



Published in Image Processing On Line on 2013-09-26.  
 Submitted on 2012-09-05, accepted on 2013-04-03.  
 ISSN 2105-1232 © 2013 IPOL & the authors CC-BY-NC-SA  
 This article is available online with supplementary materials,  
 software, datasets and online demo at  
<https://doi.org/10.5201/ipol.2013.41>

# Selective Contrast Adjustment by Poisson Equation

Ana Belén Petro<sup>1</sup>, Catalina Sbert<sup>2</sup>

<sup>1</sup>University of Balearic Islands, Spain ([anabelen.petro@uib.es](mailto:anabelen.petro@uib.es))

<sup>2</sup>University of Balearic Islands, Spain ([catalina.sbert@uib.es](mailto:catalina.sbert@uib.es))

*Communicated by* Luis Álvarez

*Demo edited by* José-Luis Lisani

## Abstract

Poisson Image Editing is a new technique permitting to modify the gradient vector field of an image, and then to recover an image with a gradient approaching this modified gradient field. This amounts to solve a Poisson equation, an operation which can be efficiently performed by Fast Fourier Transform (FFT). This paper describes an algorithm applying this technique, with two different variants. The first variant enhances the contrast by increasing the gradient in the dark regions of the image. This method is well adapted to images with back light or strong shadows, and reveals details in the shadows. The second variant of the same Poisson technique enhances all small gradients in the image, thus also sometimes revealing details and texture.

## Source Code

The source code, the code documentation, and the online demo are accessible at the [IPOL web page of this article](#)<sup>1</sup>. In this page an implementation is available for download. Compilation and usage instruction are included in the `README.txt` file of the archive.

**Keywords:** color enhancement, contrast adjustment, Fourier transform

## 1 Overview

The concept of image editing encompasses the operations by which the local content of one or several images is selected and manipulated to create new synthetic images. The simplest such operation is a copy-paste of a part of an image into another. Local contrast or color adjustments after selecting manually a part of the image (for example the shadows) is another variant.

Most image editing operations modify an input image  $I$  on a rectangular domain  $R$  by manually selecting a region  $\Omega$  in it. The task is to fill in this region by picking information from the rest of the image  $R \setminus \Omega$ , or from another image or, in the case of a selective contrast change, from the image in  $\Omega$  itself. The most recent algorithms to solve this problem are based on partial differential equations. The main challenge in image editing is to avoid suspicious color or texture alterations which would

<sup>1</sup><https://doi.org/10.5201/ipol.2013.41>

reveal the silhouette  $\partial\Omega$  of the image region which has been modified. As amply demonstrated by Perez et al. [2], the Poisson equation

$$\begin{cases} \Delta u(x) = \operatorname{div}(\mathbf{V})(x) & \text{if } x \in \Omega \\ u(x) = g(x) & \text{if } x \in \partial\Omega \end{cases}, \quad (1)$$

where

$$\Delta u(x, y) = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \quad (2)$$

is the Laplacian of  $u$  and  $\mathbf{V}$  is a prescribed vector field in  $\Omega$ , is an extremely efficient response to this problem. The proposed algorithm, a particular example of Poisson image editing, reaches a contrast improvement in the dark images. The proposed numerical method permits to automatize the local contrast enhancement proposed by Perez et al. [2]. This new tool is fast and efficient, requiring no manual selection and allowing one to edit regions with arbitrary complex topology with no additional cost. Moreover, the overall impact is to brighten up areas of poor contrast but not at the expense of saturating areas of good contrast.

## 2 Poisson Image Editing

Perez et al. [2] proposed to reconstruct the image from a "guidance vector field". Let  $R$ , a closed subset of the plane, be the image domain, and let  $\Omega$  be a closed subset of  $R$  with boundary  $\partial\Omega$ . The problem is to find the image whose gradient field is the closest, in quadratic distance, to the prescribed "guidance vector field"  $\mathbf{v}$  defined in  $\Omega$ , under given boundary conditions on  $\partial\Omega$ . The image is known in the rest of the image,  $R \setminus \Omega$ . This problem writes

$$\min_u \int_{\Omega} |\nabla u - \mathbf{v}|^2, \quad \text{with } u|_{\partial\Omega} = f|_{\partial\Omega}, \quad (3)$$

and its solution is the unique solution of the Poisson equation with Dirichlet boundary conditions

$$\Delta u = \operatorname{div} \mathbf{v}, \quad \text{with } u|_{\partial\Omega} = f|_{\partial\Omega}. \quad (4)$$

Although good results can be achieved using Poisson image editing, it is often a time-consuming process for the user. An acceleration of the method was proposed by Morel et al. [1], by solving directly the PDE by the Fourier method. This work shows that a simple global reformulation of the Poisson editing problem permits to use fast Fourier solvers. A Fourier solver for the Poisson equation is exact, non-iterative and in practice faster than a full multigrid method, even in the case of a rectangular domain. The main idea is to extend the definition domain of the guidance vector field to the whole image domain  $R$ , and to solve the variational problem over the rectangular image domain, which is easily done by a Fourier technique. The guidance vector field  $\mathbf{v}$  defined on  $\Omega$  is extended to  $R$  by

$$\mathbf{V} = \begin{cases} \mathbf{v} & \text{over } \Omega \\ \nabla f & \text{otherwise} \end{cases}, \quad (5)$$

where  $f$  is the image, which is known in  $R \setminus \Omega$ . Then, the problem is to minimize

$$\min_u \int_R |\nabla u - \mathbf{V}|^2. \quad (6)$$

The minimizer is uniquely determined by the Euler-Lagrange equation

$$\Delta u = \operatorname{div} \mathbf{V}, \quad \text{over } R, \quad (7)$$



with homogeneous Neumann boundary condition

$$\frac{\partial u}{\partial \mathbf{n}} = 0 \quad \text{over } \partial R, \quad (8)$$

where  $\mathbf{n}$  is the direction orthogonal to the boundary.

### 3 Selective Contrast Adjustment

Perez et al. [2] use the method of Fattal et al. [3] to modify smoothly the image dynamic range. The idea is to transform the gradient field of the logarithm of the image to reduce the large gradients and to increase the small ones. The transformed gradient (the guidance vector field) is defined by

$$\mathbf{v} = \alpha^\beta |\nabla f|^{-\beta} \nabla f, \quad (9)$$

where  $\alpha = 0.2$  times the average gradient norm of  $f$  over  $\Omega$  and  $\beta = 0.2$ , that is, the guidance vector field is a concave function of the gradient in the log-domain. This transformed gradient is used to reconstruct the logarithm of the image, by solving the Poisson equation. Perez et al. [2] select the region  $\Omega$  manually, for example as an under-exposed region. This only permits to select a few simple image regions. Thus, the method is not easily adapted to treat the complex dark regions of images with back light or excessively dark shadows.

The present paper implements two simpler variants of the Perez et al. idea. The first one consists in amplifying the gradient on the dark regions of the image, yet, the dark regions can be numerous, and can have a complicated topology, and therefore can hardly be selected manually. The alternative proposed here is to simply select the dark region by using a threshold  $T$ . Over the dark regions, the gradient vector of the gray level intensity is amplified linearly. In general, the region  $\Omega$  is defined by

$$\Omega = \{x \in R : f(x) \leq T\}, \quad (10)$$

and the guidance vector field, is defined as

$$\mathbf{V} = \begin{cases} a \nabla f & \text{in } \Omega \\ \nabla f & \text{otherwise} \end{cases}, \quad (11)$$

where  $a$  is some constant.

The second proposed method consists in modifying the gradient over the image domain by a concave or convex function, for example a power function. Then the guidance vector field is defined by

$$\mathbf{V} = |\nabla f|^{\alpha-1} \nabla f, \quad (12)$$

and the norm of the vector field becomes a power of the norm of the gradient. A concave power is adapted to enhance small image details (often in dark regions). A convex power may instead be adapted to overexposed images.

### 4 Fourier Transform Method

The Fourier transform method is the fastest and simplest way to solve the partial differential equation. Like in the implementation of the Retinex equation [4], the Neumann boundary condition is implicitly imposed by extending the original image symmetrically across its sides, so that the extended image, which is four times bigger, becomes symmetric and periodic. The discrete Fourier transform (DFT) permits to compute directly the Fourier coefficients of a band limited and  $(J, L)$ -periodic function  $u$  from its samples  $u_{jl}$  on a  $J \times L$  grid.

**Definition 1.** *The Discrete Fourier Transform is*

$$\widehat{u}_{mn} = \sum_{j=0}^{J-1} \sum_{l=0}^{L-1} u_{jl} e^{-i\frac{2\pi jm}{J}} e^{-i\frac{2\pi ln}{L}}, \quad m = 0, \dots, J-1, \quad n = 0, \dots, L-1. \quad (13)$$

**Definition 2.** *The Inverse Discrete Fourier Transform is*

$$u_{jl} = \frac{1}{JL} \sum_{m=0}^{J-1} \sum_{n=0}^{L-1} \widehat{u}_{mn} e^{i\frac{2\pi jm}{J}} e^{i\frac{2\pi ln}{L}}, \quad j = 0, \dots, J-1, \quad l = 0, \dots, L-1. \quad (14)$$

Thus, by a simple differentiation, the Poisson equation translates into a relationship between the Fourier coefficients of  $u$  and  $\mathbf{V}$ :

$$\left( \left( \frac{2\pi m}{J} \right)^2 + \left( \frac{2\pi n}{L} \right)^2 \right) \widehat{u}_{mn} = \frac{2\pi im}{J} \widehat{V}_{1mn} + \frac{2\pi in}{L} \widehat{V}_{2mn}, \quad (15)$$

where  $\mathbf{V} = (V_1, V_2)$ .

## 5 Algorithm

The main idea of the selective contrast adjustment is to modify the gradient and to solve the corresponding Poisson equation. Good results were observed with the simplest possible function, a linear multiplication by a factor larger than 1 on the dark regions of the image  $\Omega$ . This yields the first presented algorithm, the “Enhanced dark” option. In the implementation, the region  $\Omega$  is defined by

$$\Omega = \{x \in R : f(x) \leq T\}, \quad (16)$$

with  $T = 50$ , by default, and the guidance vector field is defined as

$$\mathbf{V} = \begin{cases} a \nabla f & \text{in } \Omega \\ \nabla f & \text{otherwise} \end{cases}, \quad (17)$$

with  $a = 2.5$ , by default.

The default values ( $T = 50$ ,  $a = 2.5$ ) are easily explained. First of all, most eight-bit digital images appear quite dark in their parts with luminance below 50. No detail can be really seen on most screens below this value. It is therefore natural to try to enhance the contrast of the low gradients in these regions. If such dark regions are small, then the method needs not be applied, no image improvement is possible. If instead the dark regions are large, this contrast enhancement can reveal valuable visual information.  $T = 50$  is the default threshold used successfully in most treated examples. Applying this threshold in classic JPEG images to the gray level intensity yields automatically the perceptually dark regions. Now the question is: why  $a = 2.5$  as default value? Here again, we assume a decent quality image, where the noise standard deviation does not exceed 4. Then multiplying the contrast by 2.5 gives back a final noise of standard deviation less than 10, which is the limit where the noise is still tolerable for users.

In short, the default values  $T = 50$  and  $a = 2.5$  are not universal. They are motivated by the average quality of eight bit images and the average sensitivity of visual perception in dark regions. If the method had to become automatic, then the best method would be to select the level  $\lambda$  such that the level set  $\{x \mid u(x) \leq \lambda\}$  has its area equal to 25% of the image area. With these default values for  $T$ ,  $a$  the proposed algorithm can become fully automatic. It is anyway easy to change them interactively.

The second option modifies the gradient with a power  $\alpha$ , a parameter of the algorithm. We call this option as the “Enhanced global” option. In this case, the gradient is modified over the whole image and not just on dark regions. Applying a concave power reduces the large gradients and increases the small ones. By default, the power  $\alpha$  is 0.8, but values lower or greater than 1 can be applied as well. The guidance vector field is computed as

$$\mathbf{V} = |\nabla f|^{\alpha-1} \nabla f. \quad (18)$$

The proposed methods are described in algorithm 1.

---

**Algorithm 1** Intensity transformation on gray scale images

---

**input** :  $T, \alpha, a$ , original image (1 channel)

**output**: Contrast changed image

Simplest Color Balance on the original image

*Normalize the image with  $s = 0.5\%$  [5]*

Compute the gradient function vector  $\nabla f = (f_x, f_y)$

*Using the discrete cosine transform*

**if** “Enhanced dark” option **then**

**foreach** pixel  $(i, j)$  **do**

$N_4(i, j) = \{(i, j), (i+1, j), (i-1, j), (i, j+1), (i, j-1)\}$

*Compute the neighborhood*

$\Omega_d = \{(i, j) \in R : u(k, l) \leq T, (k, l) \in N_4(i, j)\}$

*Compute the discrete dark region*

    Define the guidance vector field  $\mathbf{V}$  as

$$\mathbf{V} = \begin{cases} (af_x, af_y) & \text{in } \Omega_d \\ (f_x, f_y) & \text{otherwise} \end{cases}.$$

**else**

**if** “Enhanced global” option **then**

        Define the guidance vector field  $\mathbf{V}$  as

$$\mathbf{V} = (|f_x| + |f_y|)^{\alpha-1} f_x, (|f_x| + |f_y|)^{\alpha-1} f_y;$$

Solve the Poisson equation

Simplest Color Balance on the final image

*Normalize the image with  $s = 0.5\%$  [5]*

---

From previous section, we know that the Fourier transform method is the fastest and simplest way to solve the Poisson equation with Neumann boundary conditions. We have to consider that using the inverse Fourier transform, yields the value of  $u$  at each point of the grid, defined up to a constant since the constant Fourier coefficient is arbitrary. This is algorithm 2.

Remark that these algorithms are applied on monochrome images. For RGB color images we considered two options: either apply the algorithm independently on each channel, or apply it to the intensity of the color image without modifying the original  $R/G/B$  ratio of the pixels. In the first case, the color of the pixels is modified in the process. In the second case, we first compute the gray level intensity ( $I = (R + G + B)/3$ ), then this intensity is transformed into  $I'$  by the proposed algorithm. Finally, for each pixel, the three color channels are multiplied by  $I'/I$ . This process can be summarized by algorithm 3.

Other choices for the management of color images are possible, depending on how we define the “color” attribute to be maintained (hue, chroma,  $R/G/B$  ratio) and what we want to correct with

---

**Algorithm 2** Solve the Poisson equation

---

**input** :  $\mathbf{V}$ , original image (1 channel)  
**output**: Solution image of the Poisson equation  
 Quadruplicate by symmetry the discrete domain and  $\mathbf{V}$   
 Compute the discrete Fourier transforms of  $(V_1, V_2)$   
 Compute the discrete Fourier transform of the solution  $\widehat{u_{mn}}$   
 Obtain the samples  $u_{jl}$  of the solution by the inverse discrete Fourier transform  
 Restrict them to the initial domain

---



---

**Algorithm 3** Intensity transformation while preserving  $R/G/B$  ratios

---

**input** : The vector of the gray level intensity  $I$ , the vector of the transformed intensity  $I'$ ,  
 $RGB$  channels of the original image ( $red\_in, green\_in, blue\_in$ )  
**output**:  $RGB$  channels of the output image ( $red\_out, green\_out, blue\_out$ )  
**foreach** pixel  $i$  **do**  
      $A = \frac{I'[i]}{I[i]}$  *Compute the amplification factor*  
      $red\_out[i] = A \cdot red\_in[i]$  *Compute each color channel*  
      $green\_out[i] = A \cdot green\_in[i]$   
      $blue\_out[i] = A \cdot blue\_in[i]$   
     **if**  $red\_out[i] > 255$  or  $green\_out[i] > 255$  or  $blue\_out[i] > 255$  **then**  
          $B = \max(red\_in[i], green\_in[i], blue\_in[i])$   
          $A = \frac{255}{B}$   
          $red\_out[i] = A \cdot red\_in[i]$   
          $green\_out[i] = A \cdot green\_in[i]$   
          $blue\_out[i] = A \cdot blue\_in[i]$

---

the algorithm (lightness, brightness, intensity, luma, ...). A discussion about these color correction variants will be published in a later article, and we present hereafter the simplest version.

The algorithm has been implemented using *C++* as programming language. The computation time is 2 seconds for a  $700 \times 528$  image on a Intel core i7, 2,7 Ghz and 4 Gb of RAM. Using the FFTW library, the computation complexity of one FFT is  $O(N \log N)$ , where  $N$  is the number of pixels. In both options of the algorithm we compute 6 FFT's, then, the complexity of the algorithm is  $O(N \log N)$ .

**Remark.** *The classic fast Fourier transform comes with some overhead such as the padding to the nearest power of 2, but with smart Fourier libraries, like FFTW [6], having integer values that are not powers of 2 is no more a complexity issue. Products of small factors are most efficient, but an  $O(N \log N)$  algorithm is used even for prime sizes. Likewise, we mentioned for pedagogic reasons the quadruplication. In fact, again with smart libraries like FFTW the quadruplication is implicit and performed directly as a cosine transform.*

## 6 Online Demo

An online demo permits to try Selective Contrast Adjustment on any uploaded image. The demo presents two different versions of the proposed algorithm, which are explained in Section 5. The demo has six outputs, three for the implementation on the *RGB* channels, processed independently, and three for the implementation on the intensity ( $I = (R + G + B)/3$ ), where the channels  $R, G, B$  are changed proportionally to  $I$ . The first version (enhanced dark) presents two parameters, a parameter  $T$  fixing the gray level below which the gradient of the image will be enhanced, and an enhancement factor  $a$ ; the second one has a single parameter,  $\alpha$  the power of the norm of the gradient. A previous result is presented in the demo, showing the result of the Simplest Color Balance algorithm application (see the paper of Simplest Color Balance [5] for details).

## 7 Parameter Tuning and Examples

This section examines the role of the parameters of both variants on several examples. In all examples, the displayed initial image has been previously normalized by Simplest Color Balance [5]. Unless said otherwise, next examples are processed using the intensity option (see subsection 7.4).

### 7.1 Enhanced Dark: Dark Regions Case

As we have mentioned, the automatic selection of the dark regions is a main advantage of the proposed algorithm compared to the Perez et al. algorithm [2]. Figure 1 illustrates the complex topology of the dark regions. The central image shows the dark regions (in black). Selecting them manually would be very difficult. On the right, the result of “Enhanced dark” with default parameters.

### 7.2 Enhanced Dark: Tuning $T$ Parameter

The parameter  $T$  selects the dark regions. Increasing this value can yield poorer results, as can be observed in figure 2. The left hand image in the second row shows the best result, by revealing valuable visual details. On the other results, a visual contrast loss is apparent.



Figure 1: Left: original image. Center: dark zones. Right: enhanced dark result ( $a = 2.5$ ,  $T = 50$ ).



Figure 2: First row: original image and a detail of the image. Second row: selective contrast adjustment results with several  $T$  values ( $a = 2.5$ ). Left:  $T = 50$ . Center:  $T = 100$ . Right:  $T = 150$ . Bottom: detail of the second row comparison.



### 7.3 Enhanced Dark: Tuning $a$ Parameter

The parameter  $a$  controls the enhancement of the image in the dark region. A too small value of  $a$  produces little improvement. A too large value of  $a$  produces excessive saturation in the bright image parts. The value  $a = 2.5$  seems to be optimal in most cases. It generally yields a well-contrasted image; no artificial colors appear. We can observe two examples in figure 3 and in figure 4. In figure 3, notice how the back hill is saturated with values of  $a$  greater than 3. In figure 4, the noise is revealed by high values of  $a$ . Furthermore, details such as the letters on the yellow candies disappear when the value of  $a$  increases.

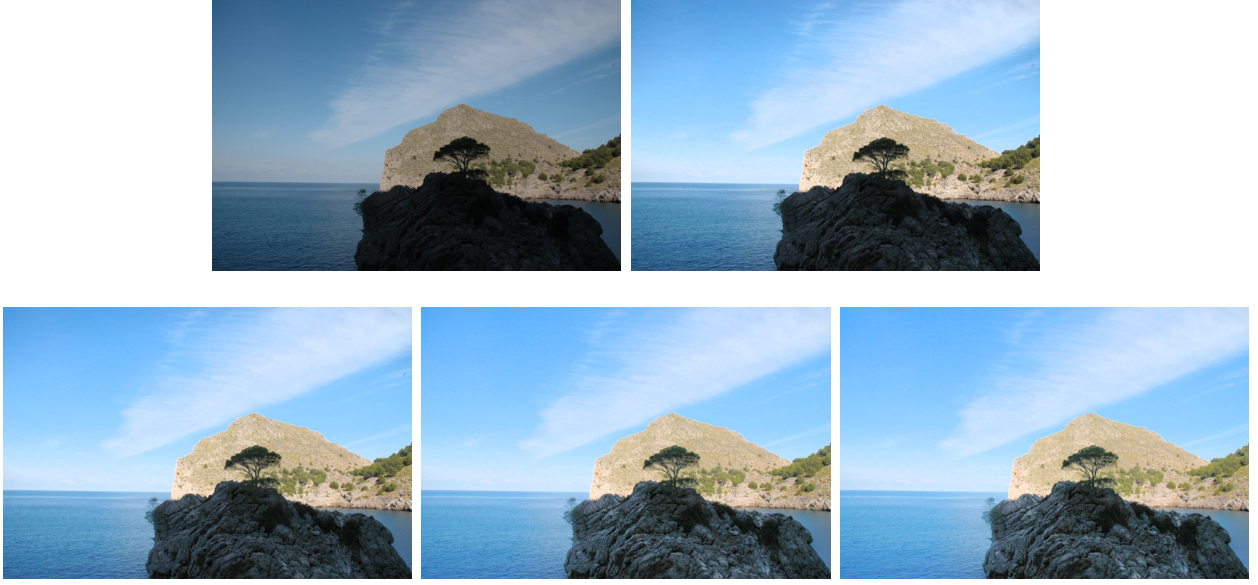


Figure 3: Variations in  $a$  parameter. Top left: original image. Selective contrast adjustment result with several values of  $a$  ( $T = 50$ ). Top right:  $a = 1.5$ . Bottom left:  $a = 2.5$ . Bottom center:  $a = 4$ . Bottom right:  $a = 5$ .

### 7.4 Enhanced Dark: Color Image Options

When dealing with  $RGB$  color images we have considered two options for the application of the algorithm: either to process independently each one of the channels ( $RGB$  option), or to process the intensity component (i.e. the average of  $R$ ,  $G$  and  $B$ ) and then compute the output  $RGB$  from the new intensity value, with the restriction that the  $R/G/B$  ratios must be kept ( $I$  option).

In the first case, the original image colors are modified (i.e. their chromatic components, hue and saturation, are changed). If the intensity component is processed, ideally the original colors are kept, since their chromaticity (understood as the  $R/G/B$  ratio) is preserved. The previous examples show the results with the  $I$  option.

Figure 5 and Figure 6 illustrate the differences between both options. As we have mentioned, the first step is the application of Simplest Color Balance [5] in the image, either independently on each the channel, or on the intensity component, depending on the selected option. This process can produce a white balance or a contrast improvement already, and the  $RGB$  option may modify the colors of the image. We can observe that the final result is better than the result of this first step, the contrast is improved and the image is enhanced. These two options are carried out because the results are very different. The choice of better option should be left to the user.



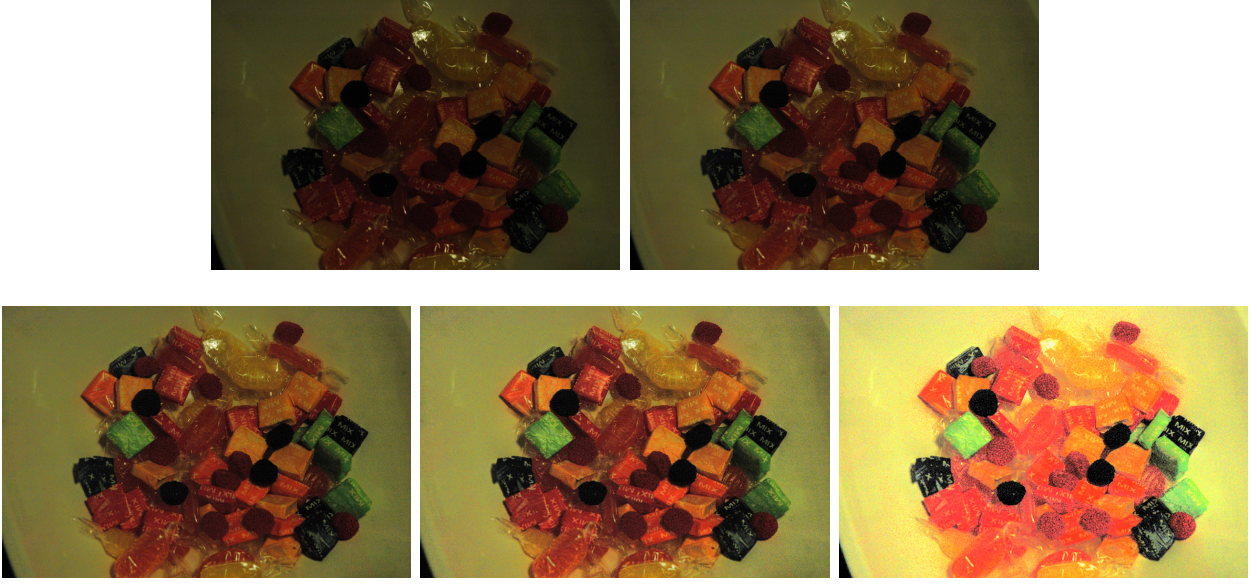


Figure 4: Variations of the  $a$  parameter. Top left: original image. Selective contrast adjustment result with different values of  $a$  ( $T = 50$ ). Top right:  $a = 1.5$ . Bottom left:  $a = 2.5$ . Bottom center:  $a = 4$ . Bottom right:  $a = 7$ .



Figure 5: Comparison between the results obtained by processing independently each channel ( $RGB$ ), and by processing the intensity component ( $I$ ), with default parameters. Top left: original image. Top center: normalized image in  $I$ , top right: result of “Enhanced dark” in  $I$ . Bottom left: normalized image in  $RGB$ . Bottom right: result of “Enhanced dark” in  $RGB$ .



Figure 6: Comparison between the results obtained by processing independently each channel ( $RGB$ ), or by processing the intensity component ( $I$ ), with default parameters. Top left: original image. Top center: normalized image in  $I$ , Top right: result of “Enhanced dark” on  $I$ . Bottom left: normalized image in  $RGB$ . Bottom right: result of “Enhanced dark” in  $RGB$ .



Although the dark regions are modified, perceptually black objects remain apparently black, as illustrated in Figure 7.

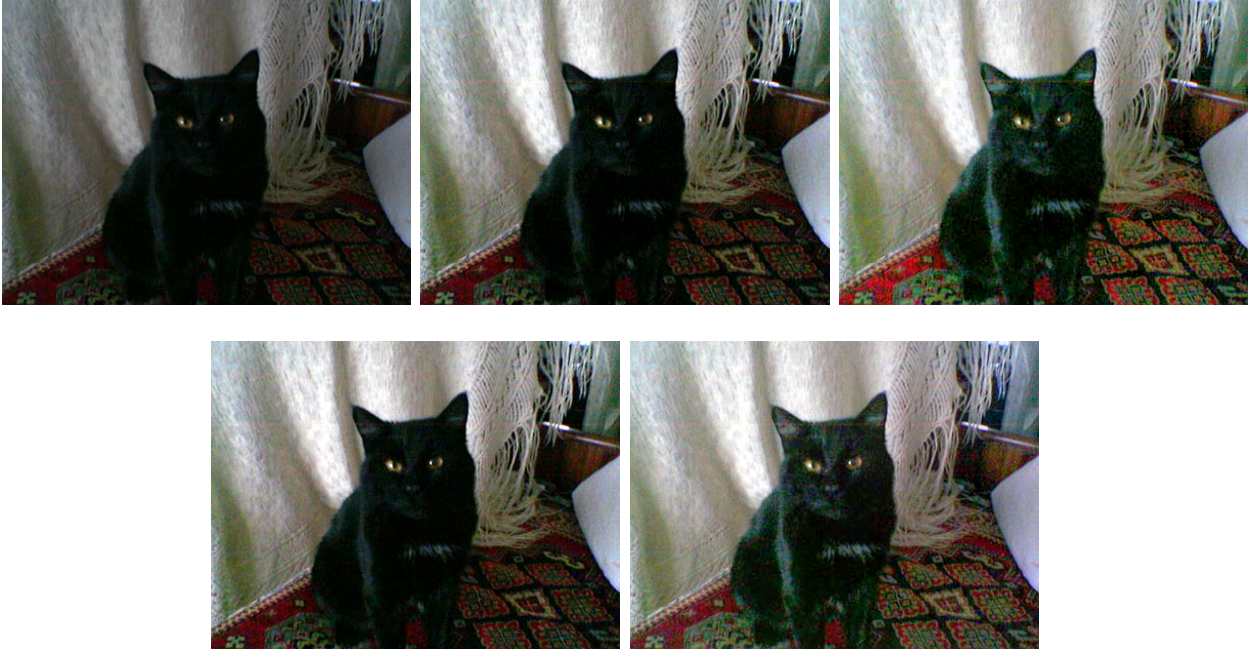


Figure 7: Comparison between the results obtained by processing independently each channel ( $RGB$ ), or by processing the intensity component ( $I$ ), with default parameters. Top left: original image. Top center: normalized image in  $I$ , top right: result of “Enhanced dark” in  $I$ . Bottom left: normalized image in  $RGB$ . Bottom right: result of “Enhanced dark” in  $RGB$ .

## 7.5 Enhanced Global: Tuning $\alpha$ Parameter

The role of the  $\alpha$  parameter is examined in figure 9. The  $\alpha$  value can be understood by observing the power function (figure 8). Figure 9 shows that small values of  $\alpha$  ( $\alpha = 0.4$  or  $0.6$ ) lighten excessively the image. The value  $\alpha = 1.2$  darkens some regions in the image, such as the left wall. Values higher than 1.2 result in a loss of gradient information. In general the 0.8 value produces good results.

On this image one can also observe the variation of the results obtained with the two color options. The color is modified in the  $RGB$  option and produces artificial colors. On the other hand, the  $I$  option maintains the true colors.

## 7.6 Comparison Between “Enhanced Dark” and “Enhanced Global”

In images with a dark region and a bright region, that is, images with zones with high contrast, the “Enhanced dark” option improves the contrast better than the “Enhanced global” option. We can observe this in Figure 10.

On images with a good contrast, where a contrast improvement is not necessary, the “Enhanced global” creates a slight improvement, but does not produce artifacts. On the other hand, for this kind of images, the “Enhanced dark” results are not convincing.

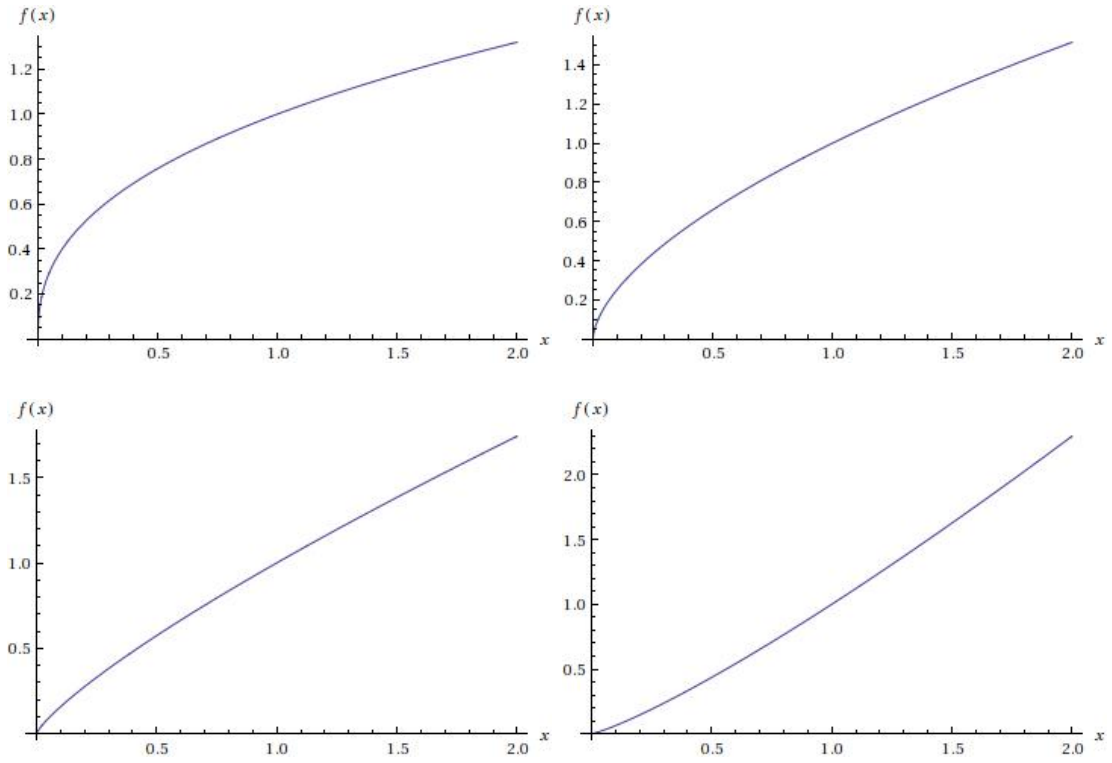


Figure 8: Images of the power function  $f(x) = x^\alpha$ , with different values of  $\alpha$ . From left to right:  $\alpha = 0.4, 0.6, 0.8, 1.2$ .

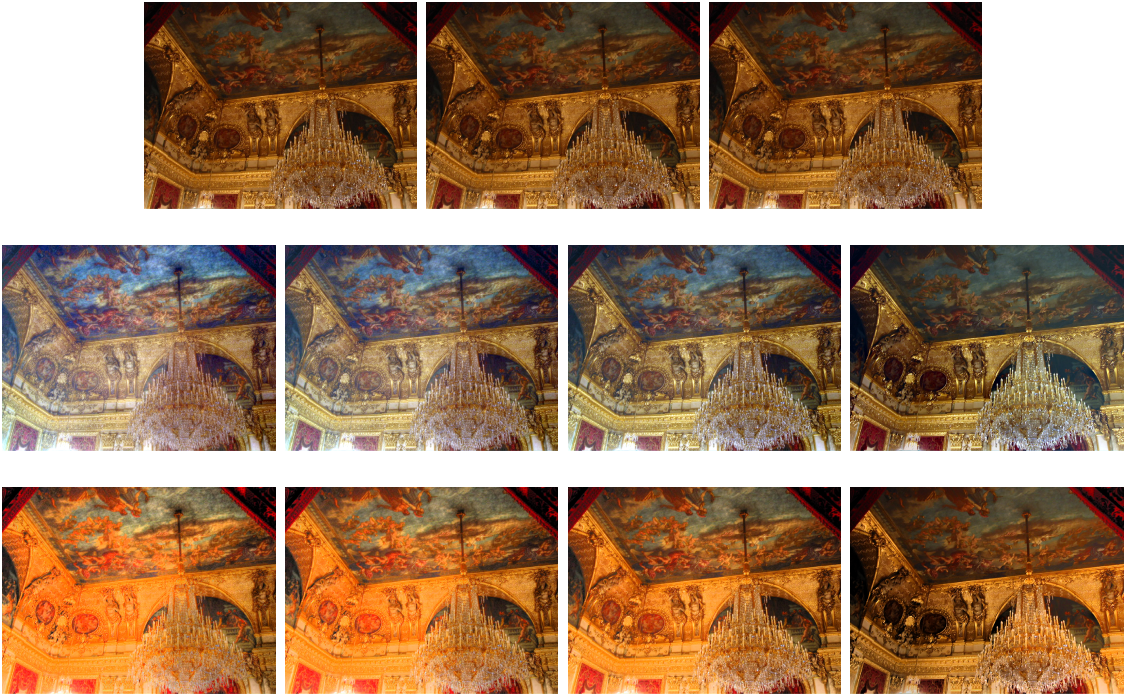


Figure 9: Comparison results of “Enhanced global” with several  $\alpha$  values. Top: Original image (left) and normalized image on  $RGB$  (center) and on  $I$  (right). Second row: result with the  $RGB$  option. From left to right:  $\alpha = 0.4, 0.6, 0.8, 1.2$ . Bottom: result with the  $I$  option. From left to right:  $\alpha = 0.4, 0.6, 0.8, 1.2$ .



Figure 10: Top: original image. Center: results on  $RGB$  option. Left: normalized image. Center: result of “Enhanced dark” with default parameters. Right: result of “Enhanced global” with default parameters. Bottom: results on  $I$  option. Left: normalized image. Center: result of “Enhanced dark” with default parameters. Right: result of “Enhanced global” with default parameters.



## 8 Conclusion

In this work, we present a fast algorithm for contrast enhancement with two variants. In “Parameter tuning and examples” section, the default value of the parameters is justified, revealing the changes produced by modifying these values. From the results of this section, also we can conclude that the choice of processing independently each one of the channels or processing the intensity component depends on the user and on the type of image. Both variants obtain good results. “Enhanced dark” enhances the contrast in dark images better than “Enhanced global”. “Enhanced global” is better to improve images of good quality.

## Acknowledgment

The authors were supported by the Ministerio de Ciencia e Innovación under Grant TIN2011-27539.

## Image Credits



Catalina Sbert CC-BY.



A.B. Petro CC-BY.



NASA<sup>2</sup>.



Photography courtesy Philip Greenspun<sup>3</sup>.

## References

- [1] J-M. Morel, A-B. Petro and C. Sbert. *Fourier Implementation of Poisson Image Editing*, Pattern Recognition Letters, Vol. 33, Is. 15, Pag.: 342-348 (2012). <http://dx.doi.org/10.1016/j.patrec.2011.10.010>.
- [2] P. Pérez, M. Gangnet and A. Blake. *Poisson Image Editing*, ACM Transactions on Image Processing. Proceedings of ACM SIGGRAPH 2003, Vol 22, Is. 3 Pag.: 313 - 318 (2003). <http://doi.acm.org/10.1145/882262.882269>.
- [3] R. Fattal, D. Lischinski and M. Werman. *Gradient Domain High Dynamic Range Compression*, ACM Transactions on Graphics, Vol 21, Issue 3 Pag.: 242 -256 (2002). <http://doi.acm.org/10.1145/566654.566573>.
- [4] N. Limare, J-M. Morel, A-B. Petro and C. Sbert. *Retinex Poisson Equation: a Model for Color Perception*, Image Processing On Line (2011). [http://dx.doi.org/10.5201/ipol.2011.lmps\\_rpe](http://dx.doi.org/10.5201/ipol.2011.lmps_rpe)
- [5] N. Limare, J-L. Lisani, J-M. Morel, A-B. Petro and C. Sbert. *Simplest Color Balance*, Image Processing On Line (2011). <http://dx.doi.org/10.5201/ipol.2011.11mps-scb>
- [6] M. Frigo and S. G. Johnson. *FFTW package*. <http://www.fftw.org/>

<sup>2</sup><http://dragon.larc.nasa.gov/retinex/>

<sup>3</sup><http://philip.greenspun.com>