
Automatic generation of bas-relief on 3D models based on 2D images for rhinoceros

Sha Liu*, Feng Yang, Shunxi Wang, Yu Chen

College of Engineering, China Agricultural University,
Beijing 100083, China

Lius02@cau.edu.cn

ABSTRACT. In view of the difficulty in generating bas-reliefs on 3D models with Rhinoceros, the most popular industrial design software in China, this paper puts forward a method to generate bas-relief on the surface of 3D models based on 2D images, and embeds the method into a Rhinoceros plug-in with the aid of RhinoScript. Through application verification, it is proved that the plug-in enables the automatic generation of base-relief from 2D images on Rhinoceros. The research findings provide an effective tool for industrial designers.

RÉSUMÉ. Compte tenu de la difficulté à générer des bas-reliefs sur des modèles 3D avec Rhinoceros, le logiciel de design industriel le plus répandu en Chine, cet article propose une méthode permettant de générer des bas-reliefs à la surface de modèles 3D basés sur des images 2D et intègre la méthode dans un plug-in Rhinocéros à l'aide de RhinoScript. Grâce à la vérification de l'application, il est prouvé que le plug-in permet la génération automatique de bas-relief à partir d'images 2D sur Rhinoceros. Les résultats de la recherche constituent un outil efficace pour les concepteurs industriels.

KEYWORDS: rhinoscript, bas-relief, 2D images, surface.

MOTS-CLÉS: rhinoscript, bas-relief, images 2D, surface.

DOI:10.3166/ISL23.6.99-113 © 2018 Lavoisier

1. Introduction

Bas-relief has been widely applied to the surface decoration and the product identification for a long time. Now more and more products will be designed and produced by digital ways. While if the bas-relief on product surface is made by Rhinoceros, SolidWorks or PCT Creo etc., the common design software for industrial designers, the processes are still very hard and fussy without the suitable ways to generate them automatically. Sometimes some patterns can't be finished by this software if they are too complex. So it is meaningful to simplify the way of designing and making bas-relief on the 3D models of products (He & Yang, 2010; Li *et al.*, 2012).

Cignoni *et al.* (1997), Policarpo, *et al.* (2005), and Sun *et al.* (2009) had discussed the algorithm of bas-relief generation. Cignoni *et al.* (1997) proposed a semi-automatic ways to generate bas-relief by a given 3D object. Following their researches. Zhao & Bian (2010) proposed a new algorithm for semi-automatic creation of bas-relief on curved surfaces of 3D mesh models by an image-based edge detection method. He & Yang, (2010) also proposed a method, which compresses a given 3D model and sticks it on a curved surface to get bas-relief. The researches have given the feasible theory and algorithm for digital generation of bas-relief (Weyrich *et al.*, 2007; Kerber *et al.*, 2009; You & Jiang, 2009).

Most bas-relief on the surface of product is designed based on 2D images or patterns. Now several CAM software had been designed to carve relief by numerical control machines, such as Artcam, Mastercam Art, Type3 and so on. By the CAM software, the gray level information will be extracted from the pixels of 2D images and to make the relief. On the other hand, in some popular 3D software, such as 3dmax, a function called "displace" has been designed to map 2D image on the mesh surface to make bas-relief, while the mesh surface cannot be used to product directly. Rhinoceros, the common software for product design, has not the function of generate bas-relief on the 3D model automatically, which makes it difficult to make patterns and logo on the surface of products, especially the curving surface. Based on the former algorithms, we try to realize a plug-in by using Rhinoceros further development which can automatically generate surface bas-relief by 2D images.

Now there are 4 ways for further development of Rhino 3D commonly: (1) Command Scripts. (2) RhinoScript which is a scripting language based on Microsoft VBScript. It is easy to learn and the development efficiency is very high. It can also easily connect to Excel and other databases. (3) Rhino C++ SDK. (4) Rhino.Net SDK. Both (3) and (4) can directly generate plug-in programs or dynamic link library for users to call in Rhino, but the developments are more difficult. Among them, RhinoScript is more suitable for the development of single auxiliary functions.

Some plug-ins for Rhinoceros had been developed to provide a variety of parametric modeling tools, such as Grasshopper for architectural design, RhinoGold for jewelry design and so on. While in China, researches on Rhino's further development is not so common and mature. Dai *et al.* (2004) proposed a way to implement some transformation of 2D sketches to 3D models on the design of shoes.

Pan & Jiang. (2007) developed a modeling system of Tiffany lamp-chimney based on Rhinoceros. Sun developed a ship digital design system based on Rhinoceros, Su & Liu (2010) realized a crystal light design plug-in for Rhinoceros; Liu *et al.* (2009), Wang Jin changed the simple 2D images to data to make the weaving household product by Rhino 3D. These researches provided useful explorations for digital design based on further-development of Rhinoceros, but the design of the bas-relief of product based on Rhino 3D has not yet been involved.

In the paper, we try to discuss a way of making bas-relief on the surface by 2D images. Firstly 2D images will be pretreated and changed to data saved in Excel database and the surface of products will be separated and redefined; Then the data of 2D images will be input and projected to 3D surfaces of products and make the relief. Finally based on the way, we develop a plug-in by VB and RhinoScript, and the effect of the plug-in will be tested by several 2D pictures (To & Sohn, 2016; Schüller *et al.*, 2014).

The whole research process is presented on Figure 1. We will focus on the following questions:

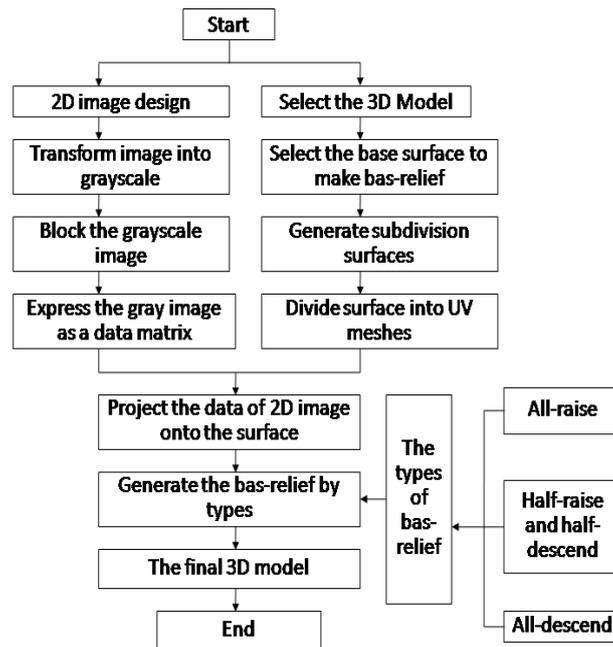


Figure1. The main technical route of the bas relief plug-in

(1) Pretreatment of the 2D images or patterns for bas-relief;

- (2) Classification of the 2D images or patterns and present them as data;
- (3) To get height field of bas-relief by grayscale-height mapping model;
- (4) Programming the bas-relief plug-in based on RhinoScript and testing its effect by several 2d images.

2. Pretreatment of the 2D images

2.1. Classification and pretreatment of 2D images or patterns

There are mainly three kinds of images or patterns used to generate bas-relief: binary image (text & vector graphics), grayscale image and color image. An image can be represented by a two-dimensional function $f(x, y)$, where x and y are the space (plane) coordinates, and the amplitude off at the point (x, y) is called the luminance of the image at this point. The coordinates and amplitudes of the points on an image need to be gotten if we want to convert it into digital format. Digitizing the coordinates is called sampling and digitizing the amplitude is called quantification.

After sampling and quantification, a real number matrix will be obtained. Suppose that an image $f(x, y)$ is sampled, the image will be represent as $M \times N$ and the values of the coordinates (x, y) are all discrete values. In order to make the notation simpler, the discrete coordinate values are represented by integers. Define the origin of the image in $(x, y) = (0, 0)$, then x is an integer from 0 to $M-1$, and Y is an integer from 0 to $N-1$. As shown in equation (1), the image can be expressed as a function $f(x, y)$:

$$f(x, y) = \begin{bmatrix} f(0,0) & f(0,1) & \cdots & f(0, N-1) \\ f(1,0) & f(1,1) & \cdots & f(1, N-1) \\ \vdots & \vdots & & \vdots \\ f(M-1,0) & f(M-1,1) & \cdots & f(M-1, N-1) \end{bmatrix} \quad (1)$$

The right side of the equation is the matrix of the digitized image and a single element in the matrix is a pixel.

(1) The binary image and grayscale image contain only luminance information. Y is its gray value and its range of value is:

$$Y_{min} \leq Y \leq Y_{max} \quad (2)$$

Theoretically Y is generally an integer and a finite value. The interval $[Y_{min}, Y_{max}]$ is called gray level. The common gray level is $[0, 255]$. When $Y_{min}=0$, it is black and when $Y_{max}=255$ it is white. Among them, all the intermediate values from black to white are gray and there is a total of 256 gray levels.

The function of a gray image whose size is $M \times N$ can be expressed as: $Y(x, y)$, Among it, $x=0, 1, 2, \dots, M-1$; $y=0, 1, 2, \dots, N-1$.

(2) Color image in computer often use RGB format. An RGB image is an array of $M \times N \times 3$ color pixels, where each color pixel can be regarded as the three components

of red, green and blue corresponding to the color image in a particular space position. Therefore, the RGB image can be seen as the stack of three gray images. C represents any vector in the space of RGB color:

$$\mathbf{c} = \begin{bmatrix} c_R \\ c_G \\ c_B \end{bmatrix} = \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

The formula shows C is the R,G,B components of a color image at one position point. At the same time, one color component is also a function of coordinates (x,y) :

$$\mathbf{c}(x, y) = \begin{bmatrix} c_R(x, y) \\ c_G(x, y) \\ c_B(x, y) \end{bmatrix} = \begin{bmatrix} R(x, y) \\ G(x, y) \\ B(x, y) \end{bmatrix}$$

(3) In order to generate bas-relief, we transform the color image into the gray image, which can simplify the expression of the image and reduce the data storage. This transformation is realized by formula (3)

$$Y = R \times 0.299 + G \times 0.587 + B \times 0.114 \quad (3)$$

After being transformed into gray image, the color image of $M \times N$ can be expressed as:

$$f(x, y) = Y(x, y)$$

Among it, $x=0, 1, 2, \dots, M-1$ and $y=0, 1, 2, \dots, N-1$.

2.2. Blocking image

In order to simplify the expression of the image and reduce the storage space, according to the actual design requirements, the gray image will be divided into $E \times F$ blocks and the average gray value $aY(i,j)$ of each block is extracted. Among them, $i=0, 1, 2, \dots, E-1, j=0, 1, 2, \dots, F-1$. Then the further simplified expression of matrix $h(i,j)$ will be obtained:

$$h(i, j) = \begin{bmatrix} aY(0,0) & aY(0,1) & \cdots & aY(0, F-1) \\ aY(1,0) & aY(1,1) & \cdots & aY(1, F-1) \\ \vdots & \vdots & \cdots & \vdots \\ aY(E-1,0) & aY(E-1,1) & \cdots & aY(E-1, F-1) \end{bmatrix} \quad (4)$$

At the same time, the maximum average gray value aY_{max} and the minimum average gray value aY_{min} of the $E \times F$ blocks will be obtained.

2.3. Data representation of different types of 2D images

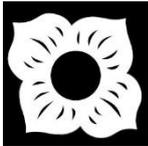
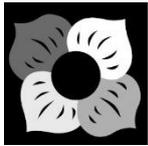
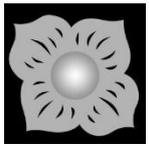
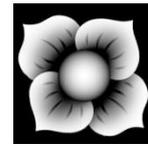
The composition of a relief is mainly influenced by light, outline, start position and perspective. The visual image of the relief mainly depends on the light and shade of the concave and convex shape of the body when the light shines on it.

Firstly the space sense of the relief is based on the light and shade of the shape. Generally speaking, the bright ones are in front and the dark ones behind. Secondly the contour line of the relief is usually represented by the start position. Start position, which is also known as "start outline", refers to the minimum starting position of the relief. For different shapes on the front and back positions, the space distance between the two shapes are distinguished through the level of the start positions. Thirdly the relief should express both the perspective of the space and the form. For one thing it should obey the perspective laws in painting: (1) the object is big when near and small when far; (2) the object is clear when near and vague when far. For another thing it also has two special laws: *a*. The object is thick when near and thin when far; *b*. The object is high when near and low when far.

So based on the above laws, we determine the relief's contour line by the size and position of the image; control the levels of start position by the gray value of the edge of the image, and generate the relief by the grayscale filled in the image.

There are many kinds of images for surface relief. According to the grayscale of images, we can divide them into four categories: single grayscale, multi-grayscale, single gradient grayscale and multi-gradient grayscale (Tab.1). Black (gray value= 0) is used as the background of the image and the pixels whose gray value are bigger than 0 (The range of value is 0~255) are corresponds to the height of relief.

Table 1. The four categories of images for surface relief

Single grayscale	Multi-grayscale	Single gradient	Multi-gradient grayscale
			

(1) For the single grayscale graphic, the background (gray value=0) and the pattern (gray value> 0) can be directly distinguished to generate relief (Fig.2).

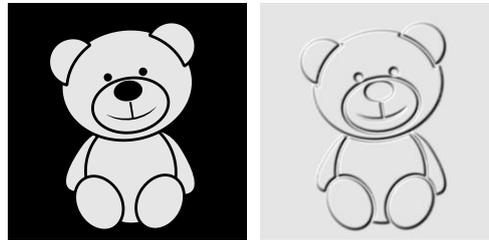


Figure 2. The single grayscale image (left) and the relief by the image(right)

(2) For simple geometry, the common gradient methods can be chosen according to their symmetry conditions to generate reliefs, such as linear gradient, radial gradient, angle gradient, symmetrical gradient, diamond gradient. The height of the relief is determined by the gray value of the graphic with gradient. Just as Tab.2, radial gradient is more suitable for circular patterns and diamond gradient is more suitable for square patterns.

Table 2. Different gray gradation and the corresponding bas-relief

Radial gradient		Diamond gradient	

(3) For free-form graphic, a step gradient can be used to generate bas-relief, which is to gradually change the gray scale in the vertical direction of each point from the outline of every single close figure. As Fig.3 shows, step gradient can be used to achieve the natural gradient effect of all the graphics with full outline and generate the bas-relief based on the grayscale.

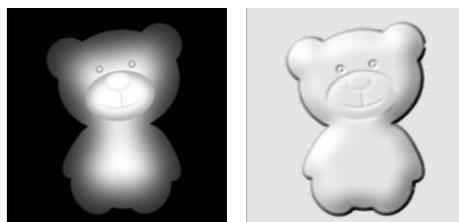


Figure 3. The step gradient and the bas-relief based on its grayscale

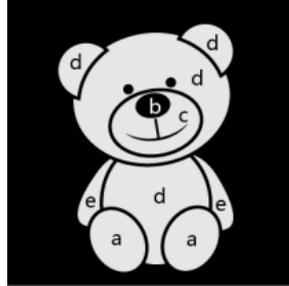


Figure 4. The image of a bear which is divided into 5 parts

(4) Complex graphics can be decomposed to several single graphic with full outlines and then turn them into step gradient by the sequence. As Fig. 4 shows, the pattern of bear can be divided into 5 levels by the spatial orders. From the front to the back, the levels of feet, nose, mouth, face, ear, body and arm can be identified as a, b, c, d, e respectively. Then, the range of the gray value of each part can be segmented according to the level. The gray value of the 0~255 range will be divided into five segments and the gray value of the five parts will be distributed as Tab.3. The appropriate gradient mode will be chosen to the corresponding part in the range of its gray value range and the bas-relief will be generated based on the grayscale (Fig. 5).

Table 3. The Range of gray value of each part

Parts	a	b	c	d	e
Gray value	204~255	153~204	102~153	51~102	0~51

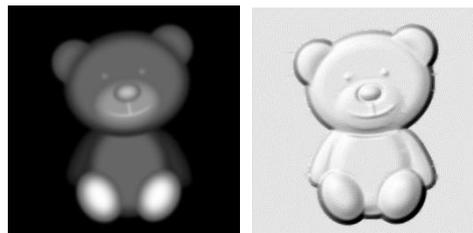


Figure 5. The gray image of the pattern and the bas-relief

3. Generating the bas-relief on the curved surface

3.1. The preprocessing of the base surfaces for bas-relief

The preprocessing of the base surfaces for bas-relief mainly includes 3 steps. (1) A rectangular area is created based on the size of the 2D image; (2) To project the rectangle area onto the surface of the 3D model; (3) To cut out the base surface of the bas-relief.

While the surface of the 3D model is usually uneven, it is difficult to control and generate the bas-relief. In order to realize the bas-relief or create better effect, it is necessary to subdivide the selected base surfaces. The procedure is as follows:

Firstly a certain number of uniformly distributed curves will be created in the U and V direction of the selected surface and a series of grids will be formed by the intersecting curves. There are uniformly distributed points on the grid. Then the coordinates of the intersection points are determined according to their orders in the directions of U and V. A new surface is generated by the coordinates of these points. Then the original surface will be deleted and the subdivided new surface will be created. The subdivided surface can be expressed as function (5):

$$\begin{cases} U = \{u_0, u_1, \dots, u_m\} \\ V = \{v_0, v_1, \dots, v_n\} \\ Pt = (u_i, v_j) \end{cases} \quad (5)$$

Among it, m is the number of subdivision segments in the U direction and n is the number of subdivision segments in the V direction; u, v are the coordinates of the subdivided points; Pt represents the node of the grid after subdivision (see Fig.6).

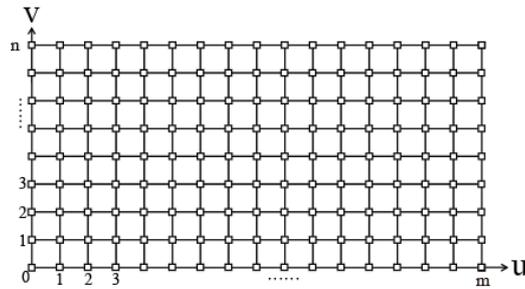


Figure 6. Subdividing a selected surface

3.2. To Project the image to the curved surface

(1) After the grayscale processing and image blocking, the 2D image can be expressed as the Eq. (4) mentioned above.

(2) Each node on the subdivide surface can be expressed as: $P(u,v)$

Among it, $u=0,1,2,\dots,F+1, v=0,1,2,\dots,E+1$.

$h(i,j)$ and $P(u,v)$ can be matched according to the following function:

$$P(j, E + 1 - i) = h(i, j)$$

Thus, the knots match from the 2D image to the 3D surface is achieved. The corresponding relation is shown as Fig.7: The area surrounded by white nodes is the 3D surface and the black nodes represent the knots which are projected from the 2D image to the 3D surface.

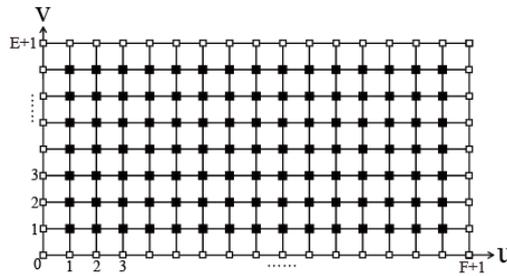


Figure 7. To project the 2D image to the curved surface

3.3. To generate the bas-relief by raising points

After the pretreatment of the 2D image, the maximum average gray value aY_{max} and the minimum average gray value aY_{min} of the image can be gotten. It is assumed that the proportion matrix of the gray value of the image is $h'(i,j)$ and its expression is as follows:

$$h'(i,j) = \frac{h(i,j)}{aY_{max} - aY_{min}} \tag{6}$$

The normal variations of the nodes projected on the 3D surface by the 2D image can be expressed as $V(\text{Column}, E+1-\text{ROW})$;

The maximum value of the height of the relief is H , which is input by users.

The equation, $y = \sqrt[n]{x}$ (n is an integer which is equal or greater than 1) is used to calculate the height of the knots which eventually form the 3D surface of the bas-relief. Since the equation can be used to adjust the change of the height of the bas-relief to change the smoothness of the bas-relief, n is called smoothness coefficient. Moreover, according to the difference among the raising ways, the formula used to calculate the height of the bas-relief includes 3 kinds:

- (1) All-raise bas-relief:

$$V(\text{Column} + 1, E + 1 - (\text{Row} + 1)) = H \times \sqrt[n]{h'(\text{Row}, \text{Column})}$$

(2) All-descend bas-relief

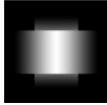
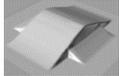
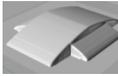
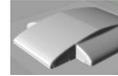
$$V(\text{Column} + 1, E + 1 - (\text{Row} + 1)) = -H \times \sqrt[n]{h'(\text{Row}, \text{Column})}$$

(3) Half-raise and half-descend bas-relief

$$V(\text{Column} + 1, E + 1 - (\text{Row} + 1)) = H \times t \times \sqrt[n]{(h'(\text{Row}, \text{Column}) - 0.5)}, (t = \pm 1)$$

As Tab.4 shows, if we turned a 2D image into an all-raise bas-relief, when $n=1$, $n=2$, $n=3$, $n=4$, the smoothness of the bas-relief is different. Fig.8 shows the contrast diagram of the contour curve of bas-relief generated by n takes 1~4. The larger n is, the more smooth the contour curve of the bas-relief will be, but the edge of the bas-relief will also be more sharp. Therefore, it is appropriate to take $n = 2\sim 3$.

Table 4. The 2D gray images and the all-raise bas-relief corresponding to different n

2D image	The all-raise bas-relief corresponding to different n			
	$n=1$	$n=2$	$n=3$	$n=4$
				

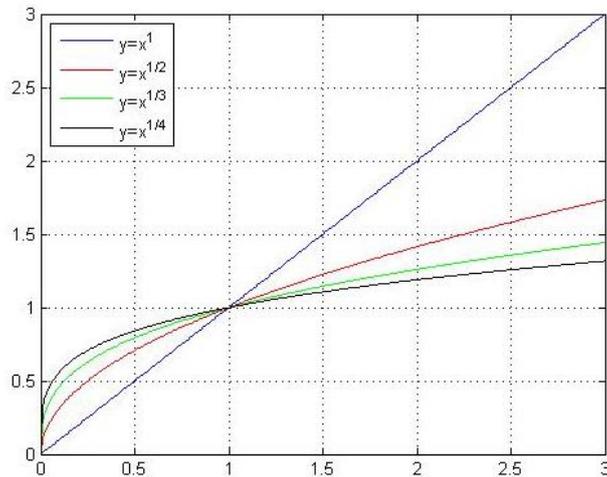


Figure 8. The contour curve of bas-relief $y = \sqrt[n]{x}$ ($n=1,2,3,4$)

4. Algorithm verification and plug-in development

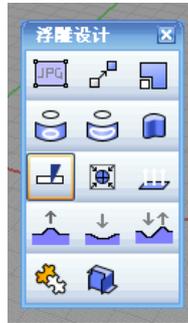


Figure 9. The interface of the plug-in

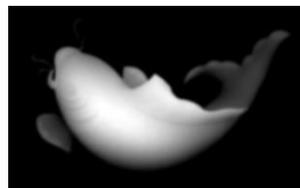
In order to verify the above algorithm, a Rhinoceros plug-in used to generate bas-relief on the surface of 3D model was developed by RhinoScript (Fig. 9). The work steps of the plug-in include:

(1) Open or create a 3D model of a product that needs to be added a bas-relief on a surface;

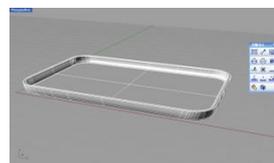
(2) Click  (Draw rectangle based on image size) and select the 2D image for generating the bas-relief. Select the starting point in the view of Rhinoceros to create the rectangle based on the image size. Move the rectangle to the suitable position to generate bas-relief.

(3) Project the rectangle to the surface of the 3D model and cut out the base surface for generating the relief and rebuild the surface to make U&V curves evenly distributed.

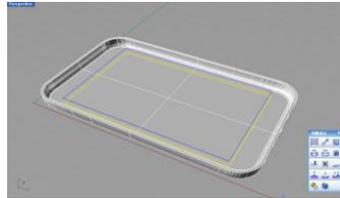
(4) Use  (All-raise bas-relief, All-descend bas-relief and half-raise and half- descend bas-relief) to choose the types. Then choose the Excel file of 2D images and select the base surface; Input the relative parameters, such as H (the maximum height) and n (the smoothness coefficient) and generate the bas-relief on the base surface. Fig. 10 shows the whole steps of how the plug-in works.



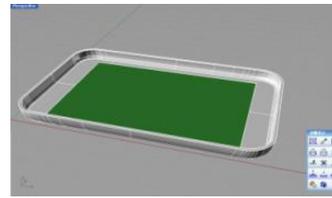
The 2D gray image



The 3D model to for generating the bas-relief



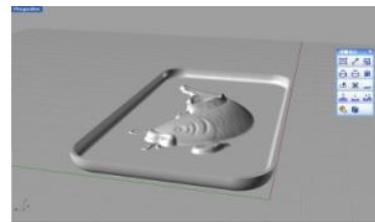
Create the rectangle by image size



Project the rectangle to the surface of 3Dmodel and cut out the base surface



Import the Excel file of 2D images and Input the relative parameters



The final bas-relief

Figure 10. The work steps of the plug-in

5. Conclusions

Aiming at reducing the difficulties of designing and creating the bas-relief, such as logos and decorative patterns on the surface of the products' 3D models by the common industrial design software, a method to generate bas-relief on the surface of 3D models based on 2D images is attempted.

The method is as follows: (1) According to the types of 2D images, the method of converting the image to a gray-scale image is selected. The image will be converted to gray image according to the basic graphic, the gradient ways and the spatial locations of graphics with full outlines. (2) After the gray image is blocked and the average gray value of each block is extracted, the value will be converted into a digital matrix and stored as a database in an Excel file. (3) The surface on the 3D model is selected to generate of the relief. Some uniformly distributed curves will be created in the U and V direction of the selected surface and the coordinates of the intersection points of the U, V curves are determined to generate a new surface which is the base surface for the bas-relief. (4) The knots of the image matrix will be matched to the knots on the base surface respectively. Then according to a formula, the nodes on the base surfaces will raise in the W direction of the base surfaces based on the gray value of the 2D image and the new surface of the 3D model will be generated by the nodes. (5) A variety of bas-relief can be achieved through different formula and the changes of the parameter in the formula. At last a Rhinoceros plug-in is achieved by the above method based on RhinoScript. The bas-relief can be generated easily and

automatically from a 2D image by the plug-in, which verified the possibility of the above method.

Acknowledgment

This research is supported by the National Key R & D program "R & D of Multifunction Cane and Beet Harvest Technology and Equipment" (2016YFD0701200), Sub-project "Research on Key Technologies of Efficient Harvesting and Intelligent control for Cane and Beet Efficient Harvesting Equipments" (2016YFD0701201) and "The research and development of the intelligent fining production management technology and equipment for livestock and poultry breeding" (2017YFD0701605-2).

References

- Cignoni P., Montani C., Scopigno R. (1997). Computer assisted generation of bas- and high-reliefs. *Journal of Graphics Tools*. Vol. 2, No. 3. pp. 15-28. <http://doi.org/10.1145/1053427.1053453>
- Dai Q. F., Yang H. X., Xu Y. Y. (2007). Design and implementation of two-dimension, three-dimension transformation for last with Rhino3D developer platform. *Control & Automation*, Vol. 28, No. 18. pp. 134-136.
- He H. Z., Yang C. N. (2010). Relief generation on surfaces. *Journal of Computer-Aided Design & Computer Graphics*, Vol. 22, No. 7. pp. 1132-1137.
- Kerber J., Belyaev A., Zayer R., Tevs A., Seidel H. P. (2009). Feature sensitive bas-relief generation. *Proceedings of IEEE International Conference on Shape Modeling and Applications*, Beijing, China. <http://doi.org/10.1109/SMI.2009.5170176>
- Lin J., Wang W. J., Guan D., Jing J. (2012). Realization of 3D embossment effect based on a relief image. *Microcomputer & Its Applications*, Vol. 31, No. 11. pp. 41-42. <http://doi.org/10.4028/www.scientific.net/AMM.229-231.1789>
- Liu X. J., Sun S. Q., Chen S. (2009). Image-based 3D pattern structure mapping of woven products. *Journal of Zhejiang University (Engineering Science)*, Vol. 43, No. 8. pp. 1367-1371. <http://doi.org/10.2514/6.1990-3963>
- Pan L., Jiang W. (2007). Modeling system of Tiffany lamp chimney based on Rhino. *Computer Engineering and Design*, Vol. 28, No. 8, pp. 4426-4428.
- Policarpo F., Oliveira M. M., Comba J. L. D. (2005). Real-time relief mapping on arbitrary polygonal surfaces. *Proceedings of the Symposium on Interactive 3D Graphics and Games, Washington D C*, pp. 155-162. <http://doi.org/10.1145/1053427.1053453>
- Schüller C., Panozzo D., Sorkine-Hornung O. (2014). Appearance-mimicking surfaces. *ACM Transactions on Graphics*, Vol. 33, No. 6. pp. 216:1-216:10 <http://doi.org/10.1145/2661229.2661267>
- Sun H., Liu X. J. (2010). Study on the rapid design techniques of crystal light. *IEEE 11th International Conference on Computer-Aided Industrial Design and Conceptual Design. CAID and CD'2010*, Vol. 1, pp. 797-799. <http://doi.org/10.1109/CAIDCD.2010.5681230>

- Sun X. F., Rosin P. L., Martin R. R., Langbein F. C. (2009). Bas-relief generation using adaptive histogram equalization. *IEEE Transactions on Visualization and Computer Graphics. Los Alamitos*, Vol. 15, No. 4, pp. 642-653. <http://doi.org/10.1109/TVCG.2009.21>
- To H. T., Sohn B. S. (2016). Bas-relief generation from face photograph based on facial feature enhancement. *Multimedia Tools and Applications*, Vol. 76, No. 8. <http://doi.org/10.1007/s11042-016-3924-y>
- Weyrich T., Deng J., Barnes C., Rusinkiewicz S., Finkelstein A. (2007). Digital bas-relief from 3D scenes. *ACM Transactions on Graphics*, Vol. 26, No. 3, pp. 32. <http://doi.org/10.1145/1276377.1276417>
- You J., Jiang X. F. (2009). The Study on the design technology of products surface relief. *Machine Tool & Hydraulics*, Vol. 37, No. 7, pp. 215-217. <http://doi.org/10.3969/j.issn.1001-3881.2009.07.071>
- Zhao P., Bian Z. (2010). The bas-relief on curved surface from 3D meshes. *Journal of Computer-Aided Design & Computer Graphics*, Vol. 22, No. 7. pp. 1126-1131.

