Appears in Proc. IEEE Int. Conference on Computer Vision (ICCV) 2009

Image Restoration using Online Photo Collections

Kevin Dale[1] Micah K. Johnson[2] Kalyan Sunkavalli[1] Wojciech Matusik[3] Hanspeter Pfister[1]

[1]Harvard University
{kdale,kalyans,pfister}@seas.harvard.edu

[2]MIT
kimo@mit.edu

[3]Adobe Systems, Inc.
wmatusik@adobe.com

Abstract

We present an image restoration method that leverages a large database of images gathered from the web. Given an input image, we execute an efficient visual search to find the closest images in the database; these images define the input's visual context. We use the visual context as an image-specific prior and show its value in a variety of image restoration operations, including white balance correction, exposure correction, and contrast enhancement. We evaluate our approach using a database of 1 million images downloaded from Flickr and demonstrate the effect of database size on performance. Our results show that priors based on the visual context consistently out-perform generic or even domain-specific priors for these operations.

1. Introduction

While advances in digital photography have made it easier for everyone to take pictures, it is still difficult to capture high-quality photographs in some settings. A skilled photographer knows when to trust a camera's automatic mechanisms, such as white balance and exposure metering, but an average user typically leaves the camera in fully automatic mode and accepts whatever picture the camera chooses to take. As a result, people often have many images with defects such as color imbalance, poor exposure, or low contrast. Image restoration operations can lessen these artifacts, but automatically applying these operations can be challenging.

The primary difficulty in automatic restorations is determining the appropriate parameters for a specific image. Typically, the problem is only loosely constrained, i.e., the parameters can be set to a wide range of values. Many approaches rely on simple heuristics to constrain the parameters, but these heuristics can fail on many images. Recent work has taken the approach of using image-derived priors that are applicable to a large number of images, and while these methods are promising, at times their success is limited by their generality.

In this work, we explore a new approach for image restoration. Instead of using general priors, we develop constraints that are tuned to the specific "context" of an image and investigate whether a small set of "semantically" similar images selected from a larger image database can provide a stronger, more meaningful set of priors for image restoration.

With our approach, results from a visual search over the image database provide a *visual context* for the input image—that is, a set of images that are similar to the input image in terms of the distance between their representation in some descriptor space. We demonstrate the utility of a visual context with novel algorithms for white balance correction, exposure correction, and contrast enhancement. While we have focused on three restorations, our underlying approach is broadly applicable and can generalize to a large class of problems.

We provide a thorough evaluation of the utility of context-specific priors through several quantitative experiments that compare our approach to existing techniques. Our fully automatic methods demonstrate that a good context-specific prior can be used to restore images with more accuracy than a generic or domain-specific prior.

2. Related Work

Our system builds upon both visual search and image restoration techniques. For visual search, our method selects semantically-similar images using a nearest neighbor search over a large image database. Recent work has demonstrated the effectiveness of such techniques for finding semantically-related images for a variety of vision and graphics tasks [14, 3, 13]. These results indicate that, despite the huge space of all images, such searches can robustly find semantically-related results for large but attainable database sizes [13].

For our restorations, we follow a general class of methods that transfer distributions over color, either for color transfer [11, 10, 12] or for grayscale image colorization [6, 9], and over a two-level grayscale image decomposition for style transfer [1]. In previous approaches, the target

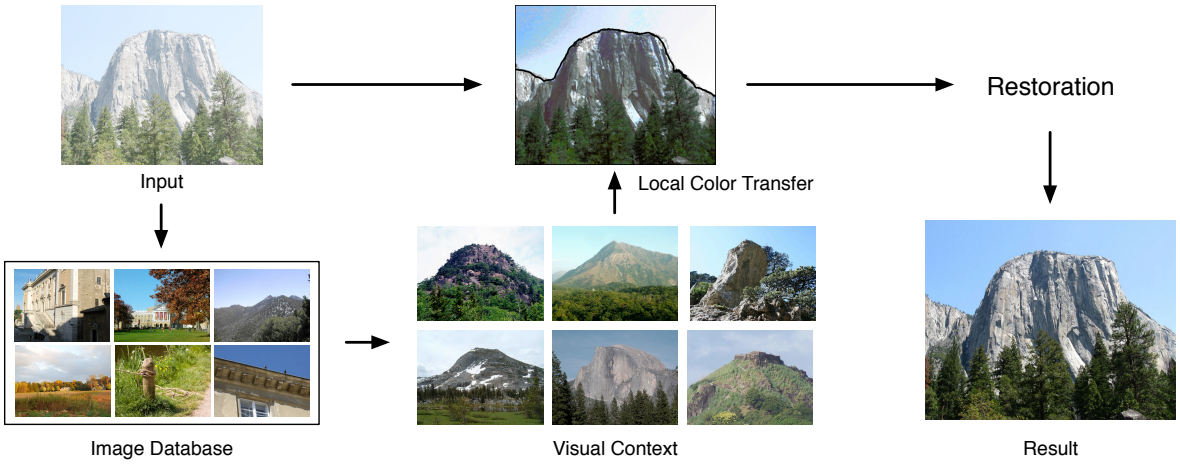


Figure 1: Given an input image, we query a large collection of photographs to retrieve the k most similar images. The k images define the *visual context* for the input image. The visual context provides a prior on colors for local color transfer. The input and color-matched images are used to estimate a global restoration that is applied to the input image to yield the final result.

statistics are manually specified by selecting model images and/or image regions, and, in large part, the metric is one of aesthetics. Here, we are most interested in restoring natural appearance to images, and the relevant components of our method, driven by the image database search, are automatic.

Liu et al. 2008 [7] follow a similar approach to ours, using image results from a web search to drive image colorization. However their method involves image registration between search results and input and requires exact scene matches, which they demonstrate only on famous landmarks for which such exact matches can be found. Our approach instead uses a visual search based on image data and only assumes similar content between input and search results, making it more general than their method.

3. Overview

Given an input image, our image restoration algorithm estimates global corrections to remove deficiencies in the image. Fundamental to our approach is the assumption that global operations can correct the input image. While this assumption does not apply to every image, there are many images where global corrections are reasonable. For example, most cameras have modes to automatically set the white balance and exposure, but these modes can make mistakes leading to color casts or poorly exposed images. Our system can go beyond the algorithms built into cameras by leveraging a large database of images to determine context-specific global corrections for a given image.

Figure 1 shows an overview of our image restoration system. First, we query an image database to retrieve the k closest matches to the input image using a visual search that is designed to be robust to the expected distortions in the input. The results from the search define the visual context for the input.

To take advantage of the visual context, the input image and search results are segmented using a cosegmentation algorithm. This step both segments the images and identifies regional correspondences. Within each region, we transfer colors from the matching segments to the input image. From the color-matched input image, we estimate parameters of a global restoration function to remove the distortion in the input. We consider white balance, contrast enhancement, and exposure correction, though our approach could be applied to other restorations. In the sections that follow, we describe the details of each of these components.

4. Visual Context

At the coarsest level, the visual context for an image should capture the scene class of the image. For example, if the input image is a landscape, the visual context should define properties that are indicative of landscapes, perhaps grass in the foreground, mountains in the background, and sky at the top of the image. Ideally, the visual context will be even more specific, capturing scene structure at approximately the same scale; i.e., similar objects and object arrangements within the scene. The representation should also be tolerant to small changes in scale and changes in illumination.

To achieve these goals, we use a visual search that includes appearance and spatial information at multiple granularities. Our image representation is based on *visual words*, or quantized SIFT [8] features. We use two visual word vocabularies of different sizes, along with the spatial pyramid scheme [5] to retain spatial information. In general, we find that our search descriptor captures many im-

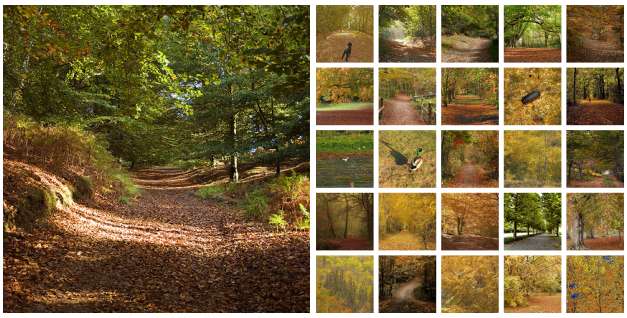
portant qualities of the input image, including scene class, scale, and often object class. In Fig. 2, we show the top 25 search results for an example image. We use the same descriptor layout, visual vocabulary structure, and dimensionality reduction approach as previous image-based retrieval systems; see Johnson et al. 2009 [4] for details of the setup followed here.

For image restoration, we would like the search to be robust to the artifacts that we are trying to correct. For example, if the input image has a faded appearance due to poor contrast, the image descriptor should not be sensitive to this distortion and the search results should be similar, provided the distortion is within a reasonable range. Combining color and gradient information helps to achieve this goal. In particular, SIFT will be near-invariant to the linear transforms for white balance and exposure changes, and, we've found, sufficiently robust to non-linear gamma transforms within a reasonable range.

As in Johnson et al. 2009 [4], we use a color term that is an 8×8 downsampled version of the color image. L*a*b* is not robust to these distortions, however. Search results for images under different white balance settings will obviously be different. Even for example using the a*,b* channels alone for exposure did not work, since these channels aren't completely decorrelated from luminance. Instead, we simply mean- and variance-normalize log-RGB values and downsample. This transforms RGB values into a representation that is invariant to uniform- and non-uniform scaling (exposure and white balance) as well as exponentiation (gamma). We found that this color representation out-performed the spatial pyramid descriptor alone as well as in combination with L*a*b*; this is discussed further in Sec. 7. We weight the pyramid descriptor and distribution-normalized log-RGB descriptor by β and $1 - \beta$, respectively, for parameter $\beta \in [0, 1]$. We found a relative weight of $\beta = 0.75$ to consistently produce good results, and this is used for all results shown in the paper.

5. Cosegmentation

Once we have the visual context, we can take advantage of scene-specific properties to help restore the input image. While there are many ways these properties could be exploited, we show that a simple approach based on color transfer yields compelling results for our three restorations. The core assumption is that the colors of the input are degraded in some way, but the colors of the visual context, when considered across the entire match set, are appropriate for this scene type and can be used to remove the degradation of the input. The simplest approach of using global color transfer techniques, as in Pitié et al. 2005 [10], works reasonably well, but we notice a distinct improvement by using local color transfer based on cosegmentation.



(a) Input (b) Matches

Figure 2: Input image (a) and top 25 search results in row-major order (b). Our image representation effectively discriminates beyond coarse scene classification. Most search results in (b) depict forest scenes at approximately the same scale as the input image, but most notably, a large portion of the matches depict a tree-lined pathway with the sun's illumination partially occluded by foliage above.

Cosegmentation solves two problems simultaneously: it segments the images and identifies regional correspondences between images. Following the work of Johnson et al. 2009 [4], we use an algorithm based on mean-shift with feature vectors that are designed to be robust to distortions of the input image.

We use a feature vector at each pixel p that is the concatenation of the pixel color in L*a*b* space; the mean and standard deviation of L*a*b* in a 3x3 window; the normalized x and y coordinates at p; and a binary indicator vector $(i_0, \ldots, i_k)$ such that i_j is 1 when pixel p is in the j^{th} image and 0 otherwise. The binary indicator vector differentiates between pixels that come from the same image versus those that come from different images. In addition, the components of the feature vector are weighted by three weights to balance the color, spatial, and index components. Before converting to L*a*b*, we normalize the image by dividing by the maximum RGB value; this is necessary for good results on dark images. In general, we find that the parameters of the cosegmentation do not need to be adjusted per image; all results presented in this paper use the same cosegmentation parameters.

Once we have segmented the input and visual context into regions, we perform color transfer within each region to restore their approximate local color distributions.

6. Image Restorations

We consider three global restorations: white balance, exposure correction, and contrast enhancement. All three restorations optimize the same mathematical model, and since we only consider global operations, we can specify them as pointwise functions on individual pixels. Let I be the input image and I^c be the color-matched input (i.e., the

image after local color transfer using the visual context). The restored image I^r at pixel p is given by

$$I^r(p) = R(I(p);\theta)\,, \quad (1)$$
$$\theta = \arg\min E(\theta; I^c, I) \quad (2)$$

where R is an image restoration function and θ is the set of parameters for R that minimizes an error function between the input image I and the color-matched image I^c.

White balance

For white balance, we model the restoration as a 3×3 diagonal transform. Let I_r, I_g, and I_b be the RGB values at pixel p for the input. The white balance restoration is defined in terms of three parameters $\theta = \begin{pmatrix} \alpha_r & \alpha_g & \alpha_b \end{pmatrix}$:

$$R(I(p),\theta) = \begin{pmatrix} \alpha_r & 0 & 0 \\ 0 & \alpha_g & 0 \\ 0 & 0 & \alpha_b \end{pmatrix} \begin{pmatrix} I_r(p) \\ I_g(p) \\ I_b(p) \end{pmatrix}. \quad (3)$$

The error function for white balance is the squared error over all pixels between the color-matched image I^c and the restored input I:

$$E(\theta; I^c, I) = \sum_p \|I^c(p) - R(I(p);\theta)\|^2\,. \quad (4)$$

The error function has an analytic minimum. For channel k of the image, the scalar α_k that minimizes the error function is:

$$\alpha_k = \frac{\sum_{p\in k} I(p) I^c(p)}{\sum_{p\in k} I(p)^2} \quad (5)$$

where $p \in k$ denotes all pixels in channel k of the image.

Exposure correction

Overall scene brightness, or key, is commonly computed as the log-average luminance of the image [15]. For image I, the key is given as

$$K(L) = \exp\left(\frac{1}{N}\sum_p \log(L(p) + \delta)\right), \quad (6)$$

where L is the luminance image computed from I, N is the number of pixels in the image, and δ is added to handle zero-valued pixels in L.

If an image is captured with an incorrect exposure, it can be approximately adjusted as a post-process by scaling the image by a factor $\alpha/K(L)$, where α is the target key. Therefore, the restoration function for exposure is simply scaling the image:

$$R(I(p),\alpha) = \alpha I(p)\,, \quad (7)$$

where the restoration parameter is a scalar α.

The parameter α can be estimated by minimizing a function that is similar to the error function for white balance, except the unknown scale factor applies across all three color channels:

$$E(\alpha; I^c, I) = \sum_p \|I^c(p) - R(I(p);\alpha)\|^2\,. \quad (8)$$

The optimal α is:

$$\alpha = \frac{\sum_p I(p) I^c(p)}{\sum_p I(p)^2}\,, \quad (9)$$

where the summation is across all pixels in all color channels.

Contrast enhancement

We model the restoration function for contrast as a gamma correction. In this case, the parameter of the restoration function is a scalar γ:

$$R(I(p),\gamma) = I(p)^\gamma\,. \quad (10)$$

The appropriate gamma is estimated from the color-matched image by solving a least-squares problem on log images:

$$E(\theta; I^c, I) = \sum_p \omega_p \|\log I^c(p) - \log R(I(p);\theta)\|^2\,, \quad (11)$$

where ω_p is a weight to prevent pixels with large magnitudes in log space (corresponding to small intensities) from skewing the result. We find that setting ω_p to the squared (normalized) intensity $I(p)$ works well in practice. As with white balance, the resulting error function has an analytic minimum:

$$\gamma = \frac{\sum_p \omega_p (\log I^c(p))(\log I(p))}{\sum_p \omega_p (\log I(p))^2} \quad (12)$$

7. Results

We perform our evaluation using a database of 1 million images crawled from Flickr using search keywords related to outdoor scenes, such as 'beach', 'forest', 'landscape', etc. [4]. From the database, we selected a set of 100 relatively artifact-free test inputs such that the various types of outdoor scenes found in the database were well-represented (see Fig. 3).

We chose to focus on outdoor scenes for several reasons. In general, we have found that the performance of our system improves with larger database sizes. By reducing the scope of the class of input images and generating a targeted database for that class, we can simulate the effect of a much larger database on a set of generic inputs. Additionally,

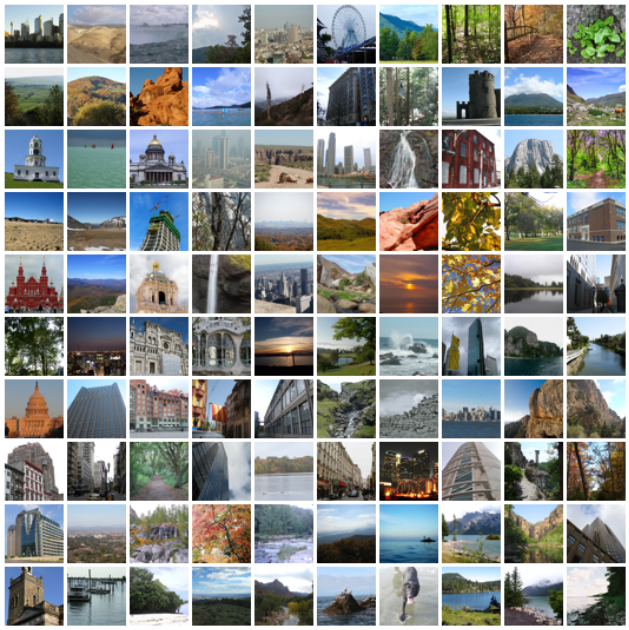


Figure 3: The set of inputs used in the synthetic tests that cover a variety of different outdoor scenes.

from preliminary results using a generic database of both indoor and outdoor scenes, the variation across search results for indoor scenes—e.g., in regular structure, complex lighting, and foreground objects—was found to be far more perceptible than for outdoor scenes. This observation suggests that indoor scenes would require a significantly larger database to yield equivalent results. Considering these issues, we chose to focus specifically on outdoor scenes for our evaluation.

We follow the same testing methodology for all three restorations: we apply a distortion to the input to approximate a real image artifact and attempt to remove the distortion using our system. In all tests, we query the database using the distorted input image and retrieve the visual context from the database using a leave-one-out strategy; i.e., we disregard a given input when it is recovered in its own visual context. We apply our restoration method to the distorted input and estimate the parameter or parameters of the distortion. To evaluate our performance, we compare the estimated and actual distortion parameters. We also apply an alternative reference algorithm based on a generic prior to the distorted input for comparison.

7.1. White balance

For white balance, we distort our input images using the following distortion model:

$$D(I(p),t) = \begin{pmatrix} 1+\frac{t}{2} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1-\frac{t}{2} \end{pmatrix} \begin{pmatrix} I_r(p) \\ I_g(p) \\ I_b(p) \end{pmatrix} . \tag{13}$$

This distortion model changes the balance of the red and blue channels relative to the green channel without changing the luminance of the image. The parameter t varies between 0 and 1.

The white balance distortion and restoration involves three parameters–the scalars on the individual color channels. To measure the error between the actual and estimated parameters, we compute the angle between these parameter sets, normalized to be unit length vectors.

For white balance tests, we compared against Gray World, Gray Edge, Max-RGB, and Shades-of-Gray methods [16]. Although Gray World is perhaps the most well-known generic prior for white balance, we found that Gray Edge performed consistently better than the other methods.

In Fig. 4a, we show our results on white balance restoration compared to both Gray World and Gray Edge. On the horizontal axis we show the distortion induced by the distortion model, Eqn. 13, and on the vertical axis, the error in the estimated distortion. Each data point is the mean over 100 images, with error bars showing standard error. For all distortions, we outperform the Gray World assumption. For small distortions, we outperform Gray Edge, though Gray Edge is better for large distortions.

We also compare white balance results for different color representations used in the visual search. Fig. 9 shows results based on search results using our normalized log-RGB color term, an L*a*b* color term, and no color term. Using an L*a*b* color descriptor produces search results with color similar to the distorted input, leading to significantly more error than when using no color term at all. However the mean- and variance-normalized log-RGB color descriptor improves results significantly across the entire range of distortions.

7.2. Exposure

The exposure distortion is a scaling of all three channels in an image by a constant factor:

$$D(I(p),t) = tI(p) . \tag{14}$$

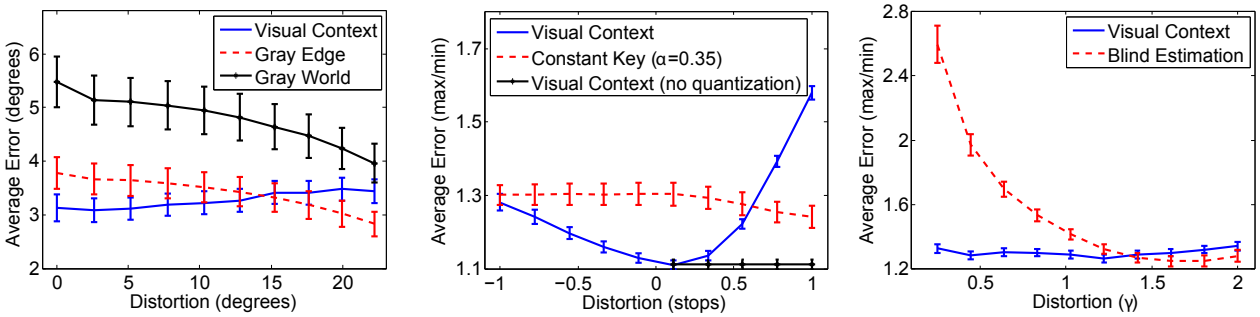
We vary the parameter t in fractional powers of two, from 2^{-1} to 2^{1}.

To measure error between the estimated and actual parameters, we compute the distance between the parameters in $\log$ space and raise this to the power 2, i.e.:

$$e(\alpha_1,\alpha_2) = 2^{|\log_2 x - \log_2 y|} . \tag{15}$$

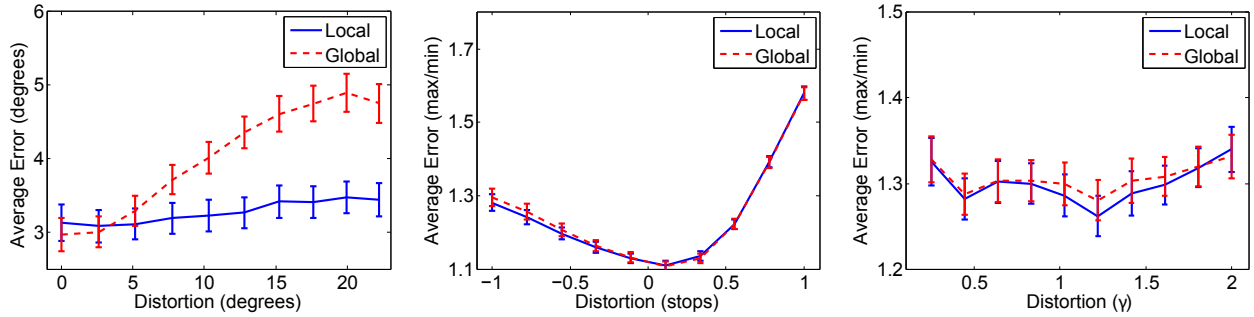
This error measure is the same as the computing the ratio $\max(x,y)/\min(x,y)$.

For exposure tests, we compare against using a constant-key assumption. A key of $\alpha = 0.18$ is a common generic target. Our Flickr database of outdoor scenes is, on average, brighter, justifying a target key of $\alpha = 0.35$.



(a) White balance (b) Exposure (c) Contrast

Figure 4: Comparison with other methods. Each plot shows average error across 100 test images for 10 distortions. While the visual context approach produces less error over a the majority of each distortion range, generic priors excel for large white balance (a) and contrast (c) distortions. In (b), the difference between the black and blue curves illustrates the impact of quantization on our method for large positive exposure distortions.



(a) White balance (b) Exposure (c) Contrast

Figure 5: Comparison between local and global approaches. (a) Local white balance shows improvement over all but the smallest distortions. Cosegmentation provides less benefit for (b) exposure and (c) contrast correction.

In Fig. 4b, we compare our restoration technique for exposure to the constant-key assumption. On the horizontal axis is the logarithmic amount of scaling (similar to exposure stops) applied to the image, i.e., scaling from 2^{-1} to 2^{1}. On the vertical axis is error measured according to Eqn. 15. For stops below $2^{0.5} \approx 1.4$, we outperform the constant-key assumption.

For stops above $2^{0.5}$, our distortion technique of clipping and quantizing the image affects our performance. Intuitively, for the extreme case of scaling by 2, all values above 128 in an 8-bit image will become saturated by this distortion. The saturation affects both the image search and cosegmentation. Without clipping and quantization, our performance is better than the constant-key assumption, even for large distortions. While this doesn't reflect performance on common JPEG-compressed 8-bit images, it is a reasonable simulation for higher-precision formats. It is becoming increasingly popular for non-professionals to work in RAW.

7.3. Contrast

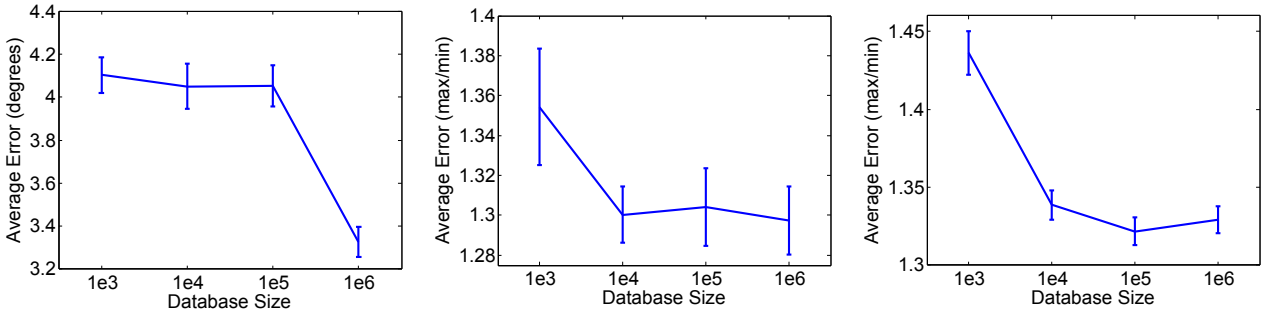
To distort contrast, we apply a gamma to the image:

$$D(I(p), t) = I(p)^t \,, \tag{16}$$

where the parameter t varies between 0.5 and 2. Here, we compare against the blind inverse gamma correction method of Farid 2001 [2]. This algorithm measures higher-order correlations in the frequency domain to estimate the gamma nonlinearity. We allow the algorithm to search over our range of distortions to estimate gamma.

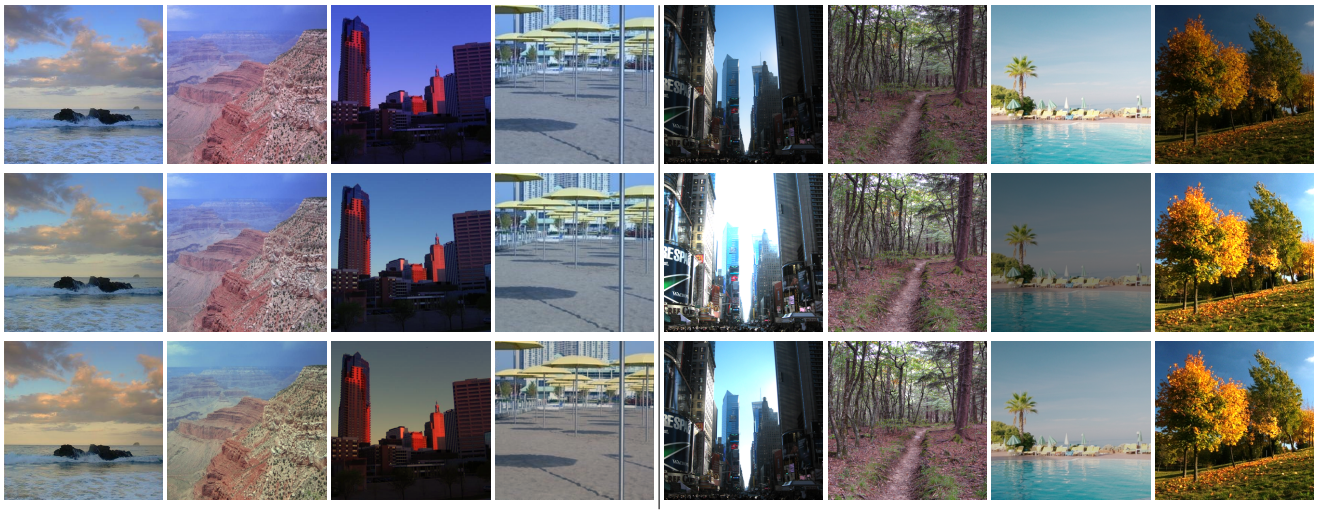
Comparison results for contrast are shown in Fig. 4c. For small γ values, we do significantly better in recovery, and we are comparable for larger values.

Finally, in addition to experimental results using synthetically distorted input images, we show examples on real input data for all three restorations. Figs. 7 and 8 show natural input images suffering from artifacts, along with results from our restoration algorithms and competing solutions.



(a) White balance (b) Exposure (c) Contrast

Figure 6: Performance across database size. We average errors across all 100×10 trials, for each database size. Moderately sized databases perform comparably to the full 1M image database for single-parameter estimation in (b) exposure and (c) contrast correction, while the 1M image database shows a significant improvement over smaller databases for (a) white balance correction.



(a) White balance (b) Exposure correction

Figure 7: White balance and exposure results on real inputs. In (a) and (b), the top row shows the input, the middle, the reference result (Gray Edge in (a) and constant key, $\alpha = 0.35$, in (b)), and the bottom, results with the visual context.

7.4. Database size

Database size and coverage can substantially affect the final restoration result. For an input image with unique features not represented in its visual context, our restoration algorithms will reduce or eliminate these features while correcting the remainder of the image. The degree to which this occurs is, in general, a property of the database and will naturally diminish with increasing database size and coverage.

This same issue manifests itself most apparently when the database search fails to find good, semantically-relevant matches. When this happens, the results from the image restoration algorithms suffer as well. The likelihood of this sort of failure will likewise decrease with a larger database.

However, the degree to which increasingly large databases can improve results for database-driven approaches such as ours is often unclear. Fig. 6 shows average error for different database sizes for white balance, contrast, and exposure. Significant improvement in results for white balance only occurs between 100K and 1M images, suggesting that an even larger database could improve the results. However for exposure and contrast, these results indicate that a relatively small 10K image database is sufficient to obtain results comparable with the larger 1M image database. While there are many different aspects to the pipeline, this is likely due to the simple difference between estimating three parameters versus one.

Figure 8: Contrast results on real inputs. The top row shows the input, the middle, the reference result (blind correction [2]), and the bottom, results with the visual context.

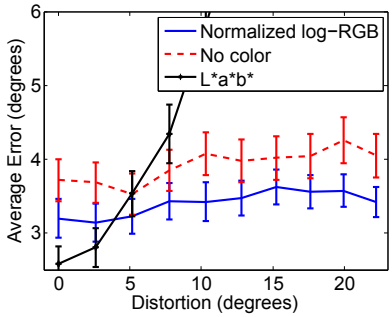


Figure 9: Results for white balance for different color representations. The L*a*b* curve continues to grow across the distortion range, with an average error of 13.1 degrees for the largest distortion.

8. Conclusion and Future Work

We have demonstrated a system that leverages a large image database for image restoration. For multiple restoration algorithms–white balance correction, contrast enhancement, and exposure correction–we have shown how specifying a prior based on the results of a visual search can produce results superior to similar algorithms using more generic image priors. Additionally, we showed that relatively small database sizes are sufficient for robust exposure and contrast correction.

Our pipeline is sufficiently flexible to be used for a number of image-based applications beyond those discussed in this paper. In general, any image-based algorithm that can benefit from a more precise prior is a candidate for this approach. While we use a coarse local approach with cosegmentation, exploring patch-based local methods built upon the visual context is one future direction. Investigating specific online collections–e.g., professional photographs and domain-specific collections–could also lead to improved results in restorations based on the visual context.

Acknowledgements

We would like to thank Sylvain Paris and Todd Zickler for their valuable feedback, Josef Sivic and Biliana Kaneva for sharing data, and Lior Shapira for providing source code. Kevin Dale and Kalyan Sunkavalli acknowledge support from the John A. and Elizabeth S. Armstrong Fellowship. Kimo Johnson acknowledges NSF grant DMS-0739255 and support from Adobe Systems.

References

[1] S. Bae, S. Paris, and F. Durand. Two-scale tone management for photographic look. *ACM TOG*, 25(3), 2006. 1

[2] H. Farid. Blind inverse gamma correction. *IEEE TIP*, 10(10), 2001. 6, 8

[3] J. Hays and A. A. Efros. Scene completion using millions of photographs. *ACM TOG*, 26(3), 2007. 1

[4] M. K. Johnson, K. Dale, S. Avidan, H. Pfister, W. T. Freeman, and W. Matusik. CG2Real: Improving the realism of computer-generated images using a large collection of photographs. Technical Report MIT-CSAIL-TR-2009-034, CSAIL MIT, 2009. 3, 4

[5] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Proc. of CVPR*, 2006. 2

[6] A. Levin, D. Lischinski, and Y. Weiss. Colorization using optimization. *ACM TOG*, 23(3), 2004. 1

[7] X. Liu, L. Wan, Y. Qu, T.-T. Wong, S. Lin, C.-S. Leung, and P.-A. Heng. Intrinsic colorization. *ACM TOG*, 27(5), 2008. 2

[8] D. G. Lowe. Object recognition from local scale-invariant features. In *Proc. of ICCV*, 1999. 2

[9] Q. Luan, F. Wen, D. Cohen-Or, L. Liang, Y.-Q. Xu, and H.-Y. Shum. Natural image colorization. In *Proc. of EGSR*, 2007. 1

[10] F. Pitié, A. C. Kokaram, and R. Dahyot. N-dimensional probability density function transfer and its application to colour transfer. In *Proc. of ICCV*, 2005. 1, 3

[11] E. Reinhard, M. Ashikhmin, B. Gooch, and P. Shirley. Color transfer between images. *IEEE CG&A*, 21(5), 2001. 1

[12] Y.-W. Tai, J. Jia, and C.-K. Tang. Local color transfer via probabilistic segmentation by expectation-maximization. In *Proc. of CVPR*, 2005. 1

[13] A. Torralba, R. Fergus, and W. T. Freeman. 80 million tiny images: A large dataset for non-parametric object and scene recognition. *IEEE PAMI*, 30(11), 2008. 1

[14] A. Torralba, K. P. Murphy, W. T. Freeman, and M. A. Rubin. Context-based vision system for place and object recognition. In *Proc. of ICCV*, 2003. 1

[15] J. Tumblin and H. Rushmeier. Tone reproduction for realistic images. *IEEE CG&A*, 13(6), 1993. 4

[16] J. van de Weijer, T. Gevers, and A. Gijsenij. Edge-based color constancy. *IEEE TIP*, 16(9), 2007. 5