Efficient hue-preserving and edge-preserving spatial color gamut mapping

Fifteenth Color Image Conference
pp. 207-212, Nov. 2007
Oyvind Kolas and Ivar Farup
Presented by Ho-Gun Ha

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

◆ Proposed method
  – New efficient hue and edge-preserving spatial color gamut mapping
    • Projecting all out of gamut colors to the destination gamut boundary
    • Modified map which is Applied an edge preserving smoothing filter
    • Compressing towards the gamut center according to the filtered map
Introduction

🔹 Classifying gamut mapping algorithms (GMA)
  – Spatially invariant GMA
  – Spatial GMA
    • Minimized functional method
      – Contrast measure
      – Retinex-related measure
      – Optimization
      – Variational approach
    • Combined method
      – Performing some kind of spatially invariant algorithm, and then reinserting some of the high-frequency information
- Advantage of spatial GMA
  - Improving the rendering details
- Disadvantage of spatial GMA
  - Halos
    - Artifacts near sharp edge when high frequency information is reinjected
  - Hue changes
    - Severely disturbing hue close to the sharp edges
  - Computation time
    - Time consuming method
    - Not easily implemented in terms of simple 3D LUT
Proposed algorithm

- General idea
  - Clipping image
  - Constructing a map indicating the relative amount of compression needed for each pixel to achieve clipping
  - Smoothing map by mean of spatial filter
  - Compressing the color of the original image according to the spatially filtered map
– Hue preserving clipping
  • Intermediate relative colorimetric image
    – Preserving hue out of gamut color

Fig. 2. Gamut clipping along straight lines towards the center of the L axis of the CILAB color space within the gamut
– Map computation
  • How much the individual colors of the image had to be compressed towards grey during clipping

\[
m = \frac{\|C' \circ g\|_2}{\|C \circ g\|_2}
\]

where \(C\) is the color of the original pixel, \(C'\) is the color of the gamut clipped image, \(g\) is the center of the gamut on the axis, and \(\| \cdot \|_2\) denotes the \(L_2\) norm of the color space.
• Computed compression map
  – Convex linear combination of the original image and gray using the map as the ratio
  – Upper bound for saturation we can reproduce for the individual colors in the image

\[ C' = mC + (1\cdot m)g \] (2)
- Spatial filtering
  - Changing the amount of compression applied to each pixel
    - Filtering the map through an edge preserving smoothing minimum filter controlled by edge information
  - Modifying the generated map
    - Not increasing values in map
    - Continuous in continuous of original image
Fig. 1. Overview of the proposed spatial gamut mapping algorithm
Results and Discussion

- Resulting image
  - HP Design Jet printer printing

Fig. 3. Images gamut mapped using the proposed method algorithm. The central column shows the original image, and the gamut clipped image is shown on the left hand side. The results of the proposed algorithm is shown on the right hand side.
– Comparison of five state of the art GMAs

Fig. 4. Images gamut mapped using various spatially variant and invariant techniques. From left to right, top to bottom, the original image and results produced by hue preserving minimum distance clipping, SGCK, the presently proposed algorithm, the algorithm of Farup et al., and the algorithm proposed by Zolliker and Simon.
– **Performance**

  • Reproduced images
    – Colors of in gamut regions are relatively colorimetrically reproduced
    – Colors of out of gamut and in vicinity have been compressed in a hue preserving manner
  
  • Analyzing the resulting image
    – Clearly rendering details
      » Particularly in dark regions
    – Not showing any signs of haloing artifacts near sharp boundaries
      » Due to edge preserving spatial filter
- Computational complexity
  - First part of the algorithm \( O(N) \), \( N \) is the number of pixels
    - Gamut clipping
      » Look up table
  - Second part of the algorithm \( O(N) \)
    - Spatial filtering
      » Constant filter size

Complexity of the proposed algorithm (the size of the filter is kept constant) = \( O(N) \)

Complexity of multilevel algorithm = \( O(N \log (N)^2) \)
Conclusion

- Proposed method
  - Spatial GMA performed with a low computational complexity
  - Well performing for large images with constantly sized 33x33 kernel
  - Applicable to tone mapping of high dynamic images