Automatic Content-Aware Color and Tone Stylization

Joon-Young Lee
Adobe Research

Kalyan Sunkavalli
Adobe Research

Zhe Lin
Adobe Research

Xiaohui Shen
Adobe Research

In So Kweon
KAIST

Abstract

We introduce a new technique that automatically generates diverse, visually compelling stylizations for a photograph in an unsupervised manner. We achieve this by learning style ranking for a given input using a large photo collection and selecting a diverse subset of matching styles for final style transfer. We also propose an improved technique that transfers the global color and tone of the chosen exemplars to the input photograph while avoiding the common visual artifacts produced by the existing style transfer methods. Together, our style selection and transfer techniques produce compelling, artifact-free results on a wide range of input photographs, and a user study shows that our results are preferred over other techniques.

1. Introduction

Photographers often stylize their images by editing their color, contrast and tonal distributions – a process that requires a significant amount of skill with tools like Adobe Photoshop. Instead, casual users use preset style filters provided by apps like Instagram to stylize their photographs. However, these fixed sets of styles do not work well for every photograph and in many cases, produce poor results.

Example-based style transfer techniques [22, 3] can transfer the look of a given stylized exemplar to another photograph. However, the quality of these results is tied to the choice of the exemplar used, and the wrong choices often result in visual artifacts. This can be avoided in some cases by directly learning style transforms from input-stylized image pairs [5, 25, 12, 28]. However, these approaches require large amounts of training data, limiting them to a small set of styles.

Our goal is to make the process of image stylization adaptive by *automatically* finding the "right" looks for a photograph (from potentially hundreds or thousands of different styles), and *robustly* applying them to produce a diverse set of stylized outputs. In particular, we consider stylizations that can be represented as global transformations of color and luminance. We would also like to do this in an unsupervised manner, without the need for input-

stylized example pairs for different content and looks.

We introduce two datasets to derive our stylization technique. The first is our manually curated *target style database*, which consists of 1500 stylized exemplar images that capture color and tonal distributions that we consider as good styles. Given an input photograph, we would like to automatically select a subset of these style exemplars that will guarantee good stylization results. We do this by leveraging our second dataset – a *large photo collection* that contains millions of photographs and spans the range of styles and semantic content that we expect in our input photographs (e.g., indoor photographs, urban scenes, landscapes, portraits, etc.). These datasets cannot be used individually for stylization; the style dataset is small and does not span the full content-style space, and the photo collection is not curated and contains both good and poorly-stylized images. The key idea of our work is that we can use the large photo collection to learn a content-to-style mapping and bridge the gap between the source photograph and the target style database. We do this in a completely unsupervised manner, allowing us to easily scale to a large range of image content and photographic styles.

We segment the large photo collection into content-based clusters using semantic features, and learn a ranking of the style exemplars for each cluster by evaluating their style similarities to the images in the cluster. At run time, we determine the semantic clusters nearest to the input photograph, retrieve their corresponding stylized exemplar rankings, and sample this set to obtain a diverse subset of relevant style exemplars.

We propose a new robust technique to transfer the global color and tone statistics of the chosen exemplars to the input photo. Doing this using previous techniques can produce artifacts, especially when the exemplar and input statistics are very disparate. We use regularized color and tone mapping functions, and use a face-specific luminance correction step to minimize artifacts in the final results.

We introduce a new benchmark dataset of 55 images with manually stylized results created by an artist. We compare our style selection method with other variants as well as the artist's results through a blind user study. We also evaluate the performance of a number of current statistics-based style transfer techniques on this dataset,

and show that our style transfer technique produces better results than all of them. To the best of our knowledge, this is the first extensive quantitative evaluation of these methods.

The technical contributions of our work include:

1. A robust style transfer method that captures a wide range of looks while avoiding image artifacts,

2. An unsupervised method to learn a content-specific style ranking using semantic and style similarity,

3. A style selection method to sample the ranked styles to ensure both diversity and quality in the results, and

4. A new benchmark dataset with professional stylizations and a comprehensive user evaluation of various style selection and transfer techniques.

2. Related Work

Example-based Style Transfer One popular approach for image stylization is to transfer the style of an exemplar image to the input image. This approach was pioneered by Reinhard et al. [22] who transferred color between images by matching the statistics of their color distributions. There are several subsequent work [24, 19, 18, 21] that improves this technique. There are several methods that compute transfer functions from correspondences [11, 13, 1]. Please refer to [26, 10] for a detailed survey of different color transfer methods. We base our chrominance transfer function on the work of Pitié et al. [18] but add a regularization term to make it robust to large differences in the color distributions being matched.

Style transfer techniques also match the contrast and tone between images. Bae et al. [3] propose a two-scale technique to transfer both global and local contrast. Aubry et al. [2] demonstrate the use of local Laplacian pyramids for contrast and tone transfer. Shih et al. [23] use a multi-scale local contrast transfer technique to stylize portrait photographs. We propose a parametric luminance reshaping curve that is designed to be smooth and avoids artifacts in the results. In addition, we propose a face luminance correction method that is specifically designed to avoid artifacts for portrait shots.

Learning-based Stylization and Enhancement Another approach for image stylization is to use supervised methods to learn style mapping functions from data consisting of input-stylized image pairs. Wang et al. [25] introduce a method to learn piece-wise smooth non-linear color mappings from image pairs. Yan et al. [28] uses deep neural networks to learn local nonlinear transfer functions for a variety of photographic effects. There are also several automatic learning-based enhancement techniques. Kang et al. [15] present a personalized image enhancement framework using distance metric learning. It was extended by [6], which proposes collaborative personalization. Bychkovsky et al. [5] build a reference dataset of input-output image pairs. Hwang et al. [12] propose a context-based local image enhancement method. Yan et al. [27] account for the intermediate decisions of a user in the editing process. While these learning-based methods show impressive adjustment results, collecting training data and generalizing them to a large number of styles is very challenging. In contrast, our technique to learn content-specific style rankings is completely unsupervised and easily generalizes to a large number of content and style classes.

Our technique is similar in spirit to two papers that leverage large image collections to restore/stylize the color and tone of photographs. Dale et al. [7] find visually similar images in a large photo collection, and use their aggregate color and tone statistics to restore the input photograph. This aggregation causes a regression to the mean that is appropriate for image restoration but not stylization. Liu et al. [17] use a user-specified keyword to search for images that are used to stylize the input photo. The final results are highly dependent on the choice of the keyword and it can be challenging to predict the right keywords to stylize a photograph. Our technique automatically predicts the right styles for the input photograph.

3. Overview

Given an input photograph, I, our goal is to automatically create a set of k stylized outputs $O_1, O_2, \cdots, O_k$. In particular, we focus on stylizations that can be represented as global transformations of the input color and luminance values. The styles we are interested in are captured by a curated set of exemplar images $S_1, S_2, \cdots, S_n, (n >> k)$. Using images as style examples makes it intuitive for users to specify the looks they are interested in.

We use an example-based style transfer algorithm to transfer the look of a given exemplar image to the input photograph. While example-based techniques can produce compelling results [10], they often cause visual artifacts when there are strong differences in the input and exemplar images being processed. In this work, we develop regularized global color and tone mapping functions (Sec. 4) that are expressive enough to capture a wide range of effects, but sufficiently constrained to avoid such artifacts.

The quality of the stylized result O is also closely tied to the choice of the exemplar S. Using an outdoor landscape image, for example, to stylize a portrait could lead to poor transfer results (see Fig. 1(b)). It is therefore important to choose the "right" set of exemplar images based on the content of the input photograph. We use a semantic similarity metric – that we learned using a convolutional neural network (CNN) – to match images with similar content. Given this semantic similarity measure, one approach would be to use it directly to find exemplar images with content similar to an input photograph and stylize it. However, the curated exemplar dataset is limited and

(a) Input photograph   (b) Random selection from the style database

(c) Semantic selection from the style database

(d) Semantic selection from the large photo collection

(e) Our style selection

Figure 1. Stylization results with different choices of the exemplar images. All exemplars are shown in insets in the top-left corner.

unlikely to contain style examples for every content class. Using the semantic similarity metric to find the closest stylized exemplar to an input photograph will not guarantee a good match, and as illustrated in Fig. 1(c), could lead to poor stylizations.

In order to learn a content-specific style ranking, we crawl a large collection of Flickr interesting photos $P_1, P_2, \cdots, P_m, (m >> n)$ that cover a wide range of different content with varying styles and levels of quality. A straightforward way of stylizing an input photograph could be to use the semantic similarity measure to directly find matching images from this large collection and transfer their statistics to the input photograph. However, this large collection of photos is not manually curated, and contains images of both good and bad quality. Performing style transfer using the low-quality photographs in the database can lead to poor stylizations, as shown in Fig. 1(d). While these results can be improved by curating the photo collection, this is an infeasible task given the size of the database.

We leverage the large photo collection to learn a style ranking for each content class in an unsupervised way. We cluster the photo collection into a set of semantic classes using the semantic similarity metric (Sec. 5.1). For each image in a semantic class, we vote for the best matching stylized exemplar using a style similarity metric (Sec. 5.2). We aggregate these votes across all the images in the class to build a content-specific ranking of the stylized exemplars.

At run time, we match an input photograph to its closest semantic classes and use the pre-computed style ranking for these classes to choose the exemplars. We use a greedy sampling technique to ensure a diverse set

of examples (Sec. 5.3), and transfer the statistics of the sampled exemplars to the input photograph using our robust example-based transfer technique. As shown in Fig. 1(e), our style selection technique chooses stylized exemplars that are not necessarily semantically similar to the input photograph, yet have the "right" color and tone statistics to transfer, and produces results that are significantly better than the approaches of directly searching for semantically similar images in the style database or the photo collection. Fig. 2 illustrates the overall framework of our stylization system.

4. Robust Example-based Style Transfer

We stylize an input photograph, I, by applying global transforms to match its color and tonal statistics to those of a style example, S. This space of transformations encompasses a wide range of stylizations that artists use, including color mixing, hue and saturation shifts, and non-linear tone adjustments. While a very flexible transfer model can capture a wide range of photographic looks, it is also important that it can be robustly estimated and does not cause artifacts; this is particularly important in our case, where the images being mapped may differ significantly in their content. With this in mind, we design color and contrast mapping functions that are regularized to avoid artifacts.

To effectively stylize images with global transforms, we first compress the dynamic ranges of the two images using a γ ($= 2.2$) mapping and convert the images into the CIELab colorspace (because it decorrelates the different channels

Figure 2. The overall framework of our system.

well). Then, we stretch the luminance (L channel) to cover the full dynamic range after clipping both the minimum and the maximum 0.5 percent pixels of luminance levels, and apply different transfer functions to the luminance and chrominance components.

Chrominance Our color transfer method maps the statistics of the chrominance channels of the two images. We model the chrominance distribution of an image using a multivariate Gaussian, and find a transfer function that creates the output image O by mapping the Gaussian statistics $\mathcal{N}_S(\mu_S, \Sigma_S)$ of the style exemplar S to the Gaussian statistics $\mathcal{N}_I(\mu_I, \Sigma_I)$ of the input image I as:

$$c_O(x) = T(c_I(x) - \mu_I) + \mu_S \quad \text{s.t.} \quad T\Sigma_I T^\top = \Sigma_S, \quad (1)$$

where T is a linear transformation that maps chrominance between the images and $c(x)$ is the chrominance at pixel x. Following Pitié and Kokaram [18], we solve for the color transform using the following closed form solution:

$$T = \Sigma_I^{-1/2} \left(\Sigma_I^{1/2} \Sigma_S \Sigma_I^{1/2}\right)^{1/2} \Sigma_I^{-1/2}. \quad (2)$$

This solution is unstable for low input covariance values, leading to color artifacts when the input has low color variation. To avoid this, we regularize this solution by clipping diagonal elements of Σ_I as:

$$\Sigma_I' = \max(\Sigma_I, \lambda_r \mathbb{I}), \quad (3)$$

and substitute it into Eq. (2). Here $\mathbb{I}$ is an identity matrix. This formulation has the advantage that it only regularizes colors channels with low variation without affecting the others. We use a regularization of $\lambda_r = 7.5$.

Luminance We match contrast and tone using histogram matching between the luminance channels of the input and style exemplar images. Direct histogram matching typically results in arbitrary transfer functions and may produce artifacts due to non-smooth mapping or excessive stretching/compressing of the luminance values. Instead, we design a new parametric model of luminance mapping that allows for strong expressiveness and regularization simultaneously. Our transfer function is defined as:

$$l_O(x) = g(l_I(x)) = \frac{\arctan(\frac{m}{\delta}) + \arctan(\frac{l_I(x)-m}{\delta})}{\arctan(\frac{m}{\delta}) + \arctan(\frac{1-m}{\delta})}, \quad (4)$$

where $l_I(x)$ and $l_O(x)$ are the input and output luminance respectively, and m and δ are the two parameters of the mapping function. m determines the inflection point of the mapping function and δ determines the degree of luminance stretching around the inflection point. This parametric function can represent a diverse set of tone mapping curves and we can easily control the degree of stretching/compressing of tone. Since the derivative of Eq. (4) is always positive and continuous, it is guaranteed to be a smooth and monotonically increasing curve. This ensures that this mapping function generates a proper luminance mapping curve for any set of parameters.

We extract a luminance feature, L, that represents the luminance histogram with uniformly sampled percentiles of the luminance cumulative distribution function (we use 32 samples). We estimate the tone-mapping parameters by minimizing the cost function:

$$(\hat{m}, \hat{\delta}) = \arg\min_{m,\delta} \|g(L_I) - \tilde{L}\|^2,$$
$$\text{s.t.} \quad \tilde{L} = L_I + (L_S - L_I)\frac{\tau}{\min(\tau, |L_S - L_I|_\infty)}, \quad (5)$$

where L_I and L_S represent the input and style luminance features, respectively. $\tilde{L}$ is an interpolation of the input and exemplar luminance features and represents how closely we want to match the exemplar luminance distribution. We set τ to 0.4 and minimize this cost using parameter sweeping in a branch-and-bound scheme.

Fig. 3 compares the quality of our style transfer method against three recent methods: the N-dimensional histogram matching technique of Pitié et al. [19], the linear Monge-Kantarovich solution of Pitié and Kokaram [18], and the three-band method of Bonneel et al. [4]. While each of these algorithms has its strengths, only our method consistently produces visually compelling results without any artifacts. We further evaluate all these methods via a comprehensive user study in Sec. 6.

Face exposure correction In the process of transferring tonal distributions, our luminance mapping method can over-darken some regions. When this happens to faces, it detracts from the quality of the result, as humans are sensitive to facial appearance. We fix this using a face-specific luminance correction. We detect face regions in

(a) Input   (c) PDF   (d) MK   (e) SMH   (f) Ours

Figure 3. Examples of our style transfer results compared with previous statistics-based transfer methods. Exemplars are shown in insets in the top-left corner of input images.

(a) Input photograph   (b) Our stylization   (c) Our stylization and face correction

Figure 4. Face exposure correction.

the input image, given by center p and radius r, using the Haar cascade classifier implemented in the OpenCV. If the median luminance in a face region, $\bar{l}$, is lower than a threshold l_{th}, we correct the luminance as:

$$
\begin{aligned}
\hat{l}(x) &= (1 - w(x)) \cdot l(x) + w \cdot l(x)^{\gamma} \;\; \text{if } \bar{l} < l_{\text{th}}, \\
w(x) &= \exp(-\alpha_r \|(x - p)/r\|^2) \exp(-\alpha_c \|c - \bar{c}\|^2), \\
\gamma &= \max(\gamma_{\text{th}}, \bar{l}/l_{\text{th}}). \qquad\qquad (6)
\end{aligned}
$$

This technique applies a simple γ-correction to the luminance, where γ_{th} determines the maximum level of exposure correction. We would like to apply it to the entire face; however, the face region is given by a coarse box and applying the correction to the entire box will produce artifacts. Instead we interpolate the corrected luminance with the original luminance using weights $w(x)$. We compute these weights based on spatial distance from the face center, and chrominance distance from the median face chrominance value, $\bar{c}$ (to capture the color of the skin). α_r and α_c are normalization parameters that control the weights of the spatial and chrominance kernels respectively. We set $\{l_{th}, \gamma_{\text{th}}, \alpha_r, \alpha_c\}$ to $\{0.5, 0.5, 0.45, 0.001\}$. Fig. 4 shows an example of our face exposure correction results.

5. Content-aware Style Selection

Given the target style database[1], we can use the method described in Sec. 4 to transfer the photographic style of a style exemplar to an input photograph. However, as noted in Sec. 3 and illustrated in Fig. 1, it is important that we choose the right set of style exemplars. Motivated by the fact that images with different semantic content require different styles, we attempt to learn the set of good styles (or their

---

[1]a curated dataset of 1500 exemplar style images

Figure 5. Examples of semantic clusters.

ranking) for each type of semantic content separately.

To achieve this, we prepare a large photo collection consisting of one million photographs downloaded from Flickr's daily interesting photograph collection[2]. As noted in Sec. 3, the curated style dataset does not contain examples for all content classes and cannot be directly used to stylize a photograph. However, by leveraging the large photo collection, we can learn style rankings of the curated style dataset even for content classes that are not represented in it.

5.1. Semantic clustering

Inspired by recent breakthroughs in the use of CNN [16], we represent the semantic information of an image using a CNN feature, trained on the ImageNet dataset [8]. We modified the *CaffeNet* [14] to have fewer nodes in the fully-connected layers[3] and fine-tuned the modified network. This results in a 512-dimensional feature vector for each image. We empirically found that this smaller CNN captures more style diversity in each content cluster compared to the original *CaffeNet* or *AlexNet* [8] which sometimes "oversegments" content into clusters with low style variation.

We perform k-means clustering on the CNN feature vectors for each image in the large photo collection to obtain semantic content clusters. A small number of clusters leads to different content classes being grouped in the same cluster, while a large number of clusters lead to the style variations of the same content class of images being split into different clusters. In our experiments, we found that using 1000 clusters was a good balance between these two aspects.

Fig. 5 shows images from four different semantic clusters. The images in a single cluster share semantically similar content but have diverse appearances (including both good and bad styles). These intra-class style variations allow us to learn the space of relevant styles for each class.

5.2. Style ranking

To choose the best style example for each semantic cluster, we compute style similarity between each style example and the images in a cluster, and use this measure

---

[2]https://www.flickr.com/services/api/flickr.interestingness.getList.html
[3]We reduced the number of nodes in the FC6 layer from 4096 to 512 and removed the FC7 layer.

(a) Input photograph   (b) Three nearest semantic clusters   (c) Five exemplars chosen by our style selection method

(d) Images in the cluster with the highest votes for the chosen exemplars in (c)

(e) Stylization results by transferring the styles of the exemplars in (c)

Figure 6. Intermediate steps of style selection. The input (a) can be semantically different from the selected exemplars (c) (second and third example especially). However, the cluster images with the highest votes for these style exemplars (d), are both semantically similar to the input and stylistically similar to the chosen exemplars. This ensures input-exemplar compatibility and leads to artifact-free stylizations (e).

to rank the styles for that cluster. As explained in Sec. 4, we represent a photograph's style using chrominance and luminance statistics. Following this, we define the style similarity measure between cluster photograph P and style image S as:

$$\mathcal{R}(P, S) = \exp\left(-\frac{\mathcal{D}_e(L_P, L_S)^2}{\lambda_l}\right) \exp\left(-\frac{\mathcal{D}_h(\mathcal{N}_P, \mathcal{N}_S)^2}{\lambda_c}\right), \quad (7)$$

where $\mathcal{D}_e$ represents the Euclidean distance between the two luminance features, and λ_l and λ_c are normalization parameters. We set $\lambda_l = 0.005$ and $\lambda_c = 0.05$ to generate all our results. $\mathcal{D}_h$ is the Hellinger distance [20] defined as:

$$\mathcal{D}_h(\mathcal{N}_P, \mathcal{N}_S) = 1 - \frac{|\Sigma_P \Sigma_S|^{1/4}}{|\bar{\Sigma}|^{1/2}} \exp\left(-\frac{1}{8}\bar{\mu}^\top \bar{\Sigma}^{-1} \bar{\mu}\right)$$

$$\text{s.t} \quad \bar{\mu} = |\mu_P - \mu_S| + \epsilon, \quad \bar{\Sigma} = \frac{\Sigma_P + \Sigma_S}{2}, \quad (8)$$

where $\mathcal{N}_P = (\mu_P, \Sigma_P)$ are the multivariate Gaussian statistics of chrominance channel for an image. We chose the Hellinger distance to measure the overlap between two distributions because it strongly penalizes large differences in covariance even if the means are close enough. $\epsilon = 1$ is added to the difference between the means to additionally penalize small covariance images.

We measure the compatibility of a stylized exemplar S, with a semantic cluster C_K, by aggregating the style similarity measure over all the images in the cluster as

$$\bar{\mathcal{R}}_K(S) = \sum_{P \in C_K} \mathcal{R}(P, S). \quad (9)$$

For each semantic cluster, we compute $\bar{\mathcal{R}}$ for all the style exemplars and determine the style example ranking by sorting $\bar{\mathcal{R}}$ in decreasing order. This voting scheme measures how often a particular exemplar's color and tonal statistics occurs in the semantic cluster. Poorly stylized cluster images are implicitly filtered out because they do not vote for any style exemplar. Meanwhile, well stylized images in the cluster vote for their corresponding exemplars, giving

us a "histogram" of the style exemplars for that cluster.

Figs. 2 and 6 show the results of each stage of our stylization pipeline. As these figures illustrate, our semantic similarity term is able to find clusters with semantically similar content (see Fig. 6(b)). Our technique does not require the selected style exemplars to be semantically similar to the input image (see Fig. 6(c)). While this might seem counter-intuitive, the final stylized results do not suffer from any artifacts because the highly-ranked styles have the same style characteristics as a large number of "auxiliary exemplars" in the training photo collection that, in turn, share the same content as the input (see Fig. 6(d)). This is an important property of our style selection scheme, and is what allows it to generalize a small style dataset to arbitrary content.

5.3. Style sampling

Given an input photograph, we can extract its semantic feature and assign it to the nearest semantic cluster. We can retrieve the pre-computed style ranking for this cluster and use the top k style images to create a set of k stylized renditions of the input photograph. However, this strategy could lead to outputs that are similar to each other. In order to improve the diversity of styles in the final results, we propose the following multi-cluster style sampling scheme.

Adjacent semantic clusters usually share similar high-level semantics but different low-level features such as object scale, color, and tone. Therefore we propose using *multiple* nearest semantic clusters to capture more diversity. We merge the style lists for the chosen semantic clusters and order them by the aggregate similarity measure (Eq. (9)). To avoid redundant styles, we sample this merged style list in order (starting with the top-ranked one) and discard styles that are within a specified threshold distance from the styles

that have already been chosen.

We define a new similarity measure for this sampling process that computes the squared Fréchet distance [9]:

$$\mathcal{D}_f(\mathcal{N}_P, \mathcal{N}_Q) = \sqrt{\|\mu_P - \mu_Q\|^2 + \mathrm{tr}[\Sigma_P + \Sigma_Q - 2(\Sigma_P \Sigma_Q)^{1/2}]}.$$

We use this distance because it measures optimal transport between distributions and is more perceptually linear. We use three semantic clusters and set the threshold to 7.5.

6. Results and Discussion

We have tested our automatic stylization results on a wide range of input images, and show a subset of our results in Figs. 1, 2, 6, and 7. Please refer to the supplementary material and video for more examples, comparisons, and a real-time demo of our technique. As can be seen from these results, our stylization method can robustly capture fairly aggressive visual styles without creating artifacts, and is able to generate diverse stylization results. Figs. 1, 2, 6, and 7 also show the automatically chosen style examples that were used to stylize the input photographs. As expected, in most cases, the style examples chosen have different semantics from the input image, but the stylizations are still of high-quality. This verifies the advantage of our method when given only a limited set of stylized exemplars.

User study Due to the subjective nature of image stylization, we validated our stylization technique through user studies that evaluate our style selection and style transfer strategies. For the study, we created a benchmark dataset of 55 images – 50 images were randomly chosen from the FiveK dataset [5] and the rest were downloaded from Flickr. We resized all test images to 500-pixels wide on the long edge and stored them using an 8-bit sRGB JPEG format.

We asked a *professional artist* to create five diverse stylizations for every image in our benchmark dataset as a baseline for evaluation. The artist was told to only use tools that globally edit the color and tone; he used the 'Levels', 'Curves', 'Exposure', 'Color Balance', 'Hue/Saturation', 'Vibrance', and 'Black and White' tools in Adobe Photoshop. Creating five different looks for every photograph is challenging even for professional artists. Instead, our artist first constructed 27 different looks, each of which evoked a particular theme (like 'old photo', 'sunny', 'romantic', etc.), applied all of them to all the images in the dataset, and picked the five diverse styles that he preferred the most.

We performed two user studies. In *Study 1*, we evaluated two style selection methods, *our style selection* and *direct semantic search* which directly searches for semantically similar images in the style database. We also explored directly searching in the photo collection using semantic similarity, but its results were consistently poor, which led us to drop this selection method in the larger study. To assess the effect of the size of the style database on the selection algorithm, we tested against two style databases:

the full database with 1500 style exemplars, and a small database with 50 style exemplars randomly chosen from the full database.

We compared five different groups of stylization results including: the reference dataset retouched by a *professional* (henceforth, PRO), our style selection with the full style database (OURS 1500) and the small style database (OURS 50), direct semantic search on the full style database (DIRECT 1500) and the small style database (DIRECT 50). For both *our style selection* and *direct semantic search*, we apply the same style sampling in Sec. 5.3 to achieve the similar levels of style diversity and create the results using the same style transfer technique (Sec. 4). Please see the supplementary material for all these results.

For each image in the benchmark dataset, we showed users five groups of five stylized results (one set each from OURS 1500, OURS 50, DIRECT 1500, DIRECT 50, and PRO). Users were asked to rate the stylization quality of each group of results on a five-point Likert scale ranging from 1 (worst) to 5 (best). A total of 37 users participated in this study, and a total of 1498 different image groups were rated, giving us an average of 27.24 ratings per group.

Fig. 8(a) shows the result of Study 1. In this study, OURS 1500 (3.820 ± 0.403) outperforms all the other techniques. We reported the mean of all user ratings and the standard deviation of the average scores of each of the 55 benchmark images. DIRECT 1500 (3.169 ± 0.444) is substantially worse than OURS 1500. When the style database becomes smaller, the performance of *direct search* drops dramatically (2.421 ± 0.436 for DIRECT 50) while *our style selection* stays stable (3.620 ± 0.413 for OURS 50). We believe that this is a result of our novel two-step style ranking algorithm that is able to learn the mapping between semantic content and style even with very few style examples. On the other hand, direct search fails to find good semantic matches when the size of the style database is reduced significantly. Interestingly, we found that even when direct search finds a semantically meaningful match, this does not guarantee a good style transfer result. An example of this is shown in Fig. 9, where the green in the background of the exemplar image influences the global statistics and causes the girl's skin to take on an undesirable green tone. Our technique aggregates style similarity across many images giving it robustness to such scenarios.

It is also worth noting that PRO (2.881 ± 0.480) got a lower mean score than {OURS 1500, OURS 50, DIRECT 1500} with the largest standard deviation of scores. We attribute this to two reasons. First, the artist-created filters do not adapt to the content of the image in the same way our example-based style transfer technique does. Second, image stylization tends to be subjective in nature; some of users might be uncomfortable with the aggressive stylizations of a professional, while our style selection is learned
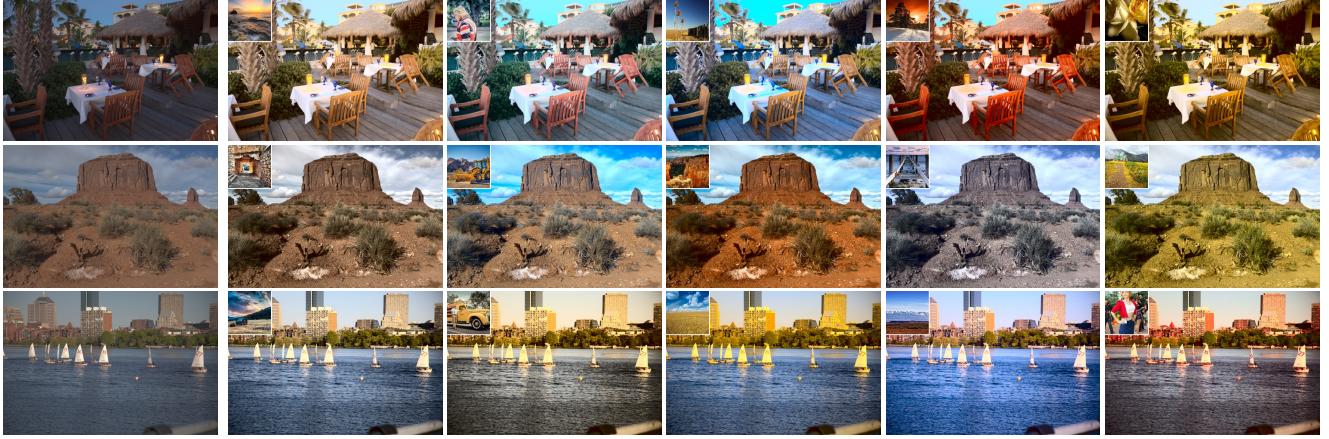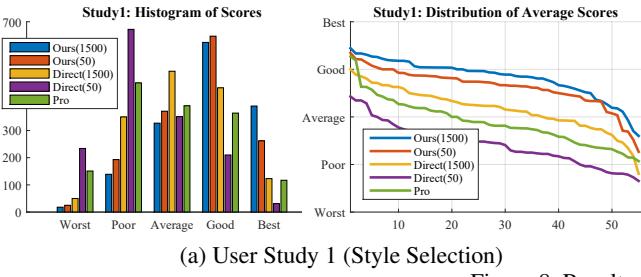
Figure 7. Our stylization results. The left most images are input photographs and the right images are our automatically stylized results.


(a) User Study 1 (Style Selection) (b) User Study 2 (Style Transfer)
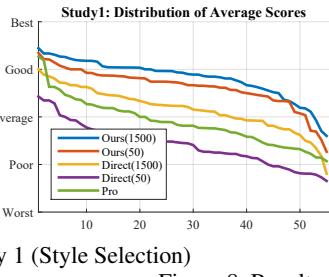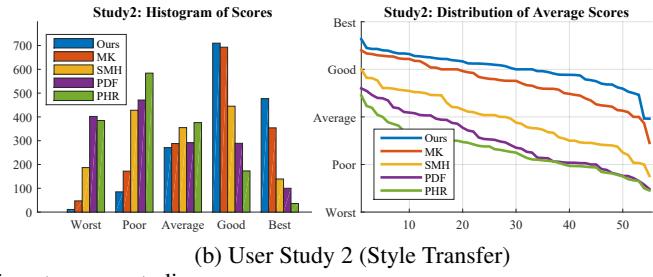
Figure 8. Results of our two user studies.

from a more 'natural' style database and does not have the same level of stylization.

In *Study 2*, we compare our style transfer technique with four different statistics-based style transfer techniques: MK, which computes an affine transform in CIELab [18], SMH, which combines three different affine transforms in different luminance bands with a non-linear tone curve [4], PDF, which use 3-d histogram matching in CIELab [19], and PHR, which progressively reshapes the histograms to make them match [21]. We used our implementation for the MK method and used the original authors' code for the other methods. Style exemplars are chosen by our style selection and these methods are used only for the transfer. We showed users an input photograph, an exemplar, and a randomly arranged set of five stylized images created using the techniques, and asked them to rate the results in terms of style transfer and visual quality on a five-point Likert scale ranging from 1 (worst) to 5 (best). 27 participants from the same pool as (Study 1) participated in this study; they rated 1554 results in total giving us 5.65 ratings per input-style pair and 28.25 rating per input.

Fig. 8(b) shows the result of Study 2. In this study, OURS (4.002 ± 0.336) records the best rating, while MK (3.730 ± 0.440) is ranked second. SHM (2.949 ± 0.545), PDF (2.494 ± 0.577), and PHR (2.286 ± 0.452) are less favored by users. These three techniques have more expressive color transfer models leading to over-fitting and poor results

(a) Input (b) Exemplar (c) Stylization result

Figure 9. Failure case of direct search.

in many cases. This demonstrates the importance of the style transfer technique for high-quality stylization; our technique balances expressiveness and robustness well.

7. Conclusion

In this work, we have proposed an automatic technique to stylize photographs based on their content. Given a set of target photographic styles, we leverage a large collection of photographs to learn a content-specific style ranking in a completely unsupervised manner. At run-time, we use the learned content-specific style ranking to adaptively stylize images based on their content. Our technique produces a diverse set of compelling, high-quality stylized results. We have extensively evaluated both style selection and transfer components of our technique and studies show that users clearly prefer our results over other variations of our pipeline.

References

[1] X. An and F. Pellacini. User-controllable color transfer. In *CGF*, volume 29, pages 263–271, 2010. 2

[2] M. Aubry, S. Paris, S. W. Hasinoff, J. Kautz, and F. Durand. Fast local laplacian filters: Theory and applications. *ACM TOG*, 33(5):167:1–167:14, Sept. 2014. 2

[3] S. Bae, S. Paris, and F. Durand. Two-scale tone management for photographic look. *ACM TOG (Proc. SIGGRAPH)*, 25(3):637–645, July 2006. 1, 2

[4] N. Bonneel, K. Sunkavalli, S. Paris, and H. Pfister. Example-based video color grading. *ACM TOG (Proc. SIGGRAPH)*, 32(4):39:1–39:12, July 2013. 4, 8

[5] 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. 1, 2, 7

[6] J. C. Caicedo, A. Kapoor, and S. B. Kang. Collaborative personalization of image enhancement. In *CVPR*, pages 249–256, 2011. 2

[7] K. Dale, M. K. Johnson, K. Sunkavalli, W. Matusik, and H. Pfister. Image restoration using online photo collections. In *ICCV*, pages 2217–2224, 2009. 2

[8] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009. 5

[9] D. Dowson and B. Landau. The fréchet distance between multivariate normal distributions. *Journal of Multivariate Analysis*, 12(3):450–455, 1982. 7

[10] H. S. Faridul, T. Pouli, C. Chamaret, J. Stauder, A. Trémeau, E. Reinhard, et al. A survey of color mapping and its applications. In *Eurographics*, pages 43–67, 2014. 2

[11] Y. HaCohen, E. Shechtman, D. B. Goldman, and D. Lischinski. Non-rigid dense correspondence with applications for image enhancement. In *ACM TOG (Proc. SIGGRAPH)*, volume 30, page 70, 2011. 2

[12] S. J. Hwang, A. Kapoor, and S. B. Kang. Context-based automatic local image enhancement. In *ECCV*, volume 7572, pages 569–582, 2012. 1, 2

[13] Y. Hwang, J.-Y. Lee, I. S. Kweon, and S. J. Kim. Color transfer using probabilistic moving least squares. In *CVPR*, 2014. 2

[14] Y. Jia. Caffe: An open source convolutional architecture for fast feature embedding. http://caffe.berkeleyvision.org/, 2013. 5

[15] S. B. Kang, A. Kapoor, and D. Lischinski. Personalization of image enhancement. In *CVPR*, pages 1799–1806, 2010. 2

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012. 5

[17] Y. Liu, M. Cohen, M. Uyttendaele, and S. Rusinkiewicz. Autostyle: Automatic style transfer from image collections to users' images. In *CGF*, volume 33, pages 21–31, 2014. 2

[18] F. Pitié and A. Kokaram. The linear monge-kantorovitch linear colour mapping for example-based colour transfer. In *CVMP*, 2007. 2, 4, 8

[19] F. Pitié, A. C. Kokaram, and R. Dahyot. N-dimensional probability density function transfer and its application to color transfer. In *ICCV*, volume 2, pages 1434–1439, 2005. 2, 4, 8

[20] D. Pollard. *A user's guide to measure theoretic probability*, volume 8. Cambridge University Press, 2002. 6

[21] T. Pouli and E. Reinhard. Progressive color transfer for images of arbitrary dynamic range. *Comp. & Graph.*, 35(1):67–80, 2011. 2, 8

[22] E. Reinhard, M. Ashikhmin, B. Gooch, and P. Shirley. Color transfer between images. *CG&A*, 21(5):34–41, 2001. 1, 2

[23] Y. Shih, S. Paris, C. Barnes, W. T. Freeman, and F. Durand. Style transfer for headshot portraits. *ACM TOG (Proc. SIGGRAPH)*, 33(4):148:1–148:14, July 2014. 2

[24] Y.-W. Tai, J. Jia, and C.-K. Tang. Soft color segmentation and its applications. *PAMI*, 29(9):1520–1537, 2007. 2

[25] B. Wang, Y. Yu, and Y.-Q. Xu. Example-based image color and tone style enhancement. In *ACM TOG (Proc. SIGGRAPH)*, volume 30, page 64. ACM, 2011. 1, 2

[26] W. Xu and J. Mulligan. Performance evaluation of color correction approaches for automatic multi-view image and video stitching. In *CVPR*, pages 263–270, 2010. 2

[27] J. Yan, S. Lin, S. B. Kang, and X. Tang. A learning-to-rank approach for image color enhancement. In *CVPR*, pages 2987–2994, 2014. 2

[28] Z. Yan, H. Zhang, B. Wang, S. Paris, and Y. Yu. Automatic photo adjustment using deep neural networks. *CoRR*, abs/1412.7725, 2015. 1, 2