A Survey of Digital Stippling—Additional Material

Domingo Martín[1] Germán Arroyo[1] Alejandro Rodríguez[1] Tobias Isenberg[2]

[1] *Department of Software Engeneering, University of Granada, Spain* [2] *Inria and Université Paris-Saclay, France*

Abstract

In this article we survey techniques for the digital simulation of hand-made stippling—one of the core techniques developed within non-photorealistic/expressive rendering. Over the years, a plethora of automatic or semi-automatic stippling algorithms have been proposed. As part of this expanding field of research, techniques have been developed that not only push the boundaries of traditional stippling but that also relate to other processes or techniques. Our general goal in this survey is thus to increase our understanding of both hand-made and computer-assisted stippling. For this purpose we not only provide an overview of the work on digital stippling but also examine its relationship to traditional stippling and to related fields such as halftoning. Finally, we propose several directions of future work in the field.

Keywords: Stippling, NPR stippling, hand-made stippling, halftoning

In this additional material we provide, for a number of example images, stipple results using several of the techniques we discussed in the survey to allow the readers to make a better comparison. For this purpose we specifically did not select images from the selection suggested by Mould and Rosin [1] due to the unclear copyright status of the images in this set. Instead, we selected a number of benchmark images used for stippling in the past (for which we could secure the rights) and some images inspired by Mould and Rosin's set but which have no copyright restrictions.

We used some of the source images from other authors without additional processing (Figure 2(a)–(c)). For the other images (Figure 1(d)–(f)) we derived a grayscale version using Gimp. In some cases adjusted the contrast of the images as suggested by Secord [2], while in others we follow the steps recommended by Martín et al. [3, 4]. Figure 2 shows the final source images in their 768×768 resolution versions that we used for all approaches, unless specified otherwise.

We generally show the set of images all on the same page, with each image always shown with the same size. For those approaches that depend on the output resolution we based the computation on the specific size of the images on these pages. For all others we used our best judgment to set meaningful parameters. In some cases we show additional images for specific approaches where different sizes or parameter settings lead to better results. By default, all images are shown at size 65mm $\times$ 65mm.

References

[1] Mould, D., Rosin, P.L.. A benchmark image set for evaluating stylization. In: Proc. NPAR (Expressive). Goslar, Germany: Eurographics Assoc.; 2016, p. 11–20. doi: 10.2312/exp.20161059

[2] Secord, A.. Weighted Voronoi stippling. In: Proc. NPAR. New York: ACM; 2002, p. 37–43. doi: 10.1145/508530.508537

[3] Martín, D., Arroyo, G., Luzón, M.V., Isenberg, T.. Example-based stippling using a scale-dependent grayscale process. In: Proc. NPAR. New York: ACM; 2010, p. 51–61. doi: 10.1145/1809939.1809946

[4] Martín, D., Arroyo, G., Luzón, M.V., Isenberg, T.. Scale-dependent and example-based stippling. Computers & Graphics 2011;35(1):160–174. doi: 10.1145/1809939.1809946

[5] Kim, S., Maciejewski, R., Isenberg, T., Andrews, W.M., Chen, W., Sousa, M.C., et al. Stippling by example. In: Proc. NPAR. New York: ACM; 2009, p. 41–50. doi: 10.1145/1572614.1572622

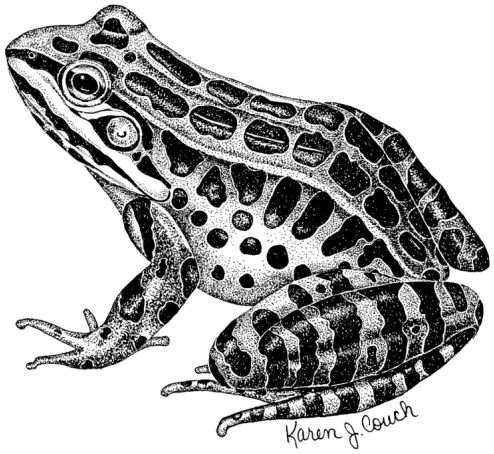
[6] Schlechtweg, S., Germer, T., Strothotte, T.. Renderbots—Multi agent systems for direct image generation. Computer Graphics Forum 2005;24(2):137–148. doi: 10.1111/j.1467-8659.2005.00838.x

[7] Balzer, M., Schlömer, T., Deussen, O.. Capacity-constrained point distributions: A variant of Lloyd's method. ACM Transactions on Graphics 2009;28(3):86:1–86:8. doi: 10.1145/1531326.1531392

[8] Kopf, J., Cohen-Or, D., Deussen, O., Lischinski, D.. Recursive Wang tiles for real-time blue noise. ACM Transactions on Graphics 2006;25(3):509–518. doi: 10.1145/1141911.1141916

[9] Li, H., Mould, D.. Structure-preserving stippling by priority-based error diffusion. In: Proc. Graphics Interface. Waterloo, ON, Canada: CHCCS; 2011, p. 127–134. doi: 10.20380/GI2011.17

(a) Plant, originally used by Secord [2]. The image © 2007 Adrian Secord, used with permission. URL: `https://goo.gl/9iVtBF`



(b) Frog. The illustration was initially created by Karen J. Couch and the image is in the public domain. URL: `https://goo.gl/ZOHxfS`

(c) Landscape. The image © 2010 Domingo Martín Perandres, used with permission. URL: `https://goo.gl/yc8Xz9`

(d) Cat, inspired by an image suggested by Mould and Rosin [1]. The image is by Ilona Krijgsman and is in the public domain. URL: `https://goo.gl/b5m68e`

(e) Lena, from the USC-SIPI Image Database. Used under the fair-use clause. URL: `https://goo.gl/4ulo79`

(f) Portrait from Wikimedia Commons by Wikimedia user "The Photographer". The image is in the public domain. URL: `https://goo.gl/vQRaro`

Figure 1: The originals from which we derived our source images (Figure 2).

(a) Plant, originally used by Secord [2]. The image © 2007 Adrian Secord, used with permission. The image is identical to Figure 1(a).

(b) Frog, originally used by Kim et al. [5]. The image © 2009 SungYe Kim (based on the image in Figure 1(b)), used with permission.

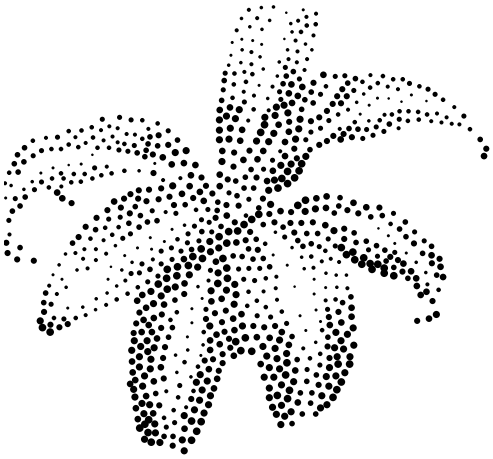
(c) Landscape, originally used by Martín et al. [4] (based on the image in Figure 1(c)). The image © 2010 Domingo Martín Perandres, used with permission.

(d) Cat, based on the image in Figure 1(d).

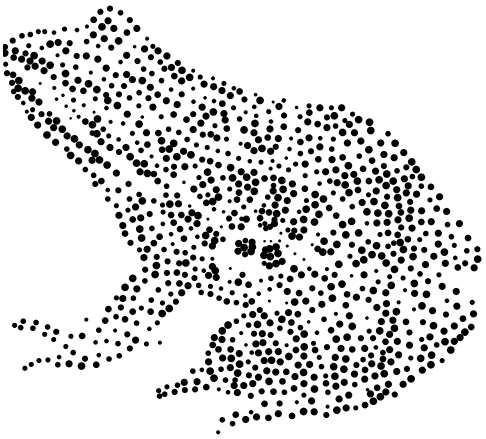
(e) Lena based on the image in Figure 1(e).

(f) Portrait based on the image in Figure 1(f).

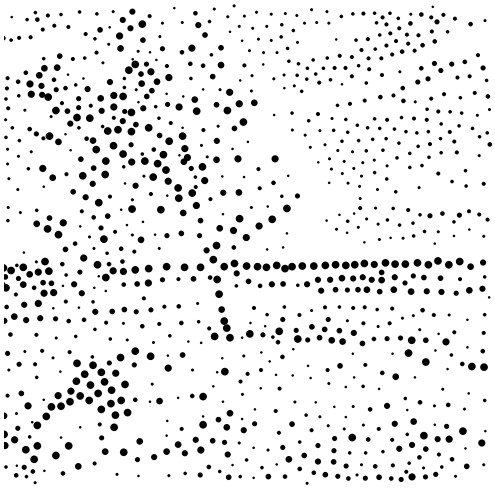
Figure 2: The source images we used for the comparison, all have a resolution of 768×768.



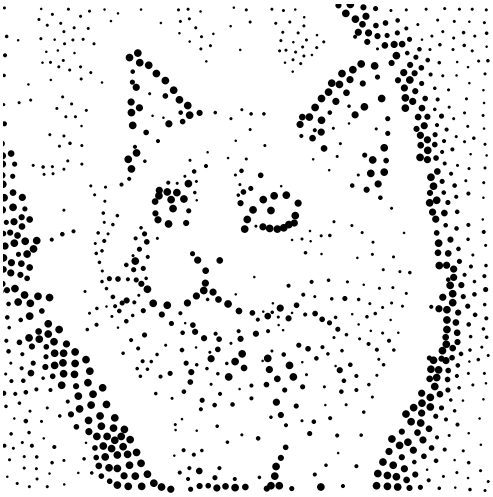
(a) 899 stipple dots.



(b) 1,000 stipple dots.

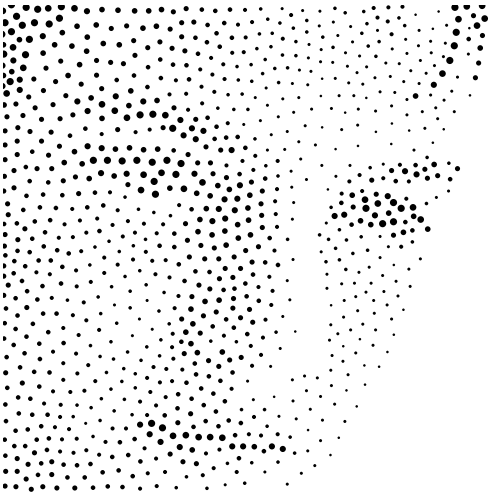


(c) 1,000 stipple dots.



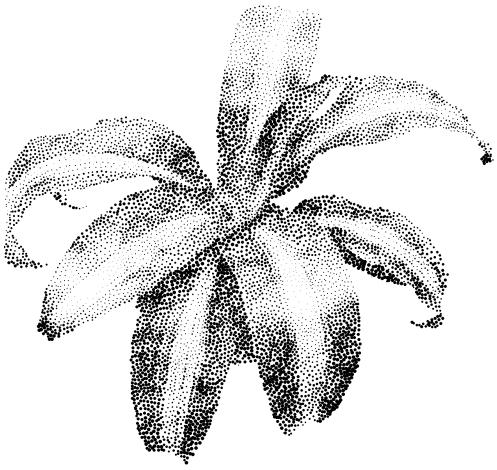
(d) 1,000 stipple dots.

(e) 874 stipple dots.

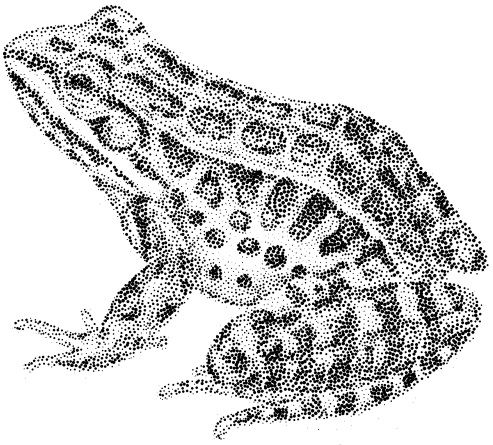


(f) 1,000 stipple dots.

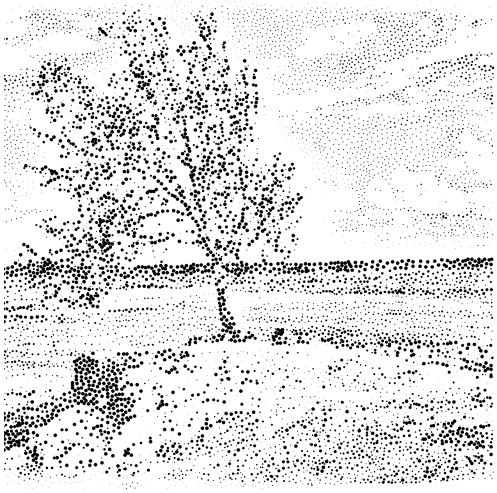
Figure 3: Stippling using Secord's [2] weighted centroidal Voronoi diagrams using a low target stipple count of 1,000 dots. The exact count differs from this target number due to the point seeding. The images were generated with StippleShop.



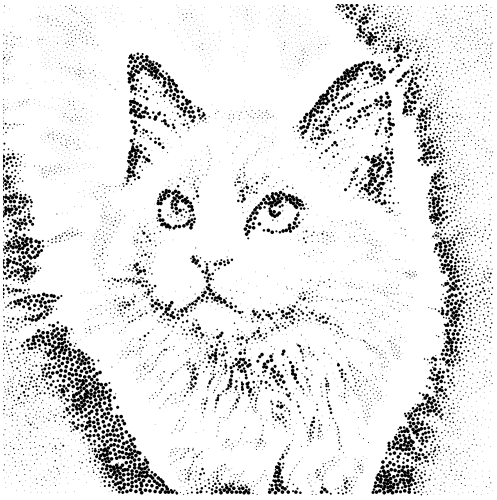
(a) 7,192 stipple dots.



(b) 7,804 stipple dots.

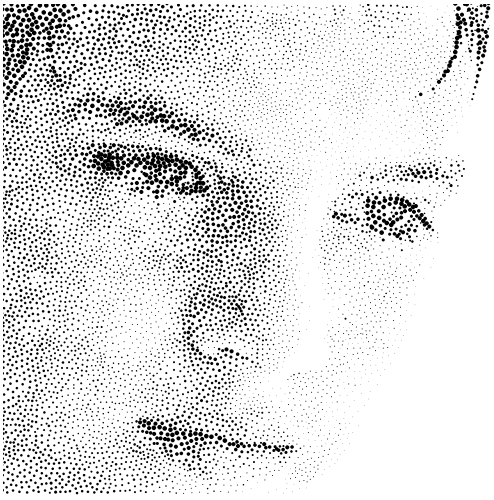


(c) 7,979 stipple dots.



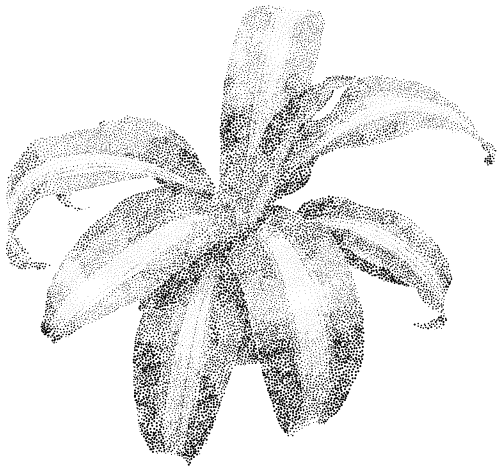
(d) 7,749 stipple dots.

(e) 6,994 stipple dots.

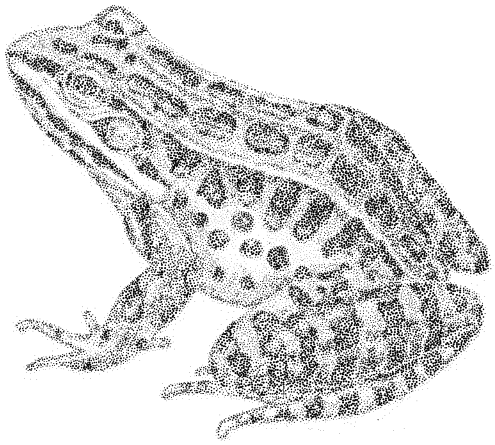


(f) 7,012 stipple dots.

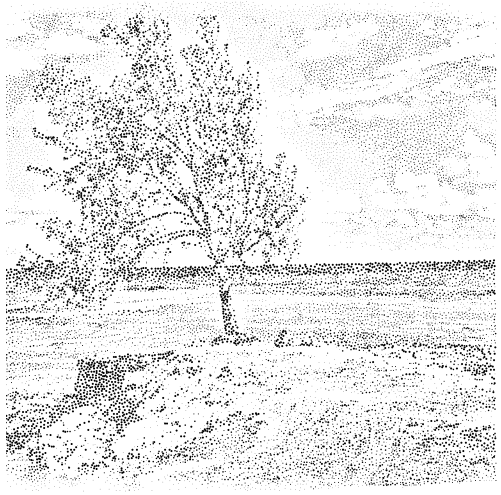
Figure 4: Stippling using Secord's [2] weighted centroidal Voronoi diagrams using a medium target stipple count of 7,500 dots. The exact count differs from this target number due to the point seeding. The images were generated with StippleShop.



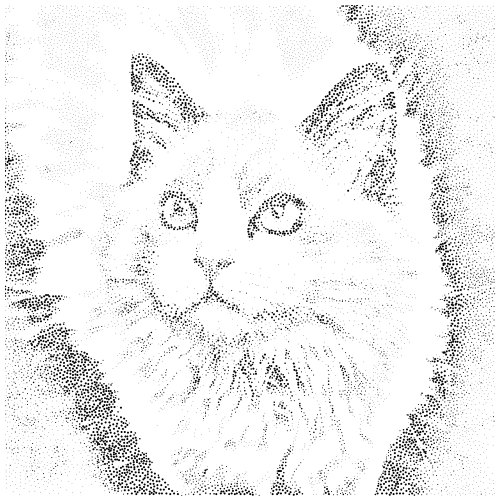
(a) 14,384 stipple dots.



(b) 18,953 stipple dots.

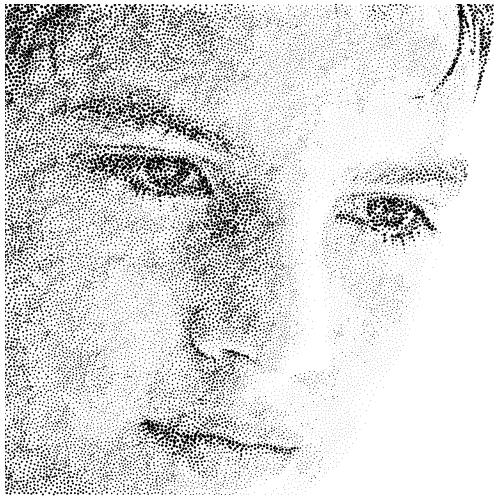


(c) 17,554 stipple dots.



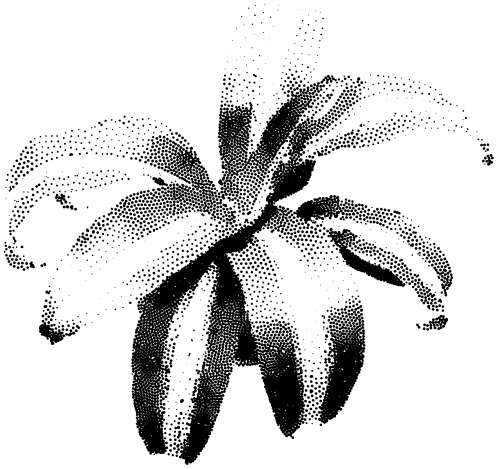
(d) 12,915 stipple dots.

(e) 12,241 stipple dots.

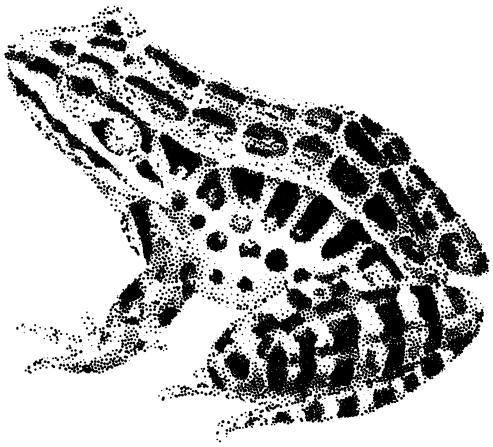


(f) 17,530 stipple dots.

Figure 5: Stippling using Secord's [2] weighted centroidal Voronoi diagrams using the same target stipple count for each of the images that was used by Martín et al.'s resolution-dependent grayscale approach [3, 4] (Figure 12). The exact count differs from this target number due to the point seeding. The images were generated with StippleShop.



(a) Approx. 7,150 stipple dots.



(b) Approx. 1,100 stipple dots.

(c) Approx. 8,700 stipple dots.

(d) Approx. 6,800 stipple dots.

(e) Approx. 7,100 stipple dots.

(f) Approx. 3,250 stipple dots.

Figure 6: Stippling using Schlechtweg et al.'s [6] RenderBots. The stipple count is controlled by the algorithm so we can only provide approximate stipple counts because the number constantly changes slightly due to the inherent animation. The images were generated with Schlechtweg et al.'s original tool at a high resolution and then converted to binary form by thresholding.

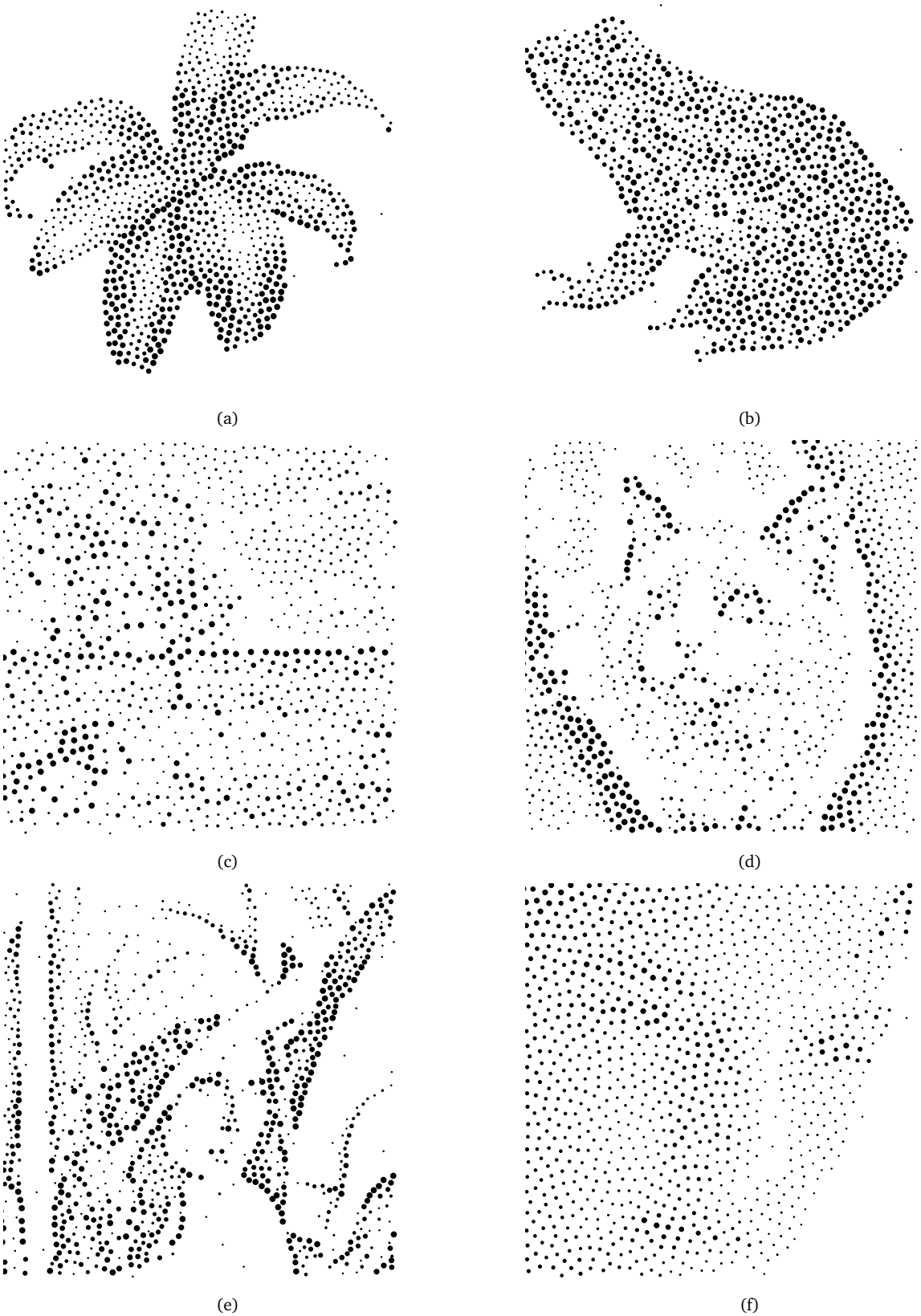
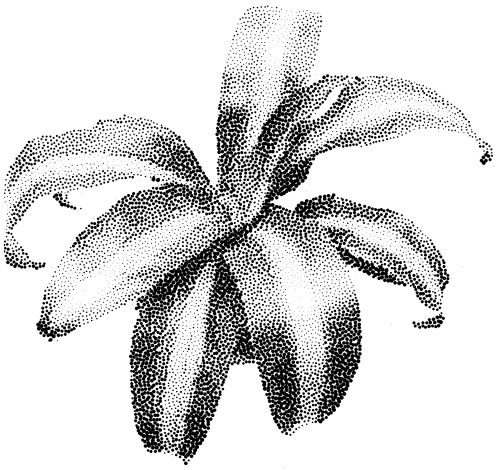
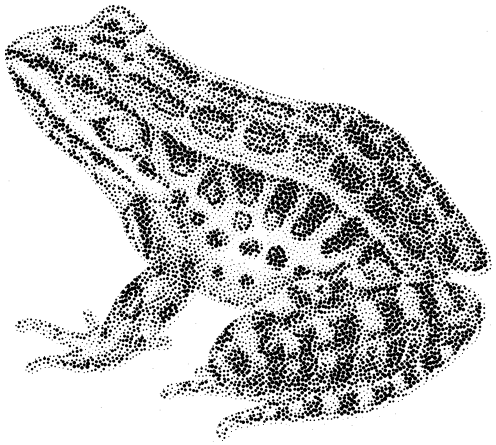


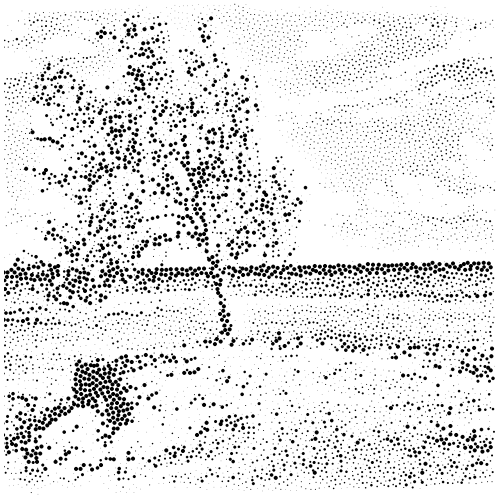
Figure 7: Stippling using Balzer et al.'s [7] capacity-constrained approach using a low stipple count of 1,000 dots. The images were generated with StippleShop.



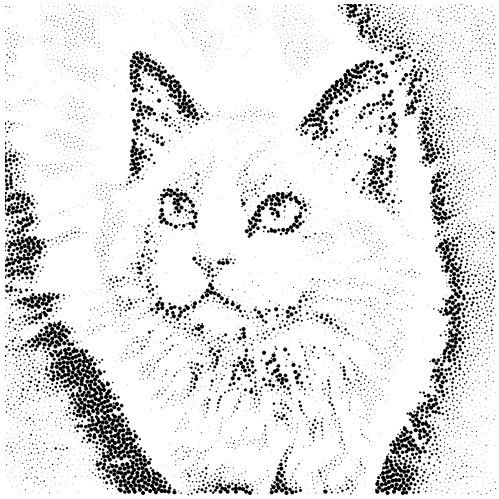
(a)



(b)

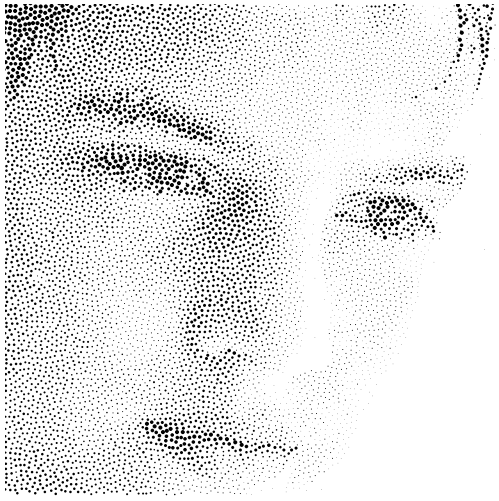


(c)



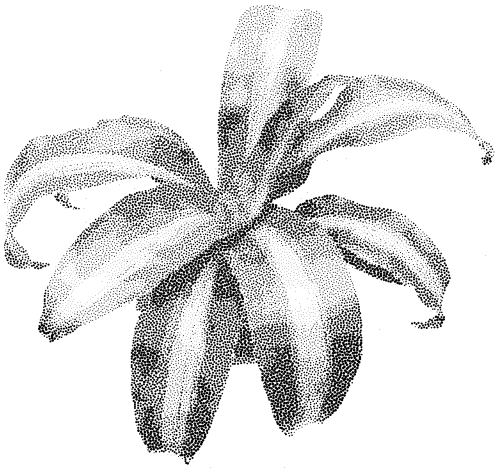
(d)

(e)

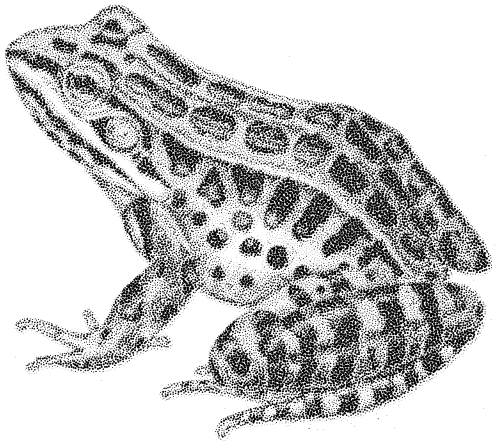


(f)

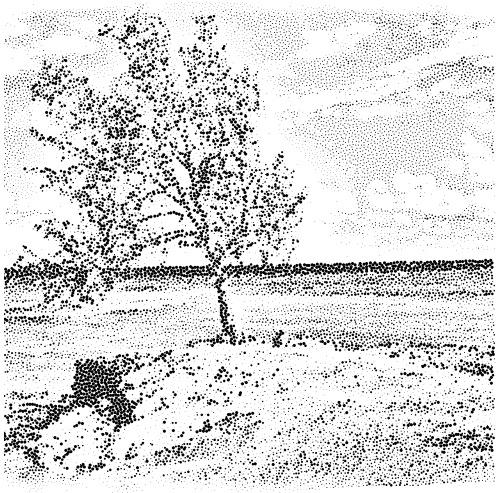
Figure 8: Stippling using Balzer et al.'s [7] capacity-constrained approach using a medium stipple count of 7,500 dots. The images were generated with StippleShop.



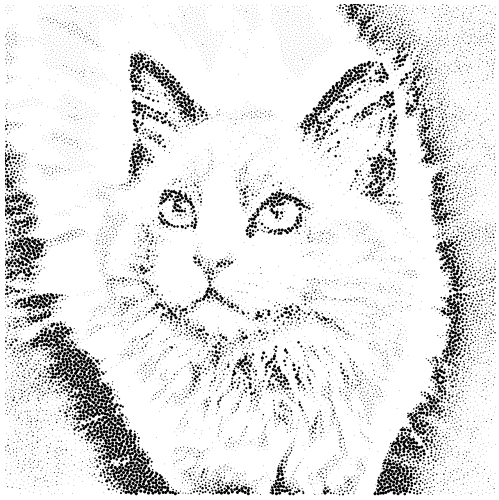
(a) 14,269 stipple dots.



(b) 19,405 stipple dots.

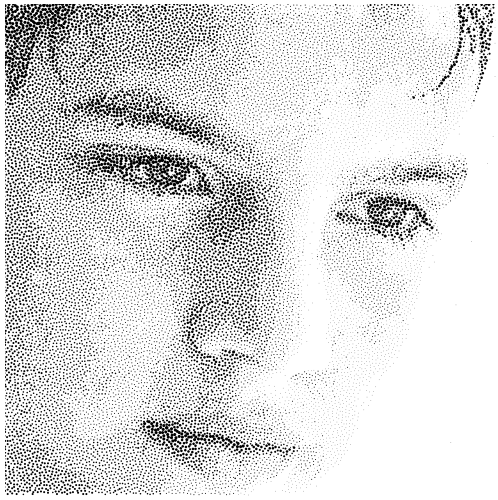


(c) 17,044 stipple dots.



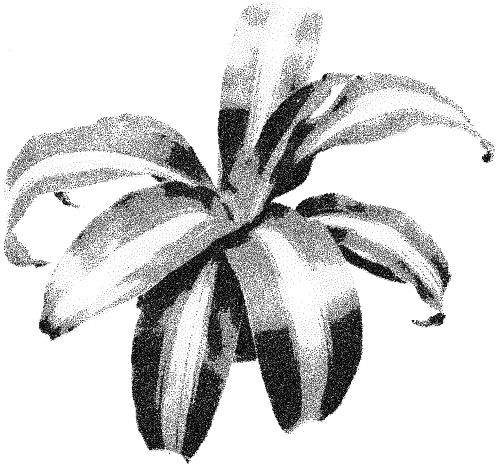
(d) 12,678 stipple dots.

(e) 12,908 stipple dots.

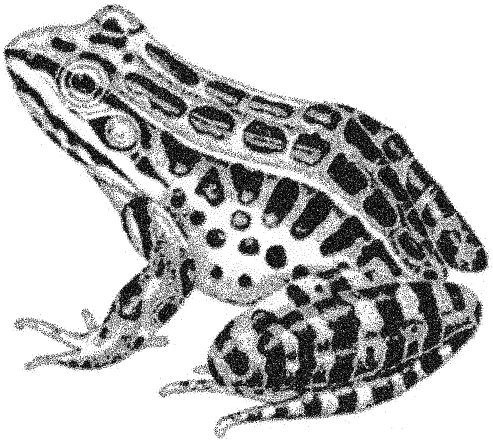


(f) 17,555 stipple dots.

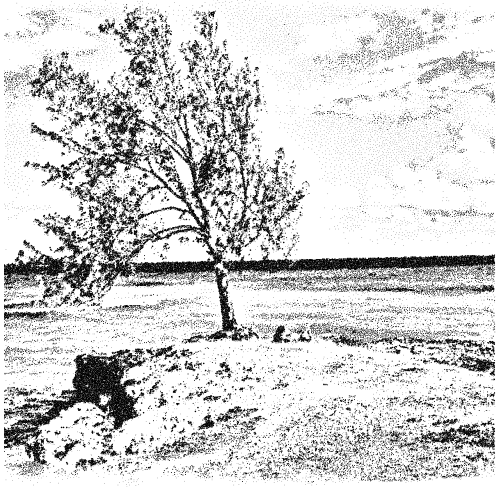
Figure 9: Stippling using Balzer et al.'s [7] capacity-constrained approach using the same stipple count for each of the images that was used by Martín et al.'s resolution-dependent grayscale approach [3, 4] (Figure 12). The images were generated with StippleShop.



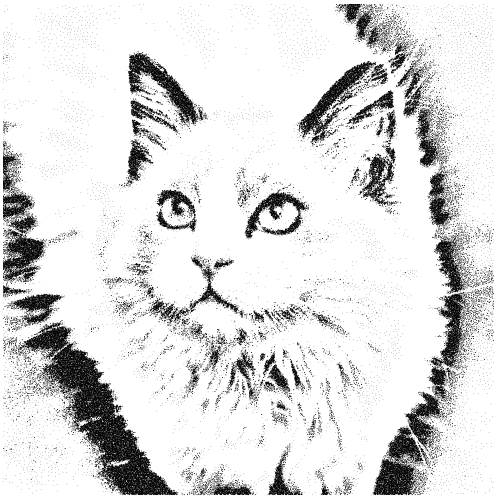
(a) 43,656 stipple dots.



(b) 59,349 stipple dots.

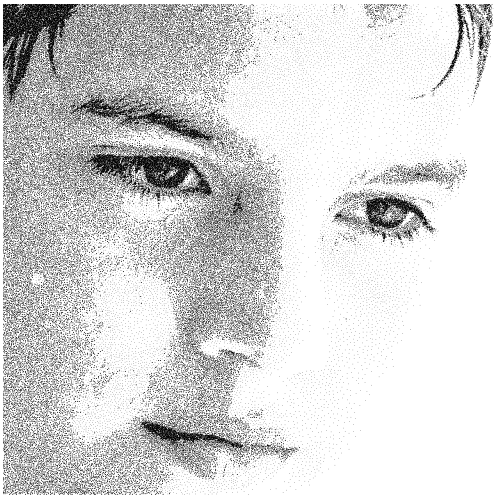


(c) 52,077 stipple dots.



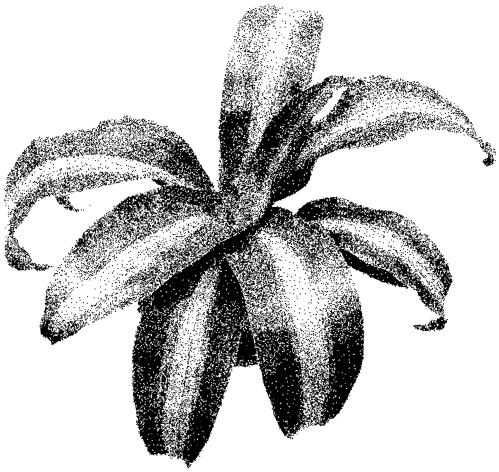
(d) 38,895 stipple dots.

(e) 39,618 stipple dots.

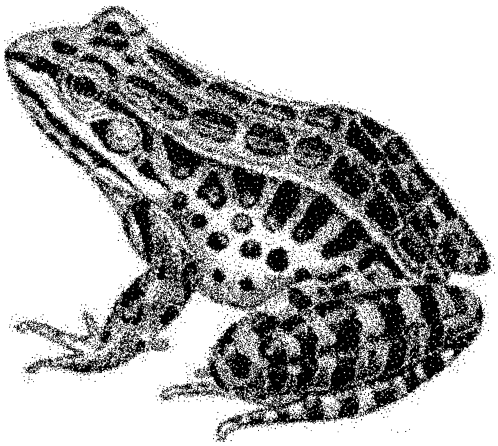


(f) 53,687 stipple dots.

Figure 10: Stippling using Kopf et al.'s [8] Wang tiling. The images were generated with Kopf et al.'s original tool.

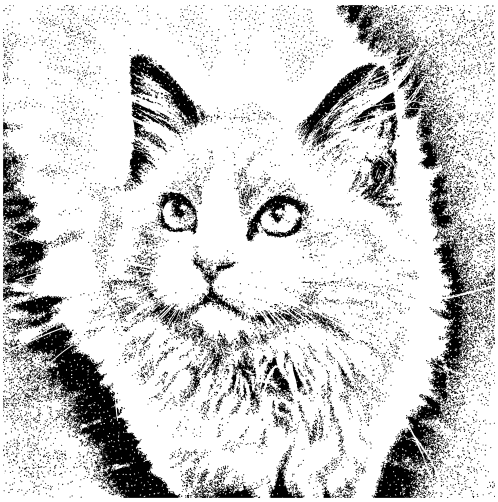


(a)



(b)

(c)

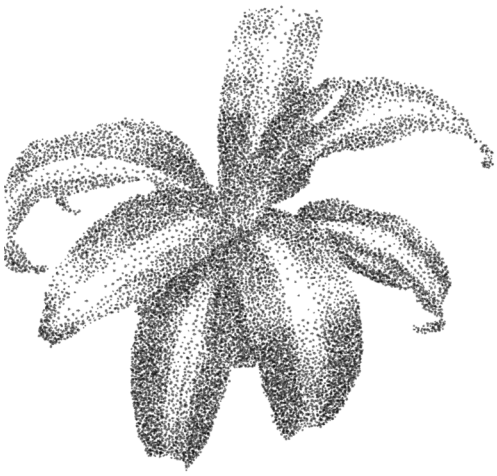


(d)

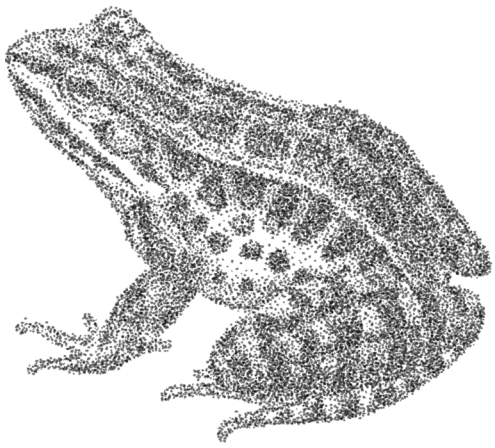
(e)

(f)

Figure 11: Stippling using Kim et al.'s [5] example-based approach, using their captured stipple profile and their code which was included into StippleShop. For this purpose we resized all source images to 4096×4096 using Gimp's cubic interpolation.

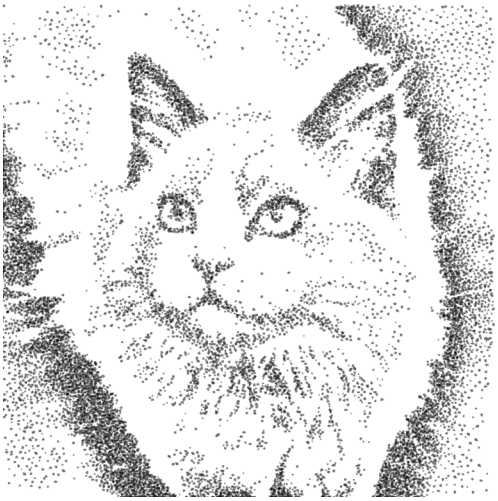


(a) 14,269 stipple dots.



(b) 19,405 stipple dots.

(c) 17,044 stipple dots.

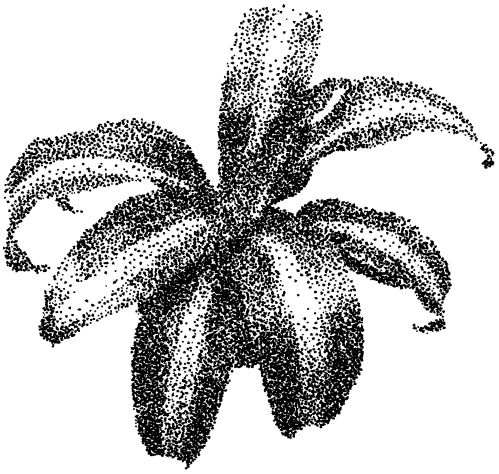


(d) 12,678 stipple dots.

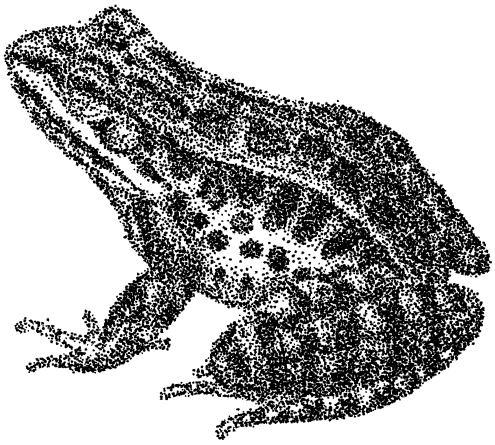
(e) 12,908 stipple dots.

(f) 17,555 stipple dots.

Figure 12: Stippling using Martín et al.'s [3, 4] resolution-dependent grayscale approach. The images were generated with StippleShop, using the target size of 65mm × 65mm and the target resolution of 300 ppi using grayscale output.

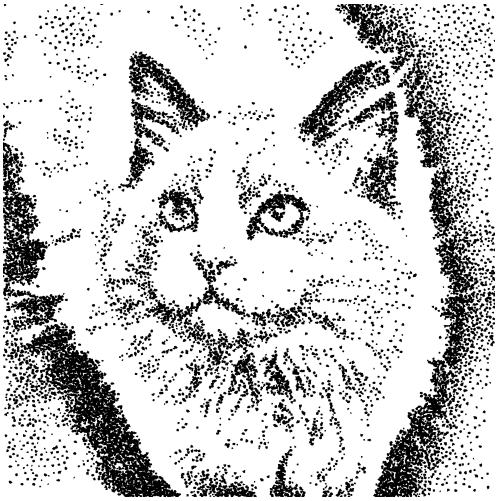


(a) 14,269 stipple dots.



(b) 19,405 stipple dots.

(c) 17,044 stipple dots.



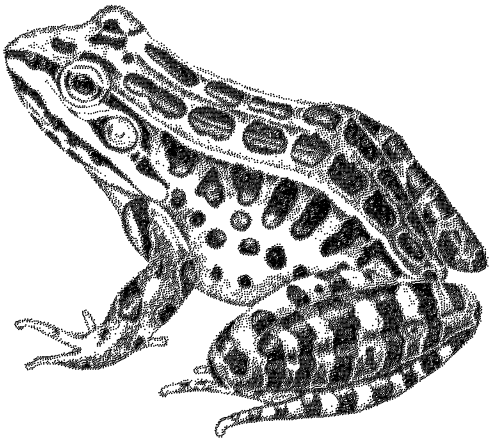
(d) 12,678 stipple dots.

(e) 12,908 stipple dots.

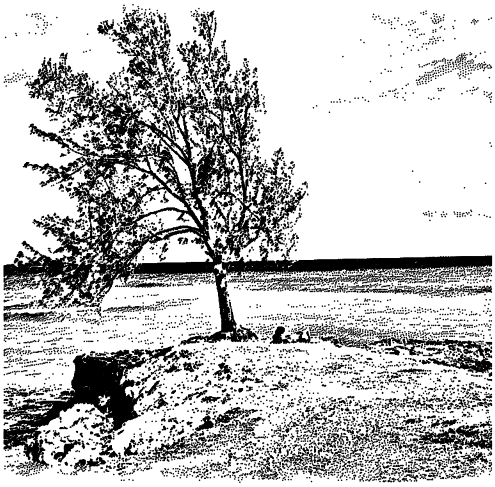
(f) 17,555 stipple dots.

Figure 13: Stippling using Martín et al.'s [3, 4] resolution-dependent grayscale approach. The images were generated with StippleShop, using the target size of 65mm × 65mm and the target resolution of 1200 ppi using black-and-white output.

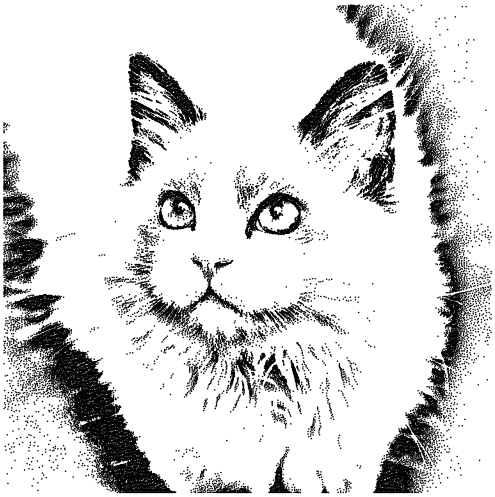
(a) 28,458 stipple dots.



(b) 41,129 stipple dots.

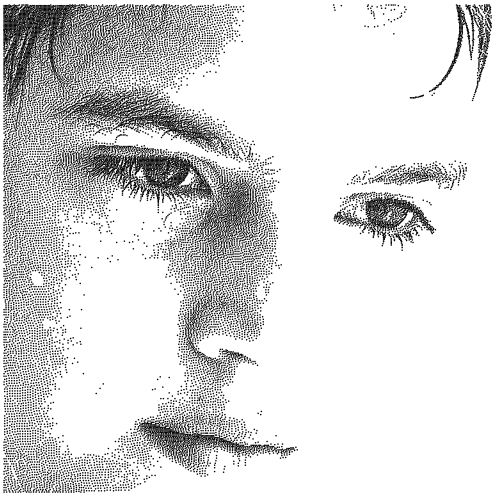


(c) 71,006 stipple dots.



(d) 40,026 stipple dots.

(e) 34,943 stipple dots.



(f) 21,288 stipple dots.

Figure 14: Stippling using Li and Mould's [9] structure-dependent approach. The images were generated with StippleShop.