Patch-Based Geometric Texture Synthesis

Zainab AlMeraj*
University of Waterloo
Kuwait University

Craig S. Kaplan
University of Waterloo

Paul Asente
Adobe

Abstract

Inspired by the results of recent studies on the perception of geometric textures, we present a patch-based geometric synthesis algorithm that mimics observed synthesis strategies. Our synthesis process first constructs an overlapping grid of copies of the exemplar, and then culls individual motifs based on overlaps and the enforcement of minimum distances.

CR Categories: I.3.3 [Computer Graphics]: Picture/Image Generation—Line and curve generation;

Keywords: Non-photorealistic rendering, texture synthesis, vector graphics

1 Introduction

Geometric Texture Synthesis (GTS) refers to a class of algorithms for filling a given container region with small vector motifs so that the resulting distribution of motifs achieves a predetermined aesthetic goal. This problem may be viewed as the geometric analogue of raster-based texture synthesis [Wei et al. 2009], without the simplifying assumption of a uniformly sampled spatial domain. It also encompasses research on packing algorithms for non-photorealistic rendering [Dalal et al. 2006], in which the goal is usually to distribute motifs so that the space between them is minimized, or as even as possible.

As with raster texture synthesis, GTS algorithms are usually example-based: the user provides an *exemplar* (a "typical" distribution of motifs within a small area) and the algorithm elaborates a similar-looking distribution over a larger container region. The visual goal, then, is to ensure that the synthesized distribution captures the idiosyncrasies of the exemplar. The corresponding challenge is that we lack a rigorous model of texture perception that can characterize these idiosyncrasies, or measure whether they have been replicated.

One way to interrogate the human experience of motif distributions is to ask human subjects to synthesize distributions manually, and to have them evaluate the similarity of distributions to the exemplars that inspired them. AlMeraj et al. [2011] conducted such a study. Their analysis yielded a set of concrete visual cues used in similarity assessments, as well as a set of high-level strategies adopted by participants during the synthesis process. They identify the following three strategies:

- **Tiling**: place motifs so that they approximate a tiled layout of copies of the exemplar.
- **Structured**: place motifs so that they replicate substructures found in the exemplar, such as clusters or filaments of closely spaced objects.
- **Random**: place motifs randomly so that they capture high-level statistical features of the exemplar, such as density and relative frequencies of distinct shapes.

These three strategies lie on a continuum: the Tiling approach clearly captures the structure of the exemplar very well, but the obvious repetition is usually objectionable. The Random approach can easily generate distributions with no repetition, but cannot account for inhomogeneities in the exemplar. The Structured approach strikes a desirable balance between these extremes, but it is also the hardest to formalize as an algorithm.

Most previous GTS algorithms can be seen as injecting Structure into a Random process, by optimizing an initially random distribution or by placing motifs one at a time. Based on the work of AlMeraj et al., our goal here is to explore a Tiling-driven approach to geometric texture synthesis. Instead of asking what must be added to a Random distribution to increase its similarity to an exemplar, we begin with a Tiling of exemplars and ask where order might be *removed* to suppress signs of repetition.

Our algorithm can be viewed as a geometric analogue of patch-based approaches in raster texture synthesis [Efros and Freeman 2001]. It is the first algorithm in this area that is based directly on a psychophysical study of how humans respond to geometric textures. As a GTS algorithm, it has the advantage of simplicity, paving the way for a robust, interactive implementation in real-world illustration software. We also hope that it will serve as one more data point in ongoing research on similarity measures and evaluation strategies for geometric texture synthesis algorithms.

2 Related work

A variety of approaches to geometric texture synthesis have been advanced in recent years. The paper by Ma et al. [2011] contains a thorough survey of relevant literature up to 2011.

One recurring theme in this area is to examine, for each motif in the exemplar, the local arrangement of its immediate neighbours. Neighbour relationships are often approximated by choosing an anchor point for each motif, such as its centroid, and computing a Delauney triangulation of the anchor points; edges in the triangulation then denote neighbours. Barla et al. [2006] construct a Poisson distribution of "seed" points and assign motifs to these seeds by matching triangulation neighbourhoods from the exemplar. Ijiri et al. [2008] avoid constructing an initial distribution of seeds by copying portions of 1-rings directly from the triangulation of the exemplar into the synthesized texture.

Alternatively, we may view the neighbourhood of a motif more holistically, measuring distances to some or all of the other motifs in the exemplar. Hurtut et al. [2009] develop a complex statistical appearance model based on distances between motifs. To synthesize a new texture, they begin with a random motif distribution and perturb it iteratively via additions and deletions of individual motifs, accepting perturbations according to their consistency with the

*e-mail: z.almeraj@gmail.com

CAe 2013, July 19 – 21, 2013, Anaheim, California.

statistical model of the input. Alves dos Passos et al. [2010] describe a motif-at-a-time growth model like that of Ijiri et al., but one in which neighbourhood similarity is based on a direct distance metric rather than a Delaunay triangulation.

Ma et al. [2011] adopt a hybrid approach. They augment motif shapes with a set of regularly-spaced sample points, and measure neighbourhood similarity based on positions of samples around a core sample. They iteratively improve an initial distribution via an energy minimization framework, moving individual motifs in order to lower a neighbourhood dissimilarity metric. Their technique is notable for its generality: it can synthesize in two or three dimensions, with rigid or deformable motifs, taking into account additional constraints such as boundary sensitivity.

Our approach is generally inspired by patch-based algorithms in raster texture synthesis, such as image quilting [Efros and Freeman 2001; Lefebvre and Hoppe 2005]. Similar techniques have been applied to generate thin shells of geometry around mesh surfaces [Zhou et al. 2006]. Some patch-based ideas have also made their way into GTS research. Alves dos Passos et al. [2010] discuss the copying of patches purely as an optimization for efficiency. Ma et al. [2011] use patch-based copying as an initialization strategy for their initial distribution, but do not consider a complete algorithm founded on patches.

3 A patch-based method

As stated in the introduction, our algorithm is based on the results of a two-phase psychophysical study by AlMeraj et al. [2011]. In the first phase, participants were given small, simple exemplars and instructed to synthesize larger textures that are visually similar. In the second phase, new participants ranked these textures based on their similarity to the exemplars upon which they were based. The first phase identified three main synthesis strategies used by participants, as shown in Figure 1 and discussed previously. The second phase demonstrated the importance of repetition in judgments of similarity: the Tiling strategy was most likely to produce arrangements deemed similar to their exemplars, followed by the Structured strategy.

A synthesis algorithm based purely on Tiling is trivial to implement: we need only stamp out regularly spaced copies of the exemplar. This algorithm clearly captures nearly all of the exemplar's statistical properties, but imposes an obvious repetitive structure that may not be intended. Our goal is to seek a minimal set of modifications to the Tiling approach that suppresses these signs of repetition. We begin with the notion of *overlapping* tiled copies of the exemplar, and work through the consequences of that decision.

In the subsections that follow, we describe the steps in our algorithm. Beginning with a suitably prepared exemplar (Section 3.1), we create a regular grid of overlapped copies (Section 3.2), resolve overlaps (Section 3.3), adjust the arrangement for density and minimum distances (Section 3.4), and vary motif orientations (Section 3.5). These steps are summarized in Figure 2.

3.1 Creating an exemplar

Like Ijiri et al. [2008] and Ma et al. [2011], and unlike Hurtut et al. [2009], we begin with the simplifying assumption that the exemplar will consist of a set of non-overlapping transformed instances of a smaller number of distinct primitive shapes, which we denote $\{S_1, \ldots, S_k\}$. This limitation is discussed further in Section 4. An exemplar of this type can readily be expressed in the Scalable Vector Graphics (SVG) format, which comes equipped with `<symbol>` and `<use>` tags to define and place reusable motifs. The instances are restricted to a subset of the plane we call the *input region*, usually a circle or square.

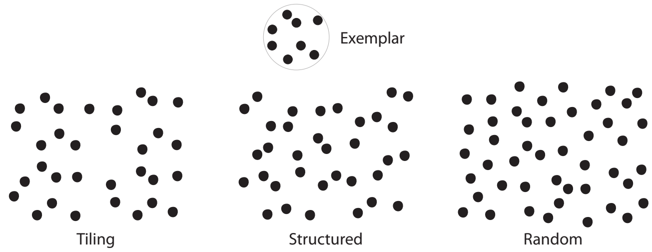

Figure 1: *Examples of arrangements created by participants in the study by AlMeraj et al. [2011]. The arrangements show typical examples of the Tiling, Structured and Random strategies.*

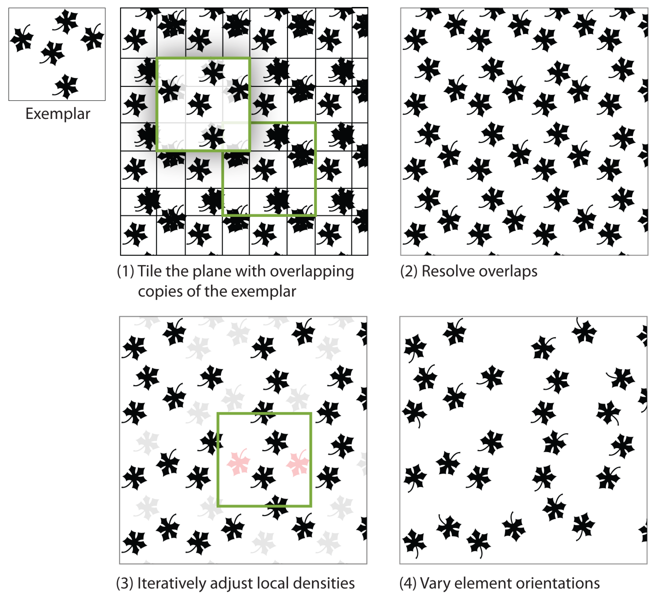

Figure 2: *An illustration of the steps in our patch-based synthesis algorithm.*

In our output texture, we will want to avoid placing two motifs such that the distance between them is less than the minimum distance between any instances of the same two primitives in the exemplar. Two motifs closer than this minimum distance might suggest a grouping in the output texture that was not present in the input. Therefore, in a preprocessing step we gather all pairwise distances between primitives. Given any two non-overlapping vector shapes A and B, we define $d(A, B)$, the distance between A and B, as

$$d(A, B) = \min_{p \in A, q \in B} ||p - q||,$$

that is, the smallest distance between any point in A and any point in B. (This distance can be computed in quadratic time for two polygons, and approximated for arbitrary shapes by converting them into polygons.) We use this distance function to compute a symmetric matrix of values d_{ij}; each d_{ij} is the minimum of all distances $d(A, B)$ where A is an instance of S_i and B is an instance of S_j. In a synthesized arrangement, we say that an instance of S_i "violates the distance rule" if there is a neighbouring instance of S_j that is closer to it than d_{ij}.

3.2 Constructing a grid

Let us assume that the input region is a square of side length r (the algorithm can easily be modified to accommodate other region shapes). Define a fractional overlap amount σ between 0 and 0.5; for most exemplars, $\sigma = 0.3$ is sufficient. We construct an initial distribution by placing copies of the exemplar translated by vectors of the form $(ar(1-\sigma), br(1-\sigma))$ for integers a and b. We use as many (a, b) pairs as necessary to cover the target region for synthesis, which we call the *output region*. Of course, non-square grid layouts are possible as well; we have also experimented with a hexagonal tiling of exemplars, particularly when the input region is a circle.

Copies of the exemplar laid out this way can frequently exhibit too much regularity. We suppress global regularity while preserving local structure by randomly rotating each exemplar copy. We rotate the copy by a rotation angle θ chosen uniformly at random from the range $[-M, M]$, where M is a user-definable limit that should depend on the overall regularity of the exemplar. However, random rotation might disrupt any perceived directionality of the exemplar; in the example of Figure 4(c), the motifs have a clear horizontal orientation. We avoid this problem by rotating each *motif* in this copy by $-\theta$, returning it to its original orientation.

3.3 Resolving overlaps

The immediate consequence of allowing copies of the exemplar to overlap above is that individual motifs might overlap in the synthesized arrangement. We must remove enough motifs to eliminate all overlaps.

We search the initial arrangement created above for pairs of overlapping motifs. Removing either motif will resolve the overlap. We implement an additional heuristic to improve the quality of our output: if either overlapping motif is found to violate the distance rule relative to its non-overlapping neighbours, it is selected preferentially for removal.

We believe that by removing motifs after having synthesized too many, our algorithm is more likely to preserve the visual properties of the exemplar than an algorithm based on placing motifs into an initially empty output region.

3.4 Adjusting density

The arrangement produced in the previous step should consist of non-overlapping motifs, and should in some sense be visually similar to the exemplar. However, because overlap resolution is a purely local operation between pairs of motifs, the arrangement's density might differ too much, both globally and locally, from that of the exemplar. We propose an iterative adjustment step that attempts to restore the approximate desired density.

We can measure the local density of an arrangement within any region of the plane by adding the areas of all the motifs that intersect the region, and dividing by the region's area. Let ρ be the density of the exemplar within the input region. Roughly speaking, if we overlay the input region anywhere on the synthesized arrangement, we would like the density within that window to be close to ρ. We can minimize any discrepancies by adding or removing motifs as necessary. If the local density is too high, we remove motifs at random to lower it, again favouring motifs that violate the distance rule. If it is too low, we search for the largest empty disc contained in the window, and insert the contents of a congruent disc superimposed at random over the exemplar. We reject the insertion and try another if any of the added motifs would violate the distance rule.

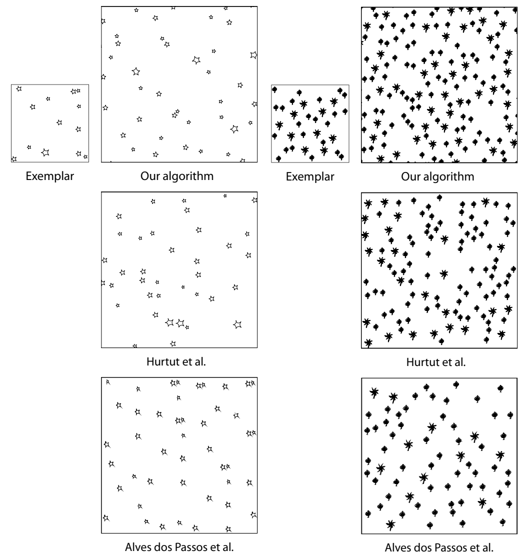

Figure 3: *Patch-based synthesis results based on two exemplars that appeared in the papers of Hurtut et al. [2009] and Alves dos Passos et al. [2010].*

We apply this process iteratively until the density of the synthesized arrangement is sufficiently close to that of the exemplar.

3.5 Varying orientations

The exemplar consists of placed instances of a set of primitive shapes; the rotational component of the matrix that carries out the placement for each motif defines its orientation. We can optionally give synthesized motifs the ability to rotate in the final arrangement, but we would like the resulting orientations to be constrained to those found in the exemplar.

In our algorithm, for each primitive shape we compute the smallest arc of the circle (representing all possible orientations) that contains the orientations of all instances of that shape, and sample new orientations uniformly at random from that range. Motifs are then rotated about their centroids into their new orientations. As a result, strongly anisotropic textures such as that of Figure 4(b) avoid undesirable variations in orientation. A more fine-grained approach might sample from narrow intervals around the orientations that occur in the exemplar.

4 Results and discussion

We have implemented our synthesis algorithm as a standalone C++ application, and as a prototype plug-in for Adobe® Illustrator®. We find that it produces generally satisfactory results across a range of exemplars, particularly those that do not have too much regularity. We offer Figure 3 as a purely subjective comparison between our algorithm and those of Hurtut et al. [2009] and Alves dos Passos et al. [2010]. We believe that our results are competitive with these algorithms. In some places we outperform them, for example in the closely spaced pairs of stars in the left example, or the overall density and ratios of primitives in the right one. Our results demonstrate how easily repetition artifacts can be suppressed in an initial Tiling-based layout of motifs, through a small amount of rotation, together with a few additions or deletions of motifs.

One particularly apposite domain for geometric texture synthesis is digital cartography, especially its application to geology. Mapmak-

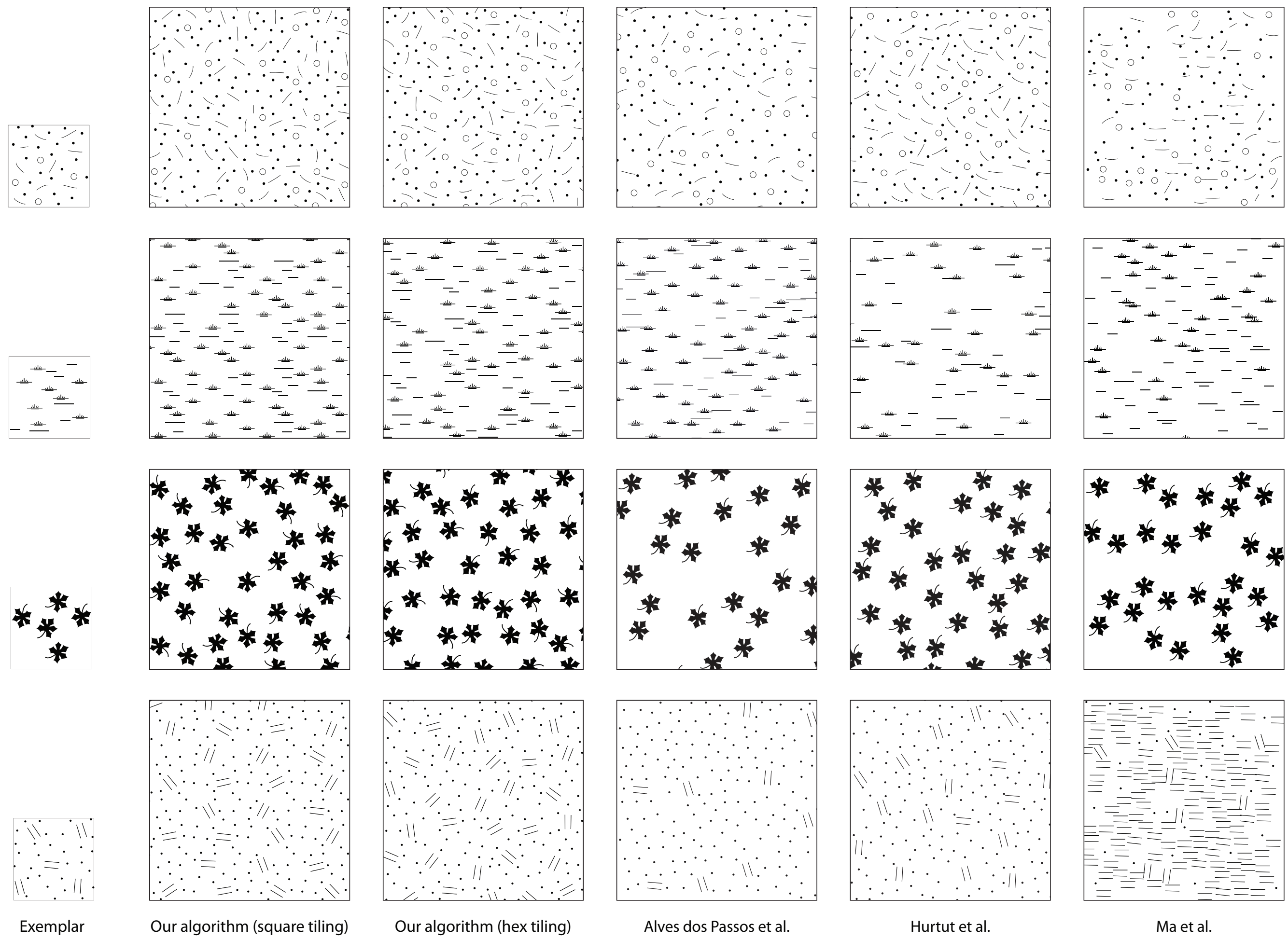

Figure 4: *Patch-based synthesis results based on four different exemplars adapted from the FGDC database mentioned in Section 4. The exemplars are synthesized using our algorithm with square and hexagonal layouts of tiles, as well as the algorithms of Alves dos Passos et al. [2010], Hurtut et al. [2009], and Ma et al. [2011].*

ers regularly fill regions with arrangements of markings for different terrains, minerals, land features, and so on, and they specifically wish for those arrangements to be irregular and organic. A small amount of research in the world of cartography has sought to develop algorithms akin to GTS [Jenny et al. 2010]. We believe that cartographic examples can form a rich, real-world set of inputs for current and future GTS algorithms.

With this domain in mind, we have sought to demonstrate our algorithm using geological textures. The US Geological Survey (USGS) publishes a standard reference for geological map symbols [US FGDC 2006]. We have constructed exemplars from several of the textures found there and synthesized larger arrangements using our algorithm; four such exemplars appear in the leftmost column of Figure 4. The second and third columns of that figure show our synthesized results, based on square and hexagonal tile layouts. In a concurrent investigation that attempts to evaluate the effectiveness of GTS algorithms [AlMeraj et al. 2013], we sent these exemplars to Alves dos Passos et al., Hurtut et al., and Ma et al., who provided arrangements synthesized with their algorithms, shown in the remaining columns.

Here we report the results of one of the empirical studies conducted in that investigation pertaining only to the results illustrated in Figure 4. On a computer screen, each participant was shown a sequence of comparisons containing an exemplar and two arrangements generated from that exemplar. For each pair of arrangements, they were asked to select the one that appeared most similar to the exemplar. Each participant was shown, in random order, every possible pairing of textures from each row of Figure 4. There were $\binom{5}{2} = 10$ possible pairings, four arrangement types, and 20 partic-

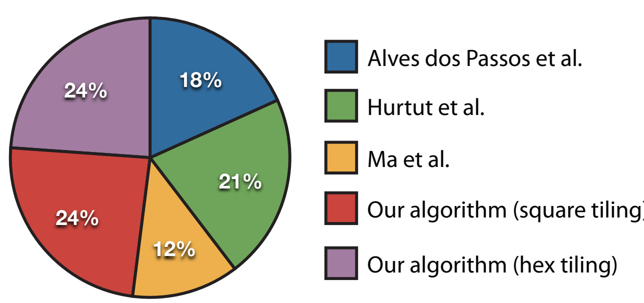

Figure 5: *A visualization of the relative percentages of times each algorithm's synthesized arrangement was chosen over another algorithm's, summed over all participants and all arrangement types.*

ipants, for a total of 800 individual comparisons.

Aggregating these comparisons, we arrive at a distribution of preferences as shown in Figure 5. We find that in side-by-side comparisons, our algorithm's arrangements were judged to be more similar more often than arrangements produced by other algorithms. There is no significant preference in the data between the square and hexagonal arrangements of tiles. On the other hand, in our experience the behaviour of GTS algorithms can vary across different exemplars, and the comparison process is subjective. This research area is lacking clear notions of similarity and aesthetic quality for geometric arrangements, and rigorous methodologies for comparing algorithms. We hope to explore these measures and methodologies in concurrent and future work [AlMeraj et al. 2013].

Our algorithm suffers from several limitations, which suggest avenues for future research. Most obviously, it requires the exemplar to contain instances of a small number of primitive shapes. We cannot handle an exemplar in which every motif is a distinct shape, as is the case in many USGS textures. In such cases, one option would be to explore a motif categorization step similar to that used by Hurtut et al. [2009]. We could then select instances at random from among the shapes in each category. Even better would be to build a GTS algorithm on top of an underlying example-based shape synthesis algorithm (see, for example, the work of Baxter and Anjyo [2006]).

We also cannot currently handle textures with long-range forms of order not well expressed by the exemplar, such as textures that flow along a vector field or structured colour variations. And we have found that care must be taken when placing motifs in the exemplar near (or across) the boundary of the input region. It is possible to misjudge the effect of any "padding" between the outermost motifs and the boundary; too much padding will cause gaps in the output. Finally, in order to make our algorithm more useful in a cartographic context, it would be helpful to adjust arrangement synthesis to respect boundaries of map regions, internal curves, or other labels and markings.

5 Conclusion

We have presented a simple example-based geometric texture synthesis algorithm, inspired by the results of a psychophysical study of human perception of geometric arrangements [AlMeraj et al. 2011]. Our approach is based on suppressing repetition artifacts in regularly spaced copies of an exemplar. We demonstrate our algorithm with exemplars derived from standardized textures from geological maps. These textures are typically stochastic. It may also be possible to apply this technique for regular or near-regular textures. But for such cases, a simpler approach based on straightforward tiling is probably adequate.

Our algorithm is reliable enough to be useful in real-world contexts, and produces acceptable results with no manual intervention, though several parameters can easily be manipulated. We hope that it may be useful in a practical illustration context, and that it will provide an algorithm and source of results that can be used in future empirical studies of geometric texture synthesis.

Acknowledgments

We would like to thank the creators of previous synthesis algorithms for their contributions to this work. Without their efforts, we would not have been able to establish a basis of comparison for our algorithm. This research is supported by a doctoral scholarship from Kuwait University and Adobe.

References

ALMERAJ, Z., KAPLAN, C. S., ASENTE, P., AND LANK, E. 2011. Towards ground truth in geometric textures. In *NPAR*, ACM, New York, NY, USA, 17–26.

ALMERAJ, Z., KAPLAN, C. S., AND ASENTE, P. 2013. Towards effective evaluation of geometric texture synthesis algorithms. In *CAe*, ACM, New York, NY, USA.

ALVES DOS PASSOS, V., WALTER, M., AND SOUSA, M. C. 2010. Sample-based synthesis of illustrative patterns. In *Pacific Graphics '10*, IEEE Computer Society, Washington, DC, USA, 109–116.

BARLA, P., BRESLAV, S., THOLLOT, J., SILLION, F. X., AND MARKOSIAN, L. 2006. Stroke pattern analysis and synthesis. *Computer Graphics Forum 25*, 3, 663–671.

BAXTER, W., AND ANJYO, K.-I. 2006. Latent doodle space. *Computer Graphics Forum 25*, 3, 477–485.

DALAL, K., KLEIN, A. W., LIU, Y., AND SMITH, K. 2006. A spectral approach to NPR packing. In *Proceedings of the 4th international symposium on Non-photorealistic animation and rendering*, ACM, New York, NY, USA, NPAR, 71–78.

EFROS, A. A., AND FREEMAN, W. T. 2001. Image quilting for texture synthesis and transfer. In *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, ACM, New York, NY, USA, 341–346.

HURTUT, T., LANDES, P.-E., THOLLOT, J., GOUSSEAU, Y., DROUILLHET, R., AND COEURJOLLY, J.-F. 2009. Appearance-guided synthesis of element arrangements by example. In *NPAR*, ACM, New York, NY, USA, 51–60.

IJIRI, T., MĚCH, R., IGARASHI, T., AND MILLER, G. 2008. An example-based procedural system for element arrangement. *Computer Graphics Forum. 27*, 2, 429–436.

JENNY, B., HUTZLER, E., AND HURNI, L. 2010. Point pattern synthesis. *The Cartographic Journal 47*, 3, 257–261.

LEFEBVRE, S., AND HOPPE, H. 2005. Parallel controllable texture synthesis. *ACM Trans. Graph. 24*, 3 (July), 777–786.

MA, C., WEI, L.-Y., AND TONG, X. 2011. Discrete element textures. *ACM Trans. Graph. 30*, 4 (Aug.), 62:1–62:10.

UNITED STATES FEDERAL GEOGRAPHIC DATA COMMITTEE, GEOLOGICAL DATA SUBCOMMITTEE. 2006. *FGDC Digital Cartographic Standard for Geologic Map Symbolization*. United States Geological Survey. Viewable online at `http://pubs.usgs.gov/tm/2006/11A02/FGDCgeostdTM11A2_web_all.pdf`.

WEI, L.-Y., LEFEBVRE, S., KWATRA, V., AND TURK, G. 2009. State of the art in example-based texture synthesis. In *Eurographics, State of the Art Report, EG-STAR*, Eurographics Association.

ZHOU, K., HUANG, X., WANG, X., TONG, Y., DESBRUN, M., GUO, B., AND SHUM, H.-Y. 2006. Mesh quilting for geometric texture synthesis. *ACM Trans. Graph. 25*, 3 (July), 690–697.