Depth-Aware Coherent Line Drawings

Marc Spicker[1]* Julian Kratt[1] Diana Arellano[2] Oliver Deussen[1]

[1]University of Konstanz, Germany
[2]Filmakademie Baden-Württemberg, Germany

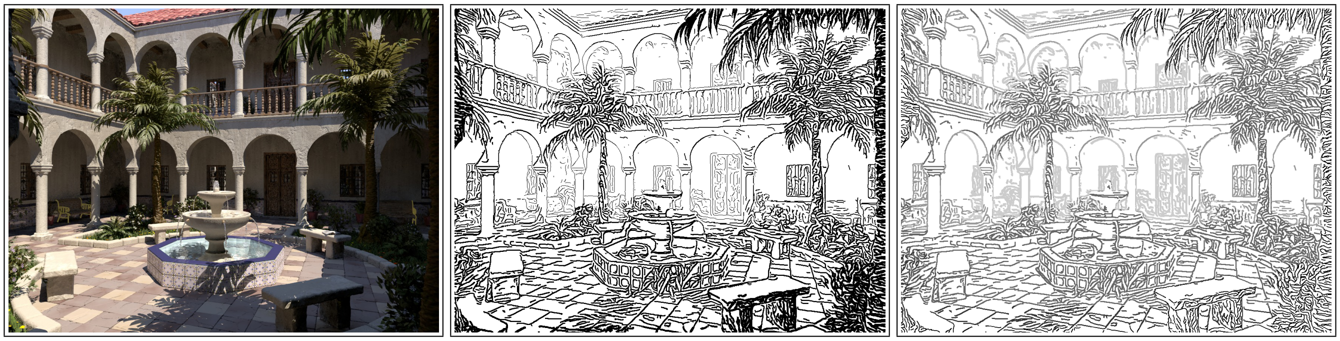

Figure 1: *Improving the depth perception of line drawings: An input 3D scene (left) and two depth-aware line drawings (center, right) using depth information of the scene in order to improve spatial awareness. Patio scene © Jaime Vives Piqueres.*

Abstract

In this paper we utilize depth information to extend a line drawing algorithm, improving depth perception and object differentiation in large and spatially complex scenes. We consider different scales of features and apply a flow-based morphological filter to the scenes. Based on this two line drawing styles are defined. The proposed algorithm works in real-time and enables users to manipulate the parameter space through instant visual feedback. We evaluated the effectiveness of our method by performing a user study.

CR Categories: I.3.3 [Computer Graphics]: Picture/Image Generation—Line and curve generation I.3.3 [Computer Graphics]: Picture/Image Generation—Display algorithms;

Keywords: non-photorealistic rendering, line drawing, edge detection, flow-based filtering, 2.5D technique

1 Introduction

Line art still surpasses photorealistic techniques in many applications. It is superior in depicting forms and structures; thus, many scientific illustrations are still being made with line drawings. However, most line drawings lack in conveying depth information, especially when complex spatial arrangements have to be shown. We extend an existing line drawing method by Kang et al. [2007] in order to combine the advantages of a flow-based approach with additional depth information. With this we improve depth perception and thus the quality of the results, while still working in real-time. Two different line rendering styles are suggested to depict depth information as illustrated in Figure 1. The first style varies line thickness to convey depth. Lines that are closer to the viewer are drawn thicker and guide the focus of the viewer towards these parts. The second style varies thickness and opacity: close lines are drawn thin and black, while lines further away are thicker and have a reduced contrast. This is useful for describing small details in close-by regions, while blurring out details in more distant parts of the scene. In summary, this work includes the following contributions:

- an improved edge tangent field construction method, which considers different scales of features,
- a flow-based morphological filter,
- two line drawing styles that aim at improving depth perception and object differentiation.

Finally, a user study was conducted showing that our method improves depth perception of the line drawings, that our line renderings are preferred over others to depict depth information, and that our method can be used to focus the users attention within the image.

2 Related Work

In the field of non-photorealistic rendering, a great amount of work has been dedicated to the field of line drawing and line stylization. Rendering descriptive lines of polygonal objects was explored by Elber [1995], who proposed simple techniques to create line drawings which enhance the quality of illustrations. Silhouettes are often regarded as important shape-conveying lines and Isenberg et al. [2002] introduced a technique to render them as connected stylized strokes at interactive rates. A system is presented by Kalnins et al. [2002] in which users directly draw strokes on 3D models with a chosen brush style. The number and placement of such strokes is adapted in new viewpoints to maintain the original look.

The field of using depth information for NPR stylization has been explored before: Gooch and Gooch [2004] enhance the depth perception of images by using artistic mattes. Depth discontinuities are used by Deussen and Strothotte [2000] to create pen-and-ink illustrations of complex tree models in an hybrid pixel-based and analytical algorithm. A method for capturing depth discontinuities as geometric features is proposed by Tan et al. [2005] to stylize

*e-mail:marc.spicker@uni-konstanz.de

depth edges captured with a multiflash capturing setup. Additional depth cues are introduced by Luft et al. [2006] by applying unsharp masking to the depth buffer to highlight spatially important areas in a scene with depth information. To enhance the depth perception in volumetric data with a large depth complexity, Bruckner and Gröller [2007] render halos in regions surrounding structural edges. Similarly, Everts et al. [2009] illustrate dense line data by creating depth-dependent halos around the lines. Lopez-Moreno et al. [2011] leverage well-known characteristics of human perception with a simple depth approximation algorithm to create new depictions of an input image. An interactive tool enables the manipulation of depth maps to explore artistic possibilities.

Continuous line illustrations, drawings consisting of exactly one line, have been automatically created by Wong and Takahashi [2011; 2013a] by inferring a graph from the edges extracted from an image. Texture information is used in flow-based anisotropic filtering by Kang et al. [2007], which has the advantage of creating coherent and smooth lines that mimic those drawn by human artists, but does not enhance depth perception and object recognition in complex scenes. Therefore we propose an extension to this method which, in contrast to other related methods, uses the depth information to improve the depth perception of the line drawing.

3 Adaptive Edge Tangent Flow

Similar to Kang et al., we create a 2D vector field called the edge tangent flow (ETF) t from an input image or a 3D scene. The vectors perpendicular to the gradients define the edge tangents. This initial edge tangent flow t^0 is still subject to noise present in the input and therefore has to be smoothed prior to usage, as shown by Kang et al. They propose a kernel-based nonlinear smoothing similar to a bilateral filter:

$$t^{i+1}(x) = \frac{1}{k} \sum_{y \in \Omega(x)} t^i(y)\, w_m(x,y)\, w_d(x,y)\ , \tag{1}$$

where $\Omega(x)$ denotes the neighbourhood of a 2D position x with radius r and k is the vector normalization term. The *magnitude weight function* w_m gives higher weight to gradients with a large magnitude, preserving strong edges. The *directional weight function* w_d gives higher weight to gradients with similar directions, creating a more uniform alignment of the vectors and avoiding swirls in the vector field. For more details about these two weight functions please consider the paper of the original authors.

The main problem with this filtering framework is that it does not consider different scales of features that are described within the edge tangent field. When choosing a large radius, small but visually important details may disappear. On the contrary, if the chosen filter size is too small, noise may still be present in larger structures. Thus, we propose an adaptive filter size, which is determined by the local average angular deviation. Before applying Eq. 1, the user-defined radius r is used to determine a locally-scaled version $r'(x)$ along the flow field ω centred at x:

$$r'(x) = \frac{1}{2r} \sum_{y \in \omega(x)} |t(x) \cdot t(y)|\ . \tag{2}$$

With similar local directions, this factor tends towards 1, leaving the filter size unchanged. Having a higher angular deviation, the factor becomes smaller, which decreases the filter size proportionally to the average angular deviation. We demonstrate the importance of this adaptive filter size in Figure 2, where we compare our adaptive filter to the constant version on an image with circles of increasing scale in front of a noisy background. While the constant filter has problems with features smaller than a certain size, the adaptive filter works well up to the point where features cannot be differentiated from noise any longer.

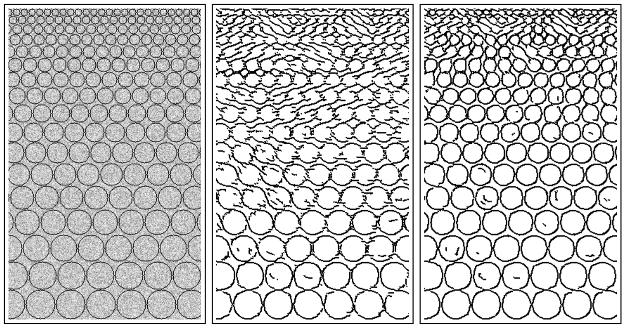

Figure 2: *Given a noisy input with features of increasing scale (left), a constant smoothing filter has problems with features smaller than a certain size (center), compared to an adaptive filter (right).*

4 Flow-Based Difference-of-Gaussians

The ETF constructed in the previous section is now used to steer a flow-based anisotropic difference-of-Gaussians (DoG) filter (FDoG). In a first step, a linear DoG filter is applied perpendicular to the edge tangent flow, that is, in the gradient direction, as proposed by Winnemöller et al. [2006]. The DoG response is computed for every pixel and in a second step accumulated along the flow, weighted by a third Gaussian function. The response of this filter can be seen as an edge-likelihood along the flow, avoiding local artefacts due to noise, and creating a locally coherent filter response. A limitation of this algorithm is that due to the underlying DoG filter, lines are not thin strokes but rather varying in thickness, depending on the magnitude of the underlying gradient. This is exemplified in Figure 3 by the light blue area along the flow (red line).

We propose a morphological filter that works in the same flow-based filtering framework. The thickness of the lines can be controlled by the maximum distance to the location of the maximum response perpendicular to the flow, represented by green dots. Using a distance of zero yields only the maximum itself, depicted as blue dashed line. In theory, more than just one local maxima is possible in gradient direction. We simply use the closest maxima to determine the distance. The depth information is used to determine two parameters, line thickness and opacity, to derive two line drawing styles described in the next section. Results of this filter with both rendering styles in comparison to the unchanged filter version are shown in Figure 6.

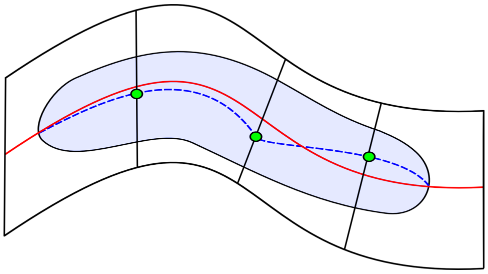

Figure 3: *Instead of thin lines, the underlying DoG filter produces an area of varying thickness. In our flow-based morphological filter we determine the maxima of the filter response (green) perpendicular the flow field (red), resulting in a thin line (blue).*

5 Line Drawing Styles

In daily life and also in photorealistic rendering the shading of objects give necessary cues about spatial arrangements in a scene. For line drawings, similar effects can be created by varying the line thickness, as depicted in Figure 4.

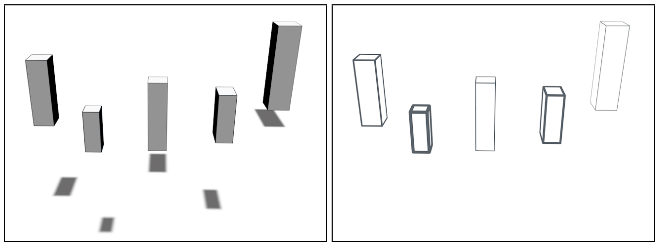

Figure 4: *Depth perception created by shading information (left). A similar effect can be achieved by varying line thickness (right).*

Using additional depth information, we propose two artistic styles aiming at creating a better depth perception of line drawings. In the conducted user study we showed their usefulness when determining the spatial arrangement of scenes.

(i) The first style varies edge thickness to convey the depth information. A larger thickness is used in regions with a low depth value since they are closer to the camera, this guides the users attention to close-by objects.

(ii) The second style varies edge thickness and also the opacity of lines. While the first style highlights close regions with thicker edges, a problem arises if object details are discarded due to the thick edges. In contrast to this, the second style uses thin, opaque lines for low depth values and thick, faded lines located at a further distance. This is analogue to most painterly rendering techniques that process the rendering from back to front, with larger brushes in the back and smaller brushes in the front [Hertzmann 1998].

In addition to varying the thickness and the contrast of the lines depending on the depth, we also adjust the level of details of the lines themselves. We increase noise suppression and smoothing in the DoG calculation for regions further away and vice versa. This results in a detailed foreground, but only larger structures remaining in the background.

6 Results

We applied our method to different inputs, consisting of 2D images and 3D scenes. Although depth information can be estimated for 2D images, e.g. by Monocular Depth Estimation [Saxena et al. 2009], a more flexible approach is wanted. We provide an interactive way to generate depth layers. The user segments the input image into layers by coloring regions using a brush. Since our method runs in real-time, immediate feedback is provided and the segmentation can be adjusted if necessary. Figure 6 shows a comparison between the algorithm proposed by Kang et al. and our line drawing styles. The segmentation directly defines the importance of areas in the final line drawing. In order to increase the depth perception, the segments are ordered from back to front. However, by changing the ordering or the segmentation itself, the user can freely set the focus to a specific region within the image. Similar ideas with interactive user-segmented regions have been proposed before: Cong et al. [2011] keep marked regions within an image realistic, while abstracting the rest of the image. Wong and Takahashi [2013b] define a type or style for each region in a continuous line drawing.

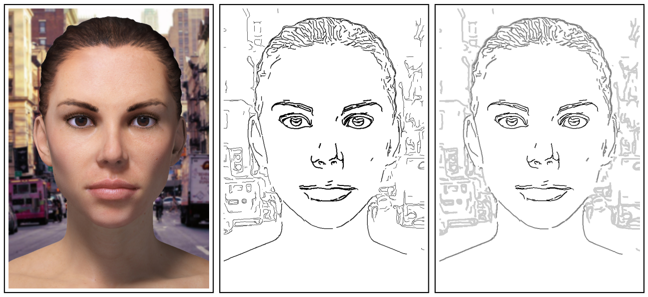

Figure 5: *Rendering of a 3D character model and corresponding line drawings with focus on the eyes and mouth region.*

A practical use case for the proposed line drawing algorithm is facial abstraction for children and adolescents diagnosed with autism spectrum disorder (ASD). Problems encountered by individuals with ASD are the inability to relate with others in a socially meaningful way, being one of the causes of a deficit in social cognition and emotion perception [Hudepohl et al. 2013], provoked by the amount of details conveyed in faces. Line drawing algorithms can be used to abstract faces to reduce the information load, and to convey the presented information more efficiently. Our proposed algorithm can generate abstract faces that consider a defined importance map, i.e. assign more significance to the eyes and mouth region. This is shown in Figure 5. For the input 3D scene in Figure 1 we use the depth map provided by the GPU to apply our line drawing styles. The objects within the scene are easier to differentiate due to different line characteristics. Without considering depth information, this would be more difficult because lines of similar orientation and edge strength are often used to depict non-connected objects and regions.

We conducted a user study with 40 undergraduate and graduate students to evaluate the effectiveness of our method. In the first task, we asked the subjects to order similar objects of different scales from back to front. These objects were rendered with diffuse shading, diffuse shading and shadows, uniform lines, and lines of different width, depending on the depth. We used the Kendall tau rank correlation coefficient to measure the difference between the orderings. While diffuse shading without shadows and uniform lines performed worst in our test (~ 0.55), non-uniform lines performed almost as well (~ 0.88) as using diffuse shading and shadows (~ 0.95). In the second task, the subjects were given an input image and two line renderings: one with our style and one with a randomly generated style based upon line thickness and opacity. When given the task to select the line rendering which created the most similar depth perception, 70% of the subjects chose one of our styles. In the last task, we showed that our method can be used to focus the users attention in a similar fashion to using depth-of-field. When given an image created by using depth-of-field, all subjects were able to determine the focus within the image. For a different image created with our method focussing on a certain depth range, 76% of the subjects were able to determine the focal area. This confirms the improved depth perception of our algorithm, as well as its potential usage to focus the users attention on certain areas of the line drawing.

7 Conclusion

Current line drawing techniques lack in conveying spatial arrangement and object recognition, especially in complex scenes. We contributed to this field by extending the algorithm of Kang et al., improving the construction of their edge tangent flow. We also enhanced the algorithm by incorporating depth information with a built-in flow-based morphological filter and suggested two line

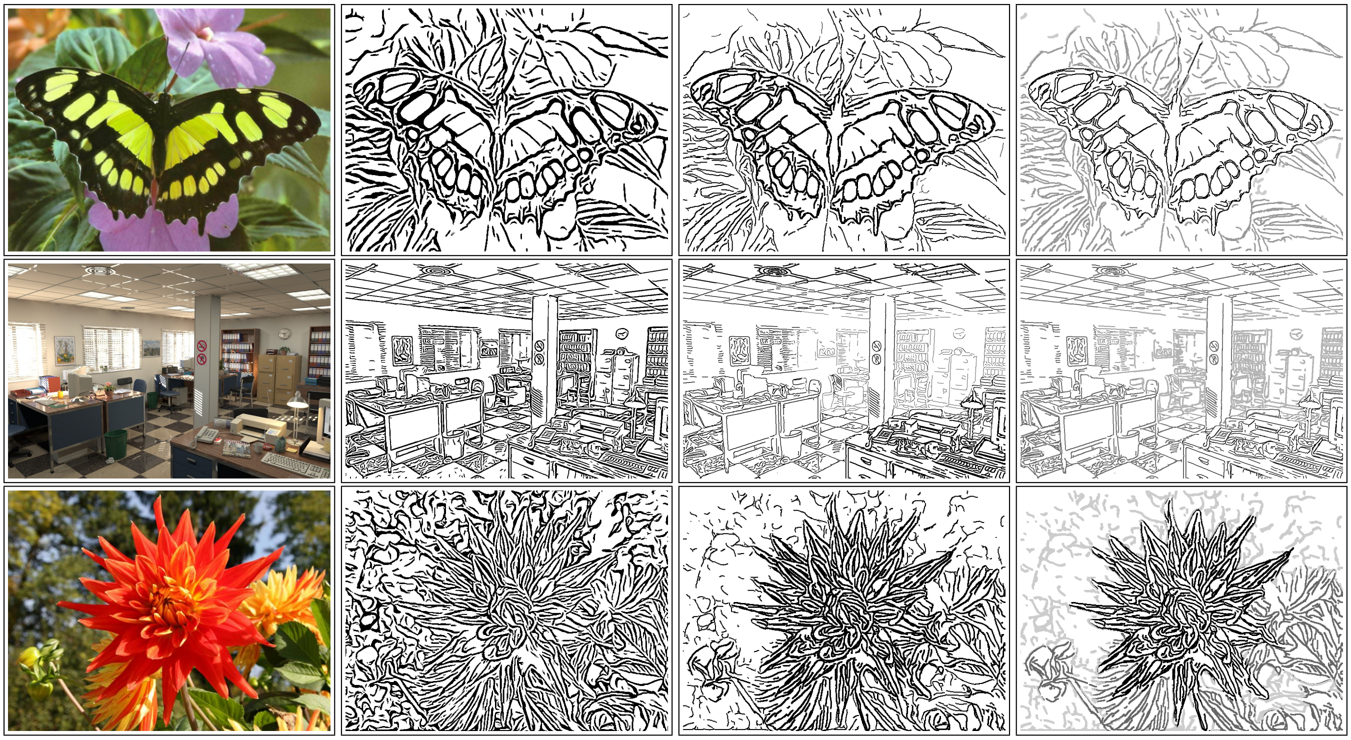

Figure 6: *Comparison to the results of Kang et al. From left to right: Input, result by Kang et al., and two line drawings generated with our framework. Office scene (center) © Jaime Vives Piqueres.*

drawing styles that create a better understanding of the spatial arrangement of the scene even in absence of cues like shading, as shown in the user study. We developed a real-time application, which enables artists to interactively explore the parameter space, creating sophisticated line drawings for a variety of different scenes, while also giving users the possibility to produce importance maps.

A limitation of our proposed algorithm is that the kernel sizes, as well as the filter radii, are scale dependent since they have absolute values. Also, while the algorithm is working in real-time even for complex scenes, the coherence is limited. It works well if the distance of a feature to the camera does not change, since the ETF then keeps its relative resolution. If this distance changes, the field also changes its resolution, resulting in popping in and out of features. Also since the ETF is a vector-based representation it suffers from artefacts as shown by Zhang et al. [2007]. For future work we plan on switching to a tensor-based representation with similar adaptive filter sizes.

Acknowledgements

We would like to thank the German Research Foundation (DFG) for financial support within the projects DE 620/18-1, as well as A04 of SFB/Transregio 161.

References

BRUCKNER, S., AND GROLLER, E. 2007. Enhancing depth-perception with flexible volumetric halos. *IEEE Trans. on Visualization and Comput. Graph. 13*, 1344–1351.

CONG, L., TONG, R., AND DONG, J. 2011. Selective image abstraction. *Vis. Comput. 27*, 187–198.

DEUSSEN, O., AND STROTHOTTE, T. 2000. Computer-generated pen-and-ink illustration of trees. SIGGRAPH '00, 13–18.

ELBER, G. 1995. Line illustrations in computer graphics. *The Visual Computer 11*, 290–296.

EVERTS, M. H., BEKKER, H., ROERDINK, J. B. T. M., AND ISENBERG, T. 2009. Depth-dependent halos: Illustrative rendering of dense line data. *IEEE Trans. on Visualization and Comput. Graph. 15*, 1299–1306.

GOOCH, A. A., AND GOOCH, B. 2004. Enhancing perceived depth in images via artistic matting. In *Proc. of the 1st Symp. on Applied Perception in Graph. and Visualization*, ACM Press, 168.

HERTZMANN, A. 1998. Painterly rendering with curved brush strokes of multiple sizes. ACM, SIGGRAPH '98, 453–460.

HUDEPOHL, M., ROBINS, D., KING, T., AND HEINRICH, C. 2013. Short report: The role of emotion perception in adaptive functioning of people with autism spectrum disorders. *Autism: The International Journal of Research & Practice*.

ISENBERG, T., HALPER, N., AND STROTHOTTE, T. 2002. Stylizing silhouettes at interactive rates: From silhouette edges to silhouette strokes. *Computer Graphics Forum*, 249–258.

ISENBERG, T., FREUDENBERG, B., HALPER, N., SCHLECHTWEG, S., AND STROTHOTTE, T. 2003. A developer's guide to silhouette algorithms for polygonal models. *IEEE Comput. Graph. Appl. 23*, 28–37.

KALNINS, R. D., MARKOSIAN, L., MEIER, B. J., KOWALSKI, M. A., LEE, J. C., DAVIDSON, P. L., WEBB, M., HUGHES, J. F., AND FINKELSTEIN, A. 2002. WYSIWYG NPR: Drawing strokes directly on 3D models. *ACM Trans. Graph. 21*, 755–762.

KANG, H., LEE, S., AND CHUI, C. K. 2007. Coherent line drawing. In *Proc. of the 5th Int. Symp. on Non-photorealistic Animation and Rendering*, NPAR '07, 43–50.

LOPEZ-MORENO, J., JIMENEZ, J., HADAP, S., ANJYO, K., REINHARD, E., AND GUTIERREZ, D. 2011. Non-photorealistic, depth-based image editing. *Computers & Graphics 35*, 99–111.

LUFT, T., COLDITZ, C., AND DEUSSEN, O. 2006. Image enhancement by unsharp masking the depth buffer. *ACM Trans. on Graph. 25*, 1206–1213.

SAXENA, A., SUN, M., AND NG, A. Y. 2009. Make3D: Learning 3D scene structure from a single still image. *IEEE Trans. on Pattern Analysis and Machine Intelligence 31*, 824–840.

TAN, K.-H., FERIS, R., TURK, M., KOBLER, J., YU, J., AND RASKAR, R. 2005. Harnessing real-world depth edges with multiflash imaging. *IEEE Comput. Graph. Appl. 25*, 32–38.

WINNEMÖLLER, H., OLSEN, S. C., AND GOOCH, B. 2006. Real-time video abstraction. In *SIGGRAPH '06*, 1221–1226.

WONG, F. J., AND TAKAHASHI, S. 2011. A graph-based approach to continuous line illustrations with variable levels of detail. *Computer Graphics Forum 30*, 7, 1931–1939.

WONG, F. J., AND TAKAHASHI, S. 2013. Hierarchical design of continuous line illustrations. In *GRAPP & IVAPP 2013*, 131–138.

WONG, F., AND TAKAHASHI, S. 2013. Abstracting images into continuous-line artistic styles. *The Visual Computer 29*, 6-8, 729–738.

ZHANG, E., HAYS, J., AND TURK, G. 2007. Interactive tensor field design and visualization on surfaces. *IEEE Transactions on Visualization and Computer Graphics 13*, 1 (Jan.), 94–107.