# Understanding and Improving the Realism of Image Composites

Su Xue
Yale University

Aseem Agarwala
Adobe Systems, Inc.

Julie Dorsey
Yale University

Holly Rushmeier
Yale University

**Figure 1:** *Our method can automatically adjust the appearance of a foreground region to better match the background of a composite. Given the proposed foreground and background on the left, we show the compositing results of unadjusted cut-and-paste, Adobe Photoshop's Match Color, the method of Lalonde and Efros[2007], and our method.*

## Abstract

Compositing is one of the most commonly performed operations in computer graphics. A realistic composite requires adjusting the appearance of the foreground and background so that they appear compatible; unfortunately, this task is challenging and poorly understood. We use statistical and visual perception experiments to study the realism of image composites. First, we evaluate a number of standard 2D image statistical measures, and identify those that are most significant in determining the realism of a composite. Then, we perform a human subjects experiment to determine how the changes in these key statistics influence human judgements of composite realism. Finally, we describe a data-driven algorithm that automatically adjusts these statistical measures in a foreground to make it more compatible with its background in a composite. We show a number of compositing results, and evaluate the performance of both our algorithm and previous work with a human subjects study.

**CR Categories:** I.4.10 [Image Processing and Computer Vision]: Image Representation—Statistical

**Links:** DL PDF WEB DATA

## 1 Introduction

Compositing is a fundamental operation in computer graphics. Combining a foreground object from one image with the background of another requires two operations to achieve a realistic composite: first, extract the foreground object by computing an alpha matte, and second, adjust the appearance of the foreground relative to its new background so that the two appear natural together. While the first problem has received considerable attention in the research community [Smith and Blinn 1996; Wang and Cohen 2007; Rhemann et al. 2009], the second has not been systematically studied. Professional compositors have several rules of thumb, but in the end, most composites are made realistic by trial-and-error adjustment of standard image controls such as brightness, contrast, hue, and saturation.

In this paper, we use statistical and visual perception experiments to study the factors that influence the realism of image composites, and propose an automated method to increase their realism. The scope of this problem is large: composite realism is influenced by semantics (e.g., is a polar bear in a rainforest realistic), and factors that require 3D reasoning to analyze (e.g., inter-reflections). However, perception research has shown that the human visual system is remarkably insensitive to certain lighting inconsistencies within an image, such as shadow directions and highlight placements [Ostrovsky et al. 2005; Lopez-Moreno et al. 2010]. If a user is reasonably careful in choosing and locating a foreground relative to a background, many composites can be made to appear realistic by performing standard image processing operations, such as color, brightness, and contrast transformations. We therefore limit the scope of our study to 2D image processing operations, and leave 3D effects and semantics to the user (or other techniques). However, adjusting an image composite to appear realistic with 2D operations is still highly challenging for a novice user, with many degrees of freedom whose correct values often seem ambiguous. Professional compositors are typically able to achieve much better results, and thus automation would be very helpful for novices. We also seek a deeper, evidence-based understanding of the factors that influence human perception of composite realism.

We are interested in three main questions. First, what are the key statistical properties that control the realism of an image composite? Second, how do variations in these properties affect human judgement of a composite's realism? Third, can an algorithm automatically adjust these properties to improve the realism of a specific composite? To answer these questions we perform three tasks. First, we use a large, labeled database of images that contain a foreground object and background, and compare a number of common image statistical measures to see which are the most correlated between foreground and background. Second, we select the most correlated statistics, and perform a human subjects study to test how changes in these statistics influence human ratings of realism. That is, we take natural images and manipulate the foreground and background to introduce statistical deviations, and measure the induced decrease in human realism ratings. Third, we use the insights gleaned from the above experiments to design an algorithm to ad-

**ACM Reference Format**
Xue, S., Agarwala, A., Dorsey, J., Rushmeier, H. 2012. Understanding and Improving the Realism of Image Composites. *ACM Trans. Graph. 31* 4, Article 84 (July 2012), 10 pages. DOI = 10.1145/2185520.2185580 http://doi.acm.org/10.1145/2185520.2185580.

just a foreground to match a background. More specifically, we find that statistics between foreground and background typically match better for one *zone* of the histogram (for example, the luminance zones of highlights, mid-tones, or shadows) than for the histogram as a whole, and the best zone varies from composite to composite. Matching zones is also common practice for professional compositors [Alexander 2011]. We therefore use machine learning techniques to predict which zone to match for a specific composite. Finally, we validate and show that our technique creates more realistic composites than previous automated techniques.

### 1.1 Related Work

The ideal method to adjust a foreground image to match a given background is to simulate its appearance under the same illumination that produced the background image. Since this task is beyond the ability of today's algorithms, most techniques simply shift the appearance of the foreground to better resemble the background. For example, color transfer techniques [Reinhard et al. 2001; Reinhard et al. 2004; Pouli and Reinhard 2010] align the means and variances of the color histograms of the two image regions. Adobe Photoshop's widely used "Match Color" feature is based on this idea. The problem with this approach is that it conflates the effects of reflectance and illumination; e.g., a forest is not green because of a green light source, and thus it is not correct to turn the skin of a person added to the scene green. One can shoot a realistic image of a red car in a desert, even though their color distributions will be very different. Lalonde and Efros et al. [2007] add to this technique by estimating the co-occurrence probability of the color distributions by finding nearest neighbors in an image database. However, this non-parametric approach requires a large database to be searched for each composite, and depends on the presence of several images that are similar to the target composite. The main focus of their work is to predict whether a given composite will appear realistic. However, they show composite adjustment as an additional application; we therefore compare the results of our method to this approach and to color transfer in Section 5.

Professional compositors employ a number of principles for adjusting a composite. One professional compositor [Alexander 2011] told us that compositors often focus on luminance zones. First, they isolate the highlights and match their color and brightness, then balance the mid-tones with gamma correction, and finally match the shadow regions. Our algorithm also adjusts based on zones, though we automatically choose different zones to match for each composite, and give statistical evidence for the benefits of using zones.

A completely different approach to making image regions compatible is to adjust their colors so that they match a predefined set of templates that are thought to encode color harmony [Cohen-Or et al. 2006]. However, harmonious images are not necessarily realistic, and the method ignores other factors such as luminance and contrast; finally, this approach has not been quantitatively evaluated.

An alternative to alpha matting is to seamlessly blend image regions with methods like feathering, Laplacian pyramids [Ogden et al. 1985], or gradient-domain compositing [Pérez et al. 2003; Jia et al. 2006; Tao et al. 2010; Sunkavalli et al. 2010]. This approach can work well if the two source images have similar colors and textures; however, in other cases color bleeding and severe discolorations can occur. In practice, most whole-object composites are still made with alpha mattes.

There are a number of related areas that inform our approach. Color constancy is the related problem of recovering the appearance of an input image under neutral lighting [Gijsenij et al. 2011]. Since the problem is highly ill-posed, current solutions are still not robust. Image forensics [Johnson and Farid 2005] seeks to detect composited images, while our goal is to understand and exploit the flexibility of the human visual system, rather than to mislead an algorithm.

### 1.2 Overview

In Section 2 we perform experiments to determine which statistics of natural images are most correlated between foreground and background. Specifically, we take $4126$ natural images with a segmented foreground, and measure the correlation of a number of candidate statistical measures. We find that the means of the high and low zones of the histogram of a statistical measure tend to correlate more strongly than the mean of the whole histogram. For example, shadows and highlights tend to match better between foreground and background than overall mean luminance. We find that luminance, color temperature, saturation, and local contrast are the most correlated statistical measures between foreground and background.

In Section 3 we perform a human subjects experiment to measure the relationship between mismatches in these selected statistical measures and the decrease in perceived realism. Specifically, we take a separate set of 20 images with a matted foreground and introduce variations to the foreground along a specific axis (e.g., luminance); we then measure the decrease in the human perception of realism. One numerical outcome of this experiment is a set of scaling values (Table 2) that linearly relates a change in each statistical measure into a change in the human perception of realism.

In Section 4 we present an algorithm for adjusting a foreground to appear realistic relative to a given background. The algorithm adjusts the foreground so that the means of the statistical measures identified in Section 2 are matched. The main component of the algorithm is a classifier that predicts the zone of the histogram of each statistical measure that, when matched between foreground and background, will produce the most realistic composite. The classifier is trained on the same $4126$ images used Section 2. The classifier also uses the scaling values computed in Section 3 to aid in the prediction. In Section 5, we evaluate the performance of our algorithm and previous work.

## 2 Identifying Key Statistical Measures

Given a foreground $f$ and a background $b$, our goal is to create a composite that is perceived as realistic. In practice the background of a composite is often fixed, so we focus on adjusting the foreground appearance $f$ into $f^*$ so that its composite with $b$ is as realistic as possible. Our first simplifying assumption is that we can achieve a realistic composite by only adjusting standard 2D statistical measures, e.g., luminance, contrast, and color histograms; we ignore issues like semantics and 3D properties. We denote the collection of statistical measures of a foreground or background region as $M_f$ and $M_b$, respectively. We therefore seek to adjust $M_f$ to $M_f^*$.

One approach to this problem is to predict the probability that a composite appears realistic given its foreground and background statistics, i.e., $P(Real|M_f, M_b)$. We could model this probability with a collection of composites, both realistic and unrealistic, along with human ratings of realism. However, the space of composites is very large, and we would need to ensure that all composites are free of issues such as poor matting or semantics, since we are not studying these factors. Another approach is to realize that real-world images are, by definition, realistic; therefore, a composite whose statistics are similar to the statistics of real-world images should also appear realistic [Lotto and Purves 2002; Lalonde and Efros 2007]. We can form a collection of real-world images with roughly segmented foreground and backgrounds, and maximize the likelihood of $f^*$ relative to this data, i.e., maximize $P(M_f|M_b, Real)$.

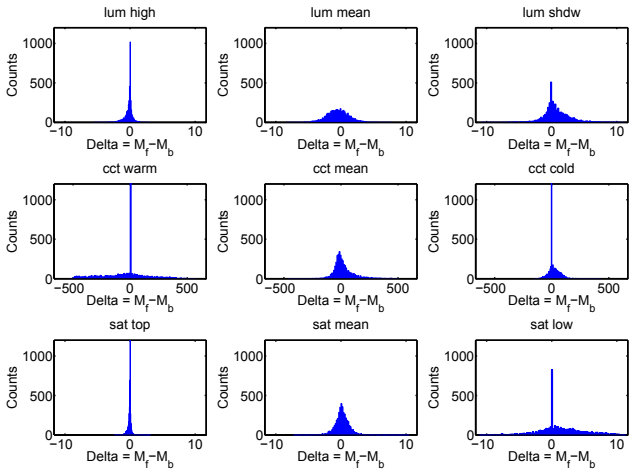


**Figure 2:** *Likelihoods of the offsets for $\mathcal{H}$,$\mathcal{M}$,and $\mathcal{L}$ of luminance, CCT, and saturation, respectively. These statistics have specific meanings for various measures, e.g., $\mathcal{H}$ and $\mathcal{L}$ of CCT correspond to the warmest and coldest colors.*

This likelihood is hard to fully model. Instead, we first identify a few key statistical measures that most affect the realism of a composite. More specifically, we select statistics that satisfy the following criteria:
**1.** The measure should be highly correlated between foreground and background in real images. Therefore we can maximize realism by adjusting the foreground to satisfy this linear relationship.
**2.** The measure should be easy to adjust in a photograph. That is, given a desired value for a measure of the foreground, we should be able to achieve that value with simple image processing operations.
**3.** The measures should be as independent of each other as possible. That way, we can adjust the foreground to satisfy one correlation without negatively affecting another.

We therefore group the set of statistical measures $M$ into five individual categories that are commonly adjusted in composites: luminance, color temperature (CCT), saturation, local contrast, and hue. Within each group, we form a histogram of the property and then compute a number of statistical measures of that histogram, such as the mean, standard deviation, etc. (see Appendix A for a complete list). We denote an individual measure within this overall set as $M^i$. We can measure correlation between $M_f^i$ and $M_b^i$ directly with a Pearson correlation coefficient, which computes the goodness of a linear fit between two measures. Better linear fits are obtained if we first transform the measures into perceptually linear scales (Appendix A).

Another way to discover simple relationships between foreground and background is to look at the offset $\delta^i = M_f^i - M_b^i$ in natural images. If the distribution of $\delta^i$ is concentrated over a small range of values, then the measure is also highly correlated between foreground and background, and described by a simple constant offset. Therefore we look both at the standard deviation of $\delta^i$ as well as the correlation coefficient between $M_f^i$ and $M_b^i$.

We collect 4126 images from the LabelMe data-set [Russell et al. 2008] with clearly labeled, meaningful, and un-occluded foreground objects in front of background scenes. We choose the images to sample a wide variety of compositing scenarios, e.g., outdoors, indoors, daylight, night, people, objects, animals, plants, etc. For each candidate statistic $M^i$, we compute the normalized standard deviation, $\sigma^*$ of $\delta^i$ as $\sigma^* = \sigma/\ell$, where $\sigma$ is the standard deviation of the normalized histogram of $\delta^i$ values across the image collection, and $\ell = \max \|M^i\|$. We also compute the correlation coefficient, $r = corr(M_f^i, M_b^i)$.

| | $\mathcal{H}$ | $\mathcal{M}$ | $\mathcal{L}$ | *std* | *kurt* | *skew* | *entropy* |
|---|---|---|---|---|---|---|---|
| Luminance | | | | | | | |
| $\sigma^*$ | **0.058** | 0.149 | 0.189 | **0.070** | 0.182 | 0.263 | **0.143** |
| $r$ | **0.241** | 0.194 | **0.493** | 0.139 | 0.049 | 0.131 | **0.228** |
| CCT | | | | | | | |
| $\sigma^*$ | 0.287 | **0.145** | **0.071** | **0.105** | 0.188 | 0.246 | 0.225 |
| $r$ | **0.310** | **0.463** | **0.495** | 0.202 | 0.068 | 0.196 | 0.218 |
| Saturation | | | | | | | |
| $\sigma^*$ | **0.027** | **0.095** | 0.299 | **0.058** | 0.205 | 0.173 | 0.159 |
| $r$ | **0.811** | **0.483** | **0.482** | 0.390 | 0.123 | 0.254 | 0.295 |
| Hue | | | | | | | |
| $\sigma^*$ | n/a | **0.348** | n/a | **0.413** | 0.492 | 0.474 | **0.158** |
| $r$ | n/a | **0.235** | n/a | **0.366** | **0.175** | 0.001 | 0.172 |
| Local Contrast | | | | | | | |
| $\sigma^*$ | 0.150 | **0.016** | **0.000** | **0.012** | 0.170 | 0.152 | 0.137 |
| $r$ | **0.719** | **0.713** | 0.403 | **0.760** | 0.192 | 0.235 | 0.681 |

**Table 1:** *Statistical measures computed across our image collection. Smaller values of $\sigma^*$ (normalized standard deviation of $\delta^i$) and larger values of $r$ (correlation coefficient between foreground and background) indicate more useful measures. For every row, the three best quantities are highlighted in bold.*

Instead of only computing these measures over the entire histogram of a category such as luminance, we found stronger correlations if we separated the histograms of luminance into high ($\mathcal{H}$), middle (whole-histogram) ($\mathcal{M}$), and low ($\mathcal{L}$) zones. The average of each zone is used as an individual statistical measure (Appendix A). For example, in real images the mean luminance of highlight regions tend to match better between foreground and background than the mean of the whole histogram, which can be seen in the first row of Figure 2; we show the likelihoods of the offsets for two other measures in the other rows. The likelihoods of the offsets for other statistics are given in supplemental materials. We also show numbers for $\sigma^*$ and $r$ across zones in Table 1. (An exception is hue, which is a circular value (Appendix A) so $\mathcal{H}$ and $\mathcal{L}$ are undefined. )

We can draw several conclusions from these results. Luminance matches better between foreground and background in the highlights and shadows than in the mid-range. Local contrast and saturation show very strong correlation. While CCT and hue are related methods of describing color, CCT shows stronger correlation than hue. Last, all the likelihoods of offsets have single-peak distributions centered at 0. This observation statistically supports the intuition behind the mean-matching approach of color transfer techniques [Reinhard et al. 2004]. However, the correlations of the high ($\mathcal{H}$) and low ($\mathcal{L}$) zones are much stronger than the for the whole histogram ($\mathcal{M}$) typically used by color transfer, which suggests that matching by zones might lead to more realistic composites. Finally, on average the means of the histogram zones $\mathcal{H}$,$\mathcal{M}$,and $\mathcal{L}$ show stronger correlation than the other statistical measures. Though the standard deviation sometimes performs well, the zone means are enough to cover the best or nearly-best values in Table 1.

Based on these conclusions, we focus on mean-matching across different zones of the histogram for luminance, color temperature, saturation, and local contrast. In the next section, we test the impact of mismatches in these statistical measures on human judgements of realism.

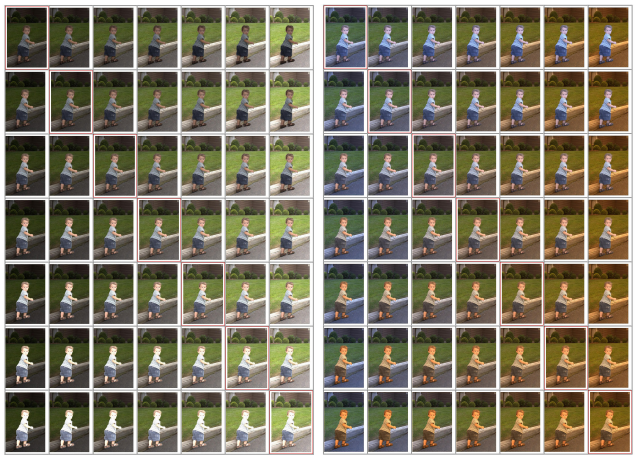

**Figure 3:** *Stimuli to study impact of variations in key statistical measures. We show the matrices of composites after manipulating luminance (left) and CCT (right) for a specific image; more examples can be found in the supplemental materials.*

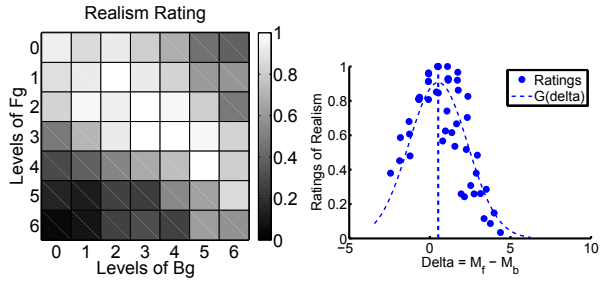


**Figure 4:** *Left: human ratings corresponding to the adjusted luminance matrix in Figure 3. Right: these ratings are fit by a Gaussian function $G(\delta^i)$, where each data point corresponds to one cell in the matrix on the left.*

## 3 Impact on Human Realism Ratings

We have identified several key statistical measures that are strongly matched between foreground and background in real images. How do mismatches in these key statistical measures affects the judgement of realism? The fact that natural images follow a certain statistical relation does not necessarily tell us the acceptable range of variation in that statistical relation before a composite appears unreal. Also, a numerical relationship between each statistical measure and its influence on human ratings of realism will help us combine these different measures into a single algorithm for automatic compositing. We therefore perform a perception experiment with human subjects on Amazon Mechanical Turk (MTurk) to numerically model this relationship.

To design our experiment it is important to have a natural control image, and only vary one variable; otherwise, issues other than the one we are studying could influence the perception of realism. We therefore take 20 natural images from a public database [Bychkovsky et al. 2011], carefully matte out the foreground, adjust the foreground and/or background along a single axis, and recomposite. In this way, we can be assured issues like semantics are already satisfied. We perform this experiment for three key statistical measures identified in the previous section: luminance, color temperature, and saturation. For these measures, we can simply increase or decrease the offset of this measure between foreground and background by shifting histograms. We then measure the perceived decrease in realism. (We skip the key statistical measure of local contrast, for now, because it is not adjustable with simple linear operations; we address this special case in Section 4.)

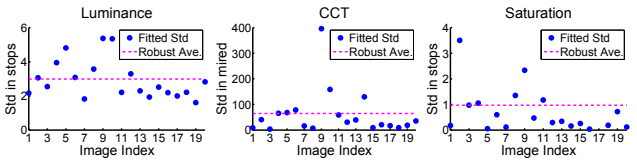


**Figure 5:** *The fitted standard deviations of all images in the stimuli of luminance, CCT, and saturation.*

| | Luminance | CCT | Saturation |
|---|---|---|---|
| Units | *stops* | *mired* | *stops* |
| $\sigma_g$ | 2.99 | 64.9 | 0.97 |

**Table 2:** *Robust average standard deviations of the fitted rating functions for luminance, CCT, and saturation.*

We generate a $7 \times 7$ matrix of compositing results by adjusting the foreground and/or background by a specific amount (two examples are shown in Figure 3). The center of the matrix is the original image, and along the diagonal foreground and background are adjusted equally. We then acquire a realism score $(0.0 \sim 1.0)$ for each composite (Appendix B).

A typical example matrix of human ratings is given on the left in Figure 4. First, as expected, the diagonal along which the foreground and background are adjusted equally is rated the most realistic. Second, realism ratings decrease smoothly away from the diagonal. Third, except for the extreme corners of the matrix, this decrease can be modeled reasonably well as a function of the offset $\mathcal{M}_f^i - \mathcal{M}_b^i$ (where $\mathcal{M}^i$ is the mean of the histogram for the $i$'th stimuli image). We therefore fit a rating function $R(\mathcal{M}_f^i, \mathcal{M}_b^i)$ as a Gaussian function $G$:

$$R(\mathcal{M}_f^i, \mathcal{M}_b^i) \approx G(\mathcal{M}_f^i - \mathcal{M}_b^i) = G(\delta^i) \qquad (1)$$

An example of this fit is shown on the right in Figure 4. See supplemental materials for fits to other statistical measures and images. However, we note that the fitted Gaussian functions for different images generally have different means and standard deviations. (For example, images with harsher or more dramatic lighting have much larger standard deviations.) We therefore fit this Gaussian separately per matrix.

The standard deviations $\sigma$ of fitted Gaussian functions for all 20 images corresponding to luminance, CCT, saturation are shown in Figure 5. We can see the variation in $\sigma$. While it may be possible to predict the shape of this Gaussian for an image given features computed from it, we would need many more than 20 training images to do so, and the need for very accurate mattes makes creating additional training data expensive.

Nonetheless, we can use these fitted functions in two ways. First, we can get a sense of how human realism ratings respond to variations in these measures. To visualize this impact, we choose an example with close-to-median standard deviation, and show the original image as well an adjusted version whose realism rating is reduced by $40\%$ (Figure 6). Second, we can compute robust average standard deviations for each statistical measure that approximately places each measure on an equal, linear scale. That is, after applying our computed scale factors we can expect equal adjustments of different statistical measures to produce equal decreases in realism rating (modulo the error introduced by using an average). We create

**Figure 6:** *Left to right: the original image, the luminance-adjusted, the CCT-adjusted, and the saturation-adjusted. In our experiment the original image received the highest realism rating, while the other three received realism ratings decreased to around* 60%.

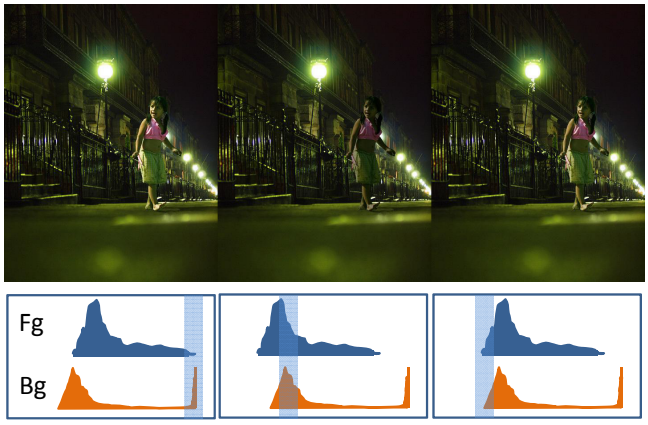


**Figure 7:** *Top: results of matching highlight, mid-tone, and shadow zones of the luminance histogram for a composite. Bottom: illustration of matching the corresponding zones of the histograms.*

these scale factors $\sigma_g$ as robust averages of the standard deviations in Figure 5, after removing the highest and lowest values. These values are given in Table 2, and are used in the next section as part of our algorithm to automatically improve the realism of a composite.

## 4 Automatic Composite Adjustment

We now present our algorithm for automatically adjusting a foreground region relative to a proposed background so that their composite appears more realistic. Our technique uses machine learning to automatically choose a zone of the histogram to match between foreground and background for luminance, CCT, and saturation. Once a zone is chosen, we shift the foreground's histogram so that the mean of the zone matches the mean of the background's zone. We adjust local contrast using a separate technique, since contrast cannot be manipulated with simple histogram shifts.

### 4.1 Using Zones

Given the evidence in Section 2, the most straightforward approach to aligning the statistics of luminance, CCT, and saturation is to select the zone of each measure's histogram with the lowest standard deviation ($\sigma^*$in Table 1). Given that zone, we could shift the histogram of the foreground so that the mean of that zone matches the mean of the zone in the background (Figure 7). This approach would select the same zones for each composite, and would be very similar to color transfer techniques [Reinhard et al. 2001] except that instead of using the entire histogram, we would use a zone.

We can evaluate this approach using the 4126 segmented images

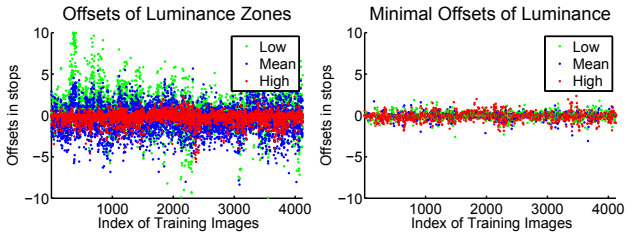


**Figure 8:** *Left: the offsets of different zones of the luminance histograms per image. Right: the minimal offset (created by selecting the minimum of low, mean, high) of the luminance histograms. See supplemental materials for the offsets of CCT and saturation.*

| | $\mathcal{H}$ | $\mathcal{M}$ | $\mathcal{L}$ | Best | Predicted |
|---|---|---|---|---|---|
| Luminance | | | | | |
| *stops* | 0.405 | 1.411 | 1.540 | 0.230 | 0.372 |
| CCT | | | | | |
| *mired* | 129.5 | 64.4 | 32.3 | 8.53 | 17.21 |
| Saturation | | | | | |
| *stop* | 0.168 | 0.811 | 2.581 | 0.096 | 0.120 |

**Table 3:** *The MAE errors for five compositing methods. Five columns show the results by matching using* high, mean, low *zones, the best zone (oracle algorithm), and our multi-label classifier.*

from Section 2 and an error metric. We denote the set of zone regions as $z$ (where $z \in \{\mathcal{H}, \mathcal{M}, \mathcal{L}\}$), and given a statistical measure $M$, we denote $M(k, z)$ as the mean value of that measure on the $k$'th image in our collection in the zone $z$. We use the mean absolute error (MAE) in this statistical measure that is incurred by selecting a specific zone to match, or $E(z) = \frac{1}{n}\sum_{k=1}^{n} |M_f(k, z) - M_b(k, z)|$, where $n$ is the number of images in our collection. The errors of these schemes are shown in the first three columns of Table 3. A low error indicates that matching the means between foreground and background using that statistical measure is better able to reproduce the original images of our collection. However, the error will never be exactly zero, since the offsets of natural images are not exactly zero (i.e., there are always tails in the distributions in Figure 2). The error is lowest if we use the $\mathcal{H}$ portion of the luminance histogram, the $\mathcal{L}$ portion of the CCT histogram, and the $\mathcal{H}$ portion of the saturation histogram. Note that using the entire histogram ($\mathcal{M}$) (i.e., not using zones) is not optimal for any measure, which supports our claim that using zones is more effective.

This algorithm ignores the particular characteristics of a composite in selecting zones; could we do better by selecting a zone dynamically per composite? This idea is motivated by the plots in Figure 8. The left plot shows that the offset between background and foreground for a zone of the luminance histogram for a particular image may not be very close to zero. However, the right plot shows that typically at least one of the $\mathcal{H}, \mathcal{M}, \mathcal{L}$ zones does have a near-zero offset. What if we could know which zone of the histogram is likely to match best between foreground and background for a composite? The fourth column of Table 3 shows the error of a hypothetical "oracle" algorithm always predicting the best zone that produces the smallest offset. Note that this error is still not zero since the smallest offset is still not exactly zero for each test image. Nonetheless, the performance is much better than always using the same zone. We therefore attempt to automatically predict the best zone with machine learning.

### 4.2 Selecting Zones

One approach to selecting zones to use for a composite is to train a classifier to choose one of three options $\{\mathcal{H}, \mathcal{M}, \mathcal{L}\}$ per composite for each statistical measure. However, there may be more than one zone with a near-zero offset, so treating such a zone as unsuitable for matching simply because another zone performs slightly better might confuse a classifier. Instead, we take a multi-label classification approach [Tsoumakas and Katakis 2007], where more than one label can be applied to an instance.

Specifically, we train three separate binary classifiers per statistical measure (one per zone), where each classifier predicts whether the zones $\{\mathcal{H}, \mathcal{M}, \mathcal{L}\}$ are individually suitable for composite matching. To form binary training data from our collection of 4126 images from Section 2, we compare the offset between foreground and background to a realism threshold $T$ for each zone $z$ of luminance, CCT, and saturation. If the offset of the training image is below this threshold we assign label 1, since matching using this measure would produce a realistic image similar to the original, and 0 otherwise. To compute threshold $T$ we use a scaled version of the robust standard deviations computed in Table 2, namely $T = s * \sigma_g$, with $s = 0.1$ chosen by several iterations of cross-validation using a small-scale trial version of our MTurk evaluation study described in Section 5. Note that we use the same $T$ value for each zone.

What features should we use for our classifier? Our expectation is that the shape of the histogram for a feature is correlated with which zone might match well. For example, a luminance histogram with significant regions of highlight or shadow may match better with the $\mathcal{H}$ or $\mathcal{L}$ zones, respectively. We therefore use as features statistical properties of the histogram of that measure for both the foreground and background of the image. For example, for luminance we use $std$, $skew$, $kurt$, $entropy$, $p_1$, $p_2$, ..., $p_{20}$, where $p_j$, $j = 1, 2, .., 20$, is the portion of the $j$th bin in the luminance histogram. We separately compute these features on the foreground and background histograms.

We use a random forest classifier [Liaw and Wiener 2002] with default settings. We also tried SVM [Chang and Lin 2011] with an RBF kernel, but found it worked slightly less well. The misclassification rates (as percentages, computed using 10-fold cross-validation on the training data) of our classifiers are shown in Table 4.

| | $\mathcal{H}$ | $\mathcal{M}$ | $\mathcal{L}$ |
|---|---|---|---|
| Luminance | 20.9% | 18.4% | 22.2% |
| CCT | 6.5% | 20.9% | 21.2% |
| Saturation | 14.8% | 14.1% | 3.4% |

**Table 4:** *Error rates of every two-label classifier for luminance, CCT, and saturation.*

Given the output of three zone classifiers for a specific statistical measure, we finally combine them into a single choice to match to produce a composite. This combination is easy if only one of the three classifiers returns 1, but what if multiple zones are suitable for matching? Our principle is to choose the zone which will cause the least change to the foreground, since large changes lead to more noticeable artifacts (e.g., clipping). We consider a zone a "candidate" if the output of its classifier is 1. If there is more than one candidate zone, we consider each candidate zone $z_i$ in turn, and sum the absolute mean shift of the histogram for all candidate zones that is induced by matching $z_i$. Note that any clipping that may occur at the right and left of the histogram is taken into account during this computation. Finally we select the $z_i$ with the minimum sum. If several $z_i$ have the same sum, we select the one with the minimum input offset. If no zones are candidates, we simply select $\mathcal{M}$.

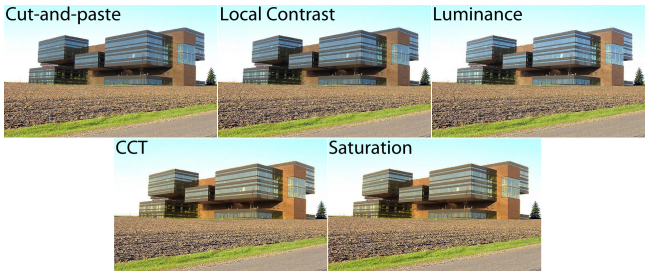


**Figure 9:** *The sequence of adjustments for a particular composite.*

The result of the above technique is the choice of a single zone per composite. We show the error rate of our technique in the fifth column of Table 3, computed with 10-fold cross-validation; it is lower than always choosing the same zone, but not as low as the perfect "oracle" selector.

### 4.3 Our Pipeline

Using the selected zone from the previous section, we adjust each key statistical measure in turn. Contrast is handled as a special case (Appendix C), since it cannot be adjusted with simple histogram shifts. Instead, inspired by Paris et al. [2011] we use an $S$-shaped curve to adjust local contrast. We select the $\mathcal{H}$ zone of the contrast histogram, and choose the shape of the $S$-curve to best match local contrast as described in Appendix C. Note that contrast and luminance are not independent adjustments; however, our contrast adjustment technique is designed to not move the mean of the luminance histogram. After contrast, we adjust luminance, CCT, and saturation, in turn. In our experiments we found other orders generally work as well, as long as contrast adjustment is performed first. Our adjustment algorithm is greedy; alternatively, it could iterate several times over this sequence of steps. However, we found the results did not change significantly after the first iteration. Figure 9 shows the intermediate results of the steps of our pipeline.

**Input:** $f$ and $b$
**1.** Adjust local contrast $M_f^0$ using $S$-shape correction.
**2.** Adjust luminance $M_f^1$.
Select zone $z$ that is best for matching.
Shift $M_f^1$, so that $M_f^1(z) - M_b^1(z) = 0$.
**3.** Adjust CCT $M_f^2$.
Select zone $z$ that is best for matching.
Shift $M_f^2$, so that $M_f^2(z) - M_b^2(z) = 0$.
**4.** Adjust saturation $M_f^3$.
Select zone $z$ that is best for matching.
Shift $M_f^3$, so that $M_f^3(z) - M_b^3(z) = 0$.
**Output:** adjusted $f^*$ and $b$.

## 5 Results and Evaluation

We show a number of compositing results adjusted using our automatic method, and compare them to other techniques. Table 3 shows one way to evaluate the success of our method on images that we already know are real; however, a more useful evaluation involves the adjustment of composites with regions taken from separate image sources. We therefore created a test set of 48 composites, and made an effort to ensure that each composite is semantically reasonable. We then created adjusted results using five techniques: simple cut-and-paste, a manually adjusted composite, Photoshop Match Color, the method of Lalonde and Efros [2007] (which we label ColorComp), and our method. Six examples are

*t*-tests at significance level 0.05

| | | | | |
|---|---|---|---|---|
| **Ours** | Cut-and-paste (0.00) | MatchColor (0.00) | ColorComp (0.01) | Manual (0.79) |
| **Manual** | Cut-and-paste (0.03) | MatchColor (0.00) | ColorComp (0.00) | Ours (0.20) |
| **ColorComp** | MatchColor (0.00) | Cut-and-paste (0.87) | Ours (0.99) | Manual (0.99) |
| **Cut-and-paste** | MatchColor (0.00) | ColorComp (0.13) | Ours (0.99) | Manual (0.99) |
| **MatchColor** | Cut-and-paste (1.00) | ColorComp (1.00) | Ours (1.00) | Manual (1.00) |

**Figure 10:** *Paired t-test results to compare all methods. For each method (every row), we list the other methods and highlight in gray those that were outperformed at a statistically-significant frequency. We also give all p-values.*

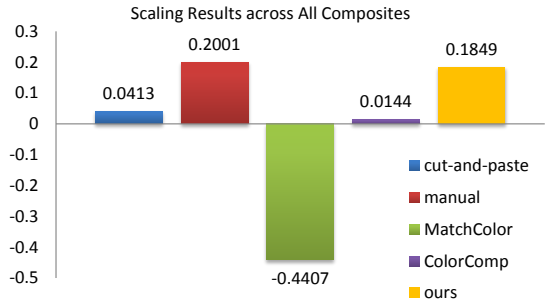


**Figure 12:** *Comparison of all methods by scaling results across all composites. Higher scores are better.*

shown in Figure 11. The manually-adjusted composite was created in Photoshop by one of the authors who has extensive Photoshop experience, and typically took 3-4 minutes to create. We computed the results of ColorComp using code and data provided by its authors, which typically took 3-5 minutes to execute per example. Photoshop Match Color uses a standard color transfer technique similar to Reinhard et al. [2004]; color transfer is useful for a number of creative tasks, and compositing is not necessarily its main application. Our method takes 5-15 seconds to execute using unoptimized Matlab code; the bottleneck is the contrast adjustment step. The results of all these techniques on all 48 composites are given in supplemental materials.

To evaluate the relative realism of these five techniques, we performed an experiment using Mechanical Turk. To simplify the task, we used a forced choice test between two alternate versions of the same composite, where the subject was asked to choose the most realistic alternative. (An alternate methodology we considered was to collect individual realism ratings for each composite. However, those ratings would be more sensitive to factors that influence realism that we are not studying, such as the semantic likelihood of the depicted scene.) The methodology of our experiment is described in Appendix B; we collected, on average, 12.3 human choices for each of the 480 possible comparisons.

First, we use one-tailed $t$-tests to compare each method against each other method. We show which methods are better than others with significance level $p < 0.05$ in Figure 10. Our method outperforms all other automatic methods, and its performance is not significantly different than manual adjustment. Second, we convert the series of paired comparisons into scaling results that place the performance of each method on a single scale; this scaling can be performed individually for each composite (Figure 11, inset in each row, as well as supplemental materials), and averaged over all composites (Figure 12). We use Thurstone's Law of Comparative Judgments, Case V [David 1988], which assumes that each method has a single quality score, and observer estimates of this score are normally distributed. The resultant scale values are linear multiples of this score, so that differences between scale values are in the units of standard deviation of preference. Higher values are better. Figure 12 shows that our method performs slightly worse than manual adjustment, but much better than all other automatic methods. ColorComp significantly outperforms color transfer; however, surprisingly it does not outperform cut-and-paste. This may be because many of our scenes have natural lighting, and thus do not require large adjustments to appear natural.

The scales for individual composites (the insets in Figure 11, and supplemental materials) can diverge quite significantly from the average. For example, in the fifth row of Figure 11, ColorComp performs best. However, the fourth row shows a typical failure of color transfer (which is also a component of ColorComp), where the green color of the forest is unnaturally transferred to the person. In the last row, our method significantly outperforms the other automatic techniques. However, it also shows a limitation of our method: we cannot correct the harsh lighting of the foreground that still makes this composite unrealistic.

### 5.1 Limitations

Along with the examples in Figure 11, the supplemental materials show additional failure cases. There are several sources of error. First, our classifier sometimes chooses the wrong offset to match. Second, even the minimal offsets between foreground and background for real images are not exactly zero, so perfect classification will still not yield perfect adjustments. Even the most peaked distributions in Figure 2 have tails, so any algorithm that uses mean-matching will have errors. Third, we do not use spatial or proximity cues, even though areas of the background close the foreground are probably more relevant than areas farther away. For example, lighting can change with respect to depth in the scene or proximity to a light source. Fourth, we use hard thresholds to decide if a zone is appropriate for matching during the training phase; near-threshold values can cause incorrect decisions.

Finally, a natural question is whether our method could be used for the main problem addressed by Lalonde and Efros [2007]: predicting whether a chosen foreground and background will appear realistic together. Unfortunately, the realism ratings we collect in Section 3 measure human response to variations along a single axis (e.g., luminance); it is unclear how to combine simultaneous variations along multiple axes into a single realism prediction.

## 6 Conclusion

In this paper we studied the problem of adjusting a composite to appear realistic; our automatic technique significantly outperforms previous methods. The biggest limitation of our method is that we limit our scope to standard 2D image processing adjustments; some composites will need more specific or complicated adjustments such as relighting to truly appear realistic.

Finally, while we have identified image statistics that are correlated with composite realism, there is still much to be done to truly understand the factors that influence human perception of realism. Why are these statistics more correlated with realism than others, and is the relationship causation or correlation? Also, our zone selection classifier is a black box; how does it determine which zone is best

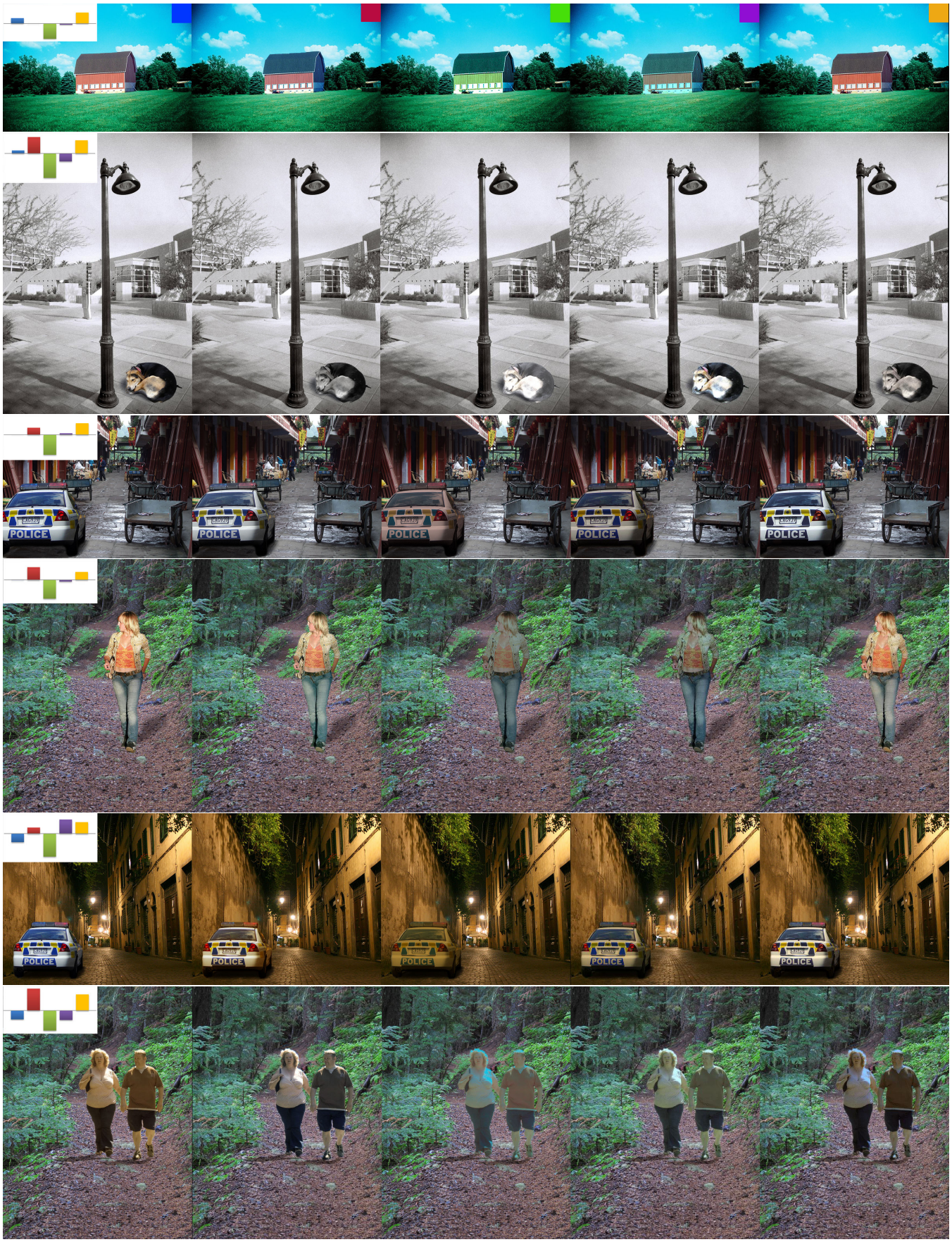


**Figure 11:** *Composites adjusted by: cut-and-paste, manual, Match Color, ColorComp, and ours. Insets show their relative scores from the human subjects study.*

for matching? Finally, while we have evaluated our results with human subjects, would expert compositors have different rankings of methods? We will explore these questions in future work.

## Acknowledgements

We thank Aaron Hertzmann, Peter O'Donovan, Dan Goldman, and Sylvain Paris for helpful discussions. This work was supported in part by Adobe. The images in the results are courtesy of Flickr users *joeldinda*, *Limbic*, *EssjayNZ*, *greekadman*, *Dell's Pics* et al. under Creative Commons license.

## References

ALEXANDER, T., 2011. Visual effects supervisor at Industry Light & Magic. Rules of thumb in image compositing. Personal communication, Oct.

BERENS, P. 2009. Circstat: a matlab toolbox for circular statistics. *Journal of Statistical Software 31*, 10, 1–21.

BYCHKOVSKY, V., PARIS, S., CHAN, E., AND DURAND, F. 2011. Learning photographic global tonal adjustment with a database of input/output image pairs. In *Proceedings of CVPR*, 97–104.

CHANG, C.-C., AND LIN, C.-J. 2011. LIBSVM: A library for support vector machines. *ACM Trans. on Intelligent Systems and Technology 2*, 27:1–27:27. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

COHEN-OR, D., SORKINE, O., GAL, R., LEYVAND, T., AND XU, Y.-Q. 2006. Color harmonization. *ACM Trans. Graph. 25* (July), 624–630.

DAVID, H. A. 1988. *The Method of Paired Comparisons*. Oxford University Press, 2nd edition.

GIJSENIJ, A., GEVERS, T., AND VAN DE WEIJER, J. 2011. Computational color constancy: Survey and experiments. *IEEE Trans. on Image Processing 20*, 9 (Sep), 2475 –2489.

JIA, J., SUN, J., TANG, C.-K., AND SHUM, H.-Y. 2006. Drag-and-drop pasting. *ACM Trans. on Graphics 25*, 3 (July), 631–637.

JOHNSON, M. K., AND FARID, H. 2005. Exposing digital forgeries by detecting inconsistencies in lighting. In *Proceedings of the 7th Workshop on Multimedia and Security*, 1–10.

LALONDE, J.-F., AND EFROS, A. 2007. Using color compatibility for assessing image realism. In *IEEE 11th International Conference on Computer Vision*, 1–8.

LIAW, A., AND WIENER, M. 2002. Classification and regression by randomforest. *R News 2*, 3, 18–22.

LOPEZ-MORENO, J., SUNDSTEDT, V., SANGORRIN, F., AND GUTIERREZ, D. 2010. Measuring the perception of light inconsistencies. In *Proceedings of APGV*, ACM, 25–32.

LOTTO, R., AND PURVES, D. 2002. The empirical basis of color perception. *Consciousness and Cognition 11*, 4 (Dec.), 609–629.

OGDEN, J. M., ADELSON, E. H., BERGEN, J., AND BURT, P. 1985. Pyramid-based computer graphics. *RCA Engineer 30*, 5, 4–15.

OHTA, N., AND ROBERTSON, A. R. 2005. *Colorimetry: Fundamentals and Applications*. Wiley, Chichester.

OSTROVSKY, Y., CAVANAGH, P., AND SINHA, P. 2005. Perceiving illumination inconsistencies in scenes. *Perception 34*, 11, 1301–1314.

PARIS, S., HASINOFF, S. W., AND KAUTZ, J. 2011. Local Laplacian filters: edge-aware image processing with a Laplacian pyramid. *ACM Trans. on Graphics 30* (Aug), 68:1–68:12.

PELI, E. 1990. Contrast in complex images. *Journal of Optical Society of America 7*, 10 (Oct), 2032–2040.

PÉREZ, P., GANGNET, M., AND BLAKE, A. 2003. Poisson image editing. *ACM Trans. on Graphics 22* (Jul), 313–318.

POULI, T., AND REINHARD, E. 2010. Progressive histogram reshaping for creative color transfer and tone reproduction. In *Proceedings of NPAR*, 81–90.

REINHARD, E., ASHIKHMIN, M., GOOCH, B., AND SHIRLEY, P. S. 2001. Color transfer between images. *IEEE Computer Graphics & Applications 21*, 5 (Sept./Oct.), 34–41.

REINHARD, E., AKÜYZ, A., COLBERT, M., HUGHES, C. E., AND OCONNOR, M. 2004. Real-time color blending of rendered and captured video. In *Interservice/Industry Training, Simulation and Education Conference*, 1–9.

RHEMANN, C., ROTHER, C., WANG, J., GELAUTZ, M., KOHLI, P., AND ROTT, P. 2009. A perceptually motivated online benchmark for image matting. In *Proceedings of CVPR*, 1826–1833.

RUSSELL, B., TORRALBA, A., MURPHY, K., AND FREEMAN, W. 2008. LabelMe: A Database and Web-Based Tool for Image Annotation. *International Journal of Computer Vision 77*, 1 (May), 157–173.

SMITH, A. R., AND BLINN, J. F. 1996. Blue screen matting. In *Proceedings of SIGGRAPH 96*, 259–268.

STOKES, M., ANDERSON, M., CHANDRASEKAR, S., AND MOTTA, R. 1996. A standard default color space for the internet-srgb. *Microsoft and Hewlett-Packard Joint Report*.

SUNKAVALLI, K., JOHNSON, M. K., MATUSIK, W., AND PFISTER, H. 2010. Multi-scale image harmonization. *ACM Trans. on Graphics 29*, 4 (July), 125:1–125:10.

TAO, M. W., JOHNSON, M. K., AND PARIS, S. 2010. Error-tolerant image compositing. In *European Conference on Computer Vision*, 31–44.

TSOUMAKAS, G., AND KATAKIS, I. 2007. Multi-label classification: An overview. *International Journal of Data Warehousing and Mining 3* (July/Sept.), 1–13.

WAGBERG, J., 2007. Optpop - a color properties toolbox, Mar. Software available at `http://www.mathworks.com/matlabcentral/fileexchange/13788`.

WANG, J., AND COHEN, M. F. 2007. Image and video matting: a survey. *Found. Trends. Comput. Graph. Vis. 3* (January), 97–175.

## Appendix A. Image Statistical Measures

### Image Statistics

To compute image statistics on an input image or composite in sRGB color space, the pixels are first inversely Gamma corrected [Stokes et al. 1996]. We then transform image statistics so that they are approximately linear to human visual perception. Luminance and saturation are converted into log domain based on Weber's law, and CCT is defined by mired [Ohta and Robertson 2005].

**Luminance**: we use $\log_2 Y$, where $Y$ (normalized to $[\epsilon, 1.0]$) is the luminance channel of $xyY$ space, where $\epsilon = 3.03 \times 10^{-4}$ (corresponding to intensity 1 in a 0-255 greyscale image before inverse Gamma correction) is used to avoid undefined log values. The unit of difference in $\log_2$ domain is a *stop*.

**Correlated Color Temperature (CCT)**: we use "mired" as the unit of CCT. 1 mired = $10^6/K$, where $K$ is the Plankian color temperature in Kelvin, clipped in $[1500, 20000]$ that is the normal range of natural lighting. CCT is computed using the package OptProp [Wagberg 2007].

**Saturation**: we use $\log_2 S$, where $S \in [\epsilon, 1.0]$ is the saturation channel of $HSV$ space.

**Hue**: we use $H$, where $H$ is a circular value in $[0.0, 1.0]$ (or $[0°, 360°]$), the hue channel of $HSV$ space.

**Local contrast**: Locally defined Weber contrast [Peli 1990], $c_x$, is used. For every pixel $x$, $c_x = L_x/\overline{L_x}$, where $L_x$ is the pixel luminance, $\overline{L_x}$ is the local average luminance, which is the output of a Gaussian filter with $\sigma = 1.5$, filter radius $r = 3$.

### Histogram Statistics

For each statistical measure $M^i$, the statistics are computed across all pixels in corresponding regions, i.e., foreground or background. For image properties sensitive to color bias, e.g., CCT, hue, and saturation, they are only computed across pixels that are neither over-nor under-exposed. In practice, we use the pixels with $0.013 \leq Y \leq 0.88$, where two thresholds correspond to intensity 60 and 240 in a 0-255 greyscale image before inverse Gamma correction.

The statistics $\mathcal{H}, \mathcal{M}, \mathcal{L}$ are defined as $\mathcal{H} = \overline{M^i > M^i_{99.9\%}}$, $\mathcal{M} = \overline{M^i}$, $\mathcal{L} = \overline{M^i < M^i_{0.1\%}}$, where $\overline{M^i > M^i_{99.9\%}}$ represents the mean of pixel luminance which are greater than the $99.9\%$ quantile. For $kurtosis$, we use the definition by which normal distributions have kurtosis of 0. $Entropy$ is computed by first scaling $M^i$ into $[0, 255]$, and using the computation routine of standard greyscale image entropies (e.g., function $entropy$ in Matlab). For other statistics, standard definitions are used. Notably, circular statistics [Berens 2009] are used for *hue*. For example, circular mean of hue is used in place of ordinary mean.

### Alpha Mattes

We assume that every composite has an alpha matte for the foreground object. When computing $M_f$, we morphologically erode the alpha matte to avoid inaccuracy of matte boundary, and then compute $M_f$ in the region where matte values are greater than 0.5. When computing $M_b$, the original alpha matte is morphologically dilated, and then $M_b$ is computed in the region where matte values are lower than 0.5. To avoid distant background, $M_b$ is only computed within an area equal to the bounding box of the foreground scaled by 3.

The elements for erosion and dilation operations are disks, whose radius for erosion is $r_e = 0.03 \times \min(w, h)$ and radius for dilation is $r_d = 0.15 \times \min(w, h)$, where $w, h$ are the width and height of the bounding box of the foreground object.

## Appendix B. Perceptual Experiments

### Impact on Human Realism Ratings

Twenty natural images are used as input, with manually created foreground object alpha mattes. For *luminance*, the foreground and background of every image are respectively manipulated to 7 levels via shifting ($\pm 3$ steps by 0.5 stop), which are then composited to form $7^2 = 49$ composites. In total, we generated $20 \times 7^2 = 980$ composites for MTurk workers to evaluate. The same number of stimuli are generated for CCT and saturation by similar procedures. The step in manipulating CCT is 40 mired, and the step of saturation is 0.25 stop in $\log_2$ domain. If the image properties of some pixels exceed the defined range in Appendix A after manipulation, clipping is performed.

Every MTurk worker is presented with a series (23) of composites to evaluate, with a two-alternative forced choice, "manipulated" or "real". Instructions and examples are given. The time for each evaluation is limited to 12 seconds. We present 3 out of 23 evaluations as test cases with very obvious status of "real" or "manipulated". Answers that fail to pass at least two of the three test questions are classified as *random answers*, and discarded in analysis. In the end, there are 1360 valid responses for luminance, 969 for CCT, and 1048 for saturation. Every composite is evaluated 15+ times. The rating for a composite is the proportion of answers of "real".

We control each specific MTurk task so that the 23 composites (20 for actual study, 3 for test) presented to a worker are all from different natural images. Any one worker is prohibited from participating in more than one experiments in 24 hours, to avoid seeing the same foreground or background twice in succession.

### Evaluating Composites

We use all 48 composites as the stimuli, for which we compare five alpha-matte-based adjusting methods. All results are shown in the supplemental materials. In the user study, we use the scheme of forced two-alternative choice (FAC), where every participant is requested to compare a series of pairs of results and pick a more realistic one (12 pairs for actual comparison and 4 pairs for test). The order of comparisons is randomized. The 4 test examples pair one obviously real image with an obviously fake composite. The answers that fail to correctly classify at least three out of four test comparisons are regarded as random answers. 653 workers responded to our test and 151 random answers are discarded. Every pair of comparison is evaluated by 10+ (by average 12.3) workers for statistical robustness.

## Appendix C. Adjust Local Contrast

We increase local contrast by a global pixel-wise luminance transformation using an $S$ shape curve; the inverse $S$ curve decreases local contrast, as shown in the inset. The input luminance value $L_{in}$ is re-mapped to a new output value $L_{out}$ along the curve; the same curve is used across the image. In this transformation, luminance is defined by $Y$ $(0 \sim 1)$ of $xyY$ space (not in log domain). The turning point $p_m$ of $S$ curve is the average luminance, by which the curve is divided into an upper and a lower sub-curve. Each sub-curve is a Bezier curve with three anchors. The upper curve is controlled by $p_m, p_U, p_1$, and the lower curve by $p_m, p_L, p_0$. The degree of transformation is controlled by $p_U$ and $p_L$, where $p_U = p_{11} + \alpha(p_{12} - p_{11})$ and $p_L = p_{01} + \alpha(p_{02} - p_{01})$. If $\alpha < 0.5$, it is an $S$ curve that increases local contrast; if $\alpha > 0.5$, it is an inverse $S$ curve that decreases local contrast. If $\alpha = 0.5$, it degrades to a straight line. In practice, we search $\alpha \in [0.4, 0.6]$ to find the best curve that matches $\mathcal{H}$ of local contrasts between foreground and background.