Color-Space CAD

Neophytos Neophytou
Center for Visual Computing,
Stony Brook University
nneophyt@cs.sunysb.edu

Klaus Mueller
Center for Visual Computing,
Stony Brook University
mueller@cs.sunysb.edu

1. Introduction

In this sketch we propose a new paradigm for interactive image processing, which represents and manipulates the range of colors in an image region as a convex 3D object in the color-space domain. For this purpose, we use a transformation that converts the color-gamut of an image region into a possibly entirely different target gamut and define a toolset that enables the user to interactively "sculpt" the image's color-space. These capabilities are made accessible in ***Color-Space CAD***, a Adobe PhotoshopCS^{TM} plug-in, which provides a 3D design interface where users manipulate the color-space by directly operating on the color gamut of an image region with a set of CAD-like operators. Finally, with interactivity being key for any creative artistic process to take place, we make ample use of graphics hardware to accelerate all of the compute-intensive mapping operations.

Gamut correction techniques have been previously used in colorimetry. An overview is provided in [Braun and Fairchild 1999]. Another application related to our work is Primatte [Imagica 2006], which uses a 3D solid representation of the RGB color-space to achieve background replacement.

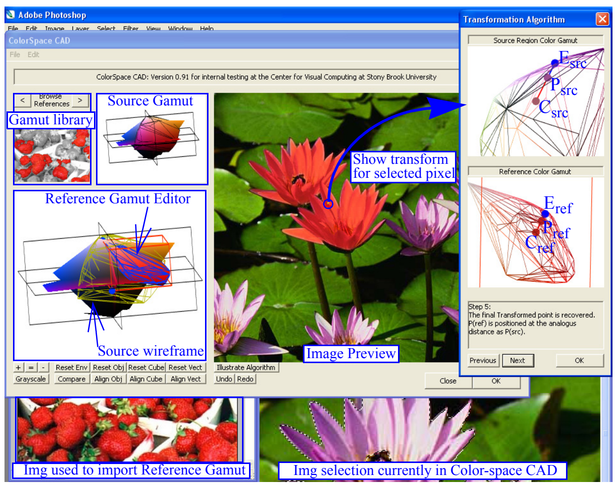

Figure 1: *Annotated screenshot of* ***Color-Space CAD*** *in action. In the above scenario, the strawberry image is first imported into the gamut library. We then select the two flowers in the water-lilly image and call the Color-Space CAD plug-in. After selecting an imported image by browsing the library (top left), we edit the imported gamut in the Reference Gamut Editor. Finally, the window in the top right illustrates the transformation process for a selected pixel.*

2. Method

The ***Color Range*** of a particular region within the image is found by first converting the colors of all pixels in that region into points in the L*a*b* space, followed by the creation of their enclosing 3D convex hull. It has been shown that the spatial relationships between different colors in the L*a*b* color-space are more intuitive to human observers, and therefore, manipulations on the color objects in this space are quite straight-forward to the user. The ***Transformation process,*** also used in [Chang et al. 2003] converts all the colors of the source color gamut such that they will fit as a group into the target (or reference) color gamut. For every point P_{src} we intersect the ray passing through the center C_{src} of the source gamut, with the edge at point E_{src}. The converted point P_{ref} is then located inside the target gamut along the same direction-ray passing through the center C_{ref} of the target gamut at the analogous distance between the center and the intersection E_{ref} with the gamut edge, as shown in Fig. 1 (top right).

3. Implementation and Features

The above method requires considerable amounts of computation, especially for traversing and finding the intersection points E_{src} and E_{ref} on the edges of the source and reference gamuts. We have adapted the transformation process to a GPU hardware accelerated algorithm that yields about two orders of magnitude of speedup over a cache-optimized software implementation. The accelerated method uses distance cube-maps to encode the color-gamuts that were initially represented as polygonal hulls. Thus, all the ray-polyhedra intersections required by the transformation process for every image pixel are reduced to simple cube-map lookups. This enables interactive color-space manipulation even on NVidia GeforceFX processors.

The Color-Space CAD plug-in includes components to visualize and manipulate the Color ranges as solid shapes in L*a*b* space, and an image preview window to provide real-time feedback. It is invoked by Photoshop after the user defines the selection using advanced tools such as color-range, magnetic lasso, etc., that result into a selection mask with anti-aliased boundaries.

The plug-in's interface provides a very simple way to transfer colors between different images by using the ***Gamut Library*** feature (top left in Fig. 1) or to directly edit the selected image gamut. ***Basic editing support*** includes *translation* (to shift along luminance or chroma direction), *resizing* (to enhance or reduce dynamic range along arbitrary directions), and *rotation* (to better align gamuts of different shapes or orientations). As sometimes color transfers between arbitrary images might cause quantization artifacts, especially when the two gamuts differ significantly in shape or orientation, the ***Automatic Gamut alignment*** function performs Principal Component Analysis (PCA) on both gamuts and rotates them to match their orientations.

Acknowledgements

We would like to thank Odaly Cruz for her artistic input, and Adobe Systems Inc. for providing us the necessary developer's SDK. This research was supported, in part, by NIH grant 5R21EB004099-02 and NSF-CAREER grant ACI-0093157.

References

BRAUN, G. J., AND FAIRCHILD, M. D. 1999. General purpose gamut mapping algorithms: Evaluation of contrast preserving rescaling functions for color gamut mapping. *In IS&T/SID Color Imaging Conference*, pp. 167–172.

CHANG, Y., SAITO, S., AND NAKAJIMA, M., 2003. A framework for transfer colors based on the basic color categories. *In proc. Computer Graphics International 2003, Tokyo, Japan,* 176 – 181.

IMAGICA CORP. OF AMERICA, 2006. Primatte White Papers. Webpage: http://primatte.com/p_whitepapers.cfm