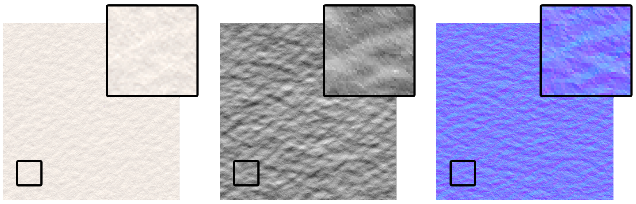


Figure 7: Scanned substrate and extracted profile data.

Once the 3D substrate data is extracted, a heightmap H and normal map N can be easily produced (see Figure 7). We use them to mimic the novel object-space dry-brush effect and to enhance the substrate granulation and distortion. They also enable the possibility to generate interactive substrate lighting.

4.1 Dry-brush and granulation

The dry-brush technique is commonly found in natural painting media. However, it has not yet been explored in object-space, which may be owing to its strong dependency on accurate substrate data. Dry-brush is a painting technique that is applied by watercolor artists with the purpose of indicating a textured appearance. Its uses vary substantially from artist to artist, but it is often employed to depict rough textures such as those produced by leaves, clouds or reflections. The rough appearance is caused when viscous pigment ("dry") only reaches the peaks of the substrate. This allows its valleys to remain unpainted, showing their color (or previously painted pigment). As a consequence, the appearance of the dry-brush application varies strongly depending on the substrate profile, the amount of pigments deposited and the pressure, direction and speed at which the brush-stroke is placed. Our object-space approach won't consider the strokes themselves, but rather the resulting general appearance of a dry-brushed area.

To begin emulating this effect, we first let the artist control the substrate roughness through a global scaling factor r that modulates the depth/height of the original heightmap $H \in [0, 1]$ in Equation 1.

$$H_r = ((H - 0.5) \times r) + 0.5\,. \tag{1}$$

We then proceed to calculate the right pigment application. Since the dry-brush effect is closely related to the substrate granulation, which corresponds to the accumulation of pigment at the valleys of the paper, these two effects are consolidated within a joint pigment application procedure. It is controllable locally through a parameter $a \in [-1, 1]$, which is directly painted by the artist on the 3D geometry. The pigment application P_a is governed by:

$$P_a = \begin{cases} t_{dry}|H_r - a|(C_s - C) + C & \text{if } H_r < a \\ C^{d_a} & \text{if } H_r \geq a \end{cases} \tag{2}$$

If the application parameter a is greater than the elevation of the heightmap H_r (case 1 in Equation 2), a dry brush is applied by linear interpolation between the substrate color C_s and the original color C, with t_{dry} a global dry-brush threshold to smoothen dry edges. Otherwise (case 2 in Equation 2), substrate granulation happens, darkening the color C by the accumulated density d_a. This effect has been thoroughly studied before [Bousseau et al. 2006; Lei and Chang 2005; Luft and Deussen 2006a; Montesdeoca et al. 2017], so we can adapt the substrate data and application parameter to take advantage of this characteristic effect. First, we convert the application parameter into a local granulation density $d_g = |a| + d_{\min}$ with $d_{\min} = 0.2$ the default amount of granulation. Then, to also obtain a darker concentration of pigments for brighter colors, we increase the granulation density (×5) through a linear interpolation of the density contribution with the object color luminance L:

$$d_g = d_g(1 - L) + (d_g \times 5)L\,. \tag{3}$$

Before darkening the pigment at the valleys of the paper to generate granulation, the heightmap amplitude is reduced, shifted and inverted: $H_{iv} = 1 - ((H_r \times 0.2) + 0.8)$. This modulation is performed to increase the density only at the valleys of the paper and adapting our substrate data to follow the approach of Montesdeoca et al. [2017]. The final pigment density accumulation d_a is therefore given by Equation 4, where D is a global density parameter affecting the entire image:

$$d_a = 1 + (d_g \times H_{iv} \times D)\,. \tag{4}$$

The accumulated density d_a is eventually processed by Equation 2 to obtain the granulated pigment application P_a (see Figures 8 and 9).

4.2 Substrate lighting

A painting is always affected by external lighting conditions, thus it is of importance to consider the substrate lighting that would affect the final watercolorized imagery. Unlike oil paintings [Hertzmann 2002], in the case of watercolors, the profile of the paper

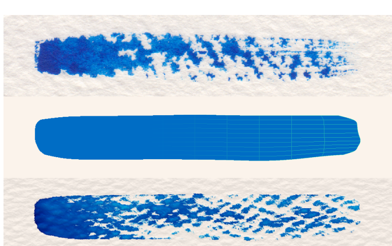


Figure 8: Results of dry-brush and granulation in object-space. From top to bottom: scanned real watercolor, 3D render, watercolorized 3D render.

Figure 9: Stylizing a tree mesh with dry-brush to imitate Figure 2b with different substrates and parameters.

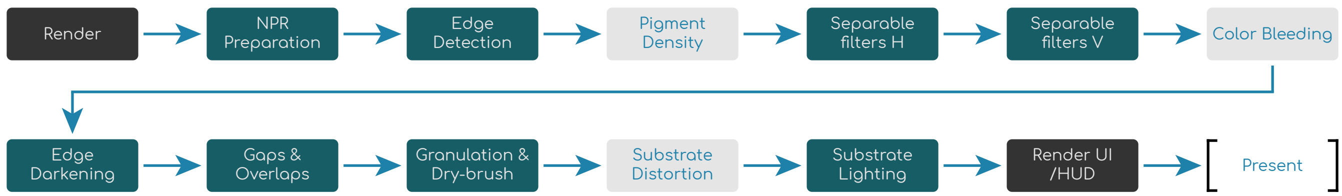


Figure 10: System schematic portraying the rendering pipeline with its different shader operations. Turquoise elements represent the stages at which the necessary data and algorithms are processed.

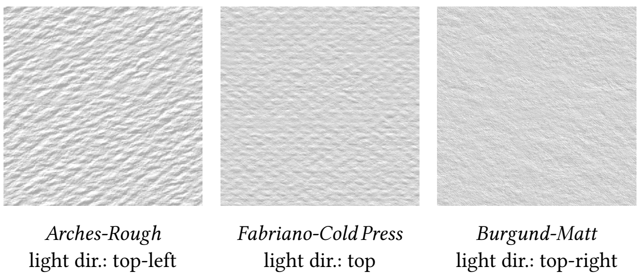


Figure 11: Three watercolor paper specimens lit from different angles. See the accompanying video for their effects on the rendered imagery.

tends to remain mostly unchanged. The carrier evaporates and the remaining pigmentation has little effect over the substrate profile – meaning that a shaded watercolor painting will be mostly affected by the roughness of the substrate. Since the normals of the physical substrate were acquired, deferred shading can be easily conducted on top of the painted image, emulating external lighting conditions on the watercolor painting and increasing the tangibility of the digital substrate (see Figure 11).

Due to unavailable substrate profile data, this effect could previously not be controlled and was fixed to pre-existing shading of the scanned substrate textures. With the extracted substrate profile and normals, we implement a simple diffuse illumination model I_d that can easily be customized through a lighting direction $\mathbf{L}$, the extracted normal map $\mathbf{N}$, the substrate roughness r and the diffuse shading contribution d_s through Equation 5.

$$I_d = 1 - ((1 - (\mathbf{L} \cdot (r \times \mathbf{N}))) \times d_s) . \tag{5}$$

5 ADDITIONAL EFFECTS AND IMPLEMENTATION

In addition to the watercolor effects proposed and presented in this paper, we have also integrated the effects of pigment density, color bleeding, hand-tremors and substrate distortion to augment the watercolor look. These were incorporated from previous approaches by [Bousseau et al. 2006; Montesdeoca et al. 2017]. The entire rendering pipeline is presented in Figure 10 and was implemented using the direct stylization framework developed in Autodesk Maya by Montesdeoca et al. [2017]. The implementation was substantially extended and is briefly described next.

The first render stage rasterizes the 3D geometry which was previously offset by a tremor value in object-space. Multiple render targets come out of this initial stage, including the color image, the z-buffer, and the control masks with the object-space parameters. During the NPR preparation stage, the z-buffer is transformed into linear depth. Then, the edge detection runs the RGBD Sobel filter, followed by the pigment density stage, which darkens or brightens different parts of the image according to their assigned turbulent density. The separable filter stages perform edge blurs for edge darkening, extended edges for gaps & overlaps, and blurs the image for later bleeding. Afterwards, the color bleeding stage blends part of the blurred image to produce localized bleeding. Additionally, the parameter masks related to edge-based effects are also modified, as darkened edges and gaps & overlaps should not appear in these bled areas.

At the second row of Figure 10, the edge- and substrate-based effects are processed. Edge-based effects begin at the edge darkening stage, concentrating the pigmentation at the previously blurred edges. Gaps & overlaps use the previously extended edges to find their neighboring colors and either mix with them or produce gaps. The substrate-based effects begin at the granulation and dry-brush stage, deriving each effect from the acquired heightmap. Then, the substrate distortion modifies the UVs according to the substrate normals, shifting the sampled color. Once all these effects have been computed, the deferred substrate lighting is performed using the substrate normals. Finally, the remaining user-interface/heads-up-display is rendered and the final watercolorized result is presented to the user. Almost all these stages make extensive use of the parameter masks, so that the effects can be controlled in object space, retaining their spatial coherence and localized art-direction.

The real-time implementation does not present significant issues. However, the parallel nature of shaders requires special consideration compared to off-line approaches, as only the currently evaluated pixel can be altered at each stage. The development of our implementation could also have benefited from a modular, node-based approach. This would have enabled us to iterate quicker and alleviate the complexity of managing multiple rendering targets and several rendering stages only through code.

Three different levels of control are included for most effects. At the highest level of control, global parameters help to set initial stylization values over the entire scene. Then, object-space stylization parameters can be assigned within the "material" shaders of each object. Finally, the lowest level of control is given through the painted parameters, stored in different vertex color sets at each vertex of the meshes. We find that painting parameters in object-space offers versatile control and feels natural, but painting each object in a convoluted scene can take a significant amount of time, especially when considering different viewpoints.

(a) Lowpoly House model, 🅭🅯 Rafael Scopel

(b) Spherebot model, 🅭🅯 Bastien Genbrugge

(c) Octopus model, 🅭🅯 Lukáš Marek

(d) Fruit plate watercolorized with (top) Montesdeoca et al. [2016] method and (bottom) our pipeline with the integrated effects.

Figure 12: Rendered frames featuring edge- and substrate-based effects. Please refer to the accompanying video to see them in motion.

6 RESULTS AND DISCUSSION

A watercolor look is composed of a series of low-level effects, that together create a defined watercolor style. Although this paper specifically focuses on edge- and substrate-based effects, we incorporated other previously studied watercolor effects into our pipeline. In this way, we can feature the contribution of the developed effects as close to its natural occurrence as possible. Some examples of watercolorized imageries featuring the effects addressed in this paper can be found in Figures 1, 9, 6 and 12.

In Figure 12a, dry-brush was used for the tree leaves, the rooftop and the lamp post to abstract and to provide texture. Some parts of the walls in the house also show a minor dry-brush application to increase the roughness of the watercolor look. Additionally, gaps & overlaps were used extensively to provide the rough and sketchy look. In Figures 12b and 12c, gaps were used to delineate the form of the characters and provide a sketchier look. Figure 12c also presents some minor dry-brush application where suitable. Finally, Figure 12d provides a comparison with the previous approach by Montesdeoca et al. [2016], demonstrating how gaps & overlaps and dry-brush contribute to the palette of watercolor effects to generate elaborate watercolor renders. To see these examples under motion and animation, please refer to the accompanying video.

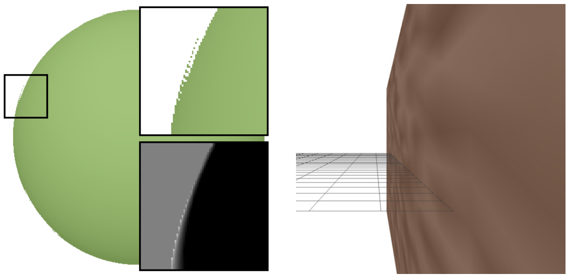
The results show an overall pleasant integration of dry-brush and gaps & overlaps to form a sketchier and rougher look, which resembles their usage in traditional watercolors. Other proposed effects such as substrate lighting enable a more realistic and tangible rendered outcome, whereas the RGBD edge detection helps create more consistent and controllable edge darkening, compared to previous approaches.

Though the GPU pipeline (see Figure 10) is quite complex, the render manages to perform in real-time (60+ frames per second) with ease at full HD (1920×1080) resolution, with the following configuration: NVIDIA GeForce GTX 1080, Intel Xeon E5-2609 @2.4Ghz and 16Gb of RAM. As most of the processing is also performed in image-space, the stylization scales well with scene complexity. This is highly beneficial for interactivity and art-direction, especially since most effects are localized.

6.1 Limitations

The data from the substrate specimens, acquired with a shape-from-shading technique, was accurate enough for our experimental and aesthetic purposes. Yet a proper physical profile scan could present an increased sharpness and fidelity, which could further improve the modeled effects and the overall watercolor look.

The dry-brush effect is relative to the substrate size, therefore, it will look significantly different under diverse paper profiles and scales, or when the camera moves closer or further away from the subject (if the substrate scale remains static). This behavior is physically correct, but it takes some time to get used to and presents temporal coherence issues in animation. A dynamic paper texture [Bénard et al. 2009; Cunzi et al. 2003; Kaplan and Cohen 2005] might be able to reduce temporal coherence problems caused by the substrate, with a potential penalty in fidelity with the original data. Additionally, the dry-brush technique changes depending on the direction of the brush stroke, and may sometimes leave a trace



(a) Geometric distortion **(b)** Perspective distortion

Figure 13: Examples of geometric (a) and perspective (b) distortions affecting the parameter masks.

of the brush bristles (see Figure 8), this was not considered in our object-space approach.

The RGBD edge detection is still prone to noise, especially when using photorealistic textures. A general pre-processing stage for textures [Semmo and Döllner 2015] would create more robust and meaningful edges, especially for effects like gaps & overlaps. Additionally the edge-detection would probably benefit from a deferred rendering pipeline, as the edges would not be influenced by shading conditions (i.e., sharp edges from specular highlights).

A more general limitation of our method comes from the parameter masks painted on the object surface. While this approach helps the spatial coherence of the render and grants higher control over the effects, the 3D surfaces are subject to geometric and perspective distortions, which may be undesirable. In Figure 13a, the spherical mapping of the parameter mask distorts the transition of parameters, increasing its abruptness and potentially creating pixel artifacts. This behavior is especially noticeable when the normals of the geometric surface are almost perpendicular to the view direction (see the bright pixel parameters at the edge of the sphere). In Figure 13b, a mask with an otherwise uniform noise pattern is foreshortened through perspective distortion, changing the scale of the parametrization relative to its distance from the image plane. While these object-space distortions might not be desirable, they are not too noticeable as long as the parameter masks are well designed.

7 CONCLUSION AND FUTURE WORK

The main contributions presented in this paper include the edge-based effects of gaps & overlaps and the substrate-based effect of dry-brush, which had little study in object-space stylization before. While working to mimic these effects, we extracted edges and acquired substrate data that helped to improve previously studied algorithms such as edge-darkening, substrate granulation and enabled the possibility of shading the finalized watercolorized imagery with external light sources.

The dry-brush effect and gaps & overlaps aid significantly to generate overall sketchier and rougher watercolor imagery, which was not possible before. There is an intrinsic difficulty to achieve naturally occurring imperfections within computer graphics and a volatile natural medium like watercolor is no exception to this. Curiously, these imperfections are believably created by resorting to more robust algorithms and the use of more physically accurate substrates. A physical substrate also enabled the possibility to generate fully controllable external lighting conditions, which have not been given much attention before, but offer a new level of tangibility to the digital creation. These effects, together with several smaller improvements to previously studied effects, enlarge the possibilities and applications that watercolor in object-space offers – bringing the medium forward and offering a stylistic change to otherwise standard rendering procedures and looks.

Future work in object-space watercolor research could focus on layered pigmentation and turbulent pigment mixing. These are unexplored areas that could address aesthetic complexity, commonly found in watercolors. Other features that could bring the stylization forward involve dynamic image-space substrates that communicate with object-space data to remain coherent, and texture painting within a direct stylization pipeline, to enhance the artist-computer interaction. Regarding substrate- and edge-based effects in general, there are other types of expressive rendering which could take advantage of a closer exploration and analysis of these two factors. All natural media, thinly applied, is affected by the substrate it is painted on, and edges are often subject of stylization by artists. While painting parameters in object-space grants extensive control and art-direction, locally painting a complex scene at each vertex can become a cumbersome endeavor. A more volumetric approach towards parameter assignment might present itself as an interesting subject to study from an interactive and aesthetic point of view. Finally, after implementing our contribution in a direct stylization pipeline, we believe that such systems could be generalized to other types of styles in which cross-media parametrization would create an unprecedented versatility and application of expressive rendering techniques.

ACKNOWLEDGMENT

We would like to thank the Multi-plAtform Game and Innovation Centre (MAGIC) and the Maverick team at Inria - Rhône-Alpes for supporting this international collaboration. Our gratitude is also with our colleagues for their comments and suggestions, together with the anonymous reviewers for their constructive feedback. This research is supported by the Interdisciplinary Graduate School (IGS) at the Nanyang Technological University, Singapore and the National Research Foundation, Prime Minister's Office, Singapore under its IDM Futures Funding Initiative.

REFERENCES

Angelos Barmpoutis, Eleni Bozia, and Robert S. Wagman. 2010. A novel framework for 3D reconstruction and analysis of ancient inscriptions. *Machine Vision and Applications* 21, 6 (2010), 989–998. DOI:https://doi.org/10.1007/s00138-009-0198-7

Adrien Bousseau, Matt Kaplan, Joëlle Thollot, and François X. Sillion. 2006. Interactive Watercolor Rendering with Temporal Coherence and Abstraction. In *Proceedings of International Symposium on Non-photorealistic Animation and Rendering (NPAR '06)*. ACM, 141–149. DOI:https://doi.org/10.1145/1124728.1124751

Adrien Bousseau, Fabrice Neyret, Joélle Thollot, and David Salesin. 2007. Video Watercolorization Using Bidirectional Texture Advection. *ACM Transactions on Graphics* 26, 3 (2007), 104. DOI:https://doi.org/10.1145/1276377.1276507

Claire Waite Brown. 2007. *The Watercolor Flower Artist's Bible: An Essential Reference for the Practicing Artist*. Chartwell Books.

Jeremy Burgess, Geoff Wyvill, and Scott A. King. 2005. A system for real-time watercolour rendering. In *Computer Graphics International*. IEEE, 234–240. DOI:

https://doi.org/10.1109/CGI.2005.1500426

Pierre Bénard, Adrien Bousseau, and Joëlle Thollot. 2009. Dynamic Solid Textures for Real-Time Coherent Stylization. In *Proceedings of the Symposium on Interactive 3D graphics and games*. ACM, 121–127. DOI: https://doi.org/10.1145/1507149.1507169

Pierre Bénard, Forrester Cole, Michael Kass, Igor Mordatch, James Hegarty, Martin Sebastian Senn, Kurt Fleischer, Davide Pesare, and Katherine Breeden. 2013. Stylizing Animation By Example. *ACM Transactions on Graphics* 32, 4 (2013), 119. DOI: https://doi.org/10.1145/2461912.2461929

Dongdong Chen, Jing Liao, Lu Yuan, Nenghai Yu, and Gang Hua. 2017. Coherent Online Video Style Transfer. *ArXiv e-prints* (2017). http://https://arxiv.org/abs/1703.09211 [accessed: 2017-04-04].

Zhili Chen, Byungmoon Kim, Daichi Ito, and Huamin Wang. 2015. Wetbrush: GPU-based 3D Painting Simulation at the Bristle Level. *ACM Transactions on Graphics* 34, 6 (2015), 200:1–200:11. DOI: https://doi.org/10.1145/2816795.2818066

Nelson S. H. Chu and Chiew-Lan Tai. 2005. MoXi: Real-time Ink Dispersion in Absorbent Paper. *ACM Transactions on Graphics* 24, 3 (2005), 504–511. DOI: https://doi.org/10.1145/1073204.1073221

John P Collomosse, David Rowntree, and Peter M Hall. 2005. Stroke surfaces: temporally coherent artistic animations from video. *IEEE Transactions on Visualization and Computer Graphics* 11, 5 (2005), 540–9. DOI: https://doi.org/10.1109/TVCG.2005.85

Matthieu Cunzi, Joëlle Thollot, Sylvain Paris, Gilles Debunne, Jean-Dominique Gascuel, and Frédo Durand. 2003. Dynamic Canvas for Immersive Non-Photorealistic Walkthroughs. In *Proceedings of Graphics Interface*. A K Peters, LTD., 121–130.

Cassidy J. Curtis, Sean E. Anderson, Joshua E. Seims, Kurt W. Fleischer, and David H. Salesin. 1997. Computer-generated Watercolor. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH 1997)*. ACM, 421–430. DOI: https://doi.org/10.1145/258734.258896

Stephen DiVerdi, Aravind Krishnaswamy, Radomír Měch, and Daichi Ito. 2013. Painting with Polygons: A Procedural Watercolor Engine. *IEEE Transactions on Visualization and Computer Graphics* 19, 5 (2013), 723–735. DOI: https://doi.org/10.1109/TVCG.2012.295

Jakub Fišer, Ondřej Jamriška, Michal Lukáč, Eli Shechtman, Paul Asente, Jingwan Lu, and Daniel Sýkora. 2016. StyLit: Illumination-guided Example-based Stylization of 3D Renderings. *ACM Transactions on Graphics* 35, 4 (2016), 92:1–92:11. DOI: https://doi.org/10.1145/2897824.2925948

Gene Franks. 1988. *Watercolor: Drybrush Technique*. Walter Foster Pub.

Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. 2015. A Neural Algorithm of Artistic Style. *CoRR* abs/1508.06576 (2015). http://arxiv.org/abs/1508.06576 [accessed: 2016-10-28].

Nathan Gossett and Baoquan Chen. 2004. Paint Inspired Color Mixing and Compositing for Visualization. In *IEEE Symposium on Information Visualization*. 113–118. DOI: https://doi.org/10.1109/INFVIS.2004.52

Aaron Hertzmann. 2002. Fast paint texture. In *Proceedings of International Symposium on Non-photorealistic Animation and Rendering (NPAR '02)*. ACM Press, 91. DOI: https://doi.org/10.1145/508530.508546

Aaron Hertzmann, Charles E. Jacobs, Nuria Oliver, Brian Curless, and David H. Salesin. 2001. Image Analogies. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '01)*. ACM, 327–340. DOI: https://doi.org/10.1145/383259.383295

Klaus Janson. 2003. *The DC Comics Guide to Inking Comics*. Watson-Guptill.

Henry Johan, Ryota Hashimoto, and Tomoyuki Nishita. 2004. Creating Watercolor Style Images Taking Into Account Painting Techniques. *The Journal of the Society for Art and Science* 3, 4 (2004), 207–215. DOI: https://doi.org/10.3756/artsci.3.207

Mizuki Kagaya, William Brendel, Qingqing Deng, Todd Kesterson, Sinisa Todorovic, Patrick J. Neill, and Eugene Zhang. 2011. Video Painting with Space-Time-Varying Style Parameters. *IEEE Transactions on Visualization and Computer Graphics* 17, 1 (2011), 74–87. DOI: https://doi.org/10.1109/TVCG.2010.25

Matthew Kaplan and Elaine Cohen. 2005. A Generative Model for Dynamic Canvas Motion. In *Proceedings of the Eurographics Conference on Computational Aesthetics in Graphics (CAe '05)*. Eurographics Association, 49–56. DOI: https://doi.org/10.2312/compaesth/compaesth05/049-056

Su Ian Eugene Lei and Chun-Fa Chang. 2005. *Real-Time Rendering of Watercolor Effects for Virtual Environments*. Springer Berlin Heidelberg, 474–481. DOI: https://doi.org/10.1007/978-3-540-30543-9_60

Jingwan Lu, Connelly Barnes, Stephen DiVerdi, and Adam Finkelstein. 2013. RealBrush: Painting with Examples of Physical Media. *ACM Transactions on Graphics* 32, 4 (2013), 1–12. DOI: https://doi.org/10.1145/2461912.2461998

Thomas Luft and Oliver Deussen. 2006a. Real-Time Watercolor for Animation. *Journal of Computer Science and Technology* 21, 2 (2006), 159–165. DOI: https://doi.org/10.1007/s11390-006-0159-9

Thomas Luft and Oliver Deussen. 2006b. Real-time Watercolor Illustrations of Plants Using a Blurred Depth Test. In *Proceedings of International Symposium on Non-photorealistic Animation and Rendering (NPAR '06)*. ACM, 11–20. DOI: https://doi.org/10.1145/1124728.1124732

Thomas Luft, Frank Kobs, Walter Zinser, and Oliver Deussen. 2008. Watercolor Illustrations of CAD Data. In *Proceedings of the Eurographics Conference on Computational Aesthetics in Graphics (CAe '08)*. The Eurographics Association. DOI: https://doi.org/10.2312/COMPAESTH/COMPAESTH08/057-063

Eric B. Lum and Kwan-Liu Ma. 2001. Non-photorealistic rendering using watercolor inspired textures and illumination. In *Proceedings of the 9th Pacific Conference on Computer Graphics and Applications*. 322–330. DOI: https://doi.org/10.1109/PCCGA.2001.962888

Chaohui Lü and Xiaolong Chen. 2014. Image watercolorization based on visual weight-map. In *International Congress on Image and Signal Processing (CISP) (CISP)*. 233–237. DOI: https://doi.org/10.1109/CISP.2014.7003783

Santiago E Montesdeoca, Hock Soon Seah, and Hans-Martin Rall. 2016. Art-directed Watercolor Rendered Animation. In *Proceedings of International Symposium on Non-photorealistic Animation and Rendering (NPAR '16)*. The Eurographics Association, 51–58. DOI: https://doi.org/10.2312/exp.20161063

Santiago E. Montesdeoca, Hock Soon Seah, Hans-Martin Rall, and Davide Benvenuti. 2017. Art-directed watercolor stylization of 3D animations in real-time. *Computers & Graphics* 65 (2017), 60 – 72. DOI: https://doi.org/10.1016/j.cag.2017.03.002

Bethan Morris. 2010. *Fashion Illustrator*. Laurence King Publishing.

Marc Nienhaus and Jürgen Döllner. 2005. Blueprint Rendering and Sketchy Drawings. In *GPU Gems II: Programming Techniques for High Performance Graphics and General-Purpose Computation*. Addison-Wesley Professional, Chapter 15, 235–252.

Giuseppe Papari and Nicolai Petkov. 2011. Edge and line oriented contour detection: State of the art. *Image and Vision Computing* 29, 2–3 (2011), 79 – 103. DOI: https://doi.org/10.1016/j.imavis.2010.08.009

Takafumi Saito and Tokiichiro Takahashi. 1990. Comprehensible rendering of 3-D shapes. *Proceedings of the 17th annual conference on Computer graphics and interactive techniques - SIGGRAPH '90* 24, 4 (1990), 197–206. DOI: https://doi.org/10.1145/97879.97901

Ahmed Selim, Mohamed Elgharib, and Linda Doyle. 2016. Painting Style Transfer for Head Portraits Using Convolutional Neural Networks. *ACM Transactions on Graphics* 35, 4 (2016), 129:1–129:18. DOI: https://doi.org/10.1145/2897824.2925968

Amir Semmo and Jürgen Döllner. 2015. Interactive image filtering for level-of-abstraction texturing of virtual 3D scenes. *Computers & Graphics* 52 (2015), 181 – 198. DOI: https://doi.org/10.1016/j.cag.2015.02.001

Amir Semmo, Tobias Dürschmid, Matthias Trapp, Mandy Klingbeil, Jürgen Döllner, and Sebastian Pasewaldt. 2016. Interactive Image Filtering with Multiple Levels-of-Control on Mobile Devices. In *Proceedings of the ACM SIGGRAPH Asia Symposium on Mobile Graphics and Interactive Applications*. ACM, NA. DOI: https://doi.org/10.1145/2999508.2999521

David Small. 1991. Simulating watercolor by modeling diffusion, pigment, and paper fibers. *SPIE Proceedings* 1460, Image Handling and Reproduction Systems Integration (1991), 140–146. DOI: https://doi.org/10.1117/12.44417

Hazel Soan. 2014. *The Artist's Color Guide - Watercolor: Understanding Palette, Pigments and Properties*. F&W Media, Incorporated. https://books.google.com.sg/books?id=GsAUngEACAAJ

Tom Van Laerhoven and Frank Van Reeth. 2005. Real-time simulation of watery paint. *Computer Animation and Virtual Worlds* 16, 3-4 (2005), 429–439. DOI: https://doi.org/10.1002/cav.95

Miaoyi Wang, Bin Wang, Yun Fei, Kanglai Qian, Wenping Wang, Jiating Chen, and Jun-Hai Yong. 2014. Towards Photo Watercolorization with Artistic Verisimilitude. *TVCG* 20, 10 (2014), 1451–1460. DOI: https://doi.org/10.1109/TVCG.2014.2303984

Mi You, Taekwon Jang, Seunghoon Cha, Jihwan Kim, and Junyong Noh. 2013. Realistic paint simulation based on fluidity, diffusion, and absorption. *Computer Animation and Virtual Worlds* 24, 3-4 (2013), 297–306. DOI: https://doi.org/10.1002/cav.1500