Automatic Generation of Stippling Illustrations from two Photographs

Germán Arroyo, Domingo Martín y M. Victoria Luzón

Departamento de Lenguajes y Sistemas Informáticos
Universidad de Granada, Spain

Resumen

En este artículo se presenta una herramienta que, a partir de dos fotografías, es capaz de reconstruir la profundidad de los objetos y obtener un modelo 2D y medio. Utilizando dicha información el sistema es capaz de diferenciar qué es fondo y qué un elemento de interés y puntearlos de forma distinta para generar la ilustración final.

La herramienta necesita una mínima interacción por parte del usuario para alinear ambas fotografías e indicarle el nivel de detalle a partir de cierta lejanía. El resto es decidido por el software que se encarga de generar la ilustración. Dicha ilustración es generada en pocos segundos por el sistema, mientras que un ilustrador profesional puede tardar aproximadamente 20 horas para realizar el mismo tipo de ilustración.

Palabras Clave: *VISUALIZACIÓN EXPRESIVA, PUNTEADO DE MONUMENTOS, INFORMÁTICA GRÁFICA, DOCUMENTACIÓN DE PIEZAS*

Abstract

In this paper we present a software tool that is able to reconstruct the depth field of the objects from two photographs and to obtain a pseudo 3D model. Using this information our system is able to difference background from foreground, and therefore, what are the interesting elements in the photographs and stipple them in different ways.

This tool needs almost no user interaction. The user simply has to align both photographs and indicate the level of detail according to the distance. The rest is decided by our software. Whereas a professional illustrator needs more than 20 hours to finish a similar illustration, our software is able to do it in just few seconds.

Key words: *NON-PHOTOREALISTIC RENDERING, STIPPLING MONUMENTS, COMPUTER GRAPHICS, DOCUMENTING PIECES*

1. Introduction

When documenting archaeological sites, a complex task of the process is the creation of technical illustrations. Illustrations allow the reader to imagine ancient sites and to add information about their elements (shape, imperfections, etc.) and maintaining the meaning of piece of art.

Stippling is the process of illustrating by means of drawing stipple dots on a paper. The technique is simple but tedious, mainly because the huge amount of stipple dots to be placed in every detail of the illustration.

Computer generated stippling illustrations are possible, however, it still requires an illustrator to choose what the important parts are or how to stipple the different regions.

Main problem of previous algorithms is that it is difficult to segment the image due to the lack of 3d information and without intensive user interaction.

In this paper, we propose to use two photographs as input of the software. Depth information is obtained with these photographs by obtaining the 3D Stereo Disparity from the images, which is often used in artificial vision and robotic vision.

The images are simply segmented and stippled using a dynamic matrix of probabilities (DMP) according to their depth information and the illumination discontinuities of one of the images.

2. Related Works

There are many works related with automatic stippling. We are specially interested in methods that produce stippled images from 2D images and photographs. Among these methods, Secord (SECORD, 2002), Deussen (DEUSSEN, 2000), Hiller (HILLER, 2003), Dalal (DALAL, 2006), Barla (BARLA, 2006) and (SECORD, 2002), use centroidal Voronoi diagrams (CVD) to distribute the dots. Their approach has problems with patterns. Schlechtweg (SCHLECHTWEG, 2005) uses a multi-agent system called Renderbots, to position different kinds of drawing primitives. The main problem is that in the borders of the tiles it produces errors that must be corrected and the dots are black circles or another geometrical shape. Mould (MOULD, 2007) creates a weighted graph that controls the location of dots, its results show the alignment of the dots on the edges, which is disturbing. Kim (KIM 2009) provides a method for extracting a tone map and dots from the original image, the method is

Virtual Archaeology Review

complex and it only works with black dots, being unrealistic. The work of Isenberg (ISENBERG, 2005) and the works of Maciejewski (MACIEJEWSKI 2008) show a guide of differences between automatic and hand-drawn stippling, which is useful to develop the new algorithms of stippling. The works of Martin et al. (MARTIN, 2011) and Arroyo et al. (ARROYO 2010) propose new methods that obtain good results but requires a lot of user interaction to prepare the photograph in order to be stippled and remove the background of the photograph. The process of removing the background is trivial with small pieces in prepared environments but is a really complex task when the photographs are taken in open spaces (like buildings or archaeological sites).

3. Construction of the Function of Probabilities

The system overview we propose takes a couple of images as input. These images are processed by using a graph cuts stereo correspondence algorithm (KZ1) appearing in Kolmogorov et al. (Kolmogorov 2003). This algorithm is applied to both images. Additionally our algorithm extracts information about edges and a version of the photograph with a normalized histogram. With this information, a statistical function is built and quantized. Finally, a set of dots are placed on the image according to the probabilistic function by means of an adaptive random search.

The KZ1 algorithm obtains an image of grey values, where the darker values means the object is farther, where lighter values indicates the object is closer to the observer.

Therefore, for every pixel of the photograph we obtain a value of distance from the observer, as shown in Figure 1.
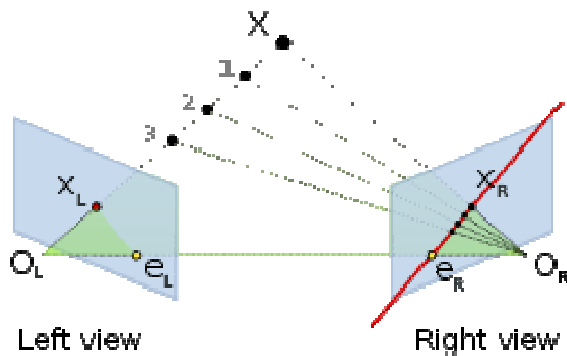

Fig. 1: Four real points projected in two images. Each image has its own point of view (O_L and O_R). e_L and e_R means the epipolar points in both images.

These values can be stored in a buffer and allow to reconstruct a half-3d image as shown in Figure 2.

According to the depth of the image the user can select the maximum level of depth when stippling, and the level of the detail for any part of the image.

In the case of stippling illustrations, it is typical that artists emphasize two kind of visual information:

- Edges

- Contrast: Shade and shadows

Edges are automatically computed from a Laplacian filter in our algorithm, which is a 2-D isotropic measure of the 2nd spatial derivative of an image, and allows to obtain edges of an image accurately.

Contrast is enhanced by using an adaptive normalized histogram equalization. Through this adjustment, the intensities of the image can be better distributed on the histogram. This allows for areas of lower local contrast to gain a higher contrast.

A probability function is created using these buffers (the Edge image, the Laplacian image, and the Depth image). The values of the function of probability depends on this formula:

$$F(x) = (L(x) \cdot g(y) + h(y) \cdot E(x))$$

where L is the Laplacian image, E is the Edges image, **g** and **h** are functions defined by the user, and **x** is the position of a pixel in the image:

$$x = (x_0, x_1).$$

y is the value of the intensity in the pixel of position **x**.

g and **h** are defined on the Depth image, in such a way that the values of **g** and **h** are defined by a piecewise-defined function. Therefore, for every value of the depth image there is a direct correspondence with a value sets by the user, as shown in Figure 3.

F is discretized and stored in a matrix of the same size as the original image, establishing a direct correspondence between a pixel and a value of probability. This matrix is called a dynamic matrix of probabilities (DMP).
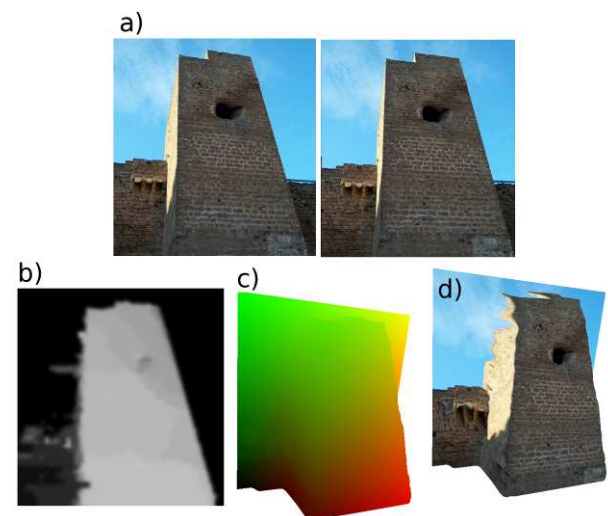

Fig. 2: a) Left and right image; b) Depth buffer; c) 3D elevation map from the depth buffer; d) 3D reconstruction of the model.

4. Algorithm of Stippling

The algorithm of stippling tries to simulate the stippling points placement when the artist is creating the illustration. Artists do not use an exact equation to place the dots on the paper, but they try to simulate the grey levels and the borders of the image avoiding patterns.

Therefore, a statistical approach is necessary. Statistical algorithms can make small mistakes, like artists when stippling. However, these algorithms obtain a quite accurate solution, just like artists with their illustrations.

We have design an algorithm that obtains randomly a set of positions. These positions are used to obtain the value of probability in the DMP. If the maximum probability is 0, no dot is placed, otherwise, a new dot is placed in the position of the maximum found probability, and the dot is blended to the rest of the placed stippling points in the canvas. The probability in that region of the DMP is decreased.

If the maximum probability is equal to or lesser than zero, the size of the set of positions is increased permanently, that allows to search farther for a better solution.

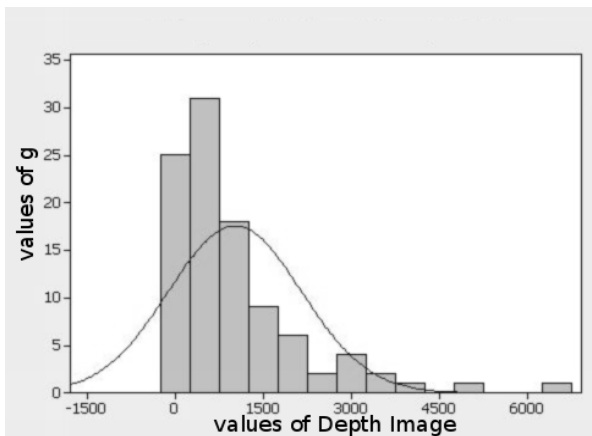The stippling dots are obtained from a library of scanned dots.

Fig. 3: Graph altered by the user. Horizontal values represents the values of depthness, whereas vertical values are adjusted by the user.
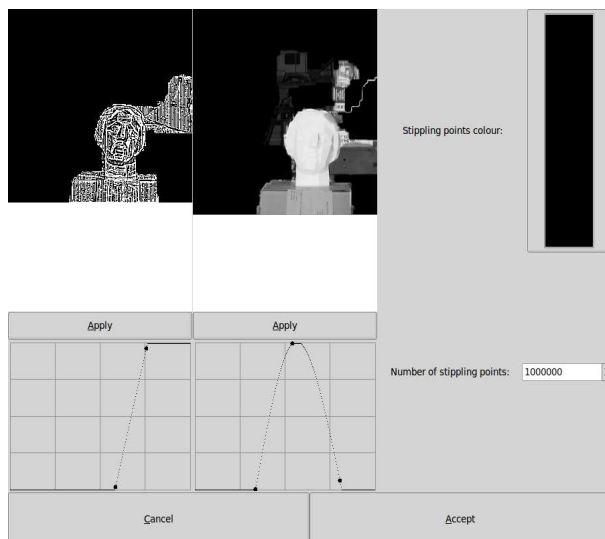

Fig. 4: Software to control the silhouettes and the contrast of the image according to the depth information.

5. Software and User Interface

The developed software is easy to use, the user load the image of the left eye and the image of the right eye (both can be obtained with the same camera), and then images are automatically corrected. Then the depth buffer is obtained and showed to the user, and two histogram functions allow to control the amount of silhouettes and the amount of shade and shadows in the image, as shown in Figure 4. Finally the illustration is automatically obtained.

6. Results

The developed software runs in the most used operative systems: MS Windows, Mac OS X Leopard, or Linux, and possibly others. The obtained depth buffer is used to segment the image and detect the distance of the object to the observer. Figure 5 shows the interface. Figures 6 and 7 show some results of the software.

7. Conclusions

In this paper, we have presented a program to generate automatic stippled illustrations. The algorithm is able to reconstruct the depth information of the scene by using two photographs. The developed software processes the image and the depth information and produces stippled illustrations in seconds.

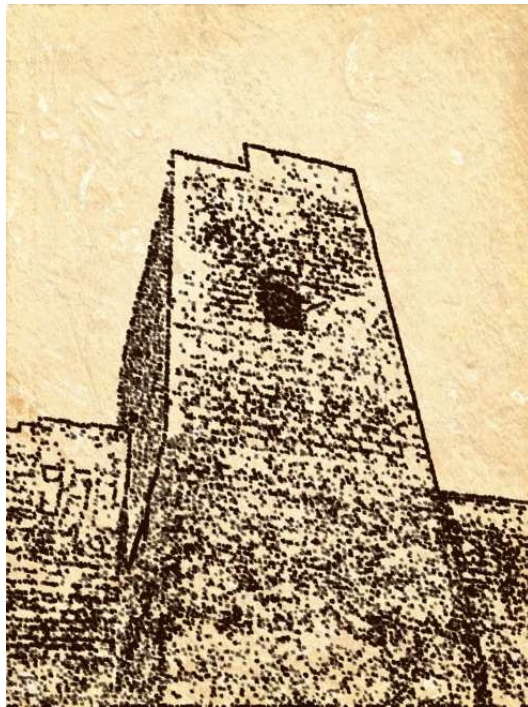*Fig. 5: Interface of the software when placing the image and obtaining the depth information.*

Fig. 6: Result of images in Figure 2. The chosen paper and ink are both sepia.

Fig. 7: Result of the algorithm on a lined paper.

Acknowledgments

References

BARLA, P. et al. (2006): "*Interactive hatching and stippling by example*". INRIA.

DALAL, K. et al. (2006): "A Spectral Approach to NPR Packing". In *Proc. of NPAR,* ACM, New York, pp. 71-78.

DEUSSEN O. et al. (2000): "Floating points: A method for computing stipple drawings". *Computer Graphics Forum* 19, pp. 40-51.

HILLER, S. et al. (2003): "Beyond Stippling - Methods for Distributing Objects on the Plane". Computer Graphics Forum 22, 3, September, pp. 515-522.

ISENBERG, T. et al. (2005): "Breaking the Pixel Barrier". In *Proc. of Cae, Eurographics Association*, Aire-la-Ville, Switzerland, pp. 41-48.

KIM, S. et al. (2009): "Stippling By Example". In *Proc. of NPAR,* ACM, New York, pp. 41-50.

MACIEJEWSKI R. et al. (2008): "Measuring Stipple Aesthetics in Hand-Drawn and Computer-Generated Images", *IEEE Computer Graphics and Applications*, pp. 62-74.

MOULD D. (2007): "Stipple placement using distance in a weighted graph". In *Proc. of Computational Aesthetics in Graphics*, no. 3. p. unknown.

SCHLECHTWEG S. et al. (2005): "Renderbots: Multi agent systems for direct image generation" *Computer Graphics Forum* 24, pp 283-290.

SECORD A. et al. (2002): "Fast primitive distribution for illustration". In *Thirteenth Eurographics Workshop on Rendering*, pp. 215-226.

SECORD, A. (2002): "Weighted voronoi stippling". In *Proc. of NPAR,* ACM Press, pp. 37-43.