

Mathematical definition and analysis of the Retinex algorithm

Edoardo Provenzi, Luca De Carli, and Alessandro Rizzi

Dipartimento di Tecnologie dell'Informazione, Università di Milano, Via Bramante 65, Crema (CR), Italy

Daniele Marini

Dipartimento di Informatica e Comunicazione, Università di Milano, Via Comelico 39, Milano, Italy

Received January 18, 2005; revised manuscript received April 27, 2005; accepted May 3, 2005

We present a detailed mathematical analysis of the original Retinex algorithm due to Land and McCann [J. Opt. Soc. Am. **61**, 1 (1071)]. To this end, we propose an analytic formula that describes the algorithm behavior. More than one Retinex version (e.g., with and without threshold) is examined. The behavior of Retinex varying the number of paths is predicted, and its recursive iterations are mathematically analyzed using the formula. The mathematical setting presented serves as a common ground for the various Retinex implementations. Its validity is confirmed by the tests on images that we have performed. © 2005 Optical Society of America
OCIS codes: 100.2000, 100.2980, 100.3020.

1. INTRODUCTION

The Retinex algorithm of Land and McCann¹ is one of the most famous attempts to model and explain how the human visual system (HVS) perceives colors. Land and McCann carried out several experiments demonstrating that color perception is not a simple signal acquisition. Our visual system automatically performs a “computation” on the acquired signal that makes our perception more complex than the trivial sampling of light distribution. Land and McCann called this computation Retinex.

During the past 40 years, the Retinex model has inspired a great variety of implementation and discussion^{2–28} with results, in some cases, that are difficult to compare. Even Land and McCann presented different Retinex versions.^{29–32} The model seems to have a renewed interest for the scientific community, as can be seen in the citations above. The state of the art of the research on Retinex can be read about in Ref. 33.

Here we present a mathematical definition of the basic Retinex algorithm.¹ Being written in mathematical language, this definition is completely independent of the peculiarities of the implementation, and we believe that it can be used as a common background for various Retinex versions. The need for such a definition has been recently pointed out in Ref. 20. We prove that the mathematical definition of the basic Retinex is well posed, showing that it is equivalent to the function computed by the original algorithm of Land and McCann.

As will be discussed in more detail, the Retinex algorithm depends on several parameters (such as threshold, number of paths, and iterations): We show that the qualitative behavior of Retinex in relation to the variation of these parameters can be predicted by using the mathematical definition. Finally, we report the results of several tests on images that confirm the predictions of the theoretical model.

2. ANALYTIC FORMULA FOR THE RETINEX ALGORITHM

Starting from the basic Retinex, we propose a formula to define the algorithm mathematically. We discuss its characteristics in the next subsection.

A. Basic Retinex Algorithm

The basic Retinex model is based on the assumption that the HVS operates with three retinal–cortical systems, each processing independently the low, middle, and high frequencies of the visible electromagnetic spectrum. Every independent process forms a separate image that determines a quantity called lightness and denoted by L . When Retinex is applied on digital RGB images,^{10,12} the triplet (L_R, L_G, L_B) of lightness values in the three chromatic channels is the information that determines, by superposition, the perception of what we call the color of every pixel of the image.

Inspired by several experiments,²⁹ Land and McCann found out that an efficient way to compute the lightness values of a pixel i in an image was to consider a certain number of paths, starting at random points and ending at i , and then to compute the average of the products of ratios between the intensity values of subsequent points in the paths, with these prescriptions:

- If the ratio does not differ from 1 more than a fixed threshold value, then the ratio is considered unitary.
- If the chain of ratios passes the value 1 at a point of the path, then the cumulated relative lightness is forced to 1, so that the computation restarts from that point.

The first prescription is called threshold mechanism, and, thanks to it, the algorithm disregards smooth changes in color due, for example, to smooth gradients of the illuminant.

The second prescription is called the reset mechanism, and it is responsible for the so-called white-patch behavior of Retinex, meaning that the points that enable the reset mechanism become the local references for white.

The only formulas available in the literature (see, e.g., Refs. 10 and 30) to describe the lightness L of a pixel i computed by Retinex in a given chromatic channel look like this:

$$L(i) = \frac{\sum_k l^{i,j_k}}{N}, \quad l^{i,j_k} = \sum_{x \in \text{path}} \delta \log \left| \frac{I_{x+1}}{I_x} \right|, \quad (1)$$

where j_k is the starting point of the k th path (i is the end point of every path) and where

$$\delta = \begin{cases} 1 & \text{if } \log \left| \frac{I_{x+1}}{I_x} \right| > \text{threshold} \\ 0 & \text{otherwise} \end{cases}. \quad (2)$$

In this paper we want to propose an alternative formulation, which is more suitable for further mathematical analysis. We hope that this formulation might stimulate the investigation of other interesting Retinex versions (e.g., Ref. 32).

B. General Background for the Formula

Given a digital image, consider a collection of N oriented paths γ_k composed of ordered chains of pixels starting at j_k and ending at i . Let n_k be the number of pixels traveled by the path γ_k , and let $t_k=1, \dots, n_k$ be its parameter, i.e., $\gamma_k: \{1, \dots, n_k\} \rightarrow \text{Image} \subset \mathbb{R}^2$, $\gamma_k(1)=j_k$ and $\gamma_k(n_k)=i$.

For simplicity, write two subsequent pixels of the path as $\gamma_k(t_k)=x_{t_k}$ and $\gamma_k(t_k+1)=x_{t_k+1}$, for $t_k=1, \dots, n_k-1$. Consider, in every fixed chromatic channel $c \in \{R, G, B\}$, the pixels' intensities $I(x_{t_k})$, $I(x_{t_k+1})$ and then compute the ratio $R_{t_k} = I(x_{t_k+1})/I(x_{t_k})$. For technical reasons put $R_0=1$ and normalize the intensities to take their values in the real unit interval (the normalization factor is 1/255 if 8 bytes are used for each pixel in every chromatic channel).

C. Formula

We claim that the (normalized) value of lightness given by Retinex for a generic pixel i , in every fixed chromatic channel c , can be obtained by this formula:

$$L(i) = \frac{1}{N} \sum_{k=1}^N \prod_{t_k=1}^{n_k-1} \delta_k(R_{t_k}), \quad (3)$$

where $\delta_k: \mathbb{R}^+ \rightarrow \mathbb{R}^+$, $k=1, \dots, N$, are functions defined in this way: $\delta_k(R_0)=1$ and, for $t_k=1, \dots, n_k-1$,

$$\delta_k(R_{t_k}) = \begin{cases} R_{t_k} & \text{if } 0 < R_{t_k} \leq 1 - \epsilon \\ 1 & \text{if } 1 - \epsilon < R_{t_k} < 1 + \epsilon \\ R_{t_k} & \text{if } 1 + \epsilon \leq R_{t_k} \leq \frac{1 + \epsilon}{\prod_{m_k=0}^{t_k-1} \delta_k(R_{m_k})} \\ \frac{1}{\prod_{m_k=0}^{t_k-1} \delta_k(R_{m_k})} & \text{if } R_{t_k} > \frac{1 + \epsilon}{\prod_{m_k=0}^{t_k-1} \delta_k(R_{m_k})} \end{cases}, \quad (4)$$

$\epsilon > 0$ being a fixed threshold.

The first option is satisfied when the intensity of the pixel x_{t_k+1} is appreciably smaller than the intensity of the pixel x_{t_k} ; then δ_k reproduces the value of the ratio R_{t_k} .

The second option occurs when only a very small change in intensity is measured between two subsequent pixels. In this case $\delta_k(R_{t_k})$ is defined to be 1, so that the product of the ratios remains exactly the same as in the previous step. This is the mathematical implementation of the threshold mechanism.

The third option is referred to as the case when the ratio R_{t_k} is greater than $1 + \epsilon$ but the product $\delta_k(R_1)\delta_k(R_2) \dots \delta_k(R_{t_k-1})R_{t_k}$ is not greater than $1 + \epsilon$. In this situation δ_k reproduces the value of R_{t_k} as in the first option.

Finally, the fourth option holds when $\delta_k(R_1)\delta_k(R_2) \dots \delta_k(R_{t_k-1})R_{t_k} > 1 + \epsilon$, and in this case δ_k resets the chain of products to 1 because a "local white pixel" has been reached. This option implements the reset mechanism (and so the white-patch behavior) of the algorithm.

It is useful to write the contribution of the single path γ_k to $L(i)$ as $L_k(i) = \prod_{t_k=1}^{n_k-1} \delta_k(R_{t_k})$, so that formula (3) reduces simply to the average of these contributions:

$$L(i) = \frac{1}{N} \sum_{k=1}^N L_k(i). \quad (5)$$

D. Proof of the Equivalence between the Formula and the Function Computed by Retinex

To prove that the definition of Retinex presented above is equivalent to the function it computes, it is enough to prove that the contributions $L_k(i)$ are the same as the ones computed by Retinex. We prove this by induction on the number of pixels: for $n_k=2$ the proof is trivial, so we start by showing the complete proof for a path with $n_k=3$ pixels, then we assume the correctness of the formula for n_k-1 , and finally we prove the correctness for n_k .

- $n_k=3$: $L_k(i) = \prod_{t_k=1}^{3-1} \delta_k(R_{t_k}) = \delta_k(R_1)\delta_k(R_2)$. We have to distinguish between 16 mutually different situations (the third and the fourth options correspond simply to $R_1=1 + \epsilon$ and $R_1 > 1 + \epsilon$, respectively); see Table 1. In all cases we recover the correct behavior of Retinex.

- n_k-1 : Assume that $L_k(i)|_{n_k-1} = \prod_{t_k=1}^{n_k-2} \delta_k(R_{t_k}) = \delta_k(R_1)\delta_k(R_2) \dots \delta_k(R_{n_k-2})$ is the expected contribution to $L(i)$.

Table 1. First Step in the Proof by Induction

$0 < R_1 \leq 1 - \epsilon$	$\delta_k(R_1) = R_1$	$0 < R_2 \leq 1 - \epsilon$	$\delta_k(R_2) = R_2$	$L_k(i) = R_1 R_2$
$0 < R_1 \leq 1 - \epsilon$	$\delta_k(R_1) = R_1$	$1 - \epsilon < R_2 < 1 + \epsilon$	$\delta_k(R_2) = 1$	$L_k(i) = R_1$
$0 < R_1 \leq 1 - \epsilon$	$\delta_k(R_1) = R_1$	$1 + \epsilon \leq R_2 \leq \frac{1 + \epsilon}{\delta_k(R_1)}$	$\delta_k(R_2) = R_2$	$L_k(i) = R_1 R_2$
$0 < R_1 \leq 1 - \epsilon$	$\delta_k(R_1) = R_1$	$R_2 > \frac{1 + \epsilon}{\delta_k(R_1)}$	$\delta_k(R_2) = \frac{1}{R_1}$	$L_k(i) = 1$
$1 - \epsilon < R_1 < 1 + \epsilon$	$\delta_k(R_1) = 1$	$0 < R_2 \leq 1 - \epsilon$	$\delta_k(R_2) = R_2$	$L_k(i) = R_2$
$1 - \epsilon < R_1 < 1 + \epsilon$	$\delta_k(R_1) = 1$	$1 - \epsilon < R_2 < 1 + \epsilon$	$\delta_k(R_2) = 1$	$L_k(i) = 1$
$1 - \epsilon < R_1 < 1 + \epsilon$	$\delta_k(R_1) = 1$	$1 + \epsilon \leq R_2 \leq \frac{1 + \epsilon}{\delta_k(R_1)}$	$\delta_k(R_2) = R_2$	$L_k(i) = R_2$
$1 - \epsilon < R_1 < 1 + \epsilon$	$\delta_k(R_1) = 1$	$R_2 > \frac{1 + \epsilon}{\delta_k(R_1)}$	$\delta_k(R_2) = \frac{1}{1}$	$L_k(i) = 1$
$R_1 = 1 + \epsilon$	$\delta_k(R_1) = R_1$	$0 < R_2 \leq 1 - \epsilon$	$\delta_k(R_2) = R_2$	$L_k(i) = R_1 R_2$
$R_1 = 1 + \epsilon$	$\delta_k(R_1) = R_1$	$1 - \epsilon < R_2 < 1 + \epsilon$	$\delta_k(R_2) = 1$	$L_k(i) = R_1$
$R_1 = 1 + \epsilon$	$\delta_k(R_1) = R_1$	$1 + \epsilon \leq R_2 \leq \frac{1 + \epsilon}{\delta_k(R_1)}$	$\delta_k(R_2) = R_2$	$L_k(i) = R_1 R_2$
$R_1 = 1 + \epsilon$	$\delta_k(R_1) = R_1$	$R_2 > \frac{1 + \epsilon}{\delta_k(R_1)}$	$\delta_k(R_2) = \frac{1}{R_1}$	$L_k(i) = 1$
$R_1 > 1 + \epsilon$	$\delta_k(R_1) = 1$	$0 < R_2 \leq 1 - \epsilon$	$\delta_k(R_2) = R_2$	$L_k(i) = R_2$
$R_1 > 1 + \epsilon$	$\delta_k(R_1) = 1$	$1 - \epsilon < R_2 < 1 + \epsilon$	$\delta_k(R_2) = 1$	$L_k(i) = 1$
$R_1 > 1 + \epsilon$	$\delta_k(R_1) = 1$	$1 + \epsilon \leq R_2 \leq \frac{1 + \epsilon}{\delta_k(R_1)}$	$\delta_k(R_2) = R_2$	$L_k(i) = R_2$
$R_1 > 1 + \epsilon$	$\delta_k(R_1) = 1$	$R_2 > \frac{1 + \epsilon}{\delta_k(R_1)}$	$\delta_k(R_2) = \frac{1}{1}$	$L_k(i) = 1$

Table 2. Last Step in the Proof by Induction

$0 < R_{n_k-1} \leq 1 - \epsilon$	$\delta_k(R_{n_k-1}) = R_{n_k-1}$	$L_k(i) _{n_k} = L_k(i) _{n_k-1} R_{n_k-1}$
$1 - \epsilon < R_{n_k-1} < 1 + \epsilon$	$\delta_k(R_{n_k-1}) = 1$	$L_k(i) _{n_k} = L_k(i) _{n_k-1}$
$1 + \epsilon \leq R_{n_k-1} \leq \frac{1 + \epsilon}{\prod_{m_k=0}^{n_k-2} \delta_k(R_{m_k})}$	$\delta_k(R_{n_k-1}) = R_{n_k-1}$	$L_k(i) _{n_k} = L_k(i) _{n_k-1} R_{n_k-1}$
$R_{n_k-1} > \frac{1 + \epsilon}{\prod_{m_k=0}^{n_k-2} \delta_k(R_{m_k})}$	$\delta_k(R_{n_k-1}) = \frac{1}{\prod_{m_k=0}^{n_k-2} \delta_k(R_{m_k})}$	$L_k(i) _{n_k} = \frac{\prod_{m_k=0}^{n_k-2} \delta_k(R_{m_k})}{\prod_{m_k=0}^{n_k-2} \delta_k(R_{m_k})} = 1$

• $n_k: L_k(i)|_{n_k} = \prod_{t_k=1}^{n_k-1} \delta_k(R_{t_k}) = L_k(i)|_{n_k-1} \delta_k(R_{n_k-1})$. We must analyze four distinct options: see Table 2. Even in this case we recover the correct contributions. Q.E.D.

E. Logarithmic Formulation

We have proved that Eq. (3) describes the function computed by the basic Retinex algorithm. Now we want to rewrite formula (3) in the so-called logarithmic formulation. Most of the implementations of Retinex in fact use this formulation instead of the original one. The reason lies in the fact that the logarithmic function transforms products into sums and divisions into subtraction, hence reducing the computational cost of the algorithm.

To construct the logarithmic formulation notice that $L_k(i) > 0$ for every $k = 1, \dots, N$; thus it makes perfect sense to apply the identity function on \mathbb{R}^+ decomposed as $\exp \circ \log$ to $L_k(i)$, and what we get is another way of writing formula (3):

$$L(i) = \frac{1}{N} \sum_{k=1}^N [\exp \circ \log(L_k(i))] = \frac{1}{N} \sum_{k=1}^N (\exp \circ \log) \left(\prod_{t_k=1}^{n_k-1} \delta_k(R_{t_k}) \right). \tag{6}$$

The usefulness of this manipulation lies in the fact that

the properties of the logarithmic function enable us to write

$$L(i) = \frac{1}{N} \sum_{k=1}^N \exp\left(\sum_{t_k=1}^{n_k-1} \log[\delta_k(R_{t_k})]\right). \tag{7}$$

We can simplify the formula by defining $\tilde{I}(x_{t_k}) = \log(I(x_{t_k}))$, $\tilde{R}_{t_k} = \log R_{t_k} = \tilde{I}(x_{t_{k+1}}) - \tilde{I}(x_{t_k})$, $\tilde{R}_0 = 0$. We define the functions δ_k to be $\tilde{\delta}_k(\tilde{R}_0) = 0$ and, for $t_k = 1, \dots, n_k - 1$,

$$\tilde{\delta}_k(\tilde{R}_{t_k}) = \begin{cases} \tilde{R}_{t_k} & \text{if } -\infty < \tilde{R}_{t_k} \leq -\tilde{\epsilon} \\ 0 & \text{if } -\tilde{\epsilon} < \tilde{R}_{t_k} < \tilde{\epsilon} \\ \tilde{R}_{t_k} & \text{if } \tilde{\epsilon} \leq \tilde{R}_{t_k} \leq \tilde{\epsilon} - \sum_{m_k=0}^{t_k-1} \tilde{\delta}_k(\tilde{R}_{m_k}) \\ -\sum_{m_k=0}^{t_k-1} \tilde{\delta}_k(\tilde{R}_{m_k}) & \text{if } \tilde{R}_{t_k} > \tilde{\epsilon} - \sum_{m_k=0}^{t_k-1} \tilde{\delta}_k(\tilde{R}_{m_k}) \end{cases}, \tag{8}$$

with $\tilde{\epsilon} = \log(1 + \epsilon)$. With these definitions the formula for the lightness can be written as

$$L(i) = \frac{1}{N} \sum_{k=1}^N \exp\left(\sum_{t_k=1}^{n_k-1} \tilde{\delta}_k(\tilde{R}_{t_k})\right). \tag{9}$$

This formula contains only sums and differences, and for small values of ϵ it coincides with the nonlogarithmic formulation. The price to pay is that we have to take the logarithms of the intensity of every pixel and we have to exponentiate, but these operations are done only once per path.

3. MATHEMATICAL ANALYSIS OF THE FORMULA

Now that we have a formula to describe Retinex, we would expect to make predictions about the model. Unfortunately the presence of the threshold makes this task hard to realize. In fact, to make predictions about the behavior of Retinex, we must know how many ratios are below threshold, but this is impossible to know unless we know the image and the topology of the paths.

What we are going to show now is that if we disregard the threshold, then with the simplified version of formula (3) we are able to make predictions about the qualitative intrinsic behavior of Retinex and also about its behavior in relation to its parameters.

A. Simplification of the Formula for $\epsilon = 0$

We want to analyze formula (3) when $\epsilon = 0$. When $\epsilon = 0$, the definition of the functions δ_k becomes much simpler:

$$\delta_k(R_{t_k}) = \begin{cases} R_{t_k} & \text{if } 0 < R_{t_k} \prod_{m_k=0}^{t_k-1} \delta_k(R_{m_k}) \leq 1 \\ 1 & \text{if } R_{t_k} \prod_{m_k=0}^{t_k-1} \delta_k(R_{m_k}) > 1 \end{cases}; \tag{10}$$

hence when the threshold is 0, δ_k behaves like the identity function or like the reset function.

To avoid a cumbersome notation we can eliminate the subscript k from the discussion by fixing attention on a given path γ starting in $\gamma(1) = j$ and ending in $\gamma(n) = i$. Let H be the value of the parameter of γ such that $\gamma(H) = x_H$ is the pixel with highest intensity in the whole path.

If we write the contribution of γ explicitly as

$$\delta\left(\frac{I(x_2)}{I(j)}\right) \dots \delta\left(\frac{I(x_H)}{I(x_{H-1})}\right) \delta\left(\frac{I(x_{H+1})}{I(x_H)}\right) \dots \delta\left(\frac{I(i)}{I(x_{n-1})}\right), \tag{11}$$

we can notice that the pixel x_H enables the reset mechanism. Suppose in fact that no reset occurs before x_H ; then we have

$$\frac{I(x_2) I(x_3)}{I(j) I(x_2)} \dots \delta\left(\frac{I(x_H)}{I(x_{H-1})}\right) \delta\left(\frac{I(x_{H+1})}{I(x_H)}\right) \dots \delta\left(\frac{I(i)}{I(x_{n-1})}\right), \tag{12}$$

and one can see immediately that the first ratios cancel each other to give

$$\frac{I(x_{H-1})}{I(j)} \delta\left(\frac{I(x_H)}{I(x_{H-1})}\right) \delta\left(\frac{I(x_{H+1})}{I(x_H)}\right) \dots \delta\left(\frac{I(i)}{I(x_{n-1})}\right). \tag{13}$$

Now, the ratio $I(x_H)/I(x_{H-1})$ is surely greater than 1 because, by hypothesis, x_H is the pixel with highest intensity in γ ; moreover, thanks to the cancellations, the product of this ratio and the previous ones reduces to

$$\frac{I(x_{H-1})}{I(j)} \frac{I(x_H)}{I(x_{H-1})} = \frac{I(x_H)}{I(j)}, \tag{14}$$

and even this ratio is greater than 1, for the same reason as above.

Thus the reset mechanism is enabled, and the chain of products reduces to

$$\delta\left(\frac{I(x_{H+1})}{I(x_H)}\right) \dots \delta\left(\frac{I(i)}{I(x_{n-1})}\right). \tag{15}$$

This conclusion remains true even if the reset mechanism is activated by some other pixel before x_H in the path γ . This can be seen by replacing j with the pixel that has produced the last reset before x_H and then using the same arguments as before.

The consequence is that when $\epsilon = 0$ all the pixels traveled by the path γ before the pixel with highest intensity are perfectly uninfluential for the computation of $L(i)$.

After the pixel x_H the reset mechanism is inhibited and the δ function reduces simply to the identity function. This statement can be proved as follows: If all the ratios

remain less than 1 until the end of the path, then the statement is trivially true; if, instead, there exists a pixel x_K , $K > H$, such that $I(x_{K+1})/I(x_K) > 1$, the reset mechanism cannot be enabled because the product of ratios $R_1 \dots R_K$ reduces to

$$\frac{I(x_{H+1})I(x_{H+2})}{I(x_H)I(x_{H+1})} \dots \frac{I(x_K)I(x_{K+1})}{I(x_{K-1})I(x_K)} = \frac{I(x_{K+1})}{I(x_H)} < 1. \quad (16)$$

If there are other pixels with the same characteristic as x_K , the conclusion is unchanged. If there is more than one pixel with the same intensity as x_H , then all the considerations above have to be referred to the last pixel with highest intensity traveled by γ . All these considerations show that the contribution of the path reduces simply to $I(i)/I(x_H)$.

Since the arguments just presented can be used for every path, we have that their contributions are

$$L_k(i) = \frac{I(i)}{I(x_{H_k})}, \quad (17)$$

where x_{H_k} is the pixel with the highest intensity traveled by γ_k for every $k=1, \dots, N$. Thus formula (3) can be written as follows:

$$L(i) = \frac{1}{N} \sum_{k=1}^N \frac{I(i)}{I(x_{H_k})}, \quad (18)$$

or

$$L(i) = \frac{1}{N} \sum_{k=1}^N \exp(\tilde{I}(i) - \tilde{I}(x_{H_k})) \quad (19)$$

in the logarithmic formulation.

Thanks to the fact that $I(i)$ is independent of k , i being the end point of all paths, formula (18) can be written in this alternative form:

$$L(i) = I(i) \frac{1}{N} \sum_{k=1}^N \frac{1}{I(x_{H_k})}, \quad (20)$$

which shows explicitly that at a mathematical level, Retinex without threshold acts on the intensity of each pixel as a multiplication operator, with the multiplicative factor given by the average of the inverse values of the highest intensities of the pixels traveled by the path γ_k .

The similarity to the von Kries algorithm is evident, the big difference between the two models being that in our model the inverses of the maximum values of intensity are taken not by scanning the whole image but by scanning only the pixels traveled by the paths.

This situation makes evident in a clear way the strong image dependency and path dependency of the algorithm: To different images and different paths there correspond, in general, different values of $I(x_{H_k})$ and hence different values of the lightness $L(i)$. Also, the denser $\gamma_1, \dots, \gamma_N$ are in the neighborhood of i , the more Retinex shows a local behavior, because the probability of finding the pixels x_{H_k} near i is greater.

Finally, we recall that the intensity values are normalized, so $0 < I(x_{H_k}) \leq 1$ for every $k=1, \dots, N$, and then $\sum_{k=1}^N 1/I(x_{H_k}) \geq N$. It follows that $L(i) \geq I(i)$ for every pixel i ;

this is a rigorous proof of the fact that *an image filtered with Retinex without threshold is always brighter than or equal to the original one*.

B. Difference between Retinex with and without Threshold

We recall that the role of the threshold is to decrease the effect of low channel intensity ratios, which correspond to smooth shadings in the image. Now we want to compute the highest difference between the values of lightness given by Retinex with and without threshold, which obviously corresponds to the situation in which the paths travel entirely a uniform gradient below threshold.

Let us consider, for example, the situation in which there is a continuously decreasing gradient traveled by a path γ_k such that the ratio of intensities at each junction is less than or equal to $1 - \epsilon$.

The contribution of the algorithm without threshold is $I(i)/I(j_k)$, whereas the algorithm with threshold gives 1. The highest difference between 1 and $I(i)/I(j_k)$ is $1 - (1 - \epsilon)^{n_k - 1}$. This can be easily shown considering the fact that since the gradient is decreasing, the highest difference is obtained by setting every ratio to $1 - \epsilon$. From the first ratio, $I(x_2)/I(j_k) = 1 - \epsilon$, one has $I(x_2) = (1 - \epsilon)I(j_k)$; hence $I(x_3)/I(x_2) = 1 - \epsilon$ implies that $I(x_3)/[(1 - \epsilon)I(j_k)] = 1 - \epsilon$, i.e., $I(x_3) = (1 - \epsilon)^2 I(j_k)$. Iterating these substitutions, one gets $I(i) = (1 - \epsilon)^{n_k - 1} I(j_k)$, and so $1 - I(i)/I(j_k) = 1 - (1 - \epsilon)^{n_k - 1}$. Analogous considerations can be written for increasing gradients.

Figure 1(a) shows the gradient of gray levels between 50 and 150; Fig. 1(b) represents its filtered version by Retinex without threshold. The gradient filtered by Retinex with threshold gives a white image, and the difference between it and Fig. 1(b) is consistent with the predictions above.

However, in real images, the probability that such a gradient is entirely traversed by a path of generic geometry is very small. For example, the usual threshold for an image of size 320×240 is $\epsilon = 0.05$, and with such a choice



(a)



(b)

Fig. 1. (a) Gradient of gray levels between 50 and 150. (b) gradient filtered by Retinex without threshold.

a path should cross 5120 pixels consecutively in the gradient to pass from intensity 0 to intensity 255.

If the path travels alternately small decreasing and increasing gradients, then the differences between Retinex with and without threshold tend to compensate each other, and the global difference reduces to a small value.

Moreover, since the lightness $L(i)$ is obtained by the average of the contributions $L_k(i)$ of N paths, the global difference between the two algorithms will be smaller than (or, at least, equal to) the differences due to the contributions of the paths that travel the uniform gradient.

Tests on images are in accordance with these considerations: The rms difference, averaged in the three chromatic channels, between the images of our test set filtered with the two algorithms is $\sim 2\%$. The chosen test set is composed of 31 images representing various subjects (e.g., portraits, landscapes, graphic images).

4. CONSEQUENCES OF THE FORMULA FOR RETINEX WITHOUT THRESHOLD

Here we show how the formula for Retinex without threshold can be used to predict what happens when the algorithm is iteratively applied on an image and when the number of paths grows to “very big values” (in a sense that will be specified later).

A. Convergence of Multiple Iterations of Retinex without Threshold

With the help of formula (18) we can prove that if an image is filtered many times with Retinex without threshold, then it converges to an image characterizable in a simple way.

Mathematically, the operation of applying m -times Retinex corresponds to the composition of the lightness function L with itself m times. To write down this composition for the simplest case of $m=2$, we note that the first application of Retinex modifies the intensity $I(i)$ of every pixel i into $L(i)$; hence

$$L^2(i) = (L \circ L)(i) = \frac{1}{N} \sum_{k=1}^N \frac{L(i)}{L(x_{H'_k})}, \quad (21)$$

where $x_{H'_k}$ is the pixel with highest value of *lightness* in a given chromatic channel along the path γ_k after the first application of Retinex. We stress that $x_{H'_k}$ is in general different from x_{H_k} , which is the pixel with highest *intensity* along γ_k before the first filtering operation.

$L(i)$ is independent from the index k and can be extracted from the sum and divided from both members to give

$$\frac{L^2(i)}{L(i)} = \frac{1}{N} \sum_{k=1}^N \frac{1}{L(x_{H'_k})}. \quad (22)$$

More generally, if we write with L^m the composition of L with itself m times, one can see immediately that



(a)



(b)

Fig. 2. (Color online) (a) Gallery original image. (b) Book original image.

$$\frac{L^m(i)}{L^{m-1}(i)} = \frac{1}{N} \sum_{k=1}^N \frac{1}{L^{m-1}(x_{H_k^{m-1}})}, \quad (23)$$

where $x_{H_k^{m-1}}$ is the pixel with highest value of lightness along the path γ_k after $m-1$ application of Retinex to the original image.

Equation (23) shows that every iteration of Retinex increases the lightness of the generic pixel i or leaves it unchanged. In fact, $0 < L^{m-1}(x_{H_k^{m-1}}) \leq 1$, $k=1, \dots, N$, and hence the right-hand side of Eq. (23) is ≥ 1 and so $L^m(i) \geq L^{m-1}(i)$ for every pixel i .

In particular, the lightness of i is forced to grow until the right-hand side of Eq. (23) reaches the value 1, but this happens if and only if $L^{m-1}(x_{H_k^{m-1}}) = 1$ or, in other words, when the lightness of (at least) one pixel in all the paths $\gamma_1, \dots, \gamma_N$ reaches the value 1. We call the image with this characteristic the “image of convergence,” since every further iteration of Retinex produces the same image.

To test our results, we considered the RGB rms distance between subsequent iterations of Retinex without threshold applied on two original images: Figs. 2(a) and 2(b). We performed tests using the same paths for every iteration and also changing the paths randomly: Figs. 3(a) and 3(b). As expected, the speed of convergence to the final image is higher when the paths are kept fixed. All the images of our test set show this decreasing behavior, but the geometry of the curve is different for each image.

Finally, we recall that the value of the lightness $L(i)$ is referred to a single chromatic channel $c \in \{R, G, B\}$, and, in general, the pixels in the paths $\gamma_1, \dots, \gamma_N$ with value 1 will be different for R , G , and B . If these pixels have the same coordinates, then by superposition they produce isolated white pixels in every path in the image of convergence. This holds true for Retinex without threshold, in

accordance with Eq. (23). The result is confirmed by our tests, as can be observed in Figs. 4(a) and 4(b).

The analysis of the convergence of multiple iterations of Retinex with threshold needs further investigation and remains an open problem.

B. Behavior of Retinex without Threshold in Relation to the Number of Paths

Here we analyze what happens when $N \rightarrow \infty$, where N is the number of the paths that scan the image.

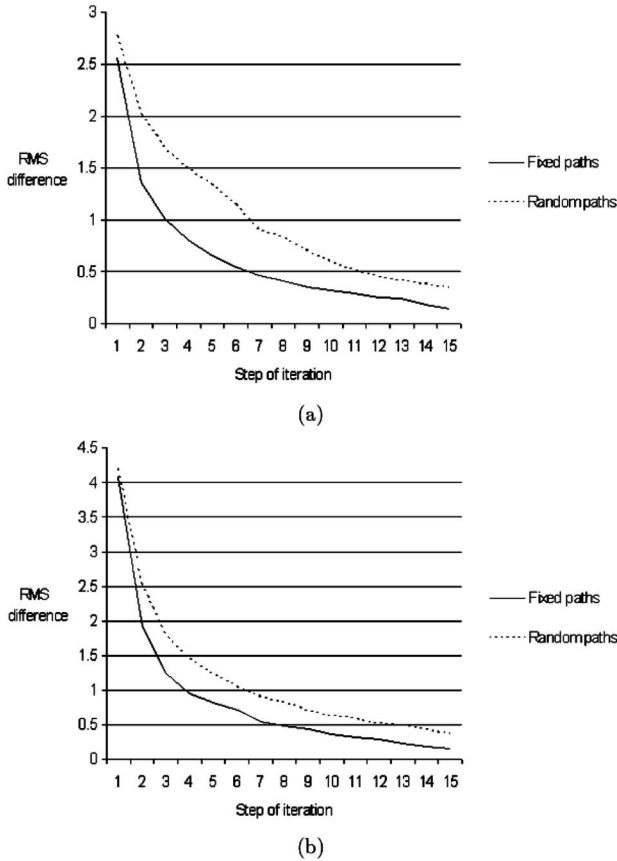


Fig. 3. (a) Gallery iteratively filtered. (b) Book iteratively filtered.

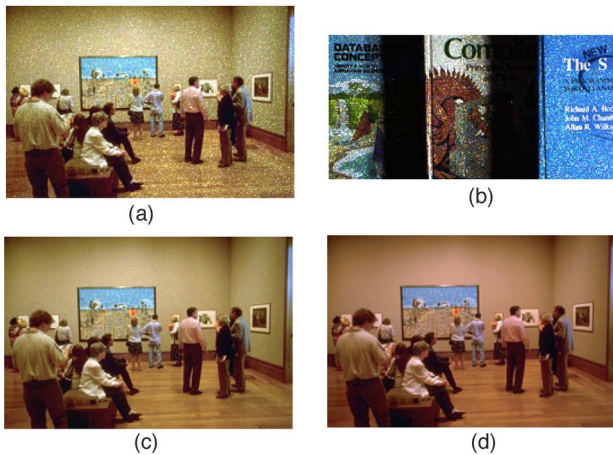


Fig. 4. (Color online) (a) Gallery image of convergence. (b) Book image of convergence. (c) Gallery filtered using paths with 2^4 nodes. (d) Gallery filtered using paths with 2^{10} nodes.

Mathematically speaking, we want to study the limit condition,

$$L_\infty(i) = I(i) \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{k=1}^N \frac{1}{I(x_{H_k})}. \quad (24)$$

This is trivially analyzed for images composed of a single color. For such images $I(x_{H_k})=I(i)$ for every k , and so $L_\infty(i)=I(i)/I(i)\lim_{N \rightarrow \infty} 1/N \sum_{k=1}^N 1 = \lim_{N \rightarrow \infty} N/N = 1$.

Let us analyze the limit condition for the more interesting situation of nonmonochromatic images. First of all notice that, *a priori*, we cannot say anything about the behavior of the argument of the limit for $N \rightarrow \infty$, because it depends on the image we are dealing with. For this reason we are forced to analyze the problem by making reasonable assumptions on *how* the number of the paths grows to infinity.

To this end, note that it is not of interest to travel a path more than once; in fact, the contribution $L_k(i)$ of a path γ_k is completely determined by the pixel with highest intensity, and this is independent of how many times it is traveled.

For this reason we choose to consider only paths traveled once and with isolated self-intersections that join a generic pixel j to the target i . Moreover, we order the paths in relation to the number of pixels they travel.

These restrictions on the topology and the ordering of the paths imply that as N grows, the paths explore bigger and bigger portions of the image before reaching the target i . Consequently the probability of traveling the pixel with the highest intensity in the whole image grows as N grows. We denote this pixel by \bar{x} .

It follows that there exists a certain number N_0 such that the paths labeled by an index $N \geq N_0$ will give the contribution $L_k(i)=I(i)/I(\bar{x})$ to the computation of $L_\infty(i)$, and so

$$L_\infty(i) = I(i) \lim_{N \rightarrow \infty} \frac{1}{N} \left(\frac{1}{I(x_{H_1})} + \frac{1}{I(x_{H_2})} + \dots + \frac{1}{I(x_{H_{N_0-1}})} + \frac{N - N_0}{I(\bar{x})} \right). \quad (25)$$

The limit for $N \rightarrow \infty$ is convergent to $1/I(\bar{x})$, and hence

$$L_\infty(i) = I(i)/I(\bar{x}). \quad (26)$$

This formula implies that the algorithm, in the limit $N \rightarrow \infty$, loses its local properties and reduces to a global white-patch model, showing the relative weakness.

Figures 4(c) and 4(d) show the difference in the ability of Retinex to remove the chromatic dominant using paths with 2^4 nodes [Fig. 4(c)] and 2^{10} nodes [Fig. 4(d)]. The rms distance between the image filtered with a von Kries algorithm and the Fig. 4(d) image is 0.

In particular, if $I(\bar{x})=1$ for every chromatic channel, then $L_\infty(i)=I(i)$; i.e., if there is a white pixel in the image, then the algorithm in the limit condition reproduces the original image.

5. CONCLUSIONS AND PERSPECTIVES

In this paper we proposed an analytic formula to describe the Retinex algorithm mathematically. The background in which the formula lies is independent of the topology of the paths used by the algorithm to scan the image, and hence the formula can be used for all the implementations of the basic Retinex that differ from the geometry of the paths used.

We developed a rigorous proof of the proposition that the formula mathematically represents Retinex, and then we analyzed its consequences. This analysis differs from the one performed in Ref. 3 because we have used different mathematical techniques and have also considered the reset mechanism.

The formula for Retinex with threshold is not predictive, in the sense that we are not able to say anything about the qualitative behavior of the algorithm without complete knowledge of the details of the image and the topology of the paths. However, if the threshold is eliminated, then the formula is consistently simplified, and this leads to the possibility of making qualitative predictions about the model.

Before the analysis of the consequences of the simplified formula, we quantified the greatest possible difference between Retinex with and without threshold. We showed that the only situation in which we expect an appreciable difference is when there are large areas of an image characterized by small continuously decreasing (or continuously increasing) gradients. In all the other cases we expect the two algorithms to have very similar behavior. Our test results agree with this expectation: We filtered over 30 different real-world images with the two algorithms and we found that the average rms difference is $\sim 2\%$.

The first consequence of the formula for Retinex without threshold is that the algorithm is intrinsically image and path dependent. This results from the fact that the algorithm acts on the intensity of each pixel simply as a multiplication operator, where the multiplicative factor is the average of the inverse values of the highest intensities of the pixels traveled by every single path; hence, varying the image or the paths causes the factor (and so the lightness of the pixel) to change.

Since the HVS is an efficient information stabilizer that is able to extract *constant* information from a visual scene regardless of illumination conditions, an interesting issue to investigate is the behavior of the model when applied iteratively on its own results. This point, discussed previously, as far as we know, only experimentally, is here treated from a mathematical point of view. The analysis confirmed that multiple iterations of Retinex converge to a limit image that admits a full characterization: Every path ending in a fixed target pixel has traveled, in each chromatic channel, at least one pixel with a maximum value. This is related to the white-patch behavior of Retinex.

An important parameter of the model, and consequently of the formula, is the number of paths N . The local properties of Retinex are strongly related to this parameter. Previous studies showed that, initially, increasing the number of paths leads to more accurate results, but this is not true for *every* value of N ;

the parameter N remains a key factor in the algorithm tuning.¹⁸

Our mathematical formulation shows that as the number of paths grows, Retinex tends to a von Kries algorithm, exhibiting only a *global* white-patch behavior. Confirming the experimental results, this behavior means that the tuning problem is still open.

The topology of paths has not been considered in this work and remains a major topic to be investigated in future studies.

The authors' e-mail addresses are provenzi@dti.unimi.it, decarli@dti.unimi.it, rizzi@dti.unimi.it, and daniele.marini@unimi.it

REFERENCES

1. E. Land and J. J. McCann, "Lightness and Retinex theory," *J. Opt. Soc. Am.* **61**, 1–11 (1971).
2. J. J. McCann, S. P. McKee, and T. H. Taylor, "Quantitative studies in Retinex theory. A comparison between theoretical predictions and observer responses to the 'color mondrian' experiments," *Vision Res.* **16**, 445–458 (1976).
3. D. H. Brainard and B. A. Wandell, "Analysis of the Retinex theory of color vision," *J. Opt. Soc. Am. A* **3**, 1651–1661 (1986).
4. R. A. Young, "Color vision and the Retinex theory," *Science* **238**, 1731–1732 (1987).
5. A. Moore, J. Allman, and R. Goodman, "A real-time neural system for color constancy," *IEEE Trans. Neural Netw.* **2**, 237–247 (1991).
6. 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. Image Process.* **6**, 965–976 (1997).
7. D. J. Jobson, Z. Rahman, and G. A. Woodell, "Properties and performance of a center/surround retinex," *IEEE Trans. Image Process.* **6**, 451–462 (1997).
8. K. Barnard and B. Funt, "Investigations into multi-scale Retinex," *Vision and Technology* (Wiley, 1999), pp. 9–17.
9. S. O. Huck, C. L. Fales, R. E. Davis, and R. Alter-Gartenberg, "Visual communication with Retinex coding," *Appl. Opt.* **39**, 1711–1730 (2000).
10. D. Marini and A. Rizzi, "A computational approach to color adaptation effects," *Image Vis. Comput.* **18**, 1005–1014 (2000).
11. J. D. Cowan and P. C. Bressloff, "Visual cortex and the Retinex algorithm," in *Human Vision and Electronic Imaging VII* B. E. Rogowitz and N. T. Pappas, eds., *Proc. SPIE* **4662**, 278–285 (2002).
12. G. D. Finlayson, S. D. Hordley, and M. S. Drew, "Removing shadows from images using retinex," in *Proceedings of the IS&T/SID Tenth Color Image Conference: Color Science and Engineering System Technology* (Society for Information Display, 2002), pp. 73–79.
13. A. C. Hurlbert and C. J. Wolf, "Contribution of local and global cone-contrasts to color appearance: a Retinex-like model," in *Human Vision and Electronic Imaging VII*, B. E. Rogowitz and N. T. Pappas, eds., *Proc. SPIE* **4662**, 286–297 (2002).
14. M. Pilu and S. Pollard, "A light-weight text image processing method for handheld embedded cameras," in *Proceedings of the British Machine Vision Conference*, (British Machine Vision Association, <http://www.cmp.uea.uk/research/bmvc/>; School of Information Systems, University of East Anglia, Norwich, UK, 2002), pp. 547–556.
15. A. Rizzi, D. Marini, and L. De Carli, "LUT and multilevel Brownian Retinex colour correction," *Mach. Graphics Vision* **11**(2/3), 153–168 (2002).

16. R. Kimmel and M. Elad, "A variational framework for Retinex," *Int. J. Comput. Vis.* **52**, 7–23 (2003).
17. G. Ramponi, L. Tenze, S. Carrato, and S. Marsi, "Nonlinear contrast enhancement based on the Retinex approach," in *Image Processing: Algorithms and Systems II*, E. Dougherty, J. T. Astola, and K. O. Egiazarian, eds., Proc. SPIE **5014**, 169–177 (2003).
18. F. Ciurea and B. Funt, "Tuning Retinex parameters," *J. Electron. Imaging* **13**, 58–64 (2004).
19. T. J. Cooper and F. A. Baqai, "Analysis and extensions of the Frankle–McCann Retinex algorithm," *J. Electron. Imaging* **13**, 85–92 (2004).
20. B. Funt, F. Ciurea, and J. J. McCann, "Retinex in Matlab," *J. Electron. Imaging* **13**, 48–57 (2004).
21. J. J. McCann, "Capturing a black cat in shade: past and present of Retinex color appearance models," *J. Electron. Imaging* **13**, 36–47 (2004).
22. L. Meylan and S. E. Süsstrunk, "Bio-inspired image enhancement for natural color images," in *Human Vision and Electronic Imaging IX*, B. E. Rogowitz and T. N. Pappas, eds., Proc. SPIE **5292**, 46–56 (2004).
23. N. Moroney and I. Tastl, "Comparison of Retinex and iCAM for scene rendering," *J. Electron. Imaging* **13**, 139–145 (2004).
24. Z. Rahman, D. J. Jobson, and G. A. Woodell, "Retinex processing for automatic image enhancement," *J. Electron. Imaging* **13**, 100–110 (2004).
25. H. K. Rising III, "Analysis and generalization of Retinex by recasting the algorithm in wavelets," *J. Electron. Imaging* **13**, 93–99 (2004).
26. A. Rizzi, C. Gatta, B. Piacentini, M. Fierro, and D. Marini, "Human-visual-system-inspired tone mapping algorithm for HDR images," in *Human Vision Electronic Imaging IX*, B. E. Rogowitz and T. N. Pappas, eds., Proc. SPIE **5292**, 57–68 (2004).
27. A. Rizzi, C. Gatta, and D. Marini, "From Retinex to automatic color equalization: issues in developing a new algorithm for unsupervised color equalization," *J. Electron. Imaging* **13**, 75–84 (2004).
28. R. Sobol, "Improving the Retinex algorithm for rendering wide dynamic range photographs," *J. Electron. Imaging* **13**, 65–74 (2004).
29. E. Land, "The Retinex theory of color vision," *Sci. Am.* March 1977, pp. 2–17.
30. E. Land, "Recent advances in retinex theory and some implications for cortical computations: color vision and the natural image," *Proc. Natl. Acad. Sci. U.S.A.* **80**, 5163–5169 (1983).
31. J. Frankle and J. J. McCann, "Method and apparatus for lightness imaging," U.S. Patent 4,348,336 May 17, 1983.
32. E. Land, "An alternative technique for the computation of the designator in the retinex theory of color vision," *Proc. Natl. Acad. Sci. U.S.A.* **83**, 3078–3080 (1986).
33. J. J. McCann and 25 other authors, "Special session on Retinex at 40," *J. Electron. Imaging* **13**, 6–145 (2004).