346, McGraw-Hill, 1991.

[20] D. Terzopoulos, "Image Analysis Using Multigrid Relaxation Methods", *IEEE Trans. on PAMI*, Vol. 8, 129–139, 1986.

[21] T. G. Stockham Jr, "Image Processing in the Context of a Visual Model" *Proc. of the IEEE*, Vol. 60, 828–842, 1972.