Example-based image color and tone style enhancement

ACM Transaction on Graphics,
Vol. 30, No. 4, 2011

Baoyuan Wang, Yizhou Yu, Ying-Qing Xu

Presented by Ran Shu

School of Electrical Engineering and Computer Science
Kyungpook National Univ.
Abstract

- Color and tone adjustments
  - Most frequent image enhancement operations
  - Defining a color and tone style
    - Set of explicit or implicit rules
    - Learning implicit color and tone adjustment rules
      - From examples
        » Corresponding images before and after adjustments
      - Finding underlying mathematical relationships
        » Connecting color and tone of corresponding pixels
  - Defining tone and color adjustment rules as mapping
    - Approximating complicated spatially varying nonlinear mapping
      - In piecewise manner
Framework in two scenarios

- Low-quality photo enhancement
  - Transferring style of high-end camera
- Photo enhancement using styles
  - Learning from photographers and designers
- Style enhancement results

**Fig. 1.** (a) Original photo taken by iPhone 3G, (b) enhanced photo that mimics the color and tone style of Canon EOS 5D Mark; (c) Original photo, (d) enhanced photo with a style learned from a photographer.
Introduction

- Tone and color adjustments
  - Determined on individual basis
  - Existing many scenarios
    - Following common implicit rules
      - Tuning temperature and tint of colors
        » Conveying specific impressions
      - Different digital cameras
        » Varying degrees of tone and color discrepancies
  - Manually adjusting tone and color of photograph
    - Tedious and labor-intensive
– Tone and color styles formulated mathematically
  • Automatically and easily applying to input image
  • Rules governing styles not explicitly
    – Difficult to mathematically summarize rules
      » Using to achieve certain impression
    – Calibrating radiance and color response curves of camera
      » Especially considering color response curves
        » Covering entire visible spectrum
– Discovering underlying mathematical relationships
  • Optimally connecting tone and color of corresponding pixels
    – In all image pairs
Learning implicit tone and color adjustment rules

- From examples before and after adjustments
  - Relationships buried in noisy data
    - Discovering hidden patterns and relationships
      » Relying on machine learning and data mining techniques
  - Relationships highly nonlinear and spatially varying
    - Camera radiance and color response curves nonlinear
    - Rules used by photographers varying
      » According to tone and color characteristics of local image regions
  - Existing other factors in relationships
    - In addition to tone and color of individual pixels
Proposing learning based method
- Defining tone and color adjustment rules as mapping
- Approximating complicated spatially varying nonlinear mapping
  - In a piecewise manner
    - Locally approximated with linear or low-order polynomial model
      » Dividing feature space into a number of subspaces
      » Approximating tone and color mappings
        » Within each subspace with low-order model
- Developing systematic framework for tone and color styles
  - Dividing feature space into subspaces
    - Using binary classification trees
- Identifying low-order parametric models
  - Local color mapping
  - Local gradient mapping
    - Contrast enhancement
- Applying framework in two scenarios
  - Learning styles of high-end digital camera
    - Using to enhancing photographs
      » Taken by low-end cameras
  - Learning styles of images manually enhanced by photographers
    - Applying to novel input photographs
Related work

- Performing tonal or color adjustments to input images
  - Without learning parametric models
    - Model of lightness and color perception of human vision
      - Multiscale center/surround retinex
        » Dynamic range compression
        » Color consistency
        » Lightness rendition
      - Automatic local and global tonal adjustments
        » Making its local and global contrasts similar to reference image
      - Editing image appearances interactively
        » Using Gaussian mixture models
Gradient domain image editing techniques and systems

- High dynamic range compression
  - Reducing magnitude of image gradients
    - Altering and re-integrating original gradient field

- Difference with proposing approach
  - Identifying connection between gradients
    - From photos with two different styles
  - Modeling connections as parametric gradient mappings
    - Previous only performing gradient domain adjustments
      » To individual input images
      » Without training parametric models
Image color transferring

- Performing statistical analysis
  - Imposing color characteristics of reference images
    - Onto another source image
  - Difference with proposing approach
    - Obvious research
      - Transferring colors from one or multiple reference images
        » To source image
      - Obtaining mappings working for specific source image
    - Proposing method
      - Training a set of tone and color mappings
        » Generally applicable to a class of images
        » Without choose a reference image for each source image
Enhancing digital images

- Obtaining from low quality imaging devices
  - Classifying pixel color into category
    - Using Gaussian mixture model
  - Local color correction
    - Using an affine transform
      » Specifically tailored for category

- Difference with proposing approach
  - Performing color correction and contrast enhancement same work
    - Previous only performing color correction
  - Color adjusting using quadratic models
    - Instead of affine models
  - Image and pixel categorizing automatically extracted from training data
    - Using hierarchical clustering
      » Previous only suitable for imaging devices
      » Not for photographer stylistic adjustments
Overview

- Training stage
  - Collecting representative training image pairs
    - Learning hidden relationships between corresponding pixels
      - Finding closed-form parametric model
        » Representing mappings in piecewise manner
  - Extracting local statistical features from image pairs
    - Features including average and standard deviation of colors
      - Within a neighborhood
      - Building binary feature space partition tree
        » Dividing feature space $F$
          » Into $n$ subspaces

\[ F = F_1 \cup F_2 \cup F_3 \cup, ..., \cup F_n \]
– Building second binary feature space partition tree
  • Relating to luminance gradients
    – Along edges affecting overall image contrast
  • Training local mappings for luminance gradients along edges
– Training stage

**Fig. 2.** Leaf nodes are colored in green while intermediate nodes are colored in yellow
Style enhancement

- Applying local mappings in binary feature partition trees
  - Adjusting color and contrast of given image
    - Enhancing intended style
- Improving image color
  - Segmenting input images into multiple soft segments
  - Trained local color mapping
    - Adjusting colors of soft segments
      » In spatially coherent manner
- Improving image contrast
  - Applying local luminance gradient mappings
    - To luminance gradients along edges
      » Leaving other pixel gradients unchanged
    - Using tone optimization step
      » Mapped luminance gradients
      » Luminance channel of previously mapped colors
Learn Color and Gradient Mappings

- Hierarchical feature space subdivision
  - Partitioning entire feature vectors
    - Collecting from all training image pairs into subsets
      - Using binary hierarchical clustering
  - Defining feature subspaces
    - According to clustering results
      - Euclidean distance to cluster centers
        » Constructing Voronoi diagram for cluster centers in feature space
        » Cluster centers only mean of all feature vectors in same cluster
        » Without boundary shape and spatial distribution of individual cluster
    - Taking alternative approach
      » At every intermediate node obtaining partition surface
        » Optimally separating feature vector
        » Using SVMS and RBF kernel
Comparison of root mean squared errors (RMSE)

- Classifier based method
- Cluster center based method

Table 1. Four groups of testing data. RMSEs were measured for the three channels of the CIE $L^*a^*b^*$ color space separately.
Learning local color mappings
- Approximating global color mapping in piecewise manner
  - Dividing feature space into subspace
    - Expecting local color mapping accurately represented
      » By low-order parametric model

\[ c_j^h = AQ_j + b \] (1)

Where \( c_j^h = (L_j^h, a_j^h, b_j^h) \) is color vector at \( p_j^h \),
\( (p_j^l, p_j^h) \) is a pair of corresponding pixels,
from training image pairs \( \{I_j^l, I_j^h\}_{i=1}^m \),
\( Q_j \) is polynomial basis vector defined for \( c_j^l \),
\( (A, b) \) is affine transformation between \( Q_j, c_j^h \),
\( A \) is 3X3 matrix for linear basis, 3X9 matrix for second-order basis,
\( b \) is always a 3-vector
Given sufficient number of corresponding pixels
  » Associating with feature subspace
    » Solving A, b

\[
\arg\min_{A,b} \sum_{j=1}^{N} \left\| AQ_j + b - c^b_j \right\|^2
\]

Where \( N \) is the number of corresponding pixels,

- Overall visual difference between enhanced result and reference

Fig. 4. (A) is an original photo taken by iPhone 3G while (D) is the reference photo taken by Canon 5D Mark. (B) is the result from an affine color mapping while (C) is the result from a quadratic color mapping.
- Exploring underlying correlation between $Q_j, c_j^h$
  - Using Canonical Correlation Analysis (CCA)
    - For multivariate correlation analysis
    - Comparison of correlation results

![Fig. 5. Comparison of correlation results for the linear and quadratic color mapping models.](image)
Learning mapping for luminance gradients

- Luminance gradients along edges
  - Affecting overall image contrast
    - Important factor of image tone style
- Building a second binary feature space partition tree
  - Relating to luminance gradients
  - Training local luminance gradient mapping
    - For every leaf node
  - Modeling a local mapping between corresponding luminance values

\[ L_j^h = L_j^l \gamma \]  
Where \( L_j^h, L_j^l \gamma \) are the luminance channels of corresponding pixels

- Taking logarithm of gradient of both sides

\[ \ln \frac{\|\nabla L_j^h\|}{\|\nabla L_j^l\|} = \ln \gamma + (\gamma - 1) \ln L_j^l \]  
(4)
• Instead of estimating $\gamma$ directly
  
  - Generalizing relationship in (4) to generic linear mapping

\[
\underset{\alpha, \beta}{\text{arg min}} \sum_{j=1}^{N} \left( \alpha + \beta \ln L_j^l - \ln \left( \frac{\nabla L_j^h}{\nabla L_j^l} \right) \right)^2
\]  

(5)

Where $\alpha, \beta$ are parameters in linear mapping

• Formulating luminance gradient mapping

\[
\nabla L_j^h = \nabla L_j^l \exp \left( \alpha + \beta \ln L_j^l \right)
\]  

(6)
Spatially coherent color and tone mapping

- Given trained partition tree and local color mappings
  
  - For every pixel in new input image
    - Extracting feature vector
    - Going through binary tree
      » Locating subspace feature vector belonging to
    - Applying local color mapping
      » Associating with subspace to color channels of pixel
        » Obtaining new color

- Straightforward process
  
  - Giving rise to noise color mapping results
    » Subspace search for neighboring pixels
      » Carried out independently
    » Two adjacent pixels end up within two different subspaces
      » with different color mappings
– Proposing segmentation-based mapping
  • Achieving spatially coherent mapping result
    – Dividing input image into multiple soft segments
      » Each associating one probability value with every pixel
      » Summing up to 1
    – Removing potential ambiguity
      » Requiring largest probability value at every pixel
      » Larger than 0.5
  • Supposing K soft segments
    – Selecting pixels
      » Probability larger than 0.5
    – Building subspace voting histogram H
      » Bins corresponding to leaves of color mapping tree
Choosing three most frequently visited subspace and local color mappings

» Applying three chosen mapping to all pixels

\[ \tilde{c}_i^t = \sum_{j=1}^{3} \lambda_j^t \left( A_j^t Q_i + b_j^t \right) \]  

(7)

Where \( Q_i \) is second order polynomial basis vector for original color of input image at pixel \( i \), supposing three chosen mapping for t-th segment \( (A_j^t, b_j^t) \) values of corresponding normalized histogram bin counters \( \lambda_j^t \)

» Final color of pixel

\[ \tilde{c}_i = \sum_{t=1}^{K} P_i^t \tilde{c}_i^t \]  

(8)
- Image enhancement pipeline

Fig. 3. Image enhancement pipeline.
Luminance gradient mapping

- Applying local mappings of luminance gradients
  - Enhancing contrast of input image
    - Applying to individual edge pixels independently
      » Without considering spatial coherence
  - Using Canny edge detector
    - Finding edge pixels
Tone optimization

- Mapped luminance value for every pixel
  - Including edge pixels
  - Luminance is one of three channels in CIE Lab color space
- Performing tone optimization
  - Producing a new luminance channel
    - Consistent with mapped luminance values and luminance gradient
    - Formulating tone optimization
      \[
      \arg\min_{\hat{L}} \sum_i \left( \omega_i \| \nabla \tilde{L}_i - g_i \|^2 + \left( \hat{L}_i - \tilde{L}_i \right)^2 \right)
      \]  \hspace{1cm} (9)

Where \( \hat{L} \) is the unknown new luminance channel,
\( \omega_i \) is a spatially varying weighting coefficient
- Replacing luminance channel estimated in (8)
  » Using new luminance channel obtaining from (9)
  » Completing our color and tone style enhancement
Fig. 6. (A) is an original photo taken by iPhone 3G. (E) is the reference photo taken by Canon 5D Mark. (B) is obtained with both color mapping and tone optimization, but a feature vector only consists of three color channels. (C) is obtained directly from color mapping as in (8) without additional tone optimization. (D) is our final result with color and gradient mappings, tone optimization, and a feature vector includes additional neighborhood statistics other than color channels.
Applications and Results

- Photo enhancement by style transfer
  - Increasing color fidelity and improving contrast for photos
    - Taken by low-end cell phone
      - Transferring and mimicking color and tone style of high-end camera
  - Data collection
    - Low-end cameras
      - iPhone 3G
      - Android Nexus One
    - High-end DSLR camera
      - Canon EOS 5D Mark
    - Learning color and gradient mapping models
      - Between low-end and high-end camera
• Taking photos indoor and out door
  − Diverse colors
  − Under different illumination conditions
    » Taking photos using low-end camera
    » Taking corresponding high-quality photos
      » From same viewpoint
• A subset of training image pairs

Fig. 7. The top four were taken by iPhone 3G while the bottom four by Canon 5D Mark.
Luminance normalization and image registration

- Performing luminance normalization
  - Improving luminance consistency
    » Among photos
  - Increasing registration accuracy
  - Converting all photos from RGB to CIE Lab
    » Involving inverse gamma correction
    » Removing most of nonlinearity caused by camera radiance response curve
  - Normalizing average luminance to 0.65

- Two different cameras not registered default
  - Performing pixel-level registration between pair images
    » Used in training stage
  - Using revised version of image matching algorithm
– Feature selection
  • Low end camera
    – Three moments of three channels of CIE Lab in local window
      » Centering at pixel
    – Gradient vectors of three color channels
    – Color correlation matrix of three color channels in same windows
  • High end camera
    – Luminance gradient
    – Pixel color
Training

• Dividing feature space for gradient mapping
  – Into 200 subspaces

• For color mapping
  – Into 300 subspaces

• Relationship between subspaces and average correlation
  – For all subspaces

**Fig. 8.** Correlation analysis is performed between the input and output of the local mappings
- Testing and validation
  - Comparing original and enhanced image with reference
    - Calculating root mean squared error (RMSE)
    - Calculating root mean squared L2 distance
      » In CIELAB color space
    - Computing average and standard deviation

Table 2. Comparison of average and standard deviation (SD) of RMSEs of the RGB channels between original and enhanced low-quality photos, and comparison of average and standard deviation of the root mean squared L2 distances in the 3D CIELAB color space, denoted as D, between original and enhanced low-quality photos.

<table>
<thead>
<tr>
<th></th>
<th>Before Enhancement</th>
<th>After Enhancement</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RGB</td>
<td>D</td>
</tr>
<tr>
<td>Average</td>
<td>(0.108, 0.112, 0.183) 0.076</td>
<td>(0.081, 0.074, 0.075) 0.037</td>
</tr>
<tr>
<td>SD</td>
<td>(0.042, 0.048, 0.056) 0.015</td>
<td>(0.039, 0.039, 0.029) 0.014</td>
</tr>
</tbody>
</table>
• Visual results for two testing photos

Fig .9. Left: Original photos (the top one was taken by Android Nexus One while the bottom one was taken by iPhone 3G). Middle: Our enhanced results. Right: Reference photos taken by Canon.
Comparison with color checker calibration

- Macbeth color checker
  - Color chart for color calibration

Fig. 10. (A) Original photos taken by Panasonic DMC-LX3GK. (B) Calibrated photos by Macbeth color checker. (C) Our results. (D) Reference photos taken by Canon 5D Mark. Our results are closer to the reference photos.
• Photo enhancement results
  – using the style of a high-end camera

**Fig. 11.** Top: Original photos of indoor and outdoor scenes (the leftmost one was taken by Android Nexus One, while the other three were taken by iPhone 3G). Middle: Enhanced results. Bottom: Reference photos taken by Canon.
Learning color and tone styles from photographers

- Training data
  - Collecting 20 pairs of training images
  - Training images from two different styles

Fig. 12. A subset of training examples for two different styles. Top: Original photos. Bottom: Photos enhanced by two photographers (the left two belong to the first style, and the right two belong to the second style).
Results

- Using styling rules
  - Applying to novel input photos to enhance desired style
  - Showing vivid results

Fig. 13. Photo enhancement results using two styles learned from photographers. Top: Original photos. Bottom: Enhanced results (the left two belong to the first style, and the rightmost one belongs to the second style).
Comparison between our method and previous color transfer method

![Comparison between our method and previous color transfer method](image)

Fig. 14. Comparison between our method and color transfer by [Piti and Kokaram 2007]. Our enhanced result is based on the first style in Figure 13.
Conclusions

- Developing a method
  - Learning color and tone styles from examples
    - Capable of discovering underlying mathematical relationships
      - In color and tone between corresponding image pairs
  - Applying in two scenarios
    - Low-quality photo enhancement
      - Transferring style of high-end camera
    - Photo enhancement
      - Using styles learned from photographers and designers
  - Limitations
    - Requiring training stage
      - Data-driven models optimized
    - Requiring pixel-level image registration
      - Within every training image pairs