Supplemental Material: An Inverse Procedural Modeling Pipeline for SVBRDF Maps

YIWEI HU, Yale University, USA
CHENGAN HE, Yale University, USA
VALENTIN DESCHAINTRE, Adobe Research and Imperial College London, UK
JULIE DORSEY, Yale University, USA
HOLLY RUSHMEIER, Yale University, USA

1 SVBRDF DECOMPOSITION RESULTS

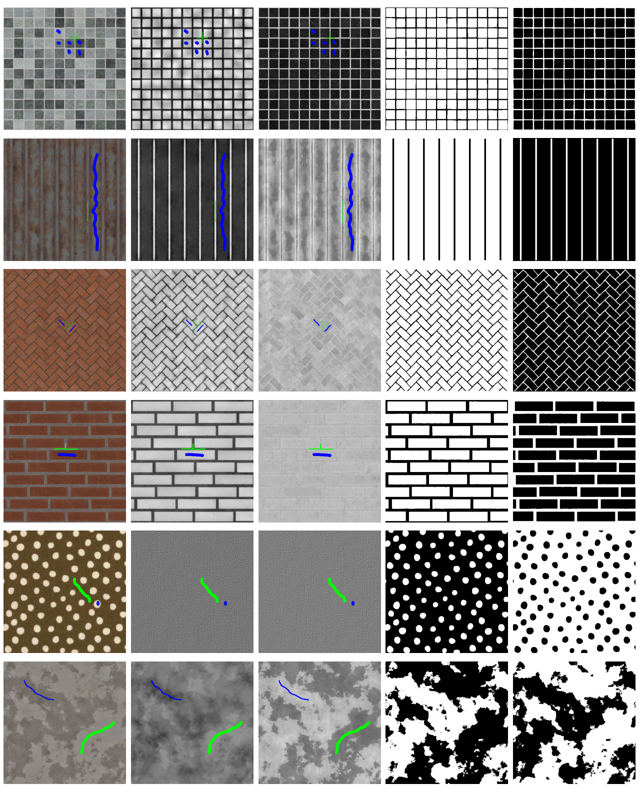


Fig. 1. Results of our SVBRDF decomposition method. We show our method can decompose material maps with various spatial structures in a few scribbles by the user. User scribbles are visualized as an additional layer superimposed over material maps, where blue and green scribbles specify different sub-materials.

We evaluate the efficiency of our interactive SVBRDF decomposition method. Fig. 1 contains decomposition results of various spatial structures which are achieved in a few scribbles by the user. Typically, one or two scribbles for each sub-material are enough to obtain good results, and the marking for sub-materials does not need to be careful or precise. For the material maps we experimented with, this process takes less than 2 minutes depending on the complexity of the input material maps.

2 COMPARISON RESULTS

In this section, we show our comparison results. Fig. 2 & 3 shows a comparison between our method with the state-of-the-art inverse material modeling methods. Different from ours, both of their frameworks require a collection of pre-defined material graphs. We applied [Hu et al. 2019]'s method to select a model from the database and use it as auxiliary input for their parameter estimation.

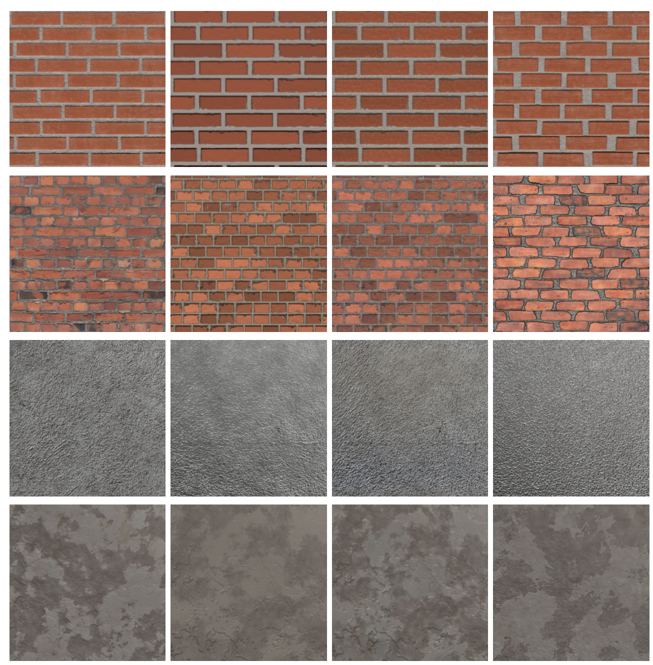
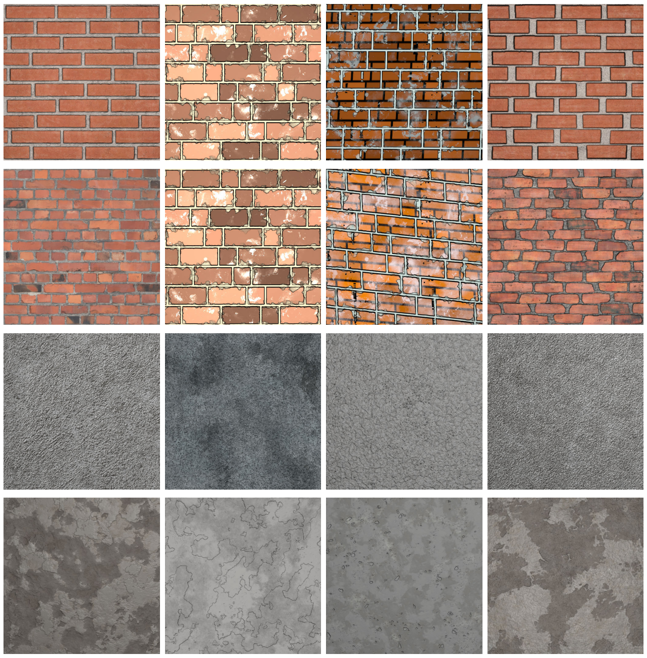


Fig. 2. Comparison of our method to [Hu et al. 2019]. The second column shows predicted procedural results by [Hu et al. 2019] while the third column is their style augmented results (non-procedural and no edibility). In contrast to their method, our pipeline generates fully procedural materials without a pre-existing material graph as an auxiliary input. The images are rendered using Blender with diffuse reflectance to match [Hu et al. 2019].

Authors' addresses: Yiwei Hu, Yale University, USA, yiwei.hu@yale.edu; Chengan He, Yale University, USA; Valentin Deschaintre, Adobe Research, Imperial College London, UK; Julie Dorsey, Yale University, USA, julie.dorsey@yale.edu; Holly Rushmeier, Yale University, USA, holly.rushmeier@yale.edu.

2021. 0730-0301/2021/11-ART $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn



Input Default MATch Ours

Fig. 3. Comparison of our method to MATch [2020]. We use Hu et al. [2019]'s framework to select a close Substance model, and use MATch to optimize its appearance. As the MATch framework does not handle discrete parameters and requires good initialization, it can generate poor output if the initialization and the discrete parameters are not hand-tuned (as seen in the brick materials). Materials are rendered using the GGX shading model.

We include comparison with the state-of-the-art example-based texture synthesis in Fig. 4. We generalize their methods, which are originally designed to process color textures, to process multi-channel SVBRDF maps by stacking albedo map, normal and roughness maps together as their input. For self-tuning texture optimization [Kaspar et al. 2015], its generalization to multi-channel material maps is not trivial and we therefore run their algorithm separately on each material map. Since we synthesize each material map individually, structure mismatches arise leading to visible limitations in renderings.

REFERENCES

Alexei A. Efros and William T. Freeman. 2001. Image Quilting for Texture Synthesis and Transfer. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '01)*. Association for Computing Machinery, New York, NY, USA, 341–346. https://doi.org/10.1145/383259.383296

Yiwei Hu, Julie Dorsey, and Holly Rushmeier. 2019. A Novel Framework for Inverse Procedural Texture Modeling. *ACM Trans. Graph.* 38, 6, Article 186 (Nov. 2019), 14 pages. https://doi.org/10.1145/3355089.3356516

Alexandre Kaspar, Boris Neubert, Dani Lischinski, Mark Pauly, and Johannes Kopf. 2015. Self Tuning Texture Optimization. *Comput. Graph. Forum* 34, 2 (May 2015), 349–359. https://doi.org/10.1111/cgf.12565

Liang Shi, Beichen Li, Miloš Hašan, Kalyan Sunkavalli, Tamy Boubekeur, Radomir Mech, and Wojciech Matusik. 2020. MATch: Differentiable Material Graphs for Procedural Material Capture. *ACM Trans. Graph.* 39, 6, Article 196 (Dec. 2020), 15 pages.

Assaf Shocher, Shai Bagon, Phillip Isola, and Michal Irani. 2019. InGAN: Capturing and Retargeting the "DNA" of a Natural Image. In *The IEEE International Conference on Computer Vision (ICCV)*.

Yang Zhou, Zhen Zhu, Xiang Bai, Dani Lischinski, Daniel Cohen-Or, and Hui Huang. 2018. Non-stationary Texture Synthesis by Adversarial Expansion. *ACM Transactions on Graphics (Proc. SIGGRAPH)* 37, 4 (2018).

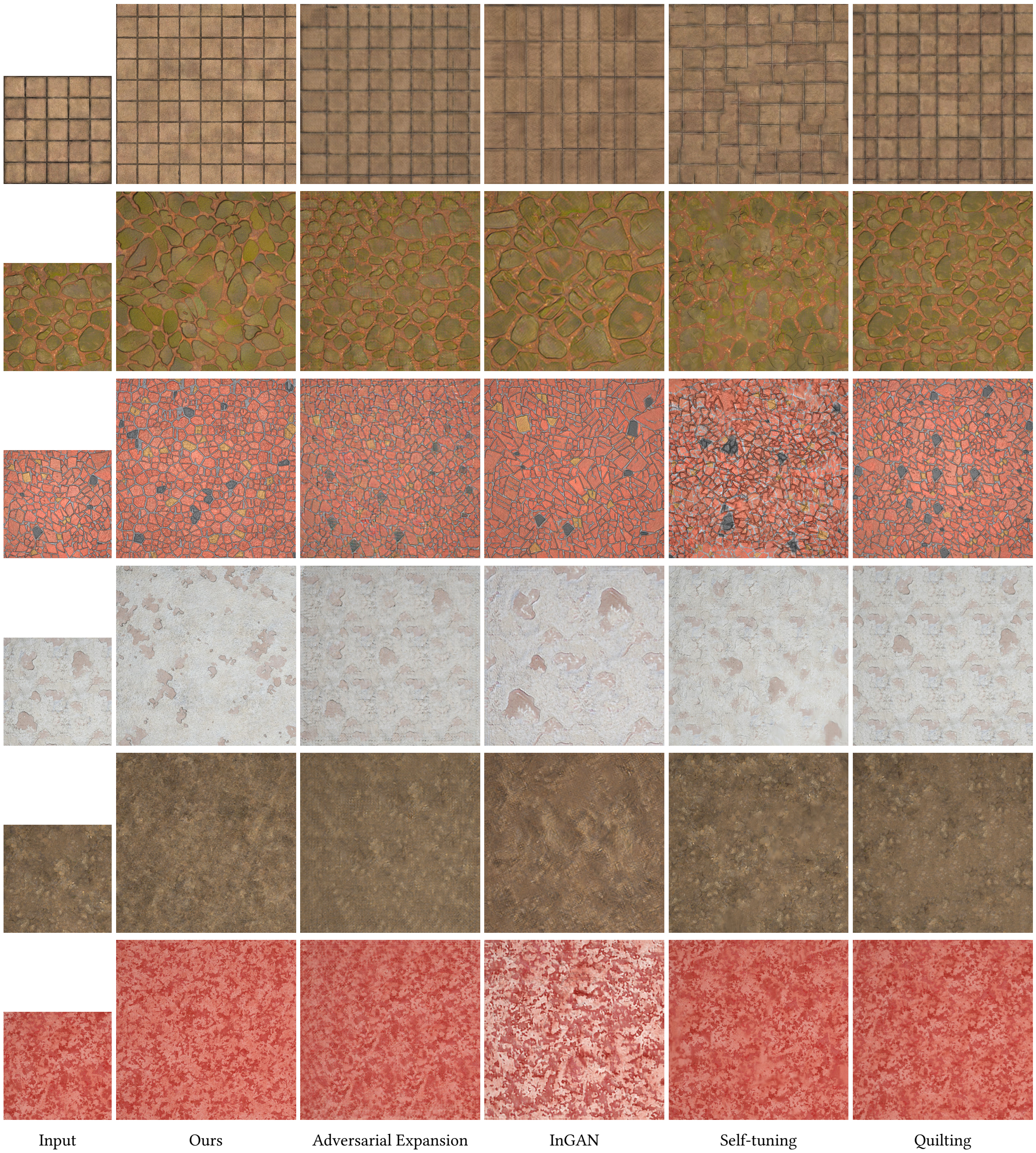


Fig. 4. Comparison of our method with example-based texture synthesis methods on SVBRDF maps. We generalize these methods to process multi-channel SVBRDF maps. We show our method; InGAN [Shocher et al. 2019]; Non-stationary Texture Synthesis by Adversarial Expansion [Zhou et al. 2018]; Self-tuning Texture Optimization [Kaspar et al. 2015]; Image Quilting [Efros and Freeman 2001]. Images shown here are rendered with the GGX shading model.