Non-Photorealistic Animation and Rendering
Pierre Bénard and Holger Winnemöller (Editors)

EXPRESSIVE 2016

Interactive NPAR: What type of tools should we create?

Tobias Isenberg

Inria, France

Abstract

I argue that we need to increase our consideration of the interaction that is possible and/or needed for the NPAR algorithms we develop. Depending on the application domain of a given algorithmic contribution, different degrees of interaction are required to make it practically useful and, thus, relevant. The spectrum of interactivity ranges from (almost) fully automatic processing to levels of control that are similar to those of traditional tools—some of the approaches even needing to support the full spectrum. Only if these considerations are first-class members of the NPAR development process can we expect others to want to work with our tools and to use them on a regular basis.

Categories and Subject Descriptors (according to ACM CCS): Computing methodologies [Computer Graphics]: Rendering—Non-photorealistic rendering

1. Introduction

The field of non-photorealistic rendering was initially inspired (at least in part) by the insight that there is more to the idea of computer graphics than simply the dictate of the photographic camera. Starting from the iconic "first papers" of the field[†] such as Saito and Takahashi's "Comprehensible Rendering of 3-D Shapes" [ST90], Haeberli's "Paint by Numbers" [Hae90], or Dooley and Cohen's "Automatic Illustration of 3D Geometric Models" [DC90a, DC90b], NPAR researchers have contributed many "non-photorealistic" rendering and animation techniques. In doing so they cover the recreation or simulation of traditional artistic media, they enable completely new forms of expression, and they assist the illustration and visualization of data.

To date, while there is certainly continued interest and work in the field, it can be argued [GLJ*10] that researchers have created well-performing techniques for simulating many if not most of the established types of traditional media (watercolor, oil painting, pencil drawing, and many more) as well as for many ways to assist data illustration and visualization. Several books [GG01, SS02, Gen10, RC13], surveys (e. g., [LS95, Her03, IFH*03, BBT11, DeC12, HGT13, KCWI13, Ise15, LP15]), and many years of proceedings from NPAR and related conferences and journals are evidence of this extended body of work. To date, however, I[‡] argue that most contributions to the field have concentrated on the creation of rendering (or animation) techniques. In contrast, less of a focus has been placed in the past on how to allow the targeted users of the technique to interact, even if most NPAR techniques have an interactive component. In this paper I thus analyze the state of the art of interaction with non-photorealistic rendering and propose a set of goals to work toward as we create, implement, and deploy future NPAR techniques. These goals then have implications for us as researchers as we implement tools, in particular if these are to be used by real people and for real tasks.

2. Discussion of Interactive NPAR in the Past

The discussion of the use and design of interaction for non-photorealistic rendering was started by Salesin in his 2002 keynote [Sal02] at the annual NPAR conference. As part of his seven grand challenges for the field, he postulated as the fourth challenge: "Interactivity—How do you build tools for »right-brained« thinking?" Salesin argued that interactive NPAR tools "should let artists and computers each do what they are good at," "need to be simple yet flexible," and "should support full design cycle" of creation, evaluation, and reworking. Salesin thus saw interactivity within NPAR primarily from the perspective of professional artists creating artwork (such as his example of an art director working on an animated CG movie), arguably only one of several potential application domains of NPAR work.

Gooch et al. [GLJ*10] revisited Salesin's challenge in their meta paper at NPAR 2010, finding that "interaction is still one of the most difficult research paradigms." In contrast to Salesin, however, they state that "interaction tools [should support] both sides of the brain." They argue that there is a need both for interaction for artists

[†] I acknowledge that it can rightfully be argued that there were several if not many contributions to NP(A)R before 1990.

[‡] Like others have done it in similar position papers [Goo10, Her10], I use the personal pronoun "I" when I talk about my own personal views, while I write "we"/"our" when I refer to work I have done jointly with others or for refering to the NPAR community as a whole.

as well as for people not trained in the arts, in particular as NPAR matures according to Heinlein's [Hei85] model of maturation of a field as Gooch et al. discuss. I agree and argue that, as we work toward the goal of achieving the seamless integration of NPAR into our everyday lives, we need to focus on the interaction with the algorithmic support that we are creating. This task has certainly been achieved in some fields such as the movie industry where stylized animated movies have now virtually replaced traditional cel animation. As researchers, however, we should focus—in addition to working on algorithmic contributions—on how to design the interaction with the algorithmic support. This interaction can take many different forms—highly dependent on the targeted application domain.

In addition to (and largely independent from) this discussion of interactivity within 'pure' NPAR, Lum and Ma [LM02] make the point that interactivity is essential (also) for "expressive visualization." They argue that, for expressive visualization applications, users are typically scientists who lack a detailed background in illustration and who thus greatly benefit from the ability to explore the parameter space of an expressive visualization technique. Based on the trend of GPU-supported rendering that started in the early 2000s and which makes many high-quality NPAR-enriched visualization techniques possible, Lum and Ma explain that more suitable visualizations can be made available to people in the various application domains. However, Lum and Ma concentrate only on this aspect of image computation at interactive frame rates, rather than exploring the space of interaction design for expressive visualization techniques.

In addition to those three major contributions to the discussion of interactivity in NPAR, there have certainly been many papers and discussions before and since that relate to various aspects of interactivity. For example, interaction was discussed at the NPAR 2010 panel discussions based on that year's position papers (in particular, [GLJ*10]). Also some of the previously mentioned surveys [KCWI13] specifically summarize aspects of interactivity. Below I mention a number of additional papers that touch on aspects of interactivity, integrated into my general discussion. A full state-of-the-art report, however, is beyond the scope of this meta paper.

3. Goals/Application Domains of NPAR

Let us thus recall the different application domains that we typically have in mind as we work toward new technical contributions in NPAR. The support of specific target users in these application domains, one may argue, drive the goals we pursue[§] as we are creating new NPAR techniques.

One of the main goals for NPAR work is certainly the **support of artists (or illustrators)** as envisioned by Salesin [Sal02]. For them, NPAR tools promise to make their life easier—i. e., "lose the technical directors" as Salesin put it. For example, NPAR can assist in traditional cel animation with computer-supported in-betweening [FBC*95, Kor02, SDC09a, WNS*10] or interactive cartoon colorization [SDC09b], can provide (photo-)realistic [sic!] simulations of media such as watercolor [CAS*97, BKTS06, LD06, DKMI13] and oil paint [BSLM01, BWL04], can provide example-based stippling for illustrators to focus on higher-level illustration goals [MALI11], and can add benefits such as layers, undo, and redo in computer-based drawing tools such as Adobe's Photoshop. NPAR also creates new forms of expression as the previously mentioned fully computer-animated movies. Yet, in all these cases the job of the NPAR tool is to take care of the lower-level and otherwise tedious tasks, so that the artists or illustrator can concentrate on the higher-level artistic exploration and, finally, decisions (i. e., "let artists and computers each do what they are good at" [Sal02]). However, artists do not only need high-level control over an algorithm encapsulated in some sort of black box, they want to be working with the created tools on various levels of granularity (and not just the ability to tweak some parameters in a program)—just like they are used to work with traditional tools such as pens and brushes [BSM88, Mei99, Sei99]. An example of an interactive combination of such traditional tool-based input with the capabilities of NPAR-based processing are the comprehensive physical watercolor and oil painting systems by Van Laerhoven/Van Reeth and colleagues [VVLC*08, VCVL*09]—a setup that is aimed at people already (somewhat) skilled in the traditional technique. Similarly, comprehensive painting solutions for working in 3D scenes [Dan99, OS02, SSGS11] exist that are also aimed at professionals.

As Gooch et al. [GLJ*10] rightfully pointed out, however, another major application domain and thus goal for NPAR work is the **creation of tools for visual expression for non-artists**. Also those people not trained in the visual arts enjoy playing and interacting with artistic media—and for many researchers in our field the potential of NPAR to create tools for this purpose may be one of their main motivating factors to start their work in the first place. This target user group of non-artists also needs to be able to control the output of an NPAR technique at a high level but probably wants less fine-grained control than artists would want. We can thus assume that many people in this user group prefer tools in the form of filters that can easily be applied to visual media and that have some easy-to-control and easy-to-understand parameters. For example, Semmo et al.'s [SLKD15, SLKD16] oilpaint filter provides both automatic filtering as well as an interactive way to "paint" corrections to the automatic suggestion, similar to several other approaches that provide comparable interaction capabilities with more or less simple "filters" (e. g., [SWHS97, DHvOS00, SIMC07]). Other filter-based tools, in particular those for video stylization, naturally provide less active interaction capabilities (e. g., [WXSC04, WOG06]).[¶] In addition to image filters, some people also prefer tools that rather aim to provide support for simple drawing/painting (e. g., [CHZ00, Aga02, RLA*06]) or tools that transform photos into a drawing-like representation (e. g., [OG11]).

All of these tools can either be presented in traditional PC environments or, increasingly, also in form of apps on smartphones and tablet computers. For example, NPAR researchers have created apps

§ These application-driven goals complement those that drive our scientific interest in NPAR as summarized, for example, by Strothotte and Schlechtweg [SS02] in Section 1.2.1 of their book.

¶ Depending on the level of control these filter-based approaches provide, some may also be suited for use by professional artists.

such as PaintCan‖ (by Winnemöller et al.) for painting on tablets, while "normal" PC-based drawing and painting suites have also partially been ported to mobile environments (e. g., Adobe's Photoshop Express and variants of Autodesk's SketchBook) and many other similar apps to support drawing and painting exist as well. Also many mobile filter apps that stylize photos are available in the various app stores (e. g., PencilFX** by Gooch et al. and SnapDot†† based on Secord's [Sec02] weighted Voronoi stippling).

In addition to these first two categories, I also see one more potential goal for NPAR methods in the domain of (semi-)automatic visualization of data: the **creation of tools for scientists or illustrators producing illustrative visualizations of their data**. In illustrations, we similarly deal with the need to abstract and change the representation, just like in 'pure' NPAR. In visualization, we change the depiction for reasons such as to reduce the cognitive load, to integrate several layers of information, to indicate different data values including uncertainty, and/or to adjust the visualization to various types of (small or large) displays or formats. The goal of using illustrative depictions is thus either to better be able to explore data and to gain knowledge from it or to best be able to communicate the results of their research to others—to the general public or in scientific publications.

The field of illustrative visualization [RBGV08] (what was called "expressive visualization" by Lum and Ma [LM02]) has been established for this purpose, which is inspired by the long history of traditional illustration. Researchers in this sub-field thus develop methods—often inspired by NPAR approaches—to create insightful and understandable visual representations of data. This group of potential target users of NPAR techniques can be skilled in the visual arts (professional illustrators) or not (the typical scientist), so both "filters" with limited control and a lot of automated processing are useful (for scientists) as well as tools that, in addition to fully automatic processing, also provide full (artistic) control over the output (for illustrators).

For example, in our own past work [GRIG12] we use semantics by analogy to allow illustrators to create and adjust the transfer functions for illustrative volume rendering by brushing in such a way that they themselves assign meaning to the input and output parameters of the mapping—the semantics of the resulting visualizations depends on the interactive input [RBG08]. In another approach [GI13] we created a tool to assist illustrators with example-based algorithmic support for low-level mark creation as they work on hatching illustrations—allowing them to concentrate on the content of the illustration rather than where to place individual marks. Other dedicated tools exist for the interactive creation of illustrative visualizations of 3D sampled data (VolumeShop [BG05]) or the interactive authoring of cutaway illustrations [LRA*07], to name two examples.

As argued above, however, not only illustrators can be assisted by dedicated interaction with illustrative illustrations—also the scientists themselves as they explore new data, present research results to others, or as their results are used to create interactive educational materials for students at different levels. For example, a very challenging goal in this context is the interactive control of the abstraction level in an illustrative visualization. For molecular visualization, we created an approach to explore the abstraction level of the molecular structure, the visualization's support of perception, and the "illustrativeness" (stylization) [vdZLBI11]—an interactive exploration of a three-dimensional abstraction space. In a related approach, Parulek et al. [PRV13, PJR*14] demonstrated how to use different molecular surface representations at different scale levels and show how to seamlessly transition between them. We also recently demonstrated such control of abstraction for the illustrative visualization of brain connectivity [EBB*15].

A special case within this last category of goals are **medical doctors producing illustrations** from data they captured from patients, and who need **to inform their patients** of their health status or to explain an upcoming procedure to them. Such explanations can be expected to be easier when illustrative visualizations are used (compared to, for example, X-rays or MRI scans) because the general public is familiar with the style of biologic illustrations as found in text books through their normal education. In this special case, the interaction has to be restricted to very few types of input due to the limited time of doctors, from selecting maybe an illustration style out of a few possibilities to just pressing a button to automatically generate the illustration.

4. An Interaction Spectrum

These application domains and thus goals for NPAR show that the type of interactivity needed for an NPAR tool highly depends on the intended audience. While there is certainly a need for fully automatic processing, most approaches will require some sort of user input. In fact, in many cases the input needs range from—at the one end—almost fully automatic processing with only high-level input to—at the other end—very fine-grained control in which the artist controls the output at the level of single marks. To date, however, most NPAR tools have only supported one of these extremes, either high-level control of a complex system (e. g., "Sisley the Abstract Painter" [ZZ10]) or low-level drawing/painting tool simulation (e. g., "RealBrush" [LBDF13]). While some of the low-level approaches have made it into the tool palettes of digitally working artists (such as in Adobe Photoshop), the tools that provide higher-level control are often less appealing to artists due to their lack of control [BSM88, Mei99, Sei99, MC01, Win13].

4.1. The Challenge of System Complexity

One of the problems with providing control to the artist (or other users of NPAR) is that today's approaches are becoming increasingly complex systems, far from the simple "hacks" of early computer graphics. State-of-the-art algorithms for NPAR often have numerous parameters, in particular since computer simulations often provide more flexibility than the real world and can thus deviate from the physical reality by freely changing the constraints of a simulation. Some researchers encapsulate the resulting parameters by choosing meaningful defaults, others expose them at the expense of the resulting system becoming difficult to control without much training.

‖ http://www.paintcanapp.com/
** http://insatiablegenius.com/
†† http://www.snapdotapp.com/

Unfortunately, only the programer of a given tool is often able to truly operate it and produce meaningful output.

One potential option to address this problem is to provide input through drawing into parameter buffers that control two-dimensional parameter fields (e. g., [HH90, Her01, IMC06, SIMC07, Har07, TABI07]). Others have explored ways to search the parameter space by means of simulated annealing [XK08] and genetic algorithms, the latter either to optimize some criterion [CH06] or to let humans rate the results of each iteration [GGD05, Col06]. Even parameter control through observed emotional state [SBC06] in a form of "empathic painting" has been explored as a form of high-level control. Such high-level forms of interaction, however, provide only indirect forms of control over the result. To provide more control over the output (in particular for professional users such as artists or illustrators), we may try to provide comprehensive toolkits with *meaningful* access to parameters, in a similar way that traditional 3D modeling and vector graphics tools are designed.

4.2. Example-Based Operators vs. Media Simulations

Such toolkits alone, however, would still not solve the issue that our systems are becoming increasingly complex and that, ideally, we require interactive control at different levels of the interaction spectrum. In addition, any form of abstraction of the interaction away from the traditional tool may feel inadequate for artists and illustrators. So the question remains how we can provide high visual fidelity of the result and an acceptable abstraction of the interaction at the same time. One way to address this issue may be, instead of running physical simulations of the target medium, the use of example-based approaches that base their processing on previously recorded examples of the target style.[‡‡] Such approaches, similar to media simulations, are potentially able to satisfy the ever-increasing demand for (photo-)realism in NPAR.

Several example-based NPAR and illustrative visualization techniques have been introduced in the past. For example, researchers have developed techniques for stroke pattern synthesis [BBT*06, HLT*09], stylized animation [BCK*13], colorization of cartoons [SBŽ04], stylized strokes [KMM*02, LYFD12], volumetric multi-field dataset illustration [LE05, BBP10], portrait sketching [CLR*04], hatching [KNBH12, GI13], stippling [KMI*09, MALI11], and painting [WWYS04, LBDF13, LFB*13].[§§] In contrast to truthful media simulations, example-based NPAR techniques have the potential to allow both higher-level interaction and low-level control. This means that that—depending on the output and the requirements of the artist or illustrator—the degree of algorithmic control could be adjusted to the need of the user: If controlled at a higher level, such an approach could use algorithmic support to place, for example, stipple, hatching, or other stroke marks without the need for the artist to attend to each individual one of them. If

Figure 1: *Existing NPAR tools in the interaction spectrum.*

low-level control is desired, the technique could also allow artists or illustrators to adjust the result in high detail, down to the individual marks. Some researchers have attempted to support such large-scale to detailed adjustment in the past, for example Deussen/Hiller et al. [DHvOS00, HHD03] with their brush-based adjustment of stippling. Yet, a professional artist would probably prefer to be able to explicitly draw marks like with traditional pens, instead of using brushes that stochastically place new dots somewhere within their scope—even if that is very small.

For example, Hurtut et al. [HLT*09] describe specific interactive adjustments to their automatic stroke pattern synthesis technique that facilitate several types of lower-level control over the generated output, in contrast to the default use with only higher-level control. In our own example-based hatching technique [GI13] we also facilitate both a completely automatic example-based patch synthesis and an adjustment of the example-based synthesis. For instance, one can select different source patches or can adjust the hatching direction field. On top of such interactions one can envision the option to create new or adjust existing individual hatching strokes if that should be necessary to create better illustrations.

4.3. Mapping the Interaction Spectrum

I have to admit that we have not yet seen many (if any) tools that would truly make use of such a range of interactive control. In fact, most approaches are likely either at the medium- to high-level control end of the spectrum, with some natural media simulations ranging at the low end (Fig. 1). Moreover, each of them probably only covers a relatively small range of the spectrum itself, with only few exceptions such as the mentioned example-based techniques.

We can now ask what type of interactive support people in each of the mentioned application domains require. As argued above, artists and illustrators who are highly skilled likely would like a range of control, from low-level to medium/high-level (Fig. 2, top). Non-artists (i. e., the general public) who is less skilled in the traditional techniques that they are trying to replicate, as stated above, probably are happier with higher-level control, with less need of covering a wide range of the spectrum (Fig. 2, bottom). As they are getting more training, however, they will gradually transition to needing more low-level control, with less satisfaction from the results of very high-level interaction (Fig. 2, middle). So the expected skill level of the target audience of a tool or technique—and potentially the amount of time the audience is expected to spend with a tool—has an influence on the required range of support of the interaction spectrum.

In fact, we probably thus need to distinguish the different target audiences of our techniques more fine-grained way. For example,

‡‡ One can certainly come up with other and more compehensive taxonomies for classifying NPAR approaches [KCWI13]. In this specific case where the goal is a realistic recreation of a traditional technique I feel, however, that example-based methods and detailed simulations of the target medium are the two extremes to compare with each other.

§§ This list is not intended to be a complete survey.

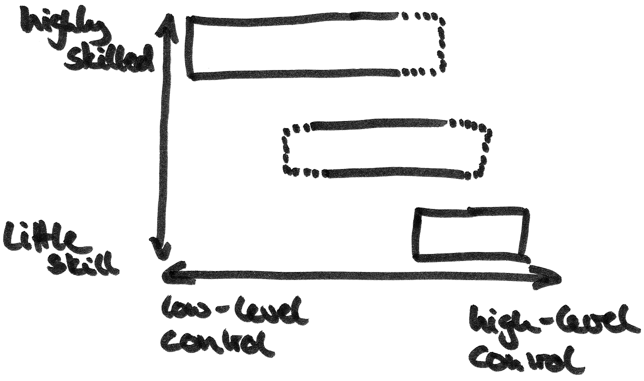

Figure 2: *Schematics of the interaction spectrum.*

there are professional artists, passionate artists, people who practice some art in their free time, non-artists who have started some training, and people who use NPARish tools just for their enjoyment—all of them with different skill levels, different needs for adjustments and editing, and different expectations with respect to the result and the usability of the tools they are working with. Laypeople who use an app on their smart phone or tablet, similarly, will have different skill levels and expectations. All of these different target audiences will require different types of tools, with different types of control ranges, and different forms of interactivity.

4.4. Lessons from HCI

Several of the thoughts in this section relate to the lessons that we can learn from general HCI research and, in particular, established design guidelines for tools that support creative thinking. For example, Resnick et al. [RMN*05] have put forward twelve design principles that should guide tool building for creatives. My concept of an interaction spectrum that ranges from high-level to low-level algorithmic support relates to, for instance, their guidelines to "choose black boxes carefully" as well as "low threshold, high ceiling, and wide walls"—the adjustable control level can make it easy for novices to try a technique as well as provide experts with fine-grained control. Their guideline of "invent things that you would want to use yourself" is probably even a driving force in many of us in NPAR research to begin with as mentioned earlier. Also, the guideline to "support many paths and many styles" encourages us to not limit ourselves to a single visual styles (and interaction mappings). Overall, Resnick et al.'s [RMN*05] guidelines touch on many design aspects that also more generally apply to human-computer interaction, including usability issues, the use of participatory design processes, and evaluation that go beyond this meta paper but which should certainly be observed when building actual tools—along with many other resources from the field of HCI and system design.

5. What Kind of Tools to Create?

So what kind of interactive NPAR tools should we strive to create? Let me put forward three theses about tools that I would like to see.

First, we should certainly strive to work more with artists and illustrators to better understand their needs, to be able to create tools that are truly useful for them. In that I am echoing previous calls for the same work that date already from the early time of NPAR research [BSM88, Mei99, Sei99]. But also today this demand is being articulated, for example Winnemöller [Win13] calls this type of work "user-centric NPR" and asks researchers to "put more emphasis on *assisting* art creation, rather than *automating* it." In the case of illustrative visualization, a user-centric approach means to talk to the respective target audience, i. e., the scientists who are supposed to use the illustrative visualization techniques in their regular work as also Lum and Ma [LM02] pointed out. So when we talk about creating NPAR tools for this purpose, as researchers we should thus not only work toward concepts and methods for better interactive use of NPAR algorithms—this type of work is only the (essential) basis for making NPAR work practically relevant to the target audience. We thus also need to try to create actual software tools that implement these concepts and methods. For such user-centric NPAR work to become possible we also need to think further about evaluating our results (and thus the concepts, methods, and actual software tools) in practice [Ise13], going beyond forms of validation that basically show a number of example results and argue that they look nice. Luckily, many recent publications in NPAR already involve user experiments and other advanced forms of evaluation.

Second, I would love to see (more) tools that provide interactive control that covers larger ranges of the mentioned interaction spectrum. In particular, I would like to see good combinations of higher-level algorithmic support and the ability to control the individual mark, without the two types of input appearing disconnected. The example-based techniques mentioned in Sect. 4.2 could be a starting point for such work. But I would not only like the spectrum to be extended to integrate more lower-level control. Instead, I would also like to see support for certain higher-level tasks that, in particular, illustrators have to do in their daily work. For example, in our work on scale-dependent and example-based stippling [MALI11] we discussed a number of high-level processes that are important for illustrators as they create an illustration. For instance, tools such as scene segmentation to support higher-level abstraction, tools to adjust the contrast for later stipple synthesis, tools for create intentionally irregular borders for natural context objects, tools to remove complexity in certain less important parts of the image, and tools to manually add additional detail where needed should be included in a holistic stippling-based tool for illustrators. It would be great to see interactive tools that incorporate explicit support for such higher-level tasks—not only for stippling but also for other artistic styles.

Third, but certainly not least, I would like to see interactive tools that are actually used in practice, success stories of NPAR in the hands of the target audiences. Here we should look beyond the (animated) movie and game industry as well as stylizing photo filters on smartphones where NPAR techniques are certainly successfully used [Win13]. For example, I would like to see more practical use of the techniques that we are creating in medical visualization and the illustration of research data in various domains of sciences. Illustrative visualization as a field is providing many specific techniques for this purpose, but similar to "core" NPAR work it would be great to see some real success stories—tools that are successful because they provide the right kind of interactivity for the people who need to and want to use them.

In conclusion, I believe that the question of interactivity in NPAR and related research is one of the challenges from Salesin's original list [Sal02] that is still standing. So I see this meta paper as a call to action for future work on NPAR needs to focus more on interaction techniques than it has in the past, in order to make our research truly relevant.

Acknowledgments

I gratefully acknowledge the valuable comments and feedback I received from Domingo Martín and Amir Semmo on earlier versions of this paper as well as the many conversations with colleagues on the topic. I would also like to thank the anonymous reviewers for their comments and suggestions on how to improve the paper.

References

[Aga02] AGARWALA A.: SnakeToonz: A semi-automatic approach to creating 2D cartoons from videos and images. In *Proc. NPAR* (2002), ACM, New York, pp. 139–146. doi: 10.1145/508530.508554

[BBP10] BUSKING S., BOTHA C. P., POST F. H.: Example-based interactive illustration of multi-field datasets. *Computers & Graphics 34*, 6 (Dec. 2010), 719–728. doi: 10.1016/j.cag.2010.07.004

[BBT*06] BARLA P., BRESLAV S., THOLLOT J., SILLION F. X., MARKOSIAN L.: Stroke pattern analysis and synthesis. *Computer Graphics Forum 25*, 3 (Sept. 2006), 663–671. doi: 10.1111/j.1467-8659.2006.00986.x

[BBT11] BÉNARD P., BOUSSEAU A., THOLLOT J.: State-of-the-art report on temporal coherence for stylized animations. *Computer Graphics Forum 30*, 8 (Dec. 2011), 2367–2386. doi: 10.1111/j.1467-8659.2011.02075.x

[BCK*13] BÉNARD P., COLE F., KASS M., MORDATCH I., HEGARTY J., SENN M. S., FLEISCHER K., PESARE D., BREEDEN K.: Stylizing animation by example. *ACM Transactions on Graphics 32*, 4 (July 2013), 119:1–119:12. doi: 10.1145/2461912.2461929

[BG05] BRUCKNER S., GRÖLLER E.: VolumeShop: An interactive system for direct volume illustration. In *Proc. Visualization* (2005), IEEE Computer Society, Los Alamitos, pp. 671–678. doi: 10.1109/VIS.2005.135

[BKTS06] BOUSSEAU A., KAPLAN M., THOLLOT J., SILLION F. X.: Interactive watercolor rendering with temporal coherence and abstraction. In *Proc. NPAR* (2006), ACM, New York, pp. 141–149. doi: 10.1145/1124728.1124751

[BSLM01] BAXTER B., SCHEIB V., LIN M. C., MANOCHA D.: DAB: Interactive haptic painting with 3D virtual brushes. In *Proc. SIGGRAPH* (2001), ACM, New York, pp. 461–468. doi: 10.1145/383259.383313

[BSM88] BLESER T. W., SIBERT J. L., MCGEE J. P.: Charcoal sketching: Returning control to the artist. *ACM Transactions on Graphics 7*, 1 (Jan. 1988), 76–81. doi: 10.1145/42188.42230

[BWL04] BAXTER W., WENDT J., LIN M. C.: IMPaSTo: A realistic, interactive model for paint. In *Proc. NPAR* (2004), ACM, New York, pp. 45–56. doi: 10.1145/987657.987665

[CAS*97] CURTIS C. J., ANDERSON S. E., SEIMS J. E., FLEISCHER K. W., SALESIN D. H.: Computer-generated watercolor. In *Proc. SIGGRAPH* (1997), ACM, New York, pp. 421–430. doi: 10.1145/258734.258896

[CH06] COLLOMOSSE J. P., HALL P. M.: Salience-adaptive painterly rendering using genetic search. *International Journal on Artificial Intelligence Tools 15*, 4 (Aug. 2006), 551–575. doi: 10.1142/S0218213006002813

[CHZ00] COHEN J. M., HUGHES J. F., ZELEZNIK R. C.: Harold: A world made of drawings. In *Proc. NPAR* (2000), ACM, New York, pp. 83–90. doi: 10.1145/340916.340927

[CLR*04] CHEN H., LIU Z., ROSE C., XU Y., SHUM H.-Y., SALESIN D. H.: Example-based composite sketching of human portraits. In *Proc. NPAR* (2004), ACM, New York, pp. 95–102. doi: 10.1145/987657.987673

[Col06] COLLOMOSSE J. P.: Supervised genetic search for parameter selection in painterly rendering. In *Proc. EvoMUSART* (2006), Springer, Berlin/Heidelberg, pp. 599–610. doi: 10.1007/11732242

[Dan99] DANIELS E.: Deep Canvas in Disney's Tarzan. In *ACM SIGGRAPH Conference Abstracts and Applications* (1999), ACM, New York, p. 200. doi: 10.1145/311625.312010

[DC90a] DOOLEY D. L., COHEN M. F.: Automatic illustration of 3D geometric models: Lines. In *Proc. I3D* (1990), ACM, New York, pp. 77–82. doi: 10.1145/91385.91422

[DC90b] DOOLEY D. L., COHEN M. F.: Automatic illustration of 3D geometric models: Surfaces. In *Proc. Visualization* (1990), IEEE Computer Society, Los Alamitos, pp. 307–314. doi: 10.1109/VISUAL.1990.146395

[DeC12] DECARLO D.: Depicting 3D Shape Using Lines. In *Proc. Human Vision and Electronic Imaging XVII* (2012), SPIE, Bellingham, Washington, pp. 829116:1–829116:16. doi: 10.1117/12.916463

[DHvOS00] DEUSSEN O., HILLER S., VAN OVERVELD C., STROTHOTTE T.: Floating points: A method for computing stipple drawings. *Computer Graphics Forum 19*, 3 (Aug. 2000), 40–51. doi: 10.1111/1467-8659.00396

[DKMI13] DIVERDI S., KRISHNASWAMY A., MĚCH R., ITO D.: Painting with polygons: A procedural watercolor engine. *IEEE Transactions on Visualization and Computer Graphics 19*, 5 (May 2013), 723–735. doi: 10.1109/TVCG.2012.295

[EBB*15] EVERTS M. H., BEGUE E., BEKKER H., ROERDINK J. B. T. M., ISENBERG T.: Exploration of the brain's white matter structure through visual abstraction and multi-scale local fiber tract contraction. *IEEE Transactions on Visualization and Computer Graphics 21*, 7 (July 2015), 808–821. doi: 10.1109/TVCG.2015.2403323

[FBC*95] FEKETE J.-D., BIZOUARN É., COURNARIE É., GALAS T., TAILLEFER F.: TicTacToon: A paperless system for professional 2-D animation. In *Proc. SIGGRAPH* (1995), ACM, New York, pp. 79–90. doi: 10.1145/218380.218417

[Gen10] GENG W.: *The Algorithms and Principles of Non-photorealistic Graphics: Artistic Rendering and Cartoon Animation*. Springer, Berlin, 2010. doi: 10.1007/978-3-642-04891-3

[GG01] GOOCH B., GOOCH A. A.: *Non-Photorealistic Rendering*. A K Peters, Ltd., Natick, 2001.

[GGD05] GRUNDLAND M., GIBBS C., DODGSON N. A.: Stylized rendering for multiresolution image representation. In *Proc. Human Vision and Electronic Imaging* (2005), vol. 5666, SPIE/IS&T, Bellingham, Washington, pp. 280–292. doi: 10.1117/12.596817

[GI13] GERL M., ISENBERG T.: Interactive example-based hatching. *Computers & Graphics 37*, 1–2 (Feb.–Apr. 2013), 65–80. doi: 10.1016/j.cag.2012.11.003

[GLJ*10] GOOCH A. A., LONG J., JI L., ESTEY A., GOOCH B. S.: Viewing progress in non-photorealistic rendering through Heinlein's lens. In *Proc. NPAR* (2010), ACM, New York, pp. 165–171. doi: 10.1145/1809939.1809959

[Goo10] GOOCH A. A.: Towards mapping the field of non-photorealistic rendering. In *Proc. NPAR* (2010), ACM, New York, pp. 159–164. doi: 10.1145/1809939.1809958

[GRIG12] GERL M., RAUTEK P., ISENBERG T., GRÖLLER E.: Semantics by analogy for illustrative volume visualization. *Computers & Graphics 36*, 3 (May 2012), 201–213. doi: 10.1016/j.cag.2011.10.006

[Hae90] HAEBERLI P.: Paint by numbers: Abstract image representations. *ACM SIGGRAPH Computer Graphics 24*, 3 (Aug. 1990), 207–214. doi: 10.1145/97879.97902

[Har07] HARVILL A.: Effective toon-style rendering control using scalar fields. In *ACM SIGGRAPH Conference Abstracts and Applications* (2007), ACM, New York, p. 53:1. doi: 10.1145/1278780.1278844

[Hei85] HEINLEIN R. A.: *The Rolling Stones*. Del Rey, 1985.

[Her01] HERTZMANN A.: Paint By Relaxation. In *Proc. CGI* (2001), IEEE Computer Society, Los Alamitos, pp. 47–54. doi: 10.1109/CGI.2001.934657

[Her03] HERTZMANN A.: A survey of stroke-based rendering. *IEEE Computer Graphics and Applications 23*, 4 (July/Aug. 2003), 70–81. doi: 10.1109/MCG.2003.1210867

[Her10] HERTZMANN A.: Non-photorealistic rendering and the science of art. In *Proc. NPAR* (2010), ACM, New York, pp. 147–157. doi: 10.1145/1809939.1809957

[HGT13] HEGDE S., GATZIDIS C., TIAN F.: Painterly rendering techniques: A state-of-the-art review of current approaches. *Computer Animation and Virtual Worlds 24*, 1 (Jan./Feb. 2013), 43–64. doi: 10.1002/cav.1435

[HH90] HANRAHAN P., HAEBERLI P.: Direct WYSIWYG painting and texturing of 3D shapes. *ACM SIGGRAPH Computer Graphics 24*, 3 (Aug. 1990), 215–223. doi: 10.1145/97880.97903

[HHD03] HILLER S., HELLWIG H., DEUSSEN O.: Beyond Stippling – Methods for distributing objects on the plane. *Computer Graphics Forum 22*, 3 (Sept. 2003), 515–522. doi: 10.1111/1467-8659.00699

[HLT*09] HURTUT T., LANDES P.-E., THOLLOT J., GOUSSEAU Y., DROUILHET R., COEURJOLLY J.-F.: Appearance-guided synthesis of element arrangements by example. In *Proc. NPAR* (2009), ACM, New York, pp. 51–60. doi: 10.1145/1572614.1572623

[IFH*03] ISENBERG T., FREUDENBERG B., HALPER N., SCHLECHTWEG S., STROTHOTTE T.: A developer's guide to silhouette algorithms for polygonal models. *IEEE Computer Graphics and Applications 23*, 4 (July/Aug. 2003), 28–37. doi: 10.1109/MCG.2003.1210862

[IMC06] ISENBERG T., MIEDE A., CARPENDALE S.: A buffer framework for supporting responsive interaction in information visualization interfaces. In *Proc. C^5* (2006), IEEE Computer Society, Los Alamitos, pp. 262–269. doi: 10.1109/C5.2006.4

[Ise13] ISENBERG T.: Evaluating and validating non-photorealistic and illustrative rendering. In *Image and Video based Artistic Stylisation*, Rosin P., Collomosse J., (Eds.), vol. 42 of *Computational Imaging and Vision*. Springer, London/Heidelberg, 2013, ch. 15, pp. 311–331. doi: 10.1007/978-1-4471-4519-6_15

[Ise15] ISENBERG T.: A survey of illustrative visualization techniques for diffusion-weighted MRI tractography. In *Visualization and Processing of Higher Order Descriptors for Multi-Valued Data*, Hotz I., Schulz T., (Eds.). Springer, Berlin/Heidelberg, 2015, ch. 12, pp. 235–256. doi: 10.1007/978-3-319-15090-1_12

[KCWI13] KYPRIANIDIS J. E., COLLOMOSSE J., WANG T., ISENBERG T.: State of the "art": A taxonomy of artistic stylization techniques for images and video. *IEEE Transactions on Visualization and Computer Graphics 19*, 5 (May 2013), 866–885. doi: 10.1109/TVCG.2012.160

[KMI*09] KIM S., MACIEJEWSKI R., ISENBERG T., ANDREWS W. M., CHEN W., SOUSA M. C., EBERT D. S.: Stippling by example. In *Proc. NPAR* (2009), ACM, New York, pp. 41–50. doi: 10.1145/1572614.1572622

[KMM*02] KALNINS R. D., MARKOSIAN L., MEIER B. J., KOWALSKI M. A., LEE J. C., DAVIDSON P. L., WEBB M., HUGHES J. F., FINKELSTEIN A.: WYSIWYG NPR: Drawing strokes directly on 3D models. *ACM Transactions on Graphics 21*, 3 (July 2002), 755–762. doi: 10.1145/566654.566648

[KNBH12] KALOGERAKIS E., NOWROUZEZAHRAI D., BRESLAV S., HERTZMANN A.: Learning hatching for pen-and-ink illustration of surfaces. *ACM Transactions on Graphics 31*, 1 (Feb. 2012), 1:1–1:17. doi: 10.1145/2077341.2077342

[Kor02] KORT A.: Computer aided inbetweening. In *Proc. NPAR* (2002), ACM, New York, pp. 125–132. doi: 10.1145/508530.508552

[LBDF13] LU J., BARNES C., DIVERDI S., FINKELSTEIN A.: RealBrush: Painting with examples of physical media. *ACM Transactions on Graphics 32*, 4 (July 2013), 117:1–117:12. doi: 10.1145/2461912.2461998

[LD06] LUFT T., DEUSSEN O.: Real-time watercolor for animation. *Journal of Computer Science and Technology 21*, 2 (Mar. 2006), 159–165. doi: 10.1007/s11390-006-0159-9

[LE05] LU A., EBERT D. S.: Example-based volume illustrations. In *Proc. IEEE Visualization* (2005), IEEE Computer Society, Los Alamitos, pp. 655–662. doi: 10.1109/VIS.2005.31

[LFB*13] LUKÁČ M., FIŠER J., BAZIN J.-C., JAMRIŠKA O., SORKINE-HORNUNG A., SÝKORA D.: Painting by feature: Texture boundaries for example-based image creation. *ACM Transactions on Graphics 32*, 4 (July 2013), 116:1–116:8. doi: 10.1145/2461912.2461956

[LM02] LUM E. B., MA K.-L.: Interactivity is the key to expressive visualization. *ACM SIGGRAPH Computer Graphics 36*, 3 (Aug. 2002), 5–9. doi: 10.1145/570332.570337

[LP15] LAWONN K., PREIM B.: *Feature Lines for Illustrating Medical Surface Models: Mathematical Background and Survey*. Tech. Rep. 1501.03605, arxiv.org, 2015.

[LRA*07] LI W., RITTER L., AGRAWALA M., CURLESS B., SALESIN D.: Interactive cutaway illustrations of complex 3D models. *ACM Transactions on Graphics 26*, 3 (July 2007), 31:1–31:11. doi: 10.1145/1275808.1276416

[LS95] LANSDOWN J., SCHOFIELD S.: Expressive rendering: A review of nonphotorealistic techniques. *IEEE Computers and Graphics 15*, 3 (May 1995), 29–37. doi: 10.1109/38.376610

[LYFD12] LU J., YU F., FINKELSTEIN A., DIVERDI S.: HelpingHand: Example-based stroke stylization. *ACM Transactions on Graphics 31*, 4 (July 2012), 46:1–46:10. doi: 10.1145/2185520.2185542

[MALI11] MARTÍN D., ARROYO G., LUZÓN M. V., ISENBERG T.: Scale-dependent and example-based stippling. *Computers & Graphics 35*, 1 (Feb. 2011), 160–174. doi: 10.1016/j.cag.2010.11.006

[MC01] MASON K., CARPENDALE S.: Artist-driven expressive graphics. In *EG Short Papers* (2001), Eurographics Association, Goslar, Germany, pp. 87–96. doi: 10.2312/egs.20011033

[Mei99] MEIER B. J.: Computers for artists who work alone. *ACM SIGGRAPH Computer Graphics 33*, 1 (Feb. 1999), 50–51. doi: 10.1145/563666.563684

[OG11] OLSEN S., GOOCH B.: Image simplification and vectorization. In *Proc. NPAR* (2011), ACM, New York, pp. 65–74. doi: 10.1145/2024676.2024687

[OS02] ODERMATT K., SPRINGFIELD C.: Creating 3D painterly environments for Disney's "Treasure Planet". In *ACM SIGGRAPH Conference Abstracts and Applications* (2002), ACM, New York, p. 160. doi: 10.1145/1242073.1242173

[PJR*14] PARULEK J., JÖNSSON D., ROPINSKI T., BRUCKNER S., YNNERMAN A., VIOLA I.: Continuous levels-of-detail and visual abstraction for seamless molecular visualization. *Computer Graphics Forum 33*, 6 (May 2014), 276–287. doi: 10.1111/cgf.12349

[PRV13] PARULEK J., ROPINSKI T., VIOLA I.: Seamless abstraction of molecular surfaces. In *Proc. SCCG* (2013), ACM, New York, pp. 107–114. doi: 10.1145/2508244.2508258

[RBG08] RAUTEK P., BRUCKNER S., GRÖLLER M. E.: Interaction-dependent semantics for illustrative volume rendering. *Computer Graphics Forum 27*, 3 (May 2008), 847–854. doi: 10.1111/j.1467-8659.2008.01216.x

[RBGV08] RAUTEK P., BRUCKNER S., GRÖLLER E., VIOLA I.: Illustrative visualization: New technology or useless tautology? *ACM SIGGRAPH Computer Graphics 42*, 3 (Aug. 2008), 4:1–4:8. doi: 10.1145/1408626.1408633

[RC13] ROSIN P., COLLOMOSSE J. (Eds.): *Image and Video based Artistic Stylisation*, vol. 42 of *Computational Imaging and Vision*. Springer, London, Heidelberg, 2013. doi: 10.1007/978-1-4471-4519-6

[RLA*06] RITTER L., LI W., AGRAWALA M., CURLESS B., SALESIN D.: Painting with texture. In *Rendering Techniques (Proc. EGSR* (2006), Eurographics Association, Goslar, Germany, pp. 371–376. doi: 10.2312/EGWR/EGSR06/371-376

[RMN*05] RESNICK M., MYERS B., NAKAKOJI K., SHNEIDERMAN B., PAUSCH R., SELKER T., EISENBERG M.: *Design Principles for Tools to Support Creative Thinking*. Working Paper ISR-816, Institute for Software Research, Carnegie Mellon University, USA, Oct. 2005.

[Sal02] SALESIN D. H.: Non-photorealistic animation & rendering: 7 grand challenges. Keynote talk at NPAR, June 2002.

[SBC06] SHUGRINA M., BETKE M., COLLOMOSSE J.: Empathic painting: Interactive stylization through observed emotional state. In *Proc. NPAR* (2006), ACM, New York, pp. 87–96. doi: 10.1145/1124728.1124744

[SBŽ04] SÝKORA D., BURIÁNEK J., ŽÁRA J.: Unsupervised colorization of black-and-white cartoons. In *Proc. NPAR* (2004), ACM, New York, pp. 121–128. doi: 10.1145/987657.987677

[SDC09a] SÝKORA D., DINGLIANA J., COLLINS S.: As-rigid-as-possible image registration for hand-drawn cartoon animations. In *Proc. NPAR* (2009), ACM, New York, pp. 25–33. doi: 10.1145/1572614.1572619

[SDC09b] SÝKORA D., DINGLIANA J., COLLINS S.: LazyBrush: Flexible painting tool for hand-drawn cartoons. *Computer Graphics Forum 28*, 2 (Apr. 2009), 599–608. doi: 10.1111/j.1467-8659.2009.01400.x

[Sec02] SECORD A.: Weighted Voronoi stippling. In *Proc. NPAR* (2002), ACM, New York, pp. 37–44. doi: 10.1145/508530.508537

[Sei99] SEIMS J.: Putting the artist in the loop. *ACM SIGGRAPH Computer Graphics 33*, 1 (Feb. 1999), 52–53. doi: 10.1145/563666.563685

[SIMC07] SCHWARZ M., ISENBERG T., MASON K., CARPENDALE S.: Modeling with rendering primitives: An interactive non-photorealistic canvas. In *Proc. NPAR* (2007), ACM, New York, pp. 15–22. doi: 10.1145/1274871.1274874

[SLKD15] SEMMO A., LIMBERGER D., KYPRIANIDIS J. E., DÖLLNER J.: Image stylization by oil paint filtering using color palettes. In *Proc. CAe* (2015), Eurographics Association, Goslar, Germany, pp. 149–158. doi: 10.2312/exp.20151188

[SLKD16] SEMMO A., LIMBERGER D., KYPRIANIDIS J. E., DÖLLNER J.: Image stylization by interactive oil paint filtering. *Computers & Graphics 55* (Apr. 2016), 157–171. doi: 10.1016/j.cag.2015.12.001

[SS02] STROTHOTTE T., SCHLECHTWEG S.: *Non-Photorealistic Computer Graphics. Modeling, Animation, and Rendering*. Morgan Kaufmann Publishers, San Francisco, 2002. doi: 10.1016/B978-1-55860-787-3.50019-0

[SSGS11] SCHMID J., SENN M. S., GROSS M., SUMNER R. W.: OverCoat: An implicit canvas for 3D painting. *ACM Transactions on Graphics 30*, 4 (July 2011), 28:1–28:10. doi: 10.1145/2010324.1964923

[ST90] SAITO T., TAKAHASHI T.: Comprehensible rendering of 3-D shapes. *ACM SIGGRAPH Computer Graphics 24*, 3 (Aug. 1990), 197–206. doi: 10.1145/97880.97901

[SWHS97] SALISBURY M. P., WONG M. T., HUGHES J. F., SALESIN D. H.: Orientable textures for image-based pen-and-ink illustration. In *Proc. SIGGRAPH* (1997), ACM, New York, pp. 401–406. doi: 10.1145/258734.258890

[TABI07] TODO H., ANJYO K.-I., BAXTER W., IGARASHI T.: Locally controllable stylized shading. *ACM Transactions on Graphics 26*, 3 (July 2007), 17:1–17:8. doi: 10.1145/1275808.1276399

[VCVL*09] VANDOREN P., CLAESEN L., VAN LAERHOVEN T., TAELMAN J., RAYMAEKERS C., FLERACKERS E., VAN REETH F.: FluidPaint: An interactive digital painting system using real wet brushes. In *Proc. ITS* (2009), ACM, New York, pp. 53–56. doi: 10.1145/1731903.1731914

[vdZLBI11] VAN DER ZWAN M., LUEKS W., BEKKER H., ISENBERG T.: Illustrative molecular visualization with continuous abstraction. *Computer Graphics Forum 30*, 3 (May 2011), 683–690. doi: 10.1111/j.1467-8659.2011.01917.x

[VVLC*08] VANDOREN P., VAN LAERHOVEN T., CLAESEN L., TAELMAN J., RAYMAEKERS C., VAN REETH F.: IntuPaint: Bridging the gap between physical and digital painting. In *Proc. TABLETOP* (2008), IEEE Computer Society, Los Alamitos, pp. 65–72. doi: 10.1109/TABLETOP.2008.4660185

[Win13] WINNEMÖLLER H.: NPR in the wild. In *Image and Video based Artistic Stylisation*, Rosin P., Collomosse J., (Eds.), vol. 42 of *Computational Imaging and Vision*. Springer, London/Heidelberg, 2013, ch. 17, pp. 353–374. doi: 10.1007/978-1-4471-4519-6_17

[WNS*10] WHITED B., NORIS G., SIMMONS M., SUMNER R. W., GROSS M., ROSSIGNAC J.: BetweenIT: An interactive tool for tight inbetweening. *Computer Graphics Forum 29*, 2 (May 2010), 605–614. doi: 10.1111/j.1467-8659.2009.01630.x

[WOG06] WINNEMÖLLER H., OLSEN S. C., GOOCH B.: Real-time video abstraction. *ACM Transactions on Graphics 25*, 3 (July 2006), 1221–1226. doi: 10.1145/1141911.1142018

[WWYS04] WANG B., WANG W., YANG H., SUN J.: Efficient example-based painting and synthesis of 2D directional texture. *IEEE Transactions on Visualization and Computer Graphics 10*, 3 (May/June 2004), 266–277. doi: 10.1109/TVCG.2004.1272726

[WXSC04] WANG J., XU Y., SHUM H.-Y., COHEN M. F.: Video tooning. *ACM Transactions on Graphics 23*, 3 (Aug. 2004), 574–583. doi: 10.1145/1015706.1015763

[XK08] XU J., KAPLAN C. S.: Artistic thresholding. In *Proc. NPAR* (2008), ACM, New York, pp. 39–47. doi: 10.1145/1377980.1377990

[ZZ10] ZHAO M., ZHU S.-C.: Sisley the abstract painter. In *Proc. NPAR* (2010), ACM, New York, pp. 99–107. doi: 10.1145/1809939.1809951