ARTICLE IN PRESS

Available online at www.sciencedirect.com

SCIENCE DIRECT®

Graphical Models xxx (2005) xxx–xxx

Graphical Models

www.elsevier.com/locate/gmod

Three-dimensional shape rendering from multiple images

Alberto Bartesaghi [a,*], Guillermo Sapiro [a], Tom Malzbender [b], Dan Gelb [b]

[a] Electrical and Computer Engineering Department, University of Minnesota, Minneapolis, MN 55455, USA
[b] Hewlett-Packard Laboratories Palo Alto, CA 94304, USA

Received 14 May 2004; received in revised form 27 December 2004; accepted 28 February 2005

Abstract

A paradigm for automatic three-dimensional shape and geometry rendering from multiple images is introduced in this paper. In particular, non-photorealistic rendering (NPR) techniques in the style of pen-and-ink illustrations are addressed, while the underlying presented ideas can be used in other modalities, such as halftoning, as well. Existing NPR approaches can be categorized in two groups depending on the type of input they use: image based and object based. Using multiple images as input to the NPR scheme, we propose a novel hybrid model that simultaneously uses information from the image and object domains. The benefit not only comes from combining the features of each approach, it also minimizes the need for manual or user assisted tasks in extracting scene features and geometry, as employed in virtually all state-of-the-art NPR approaches. As particular examples we use input images from binocular stereo and multiple-light photometric stereo systems. From the image domain we extract the tonal information to be mimicked by the NPR synthesis algorithm, and from the object domain we extract the geometry, mainly principal directions, obtained from the image set without explicitly using 3D models, to convey shape to the drawings. We describe a particular implementation of such an hybrid system and present a number of automatically generated pen-and-ink style drawings. This work then shows how to use and extend well-

* Corresponding author. Fax: +1 612 625-4583.
E-mail addresses: abarte@ece.umn.edu (A. Bartesaghi), guille@ece.umn.edu (G. Sapiro), malzbend@hpl.hp.com (T. Malzbender), dgelb@hpl.hp.com (D. Gelb).

1524-0703/$ - see front matter
doi:10.1016/j.gmod.2005.02.002

developed techniques in computer vision to address fundamental problems in shape representation and rendering.

Keywords: Shape rendering; Non-photorealistic rendering; Image-based rendering; Geometry; Multiple images; Binocular stereo; Photometric stereo; Texture synthesis

1. Introduction

One of the strengths of non-photorealistic rendering (NPR) is that it can represent complex scenes using simple schematic drawings. A sketch-like picture obtained with NPR techniques emphasizes high level or salient perceptual features and at the same time effectively communicates shape and geometry. In producing computer generated NPR [11,37], principles of traditional drawing can be established and a number of these implemented as part of automatic rendering systems. From the rich variety of NPR depiction styles, we restrict the discussion to pen-and-ink style illustration (although the concepts here introduced are general and apply to other rendering styles as well). Pen-and-ink drawings can be reproduced by adequately placing individual strokes over the image. It is the combination of two basic illustration principles: *density* and *orientation* of strokes that conveys the desired appearance to the drawings. Changing stroke density we can represent different textures/lighting conditions, and strokes oriented along principal directions of objects in the scene effectively represent shape [10]. Producing a stroke-based representation of a scene has then two separate stages:

- *Feature and geometry extraction*—extract from the available input data the density and orientation values required to adequately represent the underlying scene.
- *Rendering*—draw a configuration of single strokes that achieves the density-orientation combinations specified in the previous stage.

The emphasis of this work is on the first stage. Stroke density information can naturally be obtained from images (as done here), while orientation information is traditionally obtained from 3D shape models. We propose to use computer vision techniques so that all the information relevant to the rendering process can be extracted from multiple images without the explicit availability of 3D models. This follows the tradition in computer vision of inferring critical 3D information without 3D reconstruction [8,14]. This has also been very influential in the graphics community (e.g., in the area of *image based rendering*) and it is starting to be used for NPR as well, e.g., [24].[1]

[1] We should note that this paper on multiple flashes is posterior to the work presented in this paper, see for example http://www.ima.umn.edu/preprints/nov2002/nov2002.html for early versions of this work.

This provides computational efficiency (no need for full reconstruction when only partial information from the 3D scene is needed), avoid intermediate unnecessary steps that add noise and reduce robustness, and provide further completeness in the area of 3D shape without 3D reconstruction. Note that we also provide examples with multiple lights, an area that has received much less attention in the full 3D scene reconstruction, but that can be efficiently exploited for rendering with the framework here proposed.

1.1. Previous work

NPR algorithms can be categorized according to the type of input they use: image or object (3D model) based. Image-based NPR use 2D photographs as reference to generate the renderings. Tonal information from objects in the scene is directly extracted from image gray levels and mimicked by the synthesis algorithm. As any type of image can be used, there is no restriction in the complexity of models that can be rendered including real scenes, faces, landscapes, etc. This is the kind of data, images, used for halftoning as well. On the down side, no geometric information can be inferred from the input image which is necessary to adequately convey 3D shape to the drawings. Image-based algorithms are presented in [28,29]. Object-based NPR techniques use 3D models as input. Algorithms for parametric [42,6,23], implicit [6], and triangulated [15,23,25,13] surfaces have been studied in the literature. These algorithms are limited to work on computer generated scenes, and unless additional surface properties are attached to the models, no other information but the geometry is available. Additional properties may characterize surface material, color, etc., which change the reflectance properties of the surface causing the appearance of the renderings also to change. This additional information may encode important scene features that if ignored can prevent the NPR representation from being successful. As detailed below, the use of multiple images as proposed in this work simultaneously brings the key advantages of both categories, making tonal and geometric information both available.

NPR algorithms can also be classified according to the degree of user intervention they require. Some are interactive systems that provide certain level of automation but mostly depend on user input, others automate many of the tasks but still depend on user input to some extent, the rest are fully automatic. The one we propose here belongs to the small yet very important class of fully automatic algorithms. Note that throughout this work we relate the *automation degree* only to the feature and geometry extraction stage, and this is where existing NPR algorithms perform poorly. The rendering stage is naturally a subjective and artistic composition step in which the user selects a desired drawing style/technique to achieve a particular illustration task. The pioneering work [41] describes the principles of traditional pen-and-ink illustration style and show that a number of these can be implemented as part of an automatic rendering system. From then on, a number of systems with increasing degree of automation have been proposed. Interactive image-based systems were proposed in [28] and [29]. In [28] the user manually selects a desired *stroke texture* for the different regions in an image, the stroke direction and density being the key elements in defining a texture. The input image is solely used as a visual reference for the user. In [29] target tone values are automatically obtained from the input image, and interaction techniques are provided for the user to edit a direction field that specifies the orientation of stokes over the image. By manually aligning the direction field with surface orientation of objects in the image, compelling 3D shapes can be obtained. Other authors align strokes in the direction perpendicular to the image gradient [33,36], this gives an automatic procedure to compute orientations but fails to adequately communicate shape. In object-based NPR algorithms [15,23,13,25], target tone values are obtained using standard lighting computations at mesh vertices and stroke directions follow the orientation of principal directions of the surface. Since both (simple) tone and orientation can easily be obtained from the input geometry, these algorithms have a higher degree of automation. However, they also have the limitations of object space NPR discussed above and require user intervention to some extent.

Once we specified the target *density*-*orientation* combinations (either manually or automatically), we use these to drive the rendering stage. Three basic methods have been used in the literature to generate stroke-based pen-and-ink style renderings. The first method individually places strokes in an image to match the specified direction and density values [28,41,42,15,19,16,33]. A few strokes will do for the brighter areas, while dark or shadow areas will require more dense stroke concentrations and cross-hatching. If we allow for drawing primitives other than strokes we can get a variety NPR styles, for example, the stippling technique use single dots as drawing primitives [3,32,21]. Placing individual stipples is closely related to the problem of halftoning in image processing. Indeed, the second rendering technique is the work done in non-photorealistic and stylized halftoning. Unlike traditional halftoning algorithms that try to mask the texture produced as a by product of the dithering process, these algorithms deliberately produce texture in a controlled fashion in order to get NPR style renderings. Texture tone and direction are controlled by locally changing the halftone screen according to image-based or external features [20,40,9,38]. The third rendering method pre-builds a set of textures (each being a particular arrangement of strokes representing a given tone and orientation) and then a texture synthesis algorithm reproduces those patterns throughout the image. This is the approach used in [23,13] for object-space rendering and the one we follow here. As any type of texture can be used, a variety of rendering styles can be synthesized with this method. Both stylized halftoning and texture synthesis techniques constitute attractive alternatives to computationally intensive rendering methods that use individual placement of primitives.

Besides reproducing the essential *tone*-*orientation* information, salient scene features can also be used to further enhance the appearance of renderings. Representing silhouette curves, sharp creases, object boundaries, edges, etc. can significantly enhance visual comprehension. This was done in several image and object-based NPR algorithms [15,23,27,34,12] and in stylized halftoning [36,39,35], effectively improving the quality of drawings. Also the recent work [24] addresses the extraction of depth edges and shape features from images with different illuminations for stylized-depiction of real scenes. Any of these techniques can be incorporated into our framework providing additional shape and perceptual cues.

1.2. Overview and contributions

Using multiple images as input to the NPR scheme, we propose a novel hybrid model that simultaneously uses information from the image and object domains (see Fig. 1), this having a number of advantages:

- Target illumination values are obtained directly from image gray levels (greatly reducing geometric modeling tasks used in object-based NPR).
- We can *automatically* compute principal directions using multiple image geometry (no need for the user to manually specify orientations as done in image-based NPR).
- Salient scene features to enhance renderings can easily be extracted from input images using standard image processing techniques, with the 2D image information complemented by the 3D geometry obtained from the multiple images.

The combination of all these features allows a degree of flexibility and automatization (in the extraction of features/geometry) not achieved by any existing image or object-based NPR algorithm.

Although the examples in this paper are for single objects (as often used in NPR), the use of multiple images allows to handle more complex scenes as well. As shown in this paper, there is no need to have 3D models of the scene to be rendered, just estimations of its basic geometric features, thereby enlarging the class of renderable scenes. The only requirement is that we must be able to reliably extract scene principal directions to adequately convey shape to the drawings. Single object scenes are typical examples where a meaningful orientation field can be obtained from multiple images.

As examples of this hybrid approach, we use input from two multiple-imaging systems: binocular stereo and a multiple-light photometric stereo. For both systems we assume calibration data are available so that geometry can be properly extracted.

Our particular implementation of the hybrid NPR model proposed in this paper has the following steps:

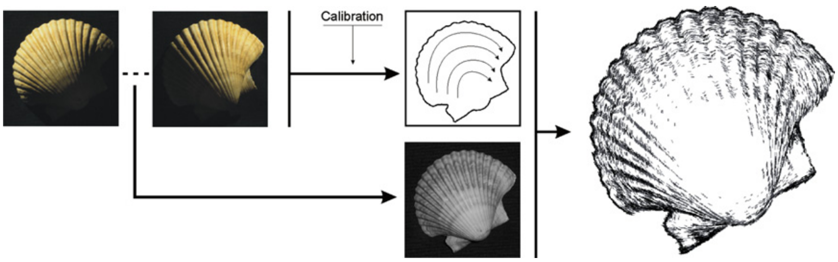

Fig. 1. System diagram for the generation of non-photorealistic rendering from multiple images. From the set of input images and the calibration data we extract principal directions of objects in the scene. Using these orientations and tone values extracted from images in the set we synthesize the output NPR image.

- Get surface normals directly from images using the calibration data (without intermediate 3D reconstruction), Sections 2.1 and 2.2.
- From this normal field, compute projection of principal directions directly on the image plane (again, without intermediate surface reconstruction), Section 2.3.
- Compute target tone values from gray levels in input images, Section 3.
- Finally, generate the drawings using texture synthesis rendering guided by target density-orientation combinations obtained above, Section 4.

We present a number of examples in Section 5.

Note that target tone values and principal directions may be obtained in many ways from multiple-image systems using well-established computer vision techniques. Likewise, the rendering stage can use any of the techniques described in Section 1.1 to produce the drawings. The particular selections in this paper are (important) examples that illustrate the proposed paradigm. The novelty and importance of this work is the concept of combining both sources of information (2D and 3D) as input for NPR, it's main implications being the increased degree of automation and the possibility to put together the advantages of each approach. This work also shows that those aspects of 3D geometry relevant to NPR can be extracted from multiple-imaging system, even without going to the trouble of explicitly generating the 3D model.

2. Geometry from multiple images

As mentioned before, the geometry we need for producing NPR are the principal directions of objects in the scene. Any multiple-image system that allows computation of principal directions can then be used as input to our system. Actually, since we get principal directions directly from the normals, any imaging system allowing computation of normals can also be used as input (this being a broader category of systems). As particular examples we consider two classic settings in computer vision: a binocular stereo system and a multiple-light photometric stereo system.

In the binocular stereo setting, two photos of the same object are taken from different positions in space [7,8,14]. If intrinsic and extrinsic camera parameters are known (full calibration) we can get 3D Euclidean reconstruction of the scene. Since the camera changes position from one photo to the other, an image registration algorithm (so called disparity map computation) is needed for identifying correspondent image points.

In the photometric stereo setting the camera is held at a fixed position and pictures are taken as we change lighting conditions on the scene. We use the acquisition devices in [18] that allow automatic acquisition of multiple images, each illuminated with an individual strobe light source. In this case, the camera is held fixed so there is no need for camera calibration or registration of images (unlike binocular stereo), only the 3D direction of each light source is needed for reconstructing geometry from the image set.

2.1. Normals from binocular stereo

In binocular stereo the usual approach is to build the 3D reconstruction of the surface from which normals and principal curvatures will then be derived. Instead, we follow the approach in [4] and use the images directly to compute surface normals without reconstructing the surface. The idea is to get an accurate estimate of both the disparity and its derivatives from the image gray values. A relation can then be established that relates these derivatives to differential properties of the surface, see [4] for details. Although principal directions can also be obtained with this method (estimating second order disparity derivatives), the accuracy of the estimates is not enough for our purposes. Instead, we choose to compute principal directions directly from the normal field, see Section 2.3.

2.2. Normals from multiple-lights

In multiple-light systems we obtain normals solving the classic photometric stereo problem [26,43]. Consider the simple illumination model:

$$I_i = \rho\, \vec{n} \cdot \vec{l}_i, \tag{1}$$

where I_i denotes the observed illumination intensities when light source i is active, and ρ is a constant depending on the light source intensity and the material's diffusion coefficient. $\vec{l}_i$ is the direction of the ith light source and $\vec{n}$ is the surface normal (both are unit vectors). The data we have available are the intensity images I_i and light source directions $\vec{l}_i$. Eq. 1 is then solved at every pixel for ρ and $\vec{n}$ by doing a least squares fitting. To avoid the effect of self-occlusions and specular reflections we do not consider image values that are excessively dark (occlusions) or extremely bright (specular reflections) when doing the fitting. We need at least three *qualifying* image values at each pixel to be able to recover the normal direction $\vec{n}$.

2.3. Principal directions

Suppose the surface of objects in the scene can be explicitly described as the graph of a differentiable function $f(x,y)$, where $(x,y) \in \Omega \subset \Re^2$ are the image coordinates and f is the unknown depth function. We parametrize the surface by $\mathcal{S} = (x, y, f(x,y))$ and compute its differential properties, see [5]. The unit normal is given by $\vec{n} = (-f_x, -f_y, 1)/\sqrt{f_x^2 + f_y^2 + 1}$, and principal curvatures and directions are the eigenvalues and eigenvectors, respectively, of the second fundamental form:

$$\mathrm{II} = (1 + f_x^2 + f_y^2)^{-\frac{3}{2}} \begin{bmatrix} f_{xx} & f_{xy} \\ f_{xy} & f_{yy} \end{bmatrix} \begin{bmatrix} 1 + f_x^2 & f_x f_y \\ f_x f_y & 1 + f_y^2 \end{bmatrix}.$$

Although $f(x,y)$ is unknown in our case (we avoid explicit 3D reconstruction), we do know the derivatives f_x and f_y from the normals $\vec{n}$ computed in Sections 2.1 and 2.2. Using then f_x, f_y and their derivatives we evaluate the second fundamental form II and solve the eigenvalue problem to get the directions of principal curvature of the surface. Note that although the principal directions are orthogonal in the object-centered coordinate system defined by local surface normal and tangent plane, they are not necessarily orthogonal when projected onto the image plane.

As pointed out in [15], stroke orientation should follow the curvature of the overall shape, whereas fine scene details are expressed through tonal variation. Accordingly, we compute principal directions in a lower resolution grid, smooth the resulting direction field, and then resample it on the original grid. Note that for the smoothing we are interested in regularizing *orientations* (invariant to π rotations), not *directions* (invariant to 2π rotations). Although any vector diffusion technique can still be applied, a simple adaptation procedure is required before doing the smoothing [22]: suppose orientations θ are originally in the $[-\frac{\pi}{2}, \frac{\pi}{2}]$ range. Take $\phi = 2\theta$ and build the 2D vector field $(u, v) = (\cos(\phi), sin(\phi))$. Regularize (u, v) using any vector diffusion technique to get the denoised field $(\tilde{u}, \tilde{v})$. Finally, perform the inverse adaptation procedure and get back the denoised orientation values

$$\tilde{\theta} = \frac{\text{atan}(\frac{\tilde{v}}{\tilde{u}})}{2}.$$

This procedure guarantees that we maintain the π-periodicity in the $\tilde{\theta}$ orientations. For the vector diffusion we used a PDE-based anisotropic diffusion technique, see [17,30], that denoises directions preserving discontinuities in the orientation field (which may correspond to relevant scene features we want to keep).

3. Target tone values

As in image-based NPR, tonal values to be mimicked by the synthesis algorithm are extracted from gray levels in the input images. Recall that in object-based NPR target tone values are obtained using lighting computations at mesh vertices, but unless material or color properties are attached to the mesh we may fail to reproduce important surface features (eyes in faces is one such example). As we look directly at gray levels in the input images, this information is readily available and we also significantly reduce geometric modeling tasks.

3.1. Extrapolated views and relighting

The synthesized NPR views are not restricted to be those of the images provided by the imaging system. On the contrary, one can produce renderings for arbitrary views/light conditions depending on the particular imaging system. For the binocular stereo case we can get any view (reasonably close to the existing reference views) by using novel view synthesis algorithms [2,31]. Note that from only two images the achievable set of extrapolated views is quite limited, using more images will give greater flexibility. In the photometric stereo case, it is straightforward to do scene relighting by simple evaluation of the illumination model used in polynomial texture maps (PTM) [18]. PTM illuminations are locally modeled by a biquadratic function of the projections of the light vector into the image plane. Once we have the approximation coefficients (obtained using a fitting algorithm), we can relight the scene from an arbitrary position by simple polynomial evaluation. Lighting changes alone do not have any effect on the projection of principal directions, but changes in the view position do change the scene geometry and orientations have to be recomputed in that case.

4. Rendering

Using scene information provided by multiple images we have automatically extracted the tone and orientation values that will now drive the rendering stage. As described in Section 1.1 we use a texture synthesis technique to generate the renderings. We first build a basic set of patterns each representing a given tonality using the *Tonal Art Maps* (TAMs) introduced in [23]. Since TAMs only represent a discrete number of tones, blending between contiguous images is needed to represent intermediate tone values. For this reason, TAMs are specifically designed to have spatial coherence, so the blending between adjacent textures can be done without artifacts. For each pixel in the synthesized image we have a target tone value t, and a desired orientation value θ. Accordingly, the texture synthesis will use a pattern representing the combination (t, θ) constructed as follows: first blend the two adjacent discrete tone images $\lfloor t \rfloor$and $\lceil t \rceil$from the TAM, and then rotate the blended image an angle θ, see Fig. 2. The texture synthesis is done with the priority-based algorithm [1], using a different sample texture at each pixel depending on the target (t, θ) values. The process starts by randomly selecting a point in the synthesized image, building the pattern representing the (t, θ) combination for that point, and copying a random small patch from that pattern into the synthesized image. The image is then grown by small patches (taken from the pattern representing the tone-orientation combination at the corresponding pixel) chosen so as to match the already synthesized portion of the image.

Note that for practical convenience we did not use any of the feature enhancement techniques mentioned in Section 1.1 (representing silhouettes, creases, object bound-

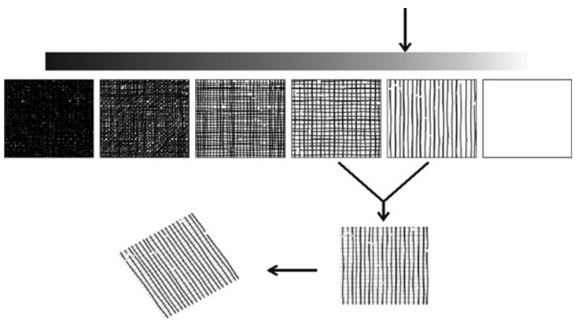

Fig. 2. Pattern formation depending on the target tone and orientation values. First blend the two adjacent images from the TAM, and then rotate the blended image.

aries, edges, etc.) which can further improve the appearance of drawings. Nevertheless, the hybrid framework potentially offers more flexibility since features can be extracted directly from the images (using standard image processing techniques) and/or the reconstructed geometry using the normals or principal directions.

5. Data sets and examples

From two calibrated binocular stereo pairs of real faces we generated Figs. 3 and 4. Note how principal directions guide the orientation of strokes so we get a compelling idea of shape and how features like the eyes are accurately reproduced (this could not be done simultaneously with single 2D or 3D existing techniques without

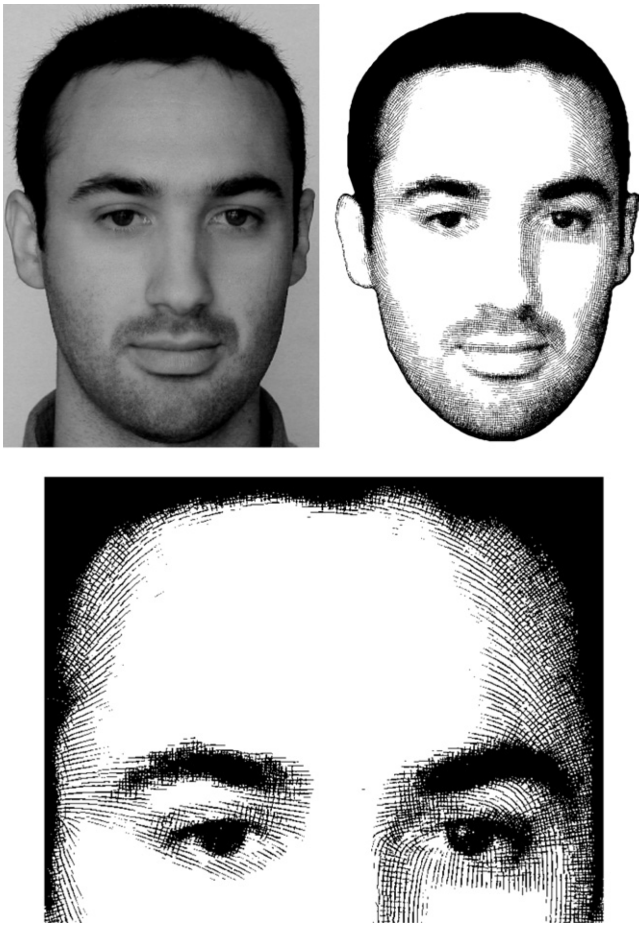

Fig. 3. NPR from binocular stereo, data taken from http://serdis.dis.ulpgc.es/~jsanchez.

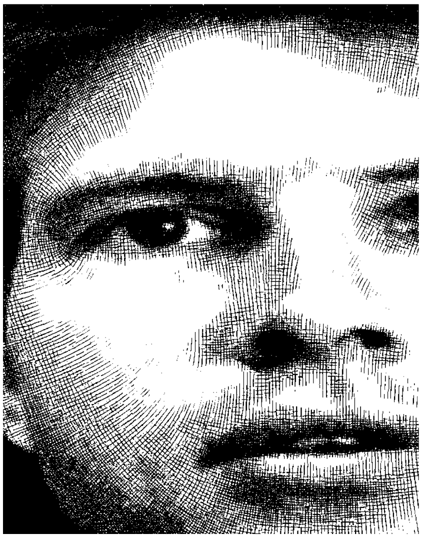

Fig. 4. NPR from binocular stereo, data taken from the RobotVis project at INRIA http://www.inria.fr/.

significant user intervention). Since the output produced by the texture synthesis algorithm is in general gray scale, we have binarized all the synthesized images in order to show a more faithful pen-and-ink representation. Examples generated from photometric stereo images are presented in Figs. 5 and 6. The shell in Fig. 5 was generated with the INSPIRE rendering system [19] (that uses individual placement of strokes), to show an alternative rendering technique. Relighting was also done on this image, as described in Section 3.1. The second object in Fig. 5 was selected on

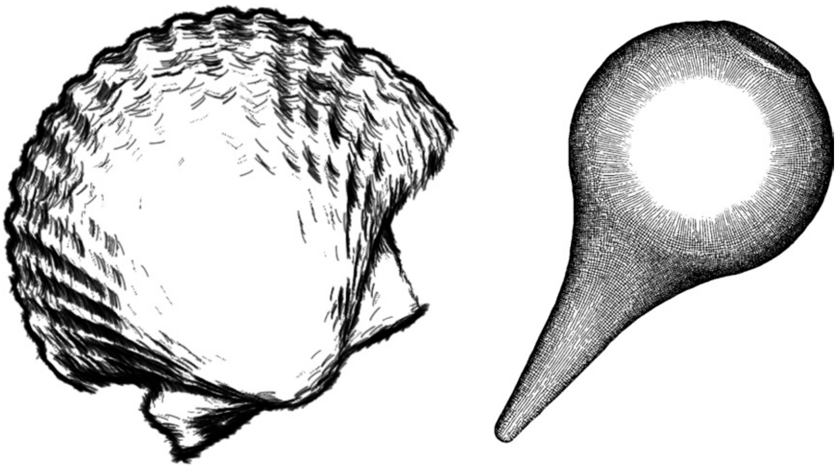

Fig. 5. NPR from multiple-lights. The shell image was produced with the INSPIRE rendering system [19] using the geometry extracted from multiple images. The object in the left shows the behavior of the algorithm when the computation of principal directions is ill posed.

Fig. 6. Statuette model from multiple-lights.

purpose to show the algorithm's behavior when the computation of principal directions is ill posed.

6. Summary and discussion

We presented a system for the generation of non-photorealistic illustrations using multiple images as input. The main idea is general in the sense that we could have used any procedure for extracting principal directions from multiple images or any available rendering technique to obtain the results. The emphasis and the novelty is in the idea of using multiple images to obtain both sources of information (image and object space) to automatically guide the rendering, and the fact that multiple images can provide them both with enough accuracy for this particular task.

Unlike 3D models which are either computer generated or obtained with specialized acquisition devices, multiple images are a more popular and inexpensive choice meaning that techniques like the one presented here can have a major impact on the applicability of existing NPR tools as they overcome the need for expensive/elaborate equipment and are not limited to computer generated scenes. Even when 3D models are available, existing object-based rendering techniques can still benefit and be further improved by including complementary image information to represent important perceptual features that would otherwise be left aside.

We now point out potential weaknesses of systems of this kind. First, as it is well know in stereo vision, we may get poor stereo estimates in some image areas due to failures in the registration process. This will give unreliable normal estimates and consequently inaccurate orientations. Although the smoothing process takes care of most of the irregularities, we may not get meaningful orientation values in these areas. Note that this will be improved considering more images and not just two as for the results shown in this paper (binocular stereo case). Second, and this is a general problem with NPR techniques, when computing principal directions we may get umbilical surface points where principal directions are not defined (this is the case of flat or spherical areas where there is not a preferred orientation for aligning the strokes). Again, if these are only isolated points the smoothing can take care of them. If not, more sophisticated filling-in procedures can be applied like in [15], or even user input may be required to specify orientations in these areas by use of a proper interface. Although all the results presented here are fully automatic, our system can also be regarded as an important component of existing tools like [29,28], that provides more sophisticated capabilities and reduces user interaction almost exclusively to the artistic/subjective aspect of stylized-rendering production.

To illustrate the ideas in this paper, we have used single objects obtained from multiple images. On the other hand, the framework of combining 2D and 3D NPR as here introduced provides the starting point for dealing with full scenes. The goal is to be able to render complicated scenes such as forests, where without full 3D reconstruction, geometry (from multiple views) is combined with tone and color (and maybe rendered using multiple styles). This is the subject of current research. Note that in this case, the image + geometry information can be obtained from multiple images as in this paper or from scanners that simultaneously output range and color information. How to deal with color images will also be addressed. Extending this work to NPR of video data and for real-time applications (where frame-to-frame coherence issues arise) is the subject of current efforts as well.

Acknowledgments

This work is partially supported by a grant from the Office of Naval Research, the Presidential Early Career Award for Scientists and Engineers (PECASE), and a National Science Foundation CAREER Award. We thank M. Nguyen and B. Chen for providing us the INSPIRE rendering system [19].

References

[1] K. Toyama. A. Criminisi, P. Prez. Object removal by exemplar-based inpainting, in: CVPR, June 2003.
[2] S. Avidan, A. Shashua, Novel view synthesis by cascading trilinear tensors, IEEE TVCG 4 (4) (1998) 293–306.
[3] O. Deussen, S. Hiller, C. van Overveld, T. Strothotte, Floating points: a method for computing stipple drawings, CGF 19 (3) (2000).
[4] F. Devernay, O. Faugeras, Computing differential properties of 3-d shapes from stereotopic images without 3-d models, Technical Report 2304, INRIA, July 1994.
[5] M.P. do Carmo, Differential Geometry of Curves and Surfaces, Prentice-Hall, Englewood Cliffs, NJ, 1976.
[6] G. Elber, Line art illustrations of parametric and implicit forms, IEEE Trans. Vis. Comput. Graph. 4 (1) (1998) 71–81, January–March.
[7] O. Faugeras, Three-dimensional Computer Vision, MIT Press, Cambridge, MA, 1993.
[8] O. Faugeras, Q. Luong, The Geometry of Multiple Images, MIT Press, Cambridge, MA, 2001.
[9] B. Freudenberg, M. Masuch, T. Strothotte, Real-time halftoning: a primitive for non-photorealistic shading, in: Rendering Techniques, 2002, pp. 227–231.
[10] A. Girshick, V. Interrante, S. Haker, T. Lemoine, Line direction matters: an argument for the use of principal directions in 3d line drawings, in: First International Symposium on Non-Photorealistic Rendering, 2000, pp. 43–52.
[11] B. Gooch, A. Gooch, Non-photorealistic Rendering, AK Peters, 2001.
[12] B. Gooch, P.J. Sloan, A. Gooch, P. Shirley, R.F. Riesenfeld, Interactive technical illustration, in: Symposium on Interactive 3D Graphics, ACM Press, 1999, pp. 31–38.
[13] G. Gorla, V. Interrante, G. Sapiro, Texture synthesis for 3d shape representation, IEEE Trans. Vis. Comput. Graph. 9 (4) (2003) 512–524.
[14] R. Hartley, A. Zisserman, Multiple View Geometry in Computer Vision, Cambridge, 2000.
[15] A. Hertzmann, D. Zorin, Illustrating smooth surfaces, in: Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques, ACM Press/Addison-Wesley Publishing, 2000, pp. 517–526.
[16] P. Jodoin, E. Epstein, M. Granger-Pich, V. Ostromoukhov, Hatching by example: a statistical approach, in: Non-Photorealistic Animation and Rendering 2002, Annecy, France, June 3–5, 2002.
[17] R. Kimmel, Numerical Geometry of Images: Theory, Algorithms, and Applications, Springer-Verlag, New York, 2004.
[18] T. Malzbender, D. Gelb, H. Wolters. Polynomial texture maps, in: SIGGRAPH 2001, Computer Graphics Proceedings, Annual Conference Series, ACM Press/ACM SIGGRAPH, 2001, pp. 519–528.
[19] M.X. Nguyen, H. Xu, X. Yuan, B. Chen, Inspire: an interactive image-assisted npr system, in: 11th Pacific Conference on Computer Graphics and Applications, Canmore, Alberta, Canada, October 8–10, 2003.
[20] V. Ostromoukhov, R. Hersch, Artistic screening, in: SIGGRAPH 95, Annual Conference Series, 1995, pp. 219–228.
[21] O.M. Pastor, B. Freudenberg, T. Strothotte, Real-time animated stippling, IEEE Comput. Graph. Appl. 23 (4) (2003) 62–68.
[22] P. Perona, Orientation diffusions, in: IEEE Transactions on Image Processing, XX(Annual Conference Series), 1999, pp. 100–107.
[23] E. Praun, H. Hoppe, M. Webb, A. Finkelstein, Real-time hatching, in: Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques, ACM Press, New York, 2001, p. 581.
[24] R. Raskar, K.-H. Tan, R. Feris, J. Yu, M. Turk, Non-photorealistic camera: depth edge detection and stylized rendering using multi-flash imaging, in: ACM SIGGRAPH, 2004.
[25] C. Rossl, L. Kobbelt. Line-art rendering of 3d models, in: Proceedings of Pacific Graphics 2000, Hong Kong, China, October 2000.

[26] H. Rushmeier, F. Bernardini. Computing consistent normals and colors from photometric data, in: Proceedings of the Second International Conference on 3D Digital Imaging and Modeling—3DIM'99, Ottawa, Canada, October 4–8 1999, pp. 99–108.
[27] T. Saito, T. Takahashi, Comprehensible rendering of 3-d shapes, Comput. Graph. 24 (4) (1990) 197–206.
[28] M.P. Salisbury, S.E. Anderson, R. Barzel, D.H. Salesin, Interactive pen-and-ink illustration, in: Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques, ACM Press, New York, 1994, pp. 101–108.
[29] M.P. Salisbury, M.T. Wong, J.F. Hughes, D.H. Salesin. Orientable textures for image-based pen-and-ink illustration, in: Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques, 1997.
[30] G. Sapiro, Geometric Partial Differential Equations and Image Processing, Cambridge University Press, New York, 2001.
[31] D. Scharstein, View Synthesis Using Stereo Vision, Springer-Verlag, Berlin, 1999.
[32] A. Secord, Weighted voronoi stippling, in: Proceedings of the Second International Symposium on Non-photorealistic Animation and Rendering, ACM Press, New York, 2002, pp. 37–43.
[33] A. Secord, W. Heidrich, L. Streit, Fast primitive distribution for illustration, in: Proceedings of the 13th Eurographics Workshop on Rendering, Eurographics Association, 2002, pp. 215–226.
[34] M.C. Sousa, P. Prusinkiewicz, A few good lines: suggestive drawing of 3d models, Comput. Graph. Forum (Proc. of EuroGraphics'03) 22 (3) (2003).
[35] L. Streit, O. Veryovka, J. Buchanan, Non-photorealistic rendering using an adaptive halftoning technique, in: Western Computer Graphics Symposium (SKIGRAPH '99), 1999.
[36] L.M. Streit, J.W. Buchanan, Importance driven halftoning, Proc. Eurograph. 17 (3) (1998) 207–217.
[37] T. Strothotte, S. Schlechtweg, Non-photorealistic Computer Graphics: Modeling, Rendering and Animation, Morgan Kaufmann, 2002.
[38] O. Veryovka, Animation with threshold textures, in: Proceedings of Graphics Interface, May 2002, pp. 9–16.
[39] O. Veryovka, J. Buchanan, Comprehensive halftoning of 3D scenes, in: Computer Graphics Forum (Eurographics '99), 1999, pp. 13–22.
[40] O. Veryovka, J. Buchanan, Texture based dither matrices, Comput. Graph. Forum 19 (1) (2000) 51–64, March.
[41] G. Winkenbach. Computer-Generated Pen-and-Ink Illustration. PhD thesis, University of Washington, 1996.
[42] G. Winkenbach, D. Salesin, Rendering Parametric Surfaces in Pen and Ink. Technical Report 96-01-05b, University of Washington, Seattle, Washington 98195, May 1996.
[43] R.J. Woodham, Gradient and curvature from the photometric-stereo method, including local confidence estimation, J Optic. Soc. America A 11 (11) (1994) 3050–3068, November.