Preserving gloss for on-screen texture mapping

Z. Wang, G. Hong* and M. R. Luo
Department of Colour and Polymer Chemistry, University of Leeds,
Leeds LS2 9JT, United Kingdom
**Applied Vision Research Institute, University of Derby,*
Derby DE22 3HL, United Kingdom
Corresponding author: Zhaohui Wang (*ccdzhw@leeds.ac.uk*)

ABSTRACT

The aim of this work is to simulate glossy objects on a CRT display. To investigate the effect, we captured gloss on the surface of satin fabric and plastic button samples with vivid colours and then separated gloss off the surface using the approach introduced in this study. To develop algorithms for gloss simulation, the specifications of gloss in different colour spaces were analysed. Two texture mapping algorithms were developed. To evaluate the performance of different algorithms, a psychophysical research method was used to evaluate the algorithms from previous work and that from this study. It was found that the two new algorithms developed which are capable of achieving better performance at preserving gloss appearance on the surface of objects.

1. INTRODUCTION

In recent years, with the advance in computing and imaging technologies, texture mapping has become a very efficient and convenient technique for generating computer graphics and simulated images. It is also commonly used by many commercial companies to visually evaluate colour samples mapped with a given texture pattern so that designers can use existing designs as references or templates for new designs.

With the recent success of colour communication using calibrated monitors, digital images are becoming more and more widely used for accurate colour communication and realistic product simulation via computer networks. In order to generate realistic colour texture for product simulation, texture mapping has to add various surface appearance characteristics and texture onto a uniform colour surface.

A generic texture mapping algorithm has been recommended by Han *et al* [1, 2] and is given in equation 1. This requires the input of a set of tristimulus values to define the desired colour together with a pre-calculated texture profile that defines the colour variation for each pixel as a measure of texture.

$$D_{Bi} = f\left(D_{Ai}\right) = tD_{Ai} + b \tag{1}$$

where b is the intrinsic colour value of a texture image, D_{Bi} indicates the output colour value for pixel i in a simulated image and D_{Ai} represents the texture profile for pixel i. The profile includes each pixel's colour variation in one or more colour channels and is pre-calculated from a chosen coloured texture image called the template image. In equation 1, t is a function modelling the statistical characteristics of a colour channel variation against different colours.

This algorithm was found to perform somewhat disappointingly in preserving the gloss effect on the surface of objects. In order to retain gloss, the influence of gloss on the textured surface was investigated. Simply, observers are able to see an object because of the way that light is reflected off the object. Objects stop incident light and reflect part of them, according to the surface characteristics of material. A glossy object typically shows a bright highlight in the mirror-reflection direction and this highlight appears to share the colour of the illuminating light (usually white). In the mean time, other areas on the object exhibit colours that are inherently associated with the bulk material properties of the object. Moreover, when the viewing angle approaches the glazing condition, highlight intensity increases dramatically and the overall colour of the object appears substantially de-saturated. Based on the investigation of the gloss on the surface of samples, two texture mapping algorithms were developed in our study.

2. METHOD

We studied two sets of samples, ten satin fabric and five plastic buttons. The satin fabrics were deliberately folded to enhance the gloss effect. These samples were digitised by a digital camera at five different angles (5°, 10°, 20°, 30° and 45°). This was achieved using mirrors to introduce the light at different angles as shown in Figure 1. Each satin sample was captured at a resolution of 600×600 pixels. Five buttons were captured at a resolution of 72×72 pixels. The RGB values of the images were converted to tristimulus values using a Gretag Macbeth colour chart including 240 colours as both training and testing sets, which gave a predictive error with an average ΔE^*_{ab} of 1.20 in CIELAB.

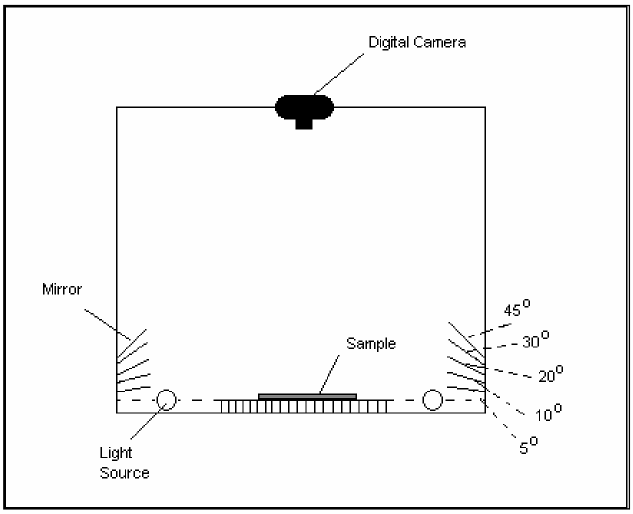

Figure 1: *setup of the image capturing system*

In order to separate gloss and shadow, a threshold function was developed from the image's mean and standard deviation. Standard deviation is used to indicate the spread of the variation of the data. By plotting the digitised samples, it was found that the gloss tends to be bright and de-saturated. These data are clustered above the image's mean values. In contrast, shadows tend to be dark and clustered below the image's mean values. The threshold was determined by analysing the gloss and shadow areas. One threshold function was developed suitable for all the samples to separate gloss and shadow and is given in equation 2.

$$
\begin{aligned}
T_{highlight} &= \overline{X} + 0.5STD \\
T_{shadow} &= \overline{X} - 0.1STD
\end{aligned} \tag{2}
$$

where $\overline{X}$ is the image's mean value and *STD* represents the standard deviation of the image.

By analysing the glossy areas on the image, the problems of the Han *et al*'s algorithm were found to be mainly due to the specification of colour in terms of tristimulus values and chromaticity coordinates. Firstly, the specification of tristimulus value is not easily interpreted in terms of human colour perception. (It uses, for example, lightness, colourfulness and hue.) Secondly, the XYZ system and the associated chromaticity coordinates are not perceptually uniform, which make it difficult to estimate the magnitude of the difference between two colour stimuli.

Investigating samples in the CIELAB colour space provides information suitable for new approaches to preserving gloss in the simulation of texture on screen. One approach is based on the variation of the L* channel (which is an extremely intuitive scale). An alternative approach was developed especially for button samples by using the threshold function to separate gloss and shadow areas when the performance would otherwise be unsatisfactory.

Two L* approaches were developed for satin fabric and button samples. These are given in equations 3 and 4,

$$
\begin{aligned}
L^*_n &= \overline{L^*} + \Delta L^* \\
a^*_n &= \overline{a^*} \\
b^*_n &= \overline{b^*}
\end{aligned} \tag{3}
$$

$$L_n^* = \overline{L^*} + \overline{L_{highlight}^*} + \overline{L_{shadow}^*} + \Delta L_{mean}^*$$
$$a_n^* = \overline{a^*} \tag{4}$$
$$b_n^* = \overline{b^*}$$

In equations 3 and 4, $\overline{L^*}$ represents the mean values of lightness of the image in question; a^* and b^* are mean values of the image in question. In equation 3, ΔL^* represents the texture profile which calculated from a standard image. In equation 4, $\overline{L_{highlight}^*}$ and $\overline{L_{shadow}^*}$ correspond to gloss and shadow areas respectively, which calculated for the whole image using threshold function. ΔL_{mean}^* represents the texture profile of the middle part, which is the area remaining after removing the gloss and shadow areas from the image.

3. RESULTS

A pair comparison method was used to evaluate the results. The purpose of this experiment was to investigate the different performance between the previous algorithms (derived from CIE XYZ tristimulus values (A1), xy chromaticity (A2) and the RGB colour space (A3)) and two new algorithms in CIELAB (A4, A5, which correspond to equations 3 and 4 respectively) in terms of colour fidelity, gloss, shadow and overall image quality (shown in Figures 2 and 3).

Nine satin samples and four button samples were chosen as physical samples in this experiment, while two samples were used to build the template profile. The simulated texture images were displayed on a calibrated HP p1100 monitor using the same size as the physical samples. The physical samples were sequentially placed in a viewing cabinet under a D65 simulator. The screen background was set to the same colour as that of viewing cabinet. Five images were simulated corresponding to five algorithms separately for each input value. In total, 45 (9 × 5) simulated images of satin samples and 20 (4 × 5) simulated images of buttons were displayed in a random order for each observer. Ten normal colour vision observers were used in this experiment. Category judgement was adopted in this experiment. Each observer was asked to do the judgement twice. In total 1300 (10 × 2 × (45 + 20)) judgements were made. Each observer was asked to sit in front of the apparatus and make their judgement between the simulated images and physical samples for four tasks (see later) by using 1 to 7 category points. On this scale, 1 is complete mismatch, 4 is an acceptable match and 7 is a perfect match. Assessments were made in a dark room without ambient light.

The judgements given by all observers for the simulated images against their corresponding physical stimuli were averaged. The results are given in Tables 1 and 2 for fabric and button samples respectively in terms of colour fidelity, gloss quality, shadow quality and image quality. A larger magnitude result indicates higher colour fidelity or improved gloss quality, shadow quality and overall quality of the simulated images.

	A1	A2	A3	A4	A5
Colour Fidelity	3.8	3.4	4.3	4.2	4.2
Gloss Quality	3.5	3.2	4.1	4.1	4.0
Shadow Quality	3.4	2.9	3.9	3.9	3.8
Image Quality	3.7	3.3	4.4	4.0	4.0

***Table 1:** performance of five mapping algorithms for satin samples*

	A1	A2	A3	A4	A5
Colour Fidelity	3.0	2.6	3.9	3.9	3.9
Gloss Quality	2.5	2.8	4.1	4.0	4.1
Shadow Quality	2.8	2.8	4.0	3.9	4.0
Image Quality	2.9	2.6	4.3	3.6	3.9

***Table 2:** performance of five mapping algorithms for button samples*

From Tables 1 and 2, it can be seen that the algorithm derived from the RGB colour space could yield better performance on image quality, which has an acceptable result for both samples. For colour fidelity and gloss quality, the algorithms derived from CIELAB and RGB could have results that are acceptable on average for the two samples sets. The algorithms A1 and A2 that were previously derived gave values less than 4.0 (acceptable match) in all cases. This corresponds to a somewhat poor performance in preserving the gloss effect on the surface of objects.

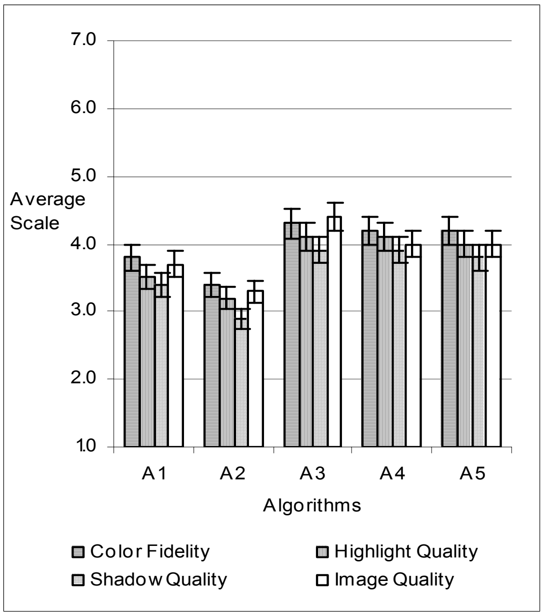

Figure 2: *performance of the five algorithms for satin samples*

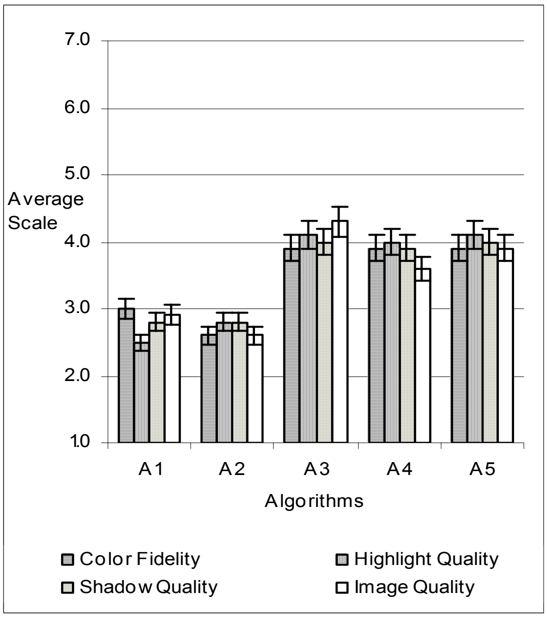

Figure 3: *performance of the five algorithms for button samples*

4. CONCLUSIONS

Two algorithms were developed for preserving gloss for on-screen texture mapping. The algorithms were compared with previous work by psychophysical experiments. The results are summarised as follows: 1) gloss appearance can be preserved by mapping a single set of CIELAB values onto a texture profile specified by the colour distribution in the L* channel; 2) the algorithm based on the RGB colour space could give acceptable results using four terms; 3) the proposed algorithms based on CIELAB improve performance obviously, which could give improved results in the four evaluation terms, especially in gloss quality.

References

1. G. Hong, M. R. Luo and B. Han, "Colorimetric Characterization of Low-end Digital Camera and Its Application for On-screen Texture Visualization," Proceedings of International Conference on Image Processing, Vancouver, Sept. 2000, pp. 741-744.
2. B. Han, G. Hong, G. Cui and M. R. Luo, "Evaluation of Texture Mapping Algorithms," CGIV 2004: The Second European Conference on Colour Graphics, Imaging and Vision, Aachen, Germany, April, 2004, pp.332-336.
3. K. Schluns and A. Koschan, "Global and Local Highlight Analysis in Colour Images," Presented at Proc. 1st International Conference on Colour in Graphics and Image Processing CGIP 2000, Saint-Etienne, France, 2000.
4. G. J. Klinder, S. A. Shafer and T. Kanade, "The measurement of Highlights in Colour Images," International Journal of Computer Vision, Vol. 2, No. 1, 1992.