Real-time Structure Aware Color Stippling

Lei Ma
Peking University
AICFVE, Beijing Film Academy

Hong Deng
Nanjing University of Science and Technology

Beibei Wang
Nanjing University of Science and Technology

Yanyun Chen
Institute of Software, Chinese Academy of Sciences

Tamy Boubekeur
LTCI, Telecom ParisTech, Paris-Saclay University

Figure 1: Comparison between our method, k-class blue noise sampling [Wei 2010]. The reconstructed image from our stippling result is also presented. Our method is the fastest (5.3 ms) and produces high quality result. The MSE is computed between the input image with the reconstruction image from our stippling result.

CCS CONCEPTS

• **Computing methodologies → Rendering**.

KEYWORDS

Color Stippling, Incremental Voronoi Set, Blue Noise Sampling

ACM Reference Format:
Lei Ma, Hong Deng, Beibei Wang, Yanyun Chen, and Tamy Boubekeur. 2019. Real-time Structure Aware Color Stippling. In *Proceedings of SIGGRAPH '19 Posters.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3306214.3338606

1 OVERVIEW

In computer graphics, stippling is a widely used non-photorealistic rendering technique. As the art of representing images with dots, one of the key problems is the placement of dots. In general, they should be distributed evenly, and with some randomness at the same time. Blue noise methods provide these characteristics and are used by state-of-the-art gray-scale algorithms to distribute dots. Color stippling, however, is more challenging as each channel should have even distribution at the same time. Existing approaches cast color stippling as a multi-class blue noise sampling problem and provide high quality results at the cost of a very long processing time. In this paper, we propose a real-time structure aware method for color stippling, based on samples generated from an incremental voronoï set. Our method can handle an arbitrary input color vector for stippling and produce significantly better results than previous methods, at real-time frame rate. We evaluate the perceptual quality of our stippling with a user study and its numerical performance by measuring the MSE between the reconstructed image from the stippling and the input image. As a result, the real time performance of our method makes interactive stippling editing possible, providing the artist with an effective tool to explore quickly a wide space of color image stippling.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
SIGGRAPH '19 Posters, July 28 - August 01, 2019, Los Angeles, CA, USA
© 2019 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-6314-3/19/07.
https://doi.org/10.1145/3306214.3338606

2 PREVIOUS WORKS

Precomputed-set based stippling. Ma et al. [Ma et al. 2018a] proposed a novel method to generate 2D stipplings in real time, using a precomputed sample sequence called Incremental Voronoi Set (IVS). Compared to other 2D sample sequences, IVS is incremental and highly parallelizable while maintaining blue-noise properties. Our proposed color stippling method is based on this IVS. This work was latter extended to 3D stippling [Ma et al. 2018b].

Color Stippling. Wei [Wei 2010] extended blue noise sampling to multi-class blue noise sampling that can guarantee the blue noise property for individual class and their union. He presented two solutions for multi-class sampling: hard disk sampling for controlling the sample spacing and soft disk sampling for controlling the sample count. This method can provide high quality color stippling, but is very time consuming. We consider it as the quality reference.

Structure Aware Stippling. Structure-aware stippling methods [Li and Mould 2010, 2011; Pang et al. 2008] emphasize image edges or the localized contrast perceived in the input image. We designed our stippling method to account for the image structure information as well.

We refer the readers to the survey of Martin et al. [Martín et al. 2017] for a complete overview.

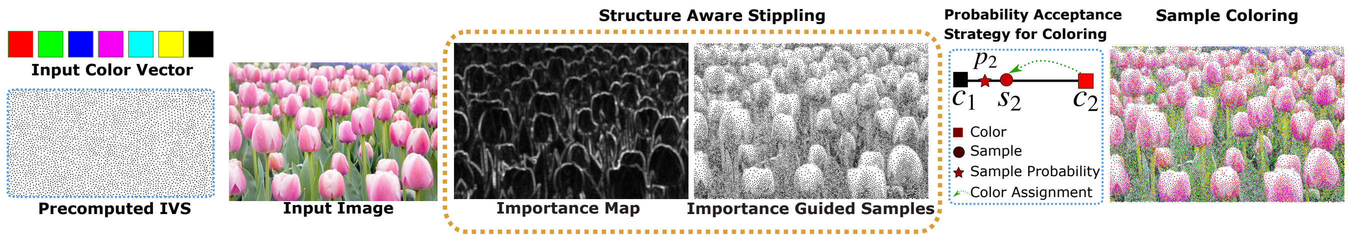

Figure 2: Starting from the input image, a precomputed sequence of sample and an input color vector, e.g. RGBCMYK, we first construct an importance map by measuring the difference between pixels, and then use the importance map to guide the sample rejecting process resulting in structure aware samples. We use a *probability acceptance strategy* to assign a color from the color vector to each sample, and get the final image.

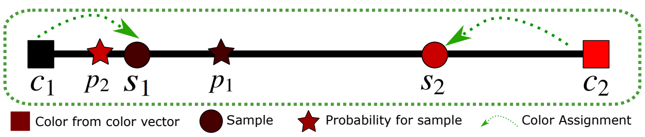

Figure 3: We assign two samples s_1 and s_2 with two colors (black and red) in the color vector. Each sample has its own p_i. Compare the input color of sample and the its p_i, and then assign a color smaller than p_i.

3 STRUCTURE AWARE COLOR STIPPLING

Our method builds upon IVS and contains two novel components: a *probabilistic coloring* scheme and a *structure-aware importance map*. The prescribed color vector can be any color combination. Let $C = \{c_k\}$ be this vector, where $0 < k \leq n$ and n is the total color count. Since we used pre-determined sample locations, the mapping $F(s_i)$ from the sample space $\mathcal{S}$ to the target color vector C is the key component of stippling.

Probabilistic Coloring in a 1D Color Space. Assuming there are only two colors in $C = \{c_1, c_2\}$, simple solutions such as nearest neighbor reconstruction leads to sharp transitions or factorization pattern between colors. Instead, we use a statistical approach, by introducing a probability acceptance strategy. More precisely, we associate to each sample s_i a probability p_i to accept one of the two colors. Any colors which are smaller than p_i, will be assigned to the c_1, otherwise c_2 (see Figure 3). The probability p_i is defined from the index i of the sample in the IVS: $p_i = \frac{1}{i+1}$. As a result, the transition between colors is smooth and preserves entropy. In general cases, where $C = \{c_1, ..., c_N\}$, we choose a tight interval (the smallest interval) where the sample initial color is located, and then use the probability acceptance strategy on this interval. The p_i is mapped from the entire domain to the tight interval correspondingly.

Probabilistic Coloring in Higher Dimensional Color Spaces. For $C = \{c_1, ..., c_N\}$, each c_i is presented as high dimensional data such as $v_0, v_1, ..., v_d$. We extend p_i into a probability vector, and it aligns along the diagonal. For the 2D case, this divides the 2D color space into four quadrants. We choose the quantized color based on the quadrant where the input sample is mapped. We then compute the Axis Aligned Bounding Boxs (AABBs) in the 2D color space that fit color locations the best. For each AABB, we remap (p_i, p_i) into it, process it similarly to the simple case, and then choose a color. If this color is not a present in the color vector, we ignore it; otherwise, we add it to the set of color candidates. Then, among all the color candidates, we compare their L_∞ distance to the sample color, and use the color which has the smallest distance as the final quantized color. Extension from the 2D case to the (traditional) 3D case (e.g., RGB) is straightforward using a 3D bounding box. In our implementation, dots radii r_i is provided as an input parameter. Then the required samples from IVS is: $m = \frac{4 I_{res}}{3\pi \times r^2}$, where I_{res} is the image resolution.

Structure Aware Stippling. To properly capture the image details, we use the image structure to guide the stippling process. Our basic observation is that the closer to a feature the sample is, the smaller its radius should be. We compute the absolute difference between the pixel and its four neighbors to represent the structure importance. We then filter the obtained structure importance map to avoid noise.

4 RESULTS AND CONCLUSION

The whole pipeline of method is demonstrated in Figure 2. One of our typical results is represented in Figure 1. Different from the optimization-based approaches, our method is highly parallelizable. The real time performance achieved by our method stems from inherent fine-grained parallel execution. We also consider the image features to capture the details in the stippling. Overall, we can provide high quality stippling results at real-time frame rate with scalability on both the dot count and the image resolution. Our method makes possible interactive stippling editing. We believe our algorithm can be a practical tool for artists.

REFERENCES

Hua Li and David Mould. 2010. Contrast-aware Halftoning. *Computer Graphics Forum* 29, 2 (2010), 273–280.

Hua Li and David Mould. 2011. Structure-preserving Stippling by Priority-based Error Diffusion. In *Proceedings of Graphics Interface 2011 (GI '11)*. 127–134.

Lei Ma, Yanyun Chen, Yinling Qian, and Hanqiu Sun. 2018a. Incremental Voronoi sets for instant stippling. *The Visual Computer* 34, 6 (2018), 863–873.

Lei Ma, Jianwei Guo, Dong-Ming Yan, Hanqiu Sun, and Yanyun Chen. 2018b. Instant Stippling on 3D Scenes. *Computer Graphics Forum* 37, 7 (2018), 255–266.

Domingo Martín, Germán Arroyo, Alejandro Rodríguez, and Tobias Isenberg. 2017. A survey of digital stippling. *Computers & Graphics* 67 (2017), 24 – 44.

Wai-Man Pang, Yingge Qu, Tien-Tsin Wong, Daniel Cohen-Or, and Pheng-Ann Heng. 2008. Structure-aware Halftoning. *ACM Trans. Graph.* 27, 3 (Aug. 2008), 89:1–89:8.

L.-Y. Wei. 2010. Multi-class blue noise sampling. *ACM Trans. on Graphics (Proc. SIGGRAPH)* 29, 4 (2010), 79:1–79:8.