Contrast adjustment via Bayesian sequential partitioning

Zhiyu Wang, Shuo Xie, Bai Jiang

Abstract

Photographs taken in dim light have low color contrast. However, traditional methods for adjusting contrast suffer from boundary artifacts created as a side effect of Gaussian smoothing. This paper presents a method for adjusting contrast without creating boundary artifacts, by relying on Bayesian sequential partitioning to group similar colors, and then adjusting grouped colors together.

1 Introduction

The perceptual phenomenon of "local contrast" is fundamental for image enhancement. We percieve the color of a feature in an image relative to its the colors in its immediate neighborhood. Thus, a color will appear darker if it surrounded by bright colors, but the same color will appear lighter if it surrounded by dark colors. A number of well-known optical illusions demonstrate this phenomenon.

The following Figure (1) provides an example. The two small boxes inside the larger

Figure 1: Example of local contrast

boxes appear to have different shades of gray; in fact, they are the same color. We percieve the small box on the right to have a lighter color than the box on the left because of local contrast. Therefore, when enhancing images, we aim to distinguish features relative to their neighborhoods.

2 Computation of local contrast

2.1 Smoothed image

Local contrast is usually measured by taking a difference between the original image and a smoothed version of the image. A smoothed image is obtained by using convolution of the input signal with a certain kernel, e.g. Gaussian kernel. The detail of an image is defined by

$$\text{Detail} = \text{Original Input Signal - Smoothed Signal} \tag{1}$$

Then we get the enhanced image by adding back the detail into the adjusted smoothed image. We also take the image (1) as an example, the results are in Figure (2) and (3).

Figure 2: Smoothed image

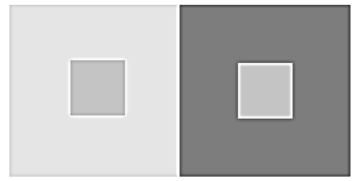


Figure 3: Traditional enhanced image

2.2 Creation of Boundary Artifacts

The enhancement procedure produces a boundary artifact, Figure (3): the color of the boundary of the box is over-adjusted, while the color on the inside of the box has not been adjusted as drastically ([3]). This leads to an artifact since the boundary now has a much different color than the inside of the box. We say that the enhanced signal is not propagated into the whole graph properly. Ideally, the boundary of the enhanced picture should have the same color as the inside of the box, as it was in the original image.

Thus, traditional contrast adjustment methods suffer from the production of boundary artifacts. But in order to prevent such boundary artifacts from occuring, one must ensure that the boundaries of objects are not treated too differently from the interiors of the same objects. Our approach is to cluster the pixels with similar features into several groups, and change the intensity of pixels identically in each group. We explain this method in the next section.

3 Our method

The key idea in our method is that pixels which are close in the RGB scale in in the original image should remain close in the output image. The image enhancement should change a group of pixels with similar colors in the same way. To do this, we turn to unsupervised learning approaches to cluster pixels based on color similarity.

3.1 Bayesian sequential partitioning

We identify groups of similarly colored pixels by using Bayesian sequential partitioning (BSP). BSP was developed by Wing Wong et al. for density estimation see ([1]). The BSP will not only cluster points into groups (regions), as the usual K-means would do, but more importantly, also learn the density of each group. More concretely, suppose we have n training data $x^{(1)}, x^{(2)}, \ldots, x^{(n)} \in \mathbb{R}^d$, assumed i.i.d. We want to estimate the probability $\mathbb{P}(x \in A)$, given a region A. The BSP method partitions the region into small rectangles A_i and estimates the density within each rectangle α_i, see Figure (4) for a 2-dimension example.

The probability $\mathbb{P}(x \in A)$ and the likelihood are given by

$$\mathbb{P}(x|\alpha_1, \ldots, \alpha_t, A_1, \ldots, A_t) = \sum_{k=1}^{t} \frac{\alpha_k}{|A_k|} \mathbb{1}_{\{x \in A_k\}}$$

$$\mathbb{P}(x^{(1)}, \ldots, x^{(n)}|\alpha_1, \ldots, \alpha_t, A_1, \ldots, A_t) = \prod_{i=1}^{n} \mathbb{P}(x^{(i)}|\alpha_1, \ldots, \alpha_t, A_1, \ldots, A_t) = \prod_{k=1}^{t} \frac{\alpha_k^{n_k}}{|A_k|^{n_k}}$$

(a) Partition the region into small regions

(b) density α for each small regions

Figure 4: BSP method illustration

respectively, where $|A_k|$ denotes the volume of A_k and n_k denotes the number of training data that fall into region A_k. All the parameters in the BSP package are set by the methods in ([1]).

3.2 Graph representation

Each pixel is considered a three dimensional vector (r, g, b). In what follows we first take a log of the (r, g, b), based on Retinex theorem ([4]). After we apply the BSP, let N denote the number of groups we get. Now we construct a graph of N nodes, representing each group as a node. If we assign a three dimensional vector (a_i^r, a_i^g, a_i^b) for each node i, and define pixels that are clustered into group i to be (a_i^r, a_i^g, a_i^b), then we obtain a new image. Furthermore, define two nodes to be adjacent if the corresponding groups are adjacent in the partitions obtained by BSP. For example, in Figure (4), A_1 and A_2 are adjacent, while A_1 and A_4 are not.

Denote $\alpha_i, (1 \leq i \leq N)$ to be the density of group i learned by BSP method. Define the weight of adjacent nodes to be

$$w_{ij} = \exp\Big\{ -\frac{1}{2\sigma^2}\frac{(\alpha_i - \alpha_j)^2}{(\alpha_i + \alpha_j)^2} \Big\}$$

while we set the weight between non-adjacent nodes to be zero. Then we obtain an undirected graph with weights defined on each edge. See Figure (5).

Remark: We obtain the graph representation from the original image, and we will use it to obtain the BSP smoothed image and output, as we will discuss later.

Figure 5: Graph representation after BSP partition

3.3 BSP smoothed image

Using the graph representation introduced in subsection 3.2, let $(s_i^r, s_i^g, s_i^b), (i = 1, 2, \ldots, N)$ be the vector corresponding to the smoothed image using traditional convolution and denote $\boldsymbol{s}^c = (s_1^c, s_2^c, \ldots, s_n^c)^T$,where $c \in \{r, g, b\}$. We want to generate an "modified" smoothed image which is as close as possible to the smoothed image, but will be free of the problematic boundary effect disscused in subsection 2.2. The modified image should thus have the following properties:

(a) Maintain the enhancement at the boundary as the smoothed image does.

(b) Propagate the signal enhancement to the remaining area of the image

We denote $\boldsymbol{a}^c = (a_1^c, a_2^c, \ldots, a_n^c)^T$ to represent the modified image we want to get. Mathematically, in order to satisfy the properties discussed above, we want a_i and a_j to be close when w_{ij} is large, and a_i close to s_i.

We design the objective function $W((a))$ to be

$$W(\boldsymbol{a^c}) = \frac{1}{2} \sum_{1 \le i,j \le N} w_{ij}(a_i^c - a_j^c)^2 + \sum_{i=1}^{N} d_i^c (a_i^c - s_i^c)^2 \tag{2}$$

where w_{ij} is the weight in the graph, and $d_i^c = |\text{Detail}_i^c|$. In more compact form, we can rewrite $W(\boldsymbol{a}^c)$ as

$$W(\boldsymbol{a}^c) = (\boldsymbol{a}^c)^T L \boldsymbol{a}^c + (\boldsymbol{a}^c - \boldsymbol{s}^c)^T D (\boldsymbol{a}^c - \boldsymbol{s}^c) \tag{3}$$

where the L is the graph Laplacian and $D = diag(d_1^c, \ldots, d_n^c)$. Our goal is to minimize $W(\boldsymbol{a}^c)$ to get the value of $\boldsymbol{a}^c$. Since the graph Laplacian is symmetric positive definite ([2]), we could solve Equation (3) in closed form by setting $\nabla W(\boldsymbol{\alpha}^c) = \mathbf{0}$, we obtain $\boldsymbol{a}^c = (L + D)^{-1} D \boldsymbol{s}^c$

Remark: Note $c \in \{r, g, b\}$, so we are essentially solving (3) three times, for r, g, b separately.

3.4 Output image

Our method for the output image is

$$\text{Output} = \exp\Big\{B \times \text{ Modified Smoothed Image} + \delta \times \text{Detail}\Big\} \tag{4}$$

where B is defined shortly. We are taking the exponential since all the previous computation were using the log of the (r, g, b), see subsection 3.2. In the graph representation, our modified smoothed image is $\boldsymbol{a}^c$. Now let us define $B = diag(b_1^c, \ldots, b_N^c)$. Firstly, chose

$$\lambda_i^c = C|a_i^c - \text{Input}_i^c|/|\text{Input}_i^c|$$

Where C is a scalar constant. This choice is to locally suppress the dynamic range of the modified smoothed image. Then chose $\boldsymbol{b}^c$ to be

$$\boldsymbol{b}^c = \arg\min_{\boldsymbol{b}^c} \Big\{(\boldsymbol{b}^c)^T L \boldsymbol{b}^c + \sum_{i=1}^{N} \lambda_i^c (b_i^c - \lambda_i^c)^2\Big\} \tag{5}$$

which can be solved similarly to Equation (3). The above optimization for $\boldsymbol{b}^c$ is to ensure the adjustment to be aligned with the image structure.

Examples of BSP modified smooth image

After we solve $\boldsymbol{a}$, we use the new modified smoothed graph instead of the original smoothed graph, and by Formula (4) we obtained the following picture, see Figure (6) and (7), which is free of boundary artifacts.

Figure 6: Modified smoothed image

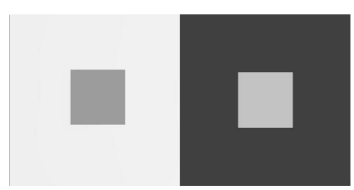
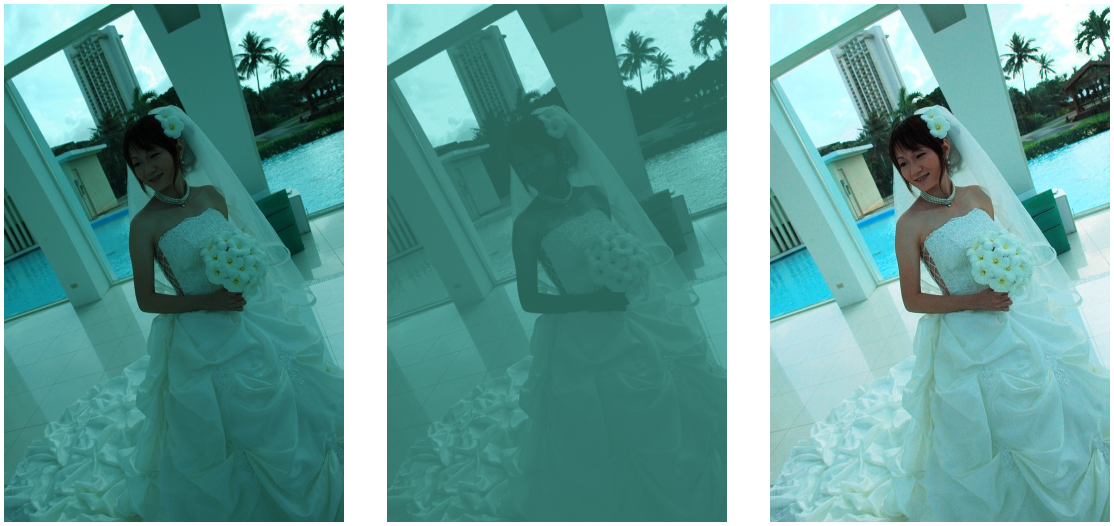


Figure 7: Enhanced image

4 More complicated examples

We applied our method to real photographs, see Figure (8). As demonstrated by our examples,



(a) Input image (b) Modified smoothed image (c) Output image

Figure 8: Image enhancement example

our contrast adjustment method based on BSP can substantially improve the contrast in an image without introducing boundary artifacts.

Acknowledgements

We are grateful for the help of the following individuals: Tung-Yu Wu, from Prof. Wong's lab, for initial ideas and fruitful thoughts, especially for the application of BSP method and for providing the code of BSP; Chen-Yu Tseng for instruction on background knowledge of imaging processing; Yingzhou Li for technical support; and Charles Zheng, for proofreading our report.

References

[1] Lu, L., Jiang, H., Wong, W. H. (2013). Multivariate density estimation by Bayesian Sequential Partitioning. *Journal of the American Statistical Association*, just-accepted.

[2] Cvetkovic, Dragos M., Michael Doob, and Horst Sachs. *Spectra of graphs: Theory and application.* Vol. 413. New York: Academic press, 1980.

[3] Jobson, D. J., Rahman, Z. U., Woodell, G. A. (1997). A multiscale retinex for bridging the gap between color images and the human observation of scenes. *Image Processing, IEEE Transactions on*, 6(7), 965-976.

[4] Land, E. H. (1986), An alternative technique for the computation of the designator in the retinex theory of color vision, *Proc. Nat. Acad. Sci.*, 83(10), pp. 3078 − 3080.