Line Drawings from 3D Models

A Tutorial

Other titles in Foundations and Trends® in Computer Graphics and Vision

Publishing and Consuming 3D Content on the Web: A Survey
Marco Potenziani, Marco Callieri, Matteo Dellepiane and Roberto Scopigno
ISBN: 978-1-68083-536-6

Crowdsourcing in Computer Vision
Adriana Kovashka, Olga Russakovsky, Li Fei-Fei and Kristen Grauman
ISBN: 978-1-68083-212-9

The Path to Path-Traced Movies
Per H. Christensen and Wojciech Jarosz
ISBN: 978-1-68083-210-5

(Hyper)-Graphs Inference through Convex Relaxations and Move Making Algorithms
Nikos Komodakis, M. Pawan Kumar and Nikos Paragios
ISBN: 978-1-68083-138-2

Line Drawings from 3D Models

A Tutorial

Pierre Bénard
LaBRI (UMR 5800, CNRS, Univ. Bordeaux)
Inria Bordeaux Sud-Ouest
pierre.benard@labri.fr

Aaron Hertzmann
Adobe Research
hertzman@dgp.toronto.edu

now
the essence of knowledge
Boston — Delft

Foundations and Trends® in Computer Graphics and Vision

Published, sold and distributed by:
now Publishers Inc.
PO Box 1024
Hanover, MA 02339
United States
Tel. +1-781-985-4510
www.nowpublishers.com
sales@nowpublishers.com

Outside North America:
now Publishers Inc.
PO Box 179
2600 AD Delft
The Netherlands
Tel. +31-6-51115274

The preferred citation for this publication is

P. Bénard and A. Hertzmann. *Line Drawings from 3D Models*. Foundations and Trends® in Computer Graphics and Vision, vol. 11, no. 1-2, pp. 1–159, 2019.

ISBN: 978-1-68083-591-5
© 2019 P. Bénard and A. Hertzmann

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, mechanical, photocopying, recording or otherwise, without prior written permission of the publishers.

Photocopying. In the USA: This journal is registered at the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923. Authorization to photocopy items for internal or personal use, or the internal or personal use of specific clients, is granted by now Publishers Inc for users registered with the Copyright Clearance Center (CCC). The 'services' for users can be found on the internet at: www.copyright.com

For those organizations that have been granted a photocopy license, a separate system of payment has been arranged. Authorization does not extend to other kinds of copying, such as that for general distribution, for advertising or promotional purposes, for creating new collective works, or for resale. In the rest of the world: Permission to photocopy must be obtained from the copyright owner. Please apply to now Publishers Inc., PO Box 1024, Hanover, MA 02339, USA; Tel. +1 781 871 0245; www.nowpublishers.com; sales@nowpublishers.com

now Publishers Inc. has an exclusive license to publish this material worldwide. Permission to use this content must be obtained from the copyright license holder. Please apply to now Publishers, PO Box 179, 2600 AD Delft, The Netherlands, www.nowpublishers.com; e-mail: sales@nowpublishers.com

Foundations and Trends® in Computer Graphics and Vision

Volume 11, Issue 1-2, 2019

Editorial Board

Editor-in-Chief

Brian Curless
University of Washington

William T. Freeman
MIT

Luc Van Gool
KU Leuven and ETH Zurich

Editors

Marc Alexa
TU Berlin

Kavita Bala
Cornel

Ronen Basri
Weizmann Institute of Science

Peter Belhumeur
Columbia University

Andrew Blake
Microsoft Research

Chris Bregler
Facebook-Oculus

Joachim Buhmann
ETH Zurich

Michael Cohen
Microsoft Research

Paul Debevec
USC Institute for Creative Technologies

Julie Dorsey
Yale

Fredo Durand
MIT

Olivier Faugeras
INRIA

Rob Fergus
NYU

Mike Gleicher
University of Wisconsin

Richard Hartley
Australian National University

Aaron Hertzmann
Adobe Research

Hugues Hoppe
Microsoft Research

C. Karen Liu
Georgia Tech

David Lowe
University of British Columbia

Jitendra Malik
Berkeley

Steve Marschner
Cornell

Shree Nayar
Columbia

James O'Brien
Berkeley

Tomas Pajdla
Czech Technical University

Pietro Perona
California Institute of Technology

Marc Pollefeys
ETH Zurich

Jean Ponce
Ecole Normale Superieure

Long Quan
HKUST

Cordelia Schmid
INRIA

Steve Seitz
University of Washington

Amnon Shashua
Hebrew University

Peter Shirley
University of Utah

Noah Snavely
Cornell

Stefano Soatto
UCLA

Richard Szeliski
Microsoft Research

Joachim Weickert
Saarland University

Song Chun Zhu
UCLA

Andrew Zisserman
Oxford

Editorial Scope

Topics

Foundations and Trends® in Computer Graphics and Vision publishes survey and tutorial articles in the following topics:

- Rendering
- Shape
- Mesh simplification
- Animation
- Sensors and sensing
- Image restoration and enhancement
- Segmentation and grouping
- Feature detection and selection
- Color processing
- Texture analysis and synthesis
- Illumination and reflectance modeling
- Shape representation
- Tracking
- Calibration
- Structure from motion
- Motion estimation and registration
- Stereo matching and reconstruction
- 3D reconstruction and image-based modeling
- Learning and statistical methods
- Appearance-based matching
- Object and scene recognition
- Face detection and recognition
- Activity and gesture recognition
- Image and video retrieval
- Video analysis and event recognition
- Medical image analysis
- Robot localization and navigation

Information for Librarians

Foundations and Trends® in Computer Graphics and Vision, 2019, Volume 11, 4 issues. ISSN paper version 1572-2740. ISSN online version 1572-2759. Also available as a combined paper and online subscription.

Contents

Line Drawings from 3D Models

Pierre Bénard[1] and Aaron Hertzmann[2]

[1] *LaBRI – Inria; pierre.benard@labri.fr*
[2] *Adobe Research; hertzman@dgp.toronto.edu*

ABSTRACT

This tutorial describes the geometry and algorithms for generating line drawings from 3D models, focusing on occluding contours.

The geometry of occluding contours on meshes and on smooth surfaces is described in detail, together with algorithms for extracting contours, computing their visibility, and creating stylized renderings and animations. Exact methods and hardware-accelerated fast methods are both described, and the trade-offs between different methods are discussed. The tutorial brings together and organizes material that, at present, is scattered throughout the literature. It also includes some novel explanations, and implementation tips.

A thorough survey of the field of non-photorealistic 3D rendering is also included, covering other kinds of line drawings and artistic shading.

Pierre Bénard and Aaron Hertzmann (2019), "Line Drawings from 3D Models", Foundations and Trends® in Computer Graphics and Vision: Vol. 11, No. 1-2, pp 1–159. DOI: 10.1561/0600000075.

1

Introduction

Humans have been drawing pictures since the days of prehistoric cave painting. Various forms of line drawing have been developed since then, including Egyptian hieroglyphs, medieval etching, and industrial-era printmaking. Nowadays, line drawing and outline drawing methods are used throughout cartoons and comics, architectural rendering, instructional tutorials, and many other settings. Drawing is the starting point for many kinds of tasks, for everyone from children making pictures to professional architects sketching ideas. Drawing seems to be fundamentally connected to how we represent the world visually.

While most computer graphics focuses on realistic visual simulation, over the past few decades, line drawing algorithms have also matured. We now have the ability to automatically create reasonable line drawings from 3D geometry, much like photorealistic rendering. These algorithms provide deep insight into the geometry and topology of line drawings, which can be surprisingly subtle, given how simple line drawing might seem. Versions of these algorithms have been used throughout art, entertainment, and visualization. User evaluation has shown that these algorithms, indeed, accurately describe important aspects of how artists draw lines. This shows that these algorithms can contribute to a scientific understanding of art.

This tutorial provides a detailed guide to the mathematical theory and computer algorithms for line drawing of 3D objects. We focus on the curves known as *occluding contours* or, simply, *contours*. These are the most important curves for line drawing of 3D surfaces. They have a rich theory around them, and, once one understands this theory, understanding how other curves operate is much simpler. We describe the different algorithms required to compute and render these curves, together with references to the literature. We also explain boundary curves and surface-surface intersection curves, since these are straightforward to include and often important. We also discuss open research problems in contour rendering.

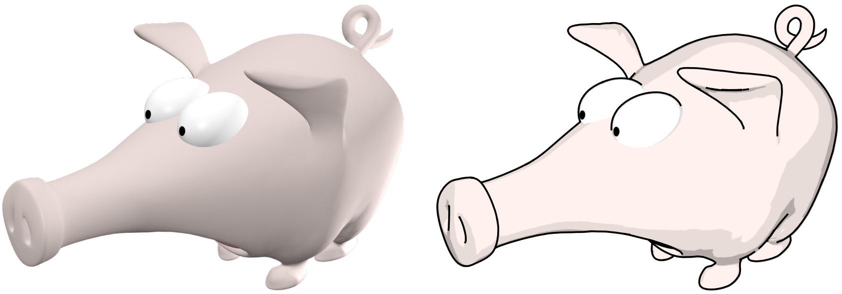
In addition, we survey of other topics in 3D non-photorealistic rendering, with extensive pointers to the literature, including: other types of curves, stroke rendering, and non-photorealistic shading. We do not cover the complementary topic of image-based non-photorealistic rendering; for a survey of image-based methods, we refer the reader to the book by Rosin and Collomosse (2013).

The theory of line drawing is currently scattered about and incomplete in research papers. The algorithms for line drawing include many subtleties that are not described in the literature, and many pitfalls await the coder attempting them. There remain some important open problems, but these gaps are not obvious from the literature. This tutorial is meant to address these issues.

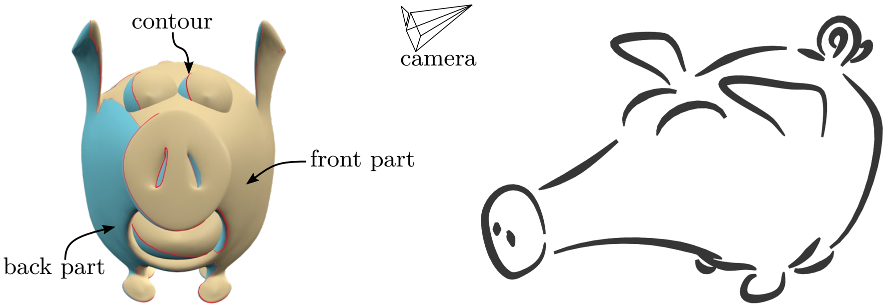
We believe that these topics ought to be known by anyone interested in understanding the curves in visual representational art. It is one where computer graphics can make a unique contribution. Arguably, the algorithmic simulation of line drawing is a crucial step in understanding visual art.

1.1 Occluding contours

This tutorial focuses on the curves known as *occluding contours* or, simply *contours*. In some computer graphics research, these have been



(a) 3D object with diffuse shading **(b)** Contours with toon shading



(c) Side view of the scene **(d)** Stylized curves

Figure 1.1: Occluding contours — The occluding contours of the 3D model "Origins of the Pig" © Keenan Crane, shown in **(a)** with diffuse shading, are depicted in **(b)** composited with toon shading to produce a cel-like drawing. As illustrated in **(c)** from a side view, they delineate the frontier between the front and back parts of the surface when seen from the camera. These contour curves can be further process to produce stylized imagery, such as the calligraphic brush strokes in **(d)**.

called *silhouettes*, though the *silhouettes* are technically a separate set of curves, as we will describe below.

The occluding contours of a simple 3D object are shown in Figure 1.1. As a first definition, suppose we have a 3D object that we wish to render from a specific viewpoint. The occluding contours are *surface curves that separate visible parts from invisible parts*. By rendering the visible portions of these 3D curves together with the object, we get a basic line drawing (Figure 1.1b).

There are many different contour detection and rendering algorithms, and some significant tradeoffs between them. The most important tradeoff is between simple algorithms that produce approximate results, and more complex algorithms that give more precision, control, and stylization capabilities. Just rendering reasonable-looking contours as solid black lines is very straightforward for a graphics programmer. These most basic algorithms can be implemented in a few additional lines of code in an existing renderer, and have been implemented in many real-time applications, including many popular video games (one of the earliest was *Jet Set Radio* in 2000). However, if we wish to stylize the curves, for example, by rendering the curves with sketchy or calligraphic strokes (Figure 1.1d), things become more difficult. With a bit of perseverance, renderings with distinctive and lovely styles can be created. At times, these renderings may still contain topological artifacts that are not suitable for very high-end production. High-quality algorithms that remove these artifacts are more complex; in the extreme, no provably-correct algorithm exists for this problem. However, there are a number of partial solutions that are good enough to be used in many circumstances, and we discuss in detail the issues involved.

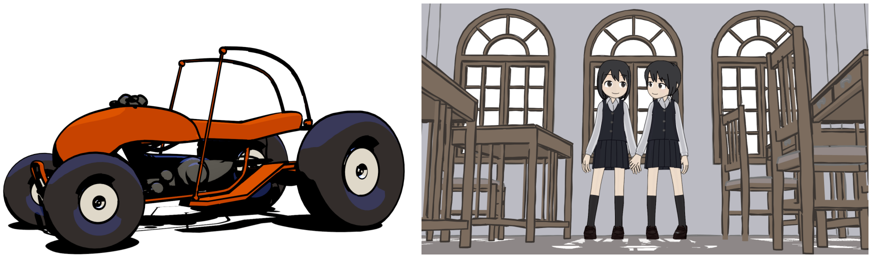
Note that, formally, the occluding contour and occluding contour generator are separate curves in 2D and 3D. However, we will frequently use the term “contour” to refer to each of them, since the correct terms are very cumbersome, and the meaning of “contour” is usually obvious from context.

1.2 How to use this tutorial

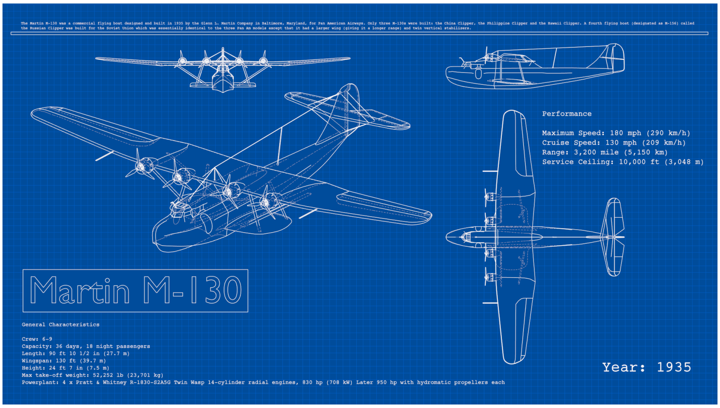
This tutorial is two things: a detailed tutorial of the core contour algorithms, and a high-level survey of nearly all of 3D non-photorealistic rendering. We cover some core topics more thoroughly than any previous publication, and, for other topics, we mainly provide pointers to the literature.

Hence, reading the tutorial directly will give a good overview of the field, but one may skim through survey sections. Alternatively, a practitioner may wish to jump directly to the algorithms relevant to their task.

Generally speaking, real-time image-based methods, especially based on graphics shaders, offer the best real-time performance and have been



(a) Buggy by Rylan Wright 🅭🅯 **(b)** Anime by mato.sus304 🅭🅯🄎



The Martin M-130 was a commercial flying boat designed and built in 1935 by the Glenn L. Martin Company in Baltimore, Maryland, for Pan American Airways. Only three M-130s were built: the China Clipper, the Philippine Clipper and the Hawaii Clipper. A fourth flying boat (designated as M-156) called the Russian Clipper was built for the Soviet Union which was essentially identical to the three Pan Am models except that it had a larger wing (giving it a longer range) and twin vertical stabilizers.

Performance

Maximum Speed: 180 mph (290 km/h)
Cruise Speed: 130 mph (209 km/h)
Range: 3,200 mile (5,150 km)
Service Ceiling: 10,000 ft (3,048 m)

General Characteristics

Crew: 6-9
Capacity: 36 days, 18 night passengers
Length: 90 ft 10 1/2 in (27.7 m)
Wingspan: 130 ft (39.7 m)
Height: 24 ft 7 in (7.5 m)
Max take-off weight: 52,252 lb (23,701 kg)
Powerplant: 4 x Pratt & Whitney R-1830-S2A5G Twin Wasp 14-cylinder radial engines, 830 hp (708 kW) Later 950 hp with hydromatic propellers each

Year: 1935

(c) Martin M-130 blueprint by LightBWK 🅭🄍

(d) Ryner by Lucas Gogol 🅭🄍

Figure 1.2: Artworks created by artists using Blender Freestyle — Each of these is a non-photorealistic rendering, using the techniques described in this tutorial in different ways.

used in many games. These are described in the next Chapter, and pointers to further reading are provided there. This chapter can also help build intuitions for all readers.

The core chapters of the tutorial focus on contour detection and visibility on 3D models. We start with 3D mesh representations, and then apply the same ideas to different smooth surface representations in the subsequent chapters.

We then cover the core topic of detecting contours on meshes (Chapter 3) and computing their visibility (Chapter 4). Contour detection and visibility on meshes is the most basic and well-understood problem, and we go into the most detail in algorithms here. We describe fast, approximate hardware based visibility in Chapter 6.

While it may be tempting to use mesh algorithms for smooth surfaces, in Chapter 5, we explain some of the problems with doing so. We then describe a method called Interpolated Contours that provides a compromise position, being almost as simple as mesh contours to implement, with relatively few inconsistencies.

We then discuss true contours on parametric surface representations (Chapter 7). Understanding these curves involves some differential geometry (reviewed in Appendix A), and the resulting mathematics and theory is rather elegant. We describe detection and visibility algorithms, which are adapted from the mesh algorithms. We describe the different strategies that have been applied to this problem and how they compare. In the following chapter, we then discuss these algorithms as applied to implicit smooth surface representations (Chapter 8).

Finally, we discuss stylized rendering and animations algorithms (Chapter 9), and conclude with a discussion of the state of research and applications in 3D non-photorealistic rendering (Chapter 10).

1.3 The importance of visualizations

Although we have done our best to explain contours in text, they can take some time to wrap your head around. Understanding how the 2D curves and 3D curves relate in an image like Figure 1.1 can be challenging. It is worthwhile spending time with these figures, perhaps starting with simpler examples like different views of a torus, to understand how the

2D and 3D shapes relate, what the curves look like at singular points, and so on.

We provide an interactive viewer at https://benardp.github.io/contours_viewer/. Experimenting with this viewer can help give intuitions on contours.

Even better is to use, or build, a 3D visualization. If you implement a 3D contour rendering system, it is essential to also implement visualizations that let you zoom into the 2D drawing and rotate around the 3D model. In each view, you should be able to render the different types of curves and singularities and their attributes. These visualizations are essential for deep understanding of these curves, as well as for debugging and algorithm development. You can start with simple 3D drawings, e.g., rendering contour edges on the 3D model, and coloring mesh faces according to facing direction as in Figure 1.1c. As your system becomes more sophisticated, you may eventually have visualizations like those in Figure 4.6.

These visualizations are also useful in making certain design choices. As we discuss, there is no current foolproof system for smooth contour rendering, and so there are some choices to be made, e.g., selecting heuristics. Good visualizations can also be helpful in understanding how different heuristics behave.

1.4 The science and perception of art

The algorithms described in this tutorial provide a new level of insight and understanding into the science of art (Hertzmann, 2010). For centuries, artists, historians, philosophers, and scientists have sought a formal understanding of visual art: how do we make it, and how do we perceive it? One of the first generative tools in art was the development of linear perspective during the Italian Renaissance. The theory of occluding contours, which is the main subject of this tutorial, originated in perceptual psychology and computer vision, and was developed into the sophisticated algorithms we described here by computer graphics researchers.

As perceptual psychology has developed, so have perceptual theories of art. For example, one of the most influential modern writers on visual art is Ernst Gombrich (1961), who argued that artists created artistic styles of depiction over the centuries. Nelson Goodman (1968) took this further to argue that all artistic style functions purely as a denotational system of symbols, like characters on a written page, and, presumably, purely learned as a product of culture. Rudolf Arnheim (1974) attempted to formulate Gestalt-like perceptual rules to drawings. Sayim and Cavanagh (2011) attempt to apply modern neuroscience to understanding the perception of art.

In attempt to formalize the description of styles, John Willats (1997) created a denotational semantics to describe different kinds of realistic styles — expanded by Willats and Durand (2005) to include insights from computer graphics.

Non-photorealistic rendering provides a generative theory for how artists create representational art. Like any theory, it does not cover every case or describe every phenomenon accurately, nor does it say anything about cultural, psychological, or other outside factors in the work. However, this generative theory provides considerable potential insight into how art is made.

We can compare the generative theory to the world before and after Newtonian mechanics. Before Newton, philosophers like Aristotle could make qualitative observations about how objects move (e.g., “heavy objects like to fall”) but could make no real predictions. Newtonian mechanics is predictive, it generates insights, and leads to real understanding. Likewise, understanding the generative model of representational art provides a potentially compact way to understand many phenomena.

Two landmark studies validate and justify the use of line drawing algorithms developed in the non-photorealistic rendering literature. Cole *et al.* (2008) undertook a careful study of how artists depict 3D objects. They asked a collection of art students to illustrate several 3D models with line drawings, and compared how the artists’ drawings related to the line drawing algorithms in this section (Figure 1.3). They showed that roughly 80% of a typical drawing could be explained by existing algorithms. This study helps show which of these algorithms are most useful, while also highlighting gaps in the literature. In a follow-up

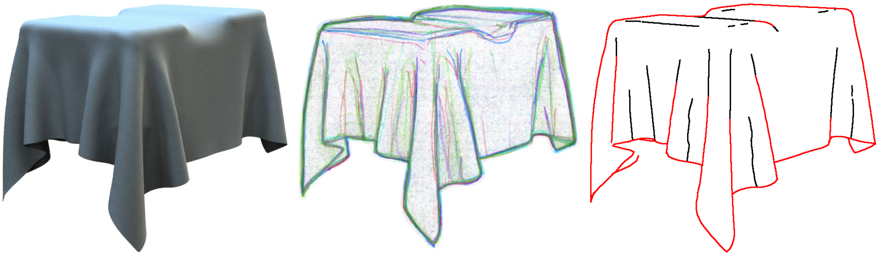


Figure 1.3: Correlation between hand-drawn lines and contours — A 3D model rendered from a given viewpoint and illumination (left) has been hand-drawn by ten artists (center). Observe how consistent the drawings are, especially near the contours of the shape. The contours (in red) and suggestive contours (in black) extracted from the 3D model are depicted in the right. Images taken from the "Javascript Drawing Viewer"[1] of Cole *et al.* (2008).

paper, Cole *et al.* (2009) showed that line drawing algorithms are also very effective at conveying 3D shape.

1.5 Survey of feature curves

This section surveys other important types of surface curves for line drawing, together with pointers to the relevant literature. The remaining chapters focus solely on contour, boundary, and surface-intersection curves.

Most of these curves have been developed both for artistic use and for visualization purposes (Lawonn *et al.*, 2018). However, some types of curves, such as ridges and valleys, seem useful for visualization without mimicking conventional artist curves as well.

Visibility-indicating curves. *Contours* indicate where parts of the surface become visible and invisible, and also indicate where visibility changes. There are a few other important curves that are important for similar reasons.

Boundary curves are simply the boundaries of the surface. Closed surfaces do not have boundaries. These curves are usually rendered when visible. Boundary curves can indicate change of visibility for curves that

---

[1] http://gfx.cs.princeton.edu/proj/ld3d/lineset/viewer/index.html

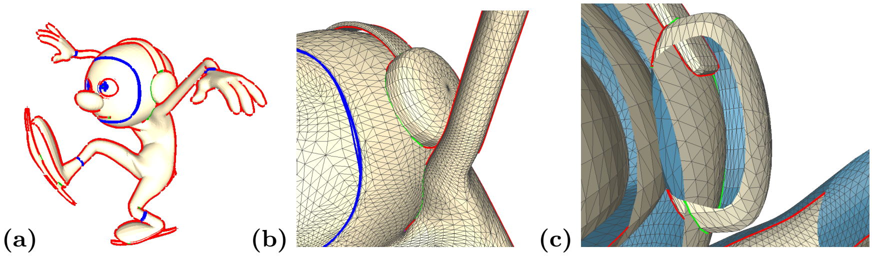


Figure 1.4: Surface-surface intersection curves (from Bénard *et al.* (2014)) — Professionally-modeled surfaces include many intersections between surface, such as this ice-skating character. Surface intersection curves are shown in green, occluding contours in red, and boundaries in blue. Observe how the ear muffs intersects the headband and the hoodie; the shoulder also happens to intersect the hoodie in this animation frame. **(a,b)** Original surface. **(c)** Cross-section from a different viewpoint. Red © Pixar ("Red" character created at Pixar by Andrew Schmidt, Brian Tindall, Bernhard Haux and Paul Aichele, based on the original design of Teddy Newton.)

they intersect in image space, so they are important to handle, and we include them in the discussions of our algorithms in this tutorial.

Surface intersection curves occur when two different sections of surface intersect. These do not occur in the clean models often used in computer graphics research. However, in professional 3D computer animation applications, modelers frequent connect different object parts this way (Figure 1.4). These curves can be detected with standard computer graphics algorithms, and are important to extract since they can indicate changes of visibility with curves that they intersect on the surface. We do not discuss them any further in this tutorial.

All other curves are essentially surface "decorations"; computing them is optional for visibility computations. They typically visualize the surface curvature rather than its outlines.

Contour generalizations. Perhaps the next most significant set of curves are those that generalize contours. These curves were first introduced by DeCarlo *et al.* (2003; 2004), who described a mathematically-elegant generalization of contours and the algorithms needed to render them. Several other variants inspired by this idea were proposed, including apparent ridges (Judd *et al.*, 2007).

The abstracted shading method (Lee *et al.*, 2007) demonstrated how these and lighting-based variants could be computed in image-space. Other variants based on image-space processing include Laplacian Lines (Zhang *et al.*, 2009) and DoG lines (Zhang *et al.* 2012; 2014). In addition to speed, image-space lines have the advantage that they automatically remove clutter as a function of image-space line density, although, like all image-based methods, they potentially lose some fine-scale precision and control.

Figure 1.5 shows some examples of these contour generalizations. Including some form of these curves seems essential for capturing how artists depict surfaces; these curves were essential in the study of Cole *et al.* (2008).

These curves have also been generalized to include highlights that illustrate shading on an object (DeCarlo and Rusinkiewicz, 2007). DeCarlo (2012) provides a thorough survey and comparison of these different types of contour generalizations.

Surface features/properties. Some intrinsic properties of the surface can be drawn, such as sharp creases on smooth surfaces (Saito and Takahashi, 1990), as well as changes in shading (Xie *et al.*, 2007). When objects have assigned texture and materials, one may wish to draw the material boundaries or the texture itself.

Hatching. Hatching strokes illustrate surface shape in line drawings. Winkenbach and Salesin (1994) use manually-authored hatching textures and orientations. For more automation, one can use the iso-parametric curves of parametric surfaces (Elber, 1995a; Winkenbach and Salesin, 1996). However, these lines depend on how the shape was authored, and do not generalize to other types of surfaces. Elber (1998) explored many different possible hatching directions, including principal curvature directions, texture gradients, and illumination gradients. Principal curvature-based hatching is supported by perceptual studies suggesting that human perceive hatching strokes as curvature directions (Mamassian and Landy, 1998). Hertzmann and Zorin (2000) refine principal curvature hatching for umbilic regions (Figure 1.6). Singh and Schaefer

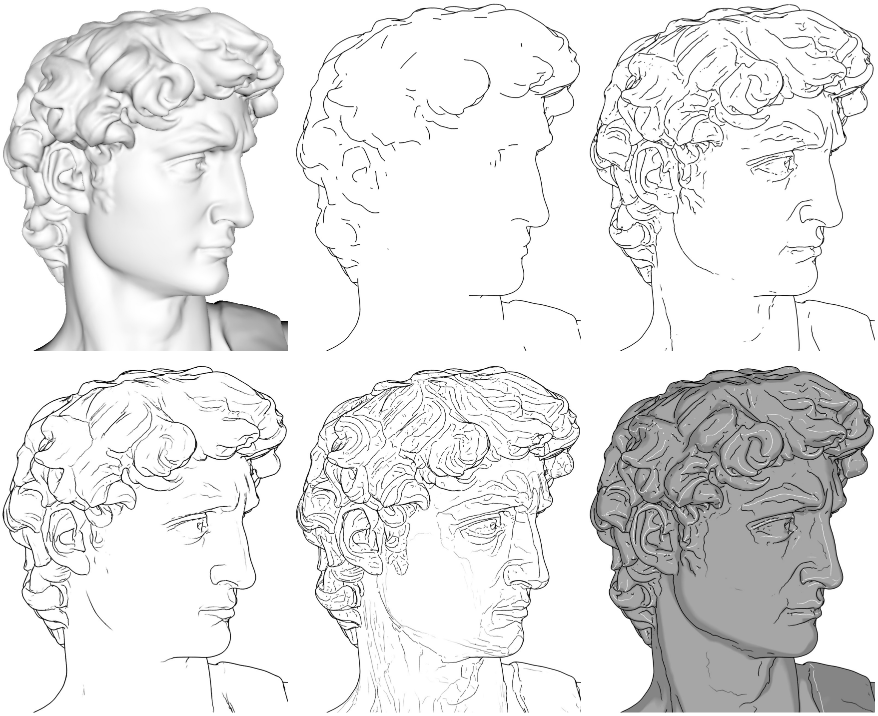


Figure 1.5: Feature curve examples — From left to right, top to bottom: diffuse rendering of the 3D scanned David model by "Scan The World" (http://mmf.io/o/2052), occluding contours (OC), OC + suggestive contours (SC) (DeCarlo *et al.*, 2004), OC + apparent ridges (Judd *et al.*, 2007), OC + ridges & valleys (Rusinkiewicz, 2004), and OC + SC + principal highlights + toon shading (DeCarlo and Rusinkiewicz, 2007). Images generated with "qrtsc" (Cole *et al.*, 2011).

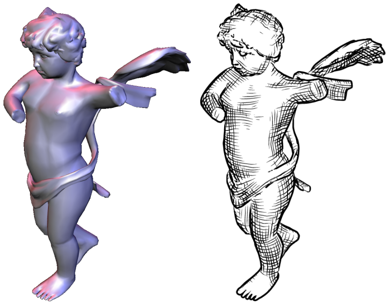


Figure 1.6: Hatching — 3D Cupid model and hatching result obtain with the method of Hertzmann and Zorin (2000).

(2010) describe hatching strokes that follow shading gradients. Since artists draw different types of hatching curves in different situations, Kalogerakis *et al.* (2012) combine these ideas, describing a machine learning system for learning hatching directions, identifying which hatching rules are used in which parts of a 3D surface. Gerl and Isenberg (2013) additionally offer interactive tools to let the user dynamically control the placement and orientation of hatches.

Surface extrema. Extremal curves, such as ridges and valleys, generalize the notion of ridges and valleys in terrain maps, identifying curves of locally maximal or minimal curvature. These types of curves are a visualization technique that can be useful in understanding surface shape; they do not typically correspond to artist-drawn curves otherwise.

Numerous algorithms have been developed to extract ridges and valleys from various types of geometric models (Interrante *et al.*, 1995; Thirion and Gourdon, 1996; Pauly *et al.*, 2003; Rusinkiewicz, 2004; Ohtake *et al.*, 2004; Yoshizawa *et al.*, 2007; Vergne *et al.*, 2011). A variant, called Demarcating Curves (Kolomenkin *et al.*, 2008; DeCarlo, 2012) can help visualize shapes of different regions on a surface.

1.6 Brief history of 3D Non-Photorealistic Rendering

The earliest 3D computer graphics algorithms were hidden-line rendering algorithms (Roberts, 1963), including methods that we discuss in this tutorial (Appel, 1967; Weiss, 1966). While the mainstream of computer graphics focused on photorealistic imagery, a few works aimed at adding artistic stroke textures to architectural drawings and technical illustrations[2], e.g., (Dooley and Cohen, 1990; Yessios, 1979); meanwhile a number of 2D computer paint programs were developed as well. Many of these papers argued for the potential virtues of hand-drawn styles in technical illustration.

In 1990, the flagship computer graphics conference SIGGRAPH held a session entitled "Non Photo Realistic Rendering," which seems to be the first usage of this term. In this session, two significant papers for the

---

[2] Many works are being omitted from this history. A much more comprehensive bibliography, up to 2011, can be found here: https://www.npcglib.org.

field were presented. Saito and Takahashi (1990) introduced depth-buffer based line enhancements (Chapter 2), which started to create cartoon-like renderings of smooth objects by emphasizing contours and other feature curves. Haeberli (1990) introduced a range of artistic 2D image-processing effects; these papers together demonstrated a significant step forward in the quality and generality of non-photorealistic effects.

Winkenbach and Salesin (1994) demonstrated the first complete line-drawing algorithm from 3D models, including contours and hatching. Their work was seminal in that their method automatically produced beautiful results from 3D models; one could, for the first time, be fooled into thinking that these images were really drawn by hand. Perhaps even more importantly, their work provided a model for one could develop algorithms by careful study of artistic techniques in textbooks and illustrations.

Meier (1996) demonstrated the first research paper focusing on 3D non-photorealistic animation, describing the problem of temporal coherence for animation. Between the beautiful images of Winkenbach and Salesin (1994; 1996) and beautiful animations of Meier (1996), non-photorealistic rendering was firmly established as an important research direction.

Research activity at SIGGRAPH increased significantly, and the inaugural NPAR symposium on Non-Photorealistic Animation and Rendering met in 2000, sponsored by the Annecy Animation Festival in France and chaired by David Salesin and Jean-Daniel Fekete. Through the following decade, many improvements and extensions to the basic ideas were published, and, occasionally, techniques like toon shading and contour edges appeared in video games and movies. DeCarlo *et al.* (2003) described Suggestive Contours, which substantially improved the quality of line renderings, while making deep connections to perception and differential geometry, notably the work of Koenderink (1984). Several systems were created to help artists design artistic rendering styles, such as WYSIWYG NPR (Kalnins *et al.*, 2002) and a procedural NPR system called Freestyle (Grabli *et al.*, 2010). Cole *et al.* (2008; 2009) performed the scientific studies described in Section 1.5 demonstrating that line drawing algorithms were quite good at capturing how artists draw lines.

Since then, research in 3D non-photorealistic rendering has tapered off, despite the presence of several significant open problems. In contrast, interest in image stylization has recently exploded, due to developments in machine learning. Still, 3D non-photorealistic rendering continues to appear in a few games and movies here and there. This tutorial aims, in part, to summarize the field and highlight open problems, to help researchers and practitioners make progress in this field in order to enable them to be more widely used. We discuss future prospects for the field in the Conclusion (Chapter 10).

References

Agarwala, A., A. Hertzmann, D. H. Salesin, and S. M. Seitz. 2004. "Keyframe-based Tracking for Rotoscoping and Animation". In: *ACM SIGGRAPH 2004 Papers. SIGGRAPH '04*. ACM. 584–591. DOI: 10.1145/1186562.1015764.

St-Amour, J.-F. 2010. "The Illustrative Rendering of Prince of Persia". In: *ACM SIGGRAPH 2010 Courses*. URL: http://www.cs.williams.edu/~morgan/SRG10.

Appel, A. 1967. "The Notion of Quantitative Invisibility and the Machine Rendering of Solids". In: *Proceedings of the 1967 22Nd National Conference. ACM '67*. ACM. 387–393. DOI: 10.1145/800196.806007.

Araújo, B. R. de, D. S. Lopes, P. Jepp, J. A. Jorge, and B. Wyvill. 2015. "A Survey on Implicit Surface Polygonization". *ACM Comput. Surv.* 47(4): 60:1–60:39. ISSN: 0360-0300. DOI: 10.1145/2732197.

Arnheim, R. 1974. *Art and Visual Perception: A Psychology of the Creative Eye*. 2nd. University of California.

Asente, P. J. 2010. "Folding Avoidance in Skeletal Strokes". In: *Proceedings of the Seventh Sketch-Based Interfaces and Modeling Symposium. SBIM '10*. Eurographics Association. 33–40. ISBN: 978-3-905674-25-5.

Barla, P., J. Thollot, and L. Markosian. 2006. “X-toon: An Extended Toon Shader”. In: *Proceedings of the 4th International Symposium on Non-photorealistic Animation and Rendering. NPAR ’06*. Annecy, France: ACM. 127–132. ISBN: 1-59593-357-3. DOI: 10.1145/1124728.1124749.

Baxter, W. V., J. Wendt, and M. C. Lin. 2004. “IMPaSTo: A realistic, interactive model for paint”. In: *Proceedings of the International Symposium on Non-Photorealistic Animation and Rendering, NPAR*. Ed. by S. N. Spencer. ACM. 45–56. DOI: 10.1145/987657.987665.

Ben-Zvi, N., J. Bento, M. Mahler, J. Hodgins, and A. Shamir. 2015. “Line-Drawing Video Stylization”. *Computer Graphics Forum*. DOI: 10.1111/cgf.12729.

Bénard, P., A. Bousseau, and J. Thollot. 2011. “State-of-the-Art Report on Temporal Coherence for Stylized Animations”. *Computer Graphics Forum*. 30(8): 2367–2386. DOI: 10.1111/j.1467-8659.2011.02075.x.

Bénard, P., F. Cole, A. Golovinskiy, and A. Finkelstein. 2010. “Self-similar Texture for Coherent Line Stylization”. In: *Proceedings of the 8th International Symposium on Non-Photorealistic Animation and Rendering. NPAR ’10*. ACM. 91–97. DOI: 10.1145/1809939.1809950.

Bénard, P., F. Cole, M. Kass, I. Mordatch, J. Hegarty, M. S. Senn, K. Fleischer, D. Pesare, and K. Breeden. 2013. “Stylizing Animation by Example”. *ACM Trans. Graph.* 32(4). DOI: 10.1145/2461912.2461929.

Bénard, P., A. Hertzmann, and M. Kass. 2014. “Computing Smooth Surface Contours with Accurate Topology”. *ACM Trans. Graph.* 33(2): 19:1–19:21. DOI: 10.1145/2558307.

Bénard, P., J. Lu, F. Cole, A. Finkelstein, and J. Thollot. 2012. “Active Strokes: Coherent Line Stylization for Animated 3D Models”. In: *Proceedings of the Symposium on Non-Photorealistic Animation and Rendering. NPAR ’12*. Eurographics Association. 37–46.

Benichou, F. and G. Elber. 1999. “Output sensitive extraction of silhouettes from polygonal geometry”. In: *Computer Graphics and Applications, 1999. Proceedings. Seventh Pacific Conference on*. 60–69. DOI: 10.1109/PCCGA.1999.803349.

Benthin, C., S. Woop, M. Nießner, K. Selgrad, and I. Wald. 2015. "Efficient Ray Tracing of Subdivision Surfaces using Tessellation Caching". In: *Proceedings of the 7th High-Performance Graphics Conference*. ACM.

Bentley, J. L. and T. A. Ottmann. 1979. "Algorithms for Reporting and Counting Geometric Intersections". *IEEE Transactions on Computers*. C-28(9): 643–647. ISSN: 0018-9340. DOI: 10.1109/TC.1979.1675432.

Bigler, J., J. Guilkey, C. Gribble, C. Hansen, and S. G. Parker. 2006. "A Case Study: Visualizing Material Point Method Data". In: *Eurographics /IEEE VGTC Symposium on Visualization*. The Eurographics Association. ISBN: 3-905673-31-2. DOI: 10.2312/VisSym/EuroVis06/299-306.

BlenderNPR. 2015a. "Edge node V1.2.4". http://blendernpr.org/edge-node-v1-2-4-july-2015/.

BlenderNPR. 2015b. "Solidify modifier Contour/Outline". http://blendernpr.org/solidify-modifier-contouroutline/.

Blinn, J. F. 1978. "A Scan Line Algorithm for Displaying Parametrically Defined Surfaces". In: *Proceedings of the 5th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH '78*. ACM. 27–. DOI: 10.1145/800248.807364.

Bo, P., G. Luo, and K. Wang. 2016. "A graph-based method for fitting planar B-spline curves with intersections". *Journal of Computational Design and Engineering*. 3(1): 14–23. ISSN: 2288-4300. DOI: 10.1016/j.jcde.2015.05.001.

Bourdev, L. 1998. "Rendering Nonphotorealistic Strokes with Temporal and Arc-Length Coherence". *MA thesis*. Brown University. URL: http://www.cs.brown.edu/research/graphics/art/bourdev-thesis.pdf..

Brabec, S. and H.-P. Seidel. 2003. "Shadow Volumes on Programmable Graphics Hardware". *Computer Graphics Forum*. 22(3): 433–440. ISSN: 1467-8659. DOI: 10.1111/1467-8659.00691.

Bremer, D. and J. F. Hughes. 1998. "Rapid Approximate Silhouette Rendering of Implicit Surfaces". In: *Proceesings of Implicit Surfaces 98*.

Breslav, S., K. Szerszen, L. Markosian, P. Barla, and J. Thollot. 2007. "Dynamic 2D Patterns for Shading 3D Scenes". *ACM Trans. Graph.* 26(3). ISSN: 0730-0301. DOI: 10.1145/1276377.1276402.

Buchholz, B., N. Faraj, S. Paris, E. Eisemann, and T. Boubekeur. 2011. "Spatio-temporal Analysis for Parameterizing Animated Lines". In: *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Non-Photorealistic Animation and Rendering. NPAR '11.* ACM. 85–92. DOI: 10.1145/2024676.2024690.

Burns, M., J. Klawe, S. Rusinkiewicz, A. Finkelstein, and D. DeCarlo. 2005. "Line Drawings from Volume Data". *ACM Trans. Graph.* 24(3): 512–518. DOI: 10.1145/1073204.1073222.

Busking, S., A. Vilanova, and J. J. van Wijk. 2008. "Particle-based non-photorealistic volume visualization". *The Visual Computer*. 24(5): 335–346. ISSN: 1432-2315. DOI: 10.1007/s00371-007-0192-x.

Campagna, S., L. Kobbelt, and H.-P. Seidel. 1998. "Directed Edges — A Scalable Representation for Triangle Meshes". *Journal of Graphics Tools*. 3(4): 1–11. DOI: 10.1080/10867651.1998.10487494.

Card, D. and J. L. Mitchell. 2002. "Non-Photorealistic Rendering with Pixel and Vertex Shaders". In: *In Direct3D ShaderX, Wordware*. Wordware Publishing, Inc. 319–333.

Cardona, L. and S. Saito. 2016. "Temporally Coherent and Artistically Intended Stylization of Feature Lines Extracted from 3D Models". *Comput. Graph. Forum*. 35(7): 137–146. ISSN: 0167-7055. DOI: 10.1111/cgf.13011.

Cardona, L. and S. Saito. 2015. "Hybrid-space Localized Stylization Method for View-dependent Lines Extracted from 3D Models". In: *Proceedings of the Workshop on Non-Photorealistic Animation and Rendering. NPAR '15.* Eurographics Association. 79–89.

Catmull, E. and J. H. Clark. 1978. "Recursively generated B-spline surfaces on arbitrary topological meshes". *Computer-Aided Design*. 10(6): 350–355. ISSN: 0010-4485. DOI: 10.1016/0010-4485(78)90110-0.

Chen, D., Y. Zhang, H. Liu, and P. Xu. 2015a. "Real-Time Artistic Silhouettes Rendering for 3D Models". In: *8th International Symposium on Computational Intelligence and Design*. Vol. 1. 494–498. DOI: 10.1109/ISCID.2015.201.

Chen, Z., B. Kim, D. Ito, and H. Wang. 2015b. “Wetbrush: GPU-based 3D Painting Simulation at the Bristle Level”. *ACM Trans. Graph.* 34(6): 200:1–200:11. ISSN: 0730-0301. DOI: 10.1145/2816795.2818066.

Choudhury, A. I. and S. G. Parker. 2009. “Ray tracing NPR-style feature lines”. In: *NPAR ’09: Proceedings of the 7th International Symposium on Non-Photorealistic Animation and Rendering.* ACM. 5–14. DOI: 10.1145/1572614.1572616.

Cipolla, R. and P. Giblin. 2000. *Visual Motion of Curves and Surfaces.* Cambridge University Press. ISBN: 0-521-63251-X.

Cole, F., M. Burns, K. Morley, A. Finkelstein, and P. Bénard. 2010. “dpix”. https://gfx.cs.princeton.edu/proj/dpix/, fork: https://github.com/benardp/dpix.

Cole, F. and A. Finkelstein. 2008. “Partial Visibility for Stylized Lines”. In: *Proceedings of the 6th International Symposium on Non-photorealistic Animation and Rendering. NPAR ’08.* ACM. 9–13. ISBN: 978-1-60558-150-7. DOI: 10.1145/1377980.1377985.

Cole, F. and A. Finkelstein. 2010. “Two Fast Methods for High-Quality Line Visibility”. *IEEE Transactions on Visualization and Computer Graphics.* 16(5): 707–717. DOI: 10.1109/TVCG.2009.102.

Cole, F., A. Golovinskiy, A. Limpaecher, H. S. Barros, A. Finkelstein, T. Funkhouser, and S. Rusinkiewicz. 2008. “Where Do People Draw Lines?” *ACM Trans. Graph.* 27(3): 88:1–88:11. DOI: 10.1145/1360612.1360687.

Cole, F., S. Rusinkiewicz, and D. DeCarlo. 2011. “qrts, a port of the original “rtsc” software”. https://github.com/benardp/qrtsc.

Cole, F., K. Sanik, D. DeCarlo, A. Finkelstein, T. Funkhouser, S. Rusinkiewicz, and M. Singh. 2009. “How Well Do Line Drawings Depict Shape?” *ACM Trans. Graph.* 28(3): 28:1–28:9. DOI: 10.1145/1531326.1531334.

Crane, K., F. de Goes, M. Desbrun, and P. Schröder. 2013. “Digital Geometry Processing with Discrete Exterior Calculus”. In: *ACM SIGGRAPH 2013 courses. SIGGRAPH ’13.* Anaheim, California: ACM. URL: https://www.cs.cmu.edu/~kmcrane/Projects/DDG/.

Csébfalvi, B., L. Mroz, H. Hauser, A. König, and E. Gröller. 2001. "Fast Visualization of Object Contours by Non-Photorealistic Volume Rendering". *Computer Graphics Forum*. 20(3): 452–460. DOI: 10.1111/1467-8659.00538.

Curtis, C. J. 1998. "Loose and Sketchy Animation". In: *ACM SIGGRAPH 98 Electronic Art and Animation Catalog. SIGGRAPH '98*. ACM. 145–. ISBN: 1-58113-045-7. DOI: 10.1145/281388.281913.

Curtis, C. J., S. E. Anderson, J. E. Seims, K. W. Fleischer, and D. H. Salesin. 1997. "Computer-generated Watercolor". In: *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH '97*. ACM. 421–430. ISBN: 0-89791-896-7. DOI: 10.1145/258734.258896.

DeCarlo, D. 2012. "Depicting 3D shape using lines". In: *Proc. SPIE*. Vol. 8291. 8291 - 8291 –16. DOI: 10.1117/12.916463.

DeCarlo, D., A. Finkelstein, and S. Rusinkiewicz. 2004. "Interactive Rendering of Suggestive Contours with Temporal Coherence". In: *Proceedings of the 3rd International Symposium on Non-photorealistic Animation and Rendering. NPAR '04*. ACM. 15–145. DOI: 10.1145/987657.987661.

DeCarlo, D., A. Finkelstein, S. Rusinkiewicz, and A. Santella. 2003. "Suggestive Contours for Conveying Shape". *ACM Trans. Graph.* 22(3): 848–855. DOI: 10.1145/882262.882354.

DeCarlo, D. and S. Rusinkiewicz. 2007. "Highlight Lines for Conveying Shape". In: *Proceedings of the 5th International Symposium on Non-photorealistic Animation and Rendering. NPAR '07*. ACM. 63–70. DOI: 10.1145/1274871.1274881.

Decaudin, P. 1996. "Cartoon Looking Rendering of 3D Scenes". *Research Report* No. 2919. INRIA. URL: http://phildec.users.sf.net/Research/RR-2919.php.

Deussen, O. and T. Strothotte. 2000. "Computer-generated Pen-and-ink Illustration of Trees". In: *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH '00*. ACM Press/Addison-Wesley Publishing Co. 13–18. DOI: 10.1145/344779.344792.

Dobkin, D. P., A. R. Wilks, S. V. F. Levy, and W. P. Thurston. 1990. “Contour Tracing by Piecewise Linear Approximations”. *ACM Trans. Graph.* 9(4): 389–423. ISSN: 0730-0301. DOI: 10.1145/88560.88575.

Dooley, D. and M. F. Cohen. 1990. “Automatic Illustration of 3D Geometric Models: Lines”. In: *Proceedings of the 1990 Symposium on Interactive 3D Graphics. I3D ’90.* ACM. 77–82. DOI: 10.1145/91385.91422.

Ebert, D. and P. Rheingans. 2000. “Volume Illustration: Non-photorealistic Rendering of Volume Models”. In: *Proceedings of the Conference on Visualization ’00. VIS ’00.* IEEE Computer Society Press. 195–202. DOI: 10.1109/VISUAL.2000.885694.

Eisemann, E., S. Paris, and F. Durand. 2009. “A Visibility Algorithm for Converting 3D Meshes into Editable 2D Vector Graphics”. *ACM Trans. Graph.* 28(3): 83:1–83:8. ISSN: 0730-0301. DOI: 10.1145/1531326.1531389.

Eisemann, E., H. Winnemöller, J. C. Hart, and D. Salesin. 2008. “Stylized Vector Art from 3D Models with Region Support”. In: *Proceedings of the Nineteenth Eurographics Conference on Rendering. EGSR ’08.* Eurographics Association. 1199–1207. DOI: 10.1111/j.1467-8659.2008.01258.x.

Elber, G. 1995a. “Line Art Rendering via a Coverage of Isoparametric Curves”. *IEEE Transactions on Visualization and Computer Graphics.* 1(3): 231–239. DOI: 10.1109/2945.466718.

Elber, G. 1995b. “Line illustrations $\in$ Computer Graphics”. *The Visual Computer.* 11(6): 290–296. DOI: 10.1007/BF01898406.

Elber, G. 1998. “Line art illustrations of parametric and implicit forms”. *IEEE Transactions on Visualization and Computer Graphics.* 4(1): 71–81. ISSN: 1077-2626. DOI: 10.1109/2945.675655.

Elber, G. 2018. “The IRIT Modeling Environment”. http://www.cs.technion.ac.il/~irit/.

Elber, G. and E. Cohen. 1990. “Hidden Curve Removal for Free Form Surfaces”. In: *Proceedings of the 17th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH ’90.* ACM. 95–104. DOI: 10.1145/97879.97890.

Elber, G. and E. Cohen. 2006. “Probabilistic silhouette based importance toward line-art non-photorealistic rendering”. *The Visual Computer*. 22(9): 793–804. ISSN: 1432-2315. DOI: 10.1007/s00371-006-0065-8.

Elber, G. and M.-S. Kim. 2001. “Geometric Constraint Solver Using Multivariate Rational Spline Functions”. In: *Proceedings of the Sixth ACM Symposium on Solid Modeling and Applications. SMA ’01*. Ann Arbor, Michigan, USA: ACM. 1–10. ISBN: 1-58113-366-9. DOI: 10.1145/376957.376958.

Farid, H. and E. P. Simoncelli. 1997. “Optimally rotation-equivariant directional derivative kernels”. In: *Computer Analysis of Images and Patterns, CAIP ’97*. Springer Berlin Heidelberg. 207–214. ISBN: 978-3-540-69556-1. DOI: 10.1007/3-540-63460-6_119.

Favreau, J.-D., F. Lafarge, and A. Bousseau. 2016. “Fidelity vs. Simplicity: A Global Approach to Line Drawing Vectorization”. *ACM Trans. Graph.* 35(4): 120:1–120:10. ISSN: 0730-0301. DOI: 10.1145/2897824.2925946.

Fišer, J., O. Jamriška, M. Lukáč, E. Shechtman, P. Asente, J. Lu, and D. Sýkora. 2016. “StyLit: Illumination-guided Example-based Stylization of 3D Renderings”. *ACM Trans. Graph.* 35(4): 92:1–92:11. ISSN: 0730-0301. DOI: 10.1145/2897824.2925948.

Fogel, E., D. Halperin, and R. Wein. 2012. *CGAL Arrangements and Their Applications: A Step-by-Step Guide*. Springer Publishing Company, Incorporated. ISBN: 3642172822.

Foster, K., P. Jepp, B. Wyvill, M. Sousa, C. Galbraith, and J. Jorge. 2005. “Pen-and-Ink for BlobTree Implicit Models”. *Computer Graphics Forum*. 24(3): 267–276. DOI: 10.1111/j.1467-8659.2005.00851.x.

Foster, K., M. C. Sousa, F. F. Samavati, and B. Wyvill. 2007. “Polygonal Silhouette Error Correction: a Reverse Subdivision Approach”. *International Journal of Computational Science and Engineering*. 3(1): 53–70. ISSN: 1742-7185. DOI: 10.1504/IJCSE.2007.014465.

Fuchs, H., Z. M. Kedem, and B. F. Naylor. 1980. “On Visible Surface Generation by a Priori Tree Structures”. In: *Proceedings of the 7th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH ’80*. ACM. 124–133. ISBN: 0-89791-021-4. DOI: 10.1145/800250.807481.

Gangnet, M., J.-C. Hervé, T. Pudet, and J.-M. van Thong. 1989. “Incremental Computation of Planar Maps”. In: *Proceedings of the 16th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH ’89*. ACM. 345–354. ISBN: 0-89791-312-4. DOI: 10.1145/74333.74369.

Gerl, M. and T. Isenberg. 2013. “Interactive Example-based Hatching”. *Computers & Graphics*. 37(1–2): 65–80. DOI: 10.1016/j.cag.2012.11.003.

Glisse, M. 2006. “An Upper Bound on the Average Size of Silhouettes”. In: *Proceedings of the Twenty-second Annual Symposium on Computational Geometry. SCG ’06*. ACM. 105–111. ISBN: 1-59593-340-9. DOI: 10.1145/1137856.1137874.

Gombrich, E. H. 1961. *Art and Illusion: A Study in the Psychology of Pictorial Representation*. 2nd. Princeton University Press.

Gooch, A. 1998. “Interactive non-photorealistic technical illustration”. *MA thesis*. Dept. of Computer Science, University of Utah.

Gooch, B., P.-P. J. Sloan, A. Gooch, P. Shirley, and R. Riesenfeld. 1999. “Interactive Technical Illustration”. In: *Proceedings of the 1999 Symposium on Interactive 3D Graphics. I3D ’99*. New York, NY, USA: ACM. 31–38. DOI: 10.1145/300523.300526.

Goodman, N. 1968. *Languages of Art: An Approach to a Theory of Symbols*. The Bobbs-Merrill Company, Inc.

Goodwin, T., I. Vollick, and A. Hertzmann. 2007. “Isophote Distance: A Shading Approach to Artistic Stroke Thickness”. In: *Proceedings of the 5th International Symposium on Non-photorealistic Animation and Rendering. NPAR ’07*. ACM. 53–62. DOI: 10.1145/1274871.1274880.

Grabli, S., F. Durand, and S. F. X. 2004. “Density measure for line-drawing simplification”. In: *Computer Graphics and Applications, 2004. PG 2004. Proceedings. 12th Pacific Conference on*. 309–318. DOI: 10.1109/PCCGA.2004.1348362.

Grabli, S., E. Turquin, F. Durand, and F. X. Sillion. 2010. “Programmable Rendering of Line Drawing from 3D Scenes”. *ACM Trans. Graph.* 29(2): 18:1–18:20. DOI: 10.1145/1731047.1731056.

Haase, C. S. and G. W. Meyer. 1992. “Modeling Pigmented Materials for Realistic Image Synthesis”. *ACM Trans. Graph.* 11(4): 305–335. ISSN: 0730-0301. DOI: 10.1145/146443.146452.

Haeberli, P. 1990. “Paint by Numbers: Abstract Image Representations”. In: *Proceedings of the 17th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH ’90.* ACM. 207–214. ISBN: 0-89791-344-2. DOI: 10.1145/97879.97902.

Hermosilla, P. and P. P. Vázquez. 2009. “Single pass GPU stylized edges”. In: *IV Iberoamerican Symposium in Computer Graphics.* 47–54.

Hertzmann, A. 1999. “Introduction to 3D Non-Photorealistic Rendering: Silhouettes and Outlines”. In: *ACM SIGGRAPH 99 Course Notes. Course on Non-Photorelistic Rendering.* Ed. by S. Green. ACM Press. URL: http://mrl.nyu.edu/publications/npr-course1999/.

Hertzmann, A. 2010. “Non-Photorealistic Rendering and the Science of Art”. In: *Proceedings of the 8th International Symposium on Non-Photorealistic Animation and Rendering. NPAR ’10.* ACM. 147–157. ISBN: 978-1-4503-0125-1. DOI: 10.1145/1809939.1809957.

Hertzmann, A., C. E. Jacobs, N. Oliver, B. Curless, and D. H. Salesin. 2001. “Image Analogies”. In: *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH ’01.* ACM. 327–340. ISBN: 1-58113-374-X. DOI: 10.1145/383259.383295.

Hertzmann, A., N. Oliver, B. Curless, and S. M. Seitz. 2002. “Curve Analogies”. In: *Proceedings of the 13th Eurographics Workshop on Rendering. EGRW ’02.* Eurographics Association. 233–246. ISBN: 1-58113-534-3. URL: http://dl.acm.org/citation.cfm?id=581896.581926.

Hertzmann, A. and D. Zorin. 2000. “Illustrating Smooth Surfaces”. In: *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH ’00.* ACM Press/Addison-Wesley Publishing Co. 517–526. DOI: 10.1145/344779.345074.

Hornung, C., W. Lellek, P. Rehwald, and W. Strasser. 1985. “An area-oriented analytical visibility method for displaying parametrically defined tensor-product surfaces”. *Computer Aided Geometric Design.* 2(1-3): 197–205.

Houghton, E. G., R. F. Emnett, J. D. Factor, and C. L. Sabharwal. 1985. “Implementation of a Divide-and-conquer Method for Intersection of Parametric Surfaces”. *Comput. Aided Geom. Des.* 2(1-3): 173–183. ISSN: 0167-8396. DOI: 10.1016/0167-8396(85)90022-6.

Hsu, S. C. and I. H. H. Lee. 1994. “Drawing and Animation Using Skeletal Strokes”. In: *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH ’94*. New York, NY, USA: ACM. 109–118. DOI: 10.1145/192161.192186.

Ikits, M., J. Kniss, A. Lefohn, and C. Hansen. 2004. “Chapter 39, Volume Rendering Techniques”. In: *GPU Gems: Programming Techniques, Tips and Tricks for Real-Time Graphics*. Ed. by R. Fernando. Addison Wesley. 667–692.

Interrante, V., H. Fuchs, and S. Pizer. 1995. “Enhancing transparent skin surfaces with ridge and valley lines”. In: *Visualization, 1995. Visualization ’95. Proceedings., IEEE Conference on.* 52–59, 438. DOI: 10.1109/VISUAL.1995.480795.

Isenberg, T., N. Halper, and T. Strothotte. 2002. “Stylizing silhouettes at interactive rates: From silhouette edges to silhouette strokes”. In: *Computer Graphics Forum*. Vol. 21. No. 3. 249–258. DOI: 10.1111/1467-8659.00584.

Jeong, K., A. Ni, S. Lee, and L. Markosian. 2005. “Detail control in line drawings of 3D meshes”. *The Visual Computer*. 21(8): 698–706. DOI: 10.1007/s00371-005-0323-1.

Judd, T., F. Durand, and E. Adelson. 2007. “Apparent Ridges for Line Drawing”. *ACM Trans. Graph.* 26(3). DOI: 10.1145/1276377.1276401.

Kalnins, R. D., P. L. Davidson, and D. M. Bourguignon. 2007. “Jot”. http://jot.cs.princeton.edu, fork: https://github.com/benardp/jot-lib.

Kalnins, R. D., P. L. Davidson, L. Markosian, and A. Finkelstein. 2003. “Coherent Stylized Silhouettes”. *ACM Trans. Graph.* 22(3): 856–861. DOI: 10.1145/882262.882355.

Kalnins, R. D., L. Markosian, B. J. Meier, M. A. Kowalski, J. C. Lee, P. L. Davidson, M. Webb, J. F. Hughes, and A. Finkelstein. 2002. “WYSIWYG NPR: Drawing Strokes Directly on 3D Models”. In: *Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH ’02*. ACM. 755–762. DOI: 10.1145/566570.566648.

Kalogerakis, E., D. Nowrouzezahrai, S. Breslav, and A. Hertzmann. 2012. “Learning Hatching for Pen-and-Ink Illustration of Surfaces”. *ACM Trans. Graphics*. 31(1).

Kaplan, M. 2007. “Hybrid Quantitative Invisibility”. In: *Proceedings of the 5th International Symposium on Non-photorealistic Animation and Rendering. NPAR ’07*. ACM. 51–52. ISBN: 978-1-59593-624-0. DOI: 10.1145/1274871.1274879.

Karsch, K. and J. C. Hart. 2011. “Snaxels on a Plane”. In: *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Non-Photorealistic Animation and Rendering. NPAR ’11*. ACM. 35–42. DOI: 10.1145/2024676.2024683.

Kass, M., A. Witkin, and D. Terzopoulos. 1988. “Snakes: Active contour models”. *International Journal of Computer Vision*. 1(4): 321–331. ISSN: 1573-1405. DOI: 10.1007/BF00133570.

Kaufman, A. and K. Mueller. 2005. “Overview of Volume Rendering”. In: vol. 7. Butterworth-Heinemann. 127–174. ISBN: 978-0-12-387582-2. DOI: 10.1016/B978-012387582-2/50009-5.

Kettner, L. and E. Welzl. 1997. “Contour Edge Analysis for Polyhedron Projections”. In: *Geometric Modeling: Theory and Practice: The State of the Art*. Ed. by W. Strasser, R. Klein, and R. Rau. Springer Berlin Heidelberg. 379–394. ISBN: 978-3-642-60607-6. DOI: 10.1007/978-3-642-60607-6_25.

Kim, K.-J. and N. Baek. 2005. “Fast extraction of polyhedral model silhouettes from moving viewpoint on curved trajectory”. *Computers & Graphics*. 29(3): 393–402. ISSN: 0097-8493. DOI: 10.1016/j.cag.2005.03.009.

Kindlmann, G., R. Whitaker, T. Tasdizen, and T. Moller. 2003. “Curvature-Based Transfer Functions for Direct Volume Rendering: Methods and Applications”. In: *Proceedings of the 14th IEEE Visualization 2003 (VIS’03). VIS ’03.* IEEE Computer Society. 67–. DOI: 10.1109/VISUAL.2003.1250414.

Kirsanov, D., P. V. Sander, and S. J. Gortler. 2003. “Simple Silhouettes for Complex Surfaces”. In: *Proceedings of the 2003 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing. SGP ’03.* Eurographics Association. 102–106. ISBN: 1-58113-687-0.

Klein, A. W., W. Li, M. M. Kazhdan, W. T. Corrêa, A. Finkelstein, and T. A. Funkhouser. 2000. “Non-photorealistic Virtual Environments”. In: *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH ’00.* ACM Press/Addison-Wesley Publishing Co. 527–534. ISBN: 1-58113-208-5. DOI: 10.1145/344779.345075.

Kobbelt, L. P., K. Daubert, and H.-P. Seidel. 1998. “Ray Tracing of Subdivion Surfaces”. In: *Rendering Techniques.*

Koenderink, J. J. 1984. “What does the occluding contour tell us about solid shape?” *Perception.* 13(3): 321–330. DOI: 10.1068/p130321.

Koenderink, J. J. and A. J. van Doorn. 1982. “The shape of smooth objects and the way contours end”. *Perception.* 11(2): 129–137. DOI: 10.1068/p110129.

Kolliopoulos, A., J. M. Wang, and A. Hertzmann. 2006. “Segmentation-based 3D Artistic Rendering”. In: *Proceedings of the 17th Eurographics Conference on Rendering Techniques. EGSR ’06.* Eurographics Association. 361–370. ISBN: 3-905673-35-5. DOI: 10.2312/EGWR/EGSR06/361-370.

Kolomenkin, M., I. Shimshoni, and A. Tal. 2008. “Demarcating Curves for Shape Illustration”. *ACM Trans. Graph.* 27(5): 157:1–157:9. DOI: 10.1145/1409060.1409110.

Kubelka, P. 1948. “New contributions to the optics of intensely light-scattering materials. Part I”. *Josa.* 38(5): 448–457.

Lake, A., C. Marshall, M. Harris, and M. Blackstein. 2000. “Stylized Rendering Techniques for Scalable Real-time 3D Animation”. In: *Proceedings of the 1st International Symposium on Non-photorealistic Animation and Rendering. NPAR ’00.* ACM. 13–20. DOI: 10.1145/340916.340918.

Lang, K. and M. Alexa. 2015. “The Markov Pen: Online Synthesis of Free-hand Drawing Styles”. In: *Proceedings of the Workshop on Non-Photorealistic Animation and Rendering. NPAR ’15.* Eurographics Association. 203–215.

Lawonn, K., I. Viola, B. Preim, and T. Isenberg. 2018. “A Survey of Surface-Based Illustrative Rendering for Visualization”. *Computer Graphics Forum.* ISSN: 1467-8659. DOI: 10.1111/cgf.13322.

Lee, Y., L. Markosian, S. Lee, and J. F. Hughes. 2007. “Line Drawings via Abstracted Shading”. *ACM Trans. Graph.* 26(3). DOI: 10.1145/1276377.1276400.

Leister, W. 1994. “Computer generated copper plates”. In: *Computer Graphics Forum.* Vol. 13. No. 1. Wiley Online Library. 69–77.

Lichtenbelt, B., R. Crane, and S. Naqvi. 1998. *Introduction to Volume Rendering.* Prentice-Hall, Inc. ISBN: 0-13-861683-3.

Loop, C. 1987. “Smooth Subdivision Surfaces Based on Triangles”. *MA thesis.* Department of Mathematics, The University of Utah.

Lorensen, W. E. and H. E. Cline. 1987. “Marching Cubes: A High Resolution 3D Surface Construction Algorithm”. In: *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH ’87.* ACM. 163–169. ISBN: 0-89791-227-6. DOI: 10.1145/37401.37422.

Lu, J., C. Barnes, S. DiVerdi, and A. Finkelstein. 2013. “RealBrush: Painting with Examples of Physical Media”. *ACM Trans. Graph.* 32(4): 117:1–117:12. ISSN: 0730-0301. DOI: 10.1145/2461912.2461998.

Lu, J., C. Barnes, C. Wan, P. Asente, R. Mech, and A. Finkelstein. 2014. “DecoBrush: Drawing Structured Decorative Patterns by Example”. *ACM Trans. Graph.* 33(4): 90:1–90:9. ISSN: 0730-0301. DOI: 10.1145/2601097.2601190.

Lum, E. B. and K.-L. Ma. 2002. “Hardware-accelerated Parallel Nonphotorealistic Volume Rendering”. In: *Proceedings of the 2nd International Symposium on Non-photorealistic Animation and Rendering. NPAR ’02*. ACM. 67–ff. DOI: 10.1145/508530.508542.

Mamassian, P. and M. S. Landy. 1998. “Observer biases in the 3D interpretation of line drawings”. *Vision Research*. 38: 2817–2832.

Markosian, L., M. A. Kowalski, D. Goldstein, S. J. Trychin, J. F. Hughes, and L. D. Bourdev. 1997. “Real-time Nonphotorealistic Rendering”. In: *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH ’97*. ACM Press/Addison-Wesley Publishing Co. 415–420. DOI: 10.1145/258734.258894.

Marr, D. 1977. “Analysis of occluding contour”. *Proceedings of the Royal Society of London B: Biological Sciences*. 197(1129): 441–475. ISSN: 0080-4649. DOI: 10.1098/rspb.1977.0080.

Marr, D. and E. Hildreth. 1980. “Theory of edge detection”. *Proceedings of the Royal Society of London B: Biological Sciences*. 207(1167): 187–217.

Masuch, M., S. Schlechtweg, and B. Schönwälder. 1997. “daLi! - Drawing Animated Lines!” In: *In Proceedings of Simulation und Animation ’97, SCS Europe*. 87–96.

Max, N. 1999. “Weights for Computing Vertex Normals from Facet Normals”. *Journal of Graphics Tools*. 4(2): 1–6. ISSN: 1086-7651. DOI: 10.1080/10867651.1999.10487501.

McGuire, M. 2004. “Observations on Silhouette Sizes”. *jgt*. 9(1): 1–12. URL: http://www.cs.brown.edu/research/graphics/games/SilhouetteSize/index.html.

McGuire, M. and J. F. Hughes. 2004. “Hardware-determined Feature Edges”. In: *Proceedings of the 3rd International Symposium on Non-photorealistic Animation and Rendering. NPAR ’04*. ACM. 35–47. DOI: 10.1145/987657.987663.

Meier, B. J. 1996. “Painterly Rendering for Animation”. In: *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH ’96*. ACM. 477–484. ISBN: 0-89791-746-4. DOI: 10.1145/237170.237288.

Meyer, M. D., P. Georgel, and R. T. Whitaker. 2005. “Robust Particle Systems for Curvature Dependent Sampling of Implicit Surfaces”. In: *Proceedings of the International Conference on Shape Modeling and Applications 2005. SMI ’05.* IEEE Computer Society. 124–133. ISBN: 0-7695-2379-X. DOI: 10.1109/SMI.2005.41.

Mitchell, J., M. Francke, and D. Eng. 2007. “Illustrative Rendering in Team Fortress 2”. In: *Proceedings of the 5th International Symposium on Non-photorealistic Animation and Rendering. NPAR ’07.* ACM. 71–76. ISBN: 978-1-59593-624-0. DOI: 10.1145/1274871.1274883.

Nagy, Z., J. Schneider, and R. Westermann. 2002. “Interactive Volume Illustration”. In: *Vision, Modeling and Visualization 2002.*

Ni, A., K. Jeong, S. Lee, and L. Markosian. 2006. “Multi-scale Line Drawings from 3D Meshes”. In: *Proceedings of the 2006 Symposium on Interactive 3D Graphics and Games. I3D ’06.* ACM. 133–137. DOI: 10.1145/1111411.1111435.

Nienhaus, M. and J. Döllner. 2004. “Blueprints: Illustrating Architecture and Technical Parts Using Hardware-accelerated Non-photorealistic Rendering”. In: *Proceedings of Graphics Interface 2004. GI ’04.* Canadian Human-Computer Communications Society. 49–56.

Nießner, M., C. Loop, M. Meyer, and T. Derose. 2012. “Feature-adaptive GPU Rendering of Catmull-Clark Subdivision Surfaces”. *ACM Trans. Graph.* 31(1): 6:1–6:11. ISSN: 0730-0301. DOI: 10.1145/2077341.2077347.

Northrup, J. D. and L. Markosian. 2000. “Artistic Silhouettes: A Hybrid Approach”. In: *Proceedings of the 1st International Symposium on Non-photorealistic Animation and Rendering. NPAR ’00.* ACM. 31–37. DOI: 10.1145/340916.340920.

O’Donovan, P. and A. Hertzmann. 2012. “AniPaint: interactive painterly animation from video”. *IEEE transactions on visualization and computer graphics.* 18(3): 475–87. ISSN: 1941-0506. DOI: 10.1109/TVCG.2011.51.

Ogaki, S. and I. Georgiev. 2018. “Production Ray Tracing of Feature Lines”. In: *SIGGRAPH Asia 2018 Technical Briefs. SA ’18.* ACM. 15:1–15:4. ISBN: 978-1-4503-6062-3. DOI: 10.1145/3283254.3283273.

Ohtake, Y., A. Belyaev, and H.-P. Seidel. 2004. “Ridge-valley Lines on Meshes via Implicit Surface Fitting”. *ACM Trans. Graph.* 23(3): 609–612. DOI: 10.1145/1015706.1015768.

Olson, M. and H. Zhang. 2006. “Silhouette Extraction in Hough Space”. *Computer Graphics Forum.* 25(3): 273–282. DOI: 10.1111/j.1467-8659.2006.00946.x.

Pauly, M., R. Keiser, and M. Gross. 2003. “Multi-scale Feature Extraction on Point-Sampled Surfaces”. *Computer Graphics Forum.* 22(3): 281–289. DOI: 10.1111/1467-8659.00675.

Pfister, H., M. Zwicker, J. van Baar, and M. Gross. 2000. “Surfels: Surface Elements As Rendering Primitives”. In: *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH ’00.* ACM Press/Addison-Wesley Publishing Co. 335–342. ISBN: 1-58113-208-5. DOI: 10.1145/344779.344936.

Pharr, M., W. Jakob, and G. Humphreys. 2016. *Physically Based Rendering: From Theory to Implementation.* 3rd. Morgan Kaufmann Publishers Inc. ISBN: 0128006455.

Pixar. 2019. “OpenSubdiv”. http://graphics.pixar.com/opensubdiv.

Plantinga, S. and G. Vegter. 2006. “Computing Contour Generators of Evolving Implicit Surfaces”. *ACM Trans. Graph.* 25(4): 1243–1280. DOI: 10.1145/1183287.1183288.

Pop, M., C. Duncan, G. Barequet, M. Goodrich, W. Huang, and S. Kumar. 2001. “Efficient Perspective-accurate Silhouette Computation and Applications”. In: *Proceedings of the Seventeenth Annual Symposium on Computational Geometry. SCG ’01.* ACM. 60–68. DOI: 10.1145/378583.378618.

Praun, E., H. Hoppe, M. Webb, and A. Finkelstein. 2001. “Real-time Hatching”. In: *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH ’01.* ACM. 581–. ISBN: 1-58113-374-X. DOI: 10.1145/383259.383328.

Preim, B. and T. Strothotte. 1995. “Tuning rendered line-drawings”. In: *WSCG’95.* 228–238.

Raskar, R. 2001. “Hardware Support for Non-photorealistic Rendering”. In: *Proceedings of the ACM SIGGRAPH/EUROGRAPHICS Workshop on Graphics Hardware. HWWS ’01.* ACM. 41–47. DOI: 10.1145/383507.383525.

Raskar, R. and M. Cohen. 1999. “Image Precision Silhouette Edges”. In: *Proceedings of the 1999 Symposium on Interactive 3D Graphics. I3D ’99*. ACM. 135–140. DOI: 10.1145/300523.300539.

Rideout, P. 2010. “Silhouette Extraction”. http://prideout.net/blog/?p=54.

Roberts, L. 1963. “Machine Perception of Three-Dimensional Solids”. *PhD thesis*. Massachusetts Institute of Technology. Dept. of Electrical Engineering.

Roettger, S. 2012. “The Volume Library”. http://schorsch.efi.fh-nuernberg.de/data/volume/.

Rosin, P. and J. Collomosse. 2013. *Image and Video-Based Artistic Stylisation*. Springer.

Rossignac, J. R. and M. van Emmerik. 1992. “Hidden Contours on a Frame-buffer”. In: *Proceedings of the Seventh Eurographics Conference on Graphics Hardware. EGGH’92*. Eurographics Association. 188–203. DOI: 10.2312/EGGH/EGGH92/188-203.

Rusinkiewicz, S. 2004. “Estimating Curvatures and Their Derivatives on Triangle Meshes”. In: *Proceedings of the 3D Data Processing, Visualization, and Transmission, 2Nd International Symposium. 3DPVT ’04*. IEEE Computer Society. 486–493. DOI: 10.1109/3DPVT.2004.54.

Rusinkiewicz, S., F. Cole, D. DeCarlo, and A. Finkelstein. 2008. “Line Drawings from 3D Models”. In: *ACM SIGGRAPH 2008 Classes. SIGGRAPH ’08*. New York, NY, USA: ACM. 39:1–39:356. DOI: 10.1145/1401132.1401188.

Sabiston, B. 2001. “Waking Life: Making Of”. URL: http://www.flatblackfilms.com/Flat_Black_Films/Rotoshop.html.

Saito, T. and T. Takahashi. 1990. “Comprehensible Rendering of 3-D Shapes”. In: *Proceedings of the 17th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH ’90*. ACM. 197–206. DOI: 10.1145/97879.97901.

Sander, P. V., X. Gu, S. J. Gortler, H. Hoppe, and J. Snyder. 2000. “Silhouette Clipping”. In: *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH ’00*. ACM Press/Addison-Wesley Publishing Co. 327–334. DOI: 10.1145/344779.344935.

Sander, P. V., D. Nehab, E. Chlamtac, and H. Hoppe. 2008. “Efficient Traversal of Mesh Edges Using Adjacency Primitives”. *ACM Trans. Graph.* 27(5): 144:1–144:9. DOI: 10.1145/1409060.1409097.

Sayim, B. and P. Cavanagh. 2011. “What Line Drawings Reveal About the Visual Brain”. *Frontiers in Human Neuroscience.* 5: 118. ISSN: 1662-5161. DOI: 10.3389/fnhum.2011.00118.

Schein, S. and G. Elber. 2004. “Adaptive Extraction and Visualization of Silhouette Curves from Volumetric Datasets”. *Vis. Comput.* 20(4): 243–252. DOI: 10.1007/s00371-003-0230-2.

Schmidt, R. 2008. “ShapeShop”. http://www.shapeshop3d.com/.

Schmidt, R., T. Isenberg, P. Jepp, K. Singh, and B. Wyvill. 2007. “Sketching, Scaffolding, and Inking: A Visual History for Interactive 3D Modeling”. In: *Proceedings of the 5th International Symposium on Non-photorealistic Animation and Rendering. NPAR ’07.* San Diego, California: ACM. 23–32. ISBN: 978-1-59593-624-0. DOI: 10.1145/1274871.1274875.

Shewchuk, J. R. 1997. “Adaptive Precision Floating-Point Arithmetic and Fast Robust Geometric Predicates”. *Discrete & Computational Geometry.* 18(3): 305–363.

Singh, M. and S. Schaefer. 2010. “Suggestive Hatching”. In: *Proc. Computational Aesthetics.*

Sloan, P.-P. J., W. Martin, A. Gooch, and B. Gooch. 2001. “The Lit Sphere: A Model for Capturing NPR Shading from Art”. In: *Proceedings of Graphics Interface 2001. GI ’01.* Canadian Information Processing Society. 143–150. ISBN: 0-9688808-0-0.

Sousa, M. C. and J. W. Buchanan. 1999. “Computer-Generated Graphite Pencil Rendering of 3D Polygonal Models”. *Computer Graphics Forum.* 18(3): 195–208. DOI: 10.1111/1467-8659.00340.

Sousa, M. C. and P. Prusinkiewicz. 2003. “A Few Good Lines: Suggestive Drawing of 3D Models”. *Computer Graphics Forum.* DOI: 10.1111/1467-8659.00685.

Stam, J. 1998a. “Evaluation of Loop Subdivision Surfaces”.

Stam, J. 1998b. “Exact Evaluation of Catmull-Clark Subdivision Surfaces at Arbitrary Parameter Values”. In: *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH ’98.* ACM. 395–404. ISBN: 0-89791-999-8. DOI: 10.1145/280814.280945.

Stich, M., C. Wächter, and A. Keller. 2007. “Efficient and Robust Shadow Volumes Using Hierarchical Occlusion Culling and Geometry Shaders”. In: *GPU Gems 3.* Addison-Wesley Professional. Chap. 11. ISBN: 9780321545428.

Stroila, M., E. Bachta, W. Jarosz, W. Su, and J. Hart. 2011. “Wickbert”. fork: https://github.com/benardp/Wickbert.

Stroila, M., E. Eisemann, and J. Hart. 2008. “Clip Art Rendering of Smooth Isosurfaces”. *IEEE Transactions on Visualization and Computer Graphics.* 14(1): 135–145. DOI: 10.1109/TVCG.2007.1058.

Tejima, T., M. Fujita, and T. Matsuoka. 2015. “Direct Ray Tracing of Full-Featured Subdivision Surfaces with Bezier Clipping”. *Journal of Computer Graphics Techniques (JCGT).* 4(1): 69–83. ISSN: 2331-7418. URL: http://jcgt.org/published/0004/01/04/.

Thibault, A. and S. Cavanaugh. 2010. “Making Concept Art Real for Borderlands”. In: *ACM SIGGRAPH 2010 Courses.* URL: http://www.cs.williams.edu/~morgan/SRG10.

Thirion, J.-P. and A. Gourdon. 1996. “The 3D Marching Lines Algorithm”. *Graphical Models and Image Processing.* 58(6): 503–509. DOI: 10.1006/gmip.1996.0042.

Vanderhaeghe, D., R. Vergne, P. Barla, and W. Baxter. 2011. “Dynamic Stylized Shading Primitives”. In: *NPAR ’11:Proceedings of the 8th International Symposium on Non-Photorealistic Animation and Rendering.* 99–104. DOI: 10.1145/2024676.2024693.

Váša, L., P. Vaněček, M. Prantl, V. Skorkovská, P. Martínek, and I. Kolingerová. 2016. “Mesh Statistics for Robust Curvature Estimation”. In: *Proceedings of the Symposium on Geometry Processing. SGP’16.* Eurographics Association. 271–280. DOI: 10.1111/cgf.12982.

Vergne, R., D. Vanderhaeghe, J. Chen, P. Barla, X. Granier, and C. Schlick. 2011. “Implicit Brushes for Stylized Line-based Rendering”. *Computer Graphics Forum.* 30(2): 513–522. DOI: 10.1111/j.1467-8659.2011.01892.x.

Webb, M., E. Praun, A. Finkelstein, and H. Hoppe. 2002. "Fine Tone Control in Hardware Hatching". In: *Proceedings of the 2Nd International Symposium on Non-photorealistic Animation and Rendering. NPAR '02*. ACM. 53–ff. ISBN: 1-58113-494-0. DOI: 10.1145/508530.508540.

Weghorst, H., G. Hooper, and D. P. Greenberg. 1984. "Improved Computational Methods for Ray Tracing". *ACM Trans. Graph.* 3(1): 52–69. ISSN: 0730-0301. DOI: 10.1145/357332.357335.

Weiss, R. A. 1966. "BE VISION, A Package of IBM 7090 FORTRAN Programs to Draw Orthographic Views of Combinations of Plane and Quadric Surfaces". *Journal of the ACM*. 13(2): 194–204. ISSN: 0004-5411. DOI: 10.1145/321328.321330.

Whited, B., E. Daniels, M. Kaschalk, P. Osborne, and K. Odermatt. 2012. "Computer-assisted Animation of Line and Paint in Disney's Paperman". In: *Proc. SIGGRAPH Talks*. Los Angeles, California: ACM. ISBN: 978-1-4503-1683-5. DOI: 10.1145/2343045.2343071.

Willats, J. 1997. *Art and Representation: New Princples in the Analysis of Pictures*. Princeton University Press.

Willats, J. and F. Durand. 2005. "Defining pictorial style: Lessons from linguistics and computer graphics". *Axiomathes*. 15(3): 319–351. DOI: 10.1007/s10516-004-5449-7.

Wilson, B. and K.-L. Ma. 2004. "Rendering Complexity in Computer-generated Pen-and-ink Illustrations". In: *Proceedings of the 3rd International Symposium on Non-photorealistic Animation and Rendering. NPAR '04*. ACM. 129–137. ISBN: 1-58113-887-3. DOI: 10.1145/987657.987674.

Winkenbach, G. and D. H. Salesin. 1994. "Computer-generated Pen-and-ink Illustration". In: *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH '94*. ACM. 91–100. DOI: 10.1145/192161.192184.

Winkenbach, G. and D. H. Salesin. 1996. "Rendering Parametric Surfaces in Pen and Ink". In: *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH '96*. ACM. 469–476. DOI: 10.1145/237170.237287.

Winnemöller, H., J. E. Kyprianidis, and S. C. Olsen. 2012. "XDoG: An eXtended difference-of-Gaussians compendium including advanced image stylization". *Computers & Graphics*. 36(6): 740–753. ISSN: 0097-8493. DOI: 10.1016/j.cag.2012.03.004.

Witkin, A. P. and P. S. Heckbert. 1994. "Using Particles to Sample and Control Implicit Surfaces". In: *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH '94*. ACM. 269–277. ISBN: 0-89791-667-0. DOI: 10.1145/192161.192227.

Xie, X., Y. He, F. Tian, H.-S. Seah, X. Gu, and H. Qin. 2007. "An Effective Illustrative Visualization Framework Based on Photic Extremum Lines (PELs)". *IEEE Transactions on Visualization and Computer Graphics*. 13(6): 1328–1335. DOI: 10.1109/TVCG.2007.70538.

Xu, H., M. X. Nguyen, X. Yuan, and B. Chen. 2004. "Interactive Silhouette Rendering for Point-based Models". In: *Proceedings of the First Eurographics Conference on Point-Based Graphics. SPBG'04*. Eurographics Association. 13–18. DOI: 10.2312/SPBG/SPBG04/013-018.

Yessios, C. I. 1979. "Computer Drafting of Stones, Wood, Plant and Ground Materials". In: *Proceedings of the 6th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH '79*. ACM. 190–198. ISBN: 0-89791-004-4. DOI: 10.1145/800249.807443.

Yoshizawa, S., A. Belyaev, H. Yokota, and H.-P. Seidel. 2007. "Fast and Faithful Geometric Algorithm for Detecting Crest Lines on Meshes". In: *Proceedings of the 15th Pacific Conference on Computer Graphics and Applications. PG '07*. IEEE Computer Society. 231–237. DOI: 10.1109/PG.2007.24.

Zhang, L., Y. He, X. Xie, and W. Chen. 2009. "Laplacian Lines for Real-time Shape Illustration". In: *Proceedings of the 2009 Symposium on Interactive 3D Graphics and Games. I3D '09*. ACM. 129–136. DOI: 10.1145/1507149.1507170.

Zhang, L., Q. Sun, and Y. He. 2014. "Splatting Lines: An Efficient Method for Illustrating 3D Surfaces and Volumes". In: *Proceedings of the 18th Meeting of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games. I3D '14*. ACM. 135–142. DOI: 10.1145/2556700.2556703.

Zhang, L., J. Xia, X. Ying, Y. He, W. Mueller-Wittig, and H.-S. Seah. 2012. "Efficient and robust 3D line drawings using difference-of-Gaussian". *Graphical Models*. 74(4): 87–98. DOI: 10.1016/j.gmod.2012.03.006.

Zorin, D. and P. Schröder. 2000. "Subdivision for Modeling and Animation". In: *ACM SIGGRAPH 2000 Courses*. URL: https://mrl.nyu.edu/publications/subdiv-course2000/.