



# Improving the perception of low-light enhanced images

JAVIER VAZQUEZ-CORRAL,<sup>1,2,\*</sup> GRAHAM D. FINLAYSON,<sup>3</sup> AND LUIS HERRANZ<sup>1,2</sup>

<sup>1</sup>Computer Vision Center, Barcelona, Spain

<sup>2</sup>Computer Sciences Department, Universitat Autònoma de Barcelona, Spain

<sup>3</sup>School of Computing Sciences, University of East Anglia, NR4 7TJ Norwich, UK

\*[javier.vazquez@cvc.uab.cat](mailto:javier.vazquez@cvc.uab.cat)

**Abstract:** Improving images captured under low-light conditions has become an important topic in computational color imaging, as it has a wide range of applications. Most current methods are either based on handcrafted features or on end-to-end training of deep neural networks that mostly focus on minimizing some distortion metric—such as PSNR or SSIM—on a set of training images. However, the minimization of distortion metrics does not mean that the results are optimal in terms of perception (i.e. perceptual quality). As an example, the perception-distortion trade-off states that, close to the optimal results, improving distortion results in worsening perception. This means that current low-light image enhancement methods—that focus on distortion minimization—cannot be optimal in the sense of obtaining a good image in terms of perception errors. In this paper, we propose a post-processing approach in which, given the original low-light image and the result of a specific method, we are able to obtain a result that resembles as much as possible that of the original method, but, at the same time, giving an improvement in the perception of the final image. More in detail, our method follows the hypothesis that in order to minimally modify the perception of an input image, any modification should be a combination of a local change in the shading across a scene and a global change in illumination color. We demonstrate the ability of our method quantitatively using perceptual blind image metrics such as BRISQUE, NIQE, or UNIQUE, and through user preference tests.

Published by Optica Publishing Group under the terms of the [Creative Commons Attribution 4.0 License](#). Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

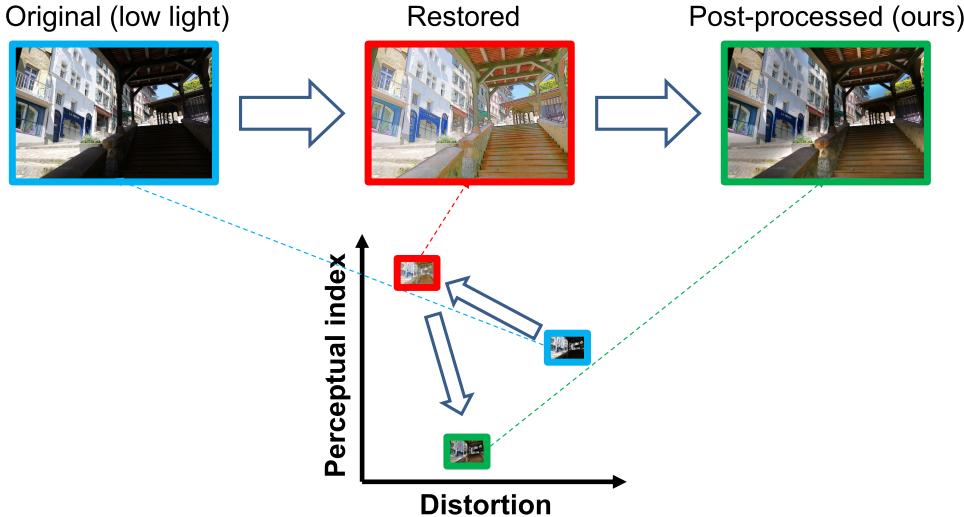
## 1. Introduction

Improving low-light images has become a topic of interest in recent years [1,2]. The goal of this problem is, given an image that has been captured on conditions of low illumination, to correct its colors so the image looks as if it was captured under ideal conditions.

This problem has been around for a number of years, but with the current trend of deep learning solutions, many of the state-of-the-art solutions are based on training an end-to-end convolutional neural network using paired data [3]. Unfortunately, these methods tend to minimize pixel-based distortion objectives, which may be achieved at the expense of perceptual quality, as suggested by the perception-distortion trade-off [4].

In this paper we wish to take the output of a low-light enhancement algorithm which has been designed to have a low distortion error but is not perceptually natural and re-compute it so that the distortion remains low but it is improved in naturalness (see Fig. 1). Our departing hypothesis is that *in order to minimally modify the perception of an input image, any modification should be a combination of a local—per pixel—change in the shading across a scene and a global change in illumination color*. Thus, given an input and output pair (where the input is taken in low-light and the output is the enhanced, low distortion, output) we find the best per-pixel shading and global light color change mapping input to output. We then apply both of them to

the input. The result is a new output that has low distortion error but looks perceptually more natural (See Fig. 1).



**Fig. 1.** General idea of our approach. Given the original low-light image and a restored image from any method, we obtain a post-processed image that is better in terms of perceptual quality than the output of the restoration.

This model allows us to decompose the low-light imaging problem as a shading problem that only affects the brightness, or magnitude, of the RGB color vectors, and a global color transformation that deals with any recoloring of the scene that is needed. Even when it may not look so evident, this decomposition is able to alleviate problems appearing in current low-light imaging methods, as we are enforcing i) a single shading for the three color channels, and ii) a single color model for the whole image. Furthermore, this formulation allows for the regularization of the shading term, for example via the use of DCT basis. In this way, the level-lines (global contrast) of the original image are better preserved. This smoothness constraint becomes key for improving our post-processing even largely.

The main contributions of our work are:

- We show how to improve low-light image enhancement algorithms in terms of perception. To do so our paper proposes a post-processing method to improve the results of current state-of-the-art models in terms of perception. Our algorithm is based on obtaining the best shading and color change mapping between the input low-light image and the output of any enhancement method.
- We present a principled way to enforce a smoothness constraint in the previously defined shading term.
- We present the results of our post-processing approach for 13 different methods, operating on 2 different datasets and we use 3 different perceptual quality metrics. As we show that our proposed post-processing is not tied to any algorithm, metric, or dataset (we demonstrate improved image quality for all combinations) we predict, therefore, that our method will work in general and can be applied to improve unseen algorithms/datasets/metrics.
- We reinforce the previous result by performing a preference test in which the observers clearly prefer our post-processed image over the not post-processed one.

A preliminary version of this work was published as [5]. While in [5] the analysis is limited to Retinex algorithms and removing their artifacts, in this work we generalize the problem to any low-light enhancement algorithm, which is enabled by looking at the problem from the perspective of perception-distortion framework as improving the perceptual quality of low-light enhancement algorithms (based on Retinex, deep networks or any other approach). In this work, we extensively validate our approach on 13 different methods, 2 different datasets, and 3 perceptual quality metrics (compared to two Retinex methods over only 20 images in [5]), and we run a users' preference test with 12 observers, 5 methods and 40 images (compared to only 15 images, 1 method, and 7 observers).

This paper is organized as follows. We start by presenting relevant related works on low-light image-enhancement. After that, we present our proposed method, which is followed by the experiments and the results section. The paper ends by summing up the conclusions.

## 2. Related work

The problem of low-light image enhancement aims at representing an image that was captured in conditions of poor illumination, as if it was properly lit at the moment of capture. This problem has attracted large interest in recent years thanks to deep learning and suitable datasets. This said, the roots of this problem can be dated back more than 40 years with the introduction of Retinex by Land [6]. The ideas of Retinex were further developed over the years, and methods such as the Multi-Scale Retinex [7], the SRIE [8], or the Milano-Retinex family approaches [9–15] were proposed.

Following the Retinex idea of representing the image as a combination of illumination and reflectance, different new models appeared, with some of the most famous being LIME [16], and NPE [17]. Other methods also looked at the possibility of fusing different exposed images, being examples of those BIMEF [18], and MF [19]. All these methods were learning-free and were based on assumptions about the statistics from natural images.

Recent years have addressed this problem using deep-learning methods. In this case, models need to be trained on large enough datasets. The first method following these ideas was Retinex-Net [3], in which the authors introduced both a network inspired in the Retinex principles, and a paired dataset allowing the training of the algorithm (LOL dataset). Thanks to the introduction of this dataset, other works started to appear shortly after. Gladnet [20] consists of two steps: first, a global module that aims at extracting the global nature of illumination and second, a module that performs the detail enhancement. KinD [21] proposed a three-modules strategy. The first module is inspired by Retinex theory and decomposes the image into reflectance and illumination. Then, each of the other two modules works specifically either on reflectance or illumination, therefore decoupling the problem into two easier ones. KinD++ [1] is an improvement over KinD in which a multi-scale attention module is proposed. Finally, and still inspired by the Retinex ideas, RUAS [2] proposes a light-weight model. More recent works such as LLFormer [22] and RetinexFormer [23] propose to use the transformer architecture for improving the low-light images.

More recently, some methods circumvent the need for paired data. Enlighten-GAN [24] proposes a GAN-based end-to-end model, while Zero-DCE [25] does not directly produce the final image, but the set of curves that are needed to be applied to the image to obtain the result. SCI [26] proposed a self-calibrated illumination learning framework.

However, these methods present a major issue: they prioritize results that either i) have image statistics that mimic theoretical ones (LIME, NPE, BIMEF), or ii) minimize the distortion with respect to the ground-truth image —for example, via PSNR. This biases these methods to present results that are not good enough in terms of perception, due to the existence of a perception-distortion trade-off, where improvements in distortion may lead to worse perceptual quality. In order to address this issue, the method of [27] added an extra module trying to tackle this issue. This new module was a U-Net minimized via adversarial learning. However, this

module is also trained and therefore depends on the structure of the recovered images, i.e. it will not work without being retrained for other architectures.

In this work, we present a principled way to address the aforementioned problem for any low-light imaging method. We do so by obtaining the result that is as closer as possible to the solution of the original method, but follows, at the same time, a physical model of formation that guarantees a superior perceptual quality.

### 3. Improving perception for low-light enhanced images

As stated in the introduction, we aim at improving the perception of state-of-the-art low-light image enhancement methods without being too harmful to their distortion. Our inspiration comes from the fact that the images of scenes that we ourselves see in the natural world remain natural as the illumination lighting the scene changes. A first approximation to this observation is to model the change in illumination as a local change in the shading across a scene and a global change in illumination color. Therefore, the hypothesis upon which we will develop our idea, is that, *in order to minimally modify the perception of an input image, any modification should be a combination of a per-pixel shading value and a global color correction transform.*

Mathematically, what we are stating is that any final result image  $\tilde{X}$  should be expressed as an  $N \times 3$  matrix

$$\tilde{X} = SYH \quad (1)$$

where  $Y$  is the original low-light image,  $H$  is the global color transform and  $S$  is the shading matrix. The image  $\tilde{X}$  is represented as a  $N \times 3$  matrix, where  $N$  is the number of pixels.  $S$  is a  $N \times N$  diagonal matrix —where each element of the diagonal is the shading element of a pixel in the image. Depending on the global color transform model, the representation of  $Y$  and  $H$  may differ slightly. If we assume a linear model,  $H$  is a  $3 \times 3$  matrix and  $Y$  is also a  $N \times 3$  matrix. If, in contrast, we consider an affine model [28],  $H$  is a  $4 \times 3$  matrix and  $Y$  is  $N \times 4$  matrix, where the additional last column consist of 1s. Operationally, both models are just linear transforms.

Our goal is then, given the image resulting from any low-light image enhancement method  $\hat{X}$  —also represented as an  $N \times 3$  matrix—, to obtain a final solution  $\tilde{X}$  as close as possible to  $\hat{X}$ , but, at the same time satisfying Eq. (1).

The most straightforward way to obtain such a solution is looking for the image  $\tilde{X}$  that minimizes

$$\|\hat{X} - \tilde{X}\| \quad (2)$$

this is equivalent, by Eq. (1), to minimize the terms  $S$  and  $H$  in

$$\min_{S,H} \|\hat{X} - SYH\| \quad (3)$$

and outputting  $\tilde{X} = SYH$  as the result of our approach.

To solve Eq. (3), we adopt an Alternating Least Squares strategy (ALS) [29]. We iteratively solve for  $S$  and  $H$  until convergence. Note that the operations computed on the ALS minimization are based on computing Least-Squares, and therefore they are not very expensive in terms of computational cost.

More in detail, the minimization of Eq. (3) presents three main steps at each iteration. Using as an example the first iteration —given the original image  $I^0 = Y$ — these steps are:

- (1) Minimizing for  $S$  (assuming  $H$  is the identity):

$$\min_{S^1} \|\hat{X} - S^1 I^0\| \quad (4)$$

- (2) Minimizing for  $H$ :

$$\min_{H^1} \|\hat{X} - S^1 I^0 H^1\| \quad (5)$$

(3) Updating the image for the next iteration:

$$I^1 = S^1 I^0 H^1 \quad (6)$$

We keep iterating over these operations until  $\|I^{i+1} - I^i\| < \epsilon$ .

Out of these three steps, the first one is the most delicate. We should recall that  $S$  is a diagonal  $N \times N$  image, where the value in each element of the diagonal represents our per-pixel shading term. When forcing  $S$  to be diagonal, the minimization of Eq. (4) can be solved in closed form as:

$$S_{jj}^1 = (I_j^0 \cdot \hat{X}_j) / \|I_j^0\|^2 \quad (7)$$

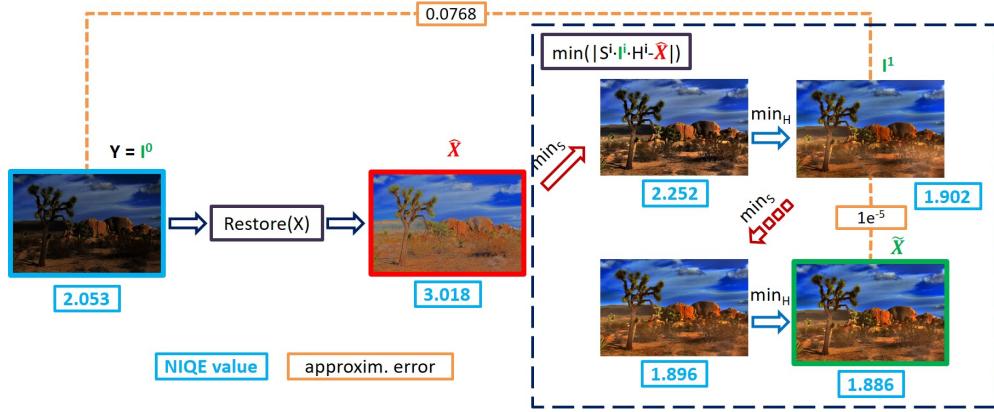
where  $I_j^0$  and  $\hat{X}_j$  are 1-by-3 vectors (the values of the row  $j$  in  $I^0$  and  $\hat{X}$ ), and  $\cdot$  denotes the scalar product.

The second step —i.e. the minimization of Eq. (5)— does not require any further constraint and can therefore be solved in closed form by:

$$H^1 = (S^1 I^0)^+ \hat{X} \quad (8)$$

where  $+$  denotes the pseudo-inverse.

In Algorithm 1 we present a pseudocode implementation of our approach, and a visual explanation is shown in Fig. 2.



**Fig. 2.** Visual explanation of our method. Given the low-light image  $Y$  and the result of some method  $\hat{X}$  we look for the image that best minimizes the distance to the method's solution, but that, at the same time, it is just a local shading plus a global color transformation aware from the original low-light image. We solve for this minimization by using Alternative Least-Squares and minimizing alternatively for  $S$  and  $H$  until convergence.

### 3.1. Enforcing shading smoothness

We would like  $S$  to be smooth in the sense that proximate image pixels should be similar to each other so that the derivative of the shading image should be bounded. A simple way to enforce smoothness is to represent the shading as a linear combination of the first  $k$  terms in a DCT expansion. For convenience we now represent  $S$  instead of a diagonal matrix, as a 2D image  $S$  where  $S = \text{diag}(\text{vec}(S))$ , where the operator  $\text{vec}()$  maps a matrix to a vector (where we stack the columns of a matrix on top of each other) and the operator  $\text{diag}()$  turns a vector into a diagonal

matrix. The 2D DCT expansion then becomes

$$\mathcal{S}(x, y) = \sum_{p=1}^k \alpha_p C_p(x, y) \quad (9)$$

where  $(x, y)$  represents a particular pixel in the image,  $C_p(x, y)$  denotes the  $p$ th term in an order  $k$  DCT expansion (described next) and  $\alpha_p$  is a scalar weighting coefficient. The  $k$ - coefficients together are denoted by the vector  $\underline{\alpha}$ .

---

**Algorithm 1** ALS algorithm for the post-processing of low-light image methods
 

---

- **Input:** The original image  $Y$  and the result of a low-light method  $\hat{X}$ .
  - Define  $I^0 = Y, i = 0$
  - repeat over  $i$ 
    1.  $\min_{S^i} \|\hat{X} - S^i I^i\| \longrightarrow S^i_{jj} = (I_j^i \cdot \hat{X}_j) / \|I_j^i\|^2$
    2.  $\min_{H^i} \|\hat{X} - S^i I^i H^i\| \longrightarrow H^i = (S^i I^i)^+ \hat{X}$
    3. Update  $I$ .  $I^{i+1} = S^i I^i H^i$
    4. Update  $i$ .  $i = i + 1$  - Until  $\|I^{i+1} - I^i\| < \epsilon$ , where  $\epsilon$  is a predefined tolerance value.
  - **Output:**  $\tilde{X} = I^{i+1}$
- 

We adopt the standard ordered DCT expansion. Thus,  $C_1(x, y) = 1$  (the constant function). The next two terms are cosines of half-periodicity in the x- and y- dimensions e.g. for an  $n \times m$  image size,  $C_2(x, y) = \cos(\frac{\pi x}{n})$  and  $C_3(x, y) = \cos(\frac{\pi y}{m})$ . The 3rd, 4th and 5th terms are  $C_4(x, y) = \cos(\frac{2\pi x}{n})$ ,  $C_5(x, y) = \cos(\frac{\pi x}{n})\cos(\frac{\pi y}{m})$  and  $C_6(x, y) = \cos(\frac{2\pi y}{m})$ . Clearly, as periodicity increases, there is a natural number of basis functions used e.g.  $k \in \{1, 3, 6, 10, 15, \dots\}$ . For more details of the 2D cosine expansion see [30].

To fit in with how our optimizations were written before, we can rewrite Eq. (9) as

$$\mathcal{S} = \text{diag}(C\underline{\alpha}) \quad (10)$$

where  $C$  is an  $N \times k$  matrix ( $N = m \times n$  the number of pixels in an image).

Our minimization for  $S$  —the first step, which assumes  $H$  is the identity— becomes

$$\min_{\underline{\alpha}, H} \|\hat{X} - [\text{diag}(C\underline{\alpha})]Y\| \quad (11)$$

To solve for  $\underline{\alpha}$  requires a little algebraic manipulation. Let  $\underline{\alpha}(j)$  denote the  $k$ -vector that is all zeros except the  $j$ th component which is equal to 1, e.g.  $\underline{\alpha}(3) = [0 \ 0 \ 1 \ 0 \ 0 \ 0]^T$  for  $k = 6$ .

Let us define the matrix  $\mathcal{V}$  that has  $k$  columns:  $\underline{V}_j$  ( $j = 1, 2, \dots, k$ ) as

$$\underline{V}_j = \text{vec}(\text{diag}(C\underline{\alpha}(j))Y). \quad (12)$$

Now, we find  $\underline{\alpha}$  in Eq. (11) by minimizing:

$$\min_{\underline{\alpha}} \|\text{vec}(\hat{X}) - \mathcal{V}\underline{\alpha}\| \quad (13)$$

Clearly, the least-squares solution to Eq. (13) is  $\mathcal{V}^+ \text{vec}(\hat{\mathbf{X}})$ . Thus, to use the DCT basis in Algorithm 1, line 1 of the pseudo code becomes:

$$\min_{\underline{\alpha}^i} \|\text{vec}(\hat{\mathbf{X}}) - \mathcal{V}^i \underline{\alpha}^i\| \quad (14)$$

where  $\mathcal{V}^i$  is calculated per iteration (according to Eq. (12)) and  $\underline{\alpha}^i$  is the least-squares solution for iteration  $i$ .

## 4. Experiments and results

### 4.1. Methods considered

In this paper, we evaluate our method using 16 different methods. 7 of them are famous image enhancement methods that are not based in deep learning approaches, namely SRIE [8], BIMEF [18], LIME [16], MF [19], NPE [17], MSR [7], and MSRCR [31]. The other 9 are recent deep-learning based methods: RetinexNet [3], GLADNet [20], KinD [21], KinD+ [1], Zero-DCE [25], RUAS [2], SCI [26], LLFormer [22], and RetinexFormer [23].

We also compare 4 different instances of our method (different values of the parameter  $k$  that sets the number of DCT basis): the results of not using the DCT constraint explained in the algorithm (equivalent to  $k = \#\text{pixels}$ ), and the results obtained by supposing 15 basis—DCT of order 5, i.e. any combination of horizontal and vertical basis that combined are of order 5 or less—, 28 basis—DCT of order 7—, and 55 basis—DCT of order 10. Let us note that we also discuss in this section the influence of this parameter.

In the main results presented in this section we have used the affine version of our method, i.e. we consider  $H$  as a  $4 \times 3$  matrix. As before, we later present a comparative analysis of this selection with respect to using a  $3 \times 3$  matrix, and also with respect to adding root-polynomial coefficients in the minimization [32].

### 4.2. Dataset

We consider two challenging datasets for the low-light image enhancement problem. The first dataset is the set of 23 images captured by Vonikakis [33]. The second dataset is built from the SICE dataset [34]. The full SICE dataset is comprised of two sets of scenes with different number of images at different illumination levels. For this work, we have selected the lowest illuminated image for each scene from the first set. Therefore, our dataset comprises a total of 360 images drawn from SICE.

We have selected these two datasets because they are used in many low-light image enhancement papers as testing data. Moreover, these two datasets are unpaired, and therefore the deep-learning methods do not present the advantage of being trained in them—as it would be the case if we used the well-known LOL dataset [3], or the low-light versions of the MIT-5K one [35].

As a cautious remark, we want to make clear that in order to be able to run some deep-learning based methods, we were required to diminish the size of the images for those methods. This fact does not allow us to compare the results of different methods among them; but we remind the reader that that was not our goal: our goal was to show how our approach improves each of the methods individually. This can be done even with the size modification as our method will also accept the reduced size image as input.

### 4.3. Metrics

We will consider 3 different metrics that are based on the perception of images, and that therefore do not consider any ground-truth data. We do not compare against distortion-aware metrics such as PSNR or SSIM because: i) As pointed in Fig. 1 for improving the perception of the image we will require to harm the distortion metrics—we are navigating the perception-distortion

trade-off—and ii) the datasets used are unpaired—as explained on the previous subsection—, so there are not reference images to compute the distortion.

By doing the aforementioned selection of metrics, we are following the seminal perception-distortion work from Michaeli and Blau [4]. From the perception-distortion tradeoff perspective,  $\text{distortion } \Delta(X, \hat{X})$  refers to full-reference measures that quantify the discrepancy between the restored image  $\hat{X}$  given the ground-truth reference image  $X$ —metrics such as PSNR or SSIM. In contrast, *perception* (or perceptual quality metrics) refers to no-reference measures that quantify the degree to which an image  $\hat{X}$  looks like a natural image (i.e. in normal lighting conditions in our case). Perceptual metrics are typically computed based on deviations from natural image statistics, so we can consider the discrepancy  $d(p_X, p_{\hat{X}})$  as a measure of the perceptual quality or perception.

Therefore, the list of metrics considered is:

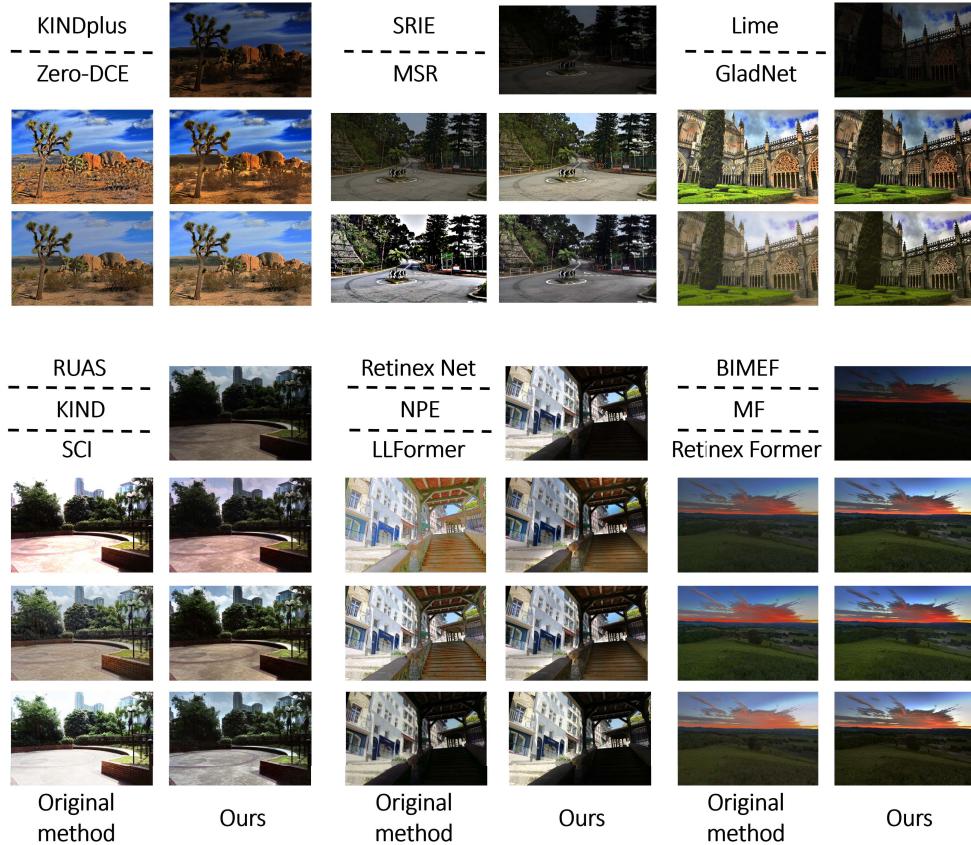
- *NIQE* [36]. This metric predicts a score in which lower means better. NIQE only makes use of measurable deviations from statistical regularities observed in natural images, and therefore, it does not need training on human-rated distorted images. Let us note that NIQE was already used as a perception metric in the original perception-distortion trade-off paper [4].
- *BRISQUE* [37]. This metric predicts a score in which lower means better. The metric is trained on a set of both corrupted and the corresponding pristine images, aiming at minimizing the differential mean opinion score (DMOS) values provided by observers.
- *UNIQUE* [38]. This metric predicts a score in which higher means better. This metric was developed by first sampling pairs of images from individual IQA databases, and computing a probability that the first image of each pair is of higher quality. This probability was later employed together with a fidelity loss to optimize a deep neural network by using a large number of such image pairs. We also included this metric since it is based on deep-learning.

#### 4.4. Qualitative observations

Figure 3 shows results for our approach, given any of the considered methods. We recommend the reader to zoom-in on the image to get a better understanding of our results. This Figure can be decomposed in 6 different parts (2 rows of 3), in which each of them presents the name of the methods in this part (top-left), the original image (top-right), the original results of the methods (center- and bottom-left) and our results for those original results (center- and bottom-right). Our results are computed using a  $4 \times 3$  matrix  $H$ , and considering the smoothness constraint (using the DCT basis for the shading term). There are some important points to remark from this Figure.

First, as it can be seen—especially in the top-left, bottom-left and top-right cases—, our method adapts the colors to the method that we start with. In the top-left example we can see how our result for the KinD+ method outputs a more orange sand and rocks than our result for the Zero-DCE case, following what happens in the results of these methods. Similarly, in the top-right case, the color of our result in the church for the case of Lime is more orange-red than for the case of GladNet, as it occurs in the original method’s results. Finally, in the bottom-left case, our method outputs a redder floor for the case of RUAS, in comparison to the cases of KinD and SCI, as is also the case with the original methods. This verifies our idea, in which we aim to output the closest possible solution to the selected method, as far as it accomplishes our model of just applying a local shading plus global color modification to the original image.

Second, we want to remark on how our method improves the naturalness of contrast in the images. Some clear examples are the contrast for the plants in the Kind and Kind+ cases, or the



**Fig. 3.** Results for our method when starting with different original ones. In each of the 6 blocks, we have the methods compared and the original image in the top row. In the other rows we show the result of the methods (left) and our post-processing results (right).

contrast between the sky, the clouds and the sunset for BIMEF and MF. In all these cases our proposed method aligns better with the natural gradients between the different stated above.

Also, our method is able to avoid artifacts that appear from over enhancement. Examples of this case are the stairs in the examples of Retinex-Net and NPE, the black areas behind the right-side arches in the GladNet case, and the wall in the MSR case. Finally, our method is able to bring back parts of the image that have been converted in highlight—as the clouds in the case of RUAS.

#### 4.5. Quantitative results

In this subsection we will discuss the quantitative results in each of the above explained metrics. In each of the Tables the best result is highlighted in green, and all the cases in which our approach outperforms the original methods are shown in bold.

##### 4.5.1. Results NIQE metric

The mean of the NIQE metric calculated over all the images is presented in Table 1. Let us recall that in this case lower values mean better results. We can see that in the case of the Vonikakis dataset our results improve as we increase the number of basis of the DCT for most of the cases. The trend is not exactly the same for the SICE dataset, as the best results in this case are obtained

when using 15 basis. We can also see that in the case of the Vonikakis dataset, our results are better than the original in 57 out of the 64 cases studied (89.1%) and we tie in 4 other cases (6.2%), while in the SICE dataset this is the case in 50 out of the 64 cases (78.1%), therefore validating the usefulness of our approach with regards to this widely used metric. The only downfall is the case of the SRIE method with the SICE dataset, in which our method does not improve the original result —although it follows extremely close to it (0.05-0.07 distance)—, in comparison with the same method in the Vonikakis dataset where we obtain an improvement of up to 0.24-0.30.

**Table 1. Results as the mean NIQE for all the images. Lower numbers represent better results. In green the best result for each case. In bold the cases in which our approach outperforms the original method result. First block: Traditional approaches. Second block: Deep Learning approaches.**

	Vonikakis dataset					SICE dataset					
		Our approach					Our approach				
		Original	No DCT	15 basis	28 basis		Original	No DCT	15 basis	28 basis	
SRIE		2.10	<b>2.10</b>	<b>1.86</b>	<b>1.82</b>	<b>1.80</b>	3.15v	3.38	3.22	3.21	3.20
BIMEF		1.93	<b>1.93</b>	<b>1.87</b>	<b>1.83</b>	<b>1.80</b>		3.26	3.51	<b>3.22</b>	<b>3.20</b>
LIME		2.24	<b>2.17</b>	<b>1.85</b>	<b>1.83</b>	<b>1.82</b>		3.98	<b>3.88</b>	<b>3.12</b>	<b>3.11</b>
MF		1.95	<b>1.95</b>	<b>1.85</b>	<b>1.82</b>	<b>1.78</b>		3.87	4.05	<b>3.16</b>	<b>3.16</b>
NPE		2.03	2.04	<b>1.89</b>	<b>1.85</b>	<b>1.80</b>		3.65	3.69	<b>3.13</b>	<b>3.11</b>
MSR		2.60	<b>2.44</b>	<b>1.80</b>	<b>1.79</b>	<b>1.79</b>		5.12	<b>4.78</b>	<b>3.13</b>	<b>3.13</b>
MSRCR		2.07	<b>2.06</b>	<b>1.81</b>	<b>1.82</b>	<b>1.84</b>		3.39	<b>3.37</b>	<b>3.05</b>	<b>3.06</b>
RetinexNET		3.56	<b>3.56</b>	<b>2.39</b>	<b>2.38</b>	<b>2.37</b>		4.27	4.36	<b>3.08</b>	<b>3.06</b>
GLADNET		2.50	2.54	2.52	<b>2.49</b>	<b>2.45</b>		3.15	3.22	3.17	<b>3.13</b>
KIND		2.67	<b>2.44</b>	<b>2.36</b>	<b>2.35</b>	<b>2.40</b>		3.58	<b>3.44</b>	<b>3.49</b>	<b>3.49</b>
KIND+		2.73	<b>2.47</b>	<b>2.40</b>	<b>2.38</b>	<b>2.33</b>		3.78	<b>3.72</b>	<b>3.49</b>	<b>3.51</b>
Zero-DCE		2.10	<b>2.01</b>	<b>1.97</b>	<b>1.93</b>	<b>1.88</b>		3.44	3.64	<b>3.23</b>	<b>3.23</b>
RUAS		3.48	<b>1.92</b>	<b>2.04</b>	2.43	2.43		3.7	<b>3.55</b>	<b>3.35</b>	<b>3.43</b>
SCI		3.02	<b>2.64</b>	<b>2.39</b>	<b>2.50</b>	<b>2.60</b>		4.01	4.06	<b>3.65</b>	<b>3.68</b>
LLFormer		3.19	<b>3.07</b>	<b>2.40</b>	<b>2.38</b>	<b>2.34</b>		5.33	<b>5.09</b>	<b>3.62</b>	<b>3.66</b>
RetinexFormer		2.70	<b>2.67</b>	<b>2.66</b>	<b>2.68</b>	<b>2.68</b>		3.15	<b>3.09</b>	3.53	3.55

#### 4.5.2. Results BRISQUE metric

The mean of the BRISQUE metric calculated over all the images is presented in Table 2. In this case lower values mean better results. Looking at the results for this metric, we can see how for both of the datasets better results are obtained with a larger number of DCT bases. From the table, we can also see that our approach improves in the large majority of cases studied, namely 54 out of 64 cases (84.4%) in the Vonikakis dataset and 53 out of 64 cases (82.9%) in the SICE dataset.

#### 4.5.3. Results UNIQUE metric

The mean of the UNIQUE metric calculated over all the images is presented in Table 3. Within this metric, higher values represent better results. Looking at the results of the Table, in this case it is difficult to assess which number of DCT basis is better. This said, we can see that as in the previous two cases our method outperforms the original image in most of the computed cases, namely 44 out of 64 (68.8%) —and we tie 1 other time (1.5%)— in the Vonikakis dataset and 39 out of 64 cases (60.9%) —and we tie 6 other times (9.4%)— in the SICE dataset. The

**Table 2.** Results as the mean BRISQUE for all the images. Lower numbers represent better results. In green the best result for each case. In bold the cases in which our approach outperforms the original method result. First block: Traditional approaches. Second block: Deep Learning approaches.

	Vonikakis dataset					SICE dataset				
		Our approach					Our approach			
		Original	No DCT	15 basis	28 basis	55 basis	Original	No DCT	15 basis	28 basis
SRIE	23.80	<b>23.19</b>	<b>22.97</b>	<b>22.51</b>	<b>21.82</b>	28.36	28.59	<b>27.65</b>	<b>27.16</b>	<b>26.84</b>
BIMEF	24.11	<b>23.48</b>	<b>23.18</b>	<b>22.41</b>	<b>21.83</b>	29.67	30.02	<b>27.99</b>	<b>27.63</b>	<b>27.27</b>
LIME	24.00	<b>22.28</b>	<b>21.26</b>	<b>20.94</b>	<b>20.90</b>	30.88	<b>30.21</b>	<b>26.53</b>	<b>26.26</b>	<b>25.88</b>
MF	23.34	<b>22.76</b>	<b>22.55</b>	<b>21.82</b>	<b>20.97</b>	29.79	30.37	<b>27.52</b>	<b>27.33</b>	<b>27.11</b>
NPE	22.25	<b>22.20</b>	23.50	23.41	22.56	30.16	<b>30.10</b>	<b>27.03</b>	<b>26.69</b>	<b>26.28</b>
MSR	38.02	<b>36.19</b>	<b>20.16</b>	<b>18.18</b>	<b>18.49</b>	40.37	<b>38.50</b>	<b>26.06</b>	<b>25.97</b>	<b>25.80</b>
MSRCR	21.22	<b>20.78</b>	<b>21.07</b>	<b>20.06</b>	<b>19.64</b>	26.53	<b>26.21</b>	<b>24.73</b>	<b>24.48</b>	<b>24.36</b>
RetinexNET	17.95	18.04	19.33	<b>17.55</b>	<b>16.46</b>	28.44	29.73	<b>26.22</b>	<b>25.71</b>	<b>25.36</b>
GLADNET	20.82	21.22	<b>18.33</b>	<b>19.45</b>	<b>19.19</b>	26.04	<b>26.04</b>	27.39	27.05	26.57
KIND	28.35	<b>23.86</b>	<b>17.22</b>	18.33	<b>18.30</b>	28.29	<b>26.77</b>	<b>22.26</b>	<b>22.24</b>	<b>22.47</b>
KIND+	25.94	<b>23.61</b>	<b>18.07</b>	<b>17.61</b>	<b>15.75</b>	27.37	<b>26.38</b>	<b>22.23</b>	<b>22.14</b>	<b>22.26</b>
Zero-DCE	25.40	<b>25.14</b>	<b>24.92</b>	<b>23.56</b>	<b>22.91</b>	29.95	30.24	<b>27.81</b>	<b>27.83</b>	<b>27.57</b>
RUAS	38.31	<b>28.47</b>	<b>23.30</b>	<b>24.41</b>	<b>24.41</b>	32.56	<b>30.42</b>	<b>28.06</b>	28.22	<b>28.85</b>
SCI	21.73	<b>15.62</b>	<b>17.73</b>	<b>19.83</b>	<b>21.03</b>	26.54	<b>25.24</b>	<b>22.89</b>	<b>22.96</b>	<b>23.61</b>
LLFormer	24.95	25.05	<b>18.23</b>	<b>17.13</b>	<b>16.60</b>	29.37	<b>27.85</b>	<b>24.15</b>	<b>24.64</b>	<b>24.82</b>
RetinexFormer	20.35	<b>19.54</b>	20.45	20.45	21.57	21.69	<b>20.66</b>	21.77	21.72	21.73

only downfall are the results for BIMEF and MSRCR in the SICE dataset and the results of for RetinexFormer in the Vonikakis dataset, in which our method is not able to outperform the method's solution. This said, as it was the case with the combination of the SRIE method with the NIQE metric, our improvement for BIMEF and MSRCR in the Vonikakis dataset is higher than the decrement in the SICE dataset (a decrement of 0.001 and 0.003 in comparison to an improvement of 0.002 and 0.02). In the other case—RetinexFormer—increment are decrement are equal (0.002).

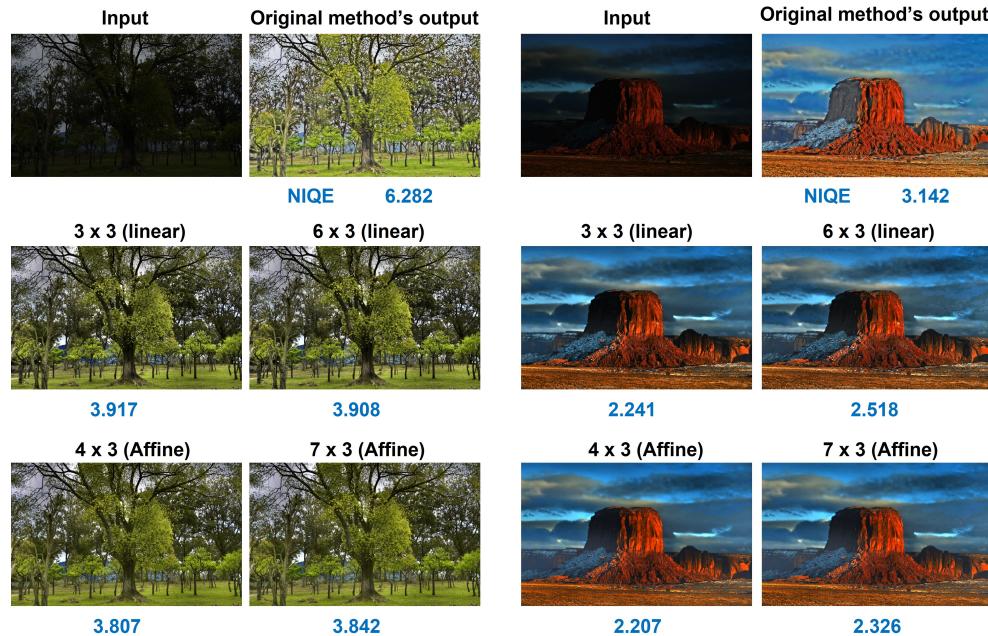
#### 4.6. On the selection of $H$

In this subsection we study how the selection of  $H$  affects our results. As stated before, all the results we have discussed were computed by defining  $H$  as an affine  $4 \times 3$  transformation. Here, we will show some visual comparisons in how the results would look when selecting a linear  $3 \times 3$  matrix, and how they will look if we add root-polynomials coefficients [32], therefore converting  $H$  into either a  $6 \times 3$  (linear) or  $7 \times 3$  (affine) matrix.

Figure 4 shows the aforementioned cases for two different images. The input low-light image and original output of the method to improve are shown in the top row, while our different corrections are shown in the other rows. From this figure we can extract two different conclusions. First, we can see how using the affine transformation allows us to get colors that are closer to the original method's output. This is clear when looking at the orange tone of the mountain in the right example, and to the green of the trees in the left example. Also, the addition of the root-polynomials does not present any advantage, rather the opposite, as the extra degrees of freedom we concede to the minimization may output some of the original's image artifacts.

**Table 3.** Results as the mean UNIQUE for all the images. Higher numbers represent better results. In green the best result for each case. In bold the cases in which our approach outperforms the original method result. First block: Traditional approaches. Second block: Deep Learning approaches.

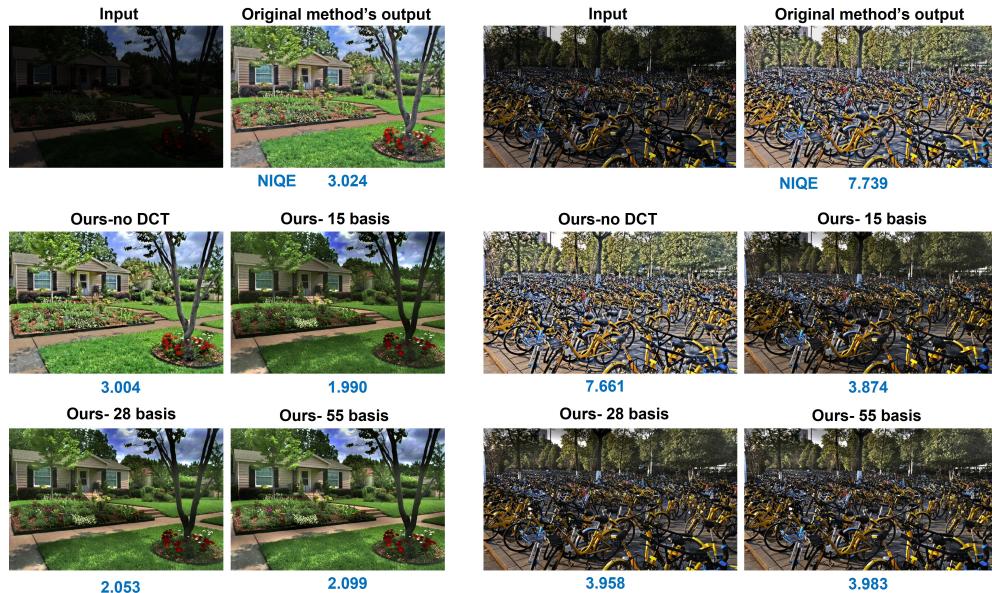
	Vonikakis dataset					SICE dataset					
	Our approach						Our approach				
	Original	No DCT	15 basis	28 basis	55 basis	Original	No DCT	15 basis	28 basis	55 basis	
SRIE	0.830	<b>0.831</b>	<b>0.835</b>	<b>0.844</b>	<b>0.839</b>	0.791	<b>0.794</b>	<b>0.799</b>	<b>0.798</b>	<b>0.799</b>	
BIMEF	0.842	<b>0.844</b>	0.833	<b>0.842</b>	0.840	<b>0.801</b>	0.800	0.797	0.796	0.797	
LIME	0.804	<b>0.805</b>	<b>0.834</b>	<b>0.845</b>	<b>0.834</b>	0.786	<b>0.787</b>	<b>0.787</b>	<b>0.786</b>	<b>0.789</b>	
MF	0.840	<b>0.841</b>	0.828	0.837	0.836	0.795	<b>0.795</b>	<b>0.795</b>	<b>0.796</b>	<b>0.797</b>	
NPE	0.827	<b>0.833</b>	<b>0.840</b>	<b>0.853</b>	<b>0.848</b>	0.785	<b>0.785</b>	<b>0.793</b>	<b>0.792</b>	<b>0.792</b>	
MSR	0.811	0.758	<b>0.836</b>	<b>0.828</b>	<b>0.824</b>	0.773	0.770	<b>0.788</b>	<b>0.787</b>	<b>0.786</b>	
MSRCR	0.812	0.794	<b>0.815</b>	<b>0.832</b>	<b>0.815</b>	0.784	0.777	0.781	0.779	0.781	
RetinexNET	0.795	<b>0.802</b>	0.767	0.791	<b>0.797</b>	0.780	<b>0.781</b>	<b>0.783</b>	<b>0.785</b>	<b>0.787</b>	
GLADNET	0.781	0.767	0.771	0.776	<b>0.782</b>	0.792	<b>0.794</b>	<b>0.795</b>	<b>0.795</b>	<b>0.796</b>	
KIND	0.765	<b>0.794</b>	0.754	<b>0.770</b>	<b>0.773</b>	0.729	<b>0.741</b>	<b>0.731</b>	0.728	<b>0.729</b>	
KIND+	0.766	<b>0.779</b>	<b>0.772</b>	<b>0.785</b>	<b>0.787</b>	0.726	<b>0.730</b>	<b>0.727</b>	<b>0.726</b>	<b>0.729</b>	
Zero-DCE	0.833	<b>0.837</b>	0.826	<b>0.837</b>	<b>0.841</b>	0.797	<b>0.799</b>	0.793	0.794	0.794	
RUAS	0.774	<b>0.796</b>	<b>0.823</b>	<b>0.811</b>	<b>0.810</b>	0.806	<b>0.807</b>	0.791	0.803	0.789	
SCI	0.733	0.729	<b>0.757</b>	<b>0.756</b>	<b>0.760</b>	0.711	<b>0.713</b>	<b>0.728</b>	<b>0.728</b>	<b>0.728</b>	
LLFormer	0.763	<b>0.786</b>	<b>0.764</b>	<b>0.775</b>	<b>0.781</b>	0.685	0.683	<b>0.734</b>	<b>0.733</b>	<b>0.733</b>	
RetinexFormer	<b>0.766</b>	0.765	0.765	0.765	0.764	0.728	<b>0.730</b>	0.727	0.727	<b>0.729</b>	



**Fig. 4.** Visual comparison for different  $H$ . We can see how the linear cases do not approximate the colors of the original method as accurate as the affine versions. Also, the addition of root-polynomials (6-, 7- dimensional matrices) do not present any apparent improvement. The results in these images are computed with a DCT of order 7.

#### 4.7. Smoothness constraint: number of DCT basis

Let us now study how the different number of DCT basis considered for our method may affect the visual quality of the results. In Fig. 5 we show, for each of the two images, the input image and the result of the original method we aim to improve in the first row, while the other two rows present our results with different number of DCT basis. From this Figure we can clearly see that when we do not use the DCT constraint our method does improve over the original image—the contrast in the grass in the left image is more realistic—but it is still too able to adapt to the original image. This behavior is corrected when we introduce smoothness constraints via DCT. In this case, we output images that are clearly more realistic—see the last three images for each example.



**Fig. 5.** Visual comparison for different basis. We can see how when we are not using a DCT the result is still too affected by the original result. This is solved when the DCT is included, obtaining images that are more realistic. The results in these images are computed with a  $H$  matrix of dimension  $4 \times 3$ .

#### 4.8. User preference tests

We are interested to compare the outputs of algorithms with the same outputs that are post-processed according to our method. Of course, investigating user preferences given so many algorithms is too onerous a task (would involve hours of judgments per observer). So, in the experiments described below we focus only on the outputs generated by the deep-net algorithms (both because they produce leading results and are representative of the majority of contemporaneous research in this field).

In our experiment, we randomly selected 40 images of the SICE dataset. For each image, five original deep-learning low-light enhancement methods were computed (KindD+, GladNet, RUAS, Zero-DCE, and RetinexNet). Then, our post-processing algorithm was applied where, here, 28 DCT functions were chosen to represent the shading field of the image (relating input to output).

Because we are not interested in ranking the different deep-nets against one another we run 5 separate preference experiments, one per deep-learning method. The enhanced images—the

output from the deep-net method and our post-processing result— were viewed on either side of the original low-light image. Randomly, the deep-net output (and our post-processed results) were shown in the left or right positions. Subjects were then asked to select the image that they preferred out of the two enhanced images. An example of this setup is shown in Fig. 6. The total number of comparisons made by each observer was 200 —40 comparisons for each of the 5 original low-light enhancing methods. On average, the experiment took around 30 minutes per observer.



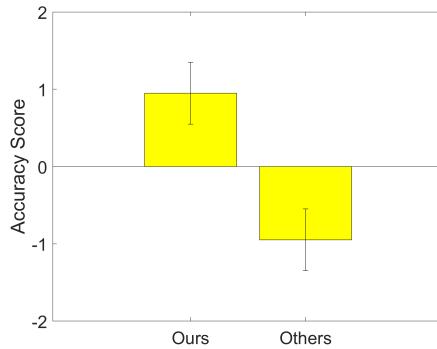
**Fig. 6.** Our experimental setup. An observer viewed the original low-light image in the center of the screen. The enhanced images—the output from the deep-net method and our post-processing result—were viewed on either side of the original low-light image. The positions of these two last images were randomized. The observer was asked to select which of two enhanced images he/she preferred.

The experiment was conducted on a DELL P2317H monitor with the following  $x, y$  primaries—red: 0.6513, 0.3383; green: 0.3246, 0.6182; blue: 0.1556, 0.0441; white: 0.3114, 0.3328—and with a peak white of 177.65 nits. The display was viewed at a distance of approximately 70 cm so that 40 pixels subtended 1 degree of visual angle. Because, we wished the observer to be able to see some detail in the low-light image (and be able to use this information in making their preference decision), the experiment was conducted in a dark room. The preference experiment was undertaken by 12 observers (none is an author of the paper). They were 9 male and 3 female, with ages in the range 21–32. All observers had normal color vision (tested using the Ishihara color blindness test).

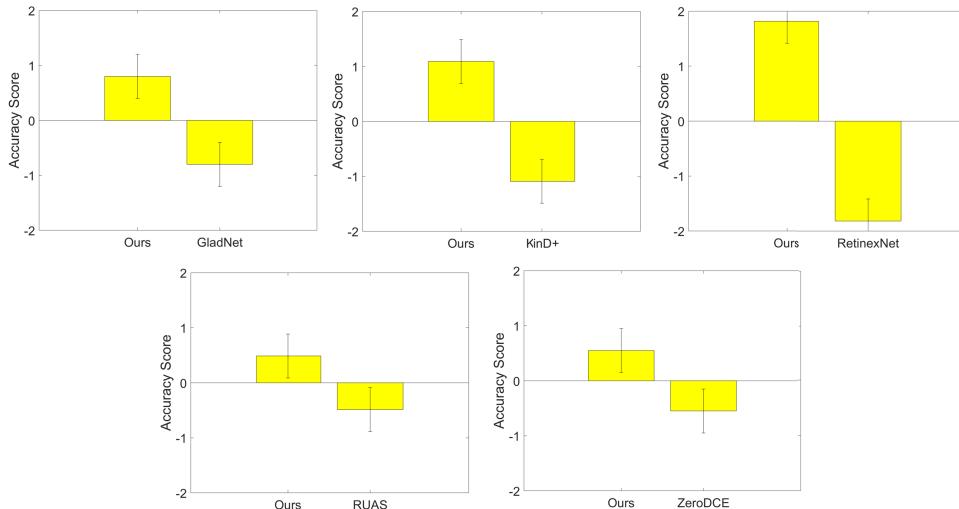
We have analyzed the result of our experiment in terms of the Thurstone Case V Law of Comparative Judgment [39]. Figure 7 presents the results for the whole set of 200 comparisons. To give a little intuition (for readers unfamiliar with Thurstonian analysis), a raw scoring matrix (that records the number of times the deep-net output is preferred/not preferred compared to our post-processing result) is recorded. Various assumptions are made that allow the raw scores to be translated into a standardized (z-score) unit together with confidence intervals. The higher the z-score the more a given algorithm is preferred. Let us notice here that other options for analyzing the data such as the Bradley-Terry method [40] also exist.

In Fig. 7 we combine all the results for all the observers and convert the raw score matrix to the standardized x-score representation. The average score is indicated by the top (or bottom) of the yellow bars shown. The vertical lines show the 95% confidence intervals. Clearly, Fig. 4 shows that our post-processing method delivers preferred outputs and—because the confidence intervals do not overlap—this result is significant (at the 95% level).

Now we look at the results of the comparisons for each specific low-light image enhancement method. The individual results for the 5 methods are presented in Fig. 8. Again we can see, our post-processing is preferred at the 95% statistical level.



**Fig. 7.** Results of the psychophysical experiment using the Thurstone Case V test for the pooled set of 200 preference choices made by each of 12 observers. Ours refer here to 28 DCT-basis functions case.



**Fig. 8.** Results of the psychophysical experiment using the Thurstone Case V test for each of the low-light enhancement deep-learning methods. Ours refer here to 28 DCT-basis functions case.

## 5. Conclusions

In this paper, we have introduced a post-processing approach aiming at improving the perception of current low-light image restoration algorithms. Our method states that the solution of post-processing must be in the form of a local shading plus a global color correction applied to the low-light image. In order to obtain such solution, we perform a minimization against the result of any well-known low-light enhancement method, using alternating least squares. Our solution is then the closest possible solution to the output of the method, but that, at the same time, satisfies our image formation model. We have shown using extensive results that our method improves the results of the original methods when looking at different perception-based metrics such as BRISQUE, NIQE, or UNIQUE. Importantly, preference experiments indicate (and indicate strongly) that the outputs generated by post-processing by 5 deep net algorithms (according to our method) are preferred compared to their raw outputs. Overall, our experiments confirm that humans are very sensitive to perceptual artifacts in enhanced low-light images resulting from overoptimizing distortion, in line with the perception-distortion tradeoff theory.

Importantly, preference experiments indicate (and indicate strongly) that the outputs generated by post-processing by 5 deep net algorithms (according to our method) are preferred compared to their raw outputs. Overall, our experiments confirm that humans are very sensitive to perceptual artifacts in enhanced low-light images resulting from overoptimizing distortion, in line with the perception-distortion tradeoff theory [4].

As further work, we would like to study the automatic selection of the number of DCT basis required for each specific image.

**Funding.** Centres de Recerca de Catalunya; Ministerio de Ciencia e Innovación, Agencia Estatal de Investigación (PID2021-128178OB-I00, and by ERDF "A way of making Europe", RYC2019-027020-I); Departament d'Universitats, Recerca i Societat de la Informació (2021SGR01499); Engineering and Physical Sciences Research Council (EP/S028730/1).

**Acknowledgment.** JVC and LH were supported by Grant PID2021-128178OB-I00 funded by MCIN/AEI and by ERDF "A way of making Europe" and by Departament de Recerca i Universitats from Generalitat de Catalunya with reference 2021SGR01499 (research group MACO). They also acknowledge the support of the Generalitat de Catalunya CERCA Program to CVC's general activities. LH was funded by the Spanish government under the Ramón y Cajal program. GF was funded by EPSRC.

**Disclosures.** GF is a part-time employer of Apple, Inc. He is also a Visiting Professor at NTNU, Norway.

**Data availability.** Datasets used in this paper are available in [33], [34].

## References

1. Y. Zhang, X. Guo, J. Ma, *et al.*, "Beyond brightening low-light images," *Int. J. Comput. Vis.* **129**(4), 1013–1037 (2021).
2. R. Liu, L. Ma, J. Zhang, *et al.*, "Retinex-inspired unrolling with cooperative prior architecture search for low-light image enhancement," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (2021), pp. 10561–10570.
3. C. Wei, W. Wang, W. Yang, *et al.*, "Deep retinex decomposition for low-light enhancement," in *British Machine Vision Conference (BMVC)*, (2018).
4. Y. Blau and T. Michaeli, "The perception-distortion tradeoff," in *International Journal of Computer Vision*, (2018), pp. 6228–6237.
5. J. Vazquez-Corral and G. D. Finlayson, "Coupled Retinex," in *Color and Imaging Conference*, vol. 2019 (2019), pp. 7–12.
6. E. H. Land, "The retinex," *Am. Sci.* **52**(2), 247–264 (1964).
7. Z.-u. Rahman, D. J. Jobson, and G. A. Woodell, "Multi-scale retinex for color image enhancement," in *IEEE International Conference on Image Processing (ICIP)*, vol. 3 (1996), pp. 1003–1006.
8. X. Fu, D. Zeng, Y. Huang, *et al.*, "A weighted variational model for simultaneous reflectance and illumination estimation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (2016), pp. 2782–2790.
9. G. Simone, M. Lecca, G. Gianini, *et al.*, "Survey of methods and evaluation of retinex-inspired image enhancers," *J. Electron. Imag.* **31**(06), 063055 (2022).
10. M. Lecca, C. M. Modena, and A. Rizzi, "Using pixel intensity as a self-regulating threshold for deterministic image sampling in milano retinex: the t-rex algorithm," *J. Electron. Imag.* **27**(01), 1 (2018).
11. A. Rizzi and C. Bonanomi, "Milano retinex family," *J. Electron. Imag.* **26**(3), 031207 (2017).
12. M. Lecca, A. Rizzi, and R. P. Serapioni, "Great: a gradient-based color-sampling scheme for retinex," *J. Opt. Soc. Am. A* **34**(4), 513–522 (2017).
13. M. Lecca, A. Rizzi, and R. P. Serapioni, "Grass: a gradient-based random sampling scheme for milano retinex," *IEEE Trans. on Image Process.* **26**(6), 2767–2780 (2017).
14. M. Lecca, A. Rizzi, and G. Gianini, "Energy-driven path search for termite retinex," *J. Opt. Soc. Am. A* **33**(1), 31–39 (2016).
15. M. Lecca and S. Messelodi, "Super: Milano retinex implementation exploiting a regular image grid," *J. Opt. Soc. Am. A* **36**(8), 1423–1432 (2019).
16. X. Guo, Y. Li, and H. Ling, "Lime: Low-light image enhancement via illumination map estimation," *IEEE Trans. on Image Process.* **26**(2), 982–993 (2017).
17. S. Wang, J. Zheng, H.-M. Hu, *et al.*, "Naturalness preserved enhancement algorithm for non-uniform illumination images," *IEEE Trans. on Image Process.* **22**(9), 3538–3548 (2013).
18. Z. Ying, G. Li, and W. Gao, "A bio-inspired multi-exposure fusion framework for low-light image enhancement," *arXiv*, arXiv:1711.00591 (2017).
19. X. Fu, D. Zeng, Y. Huang, *et al.*, "A fusion-based enhancing method for weakly illuminated images," *Signal Process.* **129**, 82–96 (2016).
20. W. Wang, C. Wei, W. Yang, *et al.*, "Gladnet: Low-light enhancement network with global awareness," in *IEEE International Conference on Automatic Face & Gesture Recognition (FG)*, (2018), pp. 751–755.
21. Y. Zhang, J. Zhang, and X. Guo, "Kindling the darkness: A practical low-light image enhancer," in *ACM International Conference on Multimedia*, (2019), pp. 1632–1640.

22. T. Wang, K. Zhang, T. Shen, *et al.*, “Ultra-high-definition low-light image enhancement: A benchmark and transformer-based method,” in *AAAI Conference on Artificial Intelligence (AAAI)*, vol. 37 (2023), pp. 2654–2662.
23. Y. Cai, H. Bian, J. Lin, *et al.*, “Retinexformer: One-stage retinex-based transformer for low-light image enhancement,” in *IEEE/CVF International Conference on Computer Vision (ICCV)*, (2023), pp. 12504–12513.
24. Y. Jiang, X. Gong, D. Liu, *et al.*, “Enlightengan: Deep light enhancement without paired supervision,” *IEEE Trans. on Image Process.* **30**, 2340–2349 (2021).
25. C. Guo, C. Li, J. Guo, *et al.*, “Zero-reference deep curve estimation for low-light image enhancement,” in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (2020), pp. 1780–1789.
26. L. Ma, T. Ma, R. Liu, *et al.*, “Toward fast, flexible, and robust low-light image enhancement,” in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (2022), pp. 5637–5646.
27. W. Yang, S. Wang, Y. Fang, *et al.*, “From fidelity to perceptual quality: A semi-supervised approach for low-light image enhancement,” in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (2020), pp. 3063–3072.
28. G. D. Finlayson, H. Gong, and R. B. Fisher, “Color homography: Theory and applications,” *IEEE Trans. Pattern Anal. Mach. Intell.* **41**(1), 20–33 (2019).
29. G. D. Finlayson, M. Mohammadzadeh Darrodi, and M. Mackiewicz, “The alternating least squares technique for nonuniform intensity color correction,” *Color. Res. & Appl.* **40**(3), 232–242 (2015).
30. H. Ochoa-Domínguez and K. Rao, *Discrete Cosine Transform* (CRC Press, 2019).
31. A. B. Petro, C. Sbert, and J.-M. Morel, “Multiscale retinex,” *Image Processing On Line* pp. 71–88 (2014).
32. G. D. Finlayson, M. Mackiewicz, and A. Hurlbert, “Color correction using root-polynomial regression,” *IEEE Trans. on Image Process. (TIP)* **24**(5), 1460–1470 (2015).
33. V. Vonikakis, “<https://sites.google.com/site/vonikakis/datasets>”.
34. J. Cai, S. Gu, and L. Zhang, “Learning a deep single image contrast enhancer from multi-exposure images,” *IEEE Trans. on Image Process.* **27**(4), 2049–2062 (2018).
35. V. Bychkovsky, S. Paris, E. Chan, *et al.*, “Learning photographic global tonal adjustment with a database of input / output image pairs,” in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (2011).
36. A. Mittal, R. Soundararajan, and A. C. Bovik, “Making a “completely blind” image quality analyzer,” *IEEE Signal Process. Lett.* **20**(3), 209–212 (2013).
37. A. Mittal, A. K. Moorthy, and A. C. Bovik, “No-reference image quality assessment in the spatial domain,” *IEEE Trans. on Image Process.* **21**(12), 4695–4708 (2012).
38. W. Zhang, K. Ma, G. Zhai, *et al.*, “Uncertainty-aware blind image quality assessment in the laboratory and wild,” *IEEE Trans. on Image Process.* **30**, 3474–3486 (2021).
39. L. L. Thurstone, “A law of comparative judgment,” in *Scaling*, (Routledge, 1927), pp. 81–92.
40. R. A. Bradley and M. E. Terry, “Rank analysis of incomplete block designs: I. the method of paired comparisons,” *Biometrika* **39**(3/4), 324–345 (1952).