A Learning-to-Rank Approach for Image Color Enhancement

Jianzhou Yan[1*] Stephen Lin[2] Sing Bing Kang[2] Xiaoou Tang[1]
[1]The Chinese University of Hong Kong [2]Microsoft Research

Abstract

We present a machine-learned ranking approach for automatically enhancing the color of a photograph. Unlike previous techniques that train on pairs of images before and after adjustment by a human user, our method takes into account the intermediate steps taken in the enhancement process, which provide detailed information on the person's color preferences. To make use of this data, we formulate the color enhancement task as a learning-to-rank problem in which ordered pairs of images are used for training, and then various color enhancements of a novel input image can be evaluated from their corresponding rank values. From the parallels between the decision tree structures we use for ranking and the decisions made by a human during the editing process, we posit that breaking a full enhancement sequence into individual steps can facilitate training. Our experiments show that this approach compares well to existing methods for automatic color enhancement.

1. Introduction

With the growth of digital photography, photo retouching tools have come into common use, as they provide an effective way to improve the visual quality of images. Retouching large sets of images with software tools such as Adobe Photoshop, however, can be complicated and time-consuming for amateur photographers with little experience in photo editing. This problem underscores the need for automatic photo enhancement tools. Enhancements can target various aspects of a photograph, including its content, composition and color. We focus in this work on image color, which has a significant impact on a photo's appearance.

Recent methods for enhancing the color of digital photographs have utilized image pairs for training, where each pair consists of an image before and after color adjustment by a human user [6, 11, 3, 4]. With such a training set, a regression function is learned that maps the feature vector of a pre-adjusted image to its corresponding enhancements in color. Image features that have been used in this context include color histograms, bags-of-words, and spatial distributions of intensity. The color adjustments to these images have mostly been represented in terms of color parameter changes.

In this paper, we present a new approach that learns not only from images before and after adjustment, but also from the process an expert photographer takes in between. As the expert adjusts various color controls to progress from the original to the final image, she makes a series of decisions that gradually enhance the color in the photograph, as illustrated in Fig. 1. For example, at one point in the process she may decide to change the color saturation of the image by decreasing it a certain amount, and then next she may choose to increase contrast to particular value. As the editing continues, a given color control may be adjusted multiple times at different points in the sequence. The steps that are taken in this manual optimization process shed light on the coloring preferences of the photographer, and we aim in this work to take advantage of this information.

This information, however, is challenging to incorporate into the regression frameworks of recent color enhancement methods. One reason is that each intermediate step itself does not bring an image to its final state. Though applying a regression function in an iterative manner may bring an image towards a solution, there is no guarantee of convergence, particularly for images that are distant from the training images in the image feature space. Another reason is that applying a sequence of intermediate steps may have some susceptibility to error accumulation for the approximate mappings provided by a regression function.

To benefit from this intermediate data, we propose to treat color enhancement as a learning-to-rank (LTR) problem. In LTR, the goal is to automatically construct a ranking model from training data consisting of partial ordering information among a set of training elements. Ranking is a central component of many information retrieval (IR) systems, where a query is given and the most relevant elements are returned in ranked order. More specifically, IR models learn an association of different values to different elements according to their relevance to a query, and these values induce a ranking among a set of elements. For our application of color enhancement, we adapt this approach by using the

[*]This work was done while Jianzhou Yan was a visiting student at Microsoft Research Asia.

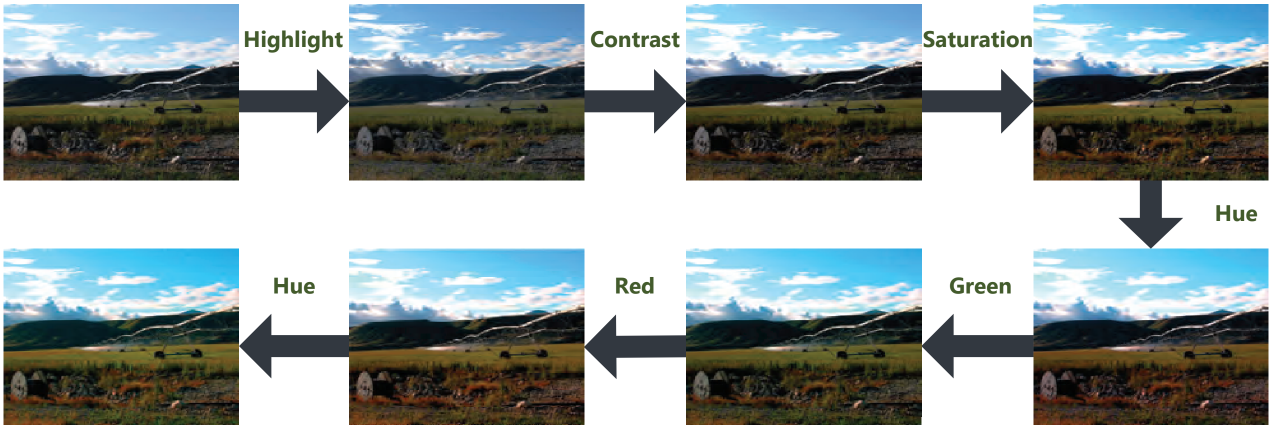


Figure 1. A sequence of enhancement steps by an expert photographer.

feature vector of the input image as the query, and evaluate possible color enhancement solutions according to their ranking values. We employ a pairwise approach [15] for rank learning, in which the training data consists of element pairs with an ordering given between the elements of each pair. For our case, the element pairs are the image feature vectors from the intermediate steps of the expert photographer's enhancement process (e.g., before and after each color adjustment step). In addition, other ordered pairs that can be inferred from the optimization sequence are included for training. To obtain the enhancement result, we sample various series of color adjustments on the input image and take the one that yields the highest rank value.

The LTR method employed in our work is Multiple Additive Regression Trees (MART) [8], a boosted prediction model formed from an ensemble of decision trees. We note an interesting parallel between a human's enhancement process and the operation of decision trees. In the enhancement process, the human makes a series of decisions on what color adjustment to apply, depending on the current appearance of the photograph. Analogously, traversal down a decision tree involves decisions at each node on which branch to take, based on the attribute values of the input. We hypothesize that the training pairs of ordered feature vectors in our work may be well-suited for MART learning because of the similar nature of their intermediate decisions.

In comparison to other regression-based color enhancement methods, our LTR approach has the advantage of utilizing greater data from a given training image, which can be especially beneficial in this application where training sets often provide only a sparse sampling over the space of image feature vectors. We moreover believe that the step-by-step data from an enhancement sequence represents more manageable chunks for training than the conventional image pairs before and after the full series of various adjustments. In our experiments, the effectiveness of this approach is supported through comparisons with state-of-the-art color enhancement techniques.

2. Related Work

Various techniques have been proposed for automatic color enhancement or correction. In the area of color constancy, there exist many works that seek to remove the color cast of an image, which is caused by the color of the scene's lighting. Such methods often employ certain assumptions about scene color, such as the gray-world model [7] or gray-edge hypothesis [20]. Rather than removing color casts, our work aims to improve the perceptual appeal of a photograph through color enhancements.

Numerous methods have also been presented for improving image quality through changes in tone. Most of them specifically target the conversion of high dynamic range (HDR) images into a low dynamic range (24-bit RGB) suitable for conventional display devices [18]. These HDR tone mapping techniques have also been used to enhance local contrast in standard RGB images. Bychkovsky et al. [3] presented a method for learning global tonal adjustments from a database of input-output image pairs. Different from the more general problem of color enhancement, tone mapping deals only with manipulating the luminance channel, mainly for the purpose of improving contrast.

For improving image color, content-aware approaches have applied enhancements targeted to specific objects or regions in a photograph. For example, Kaufman et al. [12] detect and enhance the appearance of human faces, blue skies with or without clouds, and underexposed salient regions. Berthouzoz et al. [1] transfer photo manipulations learned for particular types of image regions. While these methods effectively enhance areas that contain the targeted content, they rely heavily on accurate detection/segmentation of landmarks or regions, and may provide only basic adjustments such as detail enhancement to other image regions. Instead of operating on particular regions, Hwang et al. [9] determine pixel-level adjustments based on local scene descriptors. Since this approach lacks the high-level scene context of content-based methods, the local decisions that are made may not be correct in a global

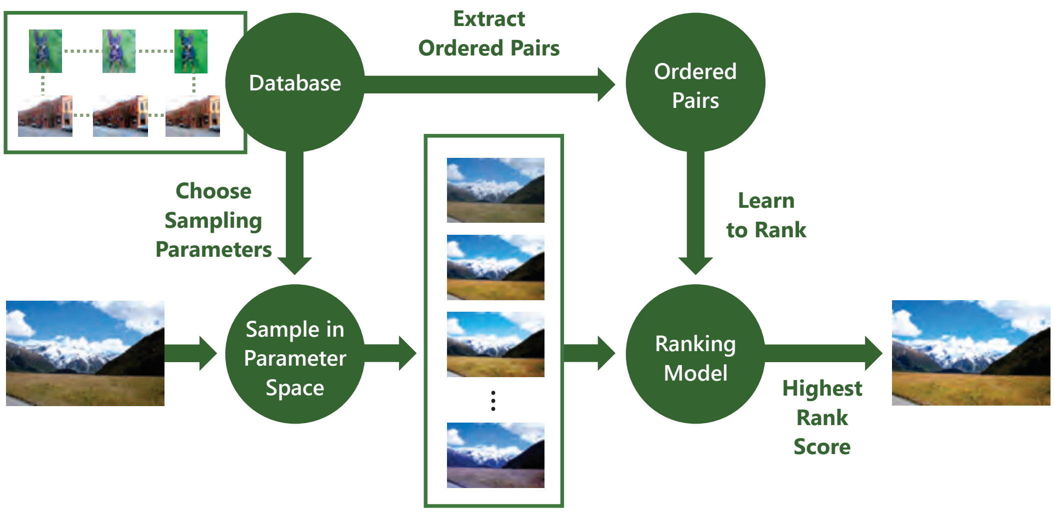


Figure 2. An overview of our learning-to-rank approach for image color enhancement.

sense.

More closely related to our work are global image enhancement methods guided by training images. Dale et al. [6] restore images by finding the closest images in a large database, generating an intermediate solution by transferring the color distributions of corresponding regions, and then determining the restoration operations that bring the input image closest to this intermediate solution. Kang et al. [11] use an examplar-based method that searches for the closest image in a database in terms of enhancement parameter similarity, and then applies its enhancement parameters to the input image. The enhancement is tailored to a given individual, since the database is trained for personal use. The idea of personalized enhancement is advanced further by Caicedo et al. [4], who propose collaborative filtering for revealing clusters of user preferences. In our work, color enhancements are learned from an individual expert photographer, though we believe that the approach we present here may also benefit from personalization.

3. Approach

We treat the color enhancement process as a search for the most visually appealing image after a series of color parameter adjustments. To model this, we regard the feature vector of the input image as a query, and sample various sets of parameter adjustments to find the one that yields a feature vector with the best enhancement result. The enhancement quality of a feature vector is determined from a learning-to-rank model trained on the recorded enhancement processes of an expert photographer on a large dataset of images. Figure 2 presents an overview of our approach.

3.1. Dataset

We collected a dataset that includes about 1,300 expertly-enhanced images from six different categories, namely animal, architecture, human, landscape, man-made object, and plant. The number of images recorded in these categories is 130, 297, 351, 308, 106 and 121, respectively. An example from each category is displayed in Fig. 3. Detailed information on the manual enhancement process of an expert photographer is also recorded for each image. This data is collected from a user interface, shown in Fig. 4. On the left side of the interface is the input image, which is updated as the photographer makes adjustments to color parameters. On the right side are ten slide-bars for manipulating these parameters, which correspond to ten popular color controls: **Contrast** for adjusting the contrast of the image; **Brightness** for adjusting the overall image brightness; **Saturation** for adjusting the saturation of colors throughout the image; **Shadow** for adjusting the darkness of shadows; **Highlight** for adjusting highlight brightness; **Red** for adjusting the red component; **Green** for adjusting the green component; **Blue** for adjusting the blue component; **Sharpen** for adjusting the level of sharpness; and **Hue** for globally adjusting the hues in the image. We record the movements of each slider, including the incremental time spent at each position within the slider range during movement.

3.2. Extraction of Ordered Pairs

To obtain data that reflects the user's decision process, we extract ordered pairs of image feature vectors based on the movements of the sliders. These ordered pairs are determined according to a couple of basic rules:

a. As a slider is moved around to adjust an image, we record each point at which it stops or changes direction. For the images that correspond to these points, we have a sequence of feature vectors, $(v_{a1}, v_{a2}, ..., v_{an})$. Since the feature vector of the final position (v_{an}) should be higher in ranking than those from other positions, we generate the following ordered pairs:

Figure 3. Example training set photos. Top row: Original photos. Bottom row: Photos after manual enhancement by an expert photographer. The intermediate steps of the enhancement process are recorded as well.

Figure 4. User interface for data collection.

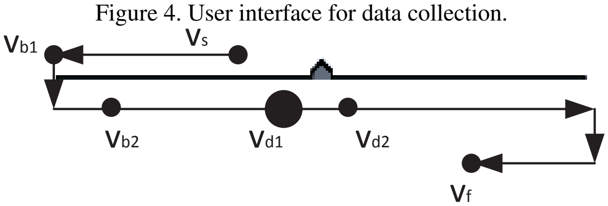


Figure 5. Illustration of implicit human decisions during slider movement, as described in Sec. 3.2.

$\{(v_{an} > v_{a1}), (v_{an} > v_{a2}), ..., (v_{an} > v_{a(n-1)})\}$. If the same slider is adjusted again later, an additional sequence is produced and also used for extraction of additional ordered pairs. Moreover, the final feature vector should be higher in ranking than those for slider positions not examined by the user. Ordered pairs that correspond to this are added for ten evenly sampled slide bar positions within its total range.

b. At each change in slider direction, the human expert implicitly decides that the new direction is better than the point at which the change is made. To model this, we include an ordered pair $\{(v_{b2} > v_{b1})\}$ as illustrated in Fig. 5, where the point $b2$ is displaced from $b1$ in the direction of change by an amount equal to 5% of the total slider range. Likewise, a decision is made when the user hesitates at a position $d1$ (for an amount of time greater than a prescribed threshold) and then moves the slider in one direction (past point $d2$). For such cases, we include the ordered pair $\{(v_{d2} > v_{d1})\}$ as shown in Fig. 5. For both cases, an exception occurs if point $b1$ or $d1$ is passed again later in the adjustment sequence, since the ordered pair then loses its validity. The pair is discarded in such instances.

3.3. Pairwise Rank Learning

With the extracted ordered pairs $\{(v_{g1} > v_{b1}), (v_{g2} > v_{b2}), ..., (v_{gn} > v_{bn})\}$, we use Multiple Additive Regression Trees (MART) [8] to train a ranking model. This pairwise training minimizes the loss function:

$$L(F(x), y) = \sum_{i=1}^{n} l(sign(y_{gi} - y_{bi}), f(x_{bi}) - f(x_{gi})) \quad (1)$$

where x_{bi} is the feature vector of the image with parameter value v_{gi}, y_{gi} is the label of feature x_{gi}, and $y_{gi} > y_{bi}$ in our case where the order within pairs is known.

In MART, the decision tree structure, where at each node a decision is made on which branch to follow, resembles the enhancement process of humans, in which decisions are made on whether to adjust a slide bar to the left or right. This similar operating principle may potentially help MART to more accurately model the human's process, as indicated in comparisons of MART to other pairwise learning-to-rank algorithms (RankSVM and RankNet) in Sec. 4.

Since categorization provides some high-level information about an image and has been shown to improve learning in the related problem of photo quality assessment [13][16][19], we conduct our training and testing on the separate categories of the image dataset.

3.3.1 Color Features

For describing the color characteristics of an image, we extract a set of color features. In this feature representation, we include not only global properties, but also separate color properties of the image foreground, image background, and human faces (if any), since these regions have their own distinct influences on the visual perception of an image.

The extracted color features are listed below:

- Foreground Histogram and Moments We extract the foreground of an image using the method described in [5]. As claimed in [14], the combination of a color histogram and color moments is more descriptive than an individual color descriptor. So we use both a histogram in HSV space (12 bins) and moments in RGB space to describe the foreground color.

- Background Histogram and Moments Similar to the foreground color descriptor, we also use an HSV histogram (6 bins) and RGB moments as color features of the background.

- Face Histogram and Moments We employ the face detector of [21] to locate face regions in an image. The combination of an HSV histogram (6 bins) and RGB color moments is used as a descriptor. When there is no face in the image, we instead compute the histogram and moments over the entire image as done in [3].

- Clipped Pixels An important consideration in color enhancement is the amount of clipping of highlights and shadows. For this, we compute the percentage of pixel values that are clipped.

- Brightness Range We calculate the range of image intensity values in the foreground, background, and face regions.

- Spatial Distributions of Brightness The distribution of brightness over an image has an effect on visual perception. For this, we use the 2D spatial Gaussian model of [3], fit to four intensity intervals but with four bins instead of ten.

- Sharpness The sharpness of an image reflects the magnitude of local color differences. We model an image's sharpness as done in [17], by taking the ratio of the region's high frequency power to its total power:

$$f_{sharpness} = \frac{||C||}{||R||} \tag{2}$$

where R denotes the whole image, $C = \{(u, v) : |F(u, v)| > \theta\}$ for a predefined threshold θ, and $F = FFT(R)$.

3.4. Sampling of Parameter Sequences

The trained ranking model outputs a rank score for the feature vector of any given image. To obtain an enhancement result, we take the input image and sample various sets of color parameter adjustments to generate a set of enhancement candidates. The candidate whose feature vector results in the highest score is taken as the final color enhancement result.

There are different ways to sample sequences of parameter adjustments. One method is to uniformly sample the space of parameter values (i.e., with each axis of the space representing one of the adjustable color parameters), and then compute each candidate by applying the parameter changes in a pre-specified order. But since the number of samples generated in this way would be exponentially related to the number of color properties, we instead optimize for each parameter sequentially, with the training set used to guide the sampling. Specifically, we first take the $K = 10$ nearest neighbors in the training set to the input image in terms of L2 distance in our color feature vector space. Then for the first color parameter, we compute from the K images a weighted sum of time distributions, which record the amount of time spent at a given slider position within its range. The time distributions among the K=10 nearest neighbors to the input image are weighted inversely proportional to their L2 distance from the input image in the color feature vector space. The sum of these weighted time distributions gives a weighted average time distribution from which we sample parameter values. We uniformly sample the inverse cumulative distribution function because this yields denser sampling around values on which more time is spent. (Since the user spent more time around these values, we consider them to be more significant in decision-making, and thus we extract more samples around them.) And then we find which of the sampled values leads to the best image, by applying them and evaluating the resulting image with the ranking model. This process is repeated for each of the color parameters in the order they appear on the user interface, and we reiterate through all the parameters until the parameter change drops below a threshold.

4. Experiments

To assess our technique, we conducted comparisons of our results to those of alternative methods. A total of 124 test images were randomly selected from multiple image categories of our dataset, with none of the test images used in training. Of these, there were 25 images compared to the automatic image enhancer of Google's Picasa, and 24 images compared to the recent technique of Hwang et al. [9], whose results were generated from code provided by the authors. Another 25 images were used in a comparison of our method trained separately on different categories and trained jointly on all the categories together. The remaining 50 images were used to compare our MART-based implementation to versions of our method using two other pairwise ranking algorithms, namely RankSVM [10] and RankNet [2], with 25 images for each comparison.

4.1. L2 Error Comparison

In Table 1, we report a comparison of results in terms of L2 error in L*ab space. The enhancement results produced by the expert photographer were used as the ground truth. Our method (with category-specific training and MART) obtains a lower L2 error than the three comparison versions of our method: category-specific training with RankSVM, category-specific training with RankNet, and all-category training with MART. The table also shows lower L2 error in comparison to Hwang et al. [9] and higher L2 error compared to Picasa. We note as in [9] that L2 error with respect to manual ground truth does not necessarily provide a good predictor of visual quality, but is reported here for reference. In the following subsection, a more informative comparison based on human users is presented.

4.2. User Study

We also evaluated our method through a user study, with 34 participants in total. In the study, each user is shown the sequence of 124 image pairs described above for the five comparisons. Each pair is enhanced from the same original image, with one image being the result of our method, and the other being the result of a comparison method. The left-right order of the images in each pair is randomized, and the images are not labeled by their enhancement method. The users were instructed to double-click the image they

	Picasa	Hwang et al. [9]	Ours (all-cat. MART)	Ours (cat.-spec. RankSVM)	Ours (cat.-spec. RankNet)	Ours (cat.-spec. MART)
L2 Error	7.0506	9.2774	8.3468	8.8010	11.7591	8.3445

Table 1. Comparison of L2 Error.

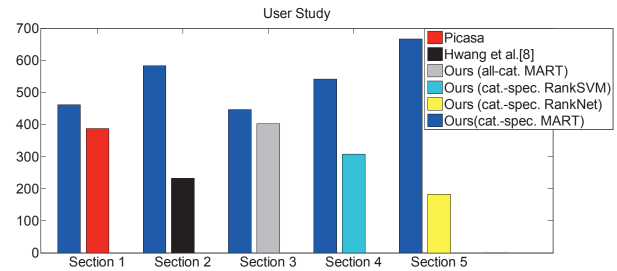


Figure 6. Results of our user study, which include comparisons to Picasa (Section 1, $p < 0.05$), Hwang et al. [9] (Section 2, $p < 0.01$), our method trained with images from all categories images together (Section 3, $p < 0.05$), our method with RankSVM (Section 4, $p < 0.01$), and our method with RankNet (Section 5, $p < 0.01$).

prefer, and no time constraints were placed on making these selections.

The result for each set of comparisons is shown in Figure 6. For each section of the user study, the graph displays the number of times a given method's enhancement result was selected as the better choice. In this comparison, our method trained for individual categories outperforms the other techniques. The study also indicates a slight preference for category-specific training of our method over the training using images from all categories together. A strong preference is shown for our method with MART over the versions with RankSVM or RankNet. To indicate the statistical significance of this study, paired t-test values are given for each comparison.

5. Discussion

Comparisons to other methods From the user study, it was found that users have some preference for the color enhancements of our method over those of Picasa and Hwang et al. [9]. Some example results are shown in Figure 7. We observed that Picasa tends to make relatively conservative color adjustments, with changes mostly to contrast and brightness. This may limit to some degree the range of enhancements obtainable by Picasa. In the results of Hwang et al. [9], we saw a trend towards brightening the images, which at times may lead to some apparent loss of contrast. Since the technique does not include contrast adjustment in its model, a tradeoff in contrast may exist for certain color changes. It may be noticed from the results that our expert photographer tends to favor some boost in the red component for plant and landscape scenes. Though the users in our study appear to generally support such changes, we note that this might not be agreeable to some. These personal differences in color preferences could potentially be accounted for in our method through a user clustering approach as described in [4].

The comparisons indicate some improvement of category-specific training over that with training on all categories together. We found such categorization helpful for images of landscape as they often contain similar elements. For images with man-made objects, categorization appeared to be less helpful, perhaps because of the much broader range of color characteristics among them. We note that the method of Hwang et al. [9], and possibly Picasa as well, do not benefit from the high-level scene context given by categorization. The performance of these methods could potentially improve with category-specific training, though the overall improvement for our method in the user study was modest.

Example results that compare our method with different pairwise ranking algorithms are exhibited in Figure 8. From the user study, a clear preference for MART over RankSVM and RankNet was found. This suggests a greater compatibility between the decision trees of MART and the intermediate decision data collected from the human user.

Training data The number of images in our dataset is smaller than that in others (e.g., 1,300 in ours vs. 5,000 used in [9] from the MIT FiveK dataset [3]). However, our method gathers significantly more information per image than just input/output pairs. Categorization reduces the number of these images used for training, but the included images generally provide the most relevant data for photos in the category. We believe that the man-made object category in our dataset may benefit from further division into sub-categories, due to the breadth of image content within this class. Adding more images to that category would likely also help.

Our method assumes that the expert photographer exercises care in making the color adjustments for our dataset images. Haphazard modifications that do not reflect mindful decisions may introduce poor data into the training. Because of this, the rules in our method for selecting ordered pairs are formulated toward including data that is likely to be reliable. Additional rules could potentially be included to extract a larger set of training pairs.

There are a few interesting observations that can be made from our dataset. One is that there are significant differences among the color parameters with regard to the amount of adjustment time spent by our expert. The parameter that commanded the most time overall was *hue*, perhaps because

(a) (b) (c) (d)

Figure 7. Comparisons with existing methods. (a) Original images. (b) From Picasa. (c) From Hwang et al. [9]. (d) From our method.

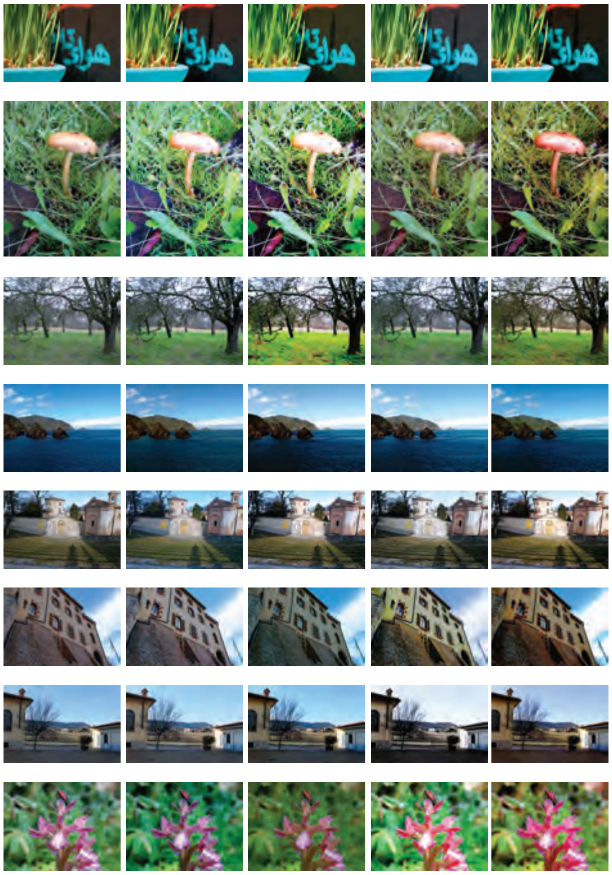
hue manipulations can lead to relatively dramatic changes in visual appearance that require examination of each region in an image. The differences in adjustment time also vary among the image categories. For example, our photographer spent more time in the human category on the highlight parameter than on saturation, and vice versa for the man-made object category. This might be because highlights on faces are an important cue for inferring shape and identity, while natural-looking saturation levels are easier to determine for human skin than for man-made objects in general.

From the dataset, we also noticed some patterns in the sequence that color parameters were adjusted. Our expert appeared to be influenced by the ordering of the color parameters in the user interface, as this order was roughly followed on many of the images, with certain sliders sometimes skipped and the user often going back through the parameters to make additional changes. This observation was used to order the parameter adjustments for candidate sampling in Section 3.4. However, we also saw that some parameters tend to be adjusted one after the other. For example, the expert often modified contrast after changing brightness, which could be explained by the reduction of apparent contrast after brightening an image. A few other parameter pairs that also were adjusted together with some frequency are brightness after saturation, red after green, and sharpness after hue. These pairings may arise due to the cross-influence one color property may have on the perception of another. They also suggest that color enhancement by humans is driven by the current appearance of an image, rather than just the appearance of the original image, and this would make knowledge on intermediate editing steps important in modeling color enhancements by humans. Data on how people search through the color parameter space is a valuable source of information that we capitalize on in this work.

6. Conclusion

We presented a technique for automatic image color enhancement that accounts for the intermediate decisions of a human user in the color editing process. This information is utilized by treating image enhancement as a learning-to-rank problem, and it is indicated through our experiments that decision tree based ranking may be particularly suitable for our training data. The results of a user study support the use of the intermediate data that we collected from an expert photographer.

As our work relies on existing techniques for foreground detection and face detection, shortcomings in these methods can degrade the quality of our results. Both of these problems, however, have been receiving considerable attention in recent years, and further advancements in these areas should benefit our algorithm. In future work, we plan



(a) (b) (c) (d) (e)

Figure 8. Comparisons of different versions of our method. (a) Original images. (b) Enhancement results using our method trained with the images from all categories together. (c) Enhanced images using our category-specific method with RankSVM. (d) Enhanced images using our category-specific method with RankNet. (e) Our results with MART and category-specific training.

to investigate other color features for image representation, and consider local color enhancements in addition to our global adjustments. We also would like to gather and study the color enhancement data of other expert photographers, for the purpose of personalizing color enhancements and to gain additional insight into the enhancement process of people.

7. Acknowledgement

This work is partially supported by the General Research Fund sponsored by the Research Grants Council of Hong Kong (Project No.CUHK 416510) and Guangdong Innovative Research Team Program (No.201001D0104648280).

References

[1] F. Berthouzoz, W. Li, M. Dontcheva, and M. Agrawala. A framework for content-adaptive photo manipulation macros: Application to face, landscape, and global manipulations. *ACM Transactions on Graphics*, 20(2):120:1–120:14, 2011.

[2] C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender. Learning to rank using gradient descent. In *ICML*, pages 89–96, 2005.

[3] V. Bychkovsky, S. Paris, E. Chan, and F. Durand. Learning photographic global tonal adjustment with a database of input/output image pairs. In *CVPR*, pages 97–104, 2011.

[4] J. Caicedo, A. Kapoor, and S. B. Kang. Collaborative personalization of image enhancement. In *CVPR*, pages 249–256, 2011.

[5] M.-M. Cheng, G.-X. Zhang, N. Mitra, X. Huang, and S.-M. Hu. Global contrast based salient region detection. In *CVPR*, pages 409–416, 2011.

[6] K. Dale, M. K. Johnson, K. Sunkavalli, W. Matusik, and H. Pfister. Image restoration using online photo collections. In *ICCV*, pages 2217–2224, 2009.

[7] G. D. Finlayson and E. Trezzi. Shades of gray and colour constancy. In *Color Imaging Conf.*, pages 37–41, 2004.

[8] J. H. Friedman. Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29:1189–1232, 2000.

[9] S. J. Hwang, A. Kapoor, and S. B. Kang. Context-based automatic local image enhancement. In *ECCV*, pages 569–582, 2012.

[10] T. Joachims. Training linear svms in linear time. In *International Conference on Knowledge Discovery and Data Mining*, pages 217–226, 2006.

[11] S. B. Kang, A. Kapoor, and D. Lischinski. Personalization of image enhancement. In *CVPR*, pages 1799–1806, 2010.

[12] L. Kaufman, D. Lischinski, and M. Werman. Content-aware automatic photo enhancement. In *Computer Graphics Forum*, volume 31, pages 2528–2540, 2012.

[13] Y. Ke, X. Tang, and F. Jing. The design of high-level features for photo quality assessment. In *CVPR*, volume 1, pages 419–426. IEEE, 2006.

[14] R. Khan, J. van de Weijer, F. S. Khan, D. Muselet, C. Ducottet, and C. Barat. Discriminative color descriptors. In *CVPR*, pages 2866–2873, 2013.

[15] T.-Y. Liu. *Learning to Rank for Information Retrieval*. Now Publishers Inc, 2009.

[16] W. Luo, X. Wang, and X. Tang. Content-based photo quality assessment. In *ICCV*, 2011.

[17] Y. Luo and X. Tang. Photo and video quality evaluation: Focusing on the subject. In *ECCV*, pages III:386–399, 2008.

[18] E. Reinhard, W. Heidrich, P. Debevec, S. Pattanaik, G. Ward, and K. Myszkowski. *High dynamic range imaging: acquisition, display, and image-based lighting*. Morgan Kaufmann, 2010.

[19] X. Tang, W. Luo, and X. Wang. Content-based photo quality assessment. *Multimedia, IEEE Transactions on*, 15(8):1930–1943, Dec 2013.

[20] J. van de Weijer, T. Gevers, and A. Gijsenij. Edge-based color constancy. *IEEE Trans. Image Proc.*, 16(9):2207–2214, Sept. 2007.

[21] R. Xiao, H. Zhu, H. Sun, and X. Tang. Dynamic cascades for face detection. In *ICCV*, 2007.