



**HAL**  
open science

# Luminance-Guided Chrominance Denoising with Debiased Coupled Total Variation

Fabien Pierre, Jean-François Aujol, Charles-Alban Deledalle, Nicolas  
Papadakis

► **To cite this version:**

Fabien Pierre, Jean-François Aujol, Charles-Alban Deledalle, Nicolas Papadakis. Luminance-Guided Chrominance Denoising with Debiased Coupled Total Variation. Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR'17), Oct 2017, Venice, Italy. pp.235-248. hal-01569571

**HAL Id: hal-01569571**

**<https://hal.science/hal-01569571>**

Submitted on 27 Jul 2017

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Luminance-Guided Chrominance Denoising with Debiased Coupled Total Variation

Fabien Pierre<sup>1,2</sup>, Jean-François Aujol<sup>1\*</sup>, Charles-Alban Deledalle<sup>1</sup>, and Nicolas Papadakis<sup>1</sup>

<sup>1</sup>Univ. Bordeaux, IMB, CNRS, UMR 5251, F-33400 Talence, France.

<sup>2</sup>Univ. Bordeaux, LaBRI, CNRS, UMR 5800, F-33400 Talence, France.

**Abstract.** This paper focuses on the denoising of chrominance channels of color images. We propose a variational framework involving TV regularization that modifies the chrominance channel while preserving the input luminance of the image. The main issue of such a problem is to ensure that the denoised chrominance together with the original luminance belong to the RGB space after color format conversion. Standard methods of the literature simply truncate the converted RGB values, which lead to a change of hue in the denoised image. In order to tackle this issue, a “RGB compatible” chrominance range is defined on each pixel with respect to the input luminance. An algorithm to compute the orthogonal projection onto such a set is then introduced. Next, we propose to extend the CLEAR debiasing technique to avoid the loss of colourfulness produced by TV regularization. The benefits of our approach with respect to state-of-the-art methods are illustrated on several experiments.

**Keywords:** Colorization, denoising, color editing, color assignment

## 1 Introduction

The representation of color images in perceptual color spaces rather than in RGB has been successfully considered in color image and video editing tasks such as colorization [17], contrast enhancement [5], color transfer [16] or JPEG compression [2]. There exists many color decompositions (HSV,  $Y_C B_C R_C$ , YUV, Lab,  $L\alpha\beta$ ...) that contain channels representing information such as luminance, chrominance, saturation and hue... In the following, we will consider the YUV color space, where  $Y$  denotes the luminance channel and  $U$  and  $V$  the chrominance ones that represent the color information of the image. Indeed, the YUV space being linear, the description of the chrominance range will be easier than for non-linear space, such as Lab. Color editing then consists in modifying one or several of these channels in function of the tackled application. In the image colorization problem, the original gray-scale image is considered as luminance

---

\* J-F. Aujol is a member of Institut Universitaire de France. This study has been carried out with financial support from the French State, managed by the French National Research Agency (ANR GOTMI) (ANR-16-CE33-0010-01).

and chrominance channel are created from examples or information given by the users. In the case of contrast enhancement, the luminance channel is modified, while the chrominance is preserved.

In this paper, we will take a point of view closer to the colorization problem, since we will only edit the chrominance, while preserving the luminance. The problem of chrominance denoising arises when restoring JPEG compressed images.

**RGB range.** In the colorization approaches of the literature, as well as for JPEG restoration techniques, the range of the chrominance is a problem which is rarely solved or considered. The range is defined, for a particular luminance value, as the set of the chrominance values such that the corresponding RGB colors are in the cube  $[0, 255]^3$ .

For instance, the iterative methods of [9] or [11] use PDE schemes on the chrominance channels, but the RGB range may be violated. Such approaches generally project the produced colors into  $[0, 255]^3$ , leading to a modification of the luminance. The problem of editing color while maintaining the range has been raised in [7, 10, 12]. These methods consider the problem into the 3-dimensional RGB space or use a specific parametrization of the RGB space. Editing color while maintaining the luminance channel is nevertheless a 2-dimensional problem that should be solved in the chrominance space. The two main issues for working in the chrominance space are the definition of the range for a particular luminance value and the orthogonal projection onto this set.

This range has been characterized as a convex set within the RGB color space in [12]. For colorization purposes, the authors of [12] proposed an oblique projection to maintain the RGB range while keeping the hue of the color examples. However, the use of an oblique projection into the optimization process does not ensure to compute a global optimum of the defined functional. In this paper, we work directly onto the chrominance values and the orthogonal projection on its range is designed, leading to an algorithm minimizing the proposed model.

**TV regularization and loss of contrast.** In order to avoid visual artefacts on the edited images, many approaches have considered the use of spatial regularization. Total Variation has therefore been extended to color images in RGB space [8] and luminance/chrominance ones for colorization [13] or JPEG image decompression or restoration [1]. In this paper, we consider the coupled TV regularization [13] designed to inpaint or to denoise chrominance channels. The results obtained with this method are efficient but they suffer from the well known bias of the Rudin, Osher and Fatemi (ROF) model [14]. In the case of gray-scale images this bias is revealed as a loss of contrast. In the case of the luminance-chrominance model of [13], the loss of contrast is visible as a loss of colorfulness of the result (see, *e.g.*, Figure 1).

An automatic debiasing strategy has been recently proposed with the CLEAR method [6] in order to deal with the loss of contrast of the original ROF model. Debiasing approaches are based on Bregman distance [3] or on the projection



**Fig. 1.** The classical ROF model [14] produces a loss of contrast which is seen as a loss of colourfulness in the case of color images. The processed image (b) is drabber than the original one. The aim of this paper is to avoid this counter-effect.

onto a linear space of unbiased solutions [6], but they are not designed to handle additional hard constraint such as the preservation of the luminance.

**Outline and contributions.** In this paper, we propose an algorithm to solve the color denoising problem with fixed luminance, while taking into account the range of admissible chrominance and debiasing the original result.

To this aim, we first characterize in Section 2 the acceptable convex set of chrominance values and design an algorithm to compute an orthogonal projection onto this set. We then present in Section 3, the chrominance denoising model related to the colorization one of [13].

Next, we propose in Section 4 to use the debiasing CLEAR method [6] to get rid of the loss of colourfulness issue. As the CLEAR has been designed for unconstrained problems, we extend this technique to our constrained luminance denoising problem. The benefits of this debiasing strategy are finally demonstrated in the experimental Section 5.

## 2 Range of Chrominance

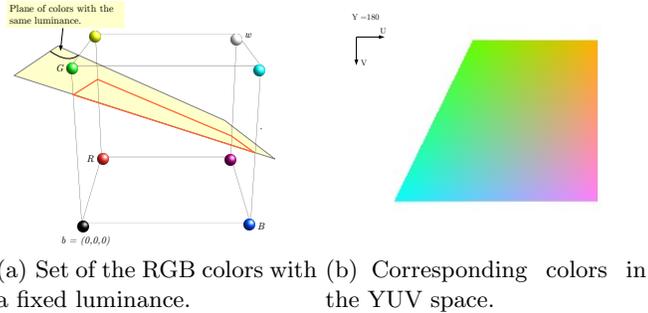
The natural problem arising when editing a color while keeping its luminance or intensity constant, is the preservation of the RGB standard range of the produced image. Most of the methods of the literature work directly in the RGB space [7, 10, 12], since it is easier to maintain the standard range. Nevertheless, working in the RGB space needs to process 3 channels, while 2 chrominance channels are sufficient to edit a color image while maintaining the luminance.

### 2.1 Description of the range

In this section, we geometrically describe the set of chrominance values which correspond to a particular luminance level, and which are contained in the RGB standard range. Let us denote by  $T(y, u, v)$  the invertible linear operator mapping YUV colors onto RGB ones.

**Proposition 1.** *Let  $y$  be a value of luminance between 0 and 255. The set of chrominance values  $(u, v)$  that satisfy  $T(y, u, v) \in [0, 255]^3$  is a convex polygon.*

*Remark 1.* For a given luminance, the chrominance values out of this polygon can be transformed into the RGB space, but they are out of the bounds. A truncation of the coordinates is usually done, but it generally changes both the luminance and the hue of the result.



**Fig. 2.** The set of the RGB colors with a particular luminance is a convex polygon. The map from RGB to YUV being affine, the set of the corresponding chrominances is also a convex polygon.

*Proof (of Proposition 1).* The intuition of the proof is given in Figure 2. The set of the colors in the RGB cube whose luminance is equal to a particular value  $y$  is a convex polygon (see, *e.g.*, [12]). Indeed, the set of colors with a particular luminance is an affine plane in  $\mathbb{R}^3$  and the intersection of the RGB cube with it is a polygon. The transformation of the RGB values into the YUV space being affine, the set of corresponding colors is thus also a convex polygon included in the set  $Y = y$ .  $\square$

## 2.2 Orthogonal projection onto the convex range

Pixel-wise, the valid chrominances are contained in a convex polygon that has, at most, 6 edges. The numerical computation of the vertex coordinates is postponed to Appendix A. When the vertices are computed, and denoted by P1, P2, *etc.*, the orthogonal projection onto the polygon is computed as follows.

The algorithm first checks if the corresponding RGB value is between 0 and 255. If so, the point is its own orthogonal projection. If not, the orthogonal projection is onto one of the edges. So, it is computed for each edges and the closest one is retained as the solution. The algorithm is summarized in Algorithm 1 and illustrated in Figure 3.

---

**Algorithm 1** Algorithm computing projection  $P_{\mathcal{R}}$ .
 

---

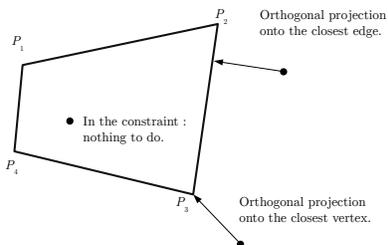
**Require:**  $X$ : chrominance vector;  $Y$  luminance value.

```

1: if  $RGB(Y, X) \notin [0, 255]^3$  then
2:   for  $i = 1 : n$  do
3:      $j \leftarrow i + 1 \bmod n$ 
4:      $\alpha \leftarrow \langle \overrightarrow{P_i P_j} | \overrightarrow{P_i X} \rangle / (\|\overrightarrow{P_i P_j}\|_2)$ 
5:     if  $\alpha > 1$  then
6:        $X_{i,j} \leftarrow P_j$ 
7:     else if  $\alpha < 0$  then
8:        $X_{i,j} \leftarrow P_i$ 
9:     else
10:       $X_{i,j} \leftarrow P_i + \alpha \overrightarrow{P_i P_j}$ 
11:    end if
12:  end for
13:   $X \leftarrow \operatorname{argmin}_{X_{i,j}} \|X - X_{i,j}\|_2$ 
14: end if

```

---



**Fig. 3.** To compute the orthogonal projection, different cases can appear. If the YUV color respects the constraint, the projection is the identity. Otherwise, the orthogonal projection onto the closest edge or vertex should be done.

### 3 Luminance-Chrominance TV-L2 Model

In the following we focus on a variational model to denoise the chrominance channels of an image while keeping the luminance unchanged. Similarly to the colorization model of [13], we want to find the minimizer  $\hat{x}(y)$  of the denoising functional:

$$\hat{x}(y) = \operatorname{argmin}_{x=(U,V)} \operatorname{TV}_{\mathfrak{C}}(x) + \lambda \|x - y\|^2 + \chi_{\mathcal{R}}(x), \quad (1)$$

with

$$\operatorname{TV}_{\mathfrak{C}}(x) = \int_{\Omega} \sqrt{\gamma \|\nabla Y\|^2 + \|\nabla U\|^2 + \|\nabla V\|^2}. \quad (2)$$

The first term is a coupled total variation which enforces the chrominance channels to have a contour at the same location as the luminance one. The fidelity data term is a classical L2 norm between chrominance channels of the unknown

$x$  and the data  $y$ . In this paper, we consider that the chrominance values live onto the convex polygon denoted by  $\mathcal{R}$  and described in the previous section. This last assumption ensures that the final solution lies onto the RGB cube, avoiding final truncation that leads to modification of the luminance channel. Model (1) is convex and can be turned into a saddle-point problem of the form:

$$\min_{x \in \mathbb{R}^2} \max_{z \in \mathbb{R}^6} \frac{\lambda}{2} \|x - y\|^2 + \langle \nabla x | z_{1,\dots,4} \rangle + \langle \gamma \nabla Y | z_{5,\dots,6} \rangle - \chi_{B(0,1)}(z). \quad (3)$$

The primal-dual algorithm [4] used to compute such saddle-point is recalled in Algorithm 2, where  $P_{\mathcal{R}}$  is the orthogonal projection described in Algorithm 1 and  $P_{\mathcal{B}}$  is defined as follows:

$$P_{\mathcal{B}}(z) = \frac{(z_{1,\dots,4}, z_{5,6} - \sigma \nabla Y)}{\max(1, \|z_{1,\dots,4}, z_{5,6} - \sigma \nabla Y\|_2)}. \quad (4)$$

---

**Algorithm 2** Minimization of (3).

---

- 1:  $x^0 = y$
  - 2:  $z^0 \leftarrow \nabla x$
  - 3: **for**  $n \geq 0$  **do**
  - 4:  $z^{n+1} \leftarrow P_{\mathcal{B}}(z^n + \sigma \nabla \bar{x}^n)$
  - 5:  $x^{n+1} \leftarrow P_{\mathcal{R}}\left(\frac{x^n + \tau(\operatorname{div}(z^{n+1}) + \lambda y)}{1 + \tau\lambda}\right)$
  - 6:  $\bar{x}^{n+1} \leftarrow 2x^{n+1} - x^n$
  - 7: **end for**
  - 8: set  $\hat{x}(y) = x^{n+1}$  and  $\hat{z} = z^{n+1}$ .
- 

The results produced by Algorithm 2 are promising but with a low data parameter  $\lambda$ , the results are drab, as illustrated in Figure 1(b).

## 4 Constrained TV-L2 Debiasing Algorithm

In this section we present a debiased algorithm for correcting the loss of colourfulness of the solution given by the optimum of (1).

### 4.1 The CLEAR method [6].

The CLEAR method [6] can be applied for debiasing estimators  $\hat{x}(y)$  obtained as :

$$\hat{x}(y) \in \operatorname{argmin}_{x \in \mathbb{R}^p} F(x, y) + G(x), \quad (5)$$

where  $F$  is a convex data fidelity term with respect to data  $y$  and  $G$  is a convex regularizer. For  $G$  being the Total Variation regularization, the estimator  $\hat{x}(y)$  is

generally computed by an iterative algorithm, and it presents a loss of contrast with respect to the data  $y$ . In the aim of debiasing this estimator, the CLEAR method refits the data  $y$  with respect to some structural information contained in the biased estimator  $\hat{x}$ . This information is encoded by the Jacobian of the biased estimator with respect to the data  $y$ :

$$J_{\hat{x}}(y)d = \lim_{\varepsilon \rightarrow 0} \frac{\hat{x}(y + \varepsilon d) - \hat{x}(y)}{\varepsilon}. \quad (6)$$

For instance, when  $G$  is the anisotropic TV regularization, the Jacobian contains the information of the support of the solution  $\hat{x}$ , on which a projection of the data can be done to estimate.

In the general case, the CLEAR method relies on the *refitting estimator*  $\mathcal{R}_{\hat{x}}(y)$  of the data  $y$  from the biased estimation  $\hat{x}(y)$ :

$$\mathcal{R}_{\hat{x}}(y) \in \operatorname{argmin}_{h \in \mathcal{H}} \|h(y) - y\|_2^2 \quad (7)$$

where  $\mathcal{H}$  is defined as the set of maps  $h : \mathbb{R}^n \rightarrow \mathbb{R}^p$  satisfying,  $\forall y \in \mathbb{R}^n$  :

$$h(y) = \hat{x}(y) + \rho J_{\hat{x}(y)}(y - \hat{x}(y)), \text{ with } \rho \in \mathbb{R}. \quad (8)$$

A closed formula for  $\rho$  can be given:

$$\rho = \begin{cases} \frac{\langle J_{\hat{x}(y)}(\delta) | \delta \rangle}{\|J_{\hat{x}(y)}(\delta)\|_2^2} & \text{if } J_{\hat{x}(y)}(\delta) \neq 0 \\ 1 & \text{otherwise.} \end{cases}, \quad (9)$$

where  $\delta = y - \hat{x}(y)$ . In practice, the global value  $\rho$  allows to recover most of the bias in the whole image domain.

An algorithm is then proposed in [6] to compute the numerical value of  $J_{\hat{x}(y)}(y - \hat{x}(y))$ . The process is based on the differentiation of the algorithm providing  $\hat{x}(y)$ .

It is important to notice that the CLEAR method applies well for estimators obtained from the resolution of unconstrained minimization problems of the form (5). Nevertheless, it is not adapted to our denoising problem (1) that contains an additional constraint  $\chi_{\mathcal{R}}(x)$  as CLEAR may violate the constraint.

## 4.2 Direct Extension of CLEAR to constrained problems.

Extending the CLEAR method to the constrained Model (1) requires to take the constraint into account in the axioms of the refitting model (7). The main difference with the original model is the addition of the constraint  $\chi_{\mathcal{R}}(x)$ . We can first notice that the refitting axioms  $h(y) = Ay + b$  for some  $A \in \mathbb{R}^{p \times n}$ ,  $b \in \mathbb{R}^p$  and  $J_h(y) = \rho J_{\hat{x}}(y)$  for some  $\rho \in \mathbb{R}$  are in line with the introduction of the constraint. In particular, the definition of the Jacobian  $J_{\hat{x}}$  in Equation (6) remains valid with the constraint, since  $\hat{x}(y)$  and  $\hat{x}(y + \varepsilon d)$  are still in the closed convex  $\mathcal{R}$ . The computation of the  $\rho$  parameter in Equation (9) may nevertheless produce,

## VIII

from Equation (8), an estimation out of the constraint, that has to be post-processed. This points out the main difference between the constrained and the unconstrained debiased estimator.

In [6], the value of  $\rho$  is computed from the minimization of a map from  $\mathbb{R}$  to  $\mathbb{R}$  defined as follows :

$$\rho \mapsto \| (I_d - \rho J_{\hat{x}(y)}) (\hat{x}(y) - y) \|_2^2. \quad (10)$$

In the case of the constrained problem, the function to be minimized is written as :

$$\rho \mapsto \|\hat{x}(y) + \rho J_{\hat{x}(y)} (y - \hat{x}(y)) - y\|_2^2 + \chi_{\mathcal{R}}(\hat{x}(y) + \rho J_{\hat{x}(y)} (y - \hat{x}(y))). \quad (11)$$

Let us denote by  $\rho$  the value defined in Equation (9). In the case when the constraint is fulfilled, *i.e.*, when  $\hat{x}(y) + \rho J_{\hat{x}(y)} (y - \hat{x}(y)) \in \mathcal{R}$ , then, the minimum of (11) is reached with  $\rho$ .

If not, let us study the function (11). This function is convex. Moreover, the value  $\rho = 0$  is in the domain of the functional because  $\hat{x}(y) \in \mathcal{R}$ . The idea is to find the maximum value of  $\rho$  such that  $\hat{x}(y) + \rho J_{\hat{x}(y)} \delta \in \mathcal{R}$ . In our case, since  $\mathcal{R}$  is a convex polygon, this computation can be done with a Ray-Tracing algorithm [15]. To this aim, we can parametrize the segment  $[\hat{x}(y), \hat{x}(y) + \rho J_{\hat{x}(y)} \delta]$ :

$$\tilde{\rho} = \max_{t \in [0,1]} t\rho \text{ such that } \hat{x}(y) + t\rho J_{\hat{x}(y)} (y - \hat{x}(y)) \in \mathcal{R}. \quad (12)$$

Equation (12) can thus be directly solved by the maximum value  $t$  such that  $\hat{x}(y) + t\rho J_{\hat{x}(y)} (y - \hat{x}(y))$  intersects the border of  $\mathcal{R}$ .

**Direct debiasing process.** Let us summarize the refitting algorithm designed for model (1). The first step consists in computing a solution of (1) with Algorithm 2. This iterative algorithm provides at convergence a first biased solution  $\hat{x}(y)$  and its dual variable  $\hat{z}$ . Once this solution has been computed, the differentiated algorithm, presented in Algorithm 3, is applied in the direction  $\delta = y - \hat{x}(y)$ . This algorithm requires the definition of the operator  $\Pi_{\hat{z}}(\tilde{z})$  which is the linearisation of the projection  $P_{\mathcal{B}}$  around  $\hat{z}$ , and reads [6]:

$$\Pi_{\hat{z}}(\tilde{z}) = \begin{cases} \tilde{z} & \text{if } \|\hat{z}\| < 1 \\ \frac{1}{\|\hat{z}\|} \left( \tilde{z} - \frac{\langle \hat{z} | \tilde{z} \rangle}{\|\hat{z}\|^2} \hat{z} \right) & \text{otherwise.} \end{cases} \quad (13)$$

Finally, the Ray-Tracing is applied to obtain  $\tilde{\rho}$  and get the debiased solution as  $\hat{x}(y) + \tilde{\rho} J_{\hat{x}(y)} (y - \hat{x}(y))$ .

Unfortunately, this direct approach does not lead to interesting results on practical cases. Indeed, if in one particular pixel the solution  $\hat{x}(y)$  is saturated, and if the debiased solution is out of  $\mathcal{R}$ , then  $\tilde{\rho} = 0$  is the unique global  $\rho$  satisfying  $\hat{x}(y) + \rho J_{\hat{x}(y)} (y - \hat{x}(y)) \in \mathcal{R}$ . Thus, in practical cases, the debiased solution is equal to the biased one.

---

**Algorithm 3** Differentiation of Algorithm 2 for computing  $J_{\hat{x}(y)}\delta$  from  $(\hat{x}(y), \hat{z})$

---

```

1:  $\tilde{x}^0 = \delta, \bar{\tilde{x}}^0 = \delta$ 
2:  $\tilde{z}^0 \leftarrow \nabla \tilde{x}$ 
3: for  $n \geq 0$  do
4:    $\tilde{z}^{n+1} \leftarrow \Pi_{\tilde{z}}(\tilde{z}^n + \sigma \nabla \bar{\tilde{x}}^n)$ 
5:    $\tilde{x}^{n+1} \leftarrow \frac{\tilde{x}^n + \tau (\text{div}(\tilde{z}^{n+1}) + \lambda \delta)}{1 + \tau \lambda}$ 
6:    $\bar{\tilde{x}}^{n+1} \leftarrow 2\tilde{x}^{n+1} - \tilde{x}^n$ 
7: end for
8:  $J_{\hat{x}(y)}\delta = \tilde{x}^{n+1}$ .

```

---

We also considered the differentiation of the projection  $P_{\mathcal{R}}$ , but it does not lead to better results since it does not guaranty that  $\hat{x}(y) + \rho J_{\hat{x}(y)}(y - \hat{x}(y)) \in \mathcal{R}$ .

Having a global  $\rho$  is thus too restrictive for our constrained problem. In the next section, we propose a model with an adaptive  $\rho$  parameter to tackle this saturated values issue.

### 4.3 Adaptive Debiasing Model for Constrained Problems.

We first notice that for a pixel  $\omega$  such that  $\hat{x}(y)_\omega + \rho J_{\hat{x}(y),\omega}(y_\omega - \hat{x}(y)_\omega)$  fulfils the constraint,  $\rho$  is the best value to refit the model according to the hypothesis of model (7). Here,  $J_{\hat{x}(y),\omega}$  denotes the value of  $J_{\hat{x}(y)}$  in pixel  $\omega$ .

On the other hand, if for a pixel  $\omega$ , the value of  $\hat{x}(y)_\omega$  and  $J_{\hat{x}(y),\omega}(y_\omega - \hat{x}(y)_\omega)$  are such that  $\hat{x}(y)_\omega + \rho J_{\hat{x}(y),\omega}(y_\omega - \hat{x}(y)_\omega) \notin \mathcal{R}$ , the  $\rho$  value has to be adapted. Thus, let us define for a pixel  $\omega$  the adapted  $\tilde{\rho}_\omega$  as follows:

$$\tilde{\rho}_\omega = \max_{t_\omega \in [0,1]} t_\omega \rho \text{ such that } \hat{x}(y)_\omega + t_\omega \rho J_{\hat{x}(y),\omega}(y_\omega - \hat{x}(y)_\omega) \in \mathcal{R}. \quad (14)$$

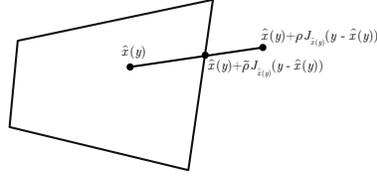
The constrained refitting model is then defined pixel-wise as:

$$R_{\hat{x}}^{\mathcal{R}}(y) = \hat{x}(y)_\omega + \tilde{\rho}_\omega J_{\hat{x}(y),\omega}(y_\omega - \hat{x}(y)_\omega) \quad (15)$$

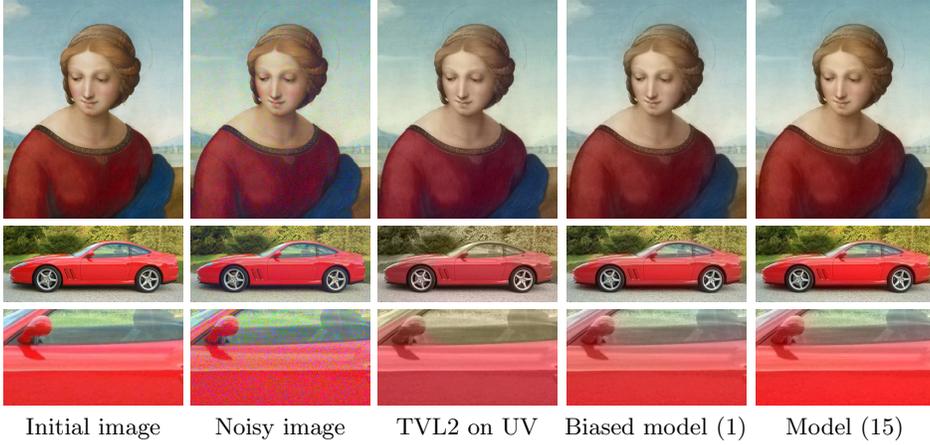
This definition ensures that the debiased estimation is in the constraint. Moreover, if the debiasing method of [6] produces an estimation in the constraint, this solution is retained. Notice however that the CLEAR hypothesis  $J_h(y) = \rho J_{\hat{x}(y)}$  for some  $\rho \in \mathbb{R}$  in model (7) is not fulfilled anymore. In numerical experiments, for most pixels, the values of  $\tilde{\rho}_\omega$  computed with our method are the same as with Model (7).

As illustrated by Figure 4, such local debiasing strategy realizes an oblique projection onto  $\mathcal{R}$ .

**Computation of the Oblique Projection.** We now describe how to compute the oblique projection when the constraint is the chrominance set for a particular value of luminance (see, *e.g.*, Section 2). To simplify the notation, we consider



**Fig. 4.** The refitting of the method of [6] may be out of the constraint. An oblique projection onto this constraint is able to respect most of hypotheses of the Model (7) while fulfilling the constraint.

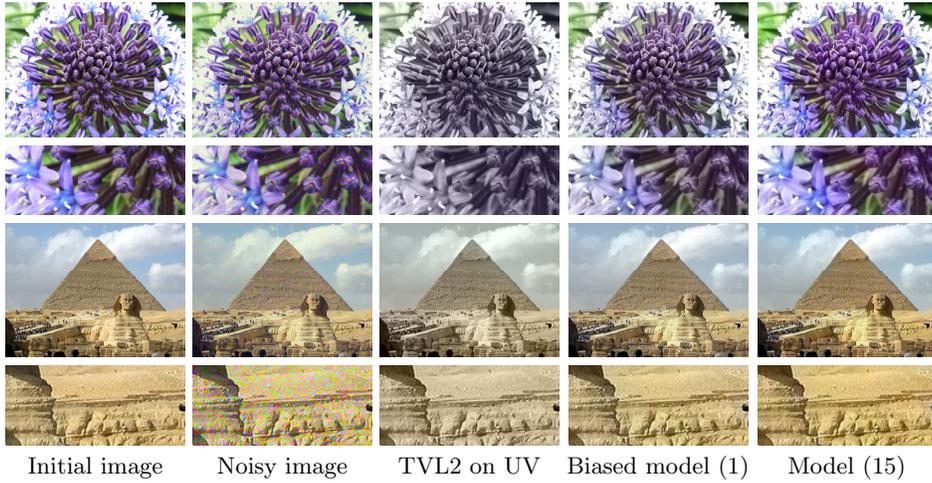


**Fig. 5.** Results of chrominance channels with a TVL2 model on chrominance, with the biased method and with the unbiased method. The debiasing algorithm produces more colorful results.

the problem for a single pixel  $\omega$  and set  $x := \hat{x}(y)_\omega$ ,  $y := J_{\hat{x}(y), \omega}(y_\omega - \hat{x}(y)_\omega)$  and  $\rho \in \mathbb{R}$  computed by the algorithm of [6].

For  $x + \rho y \notin \mathcal{R}$ , we want to compute the maximum value of  $t \in [0, 1]$  such that  $x + t\rho y \in \mathcal{R}$ . We know that  $x \in \mathcal{R}$ , thus if  $x + \rho y \notin \mathcal{R}$ , the segment  $[x, x + \rho y]$  cross one edge of the polygon.

Let us consider this problem by testing it into the RGB space. Indeed, the edges in the chrominance space correspond to edges in the RGB one, and the intersections between them correspond to intersections in the RGB space. In RGB, the problem of finding the intersection between an edge and the polygon is reduced to computing the intersection between the edge and the cube faces because the edges of the polygon are included in the cube by construction (see, *e.g.*, Figure 2(a)).



**Fig. 6.** The advantage of the coupled total variation (1) on the TVL2 model has been shown in [13]. In this work, it is refined in a better colorfulness-preserving model.

Let us denote by  $T_Y(x)$  the transformation of the chrominance values  $x = (U, V)$  to the RGB space with the luminance value  $Y$ . From the expression of the standard transformation from RGB to YUV, we have  $T_Y(x) = Y(1, 1, 1)^t + L(U, V)$  with  $L$  a linear function. We have the following equalities:

$$\begin{aligned} T_Y(x + \rho y) &= Y(1, 1, 1)^t + L(x + \rho y) \\ &= Y(1, 1, 1)^t + L(x) + \rho L(y) \\ &= T_Y(x) + \rho T_Y(y) - \rho Y(1, 1, 1)^t. \end{aligned} \quad (16)$$

We want to compute  $\tilde{\rho}$  such that  $T_Y(x + \tilde{\rho}y)$  is at the boundary of the RGB cube. To this aim, we compute the 6 different values  $\tilde{\rho}_c^v$  with  $c \in \{R, G, B\}$  and  $v \in \{0, 255\}$  corresponding to the cases where the 3 coordinates of  $T_Y(x + \tilde{\rho}y)$  are equals to 0 or 255. For instance, if the first coordinate  $R$  of  $T_Y(x + \tilde{\rho}y)$  is equal to 255, we have:

$$T_Y(x + \tilde{\rho}_R^{255}y)_R = 255 \quad (17)$$

$$T_Y(x)_1 + \tilde{\rho}_R^{255}T_Y(y)_R - \tilde{\rho}_R^{255}Y = 255. \quad (18)$$

so that

$$\tilde{\rho}_R^{255} = \frac{255 - T_Y(x)_R}{T_Y(y)_R - Y}. \quad (19)$$

For each of the six values  $\tilde{\rho}_c^v$  computed as in Equation (19), we can compute  $t_c^v = \frac{\tilde{\rho}_c^v}{\rho}$ . The values  $t_c^v$  that are between 0 and 1 correspond to an intersection of the segment  $[x, x + \rho y]$  with the boundaries of  $\mathcal{R}$ . We finally take  $t^* = \min_{t_c^v \in [0;1]} t_c^v$  and the result of Equation (14) is given by  $t^*\rho$ .

## 5 Numerical Results

In our experimental setting, the luminance channels of the initial image is preserved, while the chrominance channels are perturbed by a Gaussian noise with a standard deviation equal to 30. Let us notice that the values of the noisy images can be out of range, but our method is able to consider such issue.

Figure 5 shows comparisons between the minimization of the total variation on UV chrominance channels, the original chrominance denoising model (1) and its debiasing version (15). In general, the naive total variation on UV channels not preserve colors. It produces halo effects as well as a loss of colourfulness. In the first line, we can see that the red cloth of the Madonna is drab with Model (1), in comparison with the original image due to the bias, whereas the debiasing approach preserves the red color. Further results are presented in Figure 6. In the first line, we can see in the center of the image that the flower becomes drab with model (1), whereas it recovers its colourfulness with the debiasing one (see zoom on the second line). In the Sphinx image, the color of the sand recovers its colourfulness after debiasing.

The PSNR is a quantitative indicator to measure the quality of a denoised image. In Table 1, the PSNR of the different techniques proposed in Figures 5 and 6 are presented. These values show that the debiasing technique proposed in Section 4.3 produces images with a better PSNR, which quantitatively confirms the visual impression of Figure 5.

**Table 1.** Comparison of PSNR for techniques presented in Figures 5 and 6. This quantitative comparison highlights the quality of the results provided by the debiasing.

Image	TV on UV	Model (1)	Model (15)
Madonna	30.45	32.37	<b>33.08</b>
Ferrari	22.50	26.05	<b>27.23</b>
Purple flowers	20.88	23.29	<b>24.81</b>
Sphinx	26.36	32.95	<b>33.51</b>

## 6 Conclusion.

In this paper, we presented a model to denoise the chrominance channels of color images. The original model being biased, it produces a loss of colourfulness. By extending the CLEAR method of [6] to constrained denoising problems, we have proposed a new debiasing strategy. The denoising results obtained with our framework are promising, which is a relevant preliminary step for the application to JPEG restoration.

## A Computation of the Chrominance Range

In this section, we describe the numerical computation of the vertices of the polygon defined as the chrominance values of all the RGB colors which have a particular luminance values (see, *e.g.*, Figure 2). To this aim, the RGB coordinates of the vertices are computed and then cast into the YUV space. In the following the Algorithm which computes these RGB coordinates is described.

The computation of the vertices in the RGB space consists in intersecting the edges of the RGB cube ( $[0, 255]^3$ ) and the affine plane of the RGB colors with a particular luminance. The equation of this plane, for a luminance value equal to  $y$  is given by:

$$\mathcal{P}_Y(y) := \{(R, G, B) \text{ such that } 0.299R + 0.587G + 0.114B = y\}. \quad (20)$$

The cube having 12 edges, the resolution of 12 systems of linear equations in 3 dimension is required. This system is composed of Equation (20) and two additional equations describing the line in which the considered edge is included.

As an example, let us consider the intersection of the plane  $\mathcal{P}_Y(150)$  of colors with a luminance equal to 150 with the edge  $[(0, 0, 255), (0, 255, 255)]$ . This edge is included into the line described by the equations  $R = 0$  and  $B = 255$ . The intersection of this line with the plane  $\mathcal{P}_Y(150)$  implies the resolution of:

$$\begin{cases} 150 = 0.299R + 0.587G + 0.114B \\ R = 0 \\ B = 255. \end{cases} \quad (21)$$

The solution of this system is given by  $RGB = (0, 206, 255)$ . This vector being into the cube  $[0, 255]^3$ , the solution is a vertex of the desired polygon. When the vector is not into the cube, the solution is not a vertex of the polygon.

After computation of the vertices, the algorithm has to sort them in an order such that they represent a convex polygon. For instance, in  $\mathbb{R}^2$ ,  $(0, 0), (1, 0), (1, 1), (0, 1), (0, 0)$  is a convex polygon, whereas  $(0, 0), (1, 0), (0, 1), (1, 1), (0, 0)$  is not.

To tackle this issue, we remark that, given a particular vertex, the next one is a vertex with a common coordinate in RGB space because the polygon is defined as the intersection of a plane and the RGB cube. To avoid problem of equal points, we consider as the next vertex, the one with the most common

---

**Algorithm 4** Algorithm sorting points in the order of a convex polygon.

---

**Require:**  $P_1, \dots, P_n$   $n$  vertices onto the edges of the RGB cube.

- 1: **for**  $i = 1 : n - 1$  **do**
  - 2:     **for**  $k = i + 1 : n$  **do**
  - 3:          $\delta_k \leftarrow$  number of common coordinates between  $P_i$  and  $P_k$ .
  - 4:     **end for**
  - 5:      $k^* \leftarrow \operatorname{argmax}_k \delta_k$
  - 6:     Exchange  $P_{i+1}$  and  $P_{k^*}$
  - 7: **end for**
-

coordinates. Thus, given  $P_1, \dots, P_n$   $n$  vertices onto the edges of the RGB cube, Algorithm 4 sorts them to produce a convex polygon.

## References

1. Bredies, K.: Recovering piecewise smooth multichannel images by minimization of convex functionals with total generalized variation penalty. SFB Report 6 (2012)
2. Bredies, K., Holler, M.: Artifact-free decompression and zooming of jpeg compressed images with total generalized variation. In: Computer Vision, Imaging and Computer Graphics. Theory and Application, pp. 242–258. Springer (2013)
3. Brinkmann, E.M., Burger, M., Rasch, J., Sutour, C.: Bias-reduction in variational regularization. arXiv preprint arXiv:1606.05113 (2016)
4. Chambolle, A., Pock, T.: A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision* 40(1), 120–145 (2011)
5. Chien, C.L., Tseng, D.C.: Color image enhancement with exact hsi color model. *international journal of innovative computing, information and control* 7(12), 6691–6710 (2011)
6. Deledalle, C.A., Papadakis, N., Salmon, J., Vaiteer, S.: Clear: Covariant least-square re-fitting with applications to image restoration. *SIAM Journal on Imaging Sciences* (2017)
7. Fitschen, J.H., Nikolova, M., Pierre, F., Steidl, G.: A variational model for color assignment. In: *Scale Space and Variational Methods in Computer Vision*. pp. 437–448 (2015)
8. Goldluecke, B., Cremers, D.: An approach to vectorial total variation based on geometric measure theory. In: *IEEE Conference on Computer Vision and Pattern Recognition*. pp. 327–333 (2010)
9. Levin, A., Lischinski, D., Weiss, Y.: Colorization using optimization. *ACM Transactions on Graphics* 23(3), 689–694 (2004)
10. Nikolova, M., Steidl, G.: Fast hue and range preserving histogram specification: Theory and new algorithms for color image enhancement. *IEEE Transactions on Image Processing* 23(9), 4087–4100 (2014)
11. Peter, P., Kaufhold, L., Weickert, J.: Turning diffusion-based image colorization into efficient color compression. *IEEE Transactions on Image Processing* (2016)
12. Pierre, F., Aujol, J.F., Bugeau, A., Ta, V.T.: Luminance-hue specification in the RGB space. In: *Scale Space and Variational Methods in Computer Vision*. pp. 413–424 (2015)
13. Pierre, F., Aujol, J.F., Bugeau, A., Papadakis, N., Ta, V.T.: Luminance-chrominance model for image colorization. *SIAM Journal on Imaging Sciences* 8(1), 536–563 (2015)
14. Rudin, L.I., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena* 60(1), 259–268 (1992)
15. Williams, A., Barrus, S., Morley, R.K., Shirley, P.: An efficient and robust ray-box intersection algorithm. In: *ACM SIGGRAPH 2005 Courses*. p. 9 (2005)
16. Xiao, X., Ma, L.: Color transfer in correlated color space. In: *ACM VRCIA’06*. pp. 305–309 (2006)
17. Yatziv, L., Sapiro, G.: Fast image and video colorization using chrominance blending. *IEEE Transactions on Image Processing* 15(5), 1120–1129 (2006)