RESEARCH ARTICLE   OPEN ACCESS

Random Spray Retinex: A New Retinex Implementation to Investigate the Local Properties of the Model

A.P.Choudhary,

Principal, SVVP,Borivali,Mumbai. E-mail pdchoudhary@rcfltd.com

ABSTRACT:
In order to investigate the local filtering behavior of the Retinex model, we propose a new implementation in which paths are replaced by 2-D pixel sprays, hence the name "random spray Retinex." A peculiar feature of this implementation is the way its parameters can be controlled to perform spatial investiga-tion. The parameters' tuning is accomplished by an unsupervised method based on quantitative measures. This procedure has been validated via user panel tests. Furthermore, the spray approach has faster performances than the path-wise one. Tests and results are presented and discussed.

Index Terms: Locality of color perception, pixel sprays, Retinex.

I. INTRODUCTION

THE human visual system (HVS) does not perceive the color of an area independently from the visual scene in which it lies; instead, it is heavily influenced by the chromatic content of the other areas of the scene. This psychophysio-logical phenomenon is the *locality of color perception*. One of the earliest models able to deal with locality of perception has been Retinex by Land and McCann [1]. The scientific community has always been interested in this model and its

various applications, as reported in [2].

In the basic Land and McCann implementation of Retinex, locality is achieved through paths scanning images.

A great amount of implementations and analysis followed after this first work. These can be divided into two major groups, differing from the way they achieve locality: *sampling* the chro-matic content around a pixel [3]–[7] or *integrating* it [8]–[15].

All the sampling implementations that use a path-wise ap-proach have to deal with the following problems: strong depen-dency on paths geometry, high computational cost, and sam-pling noise.

On the basis of a recent mathematical analysis of path-wise Retinex algorithms [16], we will prove the intrinsic redundancy of this approach. Consequently, we will propose an extension that allows to keep the sampling approach, highly reducing the problems related to the use of paths.

This alternative technique is constructed replacing paths with random sprays, i.e., 2-D point distributions across the image, so the name "random spray Retinex" (RSR). We will show how it is possible to change the spray density around a pixel and how this leads to the ability of finding out information about locality of color perception within the Retinex model.

The structure of the paper is the following. In Section II, we review the basic information contained in [16] about the mathe-matical formulation of path-wise Retinex implementations and its consequences on the intrinsic properties of the model. In Sec-tion III, we motivate the passage from paths to sprays used in Section IV to implement RSR and discuss its properties. Finally, Section V is dedicated to the tuning of RSR parameters.

II. MATHEMATICAL DESCRIPTION OF RETINEX

The starting point of our analysis is the mathematical description of Retinex given in [16] and briefly recalled here. Consider a RGB digital image and a collection of N paths $\gamma_1...\gamma_N$, composed by ordered chains of pixels starting in j_k and ending in i (called *target pixel*). Let n_k be the number of pixels traveled by the kth path γ_k and let $t_k = 1, \ldots, n_k$ be its parameter, i.e., $\gamma_k : \{1, \ldots, n_k\} \to$ Image $\subset$ R^2 ; $\gamma_k(1) = j_k$ and $\gamma_k(n_k) = i$. We indicate 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 $t_k = 1, \ldots, n_k - 1$. In Retinex, the computations are performed separately in every chromatic channel $c \in \{R, G, B\}$, to avoid a cumbersome notation we omit its specification. So, we write $I(x_{t_k})$ $I(x_{t_k+1})$ to indicate the intensities of x_{t_k} and x_{t_k+1} in the three separated chromatic channels. We write their ratio as $R_{t_k} = (I(x_{t_k+1})/I(x_{t_k}))$, with $R_0 \equiv 1$. For technical reasons, it is useful to normalize the intensities; to take their values in the real unit interval $[0, 1]$, we omit the zero value because we must perform divisions. The usual way to do this is to add a small positive constant to all pixel intensities.

In [16], it has been proved that the intensity of i is recomputed by Retinex to give the (normalized) *lightness* $L(i)$ through the function that follows:

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

where $\delta_k : $ R$^+ \to$ R$^+$ $k = 1, \ldots, N$, are functions defined in this way: $\delta_k(R_0) = 1$ and, for $t_k = 1, \ldots, n_k - 1$, see equation (2), shown at the bottom of the page, $\varepsilon > 0$ being a fixed *threshold*.

When the first or the third option are satisfied, δ_k acts simply as the identity function and the formula implements the typical "chain of ratios" of Retinex [1].

The second option occurs when only a very small change of intensity is measured between two subsequent pixels. In this case, $\delta_k(R_t)=1$ so that the product of ratios remains unchanged with respect to the previous step. This implements in a mathematical fashion, the so-called *threshold mechanism*.

Finally, when the fourth situation is satisfied, i.e., when $\delta_k(R_1)\delta_k(R_2)\cdots\delta_k(R_{t_k-1})R_{t_k} > 1 + \varepsilon$, then δ_k resets the chain of product to 1 so that the last pixel traveled becomes the local white reference. This is the mathematical implementa-tion of the *reset mechanism*. The *white patch behavior* of the algorithm is determined by this operation.

All the options together realize the well-known ratio-threshold-reset mechanism of Retinex.

This description of the original Retinex algorithm is exhaus-tive but not predictive because of the presence of the threshold mechanism. In [16], we have shown that, for such values of ε, the difference between Retinex with and without threshold has an upper bound which has negligible effects for the final light-ness computation. In [16], it has been proved that the really im-portant mechanism of Retinex is the reset, which determines its filtering properties. Hence, we are going to study Retinex fixing $\varepsilon=0$ until the rest of the paper.

The great advantage to neglecting the threshold mechanism is that the mathematical formulation undergoes a significant sim-plification, and it can be proved [16] that the lightness value can be written

with this closed formula

$$L(i) = \frac{1}{N} \sum_{k=1}^{N} \frac{I(i)}{I(x_{H_k})} \qquad (3)$$

where x_{H_k} is the pixel with highest intensity traveled by the path γ_k, for every $k = 1, \ldots, N$.

It is immediate to see that $L(i)$ is the average of the single path contributions $L_k(i) = (I(i)/I(x_{H_k}))$; hence, the only information that characterizes every contribution is the value of the pixel x_{H_k} with the highest intensity traveled by the path γ_k. This also means that Retinex is a pure local white-patch algorithm, where locality depends on path geometry.

It follows that the problem of finding $L_k(i)$ reduces to the problem of finding the maximum value assumed by the image function $I(x,y)$ along γ_k, for every $k=1...N$. We stress that *this result is independent from paths geometry*.

Finally, we remember that the analysis of formula (3) performed in [16] revealed *three intrinsic characteristics of Retinex*. The first is that its filtering properties are strongly image dependent; the second is that, when path lengths tend to very big values (compared with the image size), Retinex loses its local properties and becomes a global white-patch algorithm. Finally, the third intrinsic property is that Retinex is not an idempotent operator, i.e., in general, $L(L(i))\neq L(i)$, but subsequent iterations of Retinex converge to a fixed image qualitatively characterizable as follows: Every path travels at least one pixel with intensity 1 in every fixed chromatic channel.

III. FROM PATHS TO PIXEL SPRAYS

The information given by the mathematical formulation of Retinex have strong consequences on the structure of $\mathcal{P}_i(\text{Image})$: The set of paths embedded in the image and *ending in the point i*. After formula (3), on this set, it is natural to define this equivalence relation: Given $\gamma,\eta \in \mathcal{P}_i(\text{Image})$,

$$\gamma \sim \eta \Leftrightarrow \max_{(x,y)\in\gamma^*}\{I(x,y)\} = \max_{(x,y)\in\eta^*}\{I(x,y)\} \qquad (4)$$

where γ_* and η_* are the codomain of the paths, i.e., the collec-tions of pixels traveled by γ and η, respectively.

Paths belonging to different equivalence classes give different contributions to the lightness computation, while every path in a given equivalence class is characterized by the same value of $L_k(i)$. It immediately follows that, for the purposes of Retinex, $\mathcal{P}_i(\text{Image})$ contains redundant paths and so the really inter-esting set of paths is the *quotient set* $\mathcal{P}_i(\text{Image})/\sim$, whose el-ements are the equivalence classes of paths with respect to the equivalence relation defined in (4).

Path-wise Retinex implementations are affected by two kind of redundances: From one side, many paths must be used to reduce the sampling noise; from the

other side, as just proved, they can be organized in equivalence classes, so that if one uses two paths belonging to the same class, they will lead to the same chromatic information, i.e., they are redundant.

In each equivalence class, one can choose a single represen-tative path to compute $L_k(i)$; in particular, the shortest one is the two-points path whose codomain is simply $\{x_{Hi}\}$. It fol-lows that the ordering operations needed to generate the paths are perfectly uninfluential for the final lightness computation.

$$
\delta_k(R_{t_k}) = \begin{cases} R_{t_k}, & \text{if } 0 < R_{t_k} \leq 1-\varepsilon \\ 1, & \text{if } 1-\varepsilon < R_{t_k} < 1+\varepsilon \\ R_{t_k}, & \text{if } 1+\varepsilon \leq R_{t_k} \leq \frac{1+\varepsilon}{\prod_{m_k=0}^{t_k-1} \delta_k(R_{m_k})} \\ \frac{1}{\prod_{m_k=0}^{t_k-1} \delta_k(R_{m_k})}, & \text{if } R_{t_k} > \frac{1+\varepsilon}{\prod_{m_k=0}^{t_k-1} \delta_k(R_{m_k})} \end{cases} \quad (2)
$$

Moreover, by a mathematical point of view, paths are topo-logical manifolds of dimension 1 embedded in the image, which is a topological manifold of dimension 2, so paths do not really scan local neighborhoods of a pixel, but rather *particular di-rections* in these neighborhoods. This directional extraction of information can lead to halos or artifacts in the filtered image.

The classical implementations of Retinex try to remedy this problem using a large number of paths, but this increases the filtering time and does not really overcome the problem.

We see that there are three reasons for which paths are not perfectly suitable for the analysis of locality of color perception within the Retinex model: They are redundant, their ordering is completely uninfluential, and they have inadequate topological dimension.

Thus, we are lead to use 2-D objects, such as *areas*, instead of 1-D paths to analyze locality of color perception. More pre-cisely, our idea is to implement the investigation about locality selecting pixels from these areas with a density sample that changes according to a given function of their distance with re-spect to the target pixel i. Each function generates a different kind of pixel selection around i, leading to different kind of "sprays," each of which reveals different local filtering proper-ties.

Sections IV– VI explain in detail how it is possible to make these ideas concrete, generating the new implementation of Retinex that we have denoted with *RSR: "Random Sprays Retinex."*

IV. RSR IMPLEMENTATION

RSR is a new implementation of the original Retinex model [1] which has been inspired from the results of the mathematical analysis of Retinex performed in [16]. In RSR, the role of a path γ_k traveling n_k pixels and *ending* in the target i is played by $\mathrm{Spray}_k(i)$, a spray composed by n_k pixels and *centered* in i. In fact, N random sprays are selected from a precomputed set (the symbol N now will be used to denote the number of sprays to put in stronger evidence of the correspondence between paths and sprays). The typical ratio-reset operation along a path is substituted by the search of the pixel with highest intensity in the whole spray. It will be clear from the following discussion that, once the number of points per spray is chosen, there is no need to vary it with k; hence, from now on, we will write n instead of n_k to denote the number of pixels per spray.

The functional expression of the formula (3) to compute the lightness remains exactly the same in both algorithms, so they share the same intrinsic properties recalled at the end of Sec-tion II. This is the reason why the results about locality of color perception that we will get thanks to the RSR *implementation* can be referred to the Retinex *model*.

Notice that the only operations performed by RSR in each spray are n comparisons (needed to find out the pixel with highest intensity) and one division. So, RSR is significantly faster than the previous path-wise Retinex implementations.

Let us now show how to construct $\mathrm{Spray}_k(i)$. With a random point generator, we can get a *uniform random distribution* of n values in the real unit interval $[0\,1]$. Then, by multiplication, we can extend this distribution to any real interval; in particular, we are interested to the intervals $[0\,2\pi]$ and $[0\,R]$, where R,
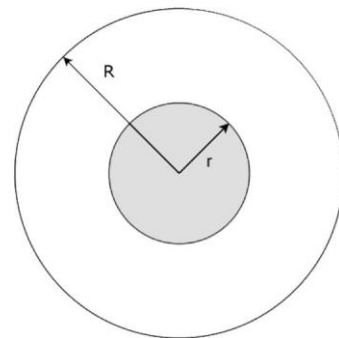

Fig. 1. Computation of the mean areolar density in function of the spray radius.

is a given positive real number that will represent the radius of the spray. We denote, respectively, with $\mathrm{RAND}_n[0\,2\pi]$ and $\mathrm{RAND}_n[0\,R]$ the corresponding, uniform random distributions.

Now, if $(i_x\,i)$ are the coordinates of i, we can define the polar coordinates of a generic pixel $j \equiv (j_x\,j)$ belonging to $\mathrm{Spray}_k(i)$; in this way.

$$\begin{cases} j_x = i_x + \rho\cos(\theta) \\ j_y = i_y + \rho\sin(\theta) \end{cases} \quad (5)$$

where $\rho \in RAND_n[0^R]$, $\theta \in RAND_n[02\pi]$. ,

These are the coordinates of pixels that have an isotropic an-gular distribution in a circle of radius R centered on the pixel i.

Notice, however, that the radial density is not isotropic; in fact, because of the rotation, the spray results are more dense near the target pixel i than far away. To compute $\delta(r)$, the *mean areolar density* variation in the function of r, consider, as in Fig. 1, a circle of arbitrary radius , r $0 < r < R$, c tered on . The area of is C_r $A = \pi r^2$, so $r = \sqrt{\frac{A}{\pi}}$, mo over, since we are dealing with uniform random distributions, the mean number of points inside C_r is $n_r = n\sqrt{A}$. Com-puting the derivative of n with respect to A, we get the rate of change of the average areolar density, in fact

$$\frac{d\left(\frac{n}{R}\sqrt{\frac{A}{\pi}}\right)}{dA} = \frac{n}{2R\sqrt{\pi A}}$$

but $A = \pi r^2$, so

$$\delta(r) = \left(\frac{n}{2\pi R}\right)\frac{1}{r}. \quad (6)$$

Thus, the mean radial density of spray pixels decreases as the inverse radius.

Fig. 2 shows an example of such a spray with 400 pixels and radius $R=1$.

The angular isotropy is a natural requirement that must be satisfied by the spray, since the presence of privileged directions generates artifacts and haloes in the filtered image.

Now, the local properties of Retinex can be analyzed in a very simple way applying a function on the coordinate ρ to change the radial density of the spray pixels around i. Precisely, given
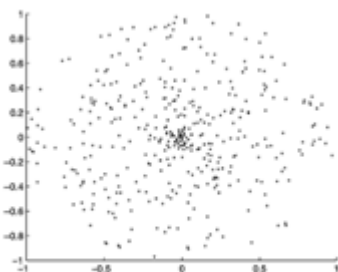

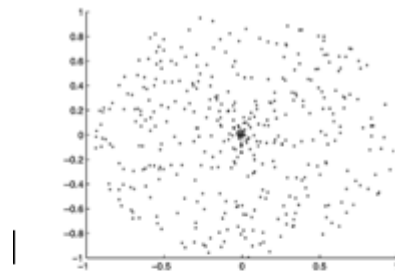
Fig. 2. Example of "naturally localized" spray.
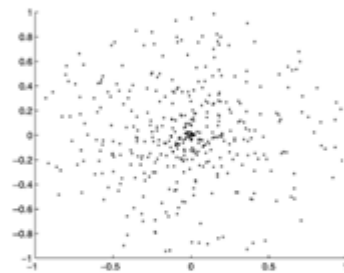

Fig. 3. Spray with f ( ) = (log(1 +)= log(2)).
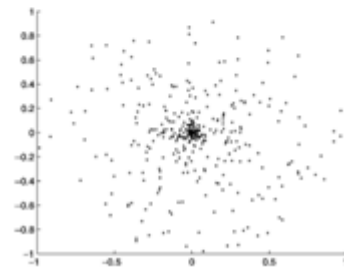

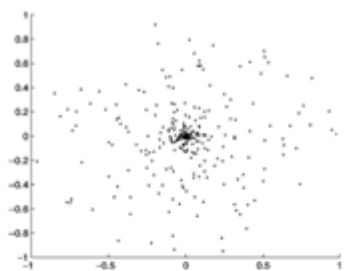Fig. 4. Spray with f ( ) = (sinh( )= sinh(1)).

Fig. 5. Spray with f ( ) = ₂.

Fig. 6. Spray with f ( ) = ₄.
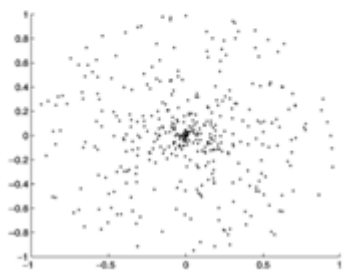
Fig. 7. Spray with f ( ) = (e ₀ 1=e ₀ 1).
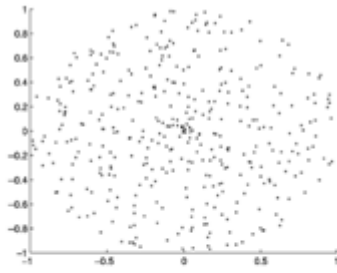
Fig. 8. Spray with f ( ) = (e_ () 1=e_ () 1).
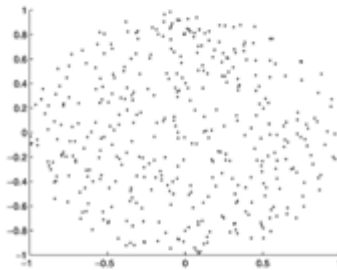

Fig. 9. Spray with f ( ) = ρ. −
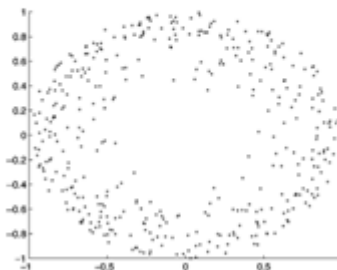

Fig. 10. Spray with f ( ) = ρ. ₅ −

any function $f: \mathbb{R}^+ \to \mathbb{R}^+$, we can consider the modified spray whose pixels have polar coordinates defined by

$$\begin{cases} x = x_i + f(\rho)\cos(\theta) \\ y = y_i + f(\rho)\sin(\theta) \end{cases} \quad (7)$$

where, again, $\rho \in \mathrm{RAND}_n[0, R]$ and $\theta \in \mathrm{RAND}_n[0, 2\pi]$. It is useful to distinguish the special case in which $f \equiv id_{\mathbb{R}^+}$, the identity function restricted on nonnegative real numbers, calling the corresponding spray "naturally localized."

Figs. 3– 10 show some examples of sprays with $R=1, n= 400$, obtained with different functions f.

It can be seen that the normalized logarithmic and hyperbolic sinus functions keep the spray density quite similar to the one of the naturally localized spray. Instead, powers of ρ with ex-ponents greater than 1 and the normalized exponential function tend to increase the density around the center. Finally, powers of ρ with exponents in $(0, 1)$ and the normalized inverse exponen-tial applied on ρ tend to delocalize the spray. The multiplication of ρ by a constant coefficient m simply changes the radial ex-tension of the naturally localized spray, expanding the radius, when $m>1$, or contracting it, when $0<m<1$.

To perform the analysis of locality in RSR, we

must tune f and the other parameters of the algorithm. Before showing the results about tuning, we briefly summarize all these parameters and discuss their meaning in Section IV-A.

A. RSR Parameters and Their Meaning

RSR depends on four parameters: R (the radius of the sprays), f (the radial density function) N (the number of sprays), and n (the number of pixels per spray).

The radius R of the spray defines the extension of the circular area analyzed around the pixel i. This area must be tuned to get enough information about the color distribution around i.

As already stated, the function f changes the radial density of the spray pixels. It must be tuned to find out what is the spray pixel distribution that better fits the computational reproduction of color perception performed by the HVS.

For each $\mathrm{Spray}(i)$, there is a non zero probability to find the pixel with highest intensity x_H in an isolated pixel not related to the context. This, of course, would produce chromatic noise in the filtered image. Since the spray pixels are generated by a random point generator, all the N sprays are different, and so, statistically speaking, the influence of isolated pixels on the global computation of $L(i)$ decreases when we average many sprays contributions $L_k(i)$. Hence, the higher the number of sprays, the lower the chromatic noise in the filtered image. This is confirmed by the tests performed (as will be discussed later), which also shown that, to avoid pattern replication all across the image, the sprays must be taken by a precomputed set of, at least, a thousand sprays.

Finally, the number n of pixels per spray determines how much information is extracted from the spray area. If we use very large values of n, we cover the whole spray area, losing the locality of the spray distribution; instead, if we use small values of n, we cannot get enough information to correctly compute L (i).

V. TUNING RSR PARAMETERS

We performed our tests on a set of over 100 very different pictures given by real-world images, portraits, landscapes, and geometric images.

A. Tuning the Spray Radius

The easiest parameter to tune has proved to be the radius: For all images and independently from the other parameter of RSR, our tests showed that the

optimal value for R is DIAG, the value of the diagonal of the image.

The reason is easily comprehensible: If one uses a smaller radius, then two pixels that lie near the extreme points of the diagonals can never be compared. The effect of using a smaller radius than DIAG can be clearly seen comparing Figs. 11 and 12, which have been filtered with $R=\text{DIAG}/2$ and $R=\text{DIAG}$,

Fig. 11. Image filtered with spray radius R - DIAG=2.

Fig. 12. Same image as in Fig. 11 filtered with spray radius R - DIAG.

Fig. 13. Original "Gallery" image.

respectively, keeping all the other parameters constant: ρ as ra-dial coordinate, $N=30$, $n=800$.

Furthermore, it is not useful to use a radius larger than DIAG, since the spray loses part of its density around the target pixel and many spray points lie outside the image area.

B. Tuning the Radial Density Function

The radial density of the spray is responsible for the local property of RSR because the probability to find out the pixel with highest intensity in the spray is greater in the image areas where the spray is denser than in the image regions where the spray has only few points.

It is well known that tests about human color perception show that the chromatic influence between two pixels decreases with their distance (e.g., [17]–[20]). This fact is implemented in every color perception model: path-wise algorithms (e.g., [6]) sample the image content with paths that are denser in the immediate neighborhood of the target pixel than far away, while integra-tive algorithms (e.g., [10]) use a center/surround technique that weights the surround of the target pixel with monotonically de-creasing functions.

Coherently with this, even RSR revealed that delocalized sprays are inadequate to correctly simulate color perception by the HVS. For example, Fig. 14 shows the result of filtering the image in Fig. 13 using $\sqrt[5]{\rho}$ as radial coordinate.
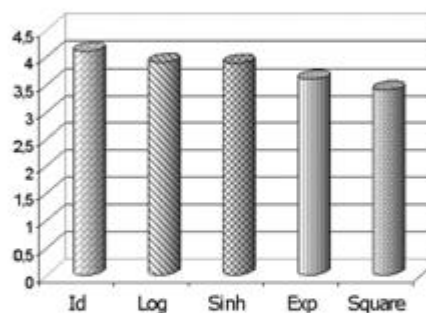

Fig. 14. Effects of a spray with radial coordinate 5.

Fig. 15. Quality test for different radial density functions.

Fig. 16. Noise induced by a spray with radial coordinate 4.

As a consequence, the only interesting radial density func-tions are those that correspond to monotonically decreasing ra-dial densities. Only such functions will be considered in the next discussion.

We conducted the tuning using both subjective quality match tests and quantitative tests about color constancy.

The first kind of tests has been developed as follows: We fil-tered our test set of images fixing n and N and varying the ra-dial density function. We have displayed the images on a middle gray background of a calibrated monitor in a dark room. Then we asked a collection of users to indicate in a scale between 1 (poor) and 5 (excellent), the degree of naturalness (color plausi-bility in relation to the personal experience), absence of noise, and detail visibility of the filtered images. The results of our tests, averaged on the three questions and on the test set images, are shown in Fig. 15.

The images filtered with the naturally localized spray have al-ways received the best judgement by the users. Starting from ρ_2 , the sprays results too localized and the corresponding filtered images show an increasing amount of noise, as can be seen in Fig. 16, that has been filtered with ρ_4 as radial coordinate and with $N=30$, $n=800$ (to be compared with Fig. 12, which has been filtered with the same values of n and N, but with ρ as radial coordinate).

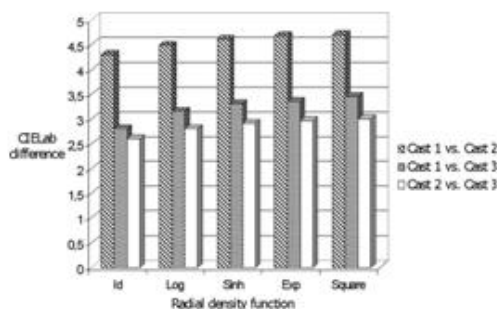Fig. 17. Image of the database YACCD for color constancy tests.

Fig. 18. Color constancy test for different radial density functions.

Regarding color constancy tests, we considered the pictures of the database described in [21], consisting in a series of pho-tographs taken under different color casts. We filtered each se-ries of pictures with different radial density functions. Then, we computed the CIELab differences between the images filtered with every given radial density function to have a measure of the corresponding algorithm ability to reduce color cast. This methodology is motivated by the fact that RSR always preserves the image content and does not collapses the dynamic range.

For more readability, we report only the results of our tests on the picture in Fig. 17 taken under three different casts: Cast 1 = PHILIPS Neon Neutral Daylight 6500K (TLD965), Cast 2 = PHILIPS Neon Fluotone 4100K (TLD840), Cast 3 = PHILIPS Neon Daylight 5000K (TLD950). Tests with the other casts shown analogous results. We choose the database in [21] since it has been devised to test color correction algorithms without fa-cilitating any of them. In fact, instead of choosing a white, gray or black background, we used two white noise backgrounds with different spatial frequencies.

The values visualized in the graphics of Fig. 18 correspond to the parameters $N=20$ and $n=400$, when these parameters are varied the numerical values of the differences change, but the relationship between the different radial density functions does not change.

It can be seen that the density function that minimizes the CIELab difference between the filtered images is the identity function. Tests on the other images exhibit analogous results.

The consequence of our subjective and quantitative tests is that the naturally localized spray is the most suitable to repro-duce the behavior of the HVS within the RSR implementation of the Retinex model. From now on, RSR will be considered only with ρ as radial coordinate.

We recall from (6) that the mean areolar density of the natu-rally localized spray decreases as the inverse distance from the center. It follows that, in RSR, a fixed pixel i, every other pixel of the image, considered as a single entity, has a "mean chromatic influences" on i that decreases as the inverse distance from i. This fact implies that, statistically speaking, the chromatic in-fluence of pixels close to i is comparable only with that of entire areas of pixels far from i, the wideness of which must increase, according to (6). This seems to be a good motivation to study multilevel extensions of RSR.

Finally, we notice that the result of this section corresponds to what found in the tuning experiments of another color per-ception model: automatic color

equalization (ACE) [22], [24]. In that algorithm, the target pixel is compared with the other image pixels, each of which is weighted with a coefficient. In [22], it has been shown that good weight coefficients are the in-verse distances from the target.

C. Tuning the Number of Sprays and Pixels Per Spray

One of the consequences of the mathematical analysis per-formed in [16] is that, as the paths length of a path-wise Retinex implementation grows to great values, the algorithm loses its local properties showing a global white patch behavior. The tuning of paths length or number is still an open problem for path-wise Retinex implementations.

We are now going to show that, with the RSR implementa-tion, it is possible to perform an unsupervised tuning of the pa-rameters N and n in a self-consistent way, highly reducing the range of their optimal values.

These two parameters are strictly related because the light-ness is computed averaging the N contributions of the sprays, each of which depends on how many points are used to find out the pixel with highest intensity.

We carried out the tuning as follows. We filtered the images of our test set increasing N from 5 to 60 with a constant step of five sprays and increasing n from 250 to 900 with a constant step of 50 points. Then, we calculated $\Delta E_N(n)$ and $\Delta E_n(N)$, the CIELab differences between the images filtered with a fixed value of N and two consecutive values of n, and vice versa, with N playing the role of n. We observed that both $\Delta E_N(n)$ and $\Delta E_n(N)$ decrease monotonically for all images.

Now, since two images are considered chromatically indis-tinguishable if $\Delta E < 1$, it is natural to tune N and n, taking the smallest values of these parameters for which this inequality holds true. In other words, this procedure is a natural compro-mise between the minimization of filtering time and the maxi-mization of filtering quality.

To have a quantitative example to discuss, let us consider the tests performed on the image in Fig. 19.

The interpolation graphic of ΔE, viewed as a function of N and n, and its intersection with the hyperplane $\Delta E \equiv 1$, is visualized in Fig. 20.

Figs. 21 and 22 represent the interpolation graphics of the functions $\Delta E_N(n)$ and $\Delta E_n(N)$, which are the level curves of the surface in Fig. 20. Indicated in the horizontal axis are the two consecutive values of n or N corresponding to the CIELab

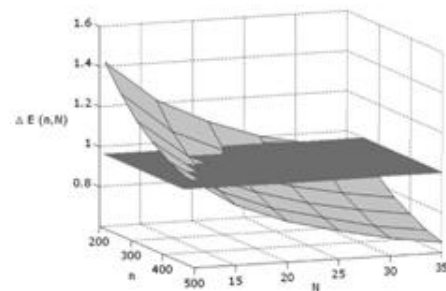Fig. 19. Image for the tuning of the parameters n and N.

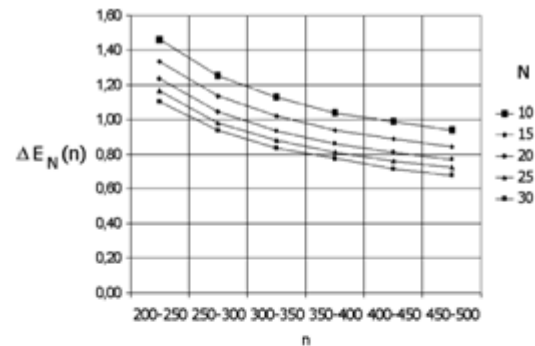Fig. 20. Surface of $\Delta E(n; N)$ intersecting the hyperplane $\Delta E \equiv 1$.

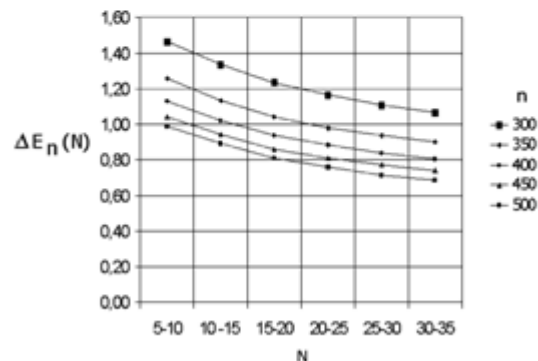Fig. 21. Graphics of $\Delta E_N (n)$ for different values of N.

Fig. 22. Graphics of $\Delta E_n(N)$ for different values of n.

difference values displayed in the graphic. Only the significant part of the curves are visualized.

Since the parameters n and N control two different charac-teristics of the filtered image, it is not sensed to take high values of n and little values of N, or vice versa, because the corre-sponding image would have

good chromatic quality, but high chromatic noise, or vice versa, respectively. Instead, the optimal
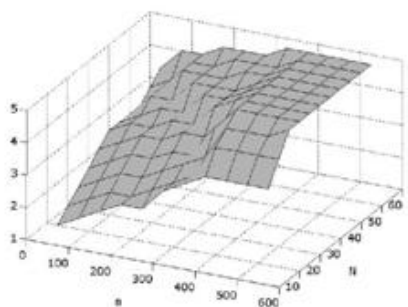

Fig. 23. Quality test for the parameters n and N with respect to Fig. 19.

Fig. 24. "Gallery" filtered with tuned parameters: N = 25, n = 800.

couple (Nn) must be chosen as the "minimal" couple of in-termediate values of N and n such that the surface $\Delta E(nN)$ lies under the hyperplane $\Delta E \equiv 1$, where, with "minimal, " we mean the couple that minimizes the product $n.N$. For ex-ample, it can be seen from Figs. 21 and 22 that both the couples $(N{=}25n{=}350)$ and $(N{=}20n{=}400)$ correspond to intermediate values of N and n such that the surface lies under $\Delta E \equiv 1$, but $25.350{=}8750{>}20.400{=}8000$, so that the optimal choice is $(N{=}20n{=}400)$, because it corresponds to 750 operations per pixels less than the other couple.

We combined the procedure just described with subjective matches analogous to those performed for the tuning of the ra-dial density function, but now changing the values of N and n every time. The results of the tests performed on the image shown in Fig. 19 are presented in Fig. 23. The surface is ob-tained interpolating the values at the nodes (nN), the value at each node is calculated averaging the degree of naturalness, ab-sence of noise and detail visibility indicated by the users.

It can be seen from the graphic in Fig. 23 that the surface re-veals a wide constant area after the couple of parameters (Nn) overcomes (20 400), as predicted by the quantitative procedure described above. It is evident that there is no reason to incre-ment the filtering time taking greater values for N and n. All the other tests performed has revealed agreement between the un-supervised procedure described above and the subjective tests involving users.

Now that we described the tuning procedure, we

show in Figs. 24– 28 some output results of RSR with tuned parameters.

As can be seen from the different values of optimal values of N and n for the various images, the tuning of N and n strongly depends on the different image content. The problem to find out a formula to precisely determine the variation of the parameters N and n in relation with the image content still remains open.

Fig. 25. Original "Books" image.

Fig. 26. "Books" filtered with tuned parameters: N = 25, n = 750.

Fig. 27. Original "Flowers" image.

Fig. 28. "Flowers" filtered with tuned parameters: N = 20, n = 450.

D. *Filtering the Same Image With Different Sizes*

If we consider a given image at different sizes, then we need a formula to extend the optimal values of N and n for a certain size to the other sizes of the same image. Our tests have shown that the optimal value of N remains constant, but, as expected, the optimal value of n changes. In fact, n determines the amount of information needed to compute the lightness and obviously this amount must increase or decrease in relation with the image size.

We can formalize the problem in this way: Suppose we have the same image at the sizes $W_0{\times}H_0$ and $W_1{\times}H_1$, and sup-pose that $n_{\mathrm{oPt}}(W_0 H_0)$, the optimal value of n for the image of width W_0 and height H_0, is known. The easiest way to find out $n_{\mathrm{oPt}}(W_1 H_1)$, the optimal value of n for the image of width W_1 and height H_1, is to impose this mathematical proportion

$$n^{\mathrm{oPt}}(W_1, H_1) : W_1 H_1 = n^{\mathrm{oPt}}(W_0, H_0) : W_0 H_0 \quad (8)$$

i.e., to impose the fraction of spray pixels per unit of image area to remain constant.

Applying the unsupervised tuning procedure previously de-scribed, all the images of our test set shown that, once the op-timal value of n is found for a given image size, formula (8) enables to correctly compute the changes of the optimal value of n in function of the new image sizes.

VI. COMPARISON BETWEEN RETINEX ALGORITHMS: AN OPEN PROBLEM

The present work is the sequel of a [16] in which the intrinsic mechanisms of the original Retinex algorithm of Land and Mc-Cann [1] have been mathematically described and analyzed. In this sequel we have proved that if the path-wise structure is sub-stituted with the random spray structure, then interesting infor-mation about local Retinex properties arise naturally. This is, of course, just a first step toward the full comprehension of the spatial properties of the Retinex model.

There is a big interest in the comparison among spatial prop-erties of all the different Retinex implementations available in literature. We believe that this is a very challenging task, and it should still be considered an open problem. In fact, the be-havior and consequent performances of such algorithms affect, and are affected by, several image characteristics: not only do algorithm parameters modify the final result, but image features also change the parameter choice. This regards several visual as-pects, among them contrast, frequency content, saturation, and noise.

To judge the pleasantness and quality of the output images, a perceptual analysis is necessary, but this is particularly difficult since there is not yet a universally accepted perceptual measure to compare image quality.

Moreover, if we consider the ability to remove color cast, a judging criterion for the algorithm efficiency, it has to be con-sider that, differently from machine (or perfect) color constancy, the human color constancy property is never complete, and it depends on several factors, such as temporal transients or illu-sive visual configurations. So, a comparison between algorithms based on this property would be insidious to implement.

Another great difficulty for a complete comparison is the fact that every Retinex implementation depends highly on its own parameters, whose tuning, in the few cases in which it has been performed [23], is based on very different criteria and image test sets.

Finally, a mathematical description of all the algorithms con-sidered would create the basis for a common background where performing comparisons about the intrinsic properties of each implementation.

All the open problems briefly described above make the im-portant issue of a proper and exhaustive comparison still a dif-ficult task that we deem interesting for future research.

VII. CONCLUSION

We presented a new implementation of the Retinex model in which the chromatic information in the image is scanned by 2-D pixel sprays instead of 1-D paths, hence the name RSR for "random spray Retinex." The passage from paths to pixel sprays is based on a recent mathematical characterization [16] of path-wise Retinex implementations.

While RSR shares the same intrinsic properties of every path-wise Retinex implementation, it has proved to be faster and more suitable to analyze the local Retinex properties, its parameters being easily handled in order to perform spatial investigation.

We have analyzed, both quantitatively and qualitatively, the RSR performances. The analysis regards two groups of param-eters, one related to the spatial exploration and the other to the amount of chromatic information considered. To tune these parameters, we proposed an unsupervised method, validated through user panel tests.

Concerning spatiality, the results have shown that the mean chromatic influence between pixels decreases as the inverse of their distance, while the amount of information (number of sprays and pixels per sprays) required for optimal results strongly

depends on the image content.

REFERENCES

[1.] E. Land and J. J. McCann, "Lightness and Retinex theory," J. Opt. Soc. Amer. A, vol. 61, pp. 1–11, 1971.

[2.] J. J. McCann (chair), "Special session on Retinex at 40," J. Electron. Imag., vol. 13, no. 1, pp. 6–145, 2004.

[3.] 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," Vis. Res., vol. 16, no. 5, pp. 445–58, 1976.

[4.] E. Land, "The Retinex theory of color vision," Sci. Amer., vol. 237, no. 3, pp. 2–17, 1977.

[5.] J. Frankle and J. J. McCann, "Method and Apparatus For Lightness Imaging," U.S. Patent 4 348 336, 1983.

[6.] D. Marini and A. Rizzi, "A computational approach to color adaptation effects," Image Vis. Comput., vol. 13, pp. 1005–1014, 2000.

[7.] T. J. Cooper and F. A. Baqai, "Analysis and extensions of the Frankle-Mccann Retinex algorithm," J. Electron. Imag., vol. 13, no. 1, pp. 85–92, 2004.

[8.] E. Land, "Recent advances in Retinex theory and some implications for cortical computations: Color vision and the natural image," Proc. Nat. Acad. Sci., vol. 80, pp. 5163–5169, 1983.

[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. Image Process., vol. 6, no. 7, pp. 965–976, Jul. 1997.

[10.] "Properties and performance of a center/surround Retinex," IEEE Trans. Image Process., vol. 6, no. 3, pp. 451–462, Mar. 1997.

[11.] K. Barnard and B. Funt, "Investigations into multi-scale Retinex," in Colour Imaging: Vision and Technology. New York: Wiley, 1999, pp. 9–17.

[12.] S. O. Huck, C. L. Fales, R. E. Davis, and R. Alter-Gartenberg, "Visual communication with Retinex coding," Appl. Opt., vol. 39, no. 11, pp. 1711–1730, 2000.

[13.] J. D. Cowan and P. C. Bressloff, B. Rogowitz and N. Pappas, Eds., "Visual cortand the Retinex algorithm," in Proc. SPIE Human Vision di Milano, Milano, Italy, and Electronic Imaging VII, 2002, vol. 4662, pp. 278–285.

[14.] B. Funt, F. Ciurea, and J. J. McCann, "Retinex in MatLab," J. Electron. Imag., vol. 13, no. 1, pp. 48–57, 2004.

[15.] Z. Rahman, D. J. Jobson, and G. A. Woodell, "Retinex processing for automatic image enhancement," J. Electron. Imag., vol. 13, no. 1, pp. 100–110, 2004.

[16.] E. Provenzi, L. D. Carli, A. Rizzi, and D. Marini, "Mathematical defi-nition and analysis of the Retinex algorithm," J. Opt. Soc. Amer. A, vol. 22, pp. 2613–2621, 2005.

[17.] O. Creutzfeld, B. Lange-Malecki, and K. Wortmann, "Darkness induc-tion, Retinex and cooperative mechanisms in vision," Exp. Brain Res., vol. 67, pp. 270–283, 1987.

[18.] [18] O. Creutzfeld, B. Lange-Malecki, and E. Dreyer, "Chromatic induction and brightness contrast: A relativistic color model," J. Opt. Soc. Amer. A, vol. 7, no. 9, pp. 1644–1653, 1990.

[19.] L. M. Hurvich and D. Jameson, "Theory of brightness and color con-trast in human vision," Vis. Res., vol. 4, pp. 135–154, 1990.

[20.] Q. Zaidi, "Color and brightness induction: From Mach bands to three-dimensional configurations," in Color Vision: From Genes to Perception, K.Gegenfurtner and L. Sharpe, Eds. New York: Cam-bridge Univ. Press, 1999.

[21.] A. Rizzi, C. Gatta, and D. Marini, "Yaccd: Yet another color constancy database," in Proc. IS&T/SPIE Int. Symp. Color Imaging VIII, San Jose, CA, 2003, vol. 5008, pp. 24–35.

[22.] "A new algortithm for unsupervised global and local color cor-rection," Pattern Recognit. Lett., vol. 124, pp. 1663–1677, 2003.

[23.] B. Funt and F. Ciurea, "Tuning Retinex parameters," J. Electron. Imag., vol. 13, no. 1, pp. 58–64, 2004.

[24.] 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. Imag., vol. 13, no. 1, pp. 75–84, Jan. 2004.