Apparel Products Simulation Using Texture Mapping with Color Fidelity

J.H. Xin, H.L. Shen and P. Wang
Institute of Textiles & Clothing,
The Hong Kong Polytechnic University,
Hung Hom, Kowloon, Hong Kong

Abstract

In this paper, a new technique to simulate apparel products is introduced. The technique includes two issues: texture mapping and color fidelity. A texture grid is generated interactively to map the pixels in the target area onto the texture space. Through this grid, user can subtly twist and stretch the texture to achieve realistic effects. As the proposed texture mapping technique is image based, the computation complication and the mapping distortion caused by three-dimensional (3D) models can be avoided. We employ the dichromatic model to achieve the color fidelity. The color signals in the target area are divided into two linear components according to the model. Then only one component is substituted by the new color specified by users while the other one is held. This substitution technique is accomplished automatically and can keep the illuminant distribution on the target area unchanged.

Introduction

Simulating apparel products is a very useful technique. It can be wildly used in many applications such as virtual exhibition, internet-purchase and computer aid design. The key problem in the course of simulation is how to generate the realistic 3D effects. It involves two issues: texture mapping and color fidelity.

Texture mapping is a technique commonly used in computer graphics to enhance the realistic effects of computer-synthesized images.[1,2] It generally focuses on building a map from a two-dimensional (2D) image (texture space) to the surface of a 3D mode so that each point on the surface is associated with a corresponding pixel in the texture space. Traditional algorithms use an intermediate surface with simple shape that can be efficiently mapped to the texture space, and then find a map from the surface of a 3D model.[5] These algorithms often result in high distortion due to the non-linearity of the texture map.[6] Therefore, many optimization algorithms have been proposed to improve the mapping so as to obtain natural mapping results.[6-9] However, many of those algorithms need the 3D representation of objects. Therefore, the potential use of those texture mapping techniques has been greatly limited.

The color fidelity is another important issue in apparel products simulation. In computer vision, the dichromatic reflection model has been widely used to describe the complicated interaction between light and surface for inhomogeneous dielectrics materials.[3,4] In this model, the light reflected from a surface is decomposed into physically different interface and body reflections. The body reflection represents the intrinsic color of the object, while the interface one, which models highlights, usually has the same spectral power distribution as the illuminant. One of the most important properties of the dichromatic reflection model is that the geometric factor on illumination and viewing and the spectral factor on interface and body reflections are separable. Therefore, we can hold the brightness distribution of the target area by substituting the body part only.

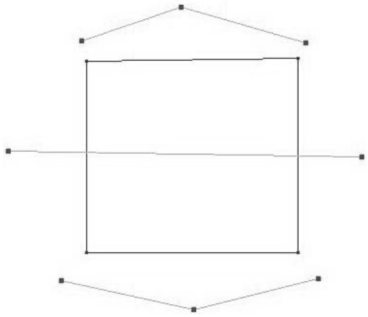
Texture Mapping

A texture grid is employed in the mapping course. The grid is build interactively and every row and column of the gird represents the texture directions. First, some polygonal lines are provided by users to indicate the texture directions in the target area. These polygonal lines are called the frame lines and are divided into two parts: horizontal and vertical ones. Then these frame lines are processed to construct a closed polygon and for each line, its two vertexes are on the polygon's border of the polygon. The texture gird is built based on the processed frame lines. We calculate the gradual frame lines based on the given horizontal and vertical frame lines respectively. These lines should change gradually from one given line to another. Then these gradual lines construct virtual horizontal and vertical texture grids. At last, these two virtual grids are combined to generate the final texture gird.

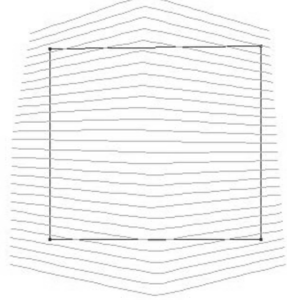
To calculate the gradual frame lines using interpolation, the texture frame lines must be parameterized. Let L_i denote one gradual frame line, the discrete parametric representation of L_i can be defined as:

$$L_i = \begin{cases} X_i(t) \\ Y_i(t) \end{cases} \tag{1}$$

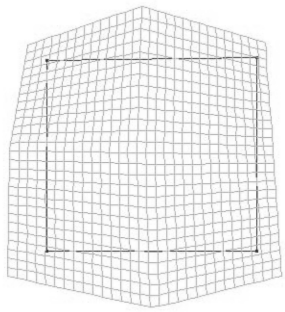
The value of t is defined as the ratio of the length from the beginning vertex to the current one and the total length of the line. $X_i(t)$ and $Y_i(t)$ are the coordinate of the vertex. For L_i, a set of t is calculated. We calculate the union of the parameter sets of all the input lines and resample these lines using it.



(a) Texture frame lines



(b) Gradual lines



(c) Virtual grid

Figure 1. Building horizontal virtual texture grid.

As the methods of generating these two grids are similar, we only discuss the horizontal one in this section. Given a line L_i, we take *y* coordinate of the center point of its bounding box as the weighting factor w_i. A series of weightings of the gradual lines, w_{g1}, w_{g2}, …, w_{gm}, is calculated by dividing the interval $[w_1, w_n]$ equally, where *m* is the number of the gradual lines. Using the weighting w_{gj}, the gradual line's parameter representation is given as:

$$X_j(t) = \frac{\sum_{i=1}^{n} \frac{X_i(t)}{|w_i - w_{gj}|}}{\sum_{i=1}^{n} \frac{1}{|w_i - w_{gj}|}}, Y_j(t) = \frac{\sum_{i=1}^{n} \frac{Y_i(t)}{|w_i - w_{gj}|}}{\sum_{i=1}^{n} \frac{1}{|w_i - w_{gj}|}} \tag{2}$$

where $X_i(t)$ and $Y_i(t)$ are the parameter representations of input frame line L_i; and *n* is the number of the input frame lines. The calculated gradual frame lines are shown in figure 1 (b).

As the weightings of the gradual lines are uniform, we use a uniform parameter set, $\{i/(N-1) \mid i=0, 1, \ldots (N-1)\}$, where *N* is the number of columns, to resample each gradual line. Finally, all the vertexes of these frame lines are used to construct the virtual texture grid as shown in figure 1 (c).

The final texture grid can be simply built by their weighted combination of the two virtual horizontal and vertical texture grids. Given a vertex *V* of the resultant grid, its counterparts of horizontal and vertical virtual grids are denoted as V_h and V_v, respectively. The coordinate of *V* is calculated as follows:

$$V = \frac{K_h \times V_h + K_v \times V_v}{K_h + K_v} \tag{3}$$

where K_h and K_v are weighting factors. In the calculation of the reference grid, their values are both 0.5. In the grid adjustment, their values should be calculated based on the characteristics of frame lines. If L_i is a horizontal frame line, we take the horizontal middle line of its bounding box as the base line. We calculate the distances between those vertexes and the corresponding base lines, and add these distances up to C_i. The value of C_i may reflect the characteristic of frame line L_i. For all the initial horizontal and vertical frame lines, their averages, $\overline{C_h}$ and $\overline{C_v}$, and variances, $D(C_h)$ and $D(C_v)$, are further calculated. If $|\overline{C_h} - \overline{C_v}| > \min(\overline{C_h}, \overline{C_v})/2$, the two weightings K_h and K_v are calculated as:

$$K_h = \frac{\overline{C_h}}{\overline{C_h} + \overline{C_v}}, K_v = \frac{\overline{C_v}}{\overline{C_h} + \overline{C_v}} \tag{4}$$

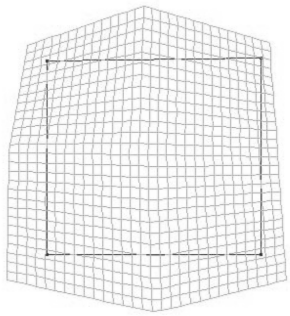
Otherwise, they become

$$K_h = \frac{D(C_h)}{D(C_h) + D(C_v)}, K_v = \frac{D(C_v)}{D(C_h) + D(C_v)} \tag{5}$$

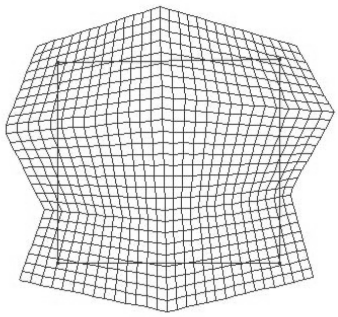
The combination process is shown in figure 2.

However, the generated grid may not be sufficient in reflecting the real texture directions. For instance, in the simulation of soft and flexible textile products, there are always many winkles that cannot be successfully represented by the generated texture grid. Therefore some operations are provided so that users can adjust the texture grid when necessary. In the technique developed in this study, users can adjust a single vertex, a line or a rectangle. Furthermore, the grid density is also adjustable so that users can perform texture mapping in different detail level.

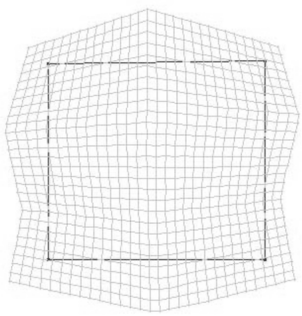
For a vertex at *i*th row and *j*th column, its texture coordinate is $y_t = i / (r-1)$, $x_t = j / (l-1)$, where *r* and *l* are the numbers of rows and columns respectively. For a pixel *P* in a quadrangle consisted by vertexes V_1, V_2, V_3, and V_4, we can calculate its texture coordinates by the bi-linear interpolation method using the coordinates of these four vertexes.



(a) Horizontal virtual texture grid



(b) Vertical virtual texture grid



(c) Final texture grid

Figure 2. Generation of texture grid.

Color Fidelity

To achieve the realistic effect in the simulation, the color fidelity is considered after texture mapping. In this study, we suppose the garment of the original image is monochromatic, and the spatial distribution of the illumination is fixed in the simulation.

In our technique, the dichromatic model is employed to accomplish color rendering. According to the dichromatic reflection model, the colors in the target area can be represented as a linear combination of two components: body and interface reflection. The color of the interface reflection is the same of the illuminant, while the color of the body reflection is the inherent characteristic of the garment. Therefore, the base idea in our color replacement method is to replace the body reflection component with the texture color and keep other parts unchanged in the rendering process. The resultant color $[R\ G\ B]^T$ (T represents vector transpose) is calculated by the linear combination of the new components:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \alpha \begin{bmatrix} R_b \\ G_b \\ B_b \end{bmatrix} + \beta \begin{bmatrix} R_I \\ G_I \\ B_I \end{bmatrix} \tag{6}$$

where $[R_b\ G_b\ B_b]^T$ and $[R_I\ G_I\ B_I]^T$ are the body color and the normalized illuminant color, respectively. For images recoded by a camera, the sensor output R, G and B can be balanced to remove the effect of the color temperature of the illuminant. Therefore, the normalized illuminant color vector can be defined as:

$$\begin{bmatrix} R_I \\ G_I \\ B_I \end{bmatrix} = \begin{bmatrix} 1/\sqrt{3} \\ 1/\sqrt{3} \\ 1/\sqrt{3} \end{bmatrix} \tag{7}$$

Then the SVD method is employed to decompose all the color signals into a 2D space, which is called the color signal plane. These dots on the plane belong to two linear clusters, according to the dichromatic model. One is the illuminant cluster and the other is the body cluster. We use the body cluster to estimate the body color vector. Substituting equation (6) and (7) into the definition of saturation channel of the HSV color space, we find that when the β value is nearer to zero, the corresponding saturation is higher. The dots which colors have largest saturation values are selected and fitted to a line on the color signal plane. The slope of the line is termed as K_b. The normalized body color vector can then be calculated as follows:

$$\vec{C}_b = \frac{V_x}{1+{K_b}^2} + \frac{K_b \times V_y}{1+{K_b}^2} \tag{8}$$

where V_x and V_y are the base vectors of the color signal plane. To decrease the fitting error, we select a range of the highest saturation values and divide it into several levels. Then the slopes corresponding to each level are calculated and the average of them is adopted as K_b.

The color $[R_I\ G_I\ B_I]^T$ and $\vec{C}_b$ are used to re-decompose all the color signals in the least square meaning. The biggest coordinate of the $\vec{C}_b$ direction is the norm of the body color. Therefore, coefficients in equation (6) can be calculated. We substitute the body color with the new color given by users while keep other components unchanged.

Simulation Examples

Some examples are shown in figure 3, 4 and 5. In figure 3, the twist and stretch of the texture are in consistent to the nature body shape and the characteristics of the cloth. The subtle changes of the texture directions in the target area are also well reflected. Figure 4 shows the color fidelity in the resultant image. The illuminant distributions in the substituted areas are the same with that in the original areas. In figure 5, the texture direction is smooth and well fitted with the natural shape of the collar.

Conclusion

In this paper we introduce a new technique to simulate apparel products. The technique includes two issues: texture mapping and color fidelity. The texture mapping is achieved by interactively generating a texture grid. Through carefully defining and adjusting the grid, user can subtly simulate the texture directions in the target area. Our texture mapping technique is image based. Therefore the computation complication is very low and it can be easily used in the internet-applications. We employ the dichromatic model to achieve the color substitution. The color signals in the target area are divided into two linear components named body color and illuminant color, according to the dichromatic model. Then the body color component is substituted by the new color given by users while the other component is held. This substitution technique is accomplished automatically and can achieve right color fidelity. In the future work, the accuracy of the estimation of the body color should be improved.

Acknowledgement

The authors wish to acknowledge the funding of this project from the Hong Kong Polytechnic University.

References

1. C. Bennis, J.M. Vezien, and G. Iglesias, Piecewise Flattening for Non-distorted Texture Mapping, Compute Graphics 25(4), 237 (1991).
2. P.S. Heckbert, Survey of Texture Mapping, IEEE Computer Graphics and Applications, 6(11), 56 (1986).
3. S.Shafer. Using color to separate reflection components. Color Research and Application, 10(4),210 (1985).
4. S. Tominaga, Dichromatic reflection models for rendering object surfaces, Journal of Imaging Science and Technology, 40(6), 549, (1996).
5. E. Bier, and K. Sloan, Two-part Texture Mapping, IEEE Computer Graphics and Applications, 6(9), 40 (1986).
6. B. Levy, and J.L. Mallet, Non-distortion Texture Mapping for Sheared Triangulated Meshes. Proceedings of ACM SIGGRAPH 98, pg.343. (1998).
7. Bruno Lévy, Constrained Texture Mapping for Polygonal Meshes. Proceedings of ACM SIGGRAPH'2001, pg.417. (2001).
8. A.Sheffer, and E. D. Sturler, Smoothing an Overlay Grid to Minimize Linear Distortion in Texture Mapping. ACM Transactions on Graphics, 21(4), 874 (2002).
9. H. Q. Sun, and H. J. Bao, Interactive Texture Mapping for Polygonal Models, Computational Geometry 15(1), 41 (2000).

Biography

John H. Xin graduated with PhD from The University of Leeds, U.K. in 1989. Thereafter, he joined multi-national textile company Coats Viyella, UK as a technologist in the color section of R&D department. He joined University of Derby, U.K. as a project coordinator for the development of a new generation computer color management and color quality control system in 1994. He joined Institute of Textiles and Clothing, the Hong Kong Polytechnic University in 1996 and is currently an associate professor. His research interests are in the areas of color management, digital color communication and reproduction, psychological aspect of color. He is a Charted Colorist, awarded by the Society of Dyers and Colorists, U.K.

(a) Original image

(b) Texture image

(c) Result image

Figure 3. First example for garment simulation

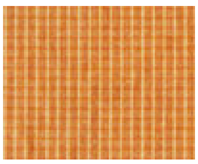
(a) Original image

(b) Texture images

(c) Simulation result

Figure 4. Second example for interior design

(a) Original image



(b) Texture image

(c) Simulation result

Figure 5. Third example for garment simulation.