Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Automatic image enhancement by learning adaptive patch selection*#

Na LI, Jian ZHANG‡

School of Science and Technology, Zhejiang International Studies University, Hangzhou 310012, China

E-mail: nli@zisu.edu.cn; jeyzhang@outlook.com

Received Feb. 21, 2017; Revision accepted June 4, 2017; Crosschecked Jan. 22, 2019

Abstract: Today, digital cameras are widely used in taking photos. However, some photos lack detail and need enhancement. Many existing image enhancement algorithms are patch based and the patch size is always fixed throughout the image. Users must tune the patch size to obtain the appropriate enhancement. In this study, we propose an automatic image enhancement method based on adaptive patch selection using both dark and bright channels. The double channels enhance images with various exposure problems. The patch size used for channel extraction is selected automatically by thresholding a contrast feature, which is learned systematically from a set of natural images crawled from the web. Our proposed method can automatically enhance foggy or under-exposed/backlit images without any user interaction. Experimental results demonstrate that our method can provide a significant improvement in existing patch-based image enhancement algorithms.

Key words: Image enhancement; Contrast enhancement; Dark channel; Bright channel; Adaptive patch based processing

https://doi.org/10.1631/FITEE.1700125 **CLC number:** TP391.4

1 Introduction

Digital cameras and smart phones are widely used in taking photos. However, due to photosensitive instrument limitations and environmental influences, digital cameras are always insensitive to variations of input lighting, which leads to low contrast in a dimly lit environment and loss of detail. Therefore, taking good photos remains a challenge for normal users. Many devices provide built-in applications to help end users improve photo quality, but these applications require end users to be experienced in image processing and are time-consuming.

To address this problem, some automatic enhancement algorithms have been proposed. Dark channel prior (He et al., 2011) was proposed for image haze removal. Bright channel prior (Wang et al., 2013) was proposed for correcting under-exposed images. Although these patch-based image processing methods improve image quality significantly, we cannot unconditionally execute batch processes on all images because (1) the algorithms are oriented to specific problems and (2) the patch size is fixed and tuned for each image; otherwise, the processed images may suffer from halo effects.

In this study, we propose a novel image enhancement framework based on both dark and bright channels with adaptive patch selection. The motivation is that existing patch-based methods are essentially local histogram stretching. In another word, when we enhance a pixel according to its neighborhood,

‡ Corresponding author

* Project supported by the Zhejiang Provincial Natural Science Foundation of China (Nos. LY17F020009 and LQ14F020003), the National Natural Science Foundation of China (No. 61303143), and the Professional Development Project for Domestic Visiting Scholars in Universities of Zhejiang Provincial Education Department (Research on Image Stylization Based on Samples)
# A preliminary version of this paper has been presented at the Pacific-Rim Conference on Multimedia, 2016

ORCID: Jian ZHANG, http://orcid.org/0000-0001-6478-9192

we actually stretch the histogram of the local patch and pick up the enhanced value for the target pixel. If the histogram is sharp, the stretching will fail. To solve this issue, we propose a learning-based algorithm which aims to select the adaptive patch size for automatic image enhancement. In addition, we combine dark and bright channels to deal with general problems in consumer images.

The main contributions of our proposed framework are as follows:

1. We combine bright and dark channels to handle local enhancement, which proves more adaptive for various problematic images than using only one channel. Image dehazing and exposure correction are solved in a unified framework.

2. We learn the threshold for adaptive patch size selection on a collection of natural images crawled from the web. The learned threshold can ensure that the pixel is enhanced with the most sufficient local information. Compared with traditional methods using a fixed patch size, our method can run automatically without any user interaction and overcome some drawbacks like halos as well.

3. Our method statistically learns the bounds of ultra violet (UV) channels in the YUV color model for natural images. The learned bounds help avoid over-saturation in image enhancement. Experimental results on various image enhancement tasks demonstrate that our strategy can produce more compelling results. An earlier version of this article (Li et al., 2016) has introduced the strategy of enhancement using double channels based on adaptive patches. In this version, we prove the strategy mathematically, infer the threshold for selecting adaptive patches by statistical learning, and propose a model for proper enhancement on UV channels. We present more results and discussion in this version.

2 Related work

Image enhancement techniques are considered thoroughly in the image processing literature and depend significantly on the underlying applications. We discuss related work dealing with consumer images rather than those from specific areas such as medical images and remote sensing images.

Most image enhancement techniques are based on global methods. The simplest global exposure correction employs histogram equalization (HE) (Gonzalez and Wintz, 1987), which is further improved as differential gray-level histogram equalization (DHE) (Nakai et al., 2013) by differentiating the gray-level histogram containing an image's edge information. Another method is exposure-based sub-image histogram equalization (ESIHE) (Singh and Kapoor, 2014), where an image is divided into sub-images of different intensity levels according to exposure thresholds and the individual histogram of sub-images is equalized independently. Assefa et al. (2014) proposed channel selection to reconstruct over-exposed images because the red-green-blue (RGB) components are unusually over-exposed at the same position. S-curve is also a common global-based technique. By estimating the best image-specific nonlinear tone curve for a given image (Yuan and Sun, 2012), the dynamic range of the image can be reasonably adjusted.

Model-based methods are another typical branch of global image enhancement, which is usually proposed for air-light correction. A physical model was proposed and optimized with multiple images (Narasimhan and Nayar, 2003). The physical model was refined via shading and transmission functions and worked on a single image (Fattal, 2008). Oakley and Bu (2007) assumed that the air-light was constant throughout the image and minimized a global cost function to estimate the air-light. Kopf et al. (2008) used a sufficiently accurate match between a photograph and a geometric model to enhance an image. Generally, global methods cannot concentrate on local regions and can hardly deal with backlit photos.

Some algorithms consider only local patch information and enhance an image pixel by pixel. An early method was adaptive histogram equalization (AHE) (Pizer et al., 1987), which transforms each pixel based on the histogram of a square surrounding the pixel. Discrete cosine transform (DCT) has been employed to enhance an image block on direct current (DC) and alternating current (AC) coefficients (Sugimura et al., 2015). Similarly, spatial entropy of pixels using spatial distribution has been employed to perform nonlinear data mapping (Celik, 2014). Using channel priors to enhance images requires a special kind of patch-based algorithms. He et al. (2011) proposed the dark channel prior to dehaze foggy images that do not include shadows, whereas Wang et al. (2013) proposed the bright channel prior

to enhance under-exposed images that do not include highlights. Algorithms based on channel priors are patch based because the prior of a pixel is calculated by the patch surrounding it as in AHE. The previously mentioned local-based enhancement algorithms consider only a fixed patch size, which results in artifacts such as halo effects. With the popularity of deep learning, convolution neural network (CNN) has been adopted in the research of image enhancement. DehazeNet (Cai et al., 2016) proves that the Maxout unit in the first layer F1 is similar to the a priori methods. Together with a novel activation function, BReLU, CNN can be trained for image haze removal.

Video enhancement is a special branch of image enhancement that involves both spatial and temporal features. Liu et al. (2007) proposed a learning-based color tone mapping algorithm to improve the enhanced face regions. The mapping function was updated temporally based on intensity change detection. Xie et al. (2011) improved the spatially based histogram equalization modification (HEM) method by introducing a temporal constraint. Chen et al. (2013) adopted a learning-based tone mapping curve for multiple region of interests (ROIs) to spatially enhance a video frame. The mapping curve was smoothed temporally for temporal consistency.

3 Adaptive patch based image enhancement using double channels

The algorithms using channel priors reviewed in Section 2 suggest that enhancing a pixel based on its local patch is efficient for either foggy images or under-exposed images. To this end, we propose a patch-based image enhancement algorithm using both channels, which is a general form of both the dark channel prior (DCP) (He et al., 2011) and bright channel prior (BCP) (Wang et al., 2013). Moreover, we propose an automatic adaptive patch selection algorithm for channel extraction to eliminate artifacts. In this section, we present the details of the proposed algorithm. The learning procedure for automatic adaptive patch selection is presented in Section 4.

3.1 Histogram stretching and double channels

1. Linear histogram stretching

A simple but effective image enhancement technique is linear histogram stretching (often known as normalization) (Jain, 1989). The technique is globally based and attempts to improve the contrast in a gray image by stretching the range of intensity values it contains to span a desired range of values.

Before stretching, it is necessary to specify the upper and lower pixel value limits, over which the gray image is to be normalized. Suppose that the desired upper and lower limits are b_{out} and d_{out} respectively; the input limits are b_{in} and d_{in} respectively. Then each pixel of the gray image $I_{\mathrm{in}}(\boldsymbol{x})$ is globally scaled using the following equation:

$$I_{\mathrm{out}}(\boldsymbol{x})=\frac{b_{\mathrm{out}}-d_{\mathrm{out}}}{b_{\mathrm{in}}-d_{\mathrm{in}}}(I_{\mathrm{in}}(\boldsymbol{x})-d_{\mathrm{in}})+d_{\mathrm{out}}. \quad (1)$$

Typically, the input gray image $I_{\mathrm{in}}(\boldsymbol{x})$ is to be stretched as much as possible, i.e., $b_{\mathrm{out}}=1$ and $d_{\mathrm{out}}=0$, assuming that the intensity level ranges from 0 to 1. Accordingly, Eq. (1) is reduced to

$$I_{\mathrm{out}}(\boldsymbol{x})=\frac{I_{\mathrm{in}}(\boldsymbol{x})-d_{\mathrm{in}}}{b_{\mathrm{in}}-d_{\mathrm{in}}}. \quad (2)$$

Furthermore, to automatically enhance a gray image, the input limits b_{in} and d_{in} can be obtained by finding the highest and lowest pixel values that currently present in the gray image: $b_{\mathrm{in}}=\max(I_{\mathrm{in}}(\boldsymbol{x}))$ and $d_{\mathrm{in}}=\min(I_{\mathrm{in}}(\boldsymbol{x}))$. In general, simply applying Eq. (2) to each RGB channel is a way of enhancing color images. The equation is written as

$$I^{\mathrm{c}}_{\mathrm{out}}(\boldsymbol{x})=\frac{I^{\mathrm{c}}_{\mathrm{in}}(\boldsymbol{x})-d^{\mathrm{c}}_{\mathrm{in}}}{b^{\mathrm{c}}_{\mathrm{in}}-d^{\mathrm{c}}_{\mathrm{in}}}, \quad (3)$$

where $c\in\{\mathrm{r},\mathrm{g},\mathrm{b}\}$.

2. Double channels

Stretching image histograms linearly and globally is computationally fast but effective for darkened images only. It cannot deal with foggy or backlit images, whose histograms almost span the full range of intensity levels. Motivated by the dark/bright channel priors, we propose a novel enhancement method by adopting the linear histogram stretching method locally. In particular, parameters $b^{\mathrm{c}}_{\mathrm{in}}$ and $d^{\mathrm{c}}_{\mathrm{in}}$ in Eq. (3) are evaluated pixel by pixel rather than sharing a single value over the whole image. As shown in Fig. 1, an image and its two patches are linearly stretched, respectively. The histogram of the whole image remains almost unchanged after histogram stretching, while the patches are widen. Given an image I_{rgb}, we define all the $b^{\mathrm{c}}_{\mathrm{in}}$ values as its bright channel and $d^{\mathrm{c}}_{\mathrm{in}}$ as its dark channel. The

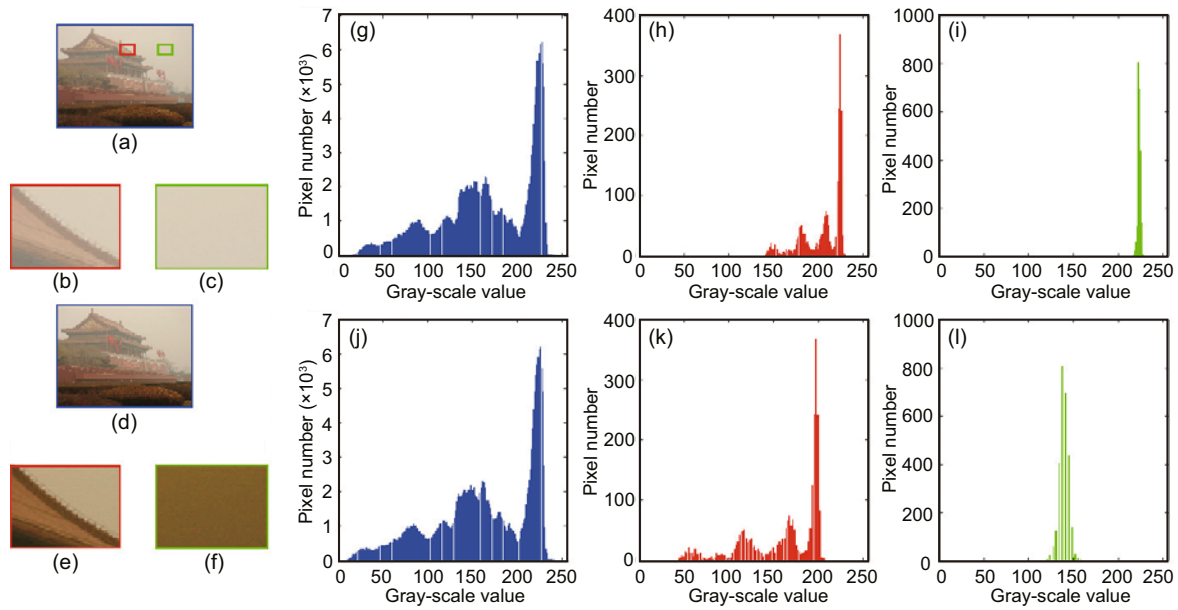

Fig. 1 Histograms of a foggy image and its two patches. The histogram is built on the red channel for clarity: (a) input image; (b) patch 1; (c) patch 2; (d) stretched image; (e) stretched patch 1; (f) stretched patch 2; (g) histogram of input image; (h) histogram of patch 1; (i) histogram of patch 2; (j) histogram of stretched image; (k) histogram of stretched patch 1; (l) histogram of stretched patch 2

dark channel is extracted as follows:

$$I_{\text{dark}}(\boldsymbol{x}) = \min_{\boldsymbol{z}\in\Omega(\boldsymbol{x},s),\ c\in\{\text{r,g,b}\}} I^{\text{c}}_{\text{rgb}}(\boldsymbol{z}), \tag{4}$$

where Ω is the local patch centered at pixel $\boldsymbol{x}$, and its patch size is s. The dark channel of a pixel is computed as the lowest value of all the RGB channels over its local patch. Note that the minimum value is no longer distinctive for different color channels for computational efficiency. Similarly, the bright channel is extracted as follows:

$$I_{\text{bright}}(\boldsymbol{x}) = \max_{\boldsymbol{z}\in\Omega(\boldsymbol{x},s),\ c\in\{\text{r,g,b}\}} I^{\text{c}}_{\text{rgb}}(\boldsymbol{z}). \tag{5}$$

To avoid block effect, both the dark and bright channels are filtered by a guided filter (He et al., 2013), where the guided image is the luminance of the input image. With both dark and bright channels, Eq. (3) is rewritten as

$$I^{\text{c}}_{\text{out}}(\boldsymbol{x}) = \frac{I^{\text{c}}_{\text{in}}(\boldsymbol{x}) - I_{\text{dark}}(\boldsymbol{x})}{I^{\text{c}}_{\text{bright}}(\boldsymbol{x}) - I_{\text{dark}}(\boldsymbol{x})}, \tag{6}$$

where $c \in \{\text{r, g, b}\}$.

3. Relationship between DCP and BCP

The proposed algorithm using double channels is a general form of DCP (He et al., 2011) and BCP (Wang et al., 2013). If only a dark channel is considered, the whole bright channel is set at the highest intensity level, i.e., $I_{\text{bright}}(\boldsymbol{x}) = 1$. Then Eq. (6) is reduced to $I^{\text{c}}_{\text{out}}(\boldsymbol{x}) = (I^{\text{c}}_{\text{in}}(\boldsymbol{x}) - I_{\text{dark}}(\boldsymbol{x}))/(1 - I_{\text{dark}}(\boldsymbol{x}))$, which is consistent with the formula in DCP where the air-light is set as pure white. In the same way, if only the bright channel is used, our algorithm is consistent with BCP. DCP is suitable for image dehazing, whereas BCP is suitable for under-exposure correction, because the shortage of shadow detail is represented by the dark channel and the shortage of highlight by the bright channel. In our proposed algorithm, both the dark and bright channels are adopted for general image enhancement.

Fig. 2 demonstrates the experimental results of Eq. (6) with a single channel and double channels. We set $I_{\text{bright}}(\boldsymbol{x}) = 1$ in Fig. 2c to simulate the single dark channel and set $I_{\text{dark}}(\boldsymbol{x}) = 0$ in Fig. 2e to simulate the single bright channel. The proposed algorithm gives similar results to DCP and BCP with either single channel (Figs. 2d and 2f), but can deal with both foggy and under-exposed/backlit images with double channels (Fig. 2b).

3.2 Adaptive patch selection

In Eq. (6), the patch size for channel extraction is fixed. However, to reach the best enhancement performance, we have to tune the patch size for different images. In Fig. 3, DCP runs on a set of

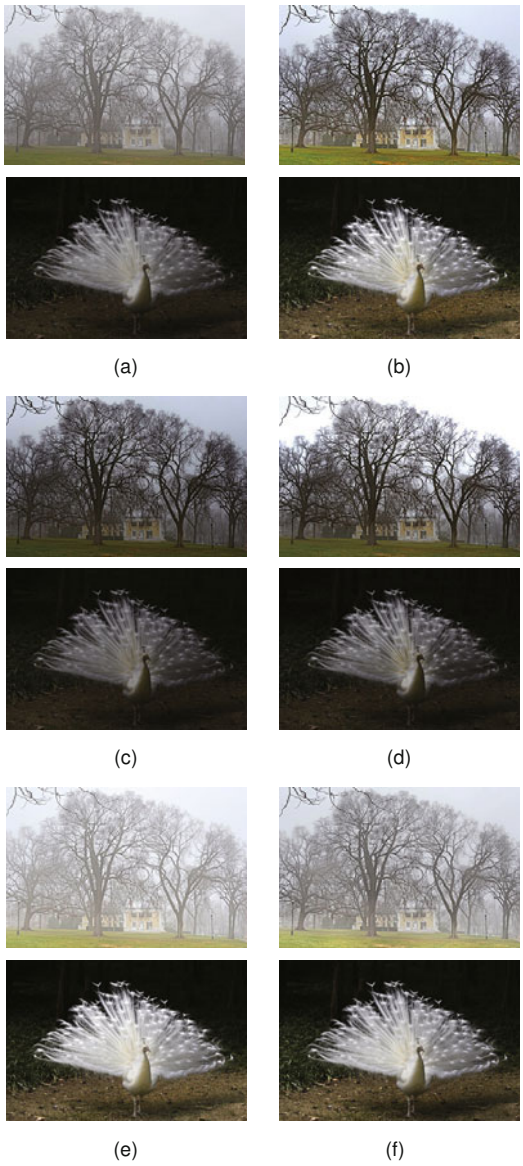

Fig. 2 Comparison between a single channel and double channels: (a) input image; (b) double channels; (c) dark channel only; (d) dark channel prior; (e) bright channel only; (f) bright channel prior

different patch sizes to either extract the dark channel or smooth it with the guided filter. Each best enhancement is highlighted by a red rectangle. Evidently, every image requires its own specific patch sizes. In this study, the aim of adaptive patch selection is to learn and automatically select best-fit local patches for channel extraction. To achieve this goal, we need to define what a best-fit patch is.

1. Best-fit local patch

As introduced in Section 3.1, our algorithm uses both dark and bright channels, which is essentially a local-histogram stretching process. One pixel's neighborhood determines its final state. For example, the sky pixels in Fig. 1 may have two different types of neighbors: one contains both the foreground and the sky, while the other contains only the sky. The latter patch could stretch its center pixel to some weird intensity levels. Because if there is only one narrow and steep peak in the histogram, the local patch is too flat to be stretched appropriately, which introduces improper enhancement (e.g., halo effects).

For patch-based image enhancement, the best-fit local patch for a pixel should contain some contrast at least; i.e., the patch cannot be totally flat. Thus, we define a feature $\phi(\boldsymbol{x}, s)$ for a local patch $\Omega(\boldsymbol{x}, s)$ centered at $\boldsymbol{x}$ with s as its size. The feature ϕ represents the contrast level of the patch. The larger the ϕ, the more contrast within the patch. If ϕ is too low, the patch will be too flat to be a proper local patch. Thus, given a threshold σ, the best-fit patch is defined as

$$\begin{aligned} s^* &= \arg \min_{s\in\{1,2,\ldots,S\}} A(\Omega(\boldsymbol{x}, s)) \\ \text{s.t.} \quad & \phi(\boldsymbol{x}, s) \geq \sigma, \end{aligned} \tag{7}$$

where $A(\Omega(\boldsymbol{x}, s))$ is the area of region $\Omega(\boldsymbol{x}, s)$ and S is the largest possible patch size within one image.

To determine a suitable feature ϕ, we observed a lot of image patches and found that the standard deviation of luminance over a patch is a good choice. A flat patch always has low standard deviation values. For a given pixel, its contrast feature over a local patch is defined as follows:

$$\phi(\boldsymbol{x}, s) = \sqrt{\frac{\sum_{\boldsymbol{z}\in\Omega(\boldsymbol{x},s)}(Y(\boldsymbol{z}) - \overline{Y(\boldsymbol{z})})^2}{s^2}}, \tag{8}$$

where $Y(\boldsymbol{z})$ is the luminance value of pixel $\boldsymbol{z}$ and $\overline{Y(\boldsymbol{z})}$ is the luminance average over the patch $\Omega(\boldsymbol{x}, s)$. In our algorithm, $Y(\boldsymbol{z})$ is simply the Y channel in YUV color space (Eq. (12)). The threshold σ is learned systematically from training data (detailed in Section 4) and fixed for enhancement of all natural images. Using standard deviation as the contrast feature representation, we can reasonably classify a local patch as flat or not and thus effectively select the best-fit patches.

If we assume that the best-fit patch sizes for all pixels in image I_{rgb} are denoted as patch size map I_{ps}, then the dark channel Eq. (4) is extracted as

$$I_{\text{dark}}(\boldsymbol{x}) = \min_{\substack{\boldsymbol{z}\in\Omega(\boldsymbol{x}, I_{\text{ps}}(\boldsymbol{x})),\\ c\in\{\text{r,g,b}\}}} I^{\text{c}}_{\text{rgb}}(\boldsymbol{z}), \tag{9}$$

Fig. 3 Comparison between fixed and adaptive patches where dark channel prior with different patch sizes for dark channel (DC) and guided filter (GF) in (b)–(d): (a) input image; (b) DC/GF:15; (c) DC:15, GF:71; (d) DC/GF:141; (e) our result. References to color refer to the online version of this figure

where Ω is the adaptive local patch centered at pixel $\boldsymbol{x}$, and its size is $I_{\mathrm{ps}}(\boldsymbol{x})$. Similarly, the bright channel Eq. (5) is extracted as

$$I_{\mathrm{bright}}(\boldsymbol{x}) = \max_{\substack{\boldsymbol{z}\in\Omega(\boldsymbol{x},I_{\mathrm{ps}}(\boldsymbol{x})),\\ c\in\{\mathrm{r,g,b}\}}} I^{c}_{\mathrm{rgb}}(\boldsymbol{z}). \tag{10}$$

Thanks to the adaptive patch size map, the dark and bright channels are free of halos. We compared the proposed adaptive patch size selection algorithm with the one using a fixed patch size (15×15 pixels for a 600×450 size image). The cause of the halo is explained in Fig. 1. The results shown in Fig. 3e demonstrate the effectiveness of our proposed algorithm. All the results were obtained without any parameter tuning.

2. Fast pyramid-based patch selection

Technically speaking, given an image, we could directly use the above contrast feature definition Eq. (8) and its corresponding threshold to obtain the best patch size for each pixel, whereas the brute-force search is definitely time-consuming. In our proposed algorithm, the adaptive patch size selection is implemented with a pyramid that includes two main steps: (1) building a contrast feature pyramid; (2) searching within the pyramid. The result may be not as precise as the brute-force search, but its accuracy is sufficient for our purpose, because we are searching for a proper local patch with some contrast, not with a specific contrast.

3. Building the contrast feature pyramid

We first build a bottom-up pyramid over the luminance channel of an input image as follows: the n^{th} level of the pyramid is used to store the ϕ of all $2^{n-1}\times2^{n-1}$ patches. For each level, the pyramid is built as

$$L_n(\boldsymbol{x}) = \phi(\boldsymbol{z}, 2^{n-1}), \tag{11}$$

where $\boldsymbol{x}=(i,j)$, ϕ is defined in Eq. (8) to extract contrast features and $\boldsymbol{z}=(i\times2^{n-1}+2^{n-2},\ j\times2^{n-1}+2^{n-2})$. If we suppose that N is the number of pyramid levels, then n ranges from 1 to N. The higher the level in the pyramid, the larger the patches involved in the contrast feature extraction.

4. Searching within the pyramid

After the pyramid is built, we compute a patch size map I_{ps} for each pixel of the input image. In the proposed method, we search the pyramid from the bottom; i.e., we start with the first level and work until the following criterion is reached: $L_t(\boldsymbol{x})>\sigma$, where σ is the feature threshold (introduced in Section 3.2). Once the search stops at level t for a given pixel $\boldsymbol{x}$, the patch size map is computed as $I_{\mathrm{ps}}(\boldsymbol{x})=2^t$. Note that 2^t is the patch size of one level higher than t. We choose a larger one to ensure that the local patch has sufficient contrast in any case; e.g., the pixel is on the edge of the corresponding square patch at level t. The patch size map stores the best-fit patch size for each pixel. An example of the patch size map is shown in Fig. 4. If the gray value of a pixel is dark, the best-fit patch size of that pixel is small.

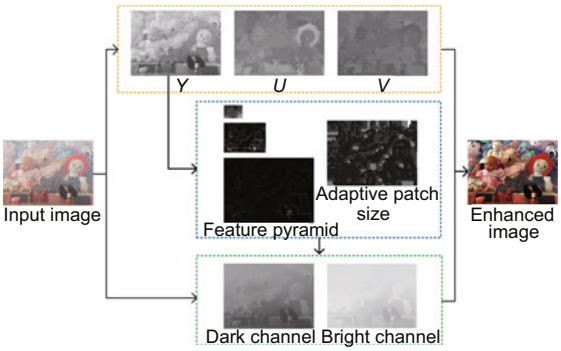

Fig. 4 Proposed automatic image enhancement pipeline

3.3 Image enhancement under the YUV color space

1. YUV color model

In our proposed algorithm, the input image is first converted to the YUV color space. As we know, the RGB color model represents images in electronic systems but nonintuitive for us humans. Comparatively, the YUV color model is a linear combination of RGB channels and represents a color with one luminance channel, Y, and two chrominance channels, U and V. Researchers proved that the YUV color model is similar to human vision (Podpora et al., 2014). We statistically learn the bounds of UV channels to avoid over-saturation. In this study, the YUV color model is derived as follows:

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.147 & -0.289 & 0.436 \\ 0.615 & -0.515 & -0.100 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}. \quad (12)$$

Accordingly, the enhancement under the RGB color space presented in Eq. (6) is transformed to the YUV color space. The Y channel is enhanced as

$$Y_{\mathrm{en}}(\boldsymbol{x}) = \frac{Y(\boldsymbol{x}) - I_{\mathrm{dark}}(\boldsymbol{x})}{Y(\boldsymbol{x}) - I_{\mathrm{bright}}(\boldsymbol{x})}, \quad (13)$$

and the UV channels are enhanced as

$$\begin{cases} U_{\mathrm{en}}(\boldsymbol{x}) = \dfrac{U(\boldsymbol{x})}{I_{\mathrm{bright}}(\boldsymbol{x}) - I_{\mathrm{dark}}(\boldsymbol{x})}, \\ V_{\mathrm{en}}(\boldsymbol{x}) = \dfrac{V(\boldsymbol{x})}{I_{\mathrm{bright}}(\boldsymbol{x}) - I_{\mathrm{dark}}(\boldsymbol{x})}. \end{cases} \quad (14)$$

2. Adaptive UV enhancement

Eq. (14) enhances UV channels linearly and may cause overflow or underflow of UV values. Simply clipping the UV values to the regular range $[-127, 128]$ will introduce over-saturation as in the RGB color model. In this study, the output of Eq. (14) is normalized to $[N_{\mathrm{U}}, M_{\mathrm{U}}]$ and $[N_{\mathrm{V}}, M_{\mathrm{V}}]$, respectively.

We trained the four parameters from the collected image data as introduced in Section 4.1. The UV channels are extracted from all the well-exposed images, and for each channel two cumulative histograms are built on two ranges: $[0, 128]$ and $[-127, 0]$. M_{U} is the upper bound of the U channel for all well-exposed images, i.e., $\Pr(0 < U(\boldsymbol{x}^k) \leq M_{\mathrm{U}} | k = \text{well-exposed}) = 1$. The other three bounds N_{U}, M_{V}, and N_{V} are inferred in the same way. Fig. 5 shows four cumulative histograms and the positions of corresponding bounds. The learned bounds are $M_{\mathrm{U}} = 105$, $N_{\mathrm{U}} = -124$, $M_{\mathrm{V}} = 125$, and $N_{\mathrm{V}} = -112$, which are used to normalize the enhanced UV values when overflow or underflow occurs. For an enhanced image, suppose that m_{U} and n_{U} are the maximum and minimum of U_{en} respectively, then the U channel is normalized as

$$U_{\mathrm{en}}^{n}(\boldsymbol{x}) = \begin{cases} U_{\mathrm{en}}(\boldsymbol{x}) : U_{\mathrm{en}}(\boldsymbol{x}) > 0 \text{ and } m_{\mathrm{U}} \leq M_{\mathrm{U}} \text{ or} \\ \qquad U_{\mathrm{en}}(\boldsymbol{x}) < 0 \text{ and } n_{\mathrm{U}} \geq N_{\mathrm{U}}, \\ \dfrac{M_{\mathrm{U}}}{m_{\mathrm{U}}} U_{\mathrm{en}}(\boldsymbol{x}) : U_{\mathrm{en}}(\boldsymbol{x}) > 0 \text{ and } m_{\mathrm{U}} > M_{\mathrm{U}}, \\ \dfrac{N_{\mathrm{U}}}{n_{\mathrm{U}}} U_{\mathrm{en}}(\boldsymbol{x}) : U_{\mathrm{en}}(\boldsymbol{x}) < 0 \text{ and } n_{\mathrm{U}} < N_{\mathrm{U}}, \end{cases} \quad (15)$$

and the V channel is normalized in the same way.

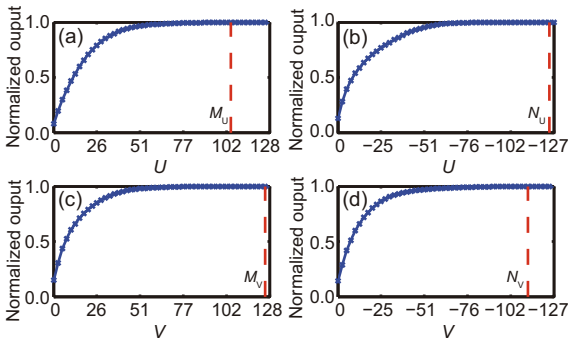

Fig. 5 Cumulative histograms of ultraviolet channels from the well-exposed image categories: (a) $U > 0$; (b) $U < 0$; (c) $V > 0$; (d) $V < 0$

We compared the adopted YUV color model with the original RGB color model. The results (Fig. 6) demonstrate that the color of the toy's hairs (the V channel overflows at 136) and the red roses (the V channel overflows at 159) are over-enhanced under the RGB color model in Fig. 6b, whereas the detail is well preserved under the proposed YUV color model in Fig. 6c.

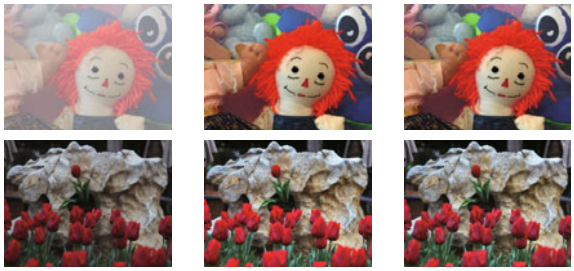

Fig. 6 Comparison between the RGB and YUV color models: (a) input image; (b) RGB model; (c) YUV model. References to color refer to the online version of this figure

3.4 Algorithm pipeline

Our automatic image enhancement is a local patch-based method and the pipeline is depicted in Fig. 4. There are four main steps: color space conversion, adaptive patch size selection, dark/bright channel extraction with adaptive patches, and image enhancement with double channels.

Algorithm 1 summarizes the pseudo code of the image enhancement algorithm. Following each step, the related equations are commented. Additionally, the method getPatchSizeMap(·) can be accelerated with a pyramid as introduced in Section 3.2.

Algorithm 1 Image enhancement algorithm

procedure imEnhance(I)
$[Y, U, V]$ = rgb2yuv(I); // Eq. (12)
$[I_{\text{ps}}]$ = getPatchSizeMap(Y); // Eqs. (7) and (8)
$[I_{\text{d}}, I_{\text{b}}]$ = extractDoubleChannels(I, I_{ps});
// Eqs. (9) and (10)
$[Y_{\text{en}}, U_{\text{en}}, V_{\text{en}}]$ = enhanceYuv(Y, U, V, I_{d}, I_{b});
// Eqs. (13), (14), and (15)
$[I_{\text{en}}]$ = yuv2rgb(Y_{en}, U_{en}, V_{en});
end procedure

4 Statistically learning to classify contrast feature

The proposed image enhancement framework can deal with various kinds of problematic images without any prior assumption. The procedure shown in Algorithm 1 is almost automatic, except that the method getPatchSizeMap(·) has one to-be-tuned parameter, σ, which is introduced from Eq. (7) and used for classifying a patch as flat or not. In this section, we present how σ is learned statistically, which guarantees that the proposed algorithm is free of parameter tuning.

Intuitively speaking, all the local patches of natural images are either flat or not. If these patches could be annotated, we would discriminatively learn a threshold effortlessly. However, the annotation is time-consuming. Instead of annotation, we rely on the statistical characteristics of the contrast feature ϕ: (1) The ϕ feature of flat patches should distribute normally, because the luminance standard deviation of a flat patch reflects noises rather than textures. On the contrary, the ϕ feature of non-flat patches distributes non-normally. (2) The ϕ feature of flat patches is always less than that of non-flat patches due to definition Eq. (8). (3) The content of an image determines the ϕ value of the patches. A non-flat patch in a well-exposed image may be almost flat perceptually if captured in the fog or under low light, but such a patch should be classified as non-flat. Similarly, a flat patch remains flat under any circumstance. Given a large number of images, the percentage of flat patches should be invariant on well-exposed or foggy/under-exposed images.

According to the summarized characteristics, the statistical learning for classifying flat/non-flat patches is designed with three steps: (1) collect images in three different categories: well-exposed, under-exposed, and foggy, and compute the contrast feature ϕ of each pixel with different patch sizes; (2) merge the datasets on patch sizes and image categories; (3) learn the threshold σ based on the invariant percentage of flat patches on different image datasets. Each step is detailed in the following subsections.

4.1 Data collection

1. Collecting images from three categories

We crawled natural images from the web with the following key words: well-exposed, under-exposed/backlit, and foggy/hazing. The collected images were subsequently grouped into three categories: $\Lambda \doteq \{\text{well-exposed, under-exposed, foggy}\}$ with 3000, 800, and 500 images, respectively. The images were down-sampled to 600 pixels on the long side for normalization. For all the images in the dataset, the contrast feature ϕ was computed at each pixel with four patch sizes: $\Gamma \doteq \{15, 31, 45, 61\}$. Then we divided the collected data into 12 groups according to different image categories and patch sizes, and computed the histogram of the contrast features of all the pixels in each group. The histogram is

represented as $p(\phi(\boldsymbol{x}^k, s)|k, s)$, where $k \in \Lambda$ and $s \in \Gamma$. The range of ϕ is $[1, 128]$, the bin width is set as 1, and thus the bin size goes to 128. Fig. 7 illustrates 12 frequency distributions for different image categories and patch sizes.

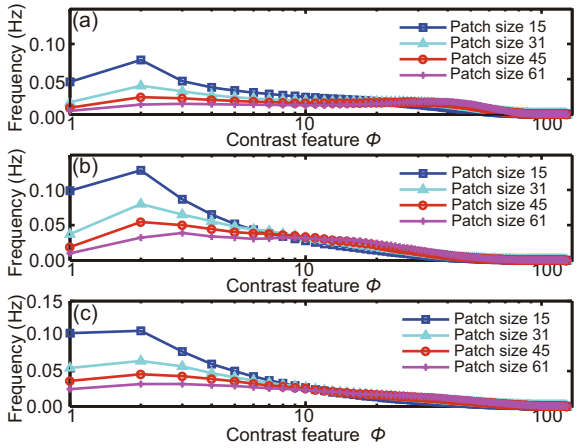

Fig. 7 Frequency distributions $p(\phi(\boldsymbol{x}^k, s)|k, s)$ of different image categories and patch sizes: (a) well-exposed; (b) foggy; (c) under-exposed

The frequency distributions $p(\phi(\boldsymbol{x}^k, s)|k, s)$ shown in Fig. 7 should be non-normal as we asserted. We performed the Jarque-Bera goodness-of-fit tests for verification. Table 1 illustrates the p-value of each Jarque-Bera test. All the p-values are below 0.05, which certified that the samples from $p(\phi(\boldsymbol{x}^k, s)|k, s)$ follow non-normality.

Table 1 p-values of Jarque-Bera tests on $p(\phi(\boldsymbol{x}^k, s)|k, s)$ of different image categories and patch sizes

Image category	p-value			
	15	31	45	61
Well-exposed	0.0010	0.0031	0.0068	0.0060
Foggy	0.0010	0.0010	0.0010	0.0010
Under-exposed	0.0010	0.0010	0.0010	0.0020

4.2 Dataset mergence

1. Merge datasets on different patch sizes

If distributions $p(\phi(\boldsymbol{x}^k, s)|k, s)$ with different size s are identical, a patch can be classified as flat/non-flat regardless of its size. To test the identity between different s, we performed two-sample Kolmogorov-Smirnov goodness-of-fit tests on a group of two $p(\phi(\boldsymbol{x}^k, s)|k, s)$. Every group depends on a pair of $s \in \Gamma$ within the same image category. The p-values of the Kolmogorov-Smirnov tests are shown in Table 2, in which all the p-values are above 0.05. This means that each image category $p(\phi(\boldsymbol{x}^k, s)|k, s)$ with different s is identically distributed. Therefore, we can reasonably merge the data sampled from different s. Fig. 8 shows the frequency distributions $p(\phi(\boldsymbol{x}^k, s)|k)$ after mergence.

Table 2 p-values of Kolmogorov-Smirnov tests on a group of two $p(\phi(\boldsymbol{x}^k, s)|k, s)$ within each image category with pairs of different patch size s

Image category	p-value				
	15/31	15/45	31/45	31/61	45/61
Well-exposed	0.713	0.194	0.108	0.324	0.713
Foggy	0.713	0.502	0.958	0.899	0.958
Under-exposed	0.502	0.108	0.989	0.989	0.502

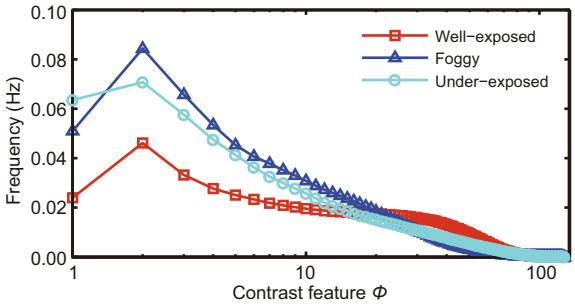

Fig. 8 Frequency distributions $p(\phi(\boldsymbol{x}^k, s)|k)$ of the original three image categories

2. Merging datasets on different exposure problems

Just like data mergence on different patch sizes, if the distributions for foggy and under-exposed are identical, the problematic images can be processed in a unified framework. Again, we performed two-sample Kolmogorov-Smirnov tests for each pair between the three image categories. Table 3 shows the p-values of the tests. This means that $p(\phi(\boldsymbol{x}^k, s)|k = \text{foggy})$ and $p(\phi(\boldsymbol{x}^k, s)|k = \text{under-exposed})$ are drawn from the same underlying continuous population, with high confidence; i.e., the p-value equals 1.0. On the other hand, $p(\phi(\boldsymbol{x}^k, s)|k = \text{well-exposed})$ is distributed differently from both foggy and under-exposed images with p-values less than 0.05. The tests verify that it is reasonable to unify the two distributions: $p(\phi(\boldsymbol{x}^k, s)|k = \text{foggy})$ and $p(\phi(\boldsymbol{x}^k, s)|k = \text{under-exposed})$. Consequently, the two categories merge into a single one called

"low-contrast" and the datasets are reorganized as $\Lambda^{\dagger} = \{\text{well-exposed}, \text{low-contrast}\}$. Fig. 9 shows the two distributions $p(\phi(\boldsymbol{x}^k, s)|k)$, $k \in \Lambda^{\dagger}$.

Table 3 p-values of Kolmogorov-Smirnov tests on $p(\phi(\boldsymbol{x}^k, s)|k)$ for pairs of every two categories

Category pair	p-value
Well-exposed / foggy	0.012
Well-exposed / under-exposed	0.012
Under-exposed / foggy	1.000

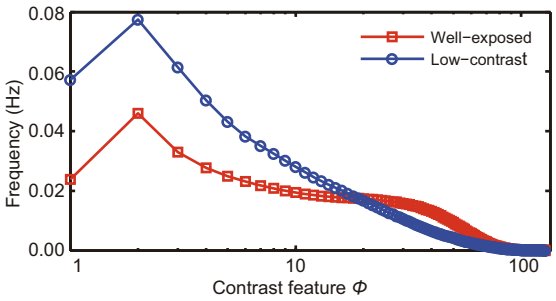

Fig. 9 Frequency distributions $p(\phi(\boldsymbol{x}^k, s)|k)$ of two image categories after merging

4.3 Threshold determination

1. Percentage of flat patches on well-exposed images

As analyzed at the beginning of this section, $p(\phi(\boldsymbol{x}^k, s)|k = \text{well-exposed})$ should be normal when only flat patches are sampled. Assume that the distributions shown in Fig. 9 are normal on a given range $[1, t]$ on which the ϕ of flat patches clusters. The distributions will become non-normal beyond that range. We iteratively performed Jarque-Bera goodness-of-fit tests of composite normality on $p(\phi(\boldsymbol{x}^k, s)|k = \text{well-exposed})$ with the range $[1, t]$ $(t \in [3, 128])$ increasing. Fig. 10 illustrates the p-values of the tests with regard to t. The p-value is 0.0573 at $t = 6$, 0.0343 at $t = 7$, and decreases as t increases. This means that when $\phi(\boldsymbol{x}, s) \leq \sigma_0$ $(\sigma_0 = 6, k = \text{well-exposed})$ the patches in a well-exposed image are most likely flat. The cumulative frequency $P(\phi(\boldsymbol{x}^k, s)|k = \text{well-exposed})$ is computed to determine the percentage of flat patches on well-exposed images. As illustrated in Fig. 11, the red curve shows that 18% of patches are flat.

2. Mapping the percentage of flat patches on low-contrast images

Taking advantage of the invariance of the

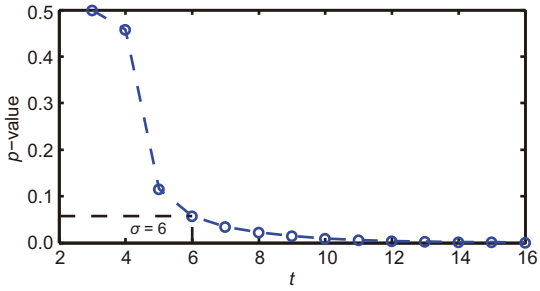

Fig. 10 Distribution of p-values of Jarque-Bera tests on $p(\phi(\boldsymbol{x}^k, s)|k = \text{well-exposed})$ on $[1, t]$

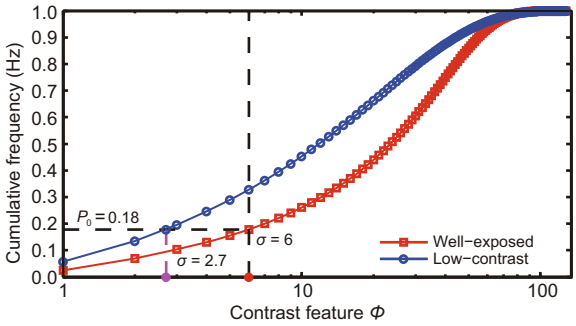

Fig. 11 Cumulative frequency distributions $p(\phi(\boldsymbol{x}^k, s)|k)$ of two image categories

percentage of flat patches among natural images, we employed the cumulative frequency $P(\phi(\boldsymbol{x}^k, s)|k)$ for comparing well-exposed and low-contrast images. Suppose that the threshold of ϕ for the low-contrast category is σ, then we have

$$\begin{aligned} &P(\phi(\boldsymbol{x}^k, s) \leq \sigma | k = \text{low-contrast}) \\ &= P(\phi(\boldsymbol{x}^k, s) \leq \sigma_0 | k = \text{well-exposed}). \end{aligned} \tag{16}$$

As illustrated in Fig. 11, 18% of the patches are flat in well-exposed images. Applying the same percentage to low-contrast images, the blue curve indicates that when $\phi(\boldsymbol{x}^k, s) \leq \sigma$ $(\sigma = 2.7,\ k = \text{low-contrast})$, the patches in a low-contrast image are most likely flat. In Eq. (7), the threshold $\sigma = 2.7$ is fixed for any image to be enhanced in our algorithm.

5 Experiments and discussion

In Section 3, Figs. 2, 3, and 6 show some results of the proposed method and demonstrate its advantages. As shown in Fig. 2, the double channel strategy guarantees appropriate enhancement on any problematic image, even if it lacks highlight or shadows. Fig. 3 demonstrates the advantage of adaptive patch selection. Fig. 6 shows the strength of the YUV color model.

In this section, we mainly compare our results with existing image enhancement algorithms qualitatively and quantitatively. Our model is abbreviated as DCAP. We performed an experiment on under-exposed images under controlled conditions to measure the capability of the proposed method.

5.1 Qualitative study

We qualitatively compare our approach with DCP (He et al., 2011), BCP (Wang et al., 2013), and linear histogram stretching globally (LHSG) (Jain, 1989), which is implemented as imadjust in MATLAB. For DCP and BCP, the patch size was set to 31×31 for a 600×400 size image. Because each algorithm is good at a specific type of problematic images, the comparison was run on three image sets: foggy images (Fig. 12), under-exposed images (Fig. 13), and backlit images (Fig. 14).

As demonstrated in the results, the DCP algorithm handled foggy images well except for some halo effects. The BCP algorithm worked well on both under-exposed and backlit images, but failed on any foggy image. Moreover, some results were a little bit over-processed. As far as LHSG is concerned, all the under-exposed images were well addressed but the foggy and backlit images remained problematic.

Comparatively, our proposed DCAP produced better results on all types of images with properly corrected exposure and vivid color. The double channel strategy guarantees that the proposed method can properly enhance a problematic image, even if it lacks highlight or shadows. The adaptive patch selection helps eliminate halo effects without any user interaction. In addition, color enhancement under the YUV color model was well controlled based on statistics.

5.2 Quantitative study

To evaluate our proposed algorithm, we built a test dataset including 46 foggy images, 46 under-exposed images, and 24 backlit images. The four methods compared in Section 5.1 were run on the test dataset. The methods were quantitatively evaluated on four objective metrics, subjective user rating, and computation time.

The four objective metrics are mean intensity (μV), mean saturation (μS), average information content (AIC) or entropy, and contrast improvement index (CII) (Chang and Chang, 2010; Chen et al., 2013). The user rating (UR) was determined by 10 users. The original test images and their enhanced results were shuffled and handed out randomly. The users rated the images with five grades, where "5" for excellent and "1" for unacceptable.

Table 4 demonstrates the comparison results. Overall, our proposed DCAP outperformed the other methods. After enhancement, both intensity and saturation increased, which is important for visual pleasure. The AIC and CII also improved significantly. Remarkably, users have rated our method as the best. From Table 4, we note that DCP (He et al., 2011) yields results with better saturation, BCP (Wang et al., 2013) always gives brighter images, and the simplest LHSG (Jain, 1989) can generally produce above-average results because its user rating is pretty good.

Table 4 Metrics on all test images

Method	μV	μS	AIC	CII	UR
Origin	93.63	0.2585	6.578	1.000	2.112
DCP	83.18	0.3610	6.411	2.150	2.155
BCP	**125.59**	0.2836	6.851	2.220	2.621
LHSG	112.31	0.2877	6.770	3.115	3.216
DCAP	116.27	**0.3780**	**7.263**	**3.560**	**3.448**

Bold numbers denote the best results

Individual tables like Table 4 are provided for each type of the exposure problem. Table 5 demonstrates the evaluation on foggy images. DCAP and DCP resulted in similar μS. DCP produced more contrast, whereas DCAP recovered more content information and its results were rated much higher. This implies that both DCAP and DCP are appropriate for enhancing foggy images, but DCAP produces more pleasant results. One interesting point is that the results of LHSG were rated as high as 3.5 by users, even better than DCP. The reason is that when the patch size is fixed, the results of DCP may contain some artifacts and thus are visually inferior.

Table 6 demonstrates the evaluation on under-exposed images. LHSG and DCAP performed similarly. From the point of view of visual experience, LHSG worked a little bit better. The global stretching method is quite suitable for under-exposed images with a narrow histogram. BCP resulted in average performance, whereas DCP was ineffective.

Table 7 demonstrates the evaluation on backlit images. For this type of image problem, DCAP

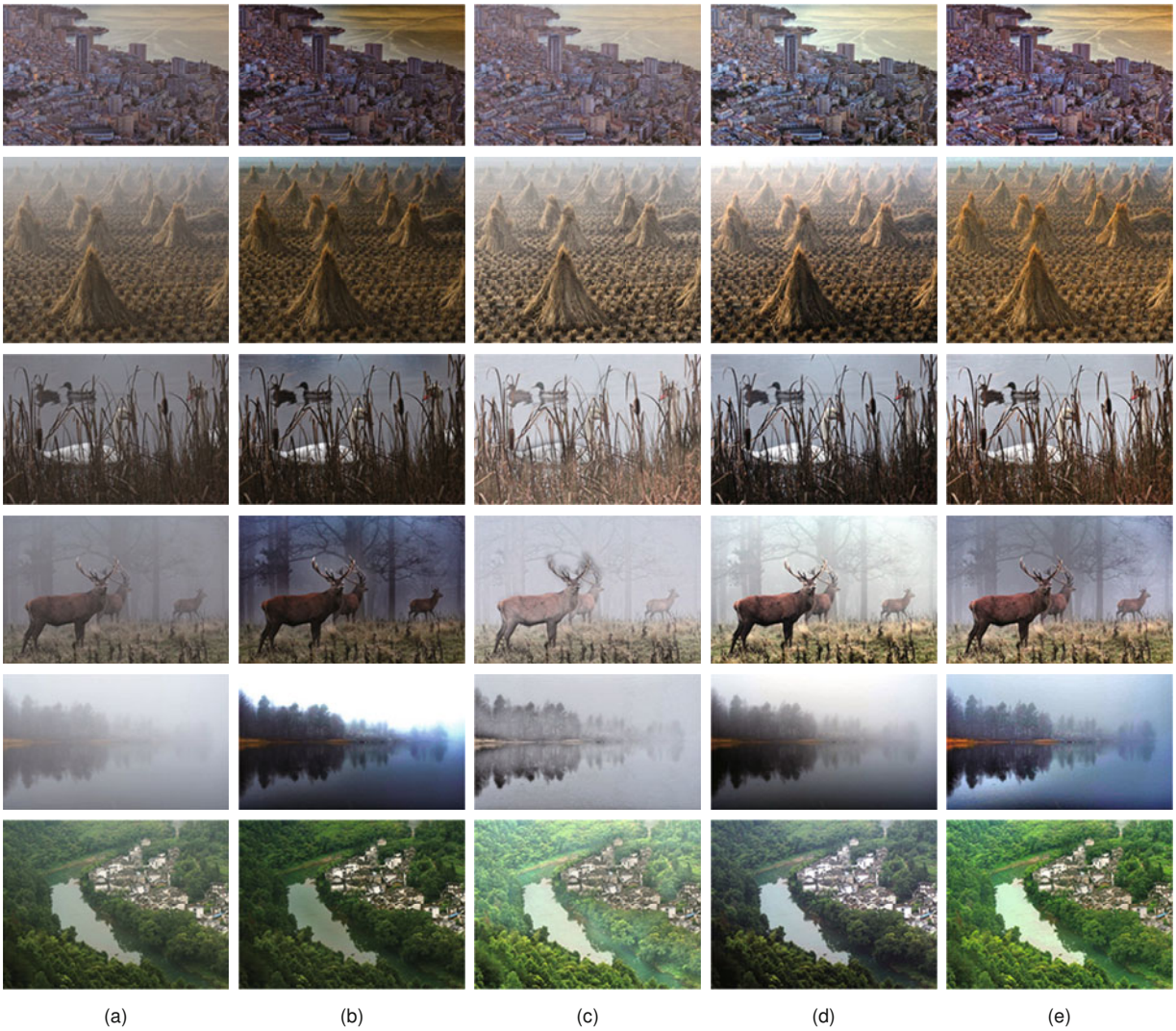

Fig. 12 Comparison on foggy images: (a) input image; (b) dark channel prior; (c) bright channel prior; (d) linear histogram stretching globally; (e) double channels with adaptive patches

Table 5 Metrics on foggy test images

Method	μV	μS	AIC	CII	UR
Origin	138.26	0.1660	7.053	1.000	2.196
DCP	113.67	**0.3280**	6.909	**3.182**	2.804
BCP	**169.89**	0.1684	7.092	1.402	2.283
LHSG	142.15	0.1882	7.304	2.069	3.500
DCAP	145.54	0.3273	**7.486**	2.603	**3.696**

Bold numbers denote the best results

Table 6 Metrics on under-exposed test images

Method	μV	μS	AIC	CII	UR
Origin	46.30	0.3623	5.977	1.000	1.957
DCP	45.78	0.4097	5.927	1.291	1.739
BCP	79.63	0.4139	6.603	3.072	2.783
LHSG	84.80	0.4177	6.215	**5.002**	**3.391**
DCAP	**85.96**	**0.4494**	**7.030**	4.811	3.304

Bold numbers denote the best results

outperforms every other method, producing both objectively and subjectively better results. Wang et al. (2013) claimed that BCP corrected backlit images to some extent. Both DCP and LHSG fail for this problem because they do not agree with their assumptions.

In summary, DCP, BCP, and LHSG do work as long as the image problem is appropriate. However, our proposed DCAP handles all the problems simultaneously and automatically. Evaluations show that it reaches or approximates the best performance on each kind of image. However, DCAP is not perfect. Its main drawback is that it is time-consuming. We tested the computation time for each method. All of them were run in MATLAB without any code optimization. The machine was configured as a Core i5

Fig. 13 Comparison on under-exposed images: (a) input image; (b) dark channel prior; (c) bright channel prior; (d) linear histogram stretching globally; (e) double channels with adaptive patches

Table 7 Metrics on backlit test images

Method	μV	μS	AIC	CII	UR
Origin	98.79	0.2369	6.822	1.000	2.250
DCP	96.42	0.3307	6.384	1.820	1.708
BCP	**128.75**	0.2547	6.864	2.153	2.958
LHSG	107.83	0.2293	6.808	1.504	2.333
DCAP	118.25	**0.3383**	**7.283**	**2.996**	**3.250**

Bold numbers denote the best results

Table 8 Comparison on time consumption

Method	Time (s)			Percentage of channel extraction
	Channel extraction	Others	Total	
DCP	1.03	0.37	1.40	74%
BCP	1.20	1.94	3.14	38%
LHSG	0.00	0.04	0.04	0%
DCAP	13.72	1.09	14.81	93%

processor and 6 GB main memory. Table 8 shows the computation time. Obviously, DCAP is much slower than the other methods, especially LHSG. A total of 93% of the DCAP computation time is consumed by channel extraction.

In Algorithm 1, channel extraction consists of getPatchSizeMap(·) and extractDoubleChannels(·). Table 9 demonstrates that the computation time of getPatchSizeMap(·) is stable over all images, but under-exposed images consume much more time on extractDoubleChannels(·). The reason is that every pixel in an under-exposed image corresponds to a large patch and needs more time to extract its dark/bright channel.

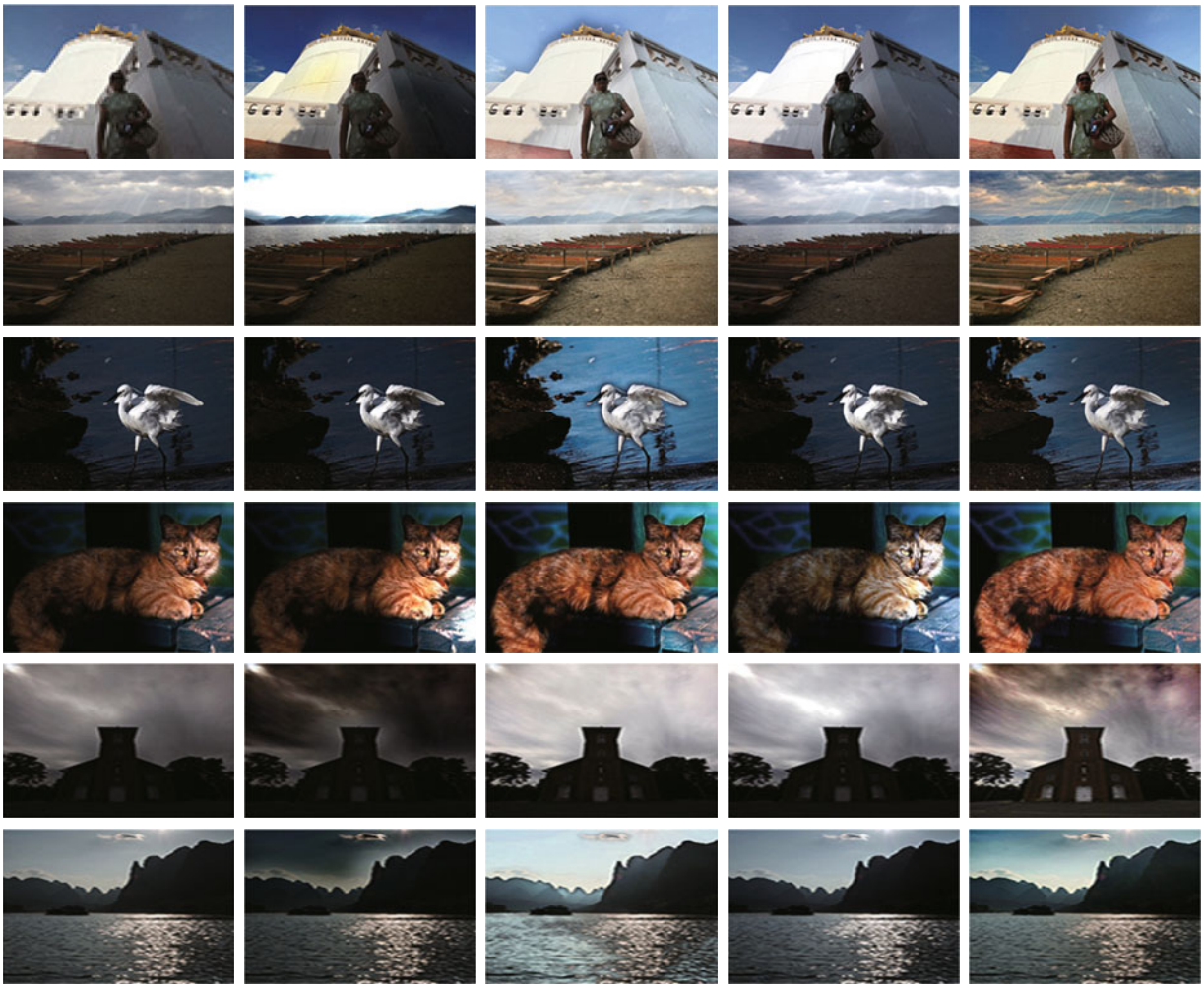

Fig. 14 Comparison on backlit images: (a) input image; (b) dark channel prior; (c) bright channel prior; (d) linear histogram stretching globally; (e) double channels with adaptive patches

Table 9 Computation time details for double channels with adaptive patches

Category	Computation time (s)				Percentage of channel extraction
	getPatchSizeMap(·)	extractDoubleChannels(·)	Channel extraction	Total	
All	2.71	11.01	13.72	14.82	93%
Foggy	2.68	8.83	11.51	12.68	91%
Under-exposed	2.79	14.14	16.93	18.02	94%
Backlit	2.60	9.01	11.61	12.61	92%

5.3 Measurement on under-exposure correction

An experiment on images with controlled exposure was carried out to measure the capability of the proposed method. The input images shown in Fig. 15 were captured by Canon EOS 20D. The exposure was set as follows: the aperture was F8, the shutter speed was 1/100 s, and ISO was 100. We locked the aperture and IOS, and adjusted the exposure value (EV) by increasing or decreasing the shutter speed. Fig. 15 shows the experiment results. Visually, the results on +2 EV, −6 EV, and −7 EV recover little color details, whereas other results are comparatively acceptable.

To quantitatively evaluate the experiment results, the four objective metrics introduced in Section 5.2 were measured on the original and enhanced images in Fig. 15. Table 10 shows the quantitative

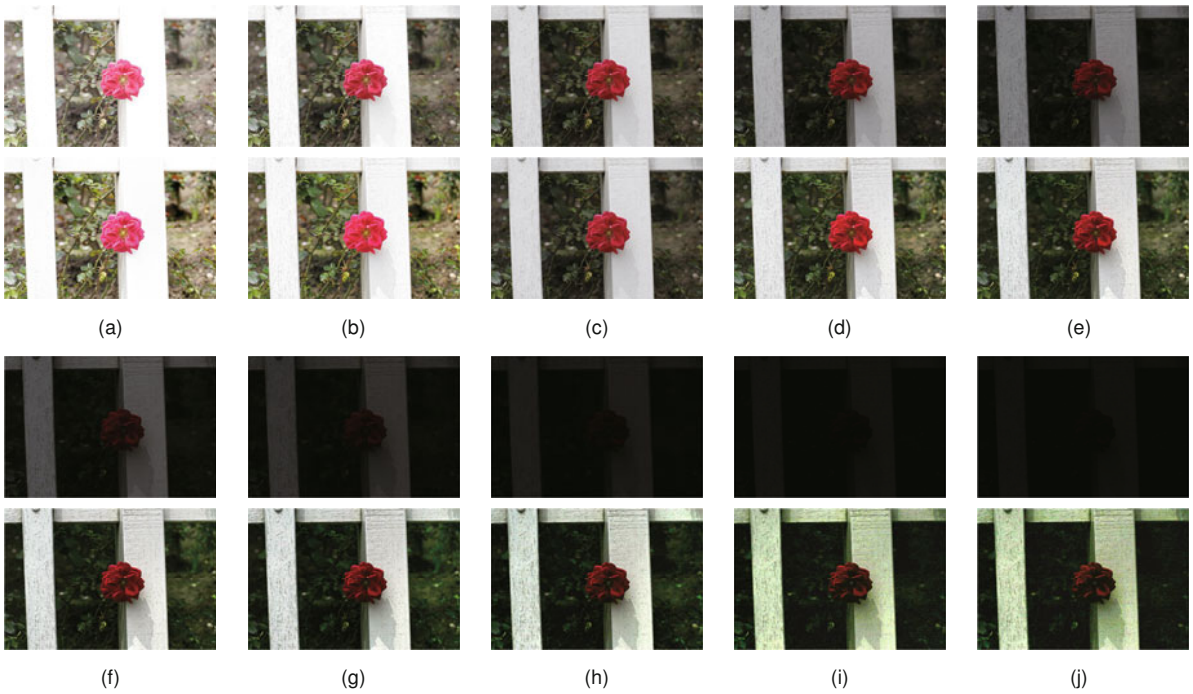

Fig. 15 Images with controlled exposure. Input image (top row) and enhanced image (bottom row): (a) +2 EV; (b) +1 EV; (c) +0 EV; (d) −1 EV; (e) −2 EV; (f) −3 EV; (g) −4 EV; (h)−5 EV; (i) −6 EV; (j) −7 EV

Table 10 Quantitative experiments for enhancement on images with controlled exposure

Enhanced level (EV)	μV		μS		AIC		CII
	Original	Enhanced	Original	Enhanced	Original	Enhanced	
+2	196	191	0.135	0.229	5.765	6.594	1.309
+1	163	174	0.177	0.266	7.219	7.342	1.201
+0	125	–	0.193	–	7.288	–	–
−1	89	138	0.164	0.269	6.980	7.590	1.163
−2	62	124	0.117	0.281	6.497	7.565	1.931
−3	38	112	0.074	0.306	5.866	7.532	8.198
−4	23	105	0.043	0.327	5.215	7.509	14.517
−5	13	99	0.065	0.320	4.314	7.232	16.266
−6	7	93	NaN	0.350	3.426	6.784	12.572
−7	6	90	NaN	0.342	3.187	6.274	11.833

μV: mean intensity; μS: mean saturation; AIC: average information content; CII: contrast improvement index; EV: exposure value; NaN: not a number

evaluation. The metric AIC (or entropy) of the image with proper exposure (0 EV) is 7.288, which means that if the AIC of enhanced images reaches this value, the enhancement is sufficient from the view of information recovery. Table 10 demonstrates that $[-5, +1]$ EV is the right range for appropriate enhancement on under-exposed images. The objective evaluation is consistent with the previous subjective evaluation.

6 Conclusions and future work

In this study, we have proposed an automatic image enhancement method using double channels and adaptive patch size. The patch size of a given pixel was selected by a contrast feature with thresholding. The threshold was learned systematically from collected images. Then dark and bright channels were both extracted based on the adaptive local patch. The final enhancement, using the YUV color model with normalization on UV channels, is simple and effective. Because all the parameters in our method were trained from a dataset, the system is totally automatic. The drawback of our proposed algorithm is that it is comparatively time-consuming, which can be optimized by engineering methods to some extent.

Some inhomogeneous regions will be occasionally produced by only employing local patches in the algorithm. With some global information being introduced, the enhancement result could be more natural on the whole. In the future work, we will explore the relationship between global and local enhancement and try to unify the two approaches into a single framework.

References

Assefa M, Poulie T, Kervec J, et al., 2014. Correction of overexposure using color channel correlations. IEEE Global Conf on Signal and Information Processing, p.1078-1082. https://doi.org/10.1109/GlobalSIP.2014.7032287

Cai B, Xu X, Jia K, et al., 2016. DehazeNet: an end-to-end system for single image haze removal. *IEEE Trans Image Process*, 25(11):5187-5198. https://doi.org/10.1109/TIP.2016.2598681

Celik T, 2014. Spatial entropy-based global and local image contrast enhancement. *IEEE Trans Image Process*, 23(12):5298-5308. https://doi.org/10.1109/TIP.2014.2364537

Chang YC, Chang CM, 2010. A simple histogram modification scheme for contrast enhancement. *IEEE Trans Consum Electron*, 56(2):737-742. https://doi.org/10.1109/TCE.2010.5505995

Chen Y, Lin W, Zhang C, et al., 2013. Intra-and-inter-constraint-based video enhancement based on piecewise tone mapping. *IEEE Trans Circ Syst Video Technol*, 23(1):74-82. https://doi.org/10.1109/TCSVT.2012.2203198

Fattal R, 2008. Single image dehazing. *ACM Trans Graph*, 27(3):1-9. https://doi.org/10.1145/1360612.1360671

Gonzalez RC, Wintz P, 1987. Digital Image Processing (2nd Ed.). Addison-Wesley, Boston, USA, p.484-486.

He K, Sun J, Tang X, 2011. Single image haze removal using dark channel prior. *IEEE Trans Patt Anal Mach Intell*, 33(12):2341-2353. https://doi.org/10.1109/TPAMI.2010.168

He K, Sun J, Tang X, 2013. Guided image filtering. *IEEE Trans Patt Anal Mach Intell*, 35(6):1397-1409. https://doi.org/10.1109/TPAMI.2012.213

Jain AK, 1989. Fundamentals of Digital Image Processing. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

Kopf J, Neubert B, Chen B, et al., 2008. Deep photo: model-based photograph enhancement and viewing. *ACM Trans Graph*, 27(5):1-10. https://doi.org/10.1145/1409060.1409069

Li N, Liu Z, Lei J, et al., 2016. Automatic color image enhancement using double channels. Pacific-Rim Conf on Multimedia, p.74-83. https://doi.org/10.1007/978-3-319-48896-7_8

Liu Z, Zhang C, Zhang Z, 2007. Learning-based perceptual image quality improvement for video conferencing. IEEE Int Conf on Multimedia and Expo, p.1035-1038. https://doi.org/10.1109/ICME.2007.4284830

Nakai K, Hoshi Y, Taguchi A, 2013. Color image contrast enhancement method based on differential intensity/saturation gray-levels histograms. Int Symp on Intelligent Signal Processing and Communication Systems, p.445-449. https://doi.org/10.1109/ISPACS.2013.6704591

Narasimhan SG, Nayar SK, 2003. Contrast restoration of weather degraded images. *IEEE Trans Patt Anal Mach Intell*, 25(6):713-724. https://doi.org/10.1109/TPAMI.2003.1201821

Oakley JP, Bu H, 2007. Correction of simple contrast loss in color images. *IEEE Trans Image Process*, 16(2):511-522. https://doi.org/10.1109/TIP.2006.887736

Pizer SM, Amburn EP, Austin JD, et al., 1987. Adaptive histogram equalization and its variations. *Comput Vision Graph Image Process*, 39(3):355-368. https://doi.org/10.1016/S0734-189X(87)80186-X

Podpora M, Korbaś GP, Kawala-Janik A, 2014. YUV vs RGB—choosing a color space for human-machine interaction. Federated Conf on Computer Science and Information Systems, p.29-34. https://doi.org/10.15439/2014F206

Singh K, Kapoor R, 2014. Image enhancement using exposure based sub image histogram equalization. *Patt Recogn Lett*, 36(1):10-14. https://doi.org/10.1016/j.patrec.2013.08.024

Sugimura D, Mikami T, Yamashita H, et al., 2015. Enhancing color images of extremely low light scenes based on RGB/NIR images acquisition with different exposure times. *IEEE Trans Image Process*, 24(11):3586-3597. https://doi.org/10.1109/TIP.2015.2448356

Wang Y, Zhuo S, Tao D, et al., 2013. Automatic local exposure correction using bright channel prior for under-exposed images. *Signal Process*, 93(11):3227-3238. https://doi.org/10.1016/j.sigpro.2013.04.025

Xie J, Lin W, Li H, et al., 2011. A new temporal-constraint-based algorithm by handling temporal qualities for video enhancement. IEEE Int Symp of Circuits and Systems, p.2789-2792. https://doi.org/10.1109/ISCAS.2011.5938184

Yuan L, Sun J, 2012. Automatic exposure correction of consumer photographs. European Conf on Computer Vision, p.771-785. https://doi.org/10.1007/978-3-642-33765-9_55