Synthesis and evaluation of geometric textures

by

Zainab AlMeraj

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Computer Science

Waterloo, Ontario, Canada, 2013

© Zainab AlMeraj 2013

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Two-dimensional geometric textures are the geometric analogues of raster (pixel-based) textures and consist of planar distributions of discrete shapes with an inherent structure. These textures have many potential applications in art, computer graphics, and cartography. Synthesizing large textures by hand is generally a tedious task. In raster-based synthesis, many algorithms have been developed to limit the amount of manual effort required. These algorithms take in a small example as a reference and produce larger similar textures using a wide range of approaches.

Recently, an increasing number of example-based geometric synthesis algorithms have been proposed. I refer to them in this dissertation as *Geometric Texture Synthesis* (GTS) algorithms. Analogous to their raster-based counterparts, GTS algorithms synthesize arrangements that ought to be judged by human viewers as similar to the example inputs. However, an absence of conventional evaluation procedures in current attempts demands an inquiry into the visual significance of synthesized results.

In this dissertation, I present an investigation into GTS and report on my findings from three projects. I start by offering initial steps towards grounding texture synthesis techniques more firmly with our understanding of visual perception through two psychophysical studies. My observations throughout these studies result in important visual cues used by people when generating and/or comparing similarity of geometric arrangements as well a set of strategies adopted by participants when generating arrangements.

Based on one of the generation strategies devised in these studies I develop a new geometric synthesis algorithm that uses a tile-based approach to generate arrangements. Textures synthesized by this algorithm are comparable to the state of the art in GTS and provide an additional reference in subsequent evaluations.

To conduct effective evaluations of GTS, I start by collecting a set of representative examples, use them to acquire arrangements from multiple sources, and then gather them into a dataset that acts as a standard for the GTS research community. I then utilize this dataset in a second set of psychophysical studies that define an effective methodology for comparing current and future geometric synthesis algorithms.

Acknowledgements

First and foremost, I thank Allah for His countless blessings throughout my life. I am truly fortunate to be where I am today, Alhamdu-Lillah now and always.

This dissertation is based on three published research papers co-authored with my supervisors Craig S. Kaplan and Paul Asente. The inquiry into perceptual reasoning of similarity in GTS (Chapter 3) was presented at NPAR 2011 [3] and both the patch-based geometric texture synthesis algorithm (Chapter 4) and the evaluation methodology (Chapter 5) were presented at Expressive 2013 in CAe and NPAR respectively [1, 2].

It is with great pleasure to be able to finally thank the institutions who supported me and the many wonderful and inspiring people who have contributed their thoughts and time into making this dissertation complete. I owe a special thank you to Kuwait University for the generous financial and academic support they presented me with throughout my studies and for securing a faculty position for me in the Department of Information Science at their new College of Computer Science and Engineering upon my return. I would also like to thank Adobe for their kind support during our collaboration.

My gratitude and biggest thank you goes to my supervisors Craig S. Kaplan and Paul Asente. The contributions in this dissertation would not have been possible without their guidance, patience and long hours of insightful discussions whether through video conferencing or email. From them I have learnt to do effective research and how to tackle the many problems that come with it. Thank you both for everything.

Many thanks also to my committee members Steve Mann, Ed Lank, Art Green and David Mould for their time spent reading my dissertation and their feedback which has helped improve the presentation of this work. An extra thank you is in order for Ed Lank for his advice, support and guidance during the design phase of my psychophysical user studies and for his tips on analyzing the results; to Steve Mann for pointing out trouble with some figures before my defence and further reviewing changes; and Bill Cowan for the insightful discussions we had on experiments, vision and psychology.

I thank the anonymous reviewers and editors of the conferences I have submitted my work to over the years, which include NPAR, CAe, GI, and Eurographics. Their comments and suggestions did help to improve my papers, posters and ultimately this dissertation.

I would also like to thank Partick Lee for developing some of the interfaces used in my studies, Elodie Fourquet for her comments on some of my paper drafts over the years, and Lesley Northam; despite a recent medical issue she was kind enough to make me a 3D scene to illustrate a concept for my future work.

My appreciation and gratitude go to Thomas Hurtut, Vladimir Alves dos Passos and Chongyang Ma for being so generous in offering their time to generate the textures used in my evaluations. The evaluation in Chapter 5 would have not been possible without them. Another special thanks goes to my undergraduate research assistant Volodymyr Lyubinets for his collaborated efforts on a 2D shape synthesis project and for being such a pleasant person to work with. Cengiz Öztireli spent a lot of his time modifying his own scripts to help me in analyzing some quantitative measures, to him I also express my appreciation.

It is only fair to want to thank all the participants in my use studies who sacrificed their time to help me gather the data I needed to complete this research. At the same time, I would like to thank our UWaterloo Statistics consultant Erin Harvey for her time and insight into analyzing some of my data and the authors of the various software I have used to complete this work such as MATLAB, Adobe Illustrator, GIMP, R package, Netbeans, Xcode, TextEdit, various software libraries and the applications that came with my Mac.

I am also grateful for the publicly available geological dataset by the US Geological Survey (USGS) Digital Cartographic Standard for Geologic Map Symbolization [126] from which I extracted some of my texture exemplars.

My gratitude to Tobias Isenberg and Brian Wyvill for writing my reference letters for Grace Hopper among other events, suggesting opportunities for collaboration which I sometimes turned down, and for checking in with me at conferences over the years.

A bundle of thanks goes to Robin Cohen for her encouragement and guidance since my first year at UWaterloo. Her pleasant nature and lovely smile meant so much. I acknowledge Jeff Orchard for letting me sit through his Medical Imaging class. His passion for Fourier transforms and human brains did nearly brush off on me!

I would like to extend my appreciation to Wendy Rush, Margret Towell, Paula Roser, and Helen Jardine for being so patient, skilled and efficient with handling all of my graduate paperwork and needs. Your jobs are very demanding and I hope you earn the recognition you deserve always. I thank Sandy Graham for the opportunity I was given to observe and be part of her introduction to Computer Science camp for high school women. My appreciation goes also the many instructors in the Centre of Teaching Excellence for their guidance in improving my teaching skills leading up to a Certificate for University Teaching. The program was great and effectively rekindled my passion for teaching.

I am grateful to have met and spent time with the many computer graphics lab members who have come and left CGL throughout the years I spent there, I wish them all success in their studies and future aspirations. I also value the time I spent with the many people I met and kept in touch with in Ontario during my stay, the kind words of support from my friends back in Kuwait, the kindness of my sisters and brothers, and the support and warm wishes of my grandparents June and Phillip Barlow.

I conclude with a few personal acknowledgements. A great big thank you goes to my daughter and best friend Batool Hajeyah. Her loving and caring nature always came through more evidently in the hardest of times for me. I thank her for all her love and the extra hugs before I left for long hours at the university. I hope that my accomplishments show Batool and other smart young women like her that they can achieve all their dreams and goals, from careers to families, with some perseverance, courage and a lot of tawakul. Another big thank you goes to my son Haidar Hajeyah. His smiles and caring thoughts always make me feel special and his love for life and learning continues to inspire me in many ways.

My parents Faisal AlMeraj and Halima Meraj were the sole supporters of my choice to pursue graduate studies. Over the years, their love, care, patience, guidance, encouragement, tolerance, prayers and tremendous support have enriched my life so much that I can not thank them enough. I am sincerely grateful to them both for having the utmost confidence in me and my pursuits and for spending so much time with my children and I far away from home. I can only pray that I may be there for them like they have been for me.

Dedication

To my parents Faisal and Halima,
and my children Batool and Haidar.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Texture synthesis refers to a class of algorithms for generating large graphical textures from little or no input. Recent texture synthesis algorithms are example-based: they begin with a small, user-provided sample called an exemplar, and generate a much larger output texture with similar visual characteristics. The exemplars can be either raster images or geometric (vector-based).

Example-based texture synthesis in general involves acquiring the specific style from an exemplar and reproducing it in a larger synthesized one. Increasing interest in geometric texture synthesis has led to a large variety of approaches, all of which claim to produce outputs that are visually similar to their corresponding exemplars [14, 54, 91]. However, these claims are usually made informally, making it difficult to judge whether any of these algorithms performs better than any other, or whether the word "better" can be given a rigorous meaning in this context. "Visually similar" or "similar" are also used in these judgements, but similarity is itself difficult to pin down. An effective measure of similarity should be algorithmically tractable, while also conforming to human perceived judgments.

Evaluation in computer graphics, non-photorealistic rendering (NPR), visualization, and human-computer interaction (HCI) is essential for emphasizing suitability of a newly proposed algorithm. In the NPR community [49, 59] there is a growing consensus that more careful and objective means of evaluating new algorithms are needed to judge the quality of results. Proposing evaluations should effectively will help us understand the success, usefulness, effectiveness, and possible application domains of computer generated results.

The recent growth of interest in synthesizing geometric (vector-based) textures inspired me to study this smaller subset of vector-based texture synthesis methods, which I introduce later as *Geometric Texture Synthesis*, to find ways to address the lack of rigorous evaluation standards in NPR and simpler means of synthesis.

Accordingly, this dissertation offers an investigation into similarity and perception of geometric textures through conducting various perceptual studies. The goals are (1) to increase our understanding of texture perception, (2) gain insights into the meaning of similarity, (3) develop new perceptual-based algorithms for texture synthesis and (4) devise effective methods to evaluate this similarity. My hope is that the various findings provide valuable insight into texture perception that may some day help formulate a formal operational definition of similarity and encourage a wider range of investigations in other areas of NPR and graphics that also lack evaluation standards.

In Section 1.1 I describe common terminologies and their context within this work. In the remainder of this chapter I introduce Geometric Texture Synthesis (GTS) and discuss challenges in this area (Section 1.2) and present an overview of the contributions in this dissertation (Section 1.3).

1.1 Terms and Definitions

1.1.1 Texture and texture arrangements

A texture can be described using a number of different properties like uniformity, density, regularity, direction and frequency. These properties differ according to the underlying *pattern* within a texture as evident in Figure 1.1. The woven basket texture (a) for example exhibits repeated instances of overlapping material and an indication of direction while the pebbles texture (d) exhibits variations in density and frequency of different sizes of pebbles. A *pattern* here is then described as the distinctive style that captures a certain characteristic of the texture, like its regularity.

In the texture spectrum shown in Figure 1.2, Lin et al. [85] classify four groups of textures according to their regularity. The images include stochastic textures like fur, wood grain, metal and sand; structured textures like leopard skin and stone walls; and more regular textures like wallpaper/woven patterns and brick walls.

It is relatively easy to identify textures in our surrounding environment, however researchers have yet to agree on one definition for texture. In an attempt to summarize existing works, Coggins [23] compiled a list of texture definitions from computer vision literature that reveals large differences across the area. These definitions were created according to

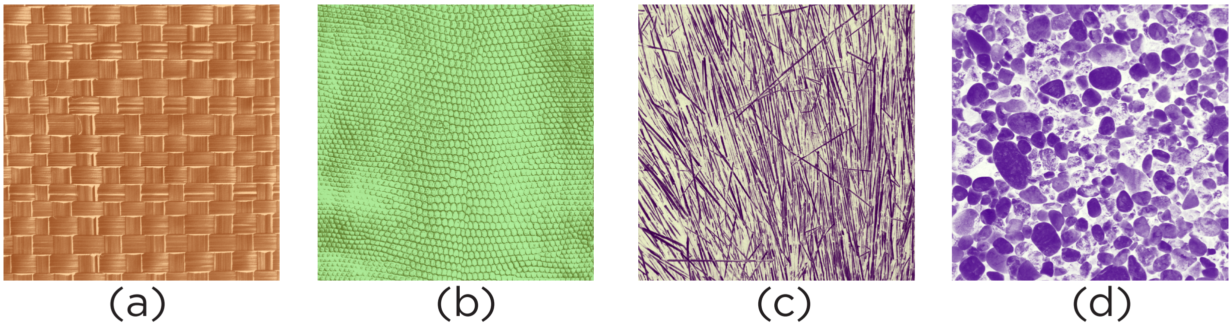


Figure 1.1: *Natural textures taken from the new coloured Brodatz database [111].*

different perceptual findings, or to suit the application areas they were used in. For instance, Haralack defines texture as an "organized area phenomenon" [46] which is accepted more widely as a structured approach to how we actually see textures. Meanwhile, Bela Julesz [63] describes texture mathematically using set of features and statistics.

In an attempt to compensate for the ambiguity involved in describing texture, Tuceryan and Jain [120] give a short overview of various two-dimensional texture properties that are used in the realm of texture analysis. They also summarize three methods of texture analysis that are adopted according to the texture content which they call statistical, model-based, and geometrical. Statistical and model-based methods describe textures using the spatial distributions of pixel intensities or local neighbouring pixels respectively, while geometrical methods define texture as composed of a set of texture objects and placement rules.

The textures investigated in this dissertation can be described in a similar manner to geometrical methods. I call these textures **geometric texture arrangements**. An *arrangement* is a special class of texture that consists of multiple instances of small, discrete, well-defined objects called *motifs*. The motifs need not be in any specific format; they can differ in their regularity across the texture spectrum; and they can be represented in two or three dimensions. Intuitively, arrangements can be decomposed into individual motifs and point locations that give us a general description of the texture, as in Figure 1.3. It is also possible that some textures are created by repeating the same small number of motifs, while others are created using motifs that are of unique shapes.

In the rest of this dissertation I investigate the perception of geometric arrangements where motifs do not overlap and are distributed irregularly on the 2D plane. These include textures that lie in the middle of the spectrum in Figure 1.2. I also show that with a better understanding of perceived "similarity" between irregular geometric arrangements we can learn more about the nature of textures. This could ultimately lead to more concise

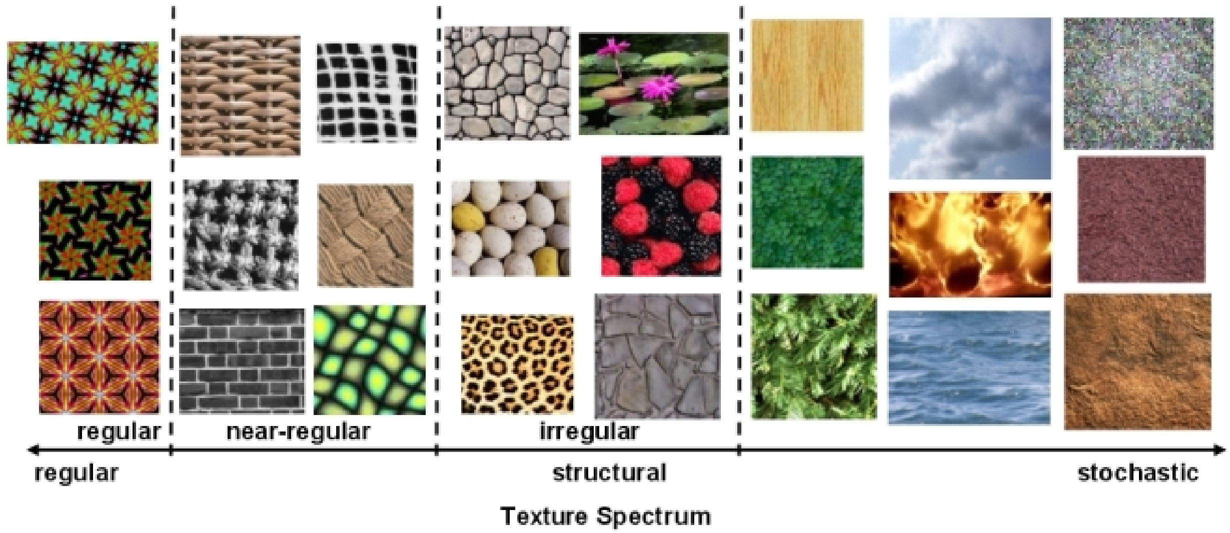


Figure 1.2: *The texture spectrum. © 2006, Lin et al. [85], used with permission.*

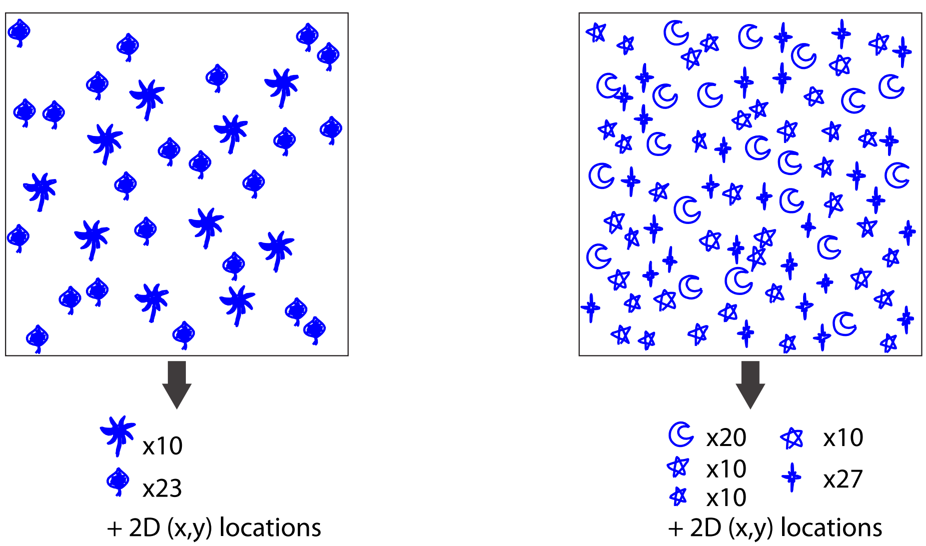


Figure 1.3: *Examples of geometric texture arrangements. © 2009, Hurtut et al. [54], used with permission.*

definitions of texture in the future.

1.1.2 Example-based texture synthesis

The creation of attractive textures has long been of interest to researchers in art and design. Computer graphics researchers have also looked closely at developing textures through algorithmic means. The general process starts by capturing properties of textures, synthesizing larger, similar ones, and then using them in application areas like texture mapping, image enhancement and completion, and game and film production.

Existing texture synthesis techniques have targeted and successfully reproduced a large number of (structurally) regular, near-regular and irregular textures using various algorithmic processes.

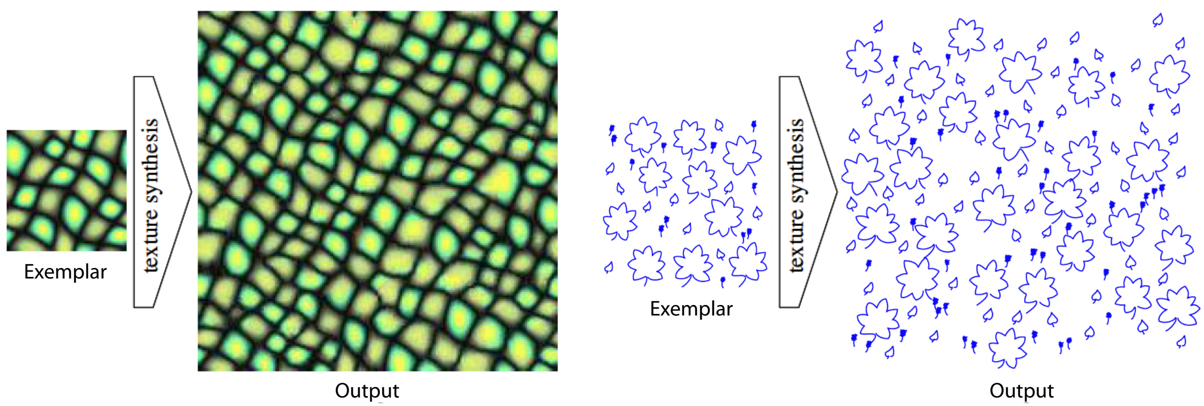


Figure 1.4: *Example-based synthesis samples from Wei et al. [128] and Hurtut et al. [54].* *© 2009. Wei et al. [128], and Hurtut et al. [54], used with permission.*

Example-based synthesis is a specific area within texture synthesis that deals with automatically constructing larger extended images from smaller *exemplar* ones (see Figure 1.4). In graphics, and particularly NPR, automatic example-based pattern synthesis has been able to mimic the appearance of different texture styles. Wei et al. [128] offer a thorough description of relevant literature up to 2009. A more detailed analysis of algorithms relevant to this dissertation will be presented in Chapter 2.

In general, graphics researchers are accustomed to synthesized textures using pixel based methods like the one on the left of Figure 1.4. But a recent movement targeting geometric texture arrangements led to a new line of algorithms that use discrete vector-based primitives, similar to the example on the right of Figure 1.4, to generate larger visually similar arrangements. I use the phrase *Geometric Texture Synthesis (GTS)* to refer to any algorithm that constructs such large arrangements by example.

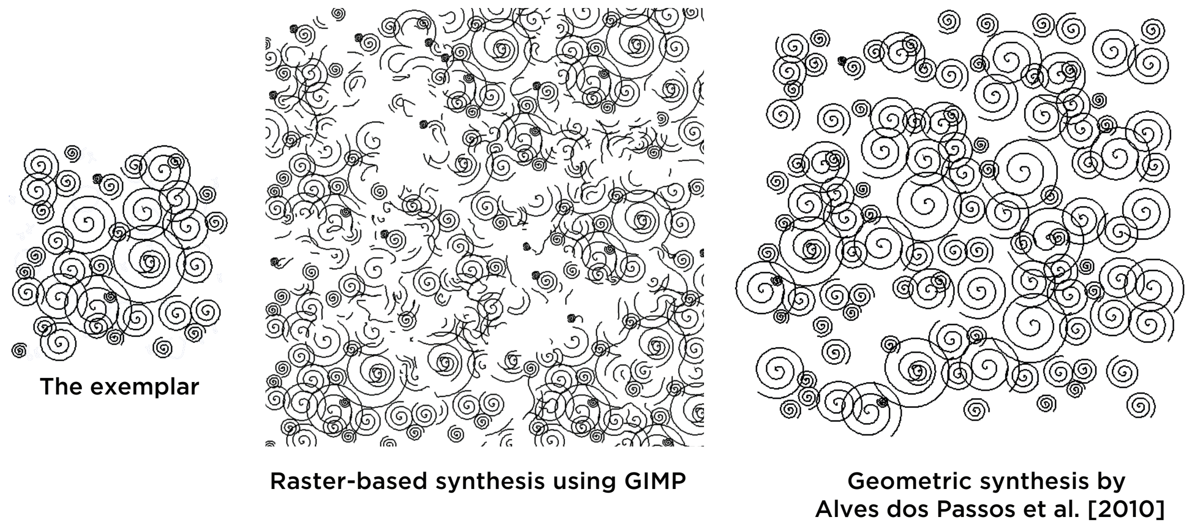


Figure 1.5: *A comparison between a raster-based and Geometric Texture Synthesis algorithm for geometric motifs. © 2010, Alves dos Passos et al. [5], used with permission.*

1.2 Geometric Texture Synthesis (GTS)

Geometric Texture Synthesis (GTS) is the discrete (vector-based) counterpart of continuous texture synthesis, and is a relatively new field that has been driven by innovations in computer graphics, vision and perception [14, 54, 91]. Interest in vector synthesis began with the introduction of the example-based parametric methods by Jodoin et al. [62] and Barla et al. [13]. These methods addressed limitations with raster-based methods when synthesizing whole discrete motifs distributed in one or two dimensions.

Raster synthesis methods have many shortcomings, most notably the inability to maintain much information about individual motifs in an exemplar. During synthesis this causes broken or merged pieces in the motifs. Figure 1.5 illustrates this with an exemplar and its corresponding raster-based and GTS synthesis output.[1] The raster method fails to maintain

[1]The raster image is created using the Re-synthesizer plugin for GIMP: http://www.logarithmic.net/pfh/resynthesizer

Algorithm	Convention
Barla et al. [14]	Non-parametric statistical method
Ijiri et al. [56]	Procedural growth approach
Hurtut et al. [54]	Statistical appearance-based approach
Alves dos Passos et al. [5]	Procedural growth approach
Ma et al. [91]	Energy-based optimization process
My approach (Chapter 4)	Patch-based approach

Table 1.1: *Current conventions used in Geometric Texture Synthesis algorithms.*

the structure of the spirals resulting in a dissimilar larger arrangement. In addition to this limitation, raster-based methods are unable to vary regularity given a distribution of motifs conformed to a regular grid like a brick wall for example. Bricks may end up broken and less regular if patches of pixels are not large enough. A vector alternative resolves most of these problems and assures that motifs are represented appropriately during synthesis and at desired locations.

With a motivation analogous to the one driving raster-based texture synthesis, GTS algorithms try to mimic the underlying structure from a set of motifs in an input exemplar and then utilize the captured information to generate larger visually similar arrangements. The most prominent of the algorithms in the area listed in Table 1.1 have resorted to slightly different conventions for analyzing exemplars and synthesizing the final output. I describe existing GTS algorithms in Chapter 2. Despite the variations in the conventions applied, these methods are deemed informally successful at producing visually similar extensions of input exemplars.

However, GTS research still has many unanswered questions and challenges worthy of deeper investigations. One such challenge is evaluating successful algorithms given a set of synthesized results, as shown in Figure 1.6. The figure shows a source exemplar together with three synthesized arrangements. We can ask, which one is more similar to the exemplar? Which one effectively captures exemplar characteristics? Which of the characteristics do we focus on to make assessments? If there were more synthesis results to compare these with, how can we make a final judgement? And can we learn from these answers to help develop new effective GTS algorithms?

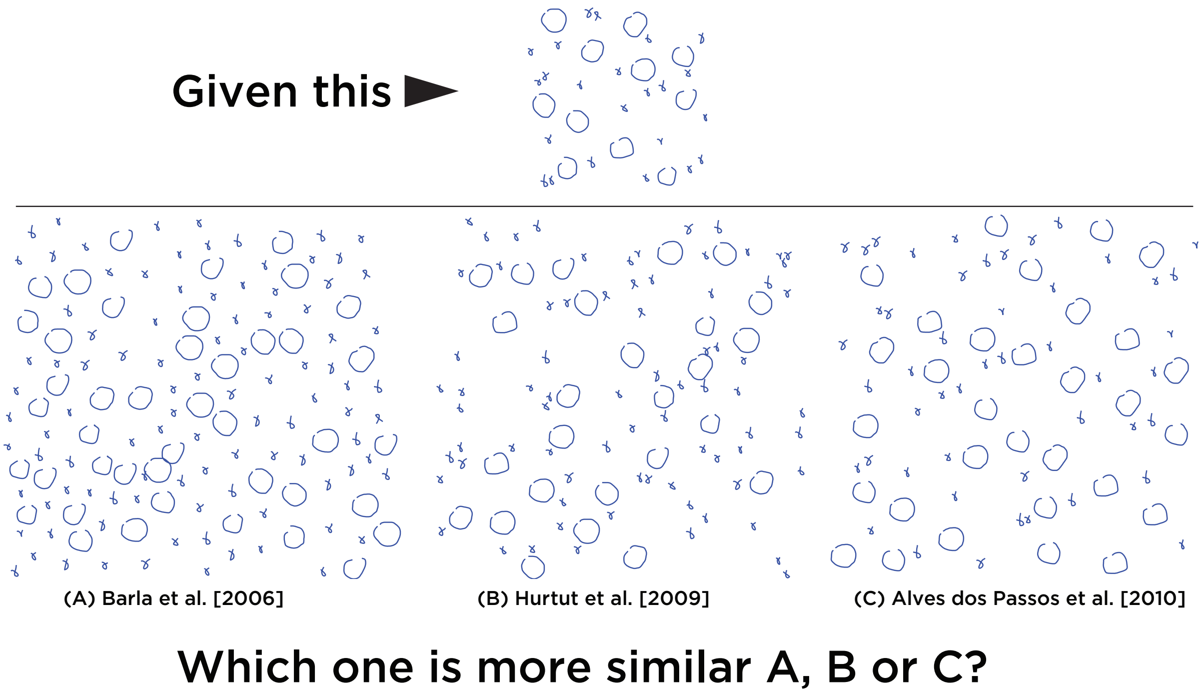


Figure 1.6: *A comparison between three GTS arrangements. © 2009, Hurtut et al. [54] and © 2010, Alves dos Passos et al. [5], used with permission.*

1.2.1 Challenges in GTS

Although there is no "silver bullet" strategy that will provide evaluation across all of NPR, GTS is a domain well suited to more focused evaluations for two reasons. First, GTS operates on relatively simple and abstract arrangements, allowing a person to easily extract meaningful geometric information from the exemplar data. Second, GTS algorithms as with other NPR areas aim to incorporate human visual and perceptual properties, so evaluation should focus on the assessment of individual perceptions of geometric arrangements to determine success of the algorithms.

As with all example-based raster synthesis methods, GTS aims to ensure that synthesized distributions capture the idiosyncracies of the *exemplar* input. The foundations on which existing GTS algorithms were built reside on mathematical and statistical methods that produce visual properties that closely relate to known perceptual processes. These algorithms in turn achieve realistic results (like those shown in Figure 1.6) but lack genuine explanations of the physiological processes needed to achieve it. Hence, there have not been any rigorous evaluation strategies that are attempted on the final results. Instead, for comparison, authors of the now many algorithms show their results alongside previous examples and leave the task of judging which algorithm produces the best results unresolved.

The difficulty that exists in understanding similarity and evaluating GTS results can be attributed to three factors: the immaturity of the area, minimal knowledge of the pertinent perceptual principles that cause GTS algorithms to succeed or fail, and trouble selecting suitable evaluation standards for comparing multiple results simultaneously. In my investigations I probe the nature of similarity in comparisons between texture arrangements and use this knowledge to address the lack of standards within the GTS area.

Decarlo and Stone advocate for better depictions of 2D patterns and ways to effectively generate them without knowing the low-level perceptual processes involved [29]. This ties the success of computer-generated texture arrangements to effective methodologies for evaluating their aesthetics and visual similarity.

Another issue in example-based GTS is how to interpret exemplars, and how do these interpretations affect our judgements when synthesizing and comparing arrangements. We could imagine interpreting regularity here as a continuum from tiling to stochastic placements of motifs. At the tiling extreme we can achieve regularity but with noticeable repetition in larger arrangements. At the stochastic extreme, we have arrangements that do not capture any characteristics found in the exemplar. This leaves the middle ground between these two extremities as an area for more investigation. Somewhere within this space are arrangements that contain an appropriate amount of idiosyncracies perceived to be similar to the irregular exemplar. One challenge is then to identify what constitutes these similar arrangements and develop accurate models of texture perception that can characterize synthesized idiosyncracies, or at least measure whether they have been replicated. A second challenge is to find a suitable algorithm that captures these characteristics effectively and that is robust enough to re-create arrangements with the same or similar quality.

A related challenge involves the lack of cohesion amongst GTS algorithms in representing input exemplars. Each algorithm uses different formats for describing the motifs and their layouts. Some require text files with locations and motif identification tags to place geometry, while others classify motifs using colour and/or size and discern locations accordingly. These cause difficulties in both replicating the algorithms and ensuring similar qualities of the synthesized results. Developing a set of standard prototype exemplars and file formats will offer researchers a solid base for effective comparisons of their algorithms to existing ones.

Finally, there does not exist a general hierarchy that combines both traditional raster-based texture synthesis algorithms with the more recent geometric texturing algorithms. Vector-based synthesis algorithms have not been fully investigated and there still exist multiple problem areas in need of further exploration [58]. This leads me into a parallel investigation to closely review texture synthesis research and find a suitable location within a hierarchy of methods to situate GTS.

1.3 Contributions

The contents of this dissertation evolved from thoughts on proposing a new example-based geometric texture synthesis algorithm. Upon examining the presentation of synthesized results in the area, it became clear that notions such as "better", "more similar", and "more effective" that are being used to determine significance have no credible weight to uphold a final judgement without stronger and more compelling comparisons. Human aesthetic judgements that arise from visual interpretations may differ from person to person. In my research I try to reduce this subjectivity by studying perceptual responses of a concise set of textures to learn more about these visual interpretation and develop ways to better evaluate the results.

The rest of my dissertation is organized around the following primary contributions:

- *A taxonomy of texture synthesis methods.* The first contribution of this dissertation is presented in Chapter 2. In this chapter I suggest a taxonomy of synthesis algorithms intended to situate the GTS field alongside other existing synthesis methods.

- *An inquiry into similarity between geometric texture arrangements.* In Chapter 3, I investigate what we mean by "visually similar" when it comes to comparing geometric texture arrangements. This investigation provides a rich resource for understanding texture arrangements and presents a list of qualitative characteristics that can be used to describe geometric texture arrangements.

- *A patch-based geometric texture synthesis (GTS) algorithm.* The third contribution is presented in Chapter 4 and is designed to enhance GTS synthesis results and is supported by results from Chapter 3. The results from this algorithm are used in my evaluation attempts throughout the remainder of this dissertation.

- *An effective geometric texture arrangement evaluation methodology.* The fourth and fifth contributions of this dissertation are presented in Chapter 5. A newly devised geometric texture arrangement dataset made from multiple sources forms the basis of two integrated psychophysical studies evaluating the effectiveness of example-based GTS algorithms. The results of this investigation feature the potential of unrestricted pile-sorting comparisons and pair-wise comparisons of texture arrangements in evaluating the effectiveness of synthesized results.

- *A quantitative analysis of synthesized geometric texture arrangements*. Finally in Chapter 6, I offer an analysis of some geometric texture arrangements from the new dataset gathered in Chapter 5. The results of the analysis shed light on the nature of GTS arrangements and highlight limitations with raster-based measures.

Chapter 2

Background

In this Chapter I survey the state of the art algorithms in raster-based and geometric texture synthesis (Section 2.1), give some insight into visual perception (Section 2.2) and an overview of evaluation methods in NPR (Section 2.3) to provide context for GTS and its evaluation.

2.1 State of the art in texture synthesis

In this section I take advantage of the general flow within a semi-structured hierarchy developed as a guide through the many advances in the area leading up to Geometric Texture Synthesis. As a first step, I categorize the texture synthesis field into three domains: procedural techniques, example-based techniques, and geometric techniques. This hierarchy is aimed at unifying existing literature and broadening the scope of texture synthesis in both 2D and 3D. It also acts as a framework and can be seen as a multi-level structure composed of a number of domains. Each existing higher-level domain can itself consist of various sub-domains describing different procedures used in texture synthesis. A compact list of the terms and definition used here can be found in Appendix A.

A number of texture synthesis surveys exist in the literature, each proposing a significant framework with various domains. I would like to go beyond previous surveys by considering texture synthesis algorithms within a larger context that incorporates other synthesis methodologies. Before I proceed with a discussion of individual papers within different domains of texture synthesis, I first list three major surveys in the area.

The first is the texture synthesis framework presented by Wei et al. [128]. The authors cover example-based synthesis and classify algorithms into the following groups: raster-based, acceleration, patch-based, and texture optimization. These groups are sufficient for categorizing example-based textures, but the algorithms reviewed are specific to one underlying mathematical process: they all use Markov Random Fields (MRF). These random fields are defined as a set of random/stochastic variables that have a Markov property usually described using an undirected graph with nodes and edges. Further explanations are included in Appendix A. A large part of this chapter analyzes example-based algorithms listing major contributions to the field and focuses specifically on techniques that offer potential applicability for promoting geometric texture synthesis, the main area of this research. I have intentionally avoided elaborating on the wide range of example algorithms previously reviewed by Wei et al. [128]. I leave it up to the reader to return to the appropriate references for further details.

Another example survey involves a procedural noise framework given by Lagae et al. [76]. Their survey gives a very clear overview of the noise function domain. Existing noise functions are grouped into the following categories: lattice gradient noises, explicit noises, and sparse convolution noises. Lagae et al. present a concise definition of noise, subjectively discuss how procedural noise functions are used for modelling, and the methods used to apply them to surfaces. Although parts of this chapter discuss procedural methods, the details found in the Lagae et al. survey are very specific and sidetrack the main intention of clarifying the procedural process related to texture synthesis.

Solid texture synthesis is also related to this research. Solid textures are algorithms capable of representing both the external and internal appearances of 3D objects as realistic 2D textures. Pietroni et al. [107] summarize the different algorithms that synthesize and represent solid texture volumes. Solid textures were initially generated using procedural methods but more recently they have adopted example-based techniques. Besides offering a framework of the solid-based synthesis field, Pietroni et al. give a novel classification for recent example-based methods. They divide them into two groups: boundary dependent, and boundary independent methods. This very idea is considered in Section 7.1, helping promote future research in GTS.

In the remainder of this section I present a general framework that combines both traditional raster-based and geometric texture synthesis algorithms. This framework is introduced through a taxonomy that suggests multiple synthesis domains as shown in Figure 2.1. I based my taxonomy on the higher level procedures used in the texture synthesis process. I present it in four parts using Tables 2.1–2.4. This classification is flexible and suffices as both a review and an evolutionary guide of the synthesis field for future texture researchers. The main intent of this taxonomy is also evident in the classification and will primarily

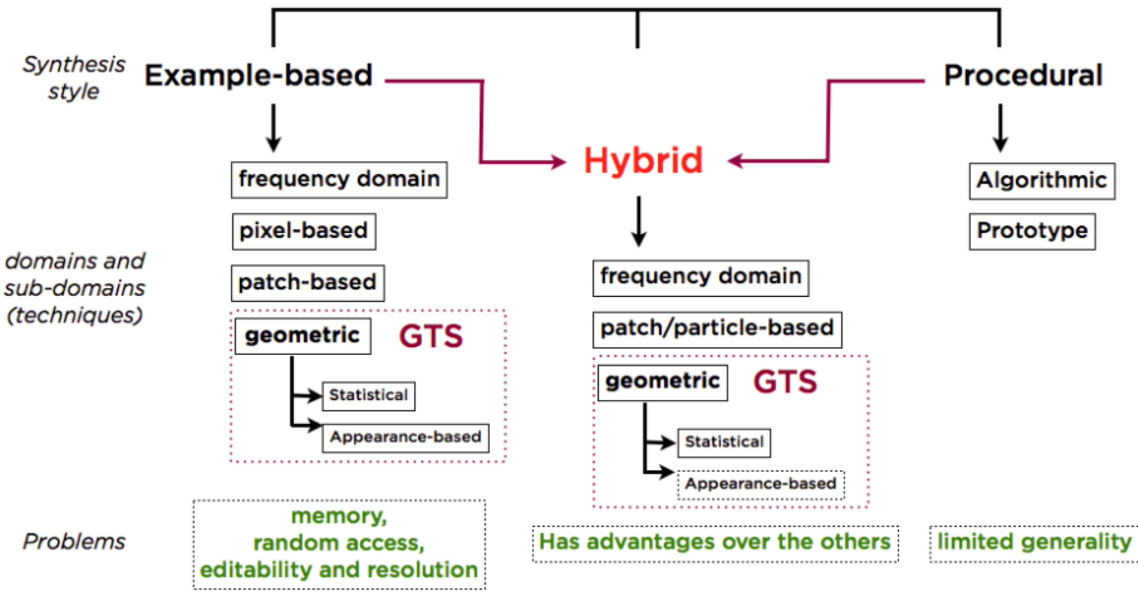


Figure 2.1: *A taxonomy of texture synthesis methods.*

benefit those researching geometric texture synthesis algorithm design.

2.1.1 Procedural texture synthesis

Procedural techniques have become an exciting, vital aspect of creating realistic computer-generated images and animations [35]. Procedural systems rely on a user-supplied explicit model for constructing textures instead of a finished example. The algorithms that produce the user-supplied model in this case are known. Procedural approaches have generally been used in computer graphics for creating plants, buildings and even decorative art designs. Some of the relevant algorithms to this research are listed in Table 2.1.

Lindenmayer's L-systems [86] use procedural modelling to simulate the growth of plants. They were one of the first inspirations for procedural texture synthesis and found their way into computer graphics through Prusinkiewicz and Lindenmayer [109]. These systems produce impressive results, but are difficult to set up and run without significant efforts to write scripts defining local-growth rules. They are also limited in generality of the styles they can produce. Despite this, procedural texture synthesis algorithms have been successful at generating visually pleasing textures. The algorithms here are grouped into algorithmic and prototype-based synthesis methods.

Texture synthesis framework (Part 1)

Procedural Techniques		
Technique/Taxonomy	**Synthesis techniques**	
	Procedural	
	Algorithmic	Prototype
Prusinkiewicz and Lindenmayer et al. [109]	✓	
Perlin [106]	✓	
Lewis [82]	✓	
Lewis [83]	✓	
Lagae et al. [77]	✓	
Worley [135]	✓	
Peachey [105]	✓	
Wong et al. [134]	✓	✓
Měch and Miller [96]	✓	✓
Hsu et al. [53]	✓	✓

Table 2.1: *Taxonomy of 11 procedural techniques for texture synthesis.*

Algorithmic

Algorithmic techniques assume that texture is generated on the fly through a systematic process. Some of the common algorithmic texture synthesis methods are based on different types of noise functions. Noise is a texture primitive generated from pseudorandom distributions in addition to a set of parameters. Procedural synthesis based on noise is produced when noise is combined with mathematical expressions [106]. According to Ebert et al. [35], procedural noise functions have compact storage requirements; they are continuous, and hence not based on discretely sampled data; they are randomly accessible, non-periodic, and may be appropriately parameterized. These advantages are taken into account when deciding the suitability and effectiveness of algorithmic texture synthesis methods over others.

Noise algorithms such as Perlin noise [106] capture procedural descriptions from stochastic textures and use sums of multi-scale noise functions including a combined spectral/histogram-based approach to achieve good results. This controllable, pseudorandom appearance technique is used mainly for realistic synthesis of the natural textures such as the appearance of marble. Some other approaches to generating noise include fractal-based algorithms [82], and sparse convolution noise [77, 83] (described as a procedural noise that offers improved control over the power spectrum). Worley developed a Poisson distribution-based noise generation method that produces cellular patterns [135]. An extension using Fourier synthesis to three dimensional solid textures was devised by Peachey [105] More examples of procedural noise algorithms can be found in the state of the art report on procedural noise functions by Lagae et al. [76]. Other synthesis methods that are not based on noise functions include synthesis by reaction diffusion [121] and halftoning [124].

Prototype

Prototype texture synthesis is a special case of algorithmic synthesis in which specified elements of a texture pattern are taken together with a rule for stamping out copies of those elements to generate larger visually pleasing textures. These textures are formed from a small number of geometric prototypes designed in advance. Procedural modelling systems have been developed in this context for mimicking growth patterns. Wong et al. [134], for instance, introduce the idea of adaptive clip art, which encapsulates rules for creating specific ornamental patterns in enclosed areas using proxies. Proxies are geometric representations of the individual texture elements specified by a user. By first placing ornamental elements algorithmically using proxies of the actual geometry, Wong et al. were able to synthesize a number of texturing styles. The ornament grows incrementally by applying rules from existing motifs into portions of the enclosed space that are not yet populated. Proxies are allocated into these empty spaces according to pre-specified growth model rules.

A significant contribution of this work is that the growth model represents the artist's process in creating aesthetic stylized plant designs. The method ultimately avoids the use of growth models such as L-systems, venturing away from imitating the growth of real florals. But the beautiful arrangements generated using the Wong et al. method involves heavy manual editing as well as artistic expertise to create appropriate geometry for the ornamental elements.

Another form of procedural texturing developed by Měch and Miller [96] addresses the synthesis of larger patches of 2D element arrangements. Inspired by artists, they create an interactive framework called Deco which is available through the Adobe® Flash Pro® software. The can to create complex structures and patterns in real-time. By selecting or creating a set of procedural rules and choosing a set of motifs, Deco synthesizes larger vector patterns incrementally confined within borders of user specified spaces. Adobe® Photoshop ® deals with the raster-based counterparts of this approach.

A final example of a prototype-based approach is pattern brushes found in Adobe® Illustrator® which follow directly in the lineage of the skeletal strokes of Hsu et al. [53]. Their stroke stylization technique makes use of a 2D mathematical deformation model in which deformable images called "brush strokes" can be anchored, scaled, or transformed at the control points. The resulting strokes can be applied in drawings and animation systems. While this algorithm produces excellent results for straight and low-curvature smooth paths, it quickly degenerates as path geometry becomes more complex (severe overlap). It also cannot handle multiple intersecting paths. A large reason for this lies in their vector representations that depends on texture mapping to render many of the artistic stroke effects. More recently, Asente [6] proposed a new geometric algorithm that mitigates the degeneracies in high-curvature regions of paths.

2.1.2 Example-based techniques

Example-based texture synthesis algorithms have been the prime focus in the texture synthesis community for over a decade. As shown in Figure 2.2, these algorithms take an example image, analyze it, and then generate a new texture in a larger area that is visually similar to it. These techniques can be grouped roughly into the following four domains: frequency-based, raster-based, patch-based, as well as the newly recognized geometric (vector) domain. Below are descriptions and a list of techniques for each of these synthesis domains. Where appropriate, the algorithmic techniques adopted are classified according to their style, either parametric (P) or non-parametric (NP) as shown in Table 2.2. Parametric methods provide a compact description of textures. They make use of statistical analysis to characterize an input texture by a set of parameters, and then attempt to synthesize similar textures with similar properties to validate the parametric model. Non-parametric techniques involve the use of iterative algorithms that work with neighbourhood comparisons between reference and target textures.

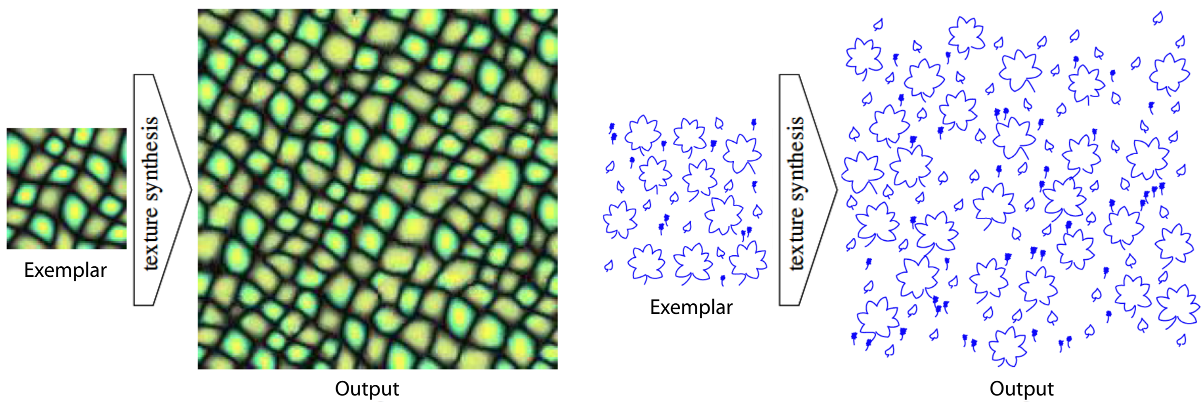


Figure 2.2: *Example-based texture synthesis [128].*

Frequency and statistical feature matching techniques

Examples of these techniques produce limited types of textures inspired by Markov random field models and implemented using parametric texture representations that adhere to human perception. Examples include techniques by Heeger and Bergen [47] and Portilla et al. [108].

The Heeger and Bergen method makes use of both Laplacian and steerable pyramid analysis of a texture input sample to create more of the same texture. They are able to capture a close representation of an input image by iteratively matching histograms through expanding and reducing the steerable pyramid of input and output. Although this technique produces good results, it is limited to synthesizing stochastic homogeneous textures with minimal structure. To overcome this constraint, Portilla et al. offer an improved version that draws on human perception and mimics computations carried out in human vision. This technique is based on capturing nth order statistics including spatial averages and coefficient correlation to produce reasonably better results. To test the perceptual validity

Texture synthesis framework (Part 2)

Example-based Techniques						
Technique/Taxonomy	**Analysis**		**Synthesis techniques**			
	P	**N**	**Example-based**			
			Freq.	Raster-based	Patch-based	Tiling
Heeger and Bergen [47]	✓		✓			
Portilla et al. [108]	✓		✓			
Efros and Leung [37]		✓		✓		
Jeremy S. De Bonet [28]	✓			✓		
Wei and Levoy [129]		✓		✓		
Ashikhmin [7]		✓		✓		
Hertzmann [50]		✓		✓		
Turk [122]		✓		✓		
Ritter et al. [110]		✓		✓		
Lefebvre and Hoppe [80]		✓		✓		
Lefebvre and Hoppe [81]		✓		✓		
Wu and Yu [136]		✓		✓		
Wei [127]		✓		✓		
Wei and Levoy [140]		✓		✓		
Dischler et al. [33]		✓			✓	
Guo et al. [38]	✓				✓	
Efros and Freeman [36]		✓			✓	
Kwatra et al. [73]		✓			✓	
Liu et al. [89]	✓				✓	
Cohen et al. [24]	✓					✓
Liu et al. [88]	✓					✓
Lagae and Dutré [75]	✓					✓
Kwatra et al. [72]		✓				✓
Landes and Soler [79]	✓					✓

Table 2.2: *Taxonomy of 24 example-based techniques for texture synthesis (P - Parametric, N - Non-parametric methods).*

of their model, Portilla et al. run the results through a comprehensive framework designed to verify statistical properties. Their method can capture both stochastic and repeated textures quite well and is useful for texture classification using perceptual principles, but it can not handle shape distributions correctly or synthesize highly structured patterns.

Parametric methods such as the one above can often describe textures using minimal parameters. Each of these parameters allow for texture editing operations making the algorithms compact, and hence more desirable. These example-based methods emphasize the importance of understanding the underlying human perceptual mechanisms of texture. Further understanding of the nature of these mechanisms may result in better synthesis algorithms and possibly enable the synthesis of a larger range of structured textured patterns.

Raster-based techniques

Raster-based techniques are able to capture the local statistics of a texture and regenerate them for a limited class of textures due to the imposed raster grid. Raster-based approaches use neighbourhood comparisons between example and generated textures and are hence all non-parametric. The method by Efros and Leung [37] gives one example that applies Markov random fields to perform the comparison. Jeremy S. De Bonet [28] and Efros and Leung [37] show that nearest-neighbour searches can produce high-quality texture synthesis in a single pass. To do this they use multi-scale and single-scale neighbourhoods, respectively. Wei and Levoy [129] also introduce their own hierarchical approach to the same synthesis process. Ashikhmin [7] modifies the Wei and Levoy method and succeeds at preserving texture coherence in the results.

Further attempts that enhance resulting textures from the above MRF methods include techniques by Hertzmann et al. [50], Turk [122], and Ritter et al. [110]. Extra features for improved matching can be found in Lefebvre and Hoppe [80], who introduce a parallel approach to neighbourhood matching that uses up-sampling as well as their high dimensional descriptive texture space in a subsequent extension [81]. Wu and Yu [136] incorporate curvilinear feature matching and texture warping to overcome noticeable seams in synthesized textures. Other methods include Wei [127], who synthesizes textures from multiple source inputs as well as the one by Wei and Levoy et al. [140] who uses pyramidal-based ordered synthesis to generate similar results.

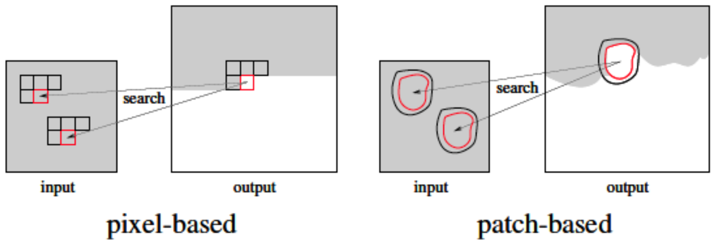


Figure 2.3: *Patch-based synthesis. © 2009, Wei et al. [128], used with permission.*

Patch-based techniques

Patch-based techniques are known to synthesize more structured textures than raster-based techniques. By copying patches of pixels rather than individual pixels, the generated textures are able to preserve local structures found in input textures (see Figure 2.3). All instances of patch-based techniques are example-based and can be either parametric or non-parametric.

In non-parametric approaches, Dischler et al. [33] introduce a unique idea of texture patches used to represent elements in textures. Their algorithm decomposes a bitmap image into small pixel patches representing textural elements called *particles*. Each particle is enclosed in a bounding box, defining it as an independent element, and spatial statistics for the particle placements are gathered using lower order statistics. The algorithm is capable of reproducing large structured and stochastic texture element arrangements, including 3D solid texture, while minimizing the visual artifacts of earlier raster-based techniques. This sampling algorithm is used at synthesis via a seeding procedure and results in fast and easy texturing of arbitrary surfaces. The extent to which the Dischler et al. algorithm relies on user input for visually similar results is unclear and presumably a major concern. Their work is also one of the first to introduce capturing 2D element spatial organizations at the texture analysis level. This is an idea recently revisited by vector-based researchers to assist in refining geometric synthesis techniques.

Guo et al. [38] offer a different texture analysis approach. In their algorithm, element appearance, density, and spatial arrangements are collected and modelled into a Gibbs distribution model. The parameters maximizing this model are estimated by gradient ascent, allowing the overall arrangement to evolve according to a Markov chain process. The statistical tools Guo uses here give an efficient approach to enforcing appearance-based statistics over the output texture. The only downside is that the synthesis process requires many iterations to get visually similar results.

In an extension to their raster-based technique, Efros and Freeman [36] used dynamic programming to find optimal cutting paths through an image, also known as the minimum cost path through the error surface at the texture overlap region. This helps reduce blockiness found at the boundary between texture patches during synthesis. Kwatra et al. [73] further improve on this and stitch pieces of texture found using a graph cut method along optimal seams, resulting in a speed-up in the synthesis process. Using the lattice structure of an input texture Liu et al. [89] are able to synthesize regular and near-regular textures. User input is needed to guide the construction of the lattices. This is essential as it primarily simplifies the efforts needed to define the different texture properties that are later used at synthesis.

Tiling techniques

Although not entirely separate from patch-based methods, "Tiling techniques" have been previously discussed in texture synthesis research as belonging to their own domain. These techniques synthesize textures by copying specially chosen image patches directly from the input texture and stitching them together to form a new synthesized image. Tiling techniques are generally concerned with improving texture synthesis algorithm speeds, as in the work of Cohen et al. [24]. They have been also been effective at synthesizing near-regular textures such as the one by Liu et al. [88] and irregular textures like they one by Lagae et al. [75].

Tiling techniques have the advantage of preserving texture details by keeping the pixel neighbourhoods intact in the synthesized textures but they do not offer any special consideration to a texture's global structure. Although these techniques enhance patch-based methods to produce more compelling results, they are again limited in the texture styles they allow, and do not offer relevant handling methods for any significant statistical information present in the texture samples.

Further advances from pixel and patch-based techniques address texture optimization as an alternative strategy. Kwatra et al. [72] successfully combine all the positive properties found in both pixel and patch-based algorithms. Their synthesis algorithm is implemented based on a greedy raster-based placement followed by optimizing a quadratic energy function on pixel neighbourhoods. This eventually leads to more visually similar outputs but continues to have problems with synthesizing highly structured images.

Alternatively, Landes and Soler [79] propose an unsupervised statistical method that analyzes and re-synthesizes 2D arrangements of shapes. He extracts patch-based descriptors (a vector-based primitive representation) along with region similarity maps to describe locations and region transformation properties found in a raster sample. This content-based description is then used to generate new images with similar patterns maintaining comparable neighbourhood regions. Advantages of this technique include an ability to sort and assemble regions to express original content, including the ability to represent overlapping shapes. Its main limitation lies in the restricted set of texture styles that the algorithm can generate.

2.1.3 Geometric techniques

Earlier in Chapter 1, I defined Geometric Texture Synthesis (GTS) to be the vector-based counterpart of raster synthesis. Various terms have been used to describe what I call GTS and including vector-based stroke synthesis, vector-based texture synthesis, element-based texture synthesis, discrete element texture synthesis and object point distribution synthesis. All of these refer to the process of synthesizing textures using defined 2D vectorized motifs either detected or predefined in a given exemplar. These motifs are each perceived individually making it difficult to apply pixel or patch-based techniques to achieve reasonable output as shown in earlier methods.

Geometric texture synthesis has targeted the synthesis of a variety of patterns. Existing synthesis approaches can be broadly classified into two categories: statistical and appearance-based techniques as illustrated in Table 2.3.

Statistical techniques

Statistical approaches attempt to preserve statistical properties found in sample textures. Initial synthesis using this approach was geared towards applying parametric methods for manipulating vector elements. Jodoin et al. [62] generate one dimensional hatching patterns from an input sample texture using MRF's. This allows the algorithm to propagate gathered local distances between the elements. Extending the synthesis to generating 2D arrangements of lines and points, Barla et al. [13] enforced specific parameter statistics on the generated elements in corrective steps.

Barla et al. [14] adopted a non-parametric method on the input texture. Their method first detect motifs from a given sample image, and group them into categories based on visual similarity. In the analysis step, Barla et al. model the arrangement of elements using a Delaunay triangulation, capturing spatial element arrangement details. This information is then used to modify a set of 2D seed points distributed uniformly on a plane. The density of these point sets is user determined. Finally, vector primitives are pasted onto the distribution using local neighbourhood matching. The Barla method is one of the first

Texture synthesis framework (Part 3)

Geometric Techniques							
Technique/Taxonomy	**Analysis**		**Synthesis techniques**				
	P	**N**	**Example-based**				
			Freq.	Raster-based	Patch-based	Tiling	Geom.
Statistical:							
Jodoin et al. [62]	✓						✓
Barla et al. [13]	✓						✓
Barla et al. [14]		✓					✓
Jenny [61]	✓						✓
Öztireli and Gross [102] (2)		✓					✓
Landes et al. [78]	✓		✓				✓
Winkenbach and Salesin [133]	✓						✓
Deussen et al. [31]	✓						✓
Salisbury et al. [112]	✓						✓
Freeman et al. [40]	✓						✓
AlMeraj et al. [4]	✓						✓
Kalnins et al. [67]	✓						✓
Hertzmann et al. [51]	✓						✓
Brunn et al. [22]		✓					✓
Appearance-based:							
Hurtut et al. [54]		✓					✓
Liu et al. [87]		✓					✓
Ma et al. [90, 91]		✓			•		✓
My new contribution presented in Chapter 4.		✓			✓		✓

Table 2.3: *Taxonomy of 18 geometric techniques for texture synthesis. ✓ stands for a technique being used, • stands for a weak classification.*

attempts to leap from raster-based to vector-based texture synthesis. The prime contribution is their novel approach of identifying elements and collecting local neighbourhood measurements. A primary limitation of their method is that it only holds for evenly-spaced distributions due to a Lloyd relaxation performed on the initial element placements. In addition, the notion of using perception when matching elements is not obvious. Although Barla et al. gather perceptual information from the input and take it into account when placing elements, the global visual appearance of the textures is not accounted for in the final arrangements.

The GTS inspired cartographic work by Jenny et al. [61] starts with a uniform grid of points and adds to them random displacements to impose irregularity. The algorithm then eliminates overlapping motifs and offers a solution for cropped motifs at boundaries. Once these conflicts are resolved, points are replaced with geometric symbols. Even though the Jenny et al. method does not attempt to understand spatial distributions of the elements, the results are promising and address main concerns with cartographic software.

More recently a statistical approach by Öztireli and Gross [102] use a second-order statistic called the Pair Correlation Function (PCF) as a guide to achieving global similarity. They show that pair-wise correlation offer a compact representation of point set characteristics. Given one or more exemplar inputs, one of their algorithms starts off with a random point set and achieves a similar distribution through minimizing the differences between the PFC of the synthesized distribution and the PCF of the input. This method generates pleasing results in both 2D and 3D; however, it is limited to irregular arrangement patterns.

Most recently Landes et al. [78] divide their statistically-based synthesis process into two parts. The first part includes a detailed analysis of pairwise shape interaction that capture the distances and orientations between their geometric representations. The synthesis step starts with a complex statistical model of point processes that represents the configuration of the input exemplar geometries. It generates new textures by iteratively matching the probability densities of the output and the exemplar model using Monte-Carlo Markov Chain (MCMC) simulations. Based on their results, the algorithm out performs state of the art synthesis algorithms in its ability to handle anisotropic and regular arrangements in both 2D and 3D but at the cost of a complex stochastic model.

Appearance-based techniques

All previously discussed element-aware texture synthesis techniques only consider element to element spatial distributions to determine how to place newly synthesized elements. Appearance-based approaches enforce observed similarities during synthesis, achieving what seem to be perceptually similar distributions over the 2D plane or even 3D. This requires proper representations of the different elements that largely contribute to the synthesized texture's success.

Research in raster-based stroke and tone rendering has long studied style transfer for better and more accurate representations of the input samples. For example, Winkenbach and Salesin [133], Deussen et al. [31], Salisbury et al. [112], Freeman et al. [40] and AlMeraj et al. [4] all aim to capture and reproduce realistic styles to maintain certain appearances in the generated images. In some cases these styles can be captured using offsets to a line's base path, similar to the WISYWYG NPR system presented by Kalnins et al. [67]. Other style-capturing strategies include MRF approaches, similar to the one given by Hertzmann et al. [51], and the wavelet curve decomposition method by Brunn et al. [22].

For geometric synthesis, Hurtut et al. [54] offer a first attempt at an appearance-based approach to texture synthesis. They developed a non-parametric algorithm for element arrangement synthesis using a statistical learning method. Drawing from gestalt grouping theory and human vision, Hurtut et al. extract and categorize geometric elements based on their appearance in the sample image. They then adopt a multi-type point process to synthesize new arrangements that respect collected spatial statistics. The resulting arrangements are somewhat different from the original texture but maintain similar visual measures. Armed with this statistical analysis model, Hurtut et al. produce compelling results that are pleasing, thus making perceptual spatial measurements an option for further investigation. However, some limitations of this technique include element categorization based on motif areas, difficulty synthesizing regular patterns dues to the point process and imperfection synthesizing overlapping hatching style textures.

A subsequent approach by Liu et al. [87] combined both the local neighbourhood analysis and setup method of Barla et al. [14], and the global features characteristics used by Hurtut et al. into a multi-stage optimization algorithm that optimizes all local, global and elemental distances to a reference input. Given an example arrangement they extract the above information and iteratively substitute elements with other elements that reduce the optimization error until convergence is reached. The main advantage of the Liu et al. method is the ability to synthesize a variety of texture distributions that range between uniform and non-uniform styles. Despite the higher cost of the optimization technique, this method still does not address problems like synthesizing longer overlapping motifs like hatching lines.

The final method by Ma et al. [91] proposes to capture both appearance and distribution properties of an input sample using an optimization solver. They first place different size patches of the sample in the new arrangement then iterate through all the elements checking their similarity and distance measures of neighbourhoods found in the sample exemplar using a physics simulator. A prominent advantage of the algorithm is its ability to ensure that the output conforms to predefined boundaries and orientations through repeated iteration until the error converges to a value close to zero. This happens at the cost of a complex energy-based optimization process. This algorithm was extended recently to include the synthesis of spatial and temporal properties found in geometric animated 2D or 3D exemplars [90].

A new contribution to this list of appearance-based techniques is the one I propose in Chapter 4. The method gives an alternative approach to synthesis by tiling the synthesis space with overlapping copies of the exemplar and then culling individual motifs based on overlaps and the enforcement of minimum distances. This simple approach yields pleasing results that are competitive with current state of the art GTS algorithms.

2.1.4 Hybrid techniques

Other texture synthesis techniques are harder to classify. While procedural texturing techniques are limited in their generality (currently a significant open problem), example-based texture synthesis techniques have been more successful at offering multiple ways to synthesize a variety of texture styles. Compared to procedural methods, example-based synthesis also has several shortcomings, such as the need for increased storage, random access support, improved editability/controllability, and difficulty with reproducing textures at higher resolutions. Wei et al. [128] show that recent advances in texture synthesis are geared towards closing the gap between both example-based and procedural methods to overcome these disadvantages.

Hybrid synthesis methods use both user supplied example inputs and procedural algorithms to generate visually similar and pleasing textures in 2D or 3D. The resulting textures are more compact, resemble a large range of styles, and are available at high resolutions. These algorithms also allow for easy user accessibility and interactive editing during synthesis. Current Hybrid methods listed in Table 2.4 can be grouped loosely into three categories: frequency-based, particle-based, and geometric techniques.

Frequency domain methods

Dischler and Ghazanfarpour [32] introduce one of the first example-based procedural algorithms. Although they target stochastic bump textures and hypertextures, Dischler et al. present a two step automatic procedural generation method for synthesizing visually similar results. They first analyze an example 1D texture model (called a *profile*) using a frequency-space transform and then perform a statistical equalization step to capture the sample spatial histogram. The synthesis algorithm extends the profile to 2D or 3D space easily and efficiently generates many variations of stochastic textures. The generated textures are defined as a sum of elementary random functions similar to those captured from the sample input textures. Using a procedural approach, Dischler et al. allow for direct user interaction and computation of textures anywhere in the Euclidean plane. Their method is limited to gathering texture information from raster-based images. The noise functions utilized can not be extended into vector space.

Gilet and Dischler [43] recently proposed an automatic synthesis process that procedurally generates purely anisotropic textures. Rather than using traditional procedural approaches to achieve descriptions of stochastic textures, they use Gabor noise functions to increase controllability of the spectral domain. While results are interestingly good, the algorithm is not easily extendable to include larger primitives or structured patterns.

Patch/Voxel/Particle-based methods

A first attempt at formally classifying hybrid methods was introduced by Gilet et al. [44]. They developed a new texture model that is mid-way between procedural textures and example-based texture synthesis. Gilet et al. extend the idea of using elemental *particles* from Dischler et al. [33] to procedurally synthesize particle distributions at any specified location in the plane. A fragment shader program built to support such synthesis uses texture maps consisting of the example texture particles, including a synthesized background texture, to generate non-periodic patterns on surfaces. This hardware-based approach by Gilet et al. allows for real-time user interaction during synthesis and supports high resolution synthesis with minimal memory cost. Limitations exist when generating regular structured textures because of the shader's per-fragment implementation.

The extension of patch-based methods to 3D volumetric synthesis has also been successful. Bhat et al. [19], for example, propose a technique for generating geometric texture on the surface of a model by analyzing the geometric information in a sample model. Their algorithm extends the raster-based analogies method by Hertzmann et al. [50] into 3D voxels, allowing extreme geometric modifications to the 3D model surface. Lagae et al. [74] synthesize output geometry that differs in its local appearance but that is perceived by viewers to be similar to the input sample. By imposing a MRF model on the input geometry, Lagae et al. reproduce similar geometry on surfaces using the stochastic process determined from the input texture. Although results are pleasing and offer interesting geometric variations from a single example input, the main drawback of this method is the computationally expensive MRF method used at synthesis.

Texture synthesis framework (Part 4)

Hybrid Techniques							
Technique/Taxonomy	**Analysis**		**Synthesis techniques**				
	P	**N**	**Procedural**		**Hybrid**		
			Alg.	Pro.	Freq.	Particle	Geom.
Frequency-based:							
Dischler and Ghazanfarpour [32]	✓		✓		✓		
Gilet and Dischler [43]		✓	✓		✓		
Particle-based:							
Gilet et al. [44]		✓	✓			✓	
Bhat et al. [19]		✓	✓			✓	
Lagae et al. [74]		✓	✓			✓	
Öztireli and Gross [102] (1)		✓	✓			✓	
Geometric:							
Statistical:							
Ijiri et al. [56]		✓		✓			✓
Jagnow et al. [60]	✓			✓			•
Appearance-based:							
Alves dos Passos et al. [5]		✓		✓			✓

Table 2.4: *Taxonomy of 10 hybrid techniques for texture synthesis.* ✓ *stands for a technique being used,* • *stands for a weak classification.*

Using a generalization of a dart throwing routine Öztireli and Gross [102] propose a procedural algorithm for geometric arrangement synthesis (the first of two algorithms). Their method procedurally generates random points within a boundary guided by a Pair Correlation Function (PCF) and rejects those that do not satisfy minimum distances to other points in the exemplar set. In addition to synthesizing in 2D, the algorithm generates visually pleasing point distributions in 3D. The final synthesized arrangements are interesting and resemble characteristics found in the exemplars.

To support their claim of similarity, Öztireli and Gross accompany synthesized arrangements with charts visualizing the synthesized and target arrangements, PCF curves and irregularity measures. This is the first attempt in the area to reduce the subjective bias involved when presenting synthesized results and comparing them to the inputs giving it an advantage over the others. One limitation however with using pair correlations is present when synthesizing regular arrangements. This is due to the stochastic nature of the algorithm. Another concern is that even with a quantifiable statistical measure, true similarity may only exist through the eyes of the viewer (based on multiple factors like aesthetics and pleasingness). This problem is evident when it comes to evaluating GTS. Some proposed geometric synthesis algorithms implicitly address this by choosing statistical measures that account for how we perceive textures but these measures have not yet been proven effective for GTS. Throughout this dissertation I emphasize the importance of perceptual studies in understanding similarity and discuss to what degree quantitative measures of similarity in GTS are informative to the nature of these arrangements.

Geometric methods

Geometric methods address the synthesis of 2D/3D vector-based textures given an input sample texture composed of an arrangement of some number of vector primitives. Existing hybrid geometric synthesis algorithms can be divided into statistical and appearance-based techniques.

Statistical techniques

In the realm of vector-based synthesis, there have been studies that tackle the simulation of the interactions between motifs of a given sample texture using statistical means. Ijiri et al. [56] approach the problem of arrangement analysis and synthesis using local neighbourhood matching. They are able to synthesize a texture given an example arrangement by analyzing a sample pattern and procedurally generating similar arrangements for larger contained areas. Their work draws from both example-based and procedural methods, combining both the neighbourhood analysis method of Barla et al. [14], and the rule-based heuristics technique of Wong et al. [134] to ensure that the ongoing incremental generations are similar to the initial texture structure. Ijiri et al. effectively integrate these two approaches and subsequently produce pleasing arrangements of synthesized elements.

They start by placing a single seed in a predefined interior region. Using information gathered from an analysis step on the sample arrangement, the iterative algorithm checks the current seed neighbourhood and locates a reference element in a sample reference arrangement with the most similar neighbourhood. Once found, the seed is replaced with the new element and new seeds representing the local neighbourhood of the matched element are placed in the appropriate locations. Finally, the seeds are connected to the arrangement using a Delaunay triangulation, and a global relaxation process is applied after each iteration to maintain smoothness of the pattern. This process is applied repeatedly until all spaces in the synthesis region are filled. Although the generated textures fail to preserve the perceptual distribution of the original elements visual attributes, the context in which it is used is mostly user determined. One limitation however is that the synthesis process is highly dependent on the user to moderately refine the overall synthesized element arrangement to remain visually similar to the original sample.

Geometric texture synthesis in 3D has also produced intriguing results. Despite the fact that they target the solid texture synthesis community, the statistical approach by Jagnow et al. [60] extracts quantitative information to synthesize three-dimensional material (composed of geometric elements) from measurements made on two-dimensional planar sections of the material. Their main contribution to the solid synthesis field is their stereological approach which provides a systematic basis for predicting certain material structure along with a constrained set of assumptions.

Appearance-based techniques

Appearance-based techniques for example-based geometric methods achieve impressive results and hold much promise for the future of geometric synthesis. Attempting an appearance based approach procedurally would not only offer all the advantages of procedural systems, but also eliminate the difficult preparations and scripting required by pure procedural techniques. In essence, this will address the concern put forward by Wei et al. [128] and strive towards further closing the gap between procedural and example-based methods.

A prime example implementing such a technique is that by Alves dos Passos et al. [5]. They use a procedural growth approach to enhance geometric synthesis results for a wide variety of texture styles (regular, near-regular, irregular, and stochastic). The algorithm starts by placing seeds in different cells of a uniform square grid, then procedurally expands within each cell substituting the seeds with elements from the sample input that have the most similar neighbourhood until the synthesized texture arrangement is complete.

Each similar local neighbourhood is determined using a Euclidean dissimilarity metric between the neighbourhood elements. These elements are categorized into groups based on similarity before synthesis begins. The resulting arrangements are visually and perceptually compelling, suggesting that this style of synthesis could be the most promising approach for GTS.

However, some limitations do exist. In an attempt to start the synthesis process with copied patches from the input sample, the algorithm results in arrangements that vary in their local densities and structures across the texture and fail to capture similar global distributions. Also, similar to the element classification problem of Hurtut et al., the Alves dos Passos et al. method suffers from a sensitive element categorization method due to the dependency on histograms.

Wei et al. [128] discuss a recent drift in the synthesis community towards the hybrid of procedural and example-based texture synthesis as a promising direction of future synthesis research. This hybrid appears to be significant, particularly for geometric texture synthesis (GTS). The taxonomy presented in this chapter found in Tables 2.1–2.4 follows a sequence from the earliest to most recent methods introduced in the synthesis field are is structured in such a way that the evolving nature of texture synthesis leading to geometric texture synthesis is made evident. Judging by this evolution, there may be many possible directions for future work.

2.2 2D Visual texture perception

Before perceptual theories came to be, vision was treated as a black box whose internal workings are unknown and studied by controlling input variables and analyzing the resulting human or animal reactions. Even today, the relationship between what we see and how we interpret it still has its mysteries. However, we have in turn accumulated more information about the different perceptual processes that occur in our brain, explaining most of our interpreted results [142].

Visual perception is a key to how humans view the world. It defines our ability to interpret information of the world around us. We are able to "see" things when our eyes focus on an object or a scene, and the resulting image is reflected onto the retina (a light-sensitive surface at the back of the eye). Early theories of visual perception include descriptive ones like unconscious inference by Helmholtz [48], and the famous Gestalt theory [132]. These theories were quickly superseded by computational models of vision that have had more success in explaining visual phenomena. Models such as those by Marr [93] and Zeki [141] were helpful for driving perceptually-based graphics algorithms.

Researchers have shown that texture perception in human vision is one of the early steps towards identifying objects and understanding scenes [18]. The perception of textures is usually achieved via visual cues found in images, such as elemental shapes, orientation and symmetrical properties, offering guidance for our brain to conduct further texture discrimination and segmentation.

The effects of texture on human perception have been closely studied in vision, neuroscience, psychophysics and computer science [99]. Psychologists who work on understanding reactions of the human mind when viewing texture generally focus on the neural aspect of visual perception. They work on understanding the mechanisms of texture detection and segmentation. Meanwhile, at the other extreme, computer vision and graphics researchers aim at simulating texture perception processes using statistical representations of textural properties. These researchers in turn work on developing computational methods that achieve highly accurate recreations of sample textures.

A major incentive for using perception to understand texture is to introduce in-depth theories of texture perception for future texture synthesis algorithms in graphics, particularly NPR, and to improve existing perceptually based synthesis algorithms. If we can explain how humans perceive textures and accordingly generate highly comparable textures, then we have a stronger claim to say we understand the whole visual and generative process.

Although vision researchers have come closer to understanding how humans perceive textures physiologically, computer graphics researchers continue to develop synthesis algorithms that merely appear to have a perceptual basis. I believe that a comprehensive perceptual basis for texture synthesis algorithms is important, and set out in the remainder of this section to explain the basic known processes of how humans perceive textures by examining theories of perception in the human visual system.

2.2.1 2D texture perception

Theories of visual perception all share the same main underlying question "Why do things look as they do?". This question, initially posed by Gestaltism pioneer Kuffka [45], is now known as the most famous question in the history of perception and the point at which all theories of visual perception start.

There exist many theories of perception that explain details of how we see and all of them are different from one another in their explanations. My goal here is to outline a path towards explaining how humans perceive two dimensional texture while simultaneously drawing on relevant concepts found in theories of visual perception. The major theories of visual perception I draw details from for this section include the Gestalt theory, Empiricism, Gibson's theory, and Marr's computational approach [45].

To explain the perceptual steps involved in the perception of textures, I adopt an information processing paradigm (from many others) to convey when and where texture perception occurs and to emphasize the major important stages of processing in the human visual system. Information processing starts at the retina and consists of: image-based processes, surface-based processes, object-based processes and category-based processes [103]. In this research, I am not in a position to give a complete account of human texture perception and only discuss image-based processes. The scope of human perception is too vast. My goal is to gather sufficient information and offer a basis to explain the perceptual processes involved in 2D texture perception leading up to an understanding of the viewed textures. Given a texture pattern image as shown in Figure 2.4, I ask, what perceptual processes take place in our visual system? How many steps are taken until we actually see the pattern and understand the texture?

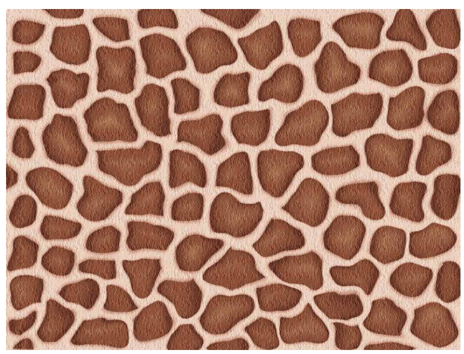


Figure 2.4: *A giraffe skin pattern texture image used to demonstrate texture perception.*

At the retina, we acquire a pair of 2D images, one from either eye, projected from a stimulus to the view point of an observer's eyes. The optical image that reaches the retina is continuous in nature but is processed in a discrete fashion by means of retinal photoreceptors. In formal and computational theories of vision, the final retinal representation is simplified to a square homogenous 2D array of spatially aligned receptors, commonly called pixels, which represent numerical values of illuminance. This representation in current scientific terms is called the *proximal stimulus.*

Gestaltists believe that when we open our eyes we see, not sensations of light, but objects and surfaces. This came as a reaction against an earlier Structuralist movement promoting visual light sensations. Gestaltists believed that given a stimulus, the mind begins to organize our percepts to distinguish between figures found in the field of view and the ground against which they are seen. We can demonstrate this by looking at Figure 2.4. Almost instantly, the larger patches of dark texture seem to lift off the lighter background giving us a sense of foreground and background.

But how do we get from optical information at the retina to perceptual knowledge of a texture stimulus? Many theories have attempted to explain this logical gap in pre-processing. The constructivist theory by Helmholz [48] argued that this gap could be bridged using hidden assumptions along with retinal images to reach perceptual conclusions about a texture. This concept, called Unconscious Inference, can be described as the highest-probability perceptual likelihood in which the visual system computes an interpretation given a retinal image. Although this theory lacks many facts about perception that are now better realized in vision science, it gives a interesting perspective as to the sheer complexity involved in texture perception. How is it that we understand everything about a texture from a two dimensional retinal representation?

After acquiring a retinal representation of a texture comes the first stage of information processing, the image-based stage. This stage takes the retinal representation and implements segmentation to collect basic information about the texture pattern layout. Common image-based processing operations include searching for shape cues, detecting lines and edges, linking lines and edges together, matching up corresponding images in the left and right eyes, defining two dimensional regions in the texture, and detecting line termination points and corners. These processing operations give us practically all the information we need to perceive 2D texture content with appropriate accuracy. More effort to explain the operations and their results in this stage will give us a better understanding of how our perceptual system reacts to textures.

On a computational note, the first steps toward psychophysical texture analysis at the image-based stage were conducted by Julesz [63]. He conducted various empirical studies in an attempt to determine how the visual system responds to changes in order statistics computed as part of a pre-attentive visual mode. One way to obtain pattern familiarity cues is to recreate them from a stochastic process. Julesz specified these processes by their nth order probability distribution defined as the probability of n points having certain brightness values. The three levels of order statistics commonly tested by Julesz and other pre-attentive researchers include: contrast in brightness levels (first-order statistics), homogeneity between brightness levels (second-order statistics) and curvature continuity of brightness levels (third-order statistics). Figure 2.5 shows a sample image that was used in these experiments, involving randomly placed texture primitives correlating the different order statistics.

The research by Julesz led to many contributions; the most relevant to this work being the

Texton Theory [64, 66]. Textons are "the putative units of pre-attentive human texture perception" that are comprised of local features found in the texture like edges, line ends, blobs, etc. [64]. The theory states that it is possible to model pre-attentive human texture discrimination using first-order density of textons. This subsequently led researchers to develop structured approaches to texture analysis that extract texture primitives as local features for texture description [93]. This was later found to be a promising approach for synthesizing 2D arrangements of elements [14]. The main properties of textons include: size, length, width, orientation and density.

Figure 2.5: *A sample synthetic image used for pre-attentive texture discrimination experiments. This texture presents three regions: the background region contains L-shaped figures; the left region contains X-shaped figures; the right region contains T-shaped figures. Julesz's experiments highlight the fact that the left region is easily discerned from the background while the right region is much harder to discriminate [18]. Order statistics are apparent in the light and dark contrast between foreground and background (first order), the line terminations (second order) and the line junctions (third order)*

Preliminary steps towards perceptual organization are also known to originate at the image-based stage of processing. Perceptual organization is a pervasive process that offers insight into how little pieces of visual information may be structured into larger units of perceived objects and their interrelations. Although there are speculations that perceptual organization is learned through experience (originally an empiricist point of view), vision researchers have acquired enough grounds to suggest that innate mechanisms as well as subsequent learning both support perceptual organization [45].

The concept of perceptual organization originates from Gestaltist views on how humans and animals view stimuli. Gestaltists predominantly studied phenomena such as perceptual grouping. Classical principles of grouping include grouping by similarity in colour, size, orientation, common fate, symmetry, parallelism, continuity, and closure [132]. Gestaltists demonstrated all these phenomena with simple geometrical elements, which gave other theorists few alternatives to explain them using different means. Despite the fact that Gestalt grouping principles fail as scientific explanations of how the visual system is structured to view the world, modern research in computational perception continues to relate closely to these ideas.

For instance, Liu et al. offered an interesting hypothesis about human perceptual organization of periodic patterns [88, 89]. They identified underlying lattice structures in regular and near-regular input textures to obtain meaningful building blocks. These blocks, also called fundamental regions, were later used to synthesize pleasing symmetric textures that were in accordance with the grouping principle of symmetry.

Recent research to explain how perceptual organization works by Stylianou-Korsnes et al. [119] shows that the recognition of textures and patterns may be based upon different ordering conditions in memory. In a related inquiry into 2D geometric arrangements, Dodgson [34] investigates whether there exists a correct balance between regularity and randomness that produces more aesthetically pleasing compositions. Through formal experimentation on two of Bridget Riley's Op Art arrangements, he showed that humans have a strong ability to distinguish between fine variations of an algorithm, and that the overall balance in pattern compositions was important when judging aesthetic pleasingness. Dodgson further went on to test the amount of a pattern that needed to be present for it to be immediately identified by an observer. The results show that a good balance can be achieved by retaining about two-thirds of the pattern, while manipulating the other one-third in some way.

These examples suggest that humans do not see textures as a whole and that they make notable inferences on the way the textures are constructed from smaller elements. This concept opposes the Gestaltist views which states that we see things as a whole, offering insight into texture versus shape perception.

The remaining three stages of information processing (i.e., surface-based, object-based and category-based stages) are beyond the scope of research pertaining to two dimensional textures. These stages of information processing involve developing a representation of the external world (visible surfaces, objects and forms) in three dimensions. Although the surface-based processing stage has been acknowledged to enforce extensive spatial layout processing of 2D surfaces in the environment, the context in which this processing is achieved goes beyond the two dimensional image representation we seek.

A final important aspect of known perceptual inferences is memory. Gibson argued that perception is an active process and that it is not matched to past experience, nevertheless his contributions are significant. He believed that instead of relying on experience, the perceptual system has evolved to compensate for certain (light-based) invariant information [101]. Perceiving textures for instance could be illustrated by this example. When we come close to a textured surface, the pattern of stimulation from the environment changes from one moment to the next. This change is not considered random and the retinal image is reported to expand when we approach a surface and contract as the texture passes beyond our field of view. These appearances are explained using transformational and structural invariants.

More recent research by Zeki [141, 142] shows that experience and memory play important roles in how we interpret what we see. In particular, knowledge of our environment is shown to be a determining influence on low-level perception. The methodology and theory conjectured by Zeki also provide measurable anatomical, neurological and physical evidence towards a perceptual theory that can help fill in the gaps that currently exist when applying texture perception ideas in computer graphics.

In general, the information captured by the eyes is not processed as a whole by our visual system. Briefly, various kinds of information take different paths and are treated in various parts of our brain in parallel. Indeed, there have been speculations that there exist four different pathways in the visual system: colour, form, stereo (depth) and motion [142]. This separation is not so clear cut, and offers only a simplification of the visual system. Nevertheless it motivates, at least at a computational level, the parallel processing of these four different kinds of information. This information is relevant for future extensions to texture perception research as it helps explain the perception of coloured patterns and animated texture patterns.

In order to satisfy the foundational goals of the perceptual research in the context of texture perception presented above, I have introduced a large number of concepts found in vision, but did not delve deeply into any specifics. Further exhaustive research is required to clarify the complete perceptual process, and more intricate analysis is necessary to compare the various perceptual theories of vision.

2.3 Evaluation of synthesis algorithms in NPR

In the NPR community there is a growing consensus that more careful and objective means of evaluating new algorithms are needed to judge the quality of results [49, 59].

However, due to the broad nature of NPR algorithms and the different sub-areas involved, few evaluation methods have been proposed.

In general, there are two ways to achieve effective evaluations. This first is conducting psychophysical studies. These studies can be described as "the analysis of perceptual processes by studying the effect on a subject's experience or behaviour of systematically varying the properties of a stimulus along one or more physical dimensions" [21]. This involves designing suitable interfaces to present algorithm results (or to compare them with others) and recruiting human participants. To elicit feedback in the study a researcher may choose to devise a set of questions to probe participant observations; these result in "qualitative" information. A researcher and may also choose to measure significances using metrics; this results in "quantitative" information. Evaluations like these are often conducted in controlled environments, but do not necessarily have to be completed in person. The second approach to evaluation involves gathering only quantitative measures from algorithm results. These measures are used to describe and compare between lower level raster-based information, spatial information, or even special phenomena investigated by the researcher. Collected measurements are then presented to the reader visually or listed according to significance.

In NPR, most evaluations involve human participants because of the subjective visual nature of the results and the aesthetics involved. Carefully designed studies essentially solicit thoughts, comments, and impressions from human participants to determine the overall success of the algorithmic sources. Quantitative metrics are also utilized to illustrate whether the results achieve expected statistical criteria.

For a comprehensive list of texture synthesis algorithms and existing evaluations please refer to the survey by Wei et al. [128]. A more recent summary of evaluation literature in NPR can also be found in the analysis by Isenberg [57]. Below, I review some of the most relevant texture synthesis evaluation methods related to this dissertation and divide them according to the types of experiments they conduct: psychophysical or purely quantitative.

2.3.1 Psychophysical evaluations

Two of the most prominent evaluations on synthesized textures are the studies by Isenberg et al. [59] and Benjamin Balas [11]. Isenberg et al. investigate the quality of automated pen-and-ink algorithms by comparing computer-generated drawings to the work of human artists. In their study, participants were given collections of drawings printed on paper and instructed to separate them into piles according to their own criteria. The results highlight differences between human and computer-generated drawings, as well as positive aspects of both. A similar pile-sorting strategy has been used in computer vision for classifying natural textures into meaningful categories [11]. In these studies, the unrestricted comparisons allow participants to accomplish the task at their own pace without external influences. I believe that both of these experimental strategies show promise for the analysis of geometric texture arrangements and attempt such tests in this research.

Lin et al. [85] present an evaluation to support their own quantitative metric designed to describe the regularity or near-regularity of raster-based textures. To do this they compare the performance of four synthesis algorithms to understand how much a near-regular texture's global regularity and local randomness affects human judgement. Lin et al. use their statistical score to measure regularity through user-defined translation vectors. This score is a statistical measurement that characterizes the regularity of a near-regular texture computed using user-defined translation vectors. In addition to this study, Lin et al. conducted a supporting subjective evaluation to determine the significance of global regularity of textures on participant similarity ratings. Participants were presented with an exemplar and two textures on a computer screen and asked to provide a similarity ranking of 1 to 4. The findings suggest a bias in favour of one of the synthesis algorithms adopted. The results also support the regularity metric as a reliable evaluation measure of structural similarity.

Texture fractalization is used in animation to ensure the temporal coherence of stylized texture content. It involves combining many versions of a source texture at different scales using alpha-blending which make the appearance of the textures flow smoothly between movie frames [17]. An analysis of the effects of fractalization on textures by Bénard et al. uses an average co-occurrence error to evaluate results. In the study they asked participants to rank pairs of original and fractalized textures according to a level of distortion induced by a fractalization algorithm. The results suggest that the average co-occurrence measure effectively correlates with perceived distortion.

In a recent aesthetic investigation, Wyvill et al. [137] conducted a user study to determine the relative aesthetic merits of parametric and implicit curves. In the first of their two studies, participants were given random pairs of images on a computer screen and asked to choose the most aesthetically pleasing image. In the second study, participants had to give an aesthetic score using a Likert scale. The results show that images generated using implicit curves were greatly preferred over parametric curves.

2.3.2 Quantitative evaluations

Metrics for quantifying statistical content in images have been used in many areas of NPR. In texture synthesis, evaluations consist of gathering statistical information from raster images and comparing them with their sources [36]. More recently GTS algorithms have used point-based measures which provide us with a higher-level understanding of spatial distributions. In the following I describe relevant examples from each of these areas. More details on quantitative metrics applied to GTS are given in Chapter 6.

Pixel-based evaluations

Evaluations in pixel-based texture synthesis involve running quantitative metrics on synthesized results to measure the amount of pixel similarity to example inputs. Despite the various different texture models that have been proposed, only those based on the Markov Random Field (MRF) can be evaluated with higher amounts of certainty. MRF algorithms model texture as a realization of a local and stationary random process. They do this by guaranteeing that local neighbourhoods in the input and output are similar to enhance the overall perceived appearance. A review of MRF methods in texture synthesis can be found in the state of the art survey by Wei et al. [128].

When comparing computer-generated and hand-drawn images, Maciejewski et al. [92] and Martín et al. [95] adopt the statistical grey-level co-occurrence measure to evaluate differences, this time between stipple images. Maciejewski et al. measure the frequency of grey levels in an image through three values: contrast, energy and correlation. Using these to compare hand-drawn and computer-generated stippling showed that human-drawn ones had better correlations overall and correlate more with natural textures. Computer-generated drawings tended to have regularities that easily distinguished them as algorithmically generated. Meanwhile Martín et al. analyze results of an example-based stippling algorithm based on grey-level co-occurrences using the same measure to validate the results. In doing so, they proved that example-based approaches to NPR are better at generating results that are less distinguishable from natural/hand-drawn ones.

Other measures like density, mean, variance and entropy can be used to understand content and compare similarity between raster images. These measures rely mainly on a uniform spatial domain, in which a synthesized texture can be analyzed as a uniform sampled signal. In Chapter 6 I give an analysis based on some of these measure and show that most of them do not contribute much information for validating similarity of GTS arrangements, at least not on their own.

Point-based evaluations

Geometric textures provide a unique domain in which to study perceptual problems related to patterns. Analyzing vector textures for discrete geometric elements is easier than drawing the same information from pixels in raster images. Despite this, few geometric texture synthesis methods have been evaluated effectively. The closest area to GTS that studies spatial relationships between points on a plane is point-based sampling.

For example, Öztireli et al. [102] and Wei and Wang [130] compare synthesized point distributions from example inputs using pair-wise correlation and spectral analysis respectively. These measures are meant to provide visual comparison between multiple distributions and suggest similarity if the functions exhibit similar patterns. Wei et al. characterize point distributions using a modification of the Fourier Transform. Öztireli et al. on the other hand develop a GTS algorithm based on a second order statistic called pair-wise correlation that captures the overall distributions of one or more elements in stochastic/random arrangements. To compare the final results, they apply the same metric to all sources and plot the values using charts. Their findings show that pair-wise correlations are effective in evaluating the spatial distributions of multiple elements. The studies by Maciejewski et al. [92] and Martín et al. [95] also included a spatial analysis of stippling to compare between synthesized and human-generated results.

Chapter 3

Similarity between 2D geometric texture arrangements

In this chapter, I propose to decipher the meaning of similarity in the context of geometric texture arrangements. I do this through subjecting human participants to a series of psychophysical experiments (Section 3.3 and 3.4). These studies help elicit information about similarity in and between geometric textures indirectly first by observing participants manipulate texture arrangements and then through ranking multiple arrangements based on similarity preferences.

3.1 Introduction

While recent GTS algorithms claim to produce visual properties that are related to known perceptual processes, none of them attempt thorough evaluations of the final results. Instead, results are presented uncritically, leaving the reader to judge the ultimate aesthetic appeal (examples include Ijiri et al. [56] and Hurtut et al. [54]). These qualitative evaluation styles are rudimentary and include ad hoc fine tuning that both inhibits reproducibility, and obscures the true value of the algorithm.

As synthesis methods evolve it is becoming even harder to compare synthesized texture results with any amount of accuracy. Assessment of whether or not results are acceptable, visually similar, representations of the exemplar input depends on both the degree of accuracy of the layout and the overall appeal. In NPR, Hertzmann et al. calls for the development of more objective means of evaluating newer algorithms [49]. For GTS, this is not immediately tangible without a better understanding of how we perceive texture arrangements and how we decide similarity.

One way to interrogate the human experience of motif arrangements is to ask human subjects to manually synthesize arrangements, and to have them evaluate their similarity to the exemplars that inspired them. In this chapter, I offer first steps towards grounding GTS with visual perception by looking closer at the aesthetic nature of arrangements and understanding the descriptive reasons that lead humans to visual preferences when judging texture similarity.

To achieve a more principled foundation for geometric texture synthesis, I conducted two psychophysical studies. The first study explores how participants analyze an exemplar with the goal of synthesizing a larger geometric arrangement that is, in their estimation, similar to the exemplar. The second experiment gives participants a set of larger textures and asks them to rank the arrangements according to their similarity to the exemplar. In both cases, alongside creating and ranking, I ask participants to explain the features they use to create and assess their arrangements.

Together, these experiments provide insight into the features people use to assess similarity between a sample and a synthesized arrangement. Understanding why and how two different example arrangements are ranked as similar yields insight into the way people analyze and assess arrangements. While similarity is only one mean of assessing the success of GTS algorithms, I argue that it is an important first step to guide more effective future evaluations of GTS.

3.2 Perceptual inquiry into similarity of 2D arrangements

The goal here is to look closer at the aesthetic and descriptive reasonings that lead to similarity preferences. Given synthesized textures such as those shown in Figure 3.1, I look at which of the results people find more similar to the sample and why. Identifying the major steps taken by people when perceiving arrangements and the factors used to compare them is necessary to provide a basis for reliable comparisons between GTS algorithms. Once identified, these steps will help clarify why people prefer certain textures from any set of samples.

To develop a viable evaluation that measures similarity between sample arrangements and synthesis results we ought to first identify important global and local visual aspects of

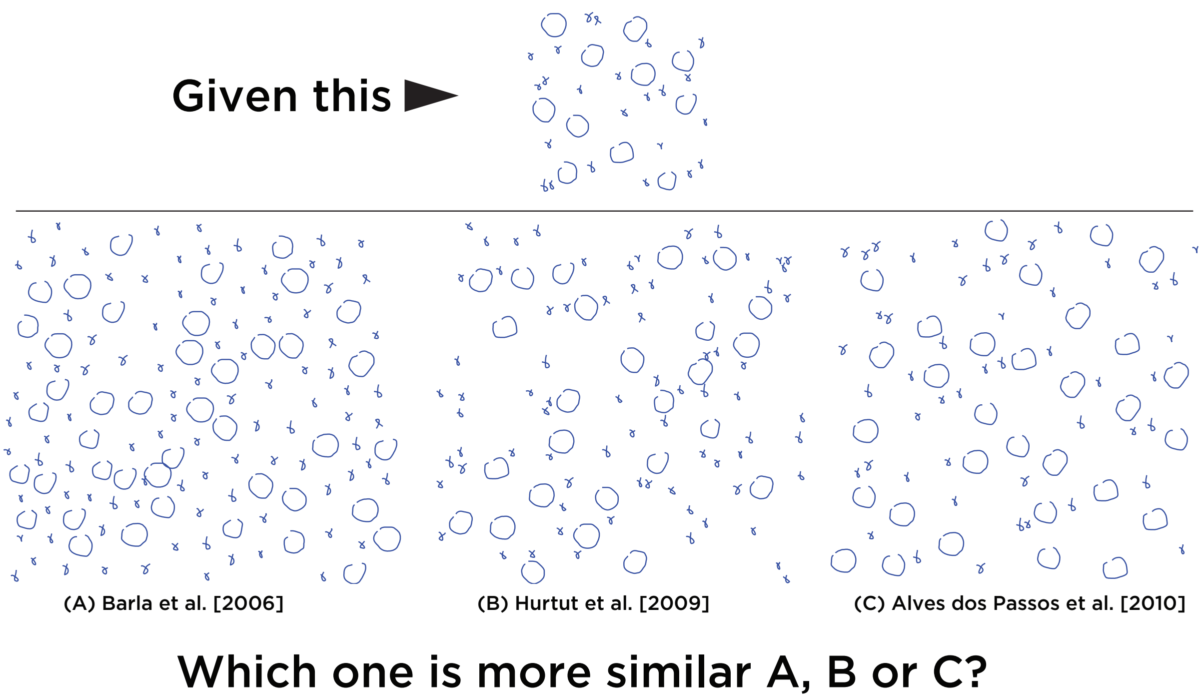


Figure 3.1: *A comparison between three GTS arrangements. © 2009, Hurtut et al. [54] and © 2010, Passos et al. [5], used with permission.*

arrangements, and then verify that similar factors are used to compare synthesized and sample arrangements. As a result, I structure this inquiry around two user studies. The first study explores how participants synthesize larger arrangements from smaller exemplars, and asks the participants to evaluate their success at generating the larger arrangements (Section 3.3). The second study examines how participants evaluate the similarity of human and computer-generated geometric arrangements to their appropriate exemplars (Section 3.4).

3.3 Acquiring and analyzing human-generated 2D geometric arrangements

This study examines how people perceive 2D geometric arrangements from given exemplars by watching them generate their own similar textures manually. Since initial visual impressions highly affect human perceptions of texture, an analytical study is an essential first step towards identifying the visual aspects of human perception involved when judging similarity between geometric arrangements. My primary goal here is to gather enough detail to form an explanation of how people judge similarity after generating their own geometric arrangements.

A mixed-method (qualitative and quantitative) research design was adopted to answer two main research questions during the study. First, what comparisons are involved during the generation of an arrangement? Second, what factors are involved when judging similarity of an arrangement against a given exemplar? I first describe the study designed to answer these questions (Section 3.3.1). Then, I present the qualitative and quantitative data collected from the experiment (Sections 3.3.2 and 3.3.3). Finally, I present the findings in light of the two primary research questions (Section 3.3.4).

3.3.1 Design and procedure

Participants: Research participants consisted of 20 university students (undergraduate and graduate). No prior knowledge of texture synthesis conventions, nor any explicit means for measuring accuracy of generated results were required. Throughout this chapter participants will be referred to as P1 to P20. All participants were compensated with gift cards for their efforts.

Stimulus arrangements: The stimulus template of exemplar arrangements developed for both user studies is composed of nine randomly generated sets of circular dots, the motifs. These random arrangements were generated using a uniform pseudorandom number generator which depends on the number of dot sizes and the desired overall density. The routine places motifs using rejection sampling: new motifs are placed only if they are sufficiently far away from all existing points.

The motifs are coloured black and vary in their sizes and numbers. Figure 3.2 shows the stimulus template with each row grouped depending on the dot sizes involved. Group A contains only large dots; Group B contains large and medium dots and Group C contains large, medium, and small dots. The one exception to random placement is a sample in which large dots are visibly paired, shown in the centre of the template. This stimulus was generated with the same random routine in addition to a constraint that allows the placement of only two motifs in closer proximity.

Interface and methodology: Using a simple computer interface, shown in Figure 3.3, each participant completed nine trials. In each trial they were given one randomly selected stimulus from the nine exemplar arrangements along with a larger, rectangular empty

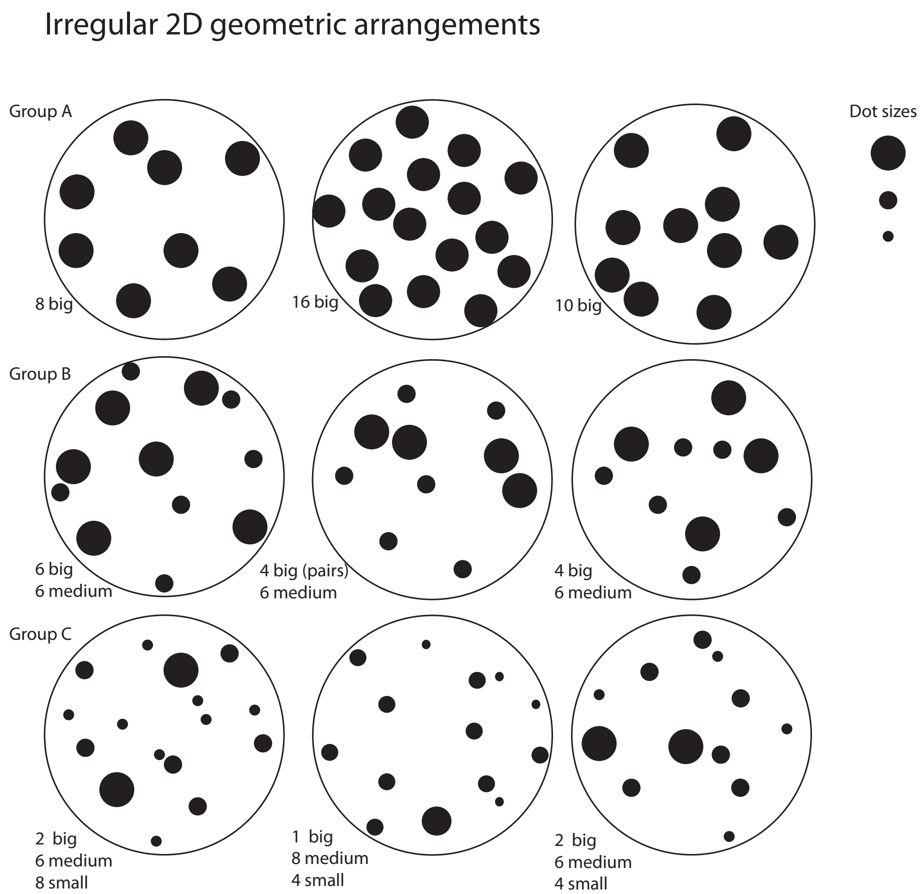


Figure 3.2: *The stimulus template of arrangements used in the user studies.*

space (the canvas). Based on their own interpretation of the stimulus, participants were asked to "construct a new larger arrangement that appears to have been generated from the same underlying process". Participants constructed their arrangements, by selecting motifs from a palette on the left. Individual motifs could be moved or deleted, but groups of motifs could not be operated upon collectively. Once an arrangement was complete, it was not possible to return to it later.

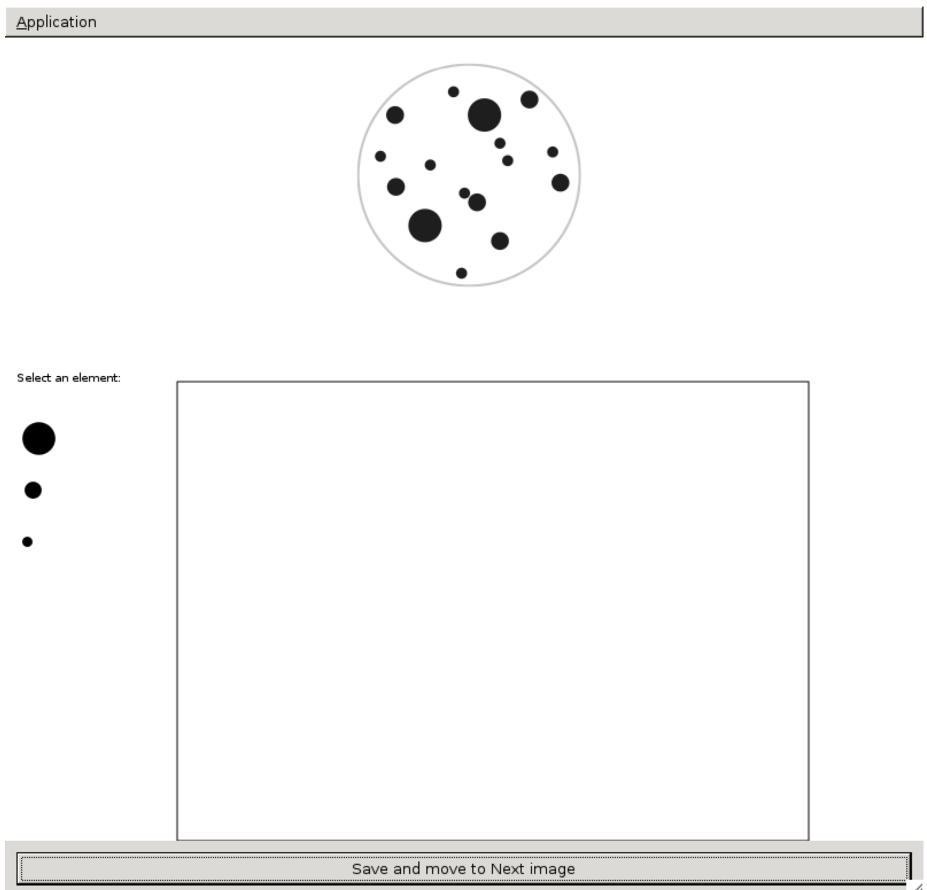


Figure 3.3: *A sample interface used during the first user study.*

Participants were also asked a set of qualitative and quantitative questions about their arrangements in the form of a questionnaire; they were encouraged to provide their answers in a semi-structured format either during or after the generation task. The list of questions is as follows:

1. Describe this sample texture.

2. What factors of the texture pattern affected your judgement the most when creating your own arrangement?
3. How did you start off creating your arrangement?
4. What steps did you take to ensure a similar arrangement?
5. What did you think the larger sample image would originally look like?
6. Rate how pleased you are with your final result in comparison to the sample texture (on a five-point Likert scale).
7. Rate how visually similar you believe your generated arrangement is to the sample arrangement (on a five-point Likert scale).
8. What factors did you consider when describing visual similarity in Question 5?

Although no time constraints were given for the generation task, participants were encouraged to consider completing the task within 90 minutes. On average participants were able to generate the complete set of nine arrangements and complete the questionnaires in 65 minutes. After completing the nine arrangements, each participant took part in an open-ended interview concerning their perception of the element layouts and their thoughts on the generation process.

Data collection and analysis: The complete data set consists of over 25 hours of audio and screen recordings, including answers to questionnaires and interviews for all of the participants. The analysis process characterizes a grounded theory approach adopted from Creswell [26], common in HCI research for understanding a target phenomenon.

3.3.2 Qualitative analysis: a grounded theory approach

The qualitative information gathered from the questionnaires helps to elucidate the factors participants felt were important when perceiving arrangements, and how they judged similarity. These important visual processes have not previously been considered in geometric synthesis algorithms. I believe that once identified, these factors can be used as effective similarity measures towards future evaluations.

Analysis began with open coding, which examines small sections of participant verbal replies made up of individual words and sentences. Replies were repeatedly sorted and codes were saturated until core categories emerged. A large poster board with movable tags was used to arrange and re-arrange codes within and across categories.

The open-ended questionnaire allowed the examination of three broad themes associated with the geometric arrangements participants constructed. The first two questions examined the attributes of the arrangements the participants were drawn to, i.e., the causal attributes that motivated the texture generation strategies employed by the participants. Questions 3 and 4 explored these strategies in further detail. Finally, Questions 5 and 8 revealed the attributes that participants used to evaluate the similarity of the constructed geometric arrangement to the original stimulus.

Causal attributes

During the analysis of participant replies to Questions 1 and 2, three main factors were seen to motivate participant generation styles. These causal attributes are (1) dominant visual properties perceived by participants, (2) identified local themes and (3) recognition of large spatial structures by participants.

The first attribute involves dominant visual properties that were perceived by participants from the stimulus before the generation of their arrangements. I subsequently classified participant verbal replies into two major categories: (1) global visual appearances and (2) local shapes and forms. Global visual factors found in the stimulus include (a) density (sparsity, number and frequency of elements, intensity), (b) distribution type (regular, irregular/random), (c) prominent focal points, and (d) spacing (white space, proximity, inter- and intra-element distances, pairs, clustering).

The second causal attribute involves identifying shapes and forms constructed from closely located elements and using them to generate arrangements. This local object searching routine was frequently adopted by participants. The most commonly noticed shapes include lines (representing continuity of close elements) and geometric shapes (triangles, rectangles, ovals, and alphabetic letters). In some instances participants saw more than one shape; for example, when asked to describe a sample P10 replied "I see the letter V and curved lines".

The last attribute combines thoughts of structures inspired at a larger scale. Some participants allowed their imaginations to guide their understanding of the sample. Participant P6 described one sample as "a watering jug, with a nozzle, handle and a body" and another sample with "a face, eyes, mouth, and hair". Once these structures were identified, it was difficult for participants to see them in any other way. When later evaluating their arrangements for similarity, participants looked for these structures first. This phenomenon can be described as a primitive form of visual illusion [103], sometimes known as a "pareidolia".

Strategies for generating geometric arrangements

Based on participant replies to Questions 3 and 4, and including written, audio and video recordings, a thorough analysis resulted in three major approaches adopted by participants for generating their arrangements: a tiling approach, a structured approach and a random approach. The three strategies are a direct result of the causal attributes discussed previously. Participants were also noticed switching between either of these generation strategies throughout the study. Below I discuss each of these approaches and then describe the visual attributes participants considered at generation. See Figure 3.4 for examples.

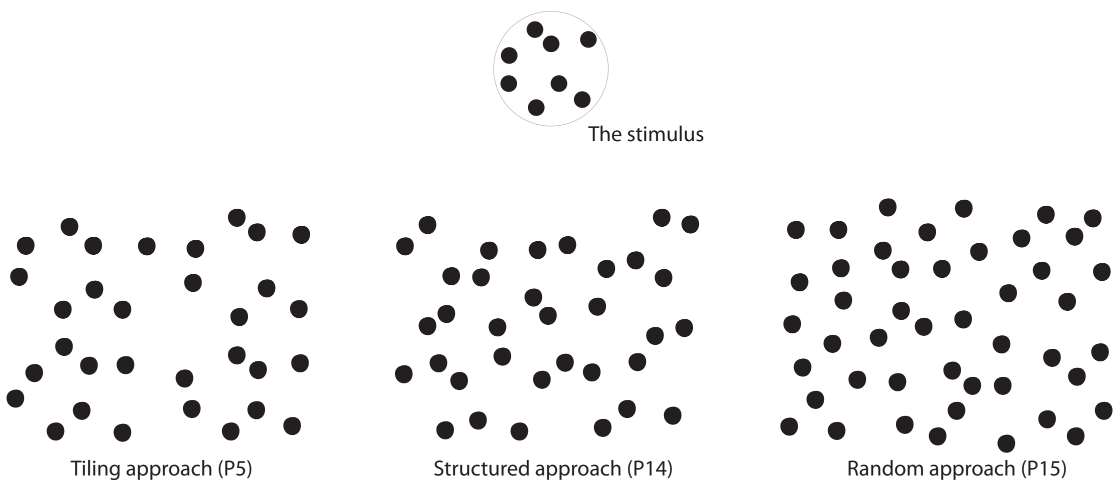


Figure 3.4: *Sample participant generated arrangements according to the three generation approaches discussed in Section 3.3.2*

The **tiling approach** involves participants copying the stimulus multiple times to fill the canvas, usually regularly and conforming to a grid. Participants relied on three key visual factors that resulted in a tiling approach. They either (a) identified the stimulus as a shape, (b) saw the whole stimulus as a pattern, or (3) could not identify an obvious pattern which led to copying the complete stimulus onto the canvas anyway. Replicating tiles involved using distances and randomness as cues.

A **structured approach** involves identifying prominent forms and objects from the stimulus and then procedurally filling the canvas with them. This was usually done by starting the arrangement at the center or top left corner of the canvas. Participants who chose to generate arrangements using a structured approach were guided mainly by three visual factors; they either (a) identified shapes in the stimulus, (b) perceived notable clusters of elements and white space, or (c) perceived focal points and element size ratios.

A **random approach** involves participants placing different sizes of motifs randomly across the plane with the aim of achieving a specific density. In this case, participants could not identify any shapes or structures, and only saw random arrangements of dots. During generation, participants strove to match density in the arrangements for each of the existing element sizes independently. The term "random" here embodies the randomness of motif placement observed during texture generation.

As an exception to the three methods above, a total of two participants saw arrangements as some form of pareidolia and consequently generated arrangements to represent their visual perception. One participant saw a face in the exemplar and drew a larger face in the synthesis space, Although an interesting perspective on synthesis, I refrained from taking them into account during analysis.

A descriptive analysis of participant replies to Questions 3 and 4 shows that five visual cues were considered when generating arrangements. These include repeatedly comparing (1) relative spacing (proximity, white space and pairwise spacing); (2) density, element size, frequency distribution type (randomness, regularity); (3) the resemblance of local element neighbourhoods via angles formed at junctions, continuity, and focal points; (4) avoiding overlapped elements, generating new shapes, symmetries and obvious horizontal and vertical group alignments; and finally (5) continuously sampling the canvas for identified patterns. Some participants were noticed placing dot pairs at varying orientations, while others ensured exact alignment to those in the stimulus. This decision was noticed later as a contributing factor in evaluating similarity.

Strategies for evaluating similarity

After generating arrangements, some distinguishing features were considered by participants when ranking similarity. Participants mentioned multiple reasons they thought affected how they judged satisfaction and similarity of their generated arrangements. These influencing factors include (a) symmetry, (b) apparent shape, (c) repetition, (d) conformity to conceptualized pattern rules, and (e) accuracy and inaccuracy of copied samples (parts or whole). Symmetric aspects that particularly influenced similarity decisions depended on the element layouts and the participant's ability to detect a horizontal or vertical axis of symmetry from a shape or part of the stimulus. For example, when asking participant P5 to describe the stimulus, one reply included "...semi-symmetric shape that's reflected". Then when asked about the factors that affected their similarity judgement the most, the participant replied "I followed the contour elements of the given sample, it's like a fish". The generated arrangement by participant P5 can be seen on the left side of Figure 3.4.

When asked to rate similarity of their final arrangements and then describe the factors they relied on (Question 8), participant data revealed three distinguishing strategies. They either chose to (1) sample the generated arrangement for the stimulus, (2) look for similar parts or discrete patterns within the arrangement, or (3) compare the overall aggregate of the arrangement to the stimulus.

The first strategy involved sampling the generated arrangement for the complete/whole stimulus using the circular perimeter of the stimulus as a guide for judging spaces. This strategy was most obvious for participants who chose to generate their textures using a tiling approach (repeated instances of the stimulus). For example, an unsatisfied participant P3 said "I imagine a circle with dot content but I resulted with lots of spaces on the outer perimeter which looked wrong".

The second strategy concerned either identifying parts composed of grouped elements or visually discerning distinct patterns within the arrangement. Participants used one or more groups of elements they located in the sample and compared them to their generated arrangement. These groups were either shapes (i.e., three elements in a cluster look like a triangle), lines/curves, alphabet letters, or even faces and other forms of pareidolia. If any of these grouping types were identified in the stimulus, participants would certainly look for them in the generated image. Participant P9 reported, "Yikes. I can't see a pattern or structure. I saw ovals or lines horizontal tilted curved lines. Density isn't an issue here just the structure". In another case participant P14 had initially noticed "two big black dots in pairs", but when evaluating the similarity of the generated image said "I can't see groups (of pairs) close together in generated image. They are too far apart than the pairwise distances in the sample". But for instances where the stimulus was not perceived to contain any obvious cluster or shape, participants depended on conceptualized rules and influencing factors, described earlier, to explain them. These rules remained vivid in the participant's mind throughout the generation process and were used when judging the similarity to the sample. Participant P13 described a stimulus as "Two large dots. The number of smaller circles formed a network in pattern. The large [elements] just fill space". This participant later went on to explain their arrangement generation style as "Copies of a small network, and branches coming out from it". Then when judging the arrangement's similarity, they rated it poorly and said "What was in mind and generated is not the same. Not sure what it should be like exactly".

Shapes were not often noticeable unless the copied distances and orientations of elements were moderately accurate. Other visual impressions led to predefined pattern rules that describe how sample arrangement elements were grouped and connected. When copying samples into the canvas, the pattern as well as accuracy were two of the primary concerns. Many participants tried with much effort to copy elements and relative distances precisely but in general such attempts still contained noticeable flaws. Participants P1, P5 and P20 distinctly stated that generated arrangements were "not accurate, spaces are not right". P9 expressed this difficulty by stating "it's too hard to make the same distances [as the sample]". These flaws influenced final similarity ratings.

The final comparison strategy includes performing an overall aggregate check on the generated arrangement. This required scrutinizing the arrangement for prominent visual properties, notably density, and relating them to the stimulus to discern similarity. Participant P13 described this process by saying "checking density, ratio of sizes. No pattern. Proximity of dots. Bottom left has too much space otherwise it would be extremely similar".

3.3.3 Quantitative analysis

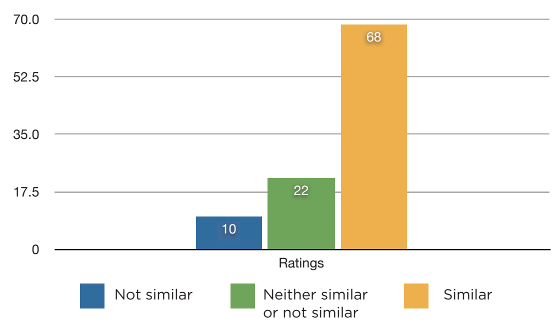
In this quantitative investigation I try to understand how participants rated visual similarity of their generated arrangements in comparison the the stimuli, which includes an analysis of replies to two 5-point Likert scale questions asked during the study session (Questions 6 and 7). The questions were as follows: (6) Rate how pleased you are with your final result in comparison to the sample texture and (7) Rate how visually similar you believe your generated arrangement is to the sample arrangement.

A Chi-square test proved that results of the two Likert scale questions were highly correlated ($P \leq 0.00001$). This indicates that almost every participant who thought their generated image was satisfactory also thought that it was similar to the given sample. Accordingly, only similarity findings will be discussed in the analysis. Answers to five-point Likert questions were converted to a three-point scale (similar, neither similar nor not similar, and not similar) to present the figures in this section.

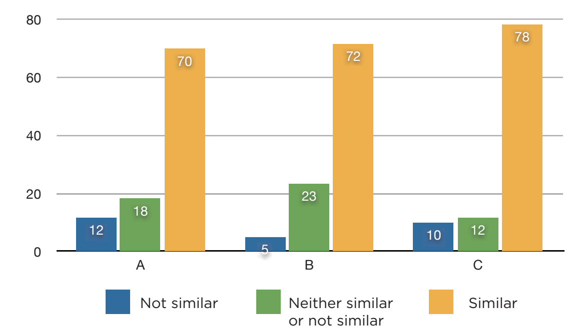
Each of the 20 participants generated arrangements for the complete set of nine stimuli shown in Figure 3.2, producing a total of 180 user-generated arrangements. Table 3.1 presents the numbers of arrangements generated according to the approach adopted.

Approach	out of 180 images (100%)
Tiling	61 (34%)
Structured	83 (46%)
Random	33 (18%)
Not a texture	3 (2%)

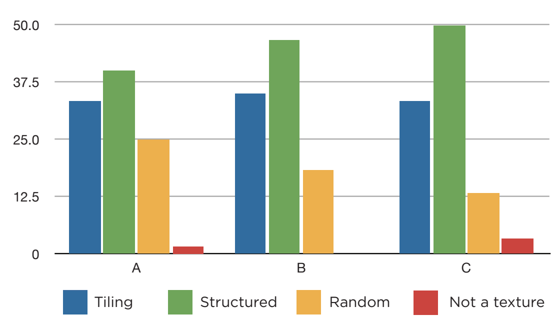
Table 3.1: *The number of arrangements according to the generation approach.*



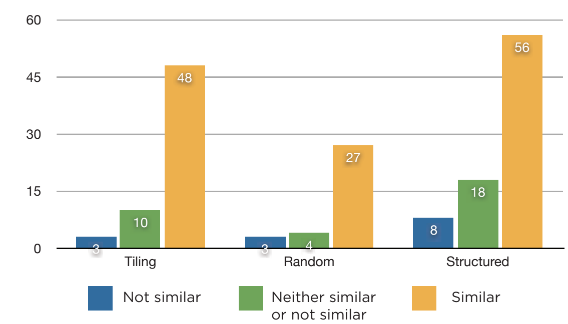
(a) Overall similarity ratings for all generated arrangements. (Likert question (2))



(b) Total similarity rating per image group type.



(c) The human-generated arrangements classified according to the image group and generation approach.



(d) Overall similarity ratings for all 180 generated arrangements according to generation approaches

Figure 3.5: *The quantitative analysis charts from the first user study data.*

The similarity chart in Figure 3.5 (a) shows that 68% of all user-generated arrangements (regardless of approach) were rated "similar" to the stimulus upon which they were based. To further understand these arrangements I split the analysis according to their generative approaches. To understand the reason behind similarity choices I looked for patterns in the adopted approaches. Table 3.2 shows a detailed breakdown of the types of approaches adopted by participants. These numbers suggest that the adopted approach may have some correlation to the context and visual aspects identified in the stimulus.

Approach	Number of participants
Tiling only	3
Structured only	2
Random only	0
Tiling and random	1
Tiling and structured	2
Structured and random	5
Tiling, structured and random	7
	20

Table 3.2: *The types of approaches adopted by participants to generate arrangements for the 9 stimuli.*

To show whether there exists correlation I looked closer at how image groups (A, B or C) affect participant similarity ratings. Figure 3.5 (b) shows that participants rated their arrangements as "similar" regardless of the image group. This means that there is no direct relationship between image group type and similarity ranks.

Further analyzing results for the dependency of the generation approach on image group type, I find that most arrangements were generated using a structured approach (Figure 3.5 (c)). Interestingly, the numbers of arrangements generated using a structured approach increased from Group A to Group B to Group C. Equivalently, random approaches to arrangement generation decreased from Group A to C. The fact that Groups A, B and C contained elements of one, two and three sizes respectively seems to have a direct correlation with how people generate arrangements. The increase in complexity (element sizes and numbers) made it easier for people to discern shapes and interactions between them. These groups and clusters of elements could explain why a structured approach was adopted more often. A random approach on the other hand was generally adopted for arrangements that have no apparent shapes or structures. These approaches were more appropriate for Group A images that have the fewest discernible features and became less and less appropriate as arrangements became more complex. Finally, arrangements generated using a tiling approach are noticed to be constant across all image group types. This shows that tiling approaches were chosen as suitable alternatives to structured approaches for generating geometric arrangements. The reason for this may either be because a participant (1) sees the whole stimulus as a complete shape or pattern and was satisfied to tile it out or (2) could not see any shape but thought tiling was a valid strategy for generating arrangements.

Further considering similarity ratings for individual generation approaches in Figure 3.5 (d), I notice that the majority of arrangements generated using either approach resulted in higher similarity ranks. This shows that participants were generally pleased with the results generated from the approaches they adopted.

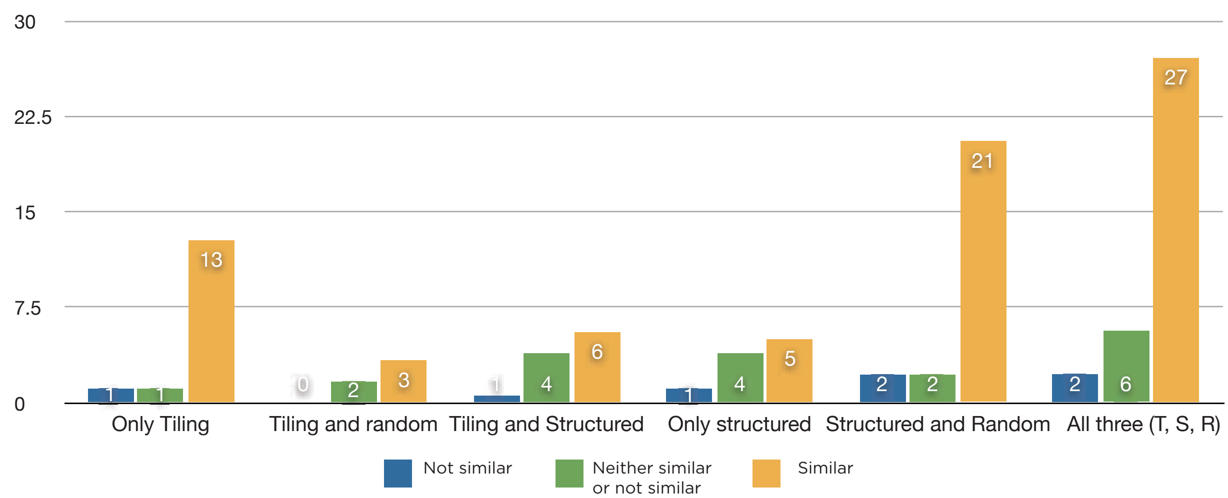


Figure 3.6: *Total participant similarity ratings according to the mixed generation approaches they adopted.*

A subsequent inquiry into the total similarity ratings of grouped approaches adopted by participants (Figure 3.6) highlighted the fact that distance copying inaccuracy, increase or decrease in overall white space, inappropriate density representation, difficulty in discerning shapes and local neighbourhoods had effects of participant decisions. Participants were noticed to have a very keen sense of detecting copy accuracy and density changes. This appeared more often in similarity ratings of tiling only, and mixed structured and random generated arrangements. Due to the low number of participants who exclusively chose only structured or tiling mixed approaches, it is difficult to validate the appropriateness of the adopted approaches. However, the goal here is not a pure quantitative evaluation of gathered participant satisfaction or similarity; instead, I aim to elicit attributes of geometric textures that influence an individuals' assessment of similarity.

3.3.4 Results and discussion

The results of the mixed-method analysis presented in this section can be summed up into (1) visual attributes and (2) strategies used by participants to generate and compare their final arrangements. When a person is asked to generate an arrangement from a stimulus, multiple perceptual processes appear to take place and result in three prominent attributes. I call them local and global visual attributes, local shapes attributes, and larger spatial structure descriptions. These attributes confirm perceptual theory advances presented by Marr [94] specifically in the context of geometric texture synthesis. Depending on the most noticeable attributes gathered from a stimulus, the participant will choose to generate their arrangement using either a tiling, structured or random approach. These attributes and strategies succeeded in providing participants with the ability to: (a) rate/judge similarity, (b) give a level of satisfaction, (c) give a sense of visual appeal, (d) recognize content, and (e) inspire other ideas.

Another finding shows that regardless of the image type, participants favoured a structured and tiling approach to texture generation over a random approach. Although visual properties played an important role in perceiving arrangements, recognition of the stimulus and local groups of elements were key factors in how participants rated the similarity of their final results. The next user study (Section 3.4) helps shed more light on this observation.

3.4 Evaluating similarity of generated arrangements

Given the similarity analysis of human generated arrangements in the previous study, how can we trust the results of participants' evaluations of the similarity of their own generated patterns? This is neither a reliable or objective means to assess true visual appeal. This next study collects all the generated arrangements from the previous study, adds computer-generated ones and gives them to a new group of people to evaluate similarity in an unbiased way.

In this evaluation I observe how people rate the quality of geometric arrangements. To do this, I collect participant feedback on the features that made them rate one arrangement as more or less similar than others. Armed with results from the previous study, the aim is to deliver a plausible set of metrics that effectively reports on how people judge similarity. To accomplish this I adopt a smaller mixed-method research design. As in the previous section, I will first describe the study we designed to capture these metrics (Section 3.4.1). Then, I will present the qualitative and quantitative data collected (Sections 3.4.2 and 3.4.3). Finally, I will discuss these findings relative to the results of the first user study (Section 3.4.4).

3.4.1 Design and setup

Participants: Research participants consisted of 20 university students (undergraduate and graduate). Five of them had previously completed the first user study at least two months prior. All participants were compensated with gift cards for their efforts.

Sample arrangement set: The sample data set of arrangements consists of nine stimulus arrangements and a total of 225 geometric arrangements. Of them, 180 were user-generated arrangement from the previous user study; 36 were computer-generated random arrangements (four per stimulus arrangement) and nine were accurately tiled instances of each template stimulus. The purpose of the additional computer-generated textures was to account for inaccuracy when copying stimuli by hand in the first study. These textures were included according to their generation approaches in the analysis. The complete dataset has been made available online.[1]

Computer-generated random arrangements were generated using the same routine used to generate the template sample arrangements described in Section 3.3.1. Computer-generated tile arrangements involved copying and repeatedly pasting the stimulus arrangement in close proximity. The final tilings each had six sample stimuli laid out on a regular grid (3 columns $\times$ 2 rows) similar to the 2 $\times$ 2 tiling sample in Figure 3.4.

Interface and methodology: In each task one stimulus arrangement was shown along with five randomly selected geometric arrangements presented side-by-side. See Figure 3.7 for a sample task.

Participants were asked to drag and drop arrangements into the appropriate boxes below from the least similar arrangement to the most similar one. They were then asked to describe why they chose the extreme least and most similar textures. Every participant saw each of the nine template stimuli five times (with five samples each), adding up to 45 sets of ranks per participant. This collectively covers all 225 samples in our sample data set. Participants were asked to complete the ranking and provide reasons for each

[1] http://www.cgl.uwaterloo.ca/~zmeraj/publications/NPAR_2011_GTS.html

set during a 1-2 minute time frame. Timing was chosen empirically, such that participants had enough time to compare textures without having too much time to overanalyze their decision.

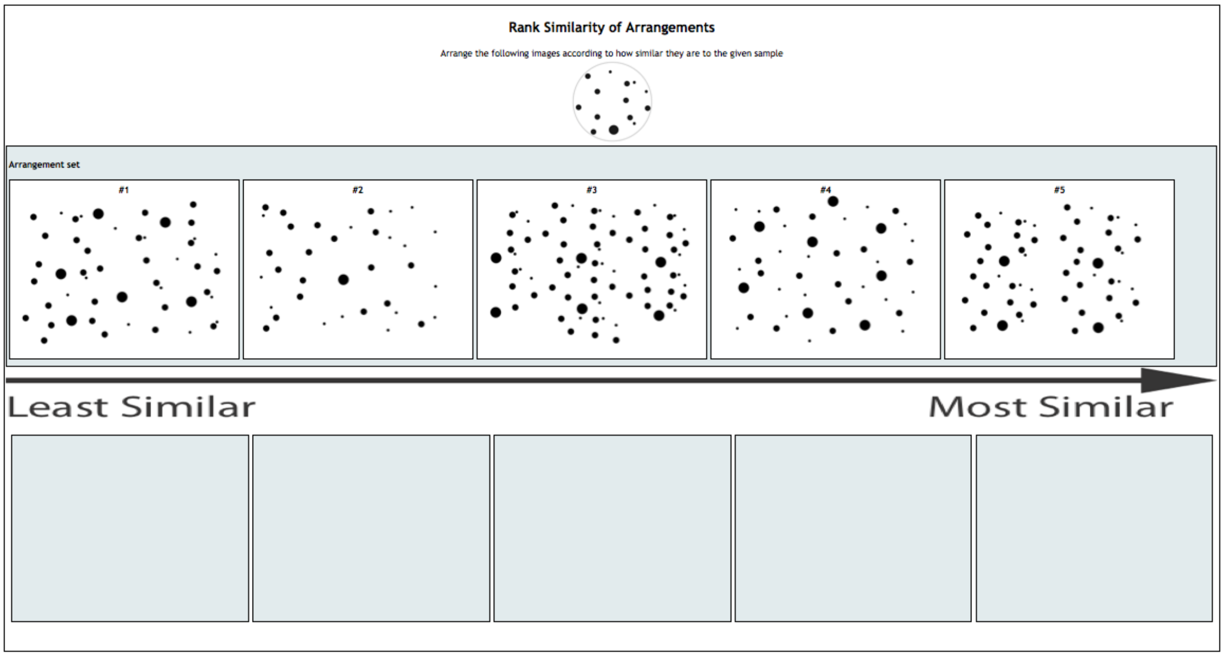


Figure 3.7: *A sample interface used during the second user study.*

Data collection and analysis: Participant interactions on the screen and their final ranks were recorded automatically; while the reasoning for the choices they made was collected in writing by myself. A mixed-method analysis was adopted to interpret the information collected.

3.4.2 Qualitative analysis

Using the same deductive reasoning as in the grounded theory analysis from the first user study, the findings show that there exist many common characteristics used by participants to judge similarity of geometric arrangements. Participants used a total of 11 properties to describe similarity. Table 3.3 summarizes the common properties reported from participants after completing the similarity rankings.

In arrangements where the stimulus pattern was most obvious, participants often reported the number of copied stimuli and the distribution style (tiled, regular, etc). When the stimulus copies were not apparent, participants reported descriptions that involved the shapes and groups present. As for the arrangements that had no obvious structures and required extra scrutiny to judge, similarity measures involved a range of the remaining properties (density, frequency of elements, and distances). Properties that were repeatedly mentioned and deemed especially influential when perceiving similarity across all arrangement styles of distribution include density, distribution type, spaces, and shapes.

Some participants rejected arrangements that did not fill the canvas. For example, participant P1 described one image saying “too much white space, large empty spaces”. Depending on noticeable accuracy of patterns and the element distribution in the generated image, participants were likely to disregard emptiness when judging similarity; for the same image P8 said that “this captures exact copies of the stimulus in it”.

3.4.3 Quantitative analysis

To understand which textures were rated the most similar and which ones were rated the least similar, I analyzed the collected ratings according to the type of approach used to generate the arrangement. The data from both the fifteen new participants and the five previous participants did not show any noticeable differences, and thus were pooled together during the analysis. The top row of Figure 3.8 shows percentages of “most” similar and “least” similar participant rankings for the complete arrangement set. This chart reveals that arrangements generated using a tiling approach were most likely to be rated by participants as similar to their original stimuli. Structured and random approaches were less likely to be rated as similar. This shows that approaches used to generate arrangements highly affect the way they are perceived.

Common similarity properties	
density	high low
clustering	pairs groups
frequency/sizes	number of elements number of different sized elements ratio of element sizes
overall pattern	discernible or not periodicity space filling
copied samples	number of copied stimuli accuracy of copied samples
distances	exact/approximate to stimulus not like sample
white space	amount
distribution type	regular/tiled irregular/random homogeneous
shape(s)	detectable/undetectable
sampling	impose circular boundary on image
symmetry	noticeable or not

Table 3.3: *Common visual properties used to judge least and most similar geometric arrangements to the stimuli upon which they were based.*

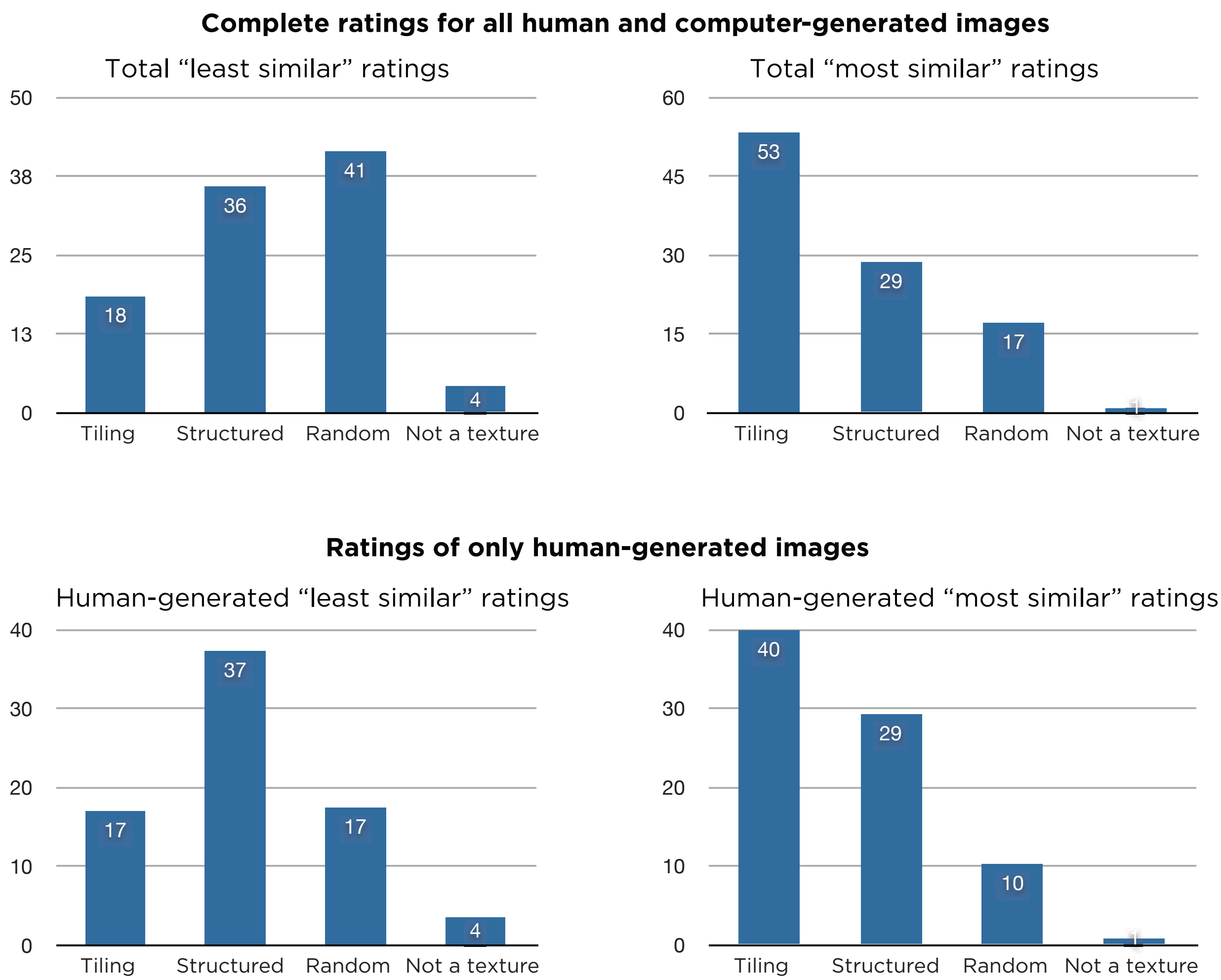


Figure 3.8: *Total most and least similar similarity ratings according to generation approaches with and without computer generated arrangements.*

Since tiling arrangements ranked highest for “most similar” arrangements and random arrangements ranked least, the perfection of computer-generated arrangements was considered to be a cause for concern. By removing all the computer-generated arrangements from the data set, a re-run of the analysis (bottom row of Figure 3.8) showed that of the remaining human-generated arrangements, the majority of “most similar” rated arrangements were still generated via a tiling approach but those “least similar” were generated from a structured approach. An interesting feature from these charts is that human-generated random textures, on the right side of Figure 3.8, rated more similar than computer-generated random ones. This hints at the possibility that people are more effective at generating "random" arrangements and also distinguishing them as more similar than completely random computer-generated ones. People may either have a keen sense for judging density of arrangements or are better at identifying non-random placements and white space within generated arrangements. This particular observation is related to the balance between regularity and randomness found in Op Art geometric arrangements [34].

The bottom left chart of Figure 3.8 shows that arrangements generated from a structured approach were rated "least similar" more often in comparison to tiling and random human-generated arrangements. This does not necessarily eliminate a structured approach as a good method for generating arrangements, since many of them were rated more similar than randomly generated ones. Detected shapes and small patterns (discussed in Section 3.4.2) may be the prime reasons for the increase in similarity ratings. Likewise, in arrangements that did not include obvious shapes or structures, global visual properties were used, hence the triumph of more convincing random distributions. More testing and evaluation are needed to shed light on the applicability of combinations of these generation approaches for future synthesis methods.

3.4.4 Results and discussion

The mixed-method analysis described above reveals a set of metrics used by people to judge similarity between arrangements. These factors also match the global and local visual factors identified in the first study (Section 3.3.2), further supporting them as reliable measures for comparing and rating similarity of 2D geometric arrangements. A subsequent quantitative analysis shows that arrangements generated using a tiling approach were ranked the highest as "most similar" out of randomly selected arrangements. Participants found it easier to detect repeated instances of a stimulus within an arrangement and were then more likely to rate it as similar.

Upon further analysis, there is an apparent hierarchy when it comes to rating similarity in geometric arrangements. Although accuracy was noticed to be a contributing factor in both user studies, people look first for complete and whole representations of the stimulus inside an arrangement. The spatial structures formed through multiple instances of stimulus patches proved to be one of the strongest measures of similarity for the observer. The second most obvious measure required the identification of themes across the arrangement. These themes consist of groups of local elements that generally form geometric shapes and are consistently distributed in the arrangement. The final measure sought after involves an overall comparison of the arrangement to its stimulus using global mathematical attributes.

Regularity, density, spacing, focal points, and ratios of element sizes and number played a notable role in rating similarity of arrangements when both spatial structures and themes were not (or minimally) spotted.

In comparing results of this study with similarity ranks in the first user study, we notice a slightly different trend. When participants were asked to generate an arrangement and then rate its similarity to the given sample, they often perceived textures generated via a structured approach as similar to the stimulus. But in the second study, arrangements generated using a structured approach had a much lower likelihood of being rated similar, while tiled arrangements ranked the highest. The perceived similarity of arrangements suggests that perhaps strict similarity is not the ultimate goal of geometric texture synthesis, and that seeking to balance between similarity and aesthetics maybe more relevant.

3.5 Conclusion and future work

In this chapter I report findings from two experimental perceptual studies on the process of generating and evaluating similarity between geometric arrangements. They include (1) identifying a set of important visual cues used by people when generating and/or comparing similarity of geometric arrangements and (2) a set of strategies (tiling, structures and random) adopted by participants when generating arrangements. These findings offer necessary preliminary steps towards grounding texture synthesis techniques more firmly with our understanding of visual perception and have not been studied in the GTS community prior or considered in previous geometric synthesis algorithms.

As with previous studies on geometric visual perception, these results are subjective and represent information from only a small group of people. However, they inspired a new methodology for geometric synthesis. In the next chapter (Chapter 4) I build on the tiling strategy to develop a new GTS algorithm that conforms to how we visualize geometric arrangements studies in this chapter. It will also be interesting to see similar extensions considered from finding of perceptual studies such as these in other areas of NPR.

It is worth noting that I have not offered a clear definition of similarity in the studies presented in this work. Each participant was left to decide upon their own criteria for similarity. Many participants in the second study selected tiled (regular and near regular) arrangements to be more similar to the samples. This itself is a interesting phenomenon for further exploration, but we should also want to understand how similarity can be viewed for non-regular textures. In the future, it would be interesting to study how choice of language used in similarity studies affects participants' interpretation of the tasks.

Another avenue for future work is to investigate arrangement regularity. Given a small sample arrangement, how do people discern the order and regularity of the larger pattern? Until now, it was difficult to explain how people may interpret a small stimulus as being part of larger regular or irregular arrangements. Based on this interpretation, they may choose different methods to generate the arrangement. Future research efforts should focus more on understanding the reasons that lie behind these choices and developing measures to account for the balance of order and regularity found in generated patterns.

These experiments also give rise to an obvious need within the texture synthesis community. Existing synthesis algorithms have no been able to reproduce a complete range of texture styles, spanning a continuum from regular to irregular. Results from these experiments may shed light on some of the commonalties that exist between different texture styles. In addition to more exhaustive experimentation, we can yield sufficient information to develop a general framework that accumulates all necessary information about all texture styles and offer it as a base for future texture synthesis algorithms.

A related goal for future work is to establish a plausible suite of benchmark samples that future algorithms can use to evaluate effectiveness. Drawing from these points, I offer some solutions to evaluation methodologies, re-implementation of GTS algorithms, and standards later in Chapter 5.

Chapter 4

Patch-based geometric texture synthesis

Computer graphics researchers have and will always be able to create ad hoc algorithms that attempt to solve geometric texture synthesis problems. However, current algorithms have become increasingly complex and involve careful manual tuning making it difficult to replicate results that are of similar quality to input exemplars.

In this chapter I adapt the idea of raster-based image quilting to develop a simple algorithm for GTS inspired by the results of the psychophysical study of human perception presented in Chapter 3. After tiling patches of geometric motifs, the algorithm suppresses repetition artifacts to achieve results that are arguably of equal quality to those of existing GTS algorithms. This algorithm is later adopted in an evaluation methodology proposed in Chapter 5, and compared with other GTS algorithm results for future insight into similarity.

4.1 Introduction

The establishment of GTS began with the work of Barla et al. [14] that proposed a geometric analogue of raster-based texture synthesis methods [128]. Geometric texture synthesis algorithms [5, 54, 56, 91] have since developed new means to compensate for the lack of expressiveness found in raster-based methods to deal with shape distributions in 2D and 3D. The problem of GTS also encompasses research on packing algorithms for non-photorealistic rendering [27, 116]. The goal of this packing is usually to distribute motifs so

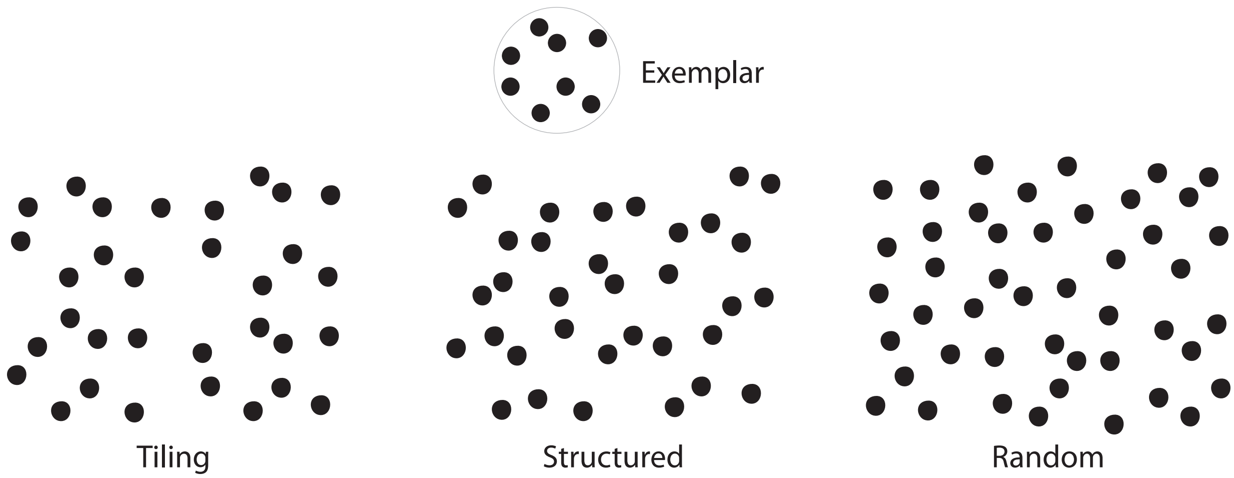


Figure 4.1: *Examples of arrangements created by participants in the study from Chapter 3. The arrangements show typical examples of the Tiling, Structured and Random strategies.*

that the space between them is minimized, or as even as possible and hence a homogeneous arrangement of motifs.

Since the work of Barla et al., researchers have proposed numerous mathematical and statistical means of achieving similar arrangements to their exemplars [5, 54, 56, 91]. It seems reasonable to draw from results discovered in the previous chapter to guide us towards a simpler and more grounded synthesis algorithm.

As described in Chapter 3, I conducted two psychophysical studies that resulted in 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 of larger arrangements given an exemplar. Typical examples of these three strategies are shown in Figure 4.1 and summarized as follows:

- **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, but the obvious repetition is a problem especially at a larger scale.

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 two extremes, but it is also the hardest to formalize as an algorithm.

Existing GTS algorithms all seem to follow a similar approach to the synthesis problem by injecting Structure into a Random process. They do this either by optimizing an initially random distribution like Barla et al. [14], Ma et al. [91] and Hurtut et al. [54], or by placing motifs one at a time as in the algorithms of Ijiri et al. [56] and Alves dos Passos et al. [5].

Based on the results from Chapter 3, I propose a complementary approach to these existing GTS algorithms by exploring a purely Tile-driven approach to the geometric synthesis process. Instead of asking what must be added to a Random distribution to increase its similarity to an exemplar, I begin with a Tiling of exemplars and ask where order might be *removed* to suppress signs of repetition.

The algorithm I present in this chapter can be viewed as a geometric analogue of patch-based approaches in raster texture synthesis [36]. It is the first algorithm in this area that is based directly on a psychophysical study of how humans respond to geometric texture arrangements. As a GTS algorithm, it has the advantage of simplicity, paving the way for a robust, interactive implementation in real-world illustration software. The ultimate goal is to have the algorithm serve as one more data point in ongoing research on similarity measures and evaluation strategies for geometric texture synthesis algorithms. A subsequent evaluation in Chapter 5 utilizes arrangements synthesized from the patch-based GTS algorithm presented here.

4.2 Patch-based geometric texture synthesis

Out of the three identified synthesis strategies discovered in the studies of Chapter 3, the Tiling strategy was most likely to produce arrangements similar to an input exemplar; this is followed by a Structured strategy. My approach is directly inspired by well known patch-based algorithms in raster texture synthesis, such as image quilting [36]. Some raster patch-based techniques have also been applied to generate thin shells of geometry around mesh surfaces [143].

The patching idea is not novel to GTS; some patching concepts have made their way into two synthesis algorithms. Alves dos Passos et al. [5] discusses copying patches purely as an optimization for efficiency. Ma et al. [91] copy multiple small random patches as an initialization strategy for their optimization, but do not consider a complete algorithm founded on patches.

Other than being viewed as a solution for enhancing speed, raster patch-based methods have many advantages including their ability to preserve texture details and similar local neighbourhoods. The algorithm I present here builds on the concept that larger sample patches with closer proximity can better ensure local neighbourhood similarity and similar global density distributions. The real challenge in this context is not to preserve the statistical properties of the sample, but to suppress obvious repetition artifacts.

I extend the image quilting ideas of Efros and Freeman [36] by patching together multiple vector sample inputs rather than patches of pixels into a larger arrangement. Like them, the synthesis process starts by placing patches side by side in a grid. I subsequently develop some ways to deal with density and repetitions to achieve consistent results.

4.2.1 The algorithm

A synthesis algorithm based purely on Tiling is trivial to implement: just stamp out regularly spaced copies of the exemplar. This clearly captures nearly all of the exemplar's statistical properties, but imposes a repetitive structure that may not be intended. To suppress these repetitions I propose a minimal set of modifications. I begin with the notion of *overlapping* tiled copies of the exemplar, and work through the consequences of that decision. In the following subsections I describe the four steps I use in the patch-based algorithm. The steps are summarized in Figure 4.2.

4.2.2 Creating an exemplar

Like Ijiri et al. [56] and Ma et al. [91], and unlike Hurtut et al. [54], I begin with the simplifying assumption that the exemplar (a *patch*) will consist of a set of non-overlapping transformed instances of a smaller number of distinct primitive shapes, denoted by $\{S_1, \ldots, S_k\}$. Limitations arising from this decision are discussed further in Section 4.3. 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 called the *input region*, usually a circle or square.

Generally, most previous GTS algorithms endow individual vector motifs in the exemplar with some kind of "anchor point", usually the motif's centre of mass or the centre of its bounding box. Arrangements are then synthesized by considering distributions of distances between these anchor points. While computationally convenient, it is possible

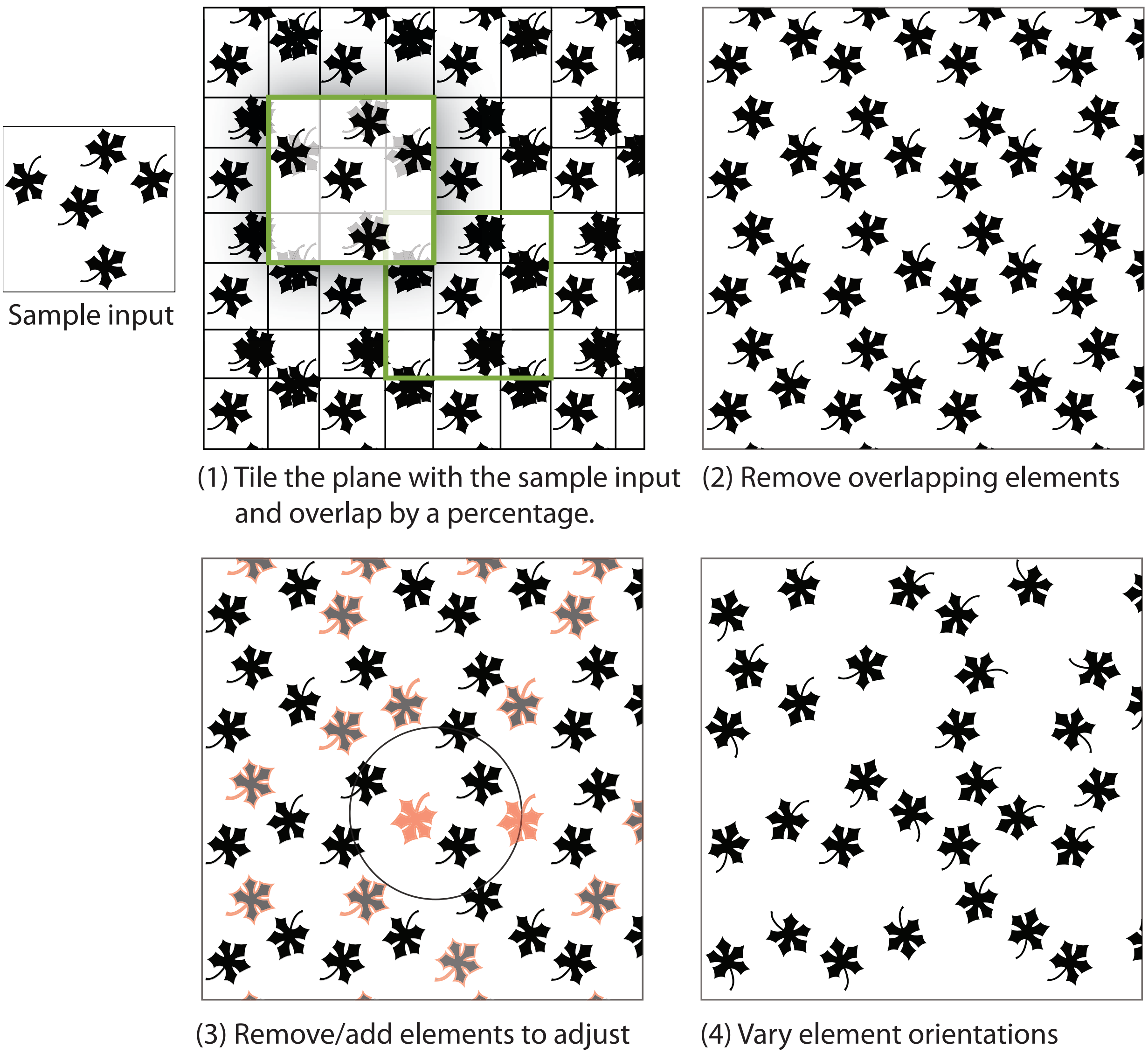


Figure 4.2: *The consecutive steps of our patch-based synthesis algorithm visualized from the initial grid layout (left) to motif orientation adjustment (right).*

that these anchor points do not accurately reflect human judgments of distance between motifs, particularly for those that are more elongated.

In the synthesized texture, an algorithm should 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. A distance between two motifs that is smaller than this minimum distance might suggest a grouping in the synthesized texture that is not apparent in the input and one of these elements has to be removed.

Therefore, distances between motifs in the exemplar are gathered first by the algorithm in a preprocessing step. For any two non-overlapping vector shapes A and B, I 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 is approximated for arbitrary shapes by converting them into polygons and measuring distances from the outlines.

The distance function is used 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. Here S_i and S_j are distinct motifs in the exemplar. In a synthesized arrangement, we can 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}. A concurrent investigation by Landes et al. [78] uses a similar approach and measures distances between the 2D motifs and extends the measure to capture distances between 3D mesh geometries. It may also be desirable depending on the exemplar pattern to measure toroidal distances between motifs from the top to bottom and left to right of the exemplar.

4.2.3 Constructing a grid

Assuming that the input region is a square of side length r, define a fractional overlap amount σ between 0 and 0.5. Through experimentation by varying grid overlaps I found that $\sigma = 0.3$ is sufficient for most exemplars. Larger overlaps tend to mis-represent the patterns used here and small overlaps result in largerer repetitive patterns that undesired. The algorithm then constructs 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 covering the entire synthesis area, the *output region*. Non-square grid layouts and non square input regions are possible as well. I have 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. To suppress global regularity while preserving local structure the algorithm is able to randomly rotate the exemplar copies. This is done using 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 4.5(b), the motifs have a clear horizontal orientation. To avoid this problem I rotate each *motif* in this copy by $-\theta$, returning it to its original orientation.

It is fair to note that regularity is less obvious for isotropic arrangements. Reasonable results can be achieved regardless of whether tiles are oriented or not. The synthesized arrangements in Figure 4.5, for example, do not include tile orientation.

4.2.4 Resolving overlaps

The immediate consequence of allowing copies of the exemplar to overlap above is that individual motifs may overlap in the synthesized arrangement. This leaves the synthesized arrangement with noticeable crowding around the areas of tile overlaps. This problem can be resolved by removing motifs.

During synthesis, the algorithm searches the arrangement created above for pairs of overlapping motifs using a straightforward polygonal path intersect detection method.[1] Removing either motif will resolve the overlap. To improve the quality of the output, I implement an additional heuristic: if either motif is found to violate the distance rule relative to its neighbours, it is selected for removal.

By removing motifs after having synthesized too many, the 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. A result from finishing this step shows the resemblances, see Figure 4.3.

4.2.5 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 as illustrated above. However, because overlap is a purely local operation between pairs of motifs, the arrangement's

[1] "Determining if a point lies on the interior of a polygon" Solution 2(2D): http://paulbourke.net/geometry/polygonmesh/

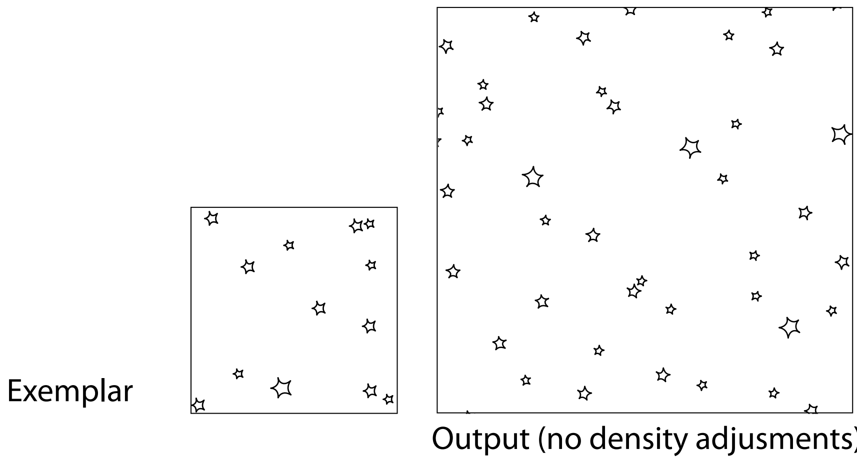


Figure 4.3: *An initial patch-based result after removing overlapping motifs.*

density might differ too much, both globally and locally, from that of the exemplar. To fix this I propose an iterative adjustment step that attempts to restore the approximate desired density.

The local density of an arrangement within any region of the plane is measured 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. When overlaying the input region anywhere on the synthesized arrangement, it is desired that the density within that window should be close to ρ.

Discrepancies in the arrangement density can be minimized by adding or removing motifs as necessary. If the local density is too high, the algorithm removes motifs to lower it, again favouring motifs that violate the distance rule. If it is too low, the algorithm searches for the largest empty disc contained in the window, and inserts contents of a congruent disc superimposed at random over the exemplar. If any of the added motifs violate the distance rule to other motifs in the arrangement, they are rejected and others are sought. I apply this process iteratively until the density of the synthesized arrangement is sufficiently close to that of the exemplar.

4.2.6 Varying orientations

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

During the pre-processing step the algorithm computes the smallest arc of the circle (representing all possible orientations) that contains the orientations of all instances of each motif, and samples 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 those in Figure 4.5(b) avoid undesirable variations in orientation. A more fine-grained approach might sample from narrow intervals around the orientations that occur in the exemplar.

4.3 Results and discussion

Overall the patch-based algorithm produces satisfactory results across a range of irregular exemplars. Figure 4.4 gives purely subjective comparisons between results from this patch algorithm and those by Hurtut et al. [54] and Alves dos Passos et al. [5].

When placed next to other algorithms, the patch-based results are very competitive. In some cases they clearly outperform other algorithms. This demonstrates how easily repetition artifacts can be suppressed when slightly overlapping a Tiling-based layout of motifs, inducing a small amount of rotation, and a few additions or deletions of motifs. The patch-based arrangements were synthesized automatically by setting the overlap σ to 0.3 and approximating the sample density. In some instances, where specified, arrangements were generated using a hexagonal grid. These arrangements are noticeably similar to their square grid counterparts.

All the arrangements synthesized by this algorithm took a matter of seconds, using a standalone C++ implementation without user intervention. I chose to demonstrate the algorithm on textures that are typically irregular. It may also be possible to apply this technique for regular textures. But for such cases, a simpler approach based on straightforward tiling without overlaps is more likely to be adequate.

Like other synthesis techniques, my patch-based 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. The algorithm cannot handle an exemplar in which every motif is a distinct shape. For example, given an exemplar with one instance each of ten different coloured flowers, a grid overlap would cause flowers at either side of the patch to be overlapped and possibly removed. This would result in an arrangement with the wrong amount of flower ratios represented in exemplar. In such

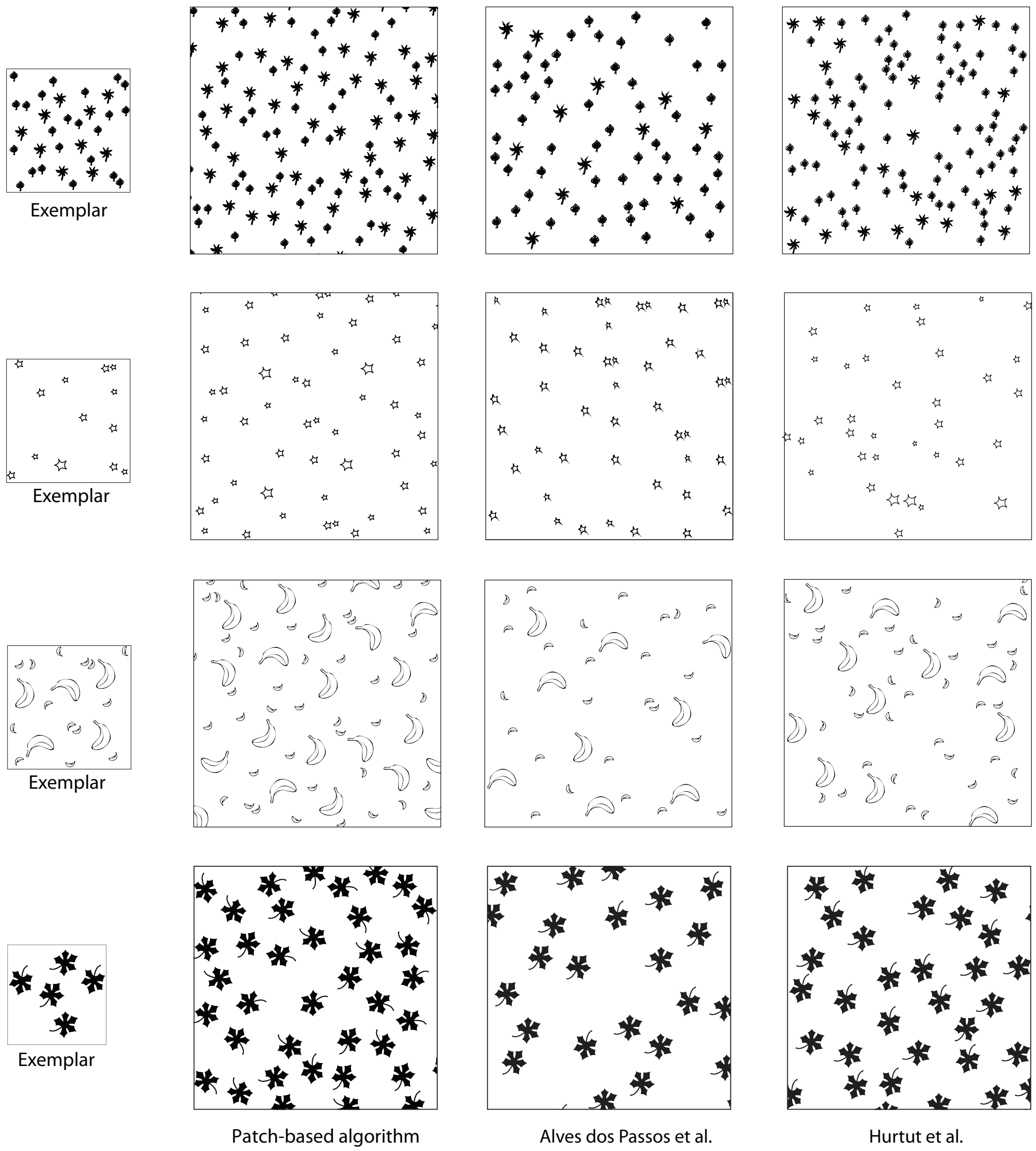


Figure 4.4: *Patch-based synthesis results based on three exemplars that appeared in the papers of Hurtut et al. [54] and Alves dos Passos et al. [5].* © *2009,2010 Hurtut et al. [54], Passos et al. [5], used with permission.*

cases, one option would be to explore a motif categorization step similar to the one used by Hurtut et al. [54]. The algorithm could then select instances at random 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. I discuss this in greater detail in future work (Chapter 7, Section 7.3).

This patching algorithm 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 that exhibit structured colour variations. I have also 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.

Raster-based patching methods are limited in the range of texture styles they are able to synthesize successfully. This limitation is also evident in the patch-based GTS extension, which works well for irregular arrangement styles. However, attempting to synthesize a near-regular arrangement with this algorithm will cause the arrangement to lose its global structure. One way to mitigate this problem may be to create sample inputs based on Wang tiles [70] and synthesize them on a grid without any overlaps.

4.4 Cartography and GTS

One particularly interesting domain for geometric texture synthesis is digital cartography, especially its application to geology. Mapmakers regularly fill regions by hand 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 [61, 113]. Hence, cartographic examples can form a rich, real-world set of inputs for current and future GTS algorithms.

The US Geological Survey (USGS) published a standard reference for geological map symbols in 2006 [126]. The exemplars in Figure 4.5 were adapted from this dataset, preprocessed and run through my patch-based algorithm to illustrate how GTS can be applied to help fill cartographic maps. The rows of Figure 4.5 show the synthesized results based on square and hexagonal tile layouts. In Chapter 5, I investigate the evaluation of GTS results and gather arrangements synthesized from different algorithms using this same exemplar set.

Current cartographic software uses a simplistic periodic tiling approach to generate illustrative earth terrain.[2] A quick attempt at filling polygonal shapes in software such as Adobe®

[2]MAPublisher: http://www.avenza.com/mapublisher

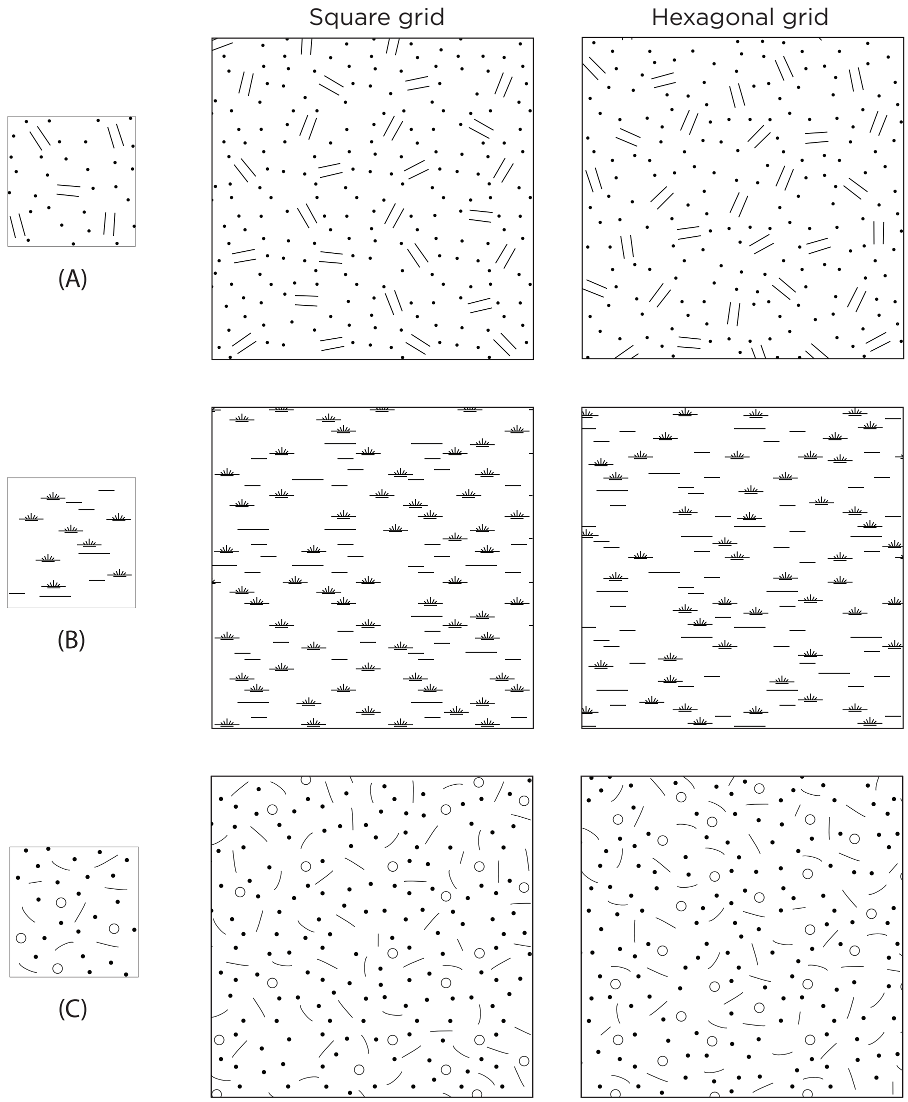


Figure 4.5: *Patch-based synthesis results based on three different exemplars adapted from the cartographic USGS database mentioned in Section 4.4. The exemplars are synthesized using our algorithm once with a square tile grid and once using a hexagonal one.*

Illustrator® will also reveal inherent repetitiveness across the area if it is large enough. Another problem is evident in the unnatural texture found at region boundaries. Resolving either of these problems could immensely improve cartographers' digital experiences.

The patch-based algorithm I propose in this chapter provides a trivial solution for the texture repetition problem. As for boundaries, synthesized textures often show chipped elements that are not desired by cartographers. Some manual solutions have been attempted, but are complex and require expertise in the software itself. Ideally, we want to adjust synthesized arrangements to respect boundaries of map regions, internal curves, or other labels and markings. A representative example is shown in Figure 4.6. Instead of clipping the trees at borders or under labels, a synthesis algorithm should find ways to delete or relocate motifs to improve the overall appearance. Investigating astute ways to automatically resolve motif placements within irregular borders is still an open problem. More details on this problem are described in Section 7.1.

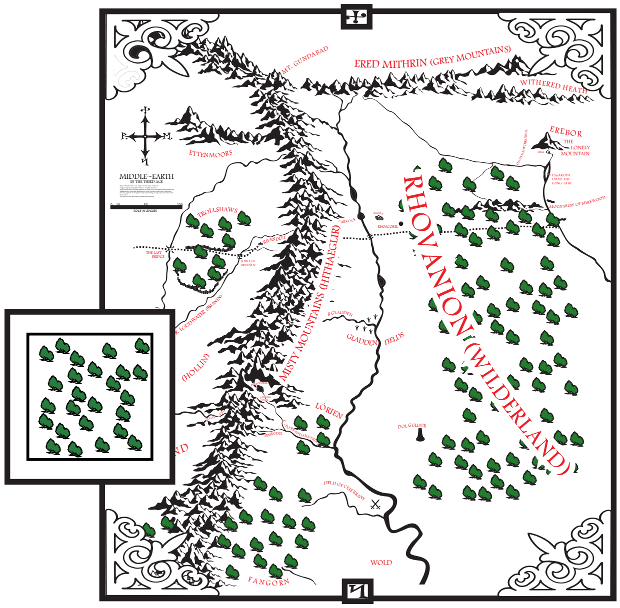


Figure 4.6: *This map of Middle Earth shows portions of forests synthesized using our method (replacing pre-existing forest texture). Used with permission from* http://www.lords-of-blah.nl/mearth/mearthmap.html

4.5 Conclusion and future work

In this chapter I present a simple example-based geometric texture synthesis algorithm inspired by the results of my psychophysical study of human perception presented in Chapter 3. The approach is based on suppressing repetition artifacts in regularly spaced copies of an exemplar through a series of additions, deletions and rotations of motifs. During synthesis, randomized selections integrated within the patch-based algorithm ensures that resulting synthesized arrangements are unique. I demonstrated the algorithm with exemplars derived from previous algorithm papers and standardized textures from geological maps. These textures are typically stochastic. In the next few paragraphs I discuss a number of possible future work ideas to overcome inherent limitations of this algorithm.

While developing an Adobe® Illustrator® plugin for the algorithm, it was apparent that re-running the synthesis algorithm every time a user decide to change the bounding geometry was inconvenient in respect to motif randomness. This could be desirable depending on the context of its application, however there is value in attempting an alternative deterministic approach. A potential algorithm driven by blue noise for example can result in large non-repeating geometric arrangements. If further enhanced with a Wang tile layout [70], such a method would dominate existing GTS algorithms as a resilient alternative to existing synthesis approaches.

As described in the previous chapter, some arrangements may be intended to be irregular, and others as regular. There is no simple automatic method for extracting global distribution styles from a small sample input. This is a difficult perceptual problem in general, I hope to research possible means for addressing it in the context of Geometric Texture Synthesis and offer some initial steps in this direction in the next chapter.

In addition to finding ways to deal with unique elements, consider the synthesis of geometric arrangements with a defined boundary. Suppose we want to fill an area with an arrangement surrounded by specific border motifs. The GTS algorithm by Ma et al. [91] synthesizes homogeneous arrangements within bounding volumes, but they do not consider different elements at the borders. In future work, Chapter 7 Section 7.1, I discuss possible solutions to this problem in more detail.

A final problem is the subjectivity involved when perceiving similarity between more than one synthesized arrangement, as presented in Figure 4.4. When shown to a group of people, many will not agree that any one arrangement is the most similar. In the Chapter 5 I offer an evaluation methodology designed to assess GTS results to reduce this inherent subjectivity. However, this is only one of the many possible attempts to understand human perception of GTS and similarity between its arrangements.

Chapter 5

Towards effective evaluation of geometric texture arrangements

Despite having plenty of attractive and visually interesting interpretations of realistic data, NPR has always suffered from a dearth of evaluations to establish the validity of algorithms. For the specific case of GTS, in Chapter 3 we have come to realize that judging visual similarity between synthesized 2D texture arrangements is fundamentally subjective. With insufficient visual conventions describing how geometric texture arrangements are actually perceived by humans, a wide range of synthesis approaches have been proposed but none have been effectively evaluated [5, 14, 54, 56, 87, 91].

In this chapter, I present a methodology for effective evaluations of GTS algorithm results. I start by establishing a geometric texture synthesis database gathered from multiple synthesis sources, then use the dataset arrangements in two psychophysical user studies to assess how well the different sources did in comparison to one another.

5.1 Introduction

In the past decade we have seen an increase of applying formal evaluation methods in the validation of new algorithms in non-photorealistic rendering (NPR), but this trend has not caught on in the field of GTS. Many GTS algorithms have been proposed, all of which seem to produce reasonable results across a range of inputs. But at best, authors run their algorithm on an exemplar from a previous paper by others, and show the old and new outputs side by side. I believe that there is a need for effective evaluation strategies in GTS, that can be applied to compare existing algorithms and validate new ones. Accordingly, my high-level goal in this chapter is to establish a practical evaluation methodology for GTS algorithms.

In Section 2.3 I list some of the most relevant previous work on evaluation in NPR and texture synthesis. The latest inquiry into suitable evaluation methods is the GTS similarity investigation I presented in Chapter 3. In it I conducted the first study that probed the nature of similarity in the perception of geometric textures. The investigation resulted in a descriptive list of visual features that people use to explain similarity between synthesized arrangements and exemplars. These studies offer the first step in NPR literature towards understanding geometric arrangements in light of human visual perception.

Building on that investigation, I attempt to push our understanding of texture similarity even further. In this chapter I gather the first comprehensive dataset of geometric textures (Section 5.2) from several different *synthesis sources*: expert human designers, state-of-the-art synthesis algorithms, and simple randomly generated textures. I then conduct two user studies based on this dataset (Sections 5.4–5.5), to see whether human judgments of similarity between synthesized textures and exemplars can be used to assess the performance of different synthesis sources. Using results from the studies I attempt a small evaluation on geometric texture synthesis algorithms (Section 5.6).

5.2 A geometric texture benchmark

To allow for more effective comparisons of GTS algorithms I collect a dataset of synthesized and hand-drawn arrangements. My goal is to use this collection as a benchmark for evaluating existing and future GTS algorithms; to further elucidate the meaning of "similarity" in the context of geometric textures; and to determine the progress and shortcomings of geometric texture synthesis as a research area. All these goals are addressed in the remainder of this chapter.

To select sample inputs, I chose to adapt four source arrangements from the US Geological Survey (USGS) Digital Cartographic Standard for Geologic Map Symbolization [126]. This resource contains textures used to indicate different features in geological maps. Jenny et al. [61] designed a tool that helps cartographers fill maps with similar features. Similar artificial textures have also been used as input by recent GTS algorithms [5], making the dataset a suitable candidate for future experimentation.

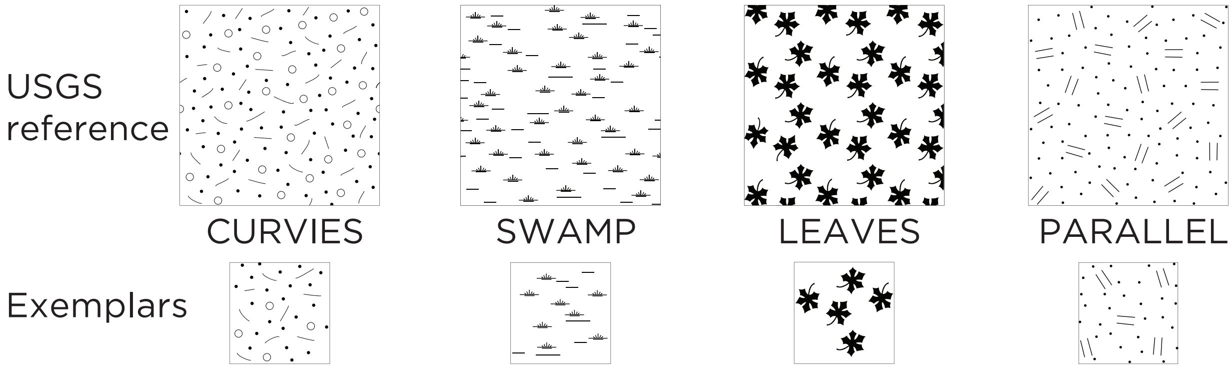


Figure 5.1: *The original source arrangements and the extracted (pre-processed) exemplars.*

I identified four distinct *patterns* in the USGS standard that use relatively few distinct motif shapes and that I will take to be representative in this chapter. As shown in Figure 5.1, I name them CURVIES, SWAMP, LEAVES and PARALLEL. From each of the references I extract a smaller exemplar to use in my studies. Slight modifications when doing this include removing cropped elements at borders of the extracted exemplar and representing all elements in SVG using symbols and use tags as described in Chapter 4.

Armed with these four exemplars, I set about collecting a diverse set of actual arrangements constructed from them. For each exemplar I gathered a set of eleven arrangement results from three sources: human experts (Section 5.2.1), existing GTS algorithms (Section 5.2.2), and a simple pseudorandom approach (Section 5.2.3). In following subsections I describe the process involved in collecting this data.

5.2.1 Arrangement collection from expert designers

To compare fairly between computer-generated and hand-drawn arrangements, I recruited expert human designers to draw large arrangements from the four exemplars. Human designers have a keen eye for texture, composition, layout and design, and can provide a rich set of subjective interpretations of the synthesis task. I found experts by word of mouth and by advertising on a forum for expert users of vector illustration software. Participants were required to have extensive experience in their field and keen aesthetic judgement. A total of four people qualified for the study.

Hereinafter I identify them and their arrangements as **H1**–**H4** (Table 5.1). I subsequently collected new sets of synthesized results from two other expert designers, **H5** and **H6**.

Their arrangements are included in Appendix C.

ID	Title	Years of expertise
H1	Visual Artist	9
H2	Technical Trainer and Consultant	12
H3	Illustrator / Graphics Artist	30
H4	Illustrator / Technical Artist	10+

Table 5.1: *Expert identifications and area of expertise.*

To collect human-generated arrangements I created a self-contained template in the form of an Adobe® Illustrator® document. A copy of this document is shown in Figure 5.2. The template describes the synthesis task as follows: "Given a small sample of arranged symbols, place copies of the symbols into a large area so that the overall impression of the larger arrangement is like the smaller one". Below that, the template includes a completed example for user reference and a set of restrictions. Experts were asked to (1) not create arrangements that repeated the exemplar exactly (2) use only the symbols that appear next to the exemplar and (3) only rotate the motifs when needed (scaling, shearing, and reflecting were not permitted). Four empty regions appear below, one for each of the USGS exemplars. Next to each region is a copy of the exemplar, and copies of symbols for the distinct motif shapes used in that exemplar. Participants were asked to drag and place motifs into the empty regions to synthesize their arrangements. Each participant received an information letter to sign, the template (in both PDF and Adobe® Illustrator® formats) and a questionnaire. Each of the participants was compensated with a gift card for their efforts.

The final submitted results can be seen in Figure 5.3. It is evident from these arrangements that they vary substantially between the experts. Each of the expert designers clearly had a *style* when generating their versions of the arrangements (see all four patterns in relation to the others). This could be as a result of their creativity or from the limited perspective of similarity expressed in the task given to them. One of the many factors to mention here is density. For example, **H2** synthesized all four arrangements with a high density, while **H1** synthesized them all with lower density. The subjectivity inherent in expert interpretations is a major concern for GTS developers. It is worth pondering on whether any GTS algorithm will be able to effectively capture what users intend when given an exemplar because of the ambiguity involved with the number of elements.

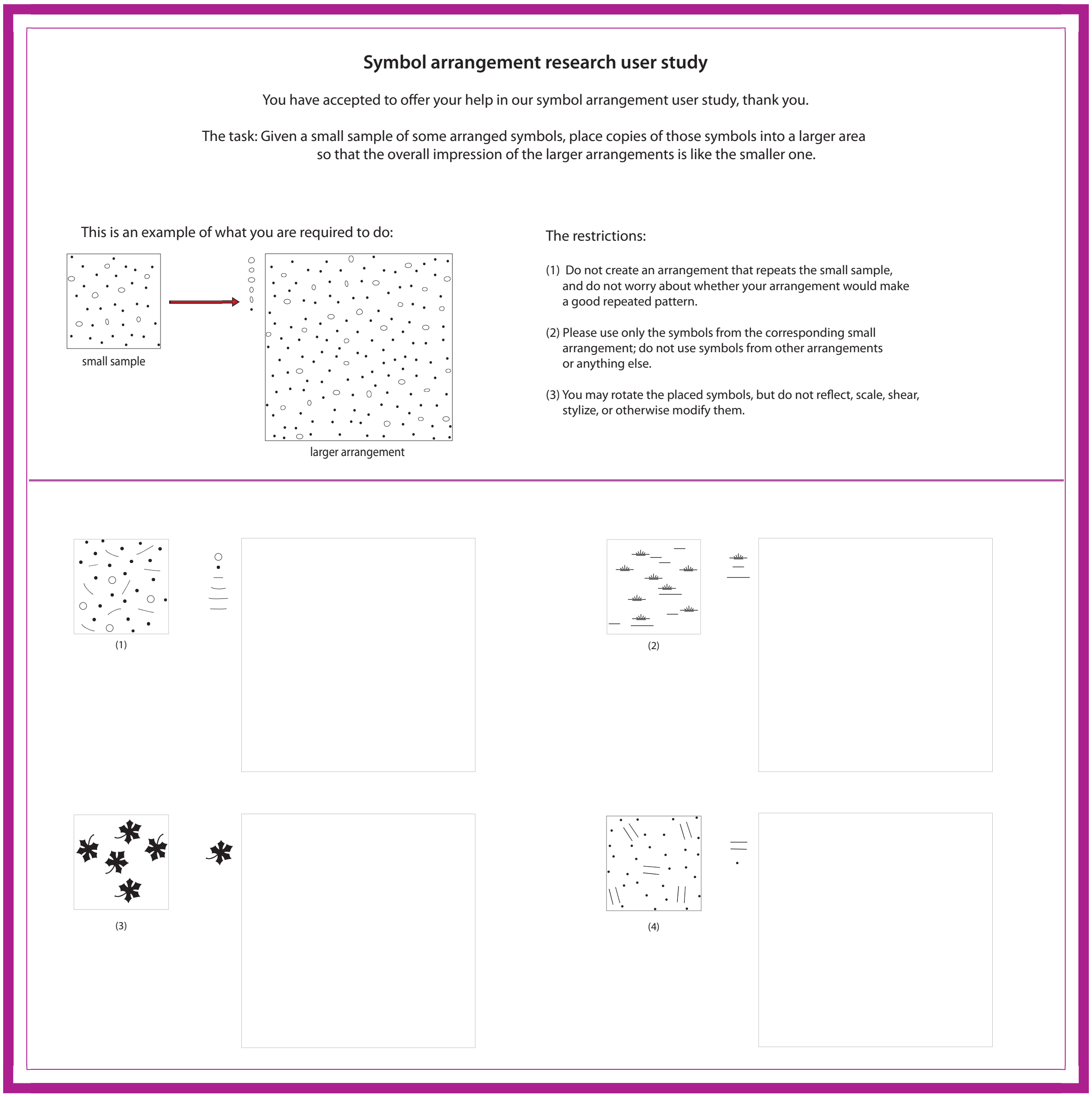


Figure 5.2: *The template used to acquire arrangements from designers.*

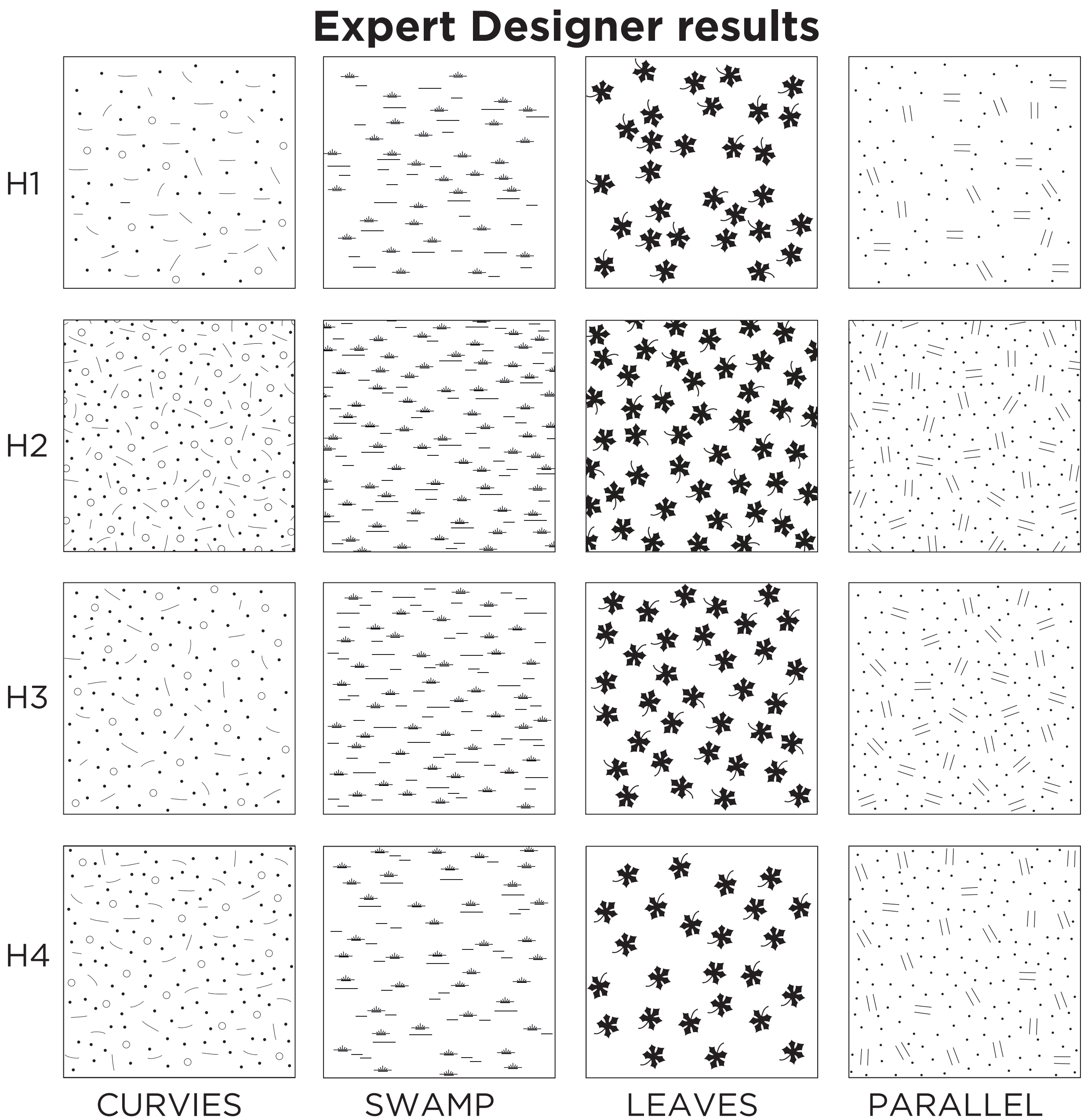


Figure 5.3: *Results gathered from our expert designers.*

5.2.2 Arrangement collection from GTS algorithms

One problem with attempting a robust evaluation of existing GTS algorithms is the difficulty of acquiring and developing the actual implementations. Reimplementing existing synthesis algorithms is difficult because they often include ad hoc fine tuning. Without the expertise of the original creators of these algorithms, their true value can be obscured. As described by Lin et al. [85], to make comparison results valid, it is important to use the original algorithms to synthesize new arrangements.

ID	Algorithm	ID	Algorithm
A1	Alves dos Passos et al. [5]	**R1**	Pseudorandom (Section 5.2.3)
A2	Hurtut et al. [54]		
A3	Ma et al. [91]	**R2**	Pseudorandom (Section 5.2.3)
A4	Patch-based GTS (Chapter 4)		
A5	Patch-based GTS (Chapter 4)		

Table 5.2: *Algorithm labels and their corresponding authors.*

To gather valid arrangements, I contacted the GTS authors of a spectrum of synthesis approaches [5, 54, 91]. The four exemplars were sent to each of the authors via email, in the format required by their algorithms. They subsequently synthesized larger arrangements while adhering to the same criteria used when generating their previously published arrangements and sent them back electronically. Their results are shown in Figure 5.4 and referred to as **A1**–**A5** as shown in Table 5.2. Synthesis sources **A4** and **A5** are both synthesized from the patch-based GTS algorithm presented in Chapter 4; they were generated using square and hexagonal arrangements of tiles, respectively.

To enable the gathering of arrangements by the above algorithm I was obliged to support the individual practices of each algorithm, as each had different input requirements. Some algorithms required text files with point locations and IDs of motifs, while others required specific vector formats. It would be easier to compare GTS algorithms if the community were to agree on a common input standard. I recommended the simple SVG-based format; the one in Chapter 4 which would be ideal for this.

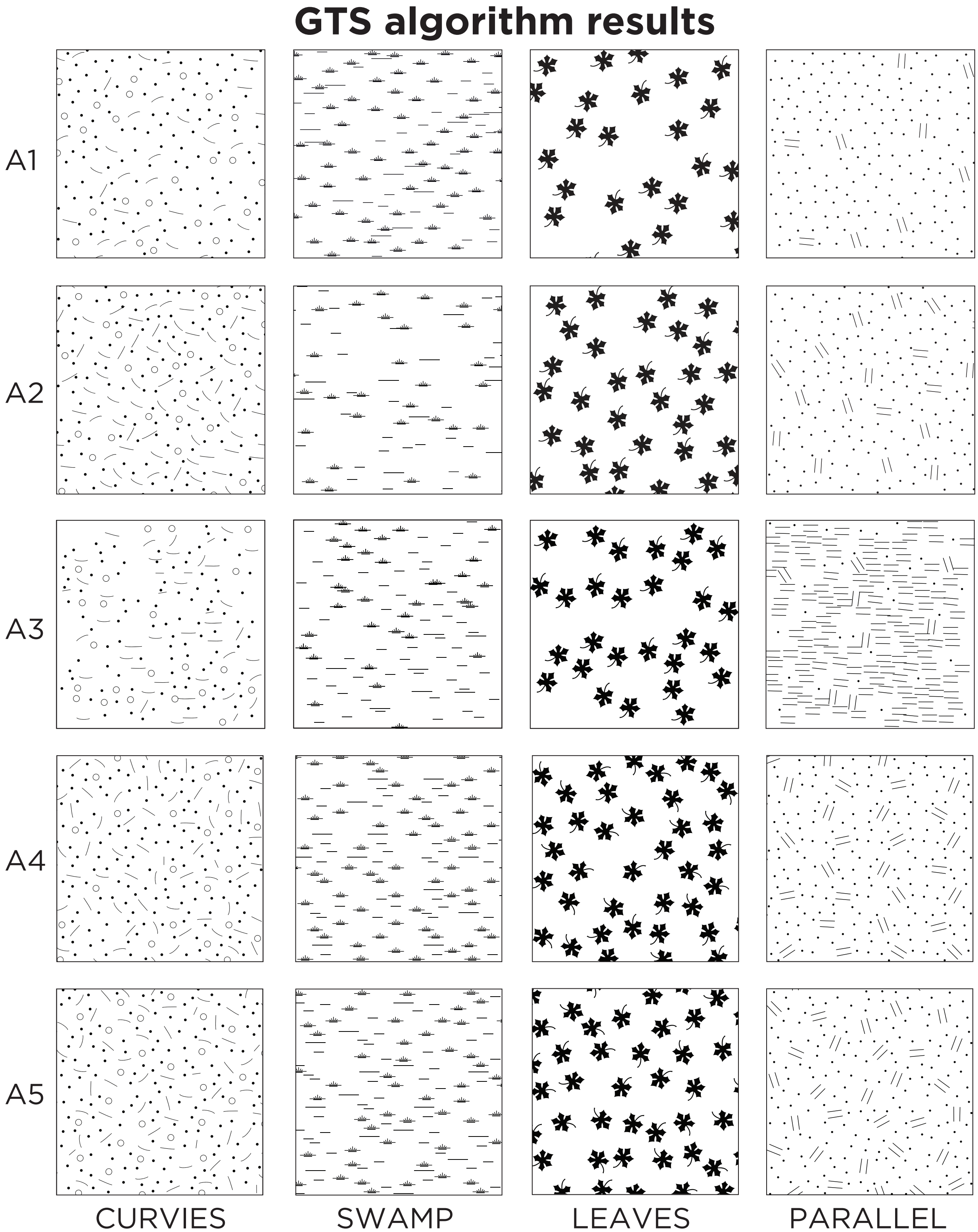


Figure 5.4: *Results gathered from the GTS algorithms.*

5.2.3 Pseudorandom texture arrangements

To test whether random arrangements would be perceived differently from the results of other synthesis sources, I include two pseudorandom arrangements per exemplar in the dataset. I developed a simple randomized synthesis algorithm and used it to generate arrangements labelled **R1** and **R2** shown in Figure 5.5.

I let d refer to the minimum distance between centroids of motifs in the exemplar, and let ρ be the density of the exemplar, i.e., the fraction of the exemplar covered by motifs. First I choose a random point P within the synthesis region, and a random motif from the exemplar to place there. Then I perform two tests on this proposed motif placement:

- If the distance from this point to any other placed motif centroid is less than d, then reject P.
- Center a window on P with the same shape as the exemplar. If the density of the synthesized arrangement within the window exceeds ρ, then reject P.

If the point P passes these tests, then place the chosen motif there. I then iterate this process until the overall density of the synthesized arrangement comes within a threshold of ρ.

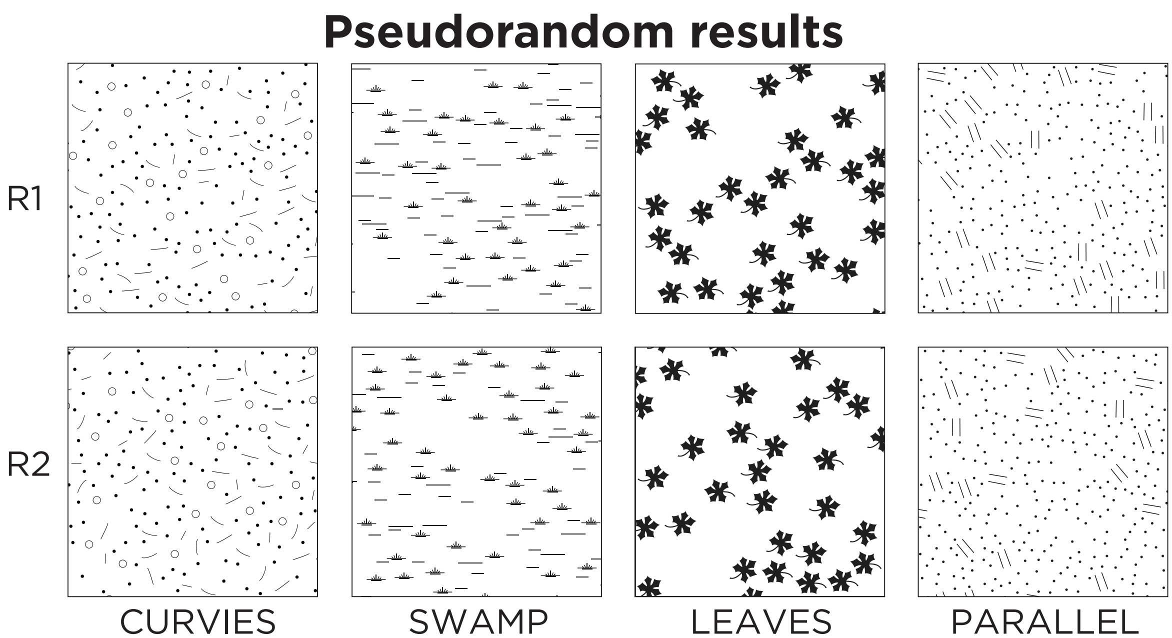


Figure 5.5: *Results gathered from the pseudorandom algorithm.*

5.3 Evaluating synthesized arrangements

After collecting this dataset I now want to effectively compare all the synthesis sources in the benchmark of synthesized texture arrangements from Figures 5.3, 5.4 and 5.5. I do this by moving beyond the subjective practices currently used, and explore a more effective study-based methodology that supports rigorous investigations into perceptual similarity. Insights gathered from these investigations should help guide researchers towards better practices for evaluating GTS algorithms.

My investigation is divided into two parts. In the first study I conduct an observational pile-sorting study and watch how human subjects sort arrangements based on their similarity using printed cards on a flat surface (Section 5.4). In the second study I conduct a pairwise comparison test using a computer interface. Participants are given pairs of synthesized arrangements and asked to click on the arrangement they believe is most similar to an exemplar (Section 5.5).

For both of the studies presented in the next two section I recruited 20 university students (undergraduate and graduate). They had no previous experience with geometric textures and did not take part in my earlier studies (Chapter 3). All participants were compensated with gift cards for their efforts.

5.4 Pile-sorting synthesized arrangements

Pile-sorting is effective for gathering qualitative data such as user observations. It can also be supported by systematic data collection through short semi-structured interviews [131]. It is particularly suitable when there are few quantifiable measures suitable for analyzing the target material.

This qualitative style of analysis was previously used by Isenberg et al. [59] to understand how people judge similarity between hand drawn and computer generated pen-and-ink drawings. I use a similar approach but this time to compare between multiple synthesized arrangements and their sources.

Card preparation: I created 44 cards from the GTS dataset, eleven for each of the four patterns. Each had the arrangement printed and glued to a 12cm $\times$ 12cm square of cardstock. I also created a card of the same size for each exemplar, printed at 6cm $\times$ 6cm inside a black border.

Setup: Participants were asked to sit on a chair in front of a large flat table surface. The exemplar source was placed approximately 100 cm away, as shown in Figure 5.6.

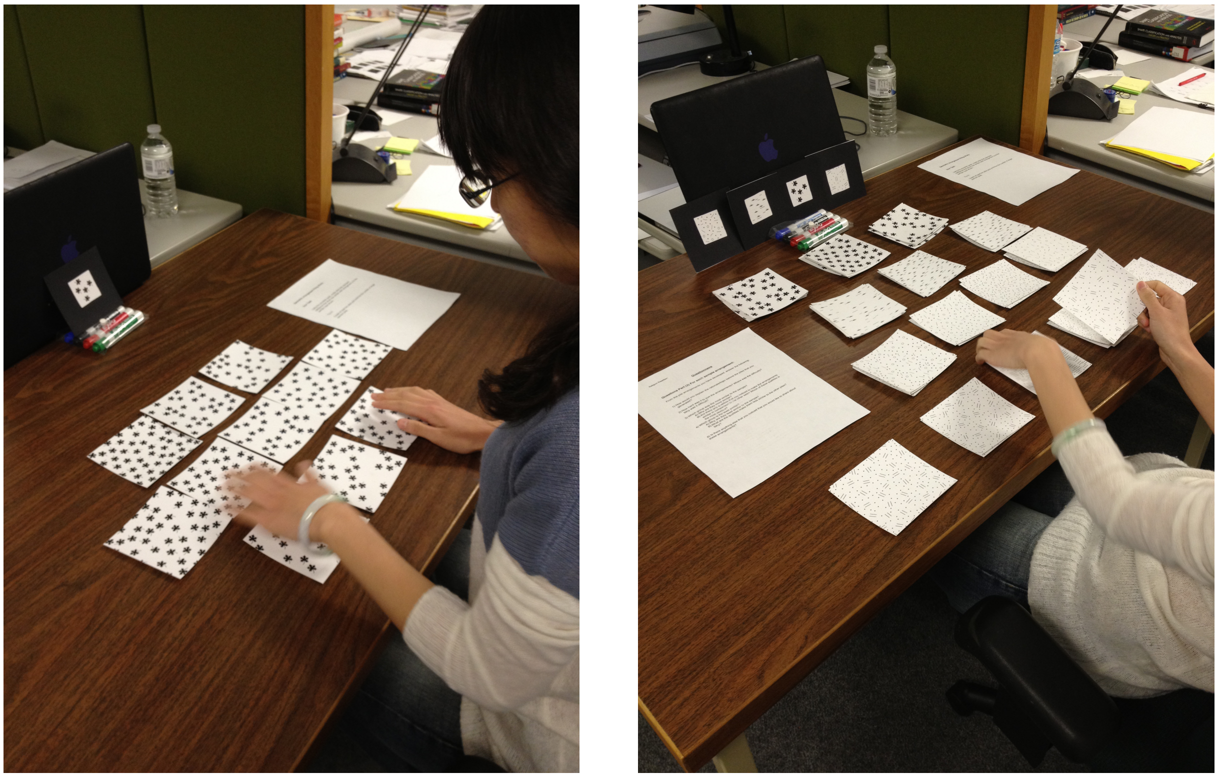


Figure 5.6: *The setup for the pile sorting study with a participant distributing the 11 piece card set while seated at a distance from the exemplar source (left). After sorting all arrangement into piles the participant discusses them with the investigator (right).*

Because synthesized arrangements depend so strongly on their exemplars, and because of the diversity of arrangements for each exemplar, I opted to show the exemplar cards as a reference during the pile sorting study.

At the beginning of the study I provided the participants with a set of cards and instructed them to read the task provided to them on a white sheet of paper, ask questions, and begin when ready. The sorting task was described as follows: "Using the provided cards, create piles that represent categories that show how similar each arrangement is to the sample input shown".

The methodology: I adopt an unconstrained pile-sorting task in which participants could make as many piles as they wanted without any time restrictions. They were encouraged to provide their thoughts during and after the study. To ensure that enough data is collected for comparisons I suggested that participants create at least two piles and minimize the number of singleton piles when possible.

At the start of the study I provided participants with a randomly chosen card set (either CURVIES, SWAMP, LEAVES, or PARALLEL) and let them generate piles using their own criteria. Most participants distributed the cards across the table before piling them, making it easier to notice differences and similarities between the cards. Once they completed piling the first card set, the piles were pushed to the side of the table and the participants were handed a card set with a different pattern. This was repeated for each card set. Participants created an average of four piles with a standard deviation of one for each card set.

For the interview, the piled cards were moved closer to the participant in the same order they were presented. After that I initiated the discussion by handing the participant a sheet of paper containing some questions.

Data collection: I recorded the resulting piles of arrangements via note taking. During the pile sorting task and semi-structured interview, participants were audio recorded. The pile sorting task took an average of 14 minutes in total, while the semi-structured interview and discussions that followed took an average of 8 minutes.

In the following subsections I analyze the results of the pile-sorting experiment broken into two parts. I first analyze the generated piles according to the four source patterns separately (Section 5.4.1) and then I analyze the data according to the synthesis sources (Section 5.4.2). These are followed by a summary of findings gathered from participant interviews (Section 5.4.3).

5.4.1 Pile-sorting according to arrangement patterns

To understand the resulting piles I created a similarity matrix for each participant's piling of each of the patterns. Similarity matrices are created by tabulating the co-occurrences of synthesized arrangements found in each pile. If a participant grouped cards from two synthesis sources into a pile, I place a 1 in the corresponding matrix entry; otherwise I place a 0 there.

For each pattern, I combine the similarity matrices for all twenty participants, as shown in the tables of Figure 5.7. In the combined matrices, each entry represents the number of participants who placed a combination of sources into the same pile. Higher scores in these tables imply that the arrangements share similar characteristics, while lower scores imply dissimilarity.

CURVIES

	H1	H2	H3	H4	A1	A2	A3	A4	A5	R1	R2
H1	20	0	1	2	4	2	9	0	0	1	1
H2		20	3	5	1	11	1	8	16	4	6
H3			20	9	10	4	4	2	2	5	5
H4				20	6	3	4	5	4	4	7
A1					20	3	5	5	2	7	4
A2						20	2	9	8	4	6
A3							20	1	1	5	2
A4								20	9	6	7
A5									20	5	7
R1										20	9
R2											20
Most	4	1	7	8	8	2	1	6	3	7	6
Least	9	4	1	3	2	4	12	3	5	4	4

SWAMP

	H1	H2	H3	H4	A1	A2	A3	A4	A5	R1	R2
H1	20	1	0	6	0	13	5	1	2	6	6
H2		20	15	2	16	0	0	16	9	0	2
H3			20	3	13	0	1	14	10	3	4
H4				20	2	3	7	4	2	8	5
A1					20	1	1	13	9	0	3
A2						20	5	1	3	8	6
A3							20	1	3	11	9
A4								20	10	2	2
A5									20	3	5
R1										20	10
R2											20
Most	11	2	3	7	1	7	8	5	8	6	7
Least	1	16	13	2	16	3	2	11	6	1	2

LEAVES

	H1	H2	H3	H4	A1	A2	A3	A4	A5	R1	R2
H1	20	1	2	7	3	4	10	4	2	6	11
H2		20	14	1	3	6	2	7	16	1	2
H3			20	2	1	6	2	10	11	2	1
H4				20	7	8	9	3	2	5	5
A1					20	4	5	2	2	7	6
A2						20	5	11	4	5	5
A3							20	2	3	9	5
A4								20	8	5	3
A5									20	2	2
R1										20	11
R2											20
Most	10	2	3	6	10	1	8	2	2	5	10
Least	2	15	11	2	3	4	1	4	12	6	6

PARALLEL

	H1	H2	H3	H4	A1	A2	A3	A4	A5	R1	R2
H1	20	2	2	4	7	5	1	2	2	2	3
H2		20	14	6	1	6	1	16	15	5	0
H3			20	9	1	8	1	16	15	2	0
H4				20	5	17	0	6	9	3	3
A1					20	6	2	1	1	7	10
A2						20	0	7	9	2	2
A3							20	0	0	0	2
A4								20	15	2	1
A5									20	2	0
R1										20	14
R2											20
Most	3	10	9	8	4	9	0	9	11	5	4
Least	3	2	2	2	2	1	16	0	1	2	2

Figure 5.7: *Correlations showing the number of times arrangement patterns were grouped together. Pairings that occurred ten or more times are highlighted in red. Each table is labelled with the corresponding pattern name. The two rows at the bottom of each table indicate the number of participants who placed a given synthesis source into their Most similar or Least similar piles.*

Once pile-sorting was complete, I asked participants to indicate which piles of cards were most and least similar to the exemplar; answers are tabulated in the bottom two rows of each table in Figure 5.7. I discuss the reasons behind participants' choices in more detail in Section 5.4.3.

In the analysis I found that some synthesis source correlations varied from one pattern to another, suggesting that similarities differed depending on the patterns. In SWAMP, for example, arrangements by **H2** and **A1** had the highest correlation while in CURVIES they were less correlated.

Common trends found in all patterns include high correlations between **H2** and **H3**, suggesting that these two experts recognized and used similar features in constructing their arrangements. Various low correlations amongst the four designer results highlight the subjectivity problem present in GTS research.

Interestingly, arrangements by **H2** and **H3** correlate highly with arrangements by **A4** and **A5** and with some arrangements by **A1**. This consistency implies that similar pattern characteristics were featured by these sources.

Even though I found high scores between **H2** and **H3** and between **H2** and **A2**, none were considered most similar to the exemplar. However, arrangements by **H3**, **A1** and **H4** had lower correlations but were chosen to be more similar. Comparable observations for the remaining tables suggest that the piling decisions are not random and that participant similarity judgements were unambiguous.

Pseudorandom LEAVES arrangements were chosen as most similar by ten participants. These similarity choices may have been influenced by this algorithm's strong emphasis on achieving the same density as the exemplar. This is an important observation that demands further investigation into the significance of pseudorandom algorithms and density for GTS.

Arrangements CURVIES and PARALLEL by **A3** stood out as least similar in the study. These arrangements are less uniform and contain different motif ratios to those present in the exemplar, which explain participant decisions. The nature of the **A3** algorithm may possibly not account for such variations.

5.4.2 Pile-sorting according to synthesis sources

The previous analysis provides an overview of common groupings that occurred when participants compared synthesized arrangements according to the four patterns. To get a more general intuition of the piles independent of the patterns I analyze the data in terms of the number of participants found to have piled synthesis source arrangements together.

	H1	H2	H3	H4	A1	A2	A3	A4	A5	R1	R2
H1	20	4	4	14	13	15	15	6	4	12	13
H2		20	19	11	16	15	4	19	19	8	9
H3			20	14	15	13	7	19	20	9	11
H4				20	13	17	16	9	13	12	12
A1					20	11	11	14	13	16	13
A2						20	8	16	15	11	15
A3							20	6	6	16	14
A4								20	20	10	10
A5									20	10	13
R1										20	18
R2											20

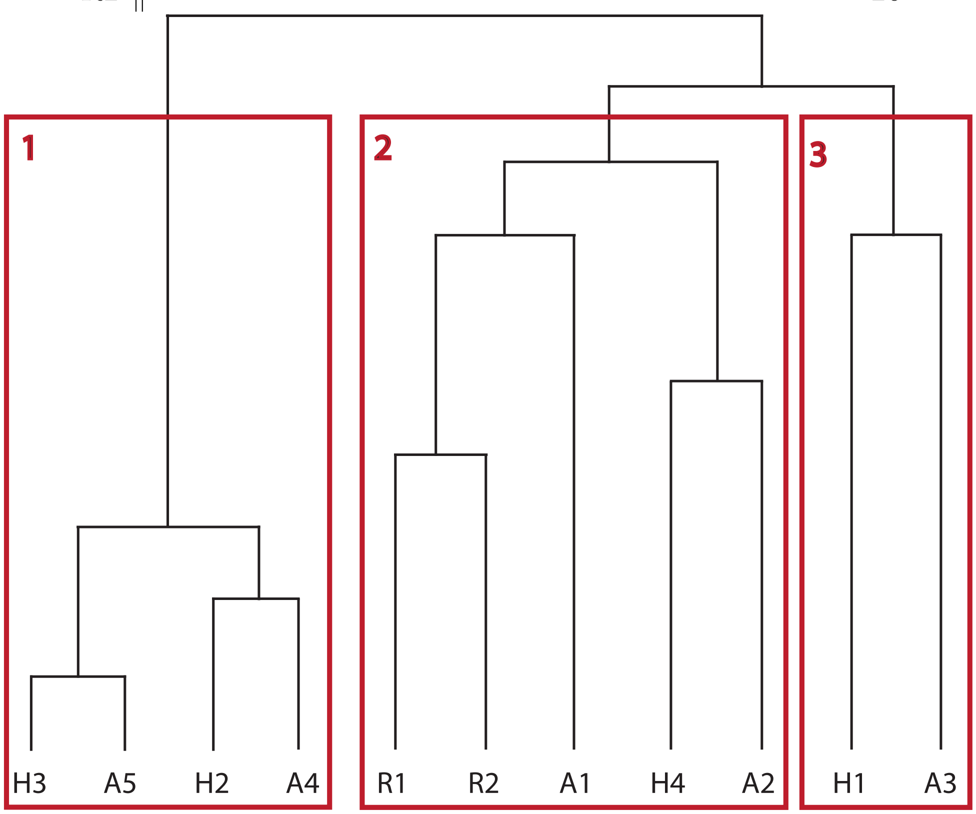


Figure 5.8: *Top: The pile-sorting correlation table shows the number of participants who have piled cards of arrangement sets together at least once. The highest correlation scores are highlighted in red. Bottom: A 2D dendrogram showing the cluster results of a hierarchical clustering analysis of sorting piles from the study based on the table.*

The goal is to highlight consistencies in the data and explain similarities and differences between the synthesis sources. A table of participant choices along with an accompanying visualization—a 2D dendrogram—is shown in Figure 5.8.

A dendrogram visualization is the result of hierarchical clustering performed on pairwise distances calculated from the table data. This descriptive analysis method is common for interpreting values found in similarity matrices [131], and has been used in research to visualize and explore relationships of large high-dimensional data set in various fields including bioinformatics [69]. To measure the dissimilarity between every two sources of synthesis I find the chi-squared measures using pair average linkage [131].

The dendrogram shows how arrangements along the x-axis merge and divide. Along the y-axis we see how far apart the merging happens. Linked arrangements near the bottom of the y-axis imply frequent placements of arrangements in one pile. Those linked higher up the y-axis and farther apart are found together less often, hence less consistent. The linkages result in three clusters of arrangement sets which are derived purely from participants' similarity choices.

For example, in Figure 5.8 **H3** and **A5** were piled together by all twenty participants, so they are connected low on the y-axis. In contrast, **A3** and **H2** were piled together by only four participants, leading to a linkage high on the y-axis. In the following points, we discuss the contents of each cluster:

- **Cluster 1**: In this cluster there are four synthesis sources: **H2**, **H3**, **A4** and **A5**. Perceptual characteristics captured by all these sources result in a larger number of co-placements by a majority of the participants. This cluster differentiates these four sources from the rest of the synthesis sources in terms of their appearance.

- **Cluster 2**: This cluster contains five synthesis sources: **H4**, **A1**, **A2**, **R1** and **R2**. Notice that arrangements by **R1** and **R2** are consistently correlated by many participants, as are **H4** and **A2**. This shows that participants are meticulous at deciphering commonalities between arrangements causing them to distinguish arrangements by **R1** and **R2** as coming from a similar source. The same observation applies for **H4** and **A2**.

- **Cluster 3**: This cluster contains two synthesis sources: **H1** and **A3**. The linkage between these sources is higher up the y axis, implying that they are less consistent than their neighbouring sources. Although a total of 15 participants were found to pile arrangement from these sources together, the linkage suggests that some patterns could be correlated more than others. This is true for arrangement patterns CURVIES and LEAVES in the tables of Figure 5.7.

In summary, synthesis sources **H2** and **H3**, **A4** and **A5**, and **R1** and **R2** were more consistent in achieving higher similarity correlations with one another than other sources. The pseudorandom sources **R1** and **R2** are successfully distinguishable as originating from the same source, so are sources **A4** and **A5**. Arrangements by other GTS synthesis sources are harder to distinguish as coming from a similar source. The different patterns used for this study may have influenced these findings. For example, the dissimilarity between synthesis sources **H1** and **A3** was evident for two of the patterns.

5.4.3 Semi-structured interview

Once participants finished sorting all the card sets, the piles were brought back and placed across the table in four rows in front of them (Figure 5.6) with a sheet containing some questions. I decided to leave questioning until after the pile-sorting task was complete to eliminate biases when piling subsequent patterns.

I asked three open-ended questions targeting the thoughts and decisions participants made during the study. The qualitative information gathered from this interview helps elucidate the visual factors participants felt important when depicting similarity as well as their overall confidence during card sorting. I repeated the same questions, in order, four times for each participant (once per pattern). The answers to these questions are discussed in the three subsections that follow.

How would you explain the rationale or logic behind the piles that you generated?

Over the course of the pile-sorting study, I observed participants use different sorting criteria. The criteria reported are summarized in order from most to least common in Table 5.3.

Of the 20 participants, 19 singled out density as one of the main factors they used for sorting the cards. This observation is consistent with my previous GTS studies in Chapter 3, in which density was identified as a crucial visual cue in texture perception. Variation in ratios of distinct motif shapes also influenced some participants in their decisions to group them separately. This was apparent for arrangement patterns PARALLEL, CURVIES and SWAMP but not for LEAVES which had only one motif. I believe that we need to understand the importance of density and motif ratios in these similarity judgements and give a small analysis of density from the pile-sorting results in Section 5.4.4.

Twelve participants mentioned the identification of noticeable patterns in exemplars. This involved either holding the card out near the exemplar and deciding whether it was a good extension to the small sample or locating small groups of motifs distributed in ways similar to groups in the exemplar.

Orientation cues were used occasionally, particularly when sorting the LEAVES and CURVIES cards. The leaf motifs in the LEAVES exemplar exhibit only three orientations, which six participants interpreted as significant. Lines in CURVIES appeared to have a principal orientation in some arrangements, which also influenced participant judgements. This behaviour was not noticed with SWAMP. Some arrangements were explicitly sorted according to how regular and chaotic their distributions appeared. Given that all arrangements are irregular/stochastic, a regular appearance did not connote similarity.

From analyzing visual cues used for each of the four patterns, I noticed that participants did not use distance between motifs as a measure of similarity for the CURVIES patterns. Since CURVIES had the largest number of different motif shapes, participants were more inclined to look at densities and motif distribution rather than local distances. This is an important finding since many GTS algorithms focus on distances between motifs to achieve similar distributions in their results.

Rationale	PARALLEL	CURVIES	SWAMP	LEAVES	Any
Density	10	15	15	17	19
Motif ratios	11	11	8	0	16
Patterns	5	5	5	3	12
Orientation	4	7	0	6	11
White space	4	3	5	5	11
Sparsity	1	3	3	5	10
Regularity	2	6	2	3	6
Distances	3	0	1	2	4

Table 5.3: *The rationale for similarity sorting and number of participants that used them.*

How hard was it to sort the arrangements? Where was the difficulty? What was difficult?

In general, participants claimed that the study was not difficult and was in fact rather enjoyable. In some instances participants had difficulties sorting certain patterns. We report these below. Note that some participants had difficulties with more than one pattern.

Seven participants stated that the CURVIES arrangements were hard to sort into piles and five thought that SWAMP arrangements were hard to sort. In these cases, participants noted that it was harder to compare arrangements that had more than two motifs. It was easier for participants to judge similarity by comparing densities and motif ratios than by looking at local distances.

Two participants found the LEAVES arrangements hard to sort. One of them believed that having only one motif type in the arrangement made comparing them hard, while the other found it difficult to explicitly match the orientations of the leaves to the exemplar angles. Only one participant in the study mentioned that PARALLEL was difficult to sort, stating that the density was hard to estimate. All the participants who indicated a difficulty spent some extra time sorting the cards but successfully completed the task.

Which pile is the most/least similar to the sample and why?

After choosing the most and least similar piles for each pattern set, participants were asked to provide the reasons for their decisions. Their answers were concise. In addition to the criteria observed when sorting the piles (Table 5.3), participants indicated the following as contributors to their similarity decisions: repetition of the source pattern, groups of motifs, broken motifs at the borders, and overlapping motifs.

To visualize which synthesis sources were chosen as more similar most often, I tabulated participant selections as shown in Figure 5.9. In the analysis, I divide the most and least results into three groups according to the percentage range they fall into (0–50%, 51–75% and 76–100%). The figure illustrates that synthesized arrangements perceived as most similar to their exemplars (as indicated by the bars on the left of the figure) had a correspondingly lower chance of being chosen as least similar (as shown on the right). Synthesis sources that are consistent with this observation are not discussed here.

In the 76–100% range of the most similar list, we find synthesis sources **R2**, **H1**, **H4**, **R1** and **A5**. The interesting observation here lies in the fact that arrangements generated by GTS algorithms are rarely selected as most similar. Despite all efforts made to develop more compelling GTS algorithms, there clearly exist missing pieces to the synthesis problem that need to be addressed. Note that the pseudorandom sources were frequently rated as being most similar to the exemplar. A closer investigation into the relevance of pseudorandom methods for irregular GTS may help us understand what is missing. And perhaps adopting such approaches for GTS synthesis is worthwhile for irregular arrangements.

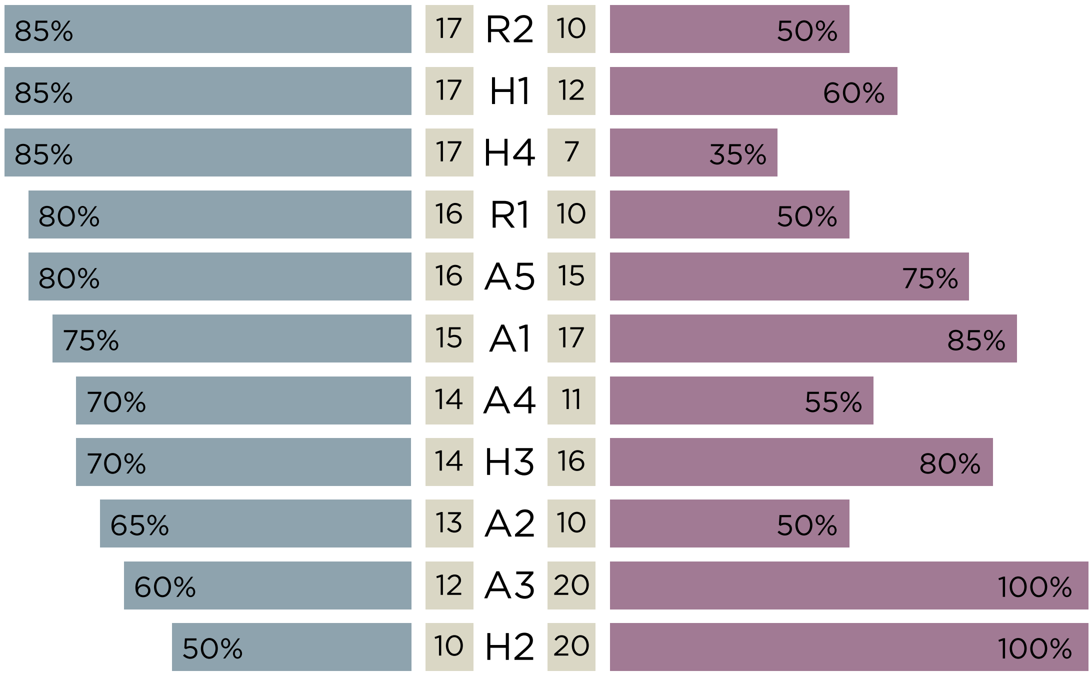


Figure 5.9: *Percentages of participant ratings of synthesis sources as most (grey) or least (purple) similar to the exemplar.*

Source **A2** was chosen as most or least similar a small number of times, indicating that participant choices were less consistent for this source. This source was found more similar in some instances for certain patterns, implying that the algorithm was better at reproducing the features of the exemplar in those cases (See the tables in Figure 5.7). Synthesis sources **A4**, **A5** and **A1** were selected as most similar approximately the same number of times they were selected as least similar. This finding suggests that regardless of the arrangement pattern similarity ratings were consistent, giving us a first hint of how to effectively determine dominance between algorithms.

5.4.4 Density of GTS

From the findings of the pile-sorting and participant replies to the interview questions, density appeared to be one of the major factors in similarity decisions. To find out how much density played a role in the pile-sorting task, I perform hierarchical clustering on each of the tables in Figure 5.7 and show the 2D dendrogram results along with the source

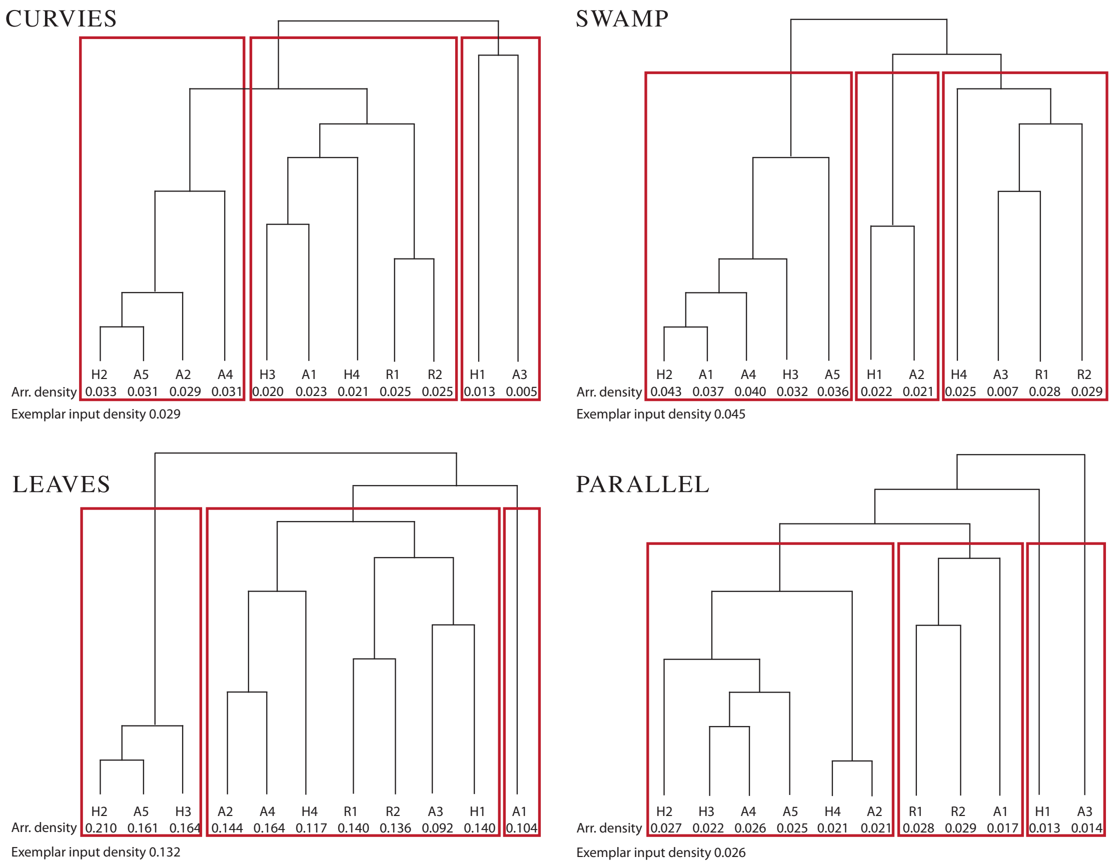


Figure 5.10: *Density measures according to the arrangement patterns (*CURVIES*,* SWAMP*,* LEAVES*, or* PARALLEL*).*

densities in Figure 5.10. The densities are calculated by adding the geometric areas of all the motifs that intersect a region, and dividing by the regions area.

Overall, there are notable density preference patterns in the observed clusterings. In patterns CURVIES, LEAVES and SWAMP, density measures are clearly separated within the hierarchal bounding clusters (in red). This suggests that people were clearly piling these arrangements based on accurate intrinsic density estimations most of the time. Participants did mention that density was used more to judge CURVIES and LEAVES, however many of them must have subconsciously estimated similarity based on density (as well as others) for the SWAMP arrangements, too.

Only the arrangement pattern PARALLEL differed in the way piles were generated. Here we find that although similar in density, **R1** and **R2** were clustered separate to sources by the designer experts and all but one GTS algorithm. This is most likely attributed to spatial and motif ratio deviations from other source results and exemplars.

These findings are again consistent with observations collected during my investigations with GTS (Section 5.4.3) supporting the claim that density is a major factor to consider when designing effective GTS algorithms. Additionally, adopting an analysis method such as pile-sorting is convenient for discovering patterns of perceptual phenomena in geometric arrangements. In Chapter 6 and Appendix D, I list other quantitative measures that could be compared using similar dendrograms as the one shown in Figure 5.10 to explore results.

5.4.5 Summary of findings

Based on the study and the subsequent analysis, we have come to a better understanding of the distinctive nature of geometric arrangements and the synthesis sources that made them. The pile-sorting study I adopted led to (1) validating a set of visual cues proposed in earlier perceptual studies of Chapter 3, and (2) provided a strategy for classifying multiple geometric arrangements based on similarity.

The main observations from the analysis of the pile-sorting data include (1) different synthesis sources correlated with one another highly, (2) pseudorandom synthesis of irregular arrangements effectively captures characteristics of irregular arrangements and could act as an alternative to GTS algorithms, (3) none of the GTS algorithms provided good results for all the arrangement patterns, and (4) amongst other factors, it is essential to effectively capture accurate densities of exemplars to achieve similar results.

I recognize that the findings are based upon a limited investigation of a small number of patterns, and do not claim that they are the last word on the relative merits of these synthesis sources. Adding more synthesized arrangements, participants and algorithmic sources to the study may reveal different results. In the same way, adding more arrangements by expert designers can benefit the whole study experience by providing a wider set of varying interpretations based on judgements of aesthetics and structure.

5.5 Pairwise comparisons of geometric texture arrangements

In the study described in the previous section I observed participants sort multiple card sets based on their similarities to an exemplar. To determine whether or not these findings are genuinely reproducible, I conducted a second psychophysical experiment. This time I asked participants to choose the most similar arrangement from a randomly presented pair.

The goal here is to look for patterns in participant choices under brief presentation of the arrangements. I intend to show that these choices are consistent to the ones found in the previous pile-sorting study. Discovering similar patterns will demonstrate that both pile-sorting and pairwise comparison studies are effective for evaluating similarity in GTS results.

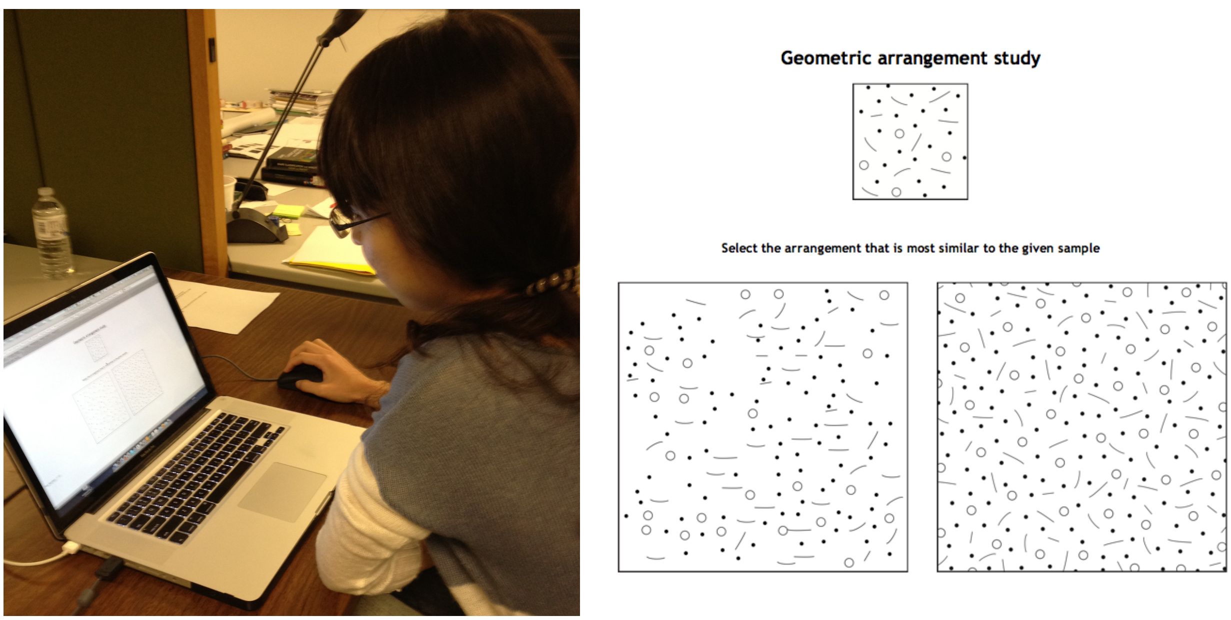


Figure 5.11: *The study setup and comparison interface.*

5.5.1 Design and setup

Sample arrangement set: I study the same synthesized arrangements as in the first study: four sets of patterns each containing 11 synthesized arrangements (Figures 5.3, 5.4 and 5.5). Pairwise combinations of these arrangements result in a total of 440 comparisons, 110 for each pattern.

Interface and methodology: Participants were seated on a chair positioned beside a table with a laptop computer. The comparison interface as shown in Figure 5.11 contains one exemplar input along with two randomly selected geometric arrangements from the same pattern placed at corners of an equilateral triangle on the screen.

The task was described as follows: "Select the arrangement that is most similar to the given sample". Participants made their selection using a mouse. A trial session of twelve random comparisons was required by all participants. They were encouraged to ask questions during the trial before proceeding onto the study.

Participants were then presented with 110 comparisons from a randomly chosen pattern. Each arrangement was compared with a result from each of the other sources, and each pair was shown twice, in both left-right orders. The result was that each arrangement appeared 20 times. I discovered that the left-right positions of each pair did not significantly affect the results, and therefore used only one comparison from each such pair in the analysis. After completing this set, I closed the interface, asked if the participant wished to take a break, and then opened a new screen containing the next set of patterns. This was repeated until all four sets were presented. To eliminate the chance of participants receiving similar sequences of arrangements, all pattern sets and arrangements within the sets were randomly presented throughout the study.

Data collection and analysis: Logs of participant selections, selection times, and arrangements paired were recorded automatically. The average time it took participants to complete this part of the study was 13.5 minutes. To interpret the data I use simple quantitative analysis.

5.5.2 Quantitative analysis of comparisons

In this study I try to identify patterns in participant similarity selection ratings. First I look at the percentages of most similar arrangement ratings according to the generating sources (expert designers, GTS algorithms or pseudorandom) irrespective of the arrangement patterns.

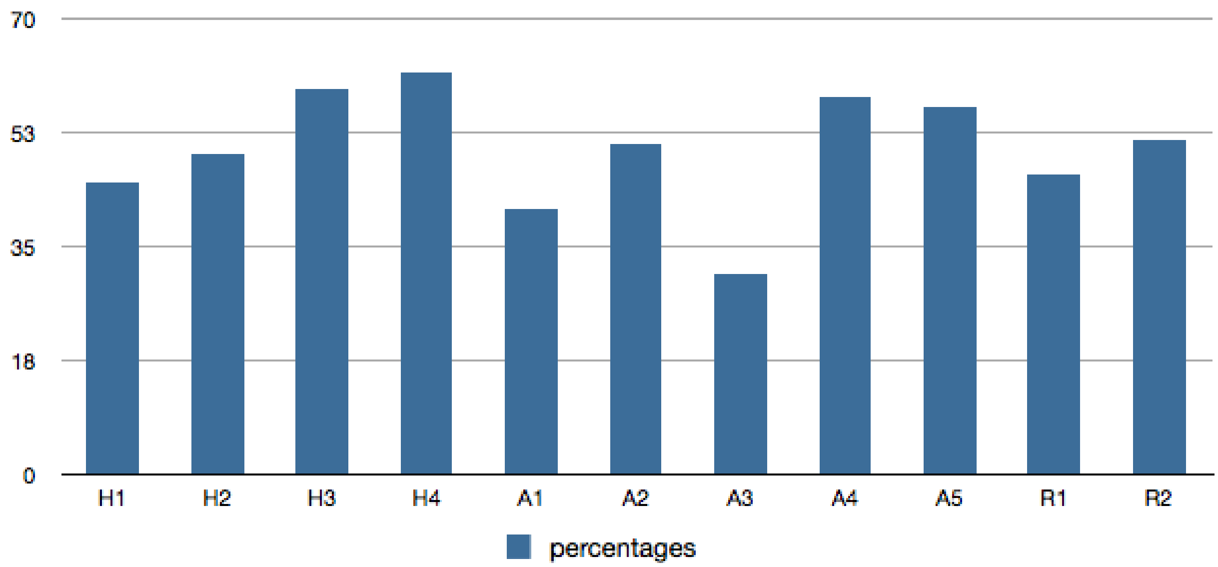


Figure 5.12: *The percentage of most similar ratings of arrangements according to the synthesis sources.*

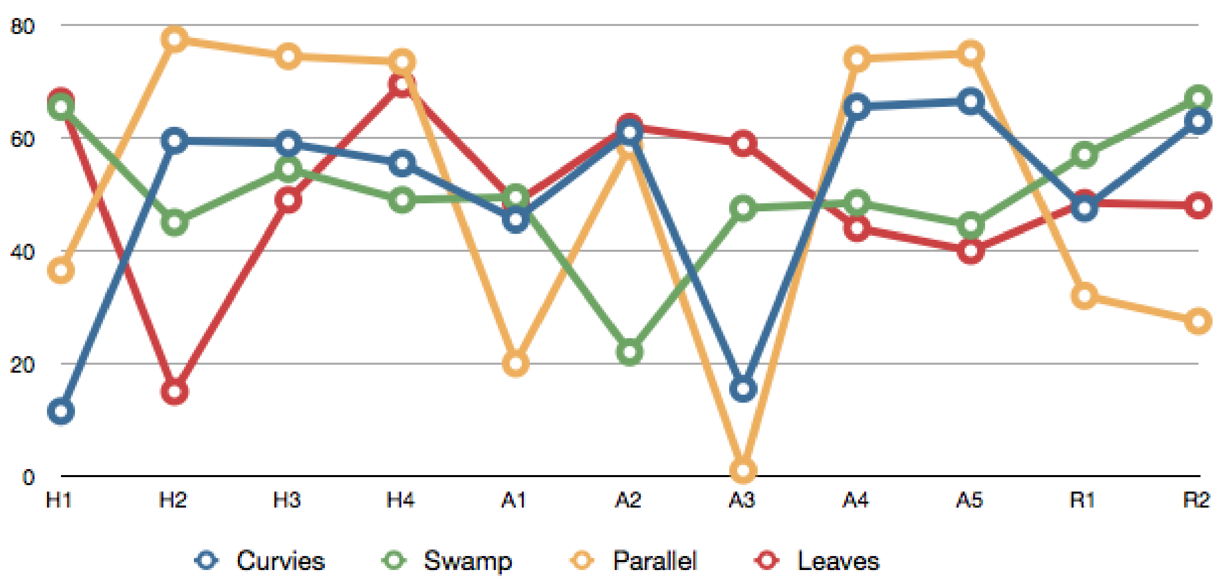


Figure 5.13: *A line chart showing the percentage of most similar ratings according to the arrangement pattern (*CURVIES*,* SWAMP*,* LEAVES*, or* PARALLEL*).*

In Figure 5.12, I find that participants were more inclined to select arrangements from all sources except **A1** and **A3** as more similar to the exemplar. Note that the least similar choices made by participants for **A1** in the pile-sorting study are more significant when presented though pairwise comparisons (see Figure 5.9). An interesting observation is that synthesis source **H2** did much better in the comparisons than in the pile-sorting study. The remaining sources performed well in this study suggesting that participants were able to compare the differences in arrangement characteristics effectively and judge the similarity quickly.

A final observation from this figure shows that **R1** and **R2** performed worse in this study than in the pile-sorting study. In the pile-sorting study both random sources performed better than all GTS algorithmic sources making them potentially more successful at synthesis. However, in this study both random sources did worse than **A2**, **A4** and **A5**. The reasons for this difference are not entirely obvious. The investigation in Section 5.4.4 rules out density as being a major factor in this dissimilarity since no big variation exists between these sources. The only other explanation could lie in the fact that the spatial distributions of motifs in **A2**, **A4** and **A5** clearly capture characteristics of the exemplar and that the random sources do not. If so, then GTS algorithms are correct in considering local relationships between motifs as a factor in understanding arrangements.

To gain insight into why sources **A1** and **A3** received low ratings, I analyzed the collected ratings according to the type of pattern used. Figure 5.13 presents a breakdown of participant similarity selections. Here, source **A1** did worse for PARALLEL and **A3** did worse for CURVIES and PARALLEL. In Section 5.4.3 I discussed the different visual cues the participants used to judge similarity; both density and motif ratios are factors in the decisions made here. To understand where the problem areas are for the remaining sources, I analyze them below according to the patterns.

In CURVIES, two of the lowest rated arrangements include sources **H1** and **A3**. This finding is consistent with the previous study and suggests that characteristics captured by these sources are different to those found in other source arrangements. For SWAMP, synthesis source **A2** had the lowest ratings, lower than those found in the previous pile-sorting analysis. The low density exhibited in the arrangements synthesized by this source appears to be more noticeable in pairwise comparisons.

Of all the LEAVES arrangements, as mentioned above, synthesis source **H2** was least likely to be chosen as similar to the source. This result separates **H2** from **H3** and **A5**, though the three were highly correlated in the pile-sorting study. Participants were more likely to select arrangements that had lower densities and avoided overly dense ones as in **H2**.

The PARALLEL arrangements show that sources **H1**, **A1**, **A3**, **R1** and **R2** were more likely to be chosen as least similar than the other synthesis sources. This observation is also consistent with findings in the pile sorting study. The patterns found for the two pseudorandom source arrangements reveal that there is a difference even between two arrangements generated by the same source. This could be a coincidence attributable to the random number generator. Determining any statistical significance here would require generating multiple arrangements, testing them, and averaging the most similar choices.

Synthesis sources **A4** and **A5** were more consistent in their ratings regardless of the pattern. The same sources also had neutral ratings in the pile-sorting study. They achieve average standing in comparison to the other sources, not always the best but never the worst.

I did not ask participants to comment on this part of the study. But the results show that participants prefer arrangements that appear to match the exemplar density. For example, I notice that in pile-sorting, **H2**, **H3**, **A4** and **A5** were often described as dense and were chosen as least similar more often than others. However in the pairwise comparisons, **H3**, **H4**, **A4** and **A5** are selected as similar to the exemplars more often indicating that density cues may be overlooked if paired with arrangements that are very different from the exemplar.

From this comparison study, I conclude that no single source of geometric texture synthesis works the best for all pattern types. This is consistent with the pile-sorting finding in Section 5.4. This is not surprising and hints at the fact that GTS algorithms still need to find better means of capturing the true essence of exemplar inputs even if they start with the comparatively simple case of irregular distributions. The results also suggest the importance of further investigating the visual cues that figure most prominently in human judgments of similarity for geometric textures.

5.6 Case report: Evaluating one GTS algorithm

The evidence I gathered from my two studies suggests apparent preferences in terms of algorithm consistency. Even though understanding similarity and how we should effectively evaluate the success of GTS algorithms is still in its early phases and is worthy of deeper explorations, a simple illustration based on the results from this chapter is informative.

In this section I attempt a comparison between the patch-based GTS algorithm from Chapter 4 and the other GTS algorithms. Studying one algorithm in light of the others offers the area a first glimpse into the suitability of the evaluation strategies proposed in this work.

Part 1—Pile-sorting study: In the analysis reported for this study (Section 5.4), I found the following: Out of all other GTS algorithms, **A4** and **A5** correlated together most often. **A4** also correlated in some cases with **A1** and **A2**, but rarely with **A3**. The LEAVES and SWAMP piles that contained **A4** and **A5** were more likely to be selected as least similar to the exemplar than other sources. In comparison to **A4** and **A5**, **A1** acquired a significantly higher number of least similar ratings for SWAMP. Source **A3** acquired even lower similarity ratings for its PARALLEL and CURVIES arrangements.

Part 2—Comparison study: I observed that similarity ratings for synthesized source arrangements **A4** and **A5** were higher, and so were the ratings for source **A2** (Figure 5.12). The ratings for **A4** and **A5** deviated much less than those for other GTS algorithms for the different patterns.

To visualize the number of times participants choose each GTS algorithm I constructed Figure 5.14. In it I show, for each algorithm source, the number of times participants choose an arrangement from that source over arrangements from any of the other four algorithms.

A chi-squared test (at one degree of freedom, $\alpha = 0.01$) on the pairwise comparisons collected from this study shows a statistically significant bias in favour of **A4** and **A5** when tested against GTS algorithms **A1**, **A2**, **A3**. This means that participants were more likely to select **A4** or **A5** as more similar when presented with an arrangement from another GTS source. When shown a pair of sources from **A4** and **A5**, the decisions participants made were less significant indicating that they were equally likely to select either source as most similar. This explains the consistency noticed between these two sources demonstrated throughout the analyses in this chapter. As discussed earlier, density measures may have been a key factor in participant decisions.

5.7 Conclusion and future work

In Chapter 3, I attempted to uncover perceptual principles that cause algorithms to succeed or fail. The research resulted in a concise set of visual cues used by study participants to generate and compare geometric arrangements. In this chapter I take a broader observational approach and look at how people compare multiple arrangements generated from different sources (expert designers, GTS algorithms and a pseudorandom routine). The methodology I propose offers the GTS field an effective evaluation strategy for gathering and assessing geometric texture arrangements.

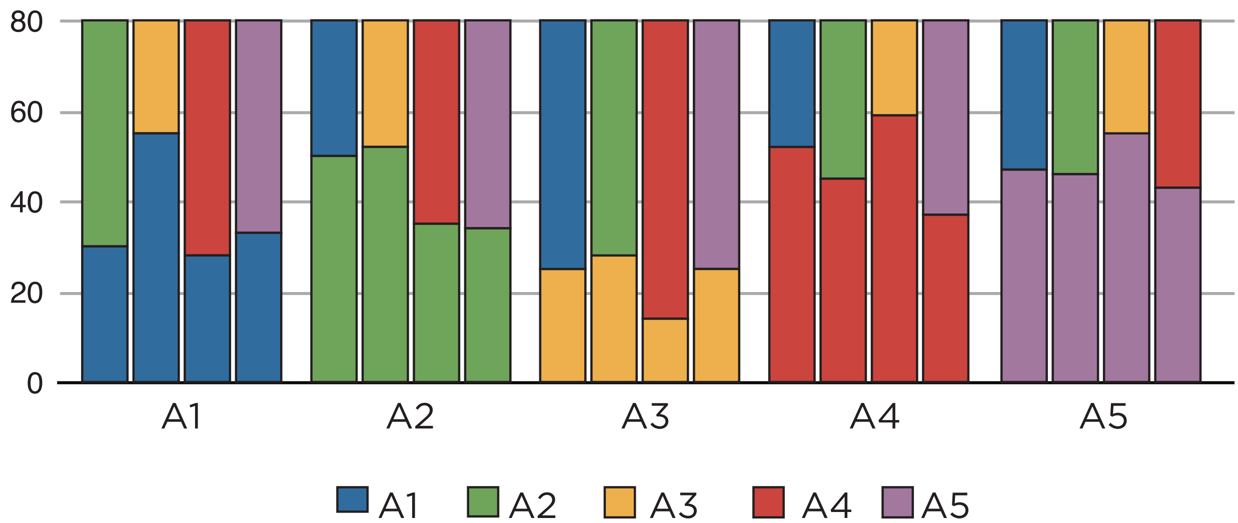


Figure 5.14: *Number of times participants choose one algorithm against another (only for GTS algorithms).*

Pile-sorting [59] and pairwise comparisons [85] have perviously been adopted as methodologies and have subsequently provided my research with a stable experimental paradigm. An interesting next step would be to adopt similar strategies for other area in NPR first as exploration tools and inevitably as evaluation methods. One concern with the style of textures investigated here is how it should be presented to a viewer. Since all the textures are black and white, presenting them on a white background in the pairwise comparison study may have influenced participants' judgements. The choice of whether to surround each texture with a black border or not could have also been a factor.

Most current GTS algorithms are heuristic in nature, and if tweaked, even slightly, could produce different arrangements biasing the results of this work. This will continue to be a major limiting factor when evaluating GTS algorithms unless standards are proposed. In this chapter and Chapter 3 I suggested standardizing the vector input style of the algorithms but further investigation into its practicality is required.

The experiments in this chapter have shown that no GTS algorithm performs well for all the patterns adopted. Future efforts should focus on developing a set of criteria to help researchers and designers decide which algorithm is best suited for their applications. Narrowing down to a succinct set of criteria would depend on collecting more arrangements and using effective study methodologies.

A final avenue of future work includes testing the significance of randomness in GTS algorithms. Since all current GTS algorithms include some aspect of randomness, it would be interesting to test the effect of randomness on the perception of similarity. Would multiple synthesized arrangements from the same algorithm be similar to one another? If so, then this could suggest increased robustness over other methods. Also, would it be possible to determine if one algorithm did equally well for multiple arrangement patterns? and what sort of study setup would be appropriate to achieve this comparison? These are only a few of the many questions one could ask to further understand geometric texture synthesis; I hope that they inspire researchers to take a fine grained approach to analyzing these algorithms in the future.

Chapter 6

Quantifying similarity of geometric texture arrangements

With the current advances in example-based geometric texture synthesis, there is a growing need for better and more effective measures to determine the quality of algorithms. In Chapter 5 I proposed a methodology that evaluates similarity of geometric arrangements gathered from multiple sources. This offers the GTS field its first reliable alternative to the comparisons currently being practised. However, relying solely on user studies leaves the task of judging successful, visually similar arrangements unfinished.

There are many quantitative measures that might be advanced as means of computing similarity between exemplars and synthesized arrangements. But despite extensive research in GTS, there has been no systematic study of the effectiveness and utility of such measures. It is important to contemplate the behaviour of simple quantitative measures of similarity, if only to rule them out as obvious bases of synthesis algorithms, and further justify the qualitative, study-based approach taken in the previous chapters. Looking beyond similarity, we should be able to use statistical analysis to perform a high-level verification of an algorithm's performance, for example ruling out the possibility that an algorithm's behaviour is completely random.

In this chapter I start by conducting an analysis of the spatial distributions found in a set of synthesized geometric arrangements from the GTS dataset and determine whether or not they are completely random (Section 6.2). I then explore some quantitative local and global measures commonly used in assessing similarity between images and correlate them with previously gathered geometric texture similarity judgments (Section 6.3).

6.1 Introduction

Research in the area of example-based texture synthesis has attempted different analysis techniques to understand, recognize and develop pleasing textures resulting in different algorithms [128]. However the problem of choosing effective perceptual similarity measures to determine success of results is a deep problem that has been studied by researchers in psychology and vision [63], and more recently in NPR and computational aesthetics [57].

To account for spatial layout differences in raster-based textures Lin et al. [85] objectively evaluate the global regularity of synthesized textures by comparing the underlying lattices of an exemplar and various synthesized textures using a geometric score. This score only targets what they refer to as *near-regular textures*, but still gives an efficient quantitative measure of how similar one arrangement's distribution is to another. A more recent method by Nan et al. [100] attempts to abstract spatial arrangements in architectural vector drawings captured using Gestalt grouping principles into a energy metric. This metric is composed of multiple spatial relationships between elements in an arrangement and gives some intuition as to the structural validity and similarity of the result.

In GTS, algorithms require exemplars to be described in terms of primitives and their point locations [5, 56, 91]. However, these higher-level measures have proven to be more useful for synthesizing arrangements than for analyzing them. Hurtut [54], for example, offers a statistical method to capture the appearance of irregular spatial arrangements that stems from geospatial analysis [41]. This appearance is computed as a quantitative measure adapted to perceptual theories in human vision. Despite the efforts to incorporate spatial analysis of textures, these measures have not been used to assess similarity of the results to those generated through other means.

My first goal in this chapter is to understand the spatial nature of GTS arrangements. I do this by examining whether or not there exists an inherent structure in the way motifs are laid out in comparison to the exemplars, i. e., rejecting the hypothesis that they are random. I then look closer at the spatial distribution of the different motif types and analyze their relationships to understand the structure. Finally I calculate some quantitative measures traditionally used in image analysis to compare raster-based textures and explore their limitations when correlated with similarity decisions gathered from my earlier studies (Chapter 5).

When choosing any quantitative measures for assessment, it is necessary to validate them experimentally, for their ability to discriminate visual characteristics within textures since they could fall short of our preconceived notions of similarity. It is possible that a measure captures only differences between textures. This does not necessarily invalidate them as being less informative of the nature of the arrangements, instead it may hint to the fact that reducing similarity to a set of numbers alone may not be a practical solution.

To illustrate my choices of quantitative measures, I adopt a subset of the CURVIES arrangements from the GTS dataset from Chapter 5, as shown in Figure 6.1: Expert Designer **H3**, GTS algorithm **A3** by Ma et al. [91], patch-based result **A5** from Chapter 4, and pseudo-random algorithm **R2**. Their basic point statistics are listed in Table 6.1. These arrangements were specifically chosen as practical representatives of the three synthesis sources adopted in that investigation. The outcomes of the evaluation in Chapter 5 specifically Figure 5.13 showed that **H3** was selected as one of the most similar human arrangement, **A3** and **A5** were selected as the least and most similar GTS algorithmic arrangements respectively and **R2** was the most similar randomly generated arrangement. The quantitative analysis methods used throughout this chapter apply analogously to all the other patterns in the GTS dataset.

6.2 Rejecting the null hypothesis

Various research areas like geology and biology use Complete Spatial Randomness tests (CSR) to investigate spatial point distributions. These tests show whether or not point sets are inherently random by observing how much they deviate from a homogenous Poisson point process [42]. Poisson processes are random sets of points. Both the number of points and their locations are random giving it a unique characteristic. All CSR tests initially assume that point sets are uniformly random (the null hypothesis) until they are proven otherwise.

Poisson point processes can be described as non-empty unordered sets of points at random locations x_i with a uniform intensity λ viewed from within a sampling window W in the plane, with the points extending outside the window infinitely:

$$x = x_1, ..., x_n \quad x_i \in W, n \geq 0 \tag{6.1}$$

The example homogeneous Poisson process in Figure 6.2 (left) was generated using the average intensity of all the point sets in Table 6.1. These points are essentially stochastic and unpredictable in their location and placement in comparison to their neighbouring points. There are three basic properties of a Poisson process:

1. The number of points that fall under any region in an area of interest Z has a mean $= \lambda$ area(Z) .

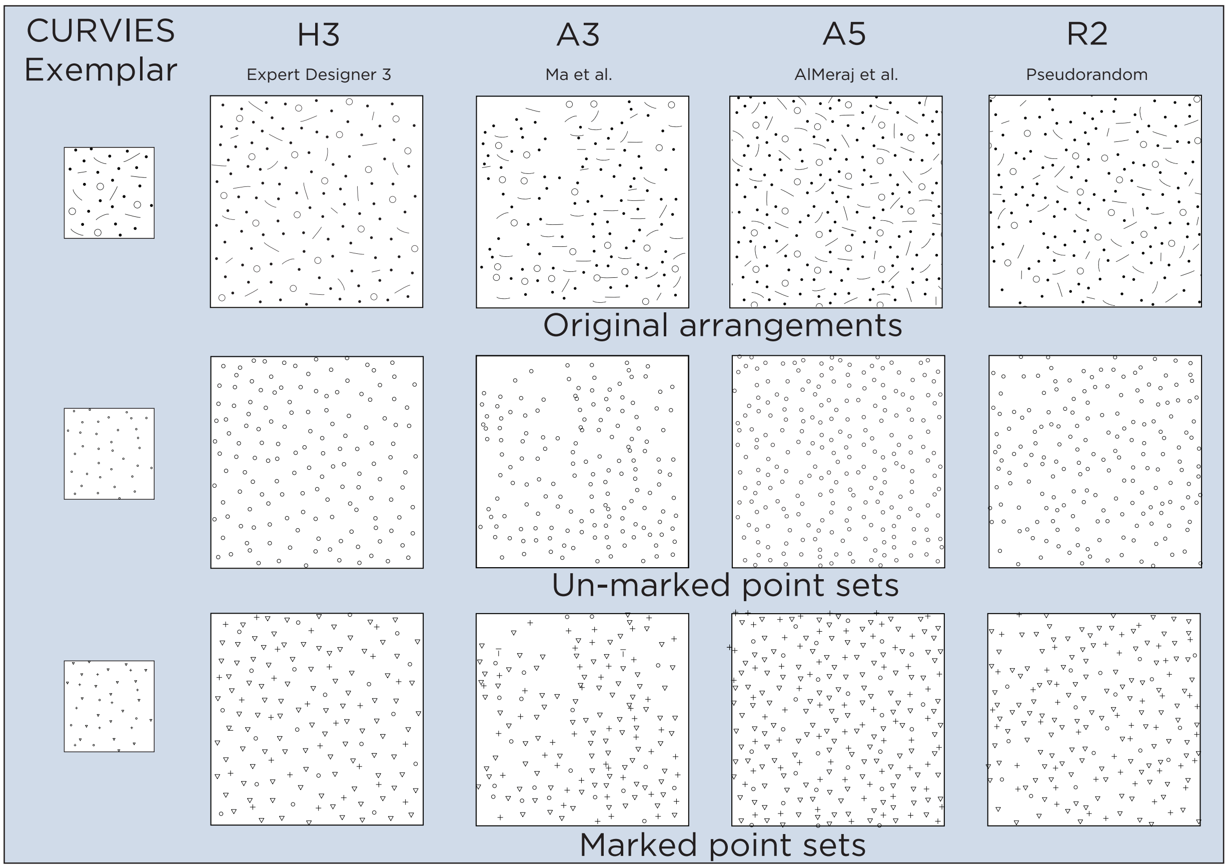


Figure 6.1: CURVIES *arrangements from the GTS dataset and their corresponding point sets gathered using spatial analysis software described later in the chapter.*

Exemplar	Average intensity 0.00407 points per square unit		
	frequency	proportion	intensity
0	3	0.0909	0.00037
1	21	0.6360	0.00259
2	9	0.2730	0.00111
H3	Average intensity 0.00216 points per square unit		
	frequency	proportion	intensity
0	20	0.137	0.000296
1	99	0.678	0.001460
2	27	0.185	0.000399
A3	Average intensity 0.00234 points per square unit		
	frequency	proportion	intensity
0	20	0.127	0.000296
1	91	0.576	0.001350
2	47	0.297	0.000695
A5	Average intensity 0.00348 points per square unit		
	frequency	proportion	intensity
0	27	0.115	0.000399
1	137	0.583	0.002030
2	71	0.302	0.001050
R2	Average intensity 0.00350 points per square unit		
	frequency	proportion	intensity
0	17	0.0919	0.000321
1	122	0.6590	0.002310
2	46	0.2490	0.000870

Table 6.1: *First order statistics from the* CURVIES *point sets from Figure 6.2*

2. The locations of points inside a region Z are independent and uniformly distributed within that region.

3. The points in two disjoint regions of a point set are independent.

Failing to satisfy any one or more of these properties means that the point set departs from a homogeneous Poisson distribution to non-uniform intensity or possesses dependencies between its points. The distribution on the right of the Figure 6.2 illustrates a non-uniform

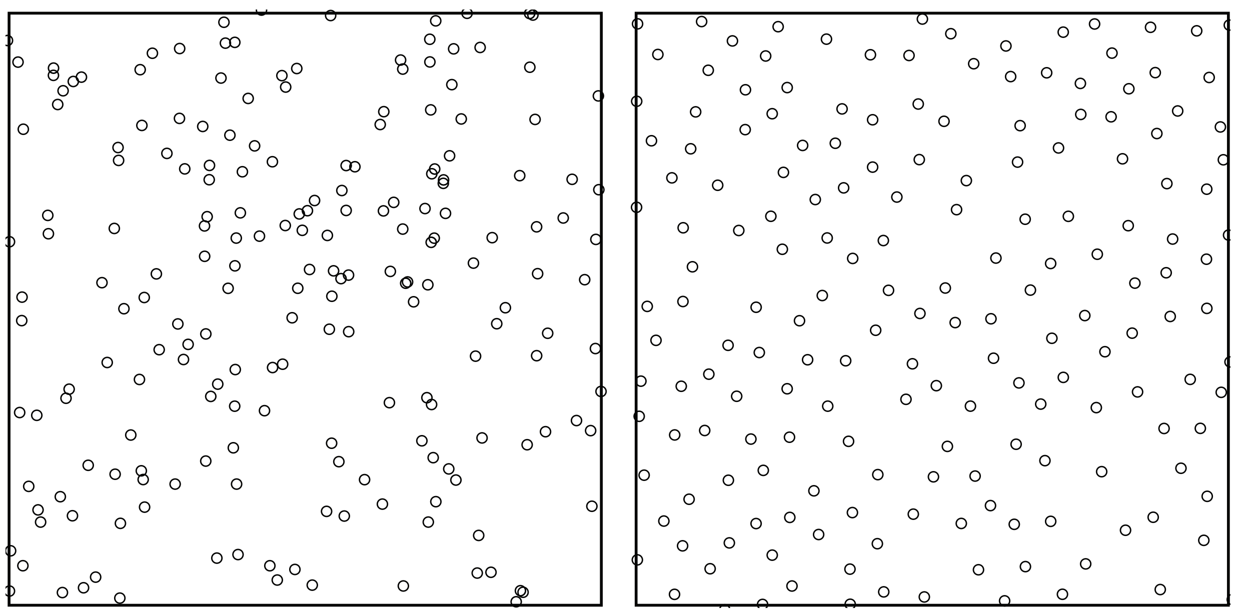


Figure 6.2: *A homogeneous Poisson point set and a regular point set.*

point set with predetermined distances between points, giving it a regular appearance. Note that regular in texture synthesis (Figure 1.2) is sightly different from the definition of regular in statistical analysis.

To show that the CURVIES point sets from Figure 6.1 are not synthesized completely at random, we have to reject the null hypothesis using one or more CSR tests. The tests proposed in this section include: the G function test for unmarked and marked point sets and a Pair Correlation Function (PCF) test. These tests are commonly adopted by geographers, geologists, cartographers, statisticians, and mathematicians in research to analyze natural phenomena such as forests [41]. However, a limiting problem with this sort of analysis as noted earlier is that it is done purely on data points, so we can not incorporate actual element geometries. I leave the perceptual studies conducted in Chapter 3 and Chapter 5 for evaluating the aesthetic appeal and overall similarity of geometric arrangements and focus only on the statistical and spatial layouts here.

To allow for spatial comparisons, I converted the geometric arrangements into their corresponding point set representations. Point sets are created by finding the centers of the motif bounding boxes and using them to describe the (x,y) locations of the motifs in the 2D plane. Point sets can be either **unmarked** or **marked**, as shown in Figure 6.1. Unmarked point sets do not have extra information about the points attached to them; in the case of our geometric arrangements, no knowledge of distinct motif shapes is present, only the locations of the motifs are known. Marked point sets distinguish between the distinct motif types using visual identifiers. For CURVIES, marks are integers, with '**0**' being the mark for the larger circles, '**1**' the mark for small circles and '**2**' the mark for the lines and curves. In Figure 6.1 these are represented as circles, triangles and plus signs respectively.

Table 6.1 contains statistics gathered from the point sets. Each table lists the intensity/density, the number of times each motif appears in the arrangements, and their proportions in comparison to other motifs. In the exemplar for instance, its overall density is 0.00407 and the small circles ('1') occur more often than the lines and curves ('2'), which occur more often than the large circles ('0').

In the next few sections I test randomness first for unmarked point sets (Section 6.2.1), and then for marked point sets (Section 6.2.2). I also analyze resulting relationships between the distinct marked point sets to gain a better understanding of their overall structures and correlations. To conduct this analysis, I use a readily available statistical computing software package called R[1] and a statistical package called spatstat [8, 9, 10]. All of the tests that follow initially assume that point sets are independent of one another, i. e., that each point set agrees with the null hypothesis which states that it is random.

6.2.1 A randomness test for unmarked point sets

The G function test:
This test helps us determine if the target point sets are completely random. In essence it is a measure of dispersion. To determine randomness for a point set with n points, the G function measures the distribution of distances from each point to its nearest neighbours as follows:

$$\hat{\mathrm{G}}(r) = \frac{\sum_{i=1}^{n} I_i}{n} \quad \text{where } I_i = \begin{cases} 1, & \text{if the } \textit{distance} \text{ is smaller than a threshold } r \\ 0, & \text{otherwise} \end{cases} \tag{6.2}$$

The G Function for an ideal Poisson process (CSR) is defined by:

$$G_{Pois}(r) = 1 - e^{-\lambda \pi r^2} \tag{6.3}$$

where λ is the number of points per unit area defined as the intensity.

[1]The R software URL http://www.r-project.org/

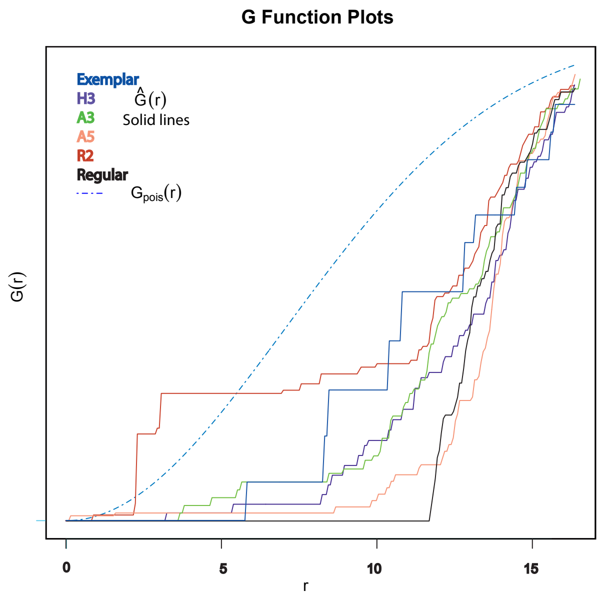


Figure 6.3: *Superimposed G Function results for the* CURVIES *arrangements.*

A $\hat{G}(r)$ that is larger than the expected $Gpois(r)$ for a point set suggests that the nearest neighbour distances in the point set are shorter than those for completely random point sets (the CSR); this means that the appearance of the point set is more clustered. Values of $\hat{G}(r)$ smaller than $Gpois(r)$ suggest a more regular pattern.The height of the G function can be described as a cumulative frequency of observed distances in the point sets.

In Figure 6.3 the $\hat{G}$ curves for nearly all the CURVIES point sets suggests that they are more regular except for one. The **R2** point set is observed to have more points at distances smaller than 10 (notice the correlation with the Poisson curve). Deviations from the Poisson curve like these reject the initial null hypothesis.

To understand the layouts, I look at neighbourhood distances captured in all the different sources; for example, the totally regular arrangement added here to illustrate randomness has a minimum distance equal to 12 between the points. The other point sets **H3** and **A3** capture similar distances between points suggesting that they both have similar global distributions of point placements.

Notice that the G function for the exemplar is zero for r $\leqslant$ 6 which indicates that there are no nearest neighbour distances shorter than 6 in the exemplar. Similar neighbourhoods at larger distances result in the stair-like curve shown. When compared to **A5**, which has a minimum distance threshold of 8, this distance does not seem to match despite the accurate boundary measures made in the algorithm. This discrepancy arises because centres of bounding boxes do not accurately reflect the true minimum distances between elongated motifs. This problem persists throughout the analysis below. Arrangement **A5** deviated from the Poisson curve even more than the other synthesized arrangements and appears to have a more regular appearance. Overall, neighbourhood distances are almost all the same at distances greater than 13.

It should be noted that estimating spatial randomness using the G functions is affected by bounding edge effects. These are caused by the lack of known point locations outside the view window (W). It is common that edge corrections are applied to reduce this bias. In the R statistical package, this is resolved by adding an edge correction weight to both the above equations.

6.2.2 Randomness tests for marked point sets

In the previous section I used the G function test to analyze randomness of unmarked point sets to show that they were not synthesized at random. In this section I analyze the marked point sets using various CSR tests and correlations to understand dependencies between the distinct motif types.

In R, a *mark* variable is an additional coordinate added to point sets that holds extra information about the particular elements or their locations. A mark can be a word or a real number depending on the information it conveys. For the CURVIES point sets as mentioned earlier in the chapter, marks are integers with '**0**' being the mark for the larger circles, '**1**' the mark for small circles and '**2**' the mark for the lines and curves. Marked point sets are also refered to as multi-variate or multitype point sets.

A marked point pattern is defined as a non-empty unordered set of n points in a region M, as in Equation 6.5:

$$y = (x_1, m_1), \ldots, (x_n, m_n), \qquad x_i \in W, m_i \in M \tag{6.4}$$

where x_i are the point locations and m_i are the corresponding point marks.

The G function (marked point sets):
This function is an extension of the G function for unmarked point sets (described in the previous section) that accommodates multiple distinct point types. It essentially measures the distribution of distances from a point of type i to the nearest point of type j and is represented by $G_{i,j}$. If the point types are independent, then the $G_{i,j}$ curve would be equal to or close to the G_{Pois} curve. Deviations between the $G_{i,j}$ curve and the G_{Pois} curve

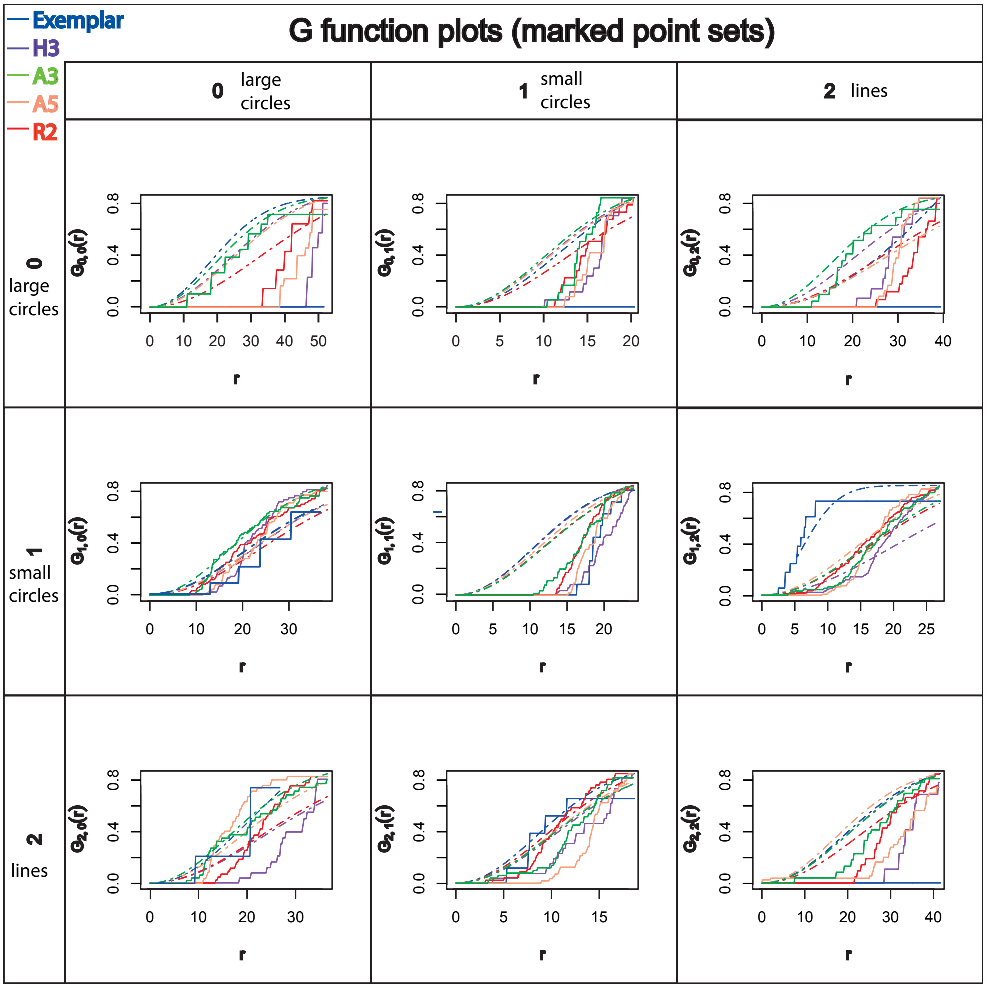


Figure 6.4: *The G function for superimposed marked* CURVIES *point sets.*

suggest local dependence between the different point types. Negative associations that fall below the G_{Pois} curves suggest that there are regularities in the dependencies between the different types in the point sets.

When plotting the CURVIES point sets with $G_{i,j}$ in Figure 6.4, the x-axis represents the distances between the points of different types, and the y-axis represents the frequency of their appearance in the point set. The nature of this function implemented in R is non-symmetric, hence the differences in plots along both sides of the diagonal. I only discuss some instances from the lower half of the diagonal.

For the exemplar, notice that the neighbouring distances between large circles relative to themselves ($G_{0,0}$) and lines relative to themselves ($G_{2,2}$) suggest complete regularity. The distances of lines relative to neighbouring small circles ($G_{2,1}$) has a positive association which exhibits a clustered dependency. This suggests that small circles are more likely to be placed closer to lines motifs.

In other point set neighbourhood associations ($G_{1,0}$, $G_{1,1}$, $G_{2,0}$ and $G_{2,1}$), the marked point sets **A5**, **H3**, **R2**, and **A3** capture similar deviations from the G_{Pois} curve as the exemplar point set suggesting a lack of dependence between the types. The relative distances between lines and large circles $G_{2,0}$ for **R2** suggest a stronger regularity dependence than the others. Also, **A3** exhibits a noticeable randomness in the relationship between large circles ($G_{0,0}$), which is not present in the other sources.

These findings show that points of the same type ($G_{0,0}$, $G_{1,1}$ and $G_{2,2}$) have the highest amount of dependencies within the point sets; and the relative dependence between points of different types is less significant. It surprisingly highlights that **A3** and **R2** have less similar motif distributions to the exemplar and other point sets. These relationships are interesting and not well investigated in GTS. They may exist more for the irregular arrangements we target in this work rather than for the regular ones. It would be even more interesting if further analysis on the other arrangements in the GTS dataset result in similar findings.

The Pair Correlation Function (PCF):
The PCF tests the probability of observing any pair of points separated by a distance r. It is calculated as follows:

$$\mathrm{PCF}(r) = \frac{\hat{K}(r)}{2\pi r} \tag{6.5}$$

where $\hat{K}$(r) is the derivative of K(r)—the Ripley K function—of the point set[2] which is divided by the probability of a Poisson process. The PCF for a Poisson process is identically equal to 1, indicating complete randomness. Values smaller than 1 suggest regularity between points; values greater than 1 suggest clustering.

In Figure 6.5 the marked pair correlations computed for each of the types of motif in the point sets show some differences between the different types. As opposed to the G plots this chart is symmetric. The probability of locating any point type at distances smaller than 10 is completely random due to an unstable kernel estimate that is not efficient at

[2]The Ripley K function is another test common for CSR. It determines deviations from spatial homogeneity by assuming isotropy across the target arrangements. This measure allows for biased sampling of pairwise distances in the point arrangements, which mainly targets the smaller distances resulting in fewer comparisons than those made in the G Function.

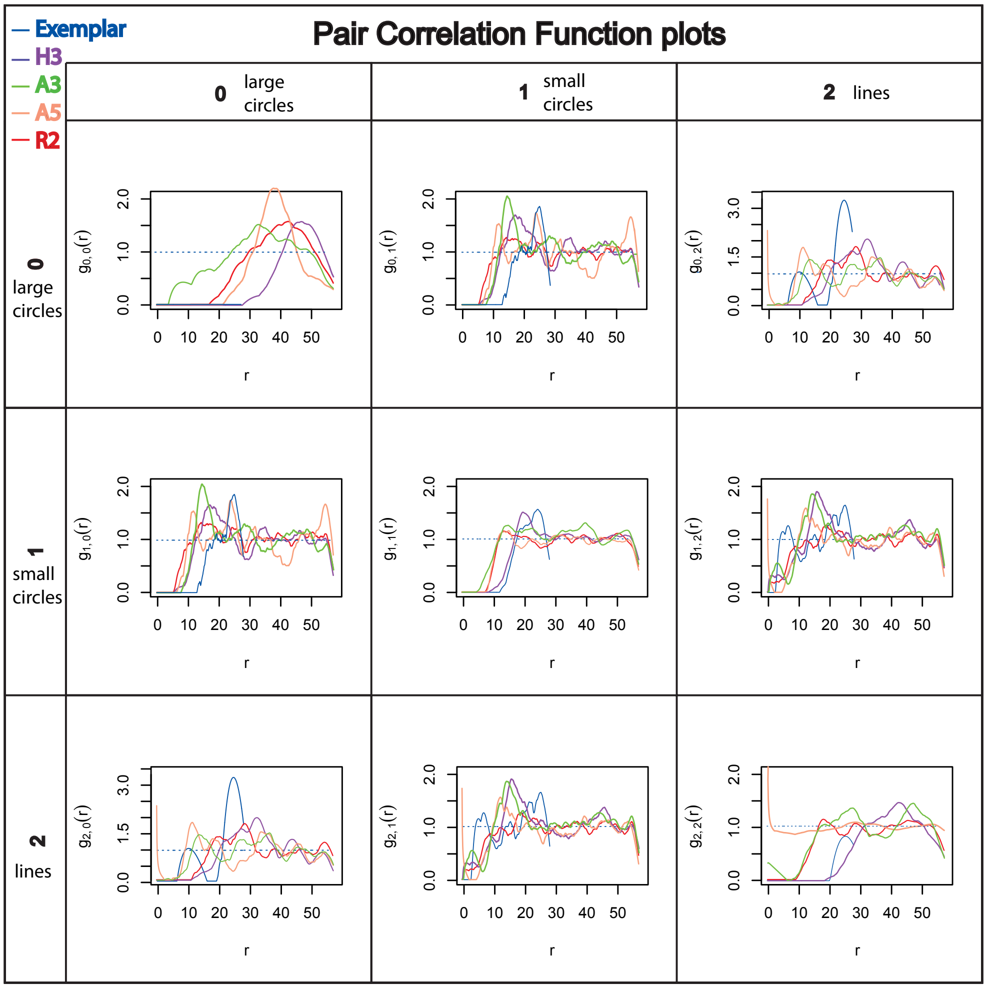


Figure 6.5: *The chart shows the Average of the Pair correlations for the* CURVIES *points sets.*

small distances. Large circles and lines in the exemplar again exhibit complete regularity in their correlations. The probability of locating small circles near larger ones also shows a regular deviation for all point sets in distances smaller than 10 which quickly increases and clustering for **H3** and **A3** at distances larger than 10. **H3** and **A3** also have similar spatial appearances across all distances between lines and small circles.

There are also high probabilities of observing small and large circles at distances between 20 and 30 for **A5** and the exemplar suggesting a similar spatial appearance in both point sets. This is most likely due to the accurate boundary-to-boundary distance measures used in the algorithm. The PCF measure in general gives a more global analysis of distributions within point sets and offers more information than those found in the G function plots discussed above.

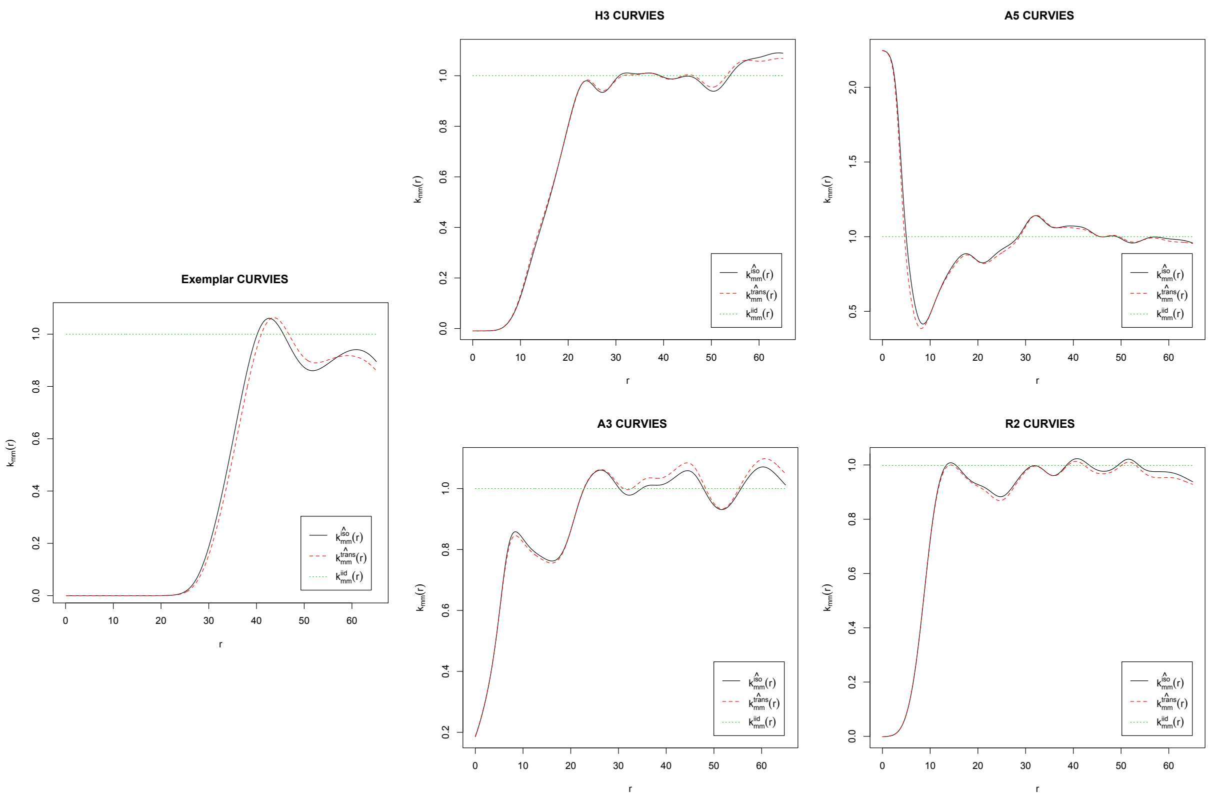


Figure 6.6: *The chart shows the average of the marked cross-correlations for the different motifs in the point sets.*

Marked correlations:
The marked correlation of the CURVIES point sets are plotted in Figure 6.6. This statistic describes the conditional probability that any two points separated by a set distance r have the same type [118]. Here the correlation is not the same as in the usual raster-based analysis where a small example is superimposed onto a larger one and differences are estimated. A value of 1 suggests lack of correlation between the points at that distance. Values larger than 1 suggest that more points of similar types exist at that distance, while values smaller than 1 indicate that nearby points are more likely to have different types.

For point set **A5** the curve starts at much larger than 1. This suggests that points separated by distances smaller than 10 are more likely to be of similar types. But the curve converges rapidly to a value less than 1, suggesting that few points in the set exhibit this phenomenon. The opposite behaviour appears for arrangements **A3** and **R2**. Points closer together have different types but increase rapidly between distances 0 and 10. This shows that again few points at these distances have different types.

H3 shows a higher frequency of points of different types at similar distances. This is the only point set to capture a similar correlation to the exemplar. Now that we know what to look for, we can see these behaviours int the arrangements themselves. Interestingly, humans are again better at capturing accurate characteristics of arrangements than GTS algorithms.

In conclusion, these spatial tests further confirm that example-based GTS methods do not synthesize arrangements at random. They also show that the inter-relationships that exist between multiple point types are complex enough to warrant further investigations into their importance in deciding overall similarity.

6.3 Raster-based statistical measures

Humans are sensitive to varying levels of brightness and contrast, which makes it essential to consider quantitative measures that target lower-level image statistics in the search for effective similarity measures. Previous studies of textures have ranged from realizing textures using their nth-order statistics [63], to more recent automatic pattern recognition methods that use machine learning [20].

A study of relevant analysis measures for comparing natural textures by Benjamin Balas [12] shows that lower-level pixel statistics such as the mean, variance, and range of luminance values are important for creating perceptually matching natural textures. Some other measures that have been used include standard deviation, density, and entropy.

For textures with more structured content, some investigations suggest that local pixel measures are as important as global ones [123]. These measures include co-occurrences, auto-correlations, magnitude correlations, feature extraction, frequency-space methods like the Fourier transform, image segmentation and supervised classification. Statistical values such as these could give us important information about the overall low-level information within these arrangements.

Generally in texture analysis, image processing techniques involve treating images as 2D signals and applying signal-processing techniques to them. These signal processes deal with measuring features quantitatively. If measures collected from a texture are similar to those of another texture, it is more likely that both of them contain similar content. However, this is not always true since textures could vary in their structures but still maintain similar statistics. To see if such measures could contribute to an analysis of GTS, I correlate them to results of the pile-sorting study in Chapter 5.

In the following, I divide the analysis of GTS using raster-based measures into global (Section 6.3.1) and local (Section 6.3.2) image measures. Table 6.2 and Figure 6.7 show these measures applied to four rasterized CURVIES arrangements. Each arrangement was first converted into a 300ppi image and processed in MATLAB as a black and white or greyscale image. In the analysis below I assume that arrangements are always viewed at the same resolution.

6.3.1 Global image measures

Density
Density is measured by summing the geometric areas of each individual motif and dividing them by the overall area. I found this to be more accurate for representing the vector motifs used in this dissertation than pixel-based density due to aliasing artifacts. There may be value in pursuing an investigation into how this measure differs from pixel-based ones (black vs. white pixels) in capturing perceived density. I have included both density measures for the complete GTS dataset in Appendix D.

In Table 6.2 we find that the density of arrangement **A5** is the closest to that of the exemplar followed by **A3** and **R2** respectively. Irrespective of the geometric context, this should mean that **A5** is more similar to the exemplar than the others. But this is not the case; the results of the pile-sorting study in Chapter 5 rated **A3** as being the least similar to the exemplar and **R2** as much more similar. The pseudorandom algorithm that generated **R2** ensures closer densities but not much more in terms of distribution and human observers noticed this effectively for this arrangement style.

At an extreme, arrangement **A5** has the highest density and was selected also as similar to the exemplar in the study. In Section 5.4.4 I illustrate how density is an important factor in judging similarity and how human categorizations of textures correlate and discuss it in more detail.

As illustrated by the metrics here, when given density alone it is difficult to confirm that visual similarity exists. There exists a trade-off between gathering perceived visual similarity and maintaining similar densities that must be considered when synthesizing geometric texture arrangements.

Measure	input exemplar	H3	A3	A5	R2
Density	0.029	0.0198	0.0257	0.0315	0.0250
Entropy	0.1916	0.1470	0.1499	0.1937	0.1681
Mean	0.9706	0.9790	0.9785	0.9701	0.9751
Variance	0.1558	0.1434	0.1451	0.1702	0.1558
Cross Corr. Coeff.	1	0.0905	0.1023	0.0880	0.0776
	USGS source				
ACE	0	339.30	99.722	27.9801	315.8

Table 6.2: *Basic quantitative pixel-based image measures.*

Entropy

Entropy is a statistical measure of randomness often used to describe texture content in images. In information theory, the value usually refers to Shannon Entropy, which quantifies the expected value of information contained in a target source [114]. Entropy is defined by $-\sum(plog_2(p))$, where p is a probability distribution of the discrete set of pixels that is equal to some non-negative value. It is mainly applied to measure variations of images in data compression but is also used in image analysis. I view it using the following guidelines: An image that contains one flat colour will have no entropy, i. e., no information, while an image with varying contrasts from one pixel to the next will have a much higher entropy. The amount of entropy in an image describes how different it is from others. For the GTS arrangements, the entropy in **A5** is relatively similar in comparison to the entropy of the exemplar. This suggests that all the synthesized images contain approximately the same amount of information. **H3** and **A3** vary the most from the exemplar entropy suggesting more differences in content.

Mean and Variance

The Mean is the arithmetic standard average of the luminance values in an image. For the CURVIES sources we find that arrangement **A5** has a similar mean to the exemplar suggesting that the two have a similar range of intensities.

Variance describes the dispersion within an image. There are several measures of dispersion, the most common being the standard deviation. This measure indicates the degree to which the pixel data is dispersed or *spread out* around the mean of the image. The variance shows how much pixels deviate from the average luminance of pixel content. Higher variances imply that more luminance is spread far from the mean.

For the CURVIES arrangements, the variance of **R2** is exactly the same as the exemplar's variance suggesting that pixel luminance's are spread similarly. **A5** has the highest variance. In the patch-based algorithm, editing regions that do not represent the exemplar by adding or removing motifs and still maintaining densities may have led to this variation.

6.3.2 Local image measures

Despite their importance in understanding image content, global measures do not consider local structures within image content. A study by Tyler et al. [123] has shown that for structured textures, local measures may be just as important as global ones. Some of the local quantitative measures relevant for analyzing GTS arrangements include co-occurrences, correlations, feature extraction, transform methods like Fourier transformations, image segmentation and supervised classification.

There is little value in attempting image segmentation and supervised classification on texture arrangements since these methods are meant to segregate the image into meaningful regions which is inappropriate for describing geometric distributions. Below, I discuss some local analysis metrics commonly used in image analysis.

Cross-correlation
Cross-correlation is a standard method of estimating the degree to which two signal series are correlated. It essentially tries to find known features by searching along the signal and comparing it with a reference one and quantifying differences. Cross correlating the CURVIES exemplar with itself would yield the highest possible value of 1, since the signals are identical, see Table 6.2.

However, when correlating the exemplar with the other arrangements, I find that the values differ significantly from the exemplar, but not much from each other. This finding is interesting, as it highlights that there exist major differences between the images, but does not tell us how similar they are.

Average Co-occurrence Error (ACE)
This measure involves calculating the difference between the grey-level co-occurrence matrices (GLC) of pairs of images, and is also characterized as a distortion measure between images. The GLC matrices estimate local image properties related to second-order image statistics (e. g., variance, standard deviation and correlation) and have been used extensively in texture analysis [46, 65]. A GLC matrix can be defined as a tabulation of how often different combinations of pixel brightness values occur in an image.

ACE was devised by Copland et al. [25] as a texture similarity metric which highly correlates with human perception of stochastic natural textures. It takes the GLC matrices of two images and computes a distance between the two to describe differences. The smaller the ACE error, the more similar the images are. It has proven effective when applied to measure similarity between fractalized (distorted textures) to preserve texture coherence in animation NPR textures [17] by distinguishing between artificially distorted images and their original sources.

The main limitation with calculating this measure is that it requires the target image to be compared to a reference image of equal size. Since the analysis up to now takes the exemplar to be the only source of comparison, applying these measures is not be possible. For the sole purpose of offering an overview as to whether or not these assessment metrics capture important details and similarities I choose to use the original USGS texture arrangement in place of the exemplar. One other option could have been to take smaller parts of the synthesized textures. The end of Table 6.2 contains the measure gathered from correlating the images with the USGS example. USGS sources can be viewed generally here as direct tilings of a texture. When comparing them with others, I do not assume that they are more or less similar to the reference. Further investigations into their similarity are needed in the future.

When applied to GTS, I find that **A5** has the smallest ACE error amongst other arrangements. This suggests that it captures a similar distribution of grey values as those found in the reference. **A3** deviated somewhat from **A5** which means that they also contain similar grey neighbourhood distributions. The difference between both of these and **H3** and **R2** is large. This could be caused by variations in distances between motifs. **H3** contains larger gaps while **R2** has smaller ones as well as a clustered appearance.

Fourier analysis

One final image processing tool used to analyze textures is Fourier transformations [52]. A Fourier transform converts the pixel image (which is in the spatial domain) into its corresponding spectral domain by decomposing it into a set of sine and cosine signal components. The common way to visualize the resulting spectral domains is through the power density and phase spectra (see Figure 6.7).

The power density spectrum describes how much the power of a image signal is distributed over different frequencies in an image. Generally images contain information at all frequencies with the higher frequency power containing more image feature information than lower ones. A phase plot on the other hand shows the locations of pixel data and highlights main features that occur in the image.

Both the power and the phase spectra are important for reconstructing the image from the frequency domain to a spatial domain. Since the phase plots for the CURVIES arrangements do not yield much information about the structure of the spatial images I will only discuss

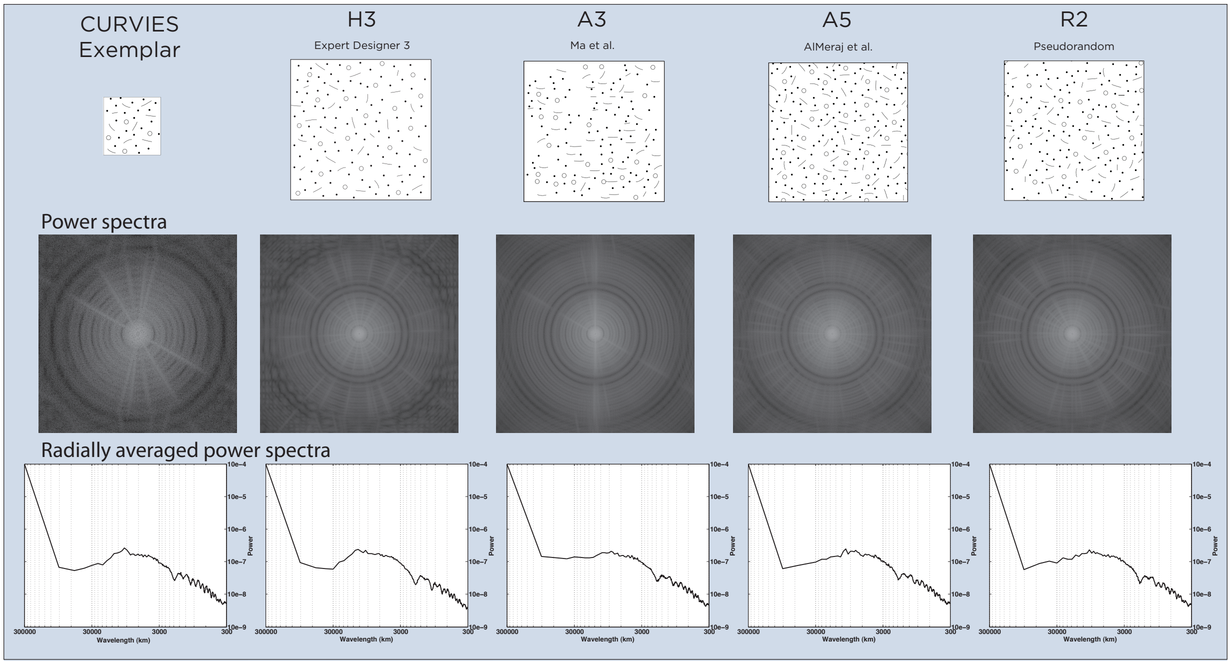


Figure 6.7: CURVIES *examples with their corresponding Fourier power spectra.*

the power spectrum results. Both spectra types for the complete GTS dataset can be found in Appendix D.

The power spectra in Figure 6.7 show dominating directions and frequencies within the CURVIES images. The circular rings centered at the middle of the Fourier power spectra suggest periodic order of relative structures at various spatial scales in the image. The power spectrum for the exemplar shows equally spaced concentric circles around the center of the spectrum and similar appearances exist in the synthesized arrangement spectra suggesting similar element repetitions.

The bright lines protruding from the center show the dominating directions in the image. These lines suggest orientations perpendicular to prominent edges found in the arrangements (the lines); A solid vertical line, for example, means that the image contains a strong horizontal bias. Power spectra are symmetrical to their centers so we notice two of each the same features present in them. The exemplar spectrum also shows five main directions as a result of motif orientations. When comparing these directions to the other spectra we notice that **H3** and **R2** capture similar orientations while **A5** captures an even wider range of angles, more than those found in the exemplar. The angle differences here can be attributed to increased rotation of the motifs by the algorithm (Chapter 4). **A3** has two prominent angles with the horizontal one being the strongest. This suggests that the source algorithm did not consider capturing motif rotations during the analysis.

There are alternative plots for visualizing differences between arrangements that can be gathered from the power spectra. For example, Ulichney [125] derives a useful one-dimensional statistic from the 2D power spectra called the radially averaged power spectrum, shown in Figure 6.7. The plot is calculated using the average of all possible directional power spectra. This measure is generally suited for understanding periodic point sets by viewing and comparing information. When comparing these charts to each other, we find that the radial statistics of **H3** exhibits similar direction artifacts as the exemplar more so than the algorithmically generated arrangements. This suggests that the expert was capturing similar orientation more effectively.

As an extension to these measures, Wie and Wang [130] convert the Fourier spectrum into a differential distribution function that is able to quantify local spatial statistics for non-uniform point sets. These measures may be relevant for analyzing geometric textures and possibly support new spectral-based synthesis algorithms for GTS.

Some of the synthesis algorithms analyzed above capture similar perceptual qualities to those generated by an expert human designer. These findings correlate with the similarity judgments gathered in the pile-sorting study indicating that perhaps there are intricate structured ways that we perceive texture distributions. Further investigation into understanding these structured ways is essential for the advancement of GTS and will lead to algorithms that capture local and global characteristics more effectively.

6.4 Conclusion

Through studying quantitative spatial and spectral similarity measures applied to CURVIES arrangements and their point sets in this chapter, I was able to show that synthesized GTS results are not random (Section 6.2) and that it is impossible to rely solely on single quantitative metrics for judging similarity due to limited content description (Section 6.3).

It is clear that global image statistics alone offer limited knowledge about the layout and composition of geometric arrangements and that local measures give us more information about neighbourhoods; but neither are sufficient to describe content. To some extent, when the statistics of two arrangements differ significantly, we can have some confidence that the arrangements themselves will be visibly dissimilar. But statistics that align very closely do not seem to predict similarity. An arrangement cannot be reduced to a single scalar value without erasing much of the high level knowledge we use in making similarity decisions.

It may be possible to combine multiple statistical measures into one higher-dimensional similarity metric that correlates more closely with human similarity judgements.

In Chapters 3 and 5, I show how similarity judgements can vary given a single exemplar. In addition to analyzing such judgments, I gathered qualitative descriptions during multiple empirical studies. I believe that taking similarity decisions and descriptors from human observations in conjunction with some informative quantitative image-based metrics can reliably support strong claims of visual similarity. Further studies that include the analysis of collections of pixel-based metrics may also give further insight into their limitations in GTS analysis.

However, until an effective balance of these measures is achieved, evaluation methodologies such as the ones I presented in this dissertation (Chapters 3 and 5) are useful for capturing perceived similarities between synthesized arrangements of current and foreseeable GTS algorithms.

Chapter 7

Conclusion and future work

Computer graphics researchers will always be able to create ad hoc algorithms that attempt to solve visual problems like geometric texture synthesis. These attempts are worthwhile and inform our understanding of the effectiveness of such algorithms. However, there is also value in attempting to uncover the underlying perceptual principles that cause these algorithms to succeed or fail. This dissertation looked into identifying such principles and how to effectively validate the success of geometric synthesis algorithms.

In a previous attempt to evaluate synthesized results, Lin et al. [85] chose a black-box approach to correlate a quantitative measurement (the geometric norm) of a synthesized arrangement with user satisfaction. This was sufficient to gain a quantitative number on “what” constitutes satisfaction for synthesized textures but did not give any reasons for “why” the participants were satisfied. The research presented in Chapter 3 looked into these reasons and employed a first of its kind complementary white-box evaluation approach that looked at aspects people were drawn to when perceiving and evaluating similarity of geometric arrangements. In doing so I discovered a descriptive step-by-step process that offers the GTS area a number of perceptual qualitative features like density, white space, and regularity. These features summarize our high level perceptual understanding of geometric arrangements and inform us of the amount of influence each of the features have on perceiving individual arrangements. The same set of features could be used to evaluate arrangements for similarity, visual appeal and attractiveness in future investigations.

When geometric arrangements were synthesized by hand, my studies highlighted three strategies adopted by participants: tiling, structures and random. Preferences were geared more towards a tile-based approach which suggests that we inherently look for structure when we compare. Although diverse, the features and strategies gathered in this work together provide a solid perceptual basis for researchers in the area of geometric synthesis to promote deeper more structured investigations into perception and new effective synthesis methods.

In Chapter 4 I showed how some of the insights gained from perceptual studies like the one above can flow back into designing effective synthesis algorithms. The resulting patch-based algorithm devised from one of the synthesis strategies is robust, simple and effective in synthesizing unique irregular geometric arrangements. Considering similar methodologies to study visual perception and using results to guide algorithm design can improve results and further increase our knowledge of perception not only for GTS but for other areas of NPR too.

With a recent surge of GTS algorithms including the one developed here, validating effectiveness of synthesis results is becoming increasingly challenging. In general, compelling evaluations in NPR are often difficult to develop and conduct. To achieve a more principled foundation for GTS in Chapter 5, I devised a geometric dataset to serve as a standard reference and an evaluation methodology that utilized the dataset to validate existing algorithms. Evaluation involved a pile-sorting strategy and pairwise comparisons of results from multiple target algorithms. This approach to evaluating GTS arrangements is novel and offers a stable platform for comparing success between many algorithms pealing away the subjectivity involved in common practices. From this investigation I found that preferences of the synthesis approaches were different depending on the style of the texture and that algorithms were not likely to succeed in synthesizing similar results for all the target irregular arrangements. These draw attention to the difficulty of the problem and could eventually lead to alternative strategies. The platform is also simple for implementing other comparisons while taking advantage of an extendable geometric dataset.

Overall, the findings presented in this dissertation provided a better understanding of similarity in the context of GTS and showed that perception is an important factor that has been until now underestimated in the texture synthesis community. Incremental steps such as the ones taken here will eventually answer some of the many open-ended questions related to defining aesthetics, similarity, appeal, and preference of the many NPR results achieved as well as enhance future perceptual investigations of GTS.

In the next few sections I discuss some ideas for future investigations leading from the contributions presented in this dissertation.

7.1 Boundary-Aware texture synthesis

Boundaries of natural textures like orange peel tend to have a different appearance to their textured interiors. To reproduce textures like watercolour paint brush strokes, other peeled fruit, and torn paper, it is important to have a synthesis algorithm able to capture the natural transitions between regions of the different textures. Likewise this extends to geometric textures. Without boundary effects, areas between adjacent textures look artificial.

In Chapter 4 I discussed the potential of incorporating boundary awareness into the synthesis of geometric element arrangements. The idea was presented primarily to improve the visual appearance of textures on maps for cartographers. Vision and perception researchers have discovered that pattern/image terminations are identified in the low levels of our human visual processing, offering an exciting area for further perceptual research [71].

Being able to superimpose a border around an arrangement of geometric elements offers a natural ending for continuous distributions, further enhancing the visual appearance of textures. All existing geometric techniques [5, 14, 54, 56, 62] do not consider the texture appearance at the perimeter of the exemplar of vector-based arrangements, except for Ma et al. [91] who proposed solutions to a similar problem for 3D geometry. However, they did not consider motifs on the borders to be different than the interior ones. Figure 7.1 shows an arrangement generated with homogeneous motifs by Ma et al. and another two that illustrate boundary induced arrangements we want to achieve.

Another relevant research area that addresses the issue of synthesizing boundaries is solid texture synthesis. This form of synthesis is described as an effective way to represent the external and internal appearance of 3D models [107]. Solid-based texture synthesis techniques can provide useful insight for incorporating boundary awareness into geometric texture synthesis algorithms. While still not yet a major field in itself, boundary dependant synthesis methods have been addressed in solid-based texture synthesis. Pietroni et al. [107] presented a novel classification of solid texture synthesis methods; it includes boundary-independent and boundary-dependent techniques applied when synthesizing solid textures on surfaces and within volumes. Pietroni et al. describe boundary dependent techniques as methods that exploit an object's volume to orient and guide the synthesis process. Interior textures are synthesized to conform to the known texture and shape of the boundary, as shown in Figure 7.2.

One concerning aspect of solid texture synthesis research that address boundaries is that the boundary-dependent techniques are semi-automatic. Existing algorithms rely heavily on user input of volumetric texture properties. The system then infers how to synthesize

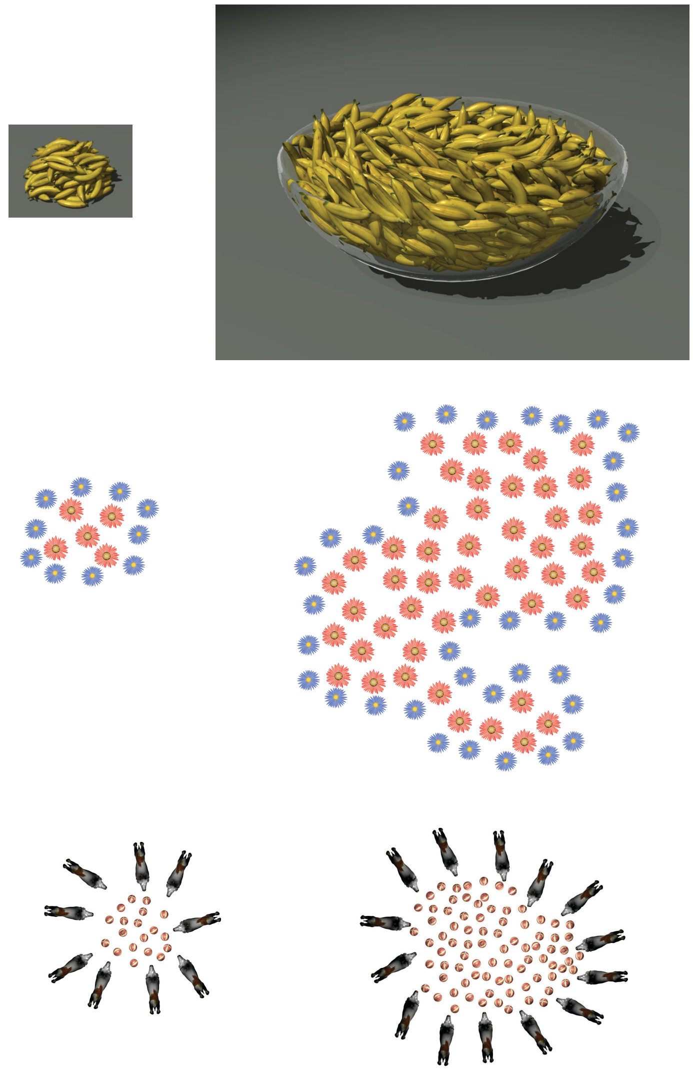


Figure 7.1: *Texture synthesis by Ma et al. with shape awareness and a 2D proposed example with boundary awareness.*

the interior and exterior texture accordingly. Various methods have been proposed to visualize in real-time object volumes and their interiors using one of three texturing styles: isotropic, layered, and oriented textures. More information on these methods can be found in the survey by Pietroni et al. [107].

A different but also interesting example-based boundary synthesis method is the pixel-based texturing model by Ritter et al. [110]. They are able to mimic the appearance

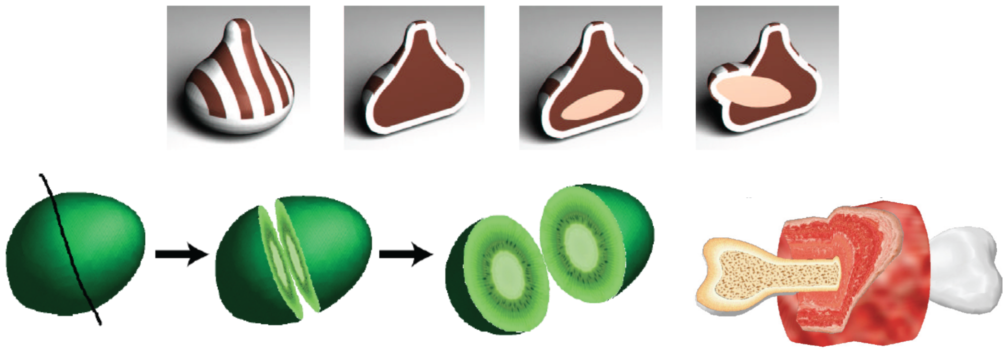


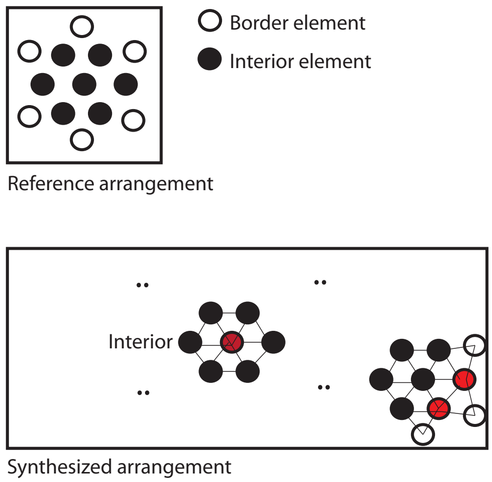
Figure 7.2: *Boundary-dependant techniques in solid texture synthesis [107].*

of painted brush strokes on canvas, capturing both internal paint texture properties and the exterior stroke boundary appearances. The Ritter et al. method takes in a sample texture as input and subsequently allows the user to draw more of it on a canvas via a virtual brush. To achieve realistic results, the system dynamically generates texture in the regions covered by the brush using the Image Analogies system of Hertzmann et al. [50]. The synthesis of the region interior involves the use of a multi-scale pixel neighbourhood matching algorithm and a prescribed energy function to capture features at different scales. To capture boundaries of textures during synthesis, Ritter et al. add an energy function that computes the energy contribution based on pixel colour and neighbourhood shape, to distinguish between boundary pixels and interiors of textures.

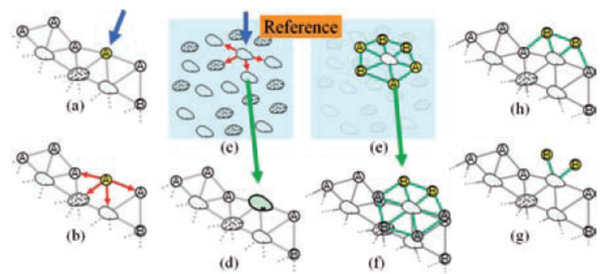
A similar approach to mimic the appearance of geometric arrangements across a 2D plane would be worthwhile. Given a sample geometric arrangement enclosed within geometric boundary elements, an algorithm should synthesize a similar arrangement within a predefined space. The routine should accurately analyze the input example for its boundary elements and distinguish them from interior elements while taking into account their spatial distributions. The results in turn should offer visually pleasing textures with relevant representations at the boundaries.

There are a number of ways in which to develop a boundary aware synthesis algorithm. Consider synthesizing textures similar to the one shown in Figure 7.3. One synthesis possibility is to choose a statistical approach that offers a quantitative platform for later measuring visual appearances of synthesized textures. Steps here involve combining ideas from the pixel-based Ritter algorithm on boundary awareness with a procedural texture synthesis technique similar to that of Ijiri et al. to synthesize boundary-aware geometric element arrangements.

Figure 7.3 shows an outline of a proposed approach to the synthesis process. The same



(a) Proposed boundary handeling algorithm overview



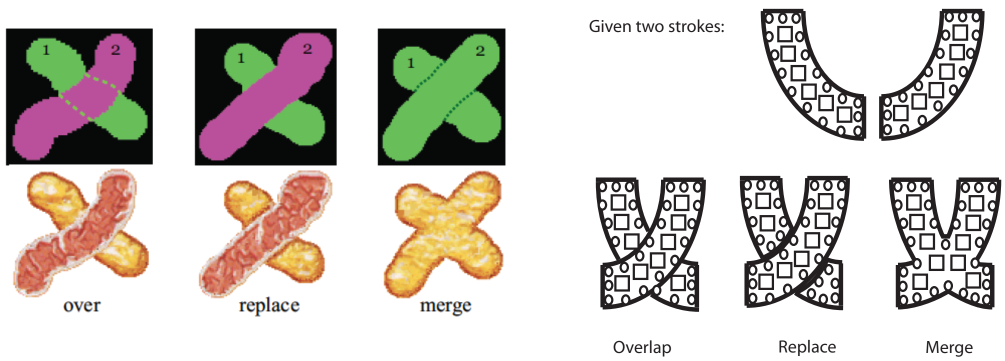
(b) The Ijiri et al. synthesis algorithm local growth overview. Each circle with an id is a seed. Seeds are replaced with a reference element that has the most similar neighbourhood.

Figure 7.3: *A visualization of a proposed boundary aware texture synthesis.*

methodology as the Ijiri et al. synthesis algorithm (details described earlier in Chapter 2) can be used to synthesize interiors of a synthesis region. When approaching boundaries, the matching algorithm is compelled to consider only reference neighbourhoods from the exemplar that include boundary motifs. The final matched motifs fill empty boundary spaces resulting in a close approximation to the exemplar boundaries. We may also consider internal features like labels within maps as boundaries and synthesize around them.

There is no reliable way to anticipate similar element appearance at either of these boundary types. More research is needed to ensure that placed boundary locations are aware of other existing boundary elements and that synthesized borders reflect those found in the reference sample. Through personal experimentation, I found that re-implementing the Ijiri et al. algorithm was difficult. Synthesized results were not similar to the exemplar distributions and a great deal of tweaking had to be done to accommodate different input styles. Developing a more robust approach to this algorithm may be highly rewarding.

In addition to texturing the 2D plane, specific interesting challenges arise when boundary



(a) Layering modes, Ritter et al..

(b) Proposed handeling of geometric stroke interactions synthesis.

Figure 7.4: *Stroke interactions.*

aware methods are applied in the context of filling strokes. Since every stroke is bounded on both sides it is clear that considering boundaries at synthesis will involve accounting for special cases like high curvature and corners. To achieve pleasing stroke boundaries, the primary step would be to develop an interactive rendering system that allows for geometric elements to be placed in tighter bounded spaces. Ritter et al. introduced this same concept in their interactive system for pixel-based texture synthesis of painted brush strokes.

It is particularly interesting to develop a layering model similar to that of Ritter et al. to allow users to manipulate strokes, whether it be to **overlap** strokes, **replace**, or **merge** them at an intersections interactively, as shown in Figure 7.4. The resulting textures should consistently maintain successful representation of appropriate boundary elements found in the reference arrangement for all separate and connected texture regions. Here is a breakdown of the three modes offered for user stroke interaction:

- Overlap. This layering mode is used when the user draws two strokes each on a different layer. The upper layer and the underlying layer may be synthesized in parallel. For further speed, it is interesting to identify the overlapped area and exclude it from the synthesis process.
- Replace. This mode applies when the user draws a different textured stroke on top of another within the same layer, creating a break in the underlying stroke. In this case

the stroke placed first is divided up; its boundaries are split to create two regions on both sides of the intersected second stroke. The resulting regions are then synthesized separately.

- Merge. This mode applies when the user draws a stroke over another existing stroke on the same layer with the same texture. The system merges the two stroke borders to form a joint intersection. The resulting merged region is then synthesized by placing a seed at any location.

A boundary aware synthesis algorithm can be sufficient for handling small, compact geometric elements at synthesis; however, one obvious limitation that needs to be addressed is the synthesis of longer, continuous elements. Lines and curves for example offer a deeper challenge for this sort of synthesis algorithm. The method by Ijiri et al. does not have a direct solution that takes the synthesis of continuous element strips into account. Brunn et al. [22] apply a multi-resolution method that specifically captures longer stroke styles from user input for effective curve synthesis. They achieve interestingly pleasing results by separating the path (the direction) from the style (the gesture) of a given stroke and re-assembling variations of the gesture along alternate paths. This approach could be incorporated into GTS algorithms along with an efficient element recognition step done at the analysis phase.

7.2 Gestalt-based geometric texture synthesis

Despite the various approaches to synthesizing GTS arrangements [5, 14, 54, 91], there may still be many more yet to be discovered. I believe that an interesting GTS algorithm could be developed from concepts in Gestalt vision and grouping theory (previously explained in Chapter 2). Gestaltists have closely studied perceptual grouping and have subsequently derived a set of classical principles. These include grouping by similarity in colour, size, orientation, common fate, symmetry, parallelism, continuity, and closure.

Despite the fact that Gestalt grouping principles fail as scientific explanations of how the visual system is structured to view the world, there have been attempts to explore its relevance in Computer Science and Graphics. For example, Liu et al. [88, 89] offered an interesting hypothesis about human perceptual organization of periodic patterns and used it to identify regularity in textures. The example-based method by Nan et al. [100] described spatial arrangements in architectural vector drawings using graphs that highlight Gestalt grouping principles. They devised energy metrics which are composed of spatial descriptors that capture structural content in architectural drawing exemplars. They then optimize an energy function composed of multiple Gestalt cost metrics to generate more of the exemplar. Although abstract, this method produced pleasing results and highlights the effectiveness of Gestaltism in describing arrangements, hence my interest in extending it to GTS.

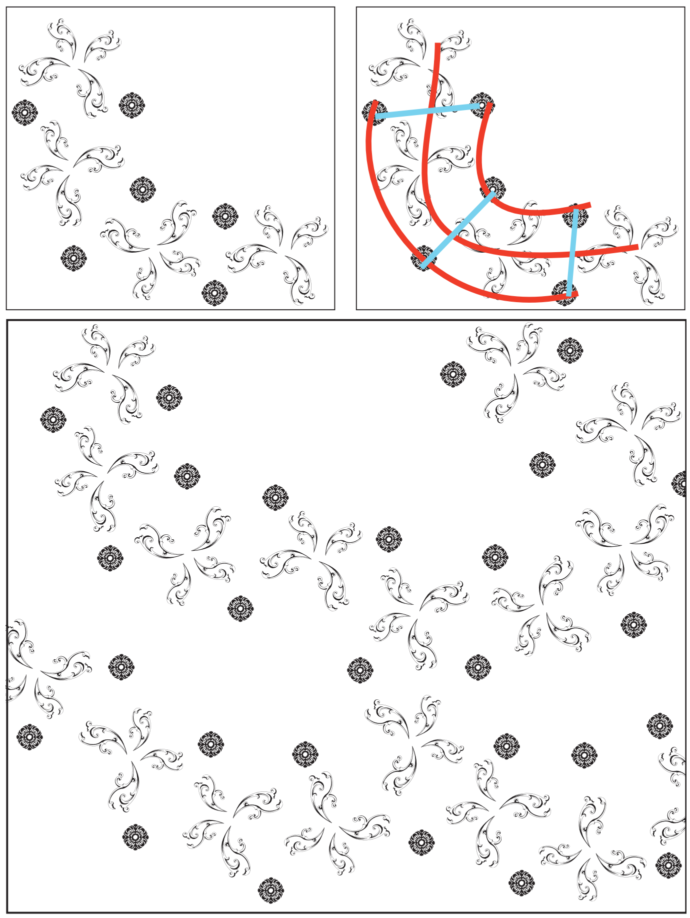


Figure 7.5: *Geometric texture synthesis using a Gestalt grouping model.*

I was inspired by symmetry detection [139], Gestalt modelling [100] and a procedural synthesis approach similar to the Deco style synthesis by Měch and Miller [96] to develop a different GTS synthesis method that can account for complex structures.

Based on grouping theory, a geometric exemplar made of multiple motifs can be decomposed into subgroups, either by the proximity of its motifs, colours, or local and global symmetries. Symmetry here is a very important factor that can help maintain highly structured exemplars similar to the 2D work by Yeh and Měch [139]. Once gathered, motif relationships can be decoded into association graphs and symmetries can be used to guide the synthesis of larger geometric textures that follow similar characteristics. The same symmetries can then be used to analyze results to compare with other larger arrangements.

A simple example illustrating Gestalt detection applied to a geometric arrangements is shown in Figure 7.5. Here the exemplar analysis step would involve taking the vector shapes that are relatively close in their locations like the four small pieces making up butterfly wings and representing them as whole motifs. These motifs have reflectional symmetry and were intentionally drawn to follow a curved path as illustrated. Aligning paths to geometry has been shown previously to make continuity and symmetry detection simpler [139]. This same methodology can be used to ensure continuity in GTS. The smaller circular ornaments along side the butterflies have mirror-like reflections and they too follow two other paths of their own. The overall paths along with their maximum and minimum curvatures can be used as guides in larger areas. Multiple motifs can be copied and placed into the synthesis space using a jigsaw-like approach with the aid of an iterative checking step to ensures that all placements conform to existing symmetries and curvatures. This is repeated until the space is filled.

The only limitation with such an algorithm is overcoming issues with symmetry detection. Although many solutions target only subsets of symmetries [97, 139], the task of detecting all symmetries and deciding which ones are prominent and which ones are not is a difficult problem. Including the user in the loop may resolve this issue and ensure that desired effects are captured in the final synthesis.

7.3 2D Shape synthesis and retrieval

As noticed from the various GTS algorithms reviewed in this dissertation, developing a compact representation of the relationships between motifs in an input exemplar is one of the most important steps towards successfully synthesizing perceptually similar arrangements. The representations generally capture the types, locations and distances of motifs to one another in an arrangement. When described this way, GTS analysis closely resembles the fundamental analysis goals of shape analysis.

Studying shape structure and the relationships between parts is currently an emerging topic in computer graphics [98]. The process is decomposed into shape analysis and synthesis.

Shape analysis involves understanding the underlying structure of a shape by decomposing the object into multiple pieces. Shape synthesis involves developing models from that representation and impose variations to it usually by exchanging parts with similar ones gathered from a dataset of domain specific labelled objects. Most tools that automatically generate new plausible shapes have commonly targeted 3D objects and involve machine learning [68, 138]. Shape synthesis has also been applied to synthesize 3D arrangements of objects [39]. However, few attempts have been made for 2D object synthesis.

In GTS, Hurtut et al. [54] and Alves dos Passos et al. [5] include an input classification step in their algorithms that acts as a shape labeller. Classification involves defining relative characteristics of the motifs to identify similar ones and group them into suitable categories accordingly. A natural step would be to generate new variations from the motifs in these shape categories in a similar manner to Baxter and Anjyo [15]. Given a set of 2D exemplars, their method proposes an interpolation between parts to generate new results. Their synthesized results are promising but require extensive user inputs for highlighting stroke correspondences. A second attempt to produce variations of 2D objects is the doodle synthesis method by Hurtut and Landes [55]. They capture the structure of a 1D doodle stroke through an inclusion tree of the various stroke overlaps. They then exchange strokes that have a similar shapes and placements in the tree to achieve variations.

A similar approach would inevitably enhance geometric arrangement synthesis algorithms by introducing variety and distinctiveness into their synthesized arrangements. With some research it appears that there are two ways to tackle this problem: shape representation can be expressed using either: shape grammars or graphs. Although shape grammars have been successful as shape representations [84], 2D graphs look like a more suitable fit for vector geometry.

The problem can be summarized as follows: given a set of 2D vector objects from a specific domain (flowers, cars, chairs, etc.), generate a new set of unique figures that preserve the overall structure. When working with 2D vector graphics it is important to use an easy suitable language to parse and understand the geometry. As previously adopted in Chapter 4, an SVG format along with the `<g>` tag seem most appropriate. These tags help by providing preliminary groupings of parts within the objects; other machine learning algorithms could be investigated to automate this initial step. A chair for example can be divided into the back, seat and legs by grouping the geometries together in the SVG (see Figure 7.6). When parsing in each of the input exemplar 2D objects, various properties are calculated and added to their graph representations such as distances between the convex hulls of groups of strokes, the angle between the subparts, the area of each part, and shape contexts [16]. These same properties are also used to formulate an energy-based similarity measure to judge the overall match quality of the final synthesized result and the input exemplars.

An illustration of the matching and synthesis steps is shown in Figure 7.7. Although these objects are pleasing, many problems exist with this approach. Some of the most notable ones include the need to understand object perspectives in 2D and accommodating for inappropriately sized part replacements. Further investigations into these areas could lead to more efficient methods for 2D shape synthesis.

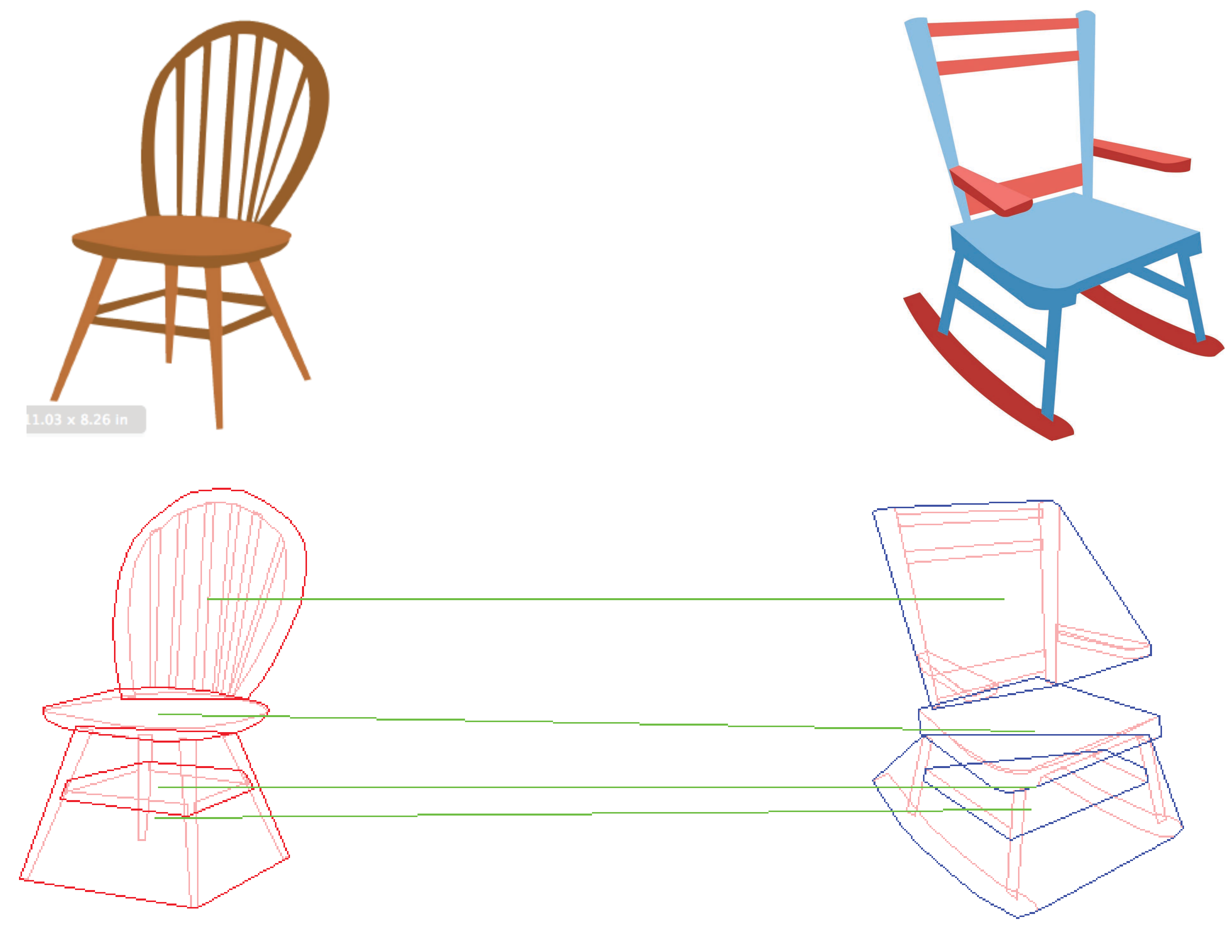


Figure 7.6: *Example chairs with grouped parts using predefined SVG* `<g>` *tags and their correspondence matching.*

Similar to the research by Xu and Kalogerakis [68, 138], synthesis of 3D geometry is becoming very compelling. But only minimal efforts have been made to retrieve vector geometry and distributions of geometry that are not necessarily labelled from databases.

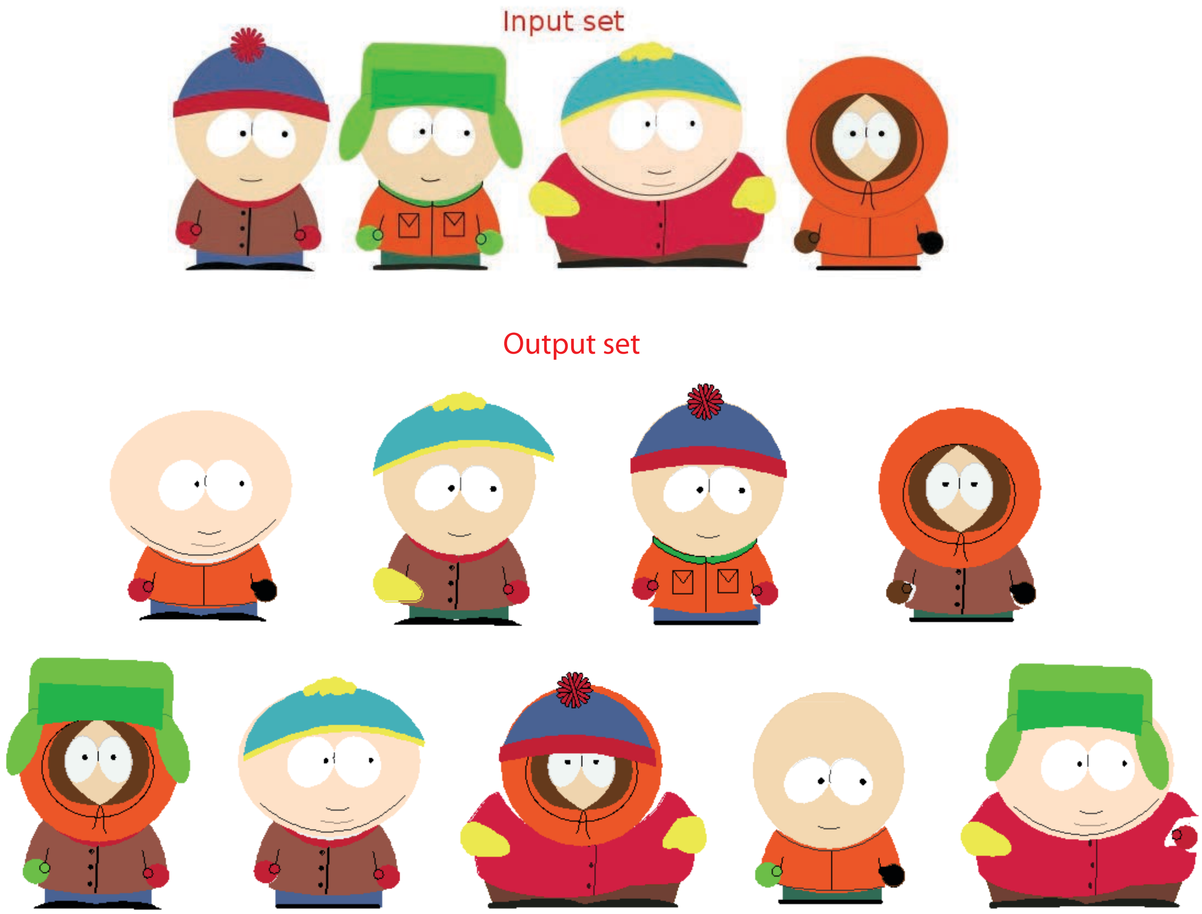


Figure 7.7: *An exemplar set and preliminary synthesis results. The character's hand on the bottom right shows a flaw in the part substitution step.*

In the raster-based area on the other hand, the method by Landes and Soler [79] detects and extracts shapes from a sample pixel image using SIFT (scale invariant feature-transform) descriptors, without the need for any a priori vector knowledge. Similarly, Google's complex SIFT measure is also efficient at capturing and classifying elements. Another example is the sketch based image retrieval by Sousa and Fonesca [117].

The most relevant active area that researches better 2D vector retrieval is multimedia systems. Two examples of such work include the graph inspired shape representations and retrieval by Demirci [30] and the point set retrieval through discrepancy by Shoa et al. [115]. The application presented by Pang [104] offers an interesting way of capturing visual similarity based on a scalar metric and showing retrieved results that match this value appropriately. Instead of laying out arrangements based on their features, it would be more interesting to start with an example and then retrieve similar ones from the dataset. Along these lines, I believe it would be worthwhile to create a geometric arrangement dataset from the USGS database like the one presented in Chapter 5; then given an exemplar, find appropriate solutions to gather arrangements that have similar motifs and distributions.

APPENDICES

Appendix A

Glossary: Definitions and classifications

This appendix provides definitions for the different terms and categorization fields used throughout the report. These definitions are meant to have long term applicability for newly proposed frameworks and algorithms in texture synthesis. The aim here is to provide descriptions that are clear, concise and resilient enough to allow flexibility for this framework to evolve.

Algorithmic:

A procedural method which assumes that 2D or 3D texture is generated on the fly algorithmically. It achieves realistic representations of natural elements such as wood, marble or stone using texture generating functions that are commonly implemented using noise functions.

Analysis:

A computational process that estimates the underlying generation of a given finite texture sample. The estimated process should be able to model both the structural and stochastic parts of the input texture. The success of the model is determined by the visual fidelity of the synthesized textures with respect to the given samples.

Anisotropic

Properties of a texture that depend on the direction.

Appearance-based:

A synthesis method that uses example-based texture synthesis to synthesize each frame, and enforces visual similarity to ensure that the synthesis remains close to the example texture.

Arrangement:

A texture described as one or multiple sets of elements distributed on a 2D plane.

Example-based synthesis: A synthesis process that accepts a user supplied texture, then synthesizes a new texture that, when perceived by a human observer appears to be generated by the same underlying process.

Frequency domain synthesis technique: A technique able to produce a limited subset of textures based on parametric models of human perception.

Geometric texture synthesis:

A the process of capturing spatial and element interaction information existing in a user-specified example consisting of vector-based 2D/3D element arrangements, and faithfully reproducing them to generate new resembling arrangements.

Isotropic

Properties of a texture elements are identical in all directions

Markov Random Fields (MRF): A set of random/stochastic variables that have a Markov property described using an undirected graph with nodes and edges. In it, future states depend only upon the present states and not on prior ones. When the probability distribution is positive it is called a Gibbs random field.

Mixed texture:

A texture that has a mixture of deterministic and stochastic characteristics (e.g., woven fabric, wood grain, plowed fields).

Motif:

A vector shape that represents an element in a 2D texture arrangement.

Near-regular texture:

A statistical departure from regular textures along different dimensions.

Non-parametric method:

A method that involves iterative algorithms that work with neighbourhood comparisons between reference and target textures. The size and shape of the neighbourhoods vary from one technique to the other. Some methods work in scan line order, while others grow the texture from a central starting point spiralling outwards, or they may be done procedurally. The synthesis results are usually more convincing and the types of synthesized textures are more general.

Parametric method:

A method that provides a compact description of textures. They make use of statistical analysis to characterize an input texture by a set of parameters, and then attempt to synthesize similar textures with similar properties in order to validate the parametric model.

Patch-based/Particle-based:

A synthesis method that uses a larger window of pixels at texture generation, and are able to preserve the global structure of the texture as well as its local properties.

Procedural and example-based:

A synthesis process that accepts a user supplied texture, then synthesizes a new texture using a procedural model. When complete, the texture appears to a human observer to have been generated by the same underlying process as the original input texture.

Procedural synthesis:

A synthesis process that relies on a user-supplied explicit model for constructing textures instead of a finished example. The algorithm that produced the user-supplied model in this case is known and can be defined as a function over space. Procedural approaches can be generally grouped into either algorithmic or prototype-based synthesis styles.

Prototype:

A special case of procedural methods where the generating model is simply a single copy of part of a pattern, together with a rule for stamping out copies of the part.

Raster-based synthesis:

A synthesis method that uses neighbourhood comparisons between example and generated textures. They are able to capture the local statistics of a texture and regenerate them very well.

Regular/Deterministic texture:

A texture that consists of a periodic pattern. The colour and shape of the texture elements repeat in equal intervals along two linearly independent directions.

Statistical synthesis:

A synthesis method that produces textures that conform to a set of numerical properties captured from a user-specified texture.

Stochastic texture:

A texture which does not have easily identifiable primitives (e.g., granite, bark, sand).

Synthesis:

A method that produces new textures from a given analysis model.

Texture:

A mapped rectangular image containing arbitrary patterns that are either regular, irregular or near-regular.

Texture spectrum:

A continuous spectrum of textures that vary gradually in their regularity between two extremes (stochastic and regular); proposed by Lin et al. [85].

Texture synthesis:

A process generally decomposed into two main components for simulating texture, analysis and synthesis. Analysis: Given a finite texture sample as input, estimate the underlying generation process. Synthesis: Develop an algorithm able to produce new textures from the derived analysis model. The results of this algorithm should appear to be generated by the same underlying process.

Tiling domain synthesis technique:

A texture synthesis technique that directly copies image patches from an example input texture and stitches them together in a newly synthesized image. Although they produce more compelling synthesized textures than pixel-based methods, they are limited in the texture styles they allow (mainly supporting only non-homogeneous textures).

Appendix B

Continued GTS evaluation probabilities

The charts shown below are the number of times participants made a most similar selection for GTS synthesized arrangements in comparison to an exemplar for all four source types (LEAVES, CURVIES, PARALLEL and SWAMP) as presented in Chapter 5.

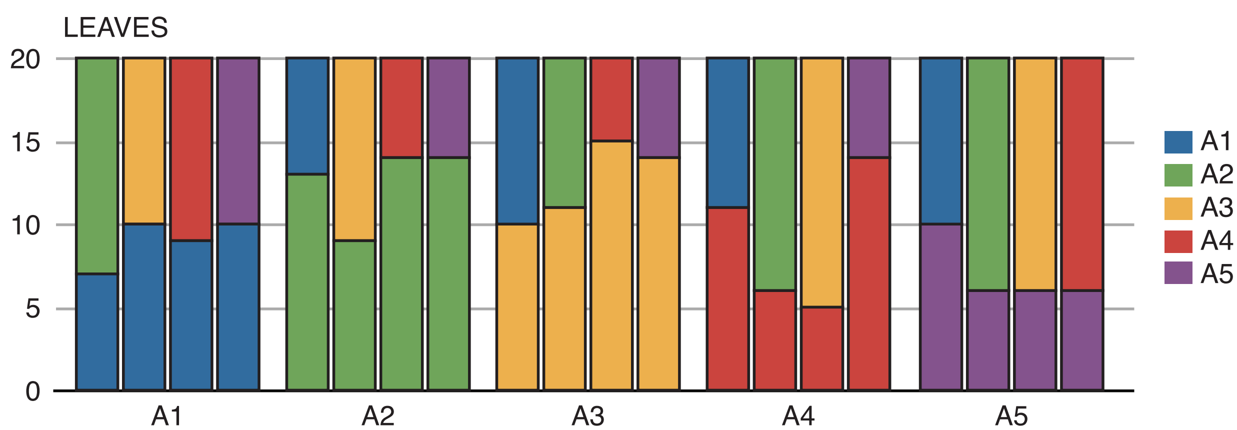


Figure B.1: *Leaves comparison selections.*

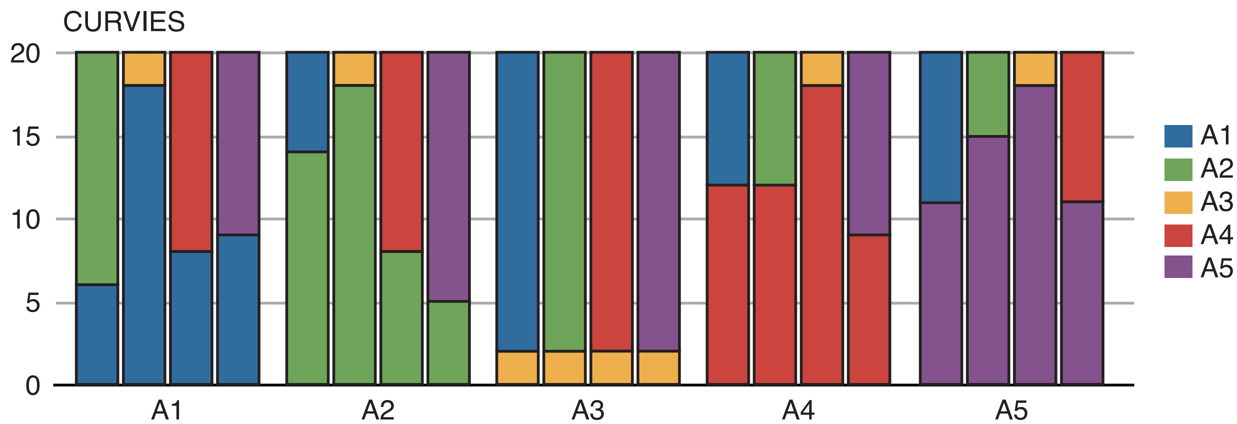


Figure B.2: *Curvies comparison selections.*

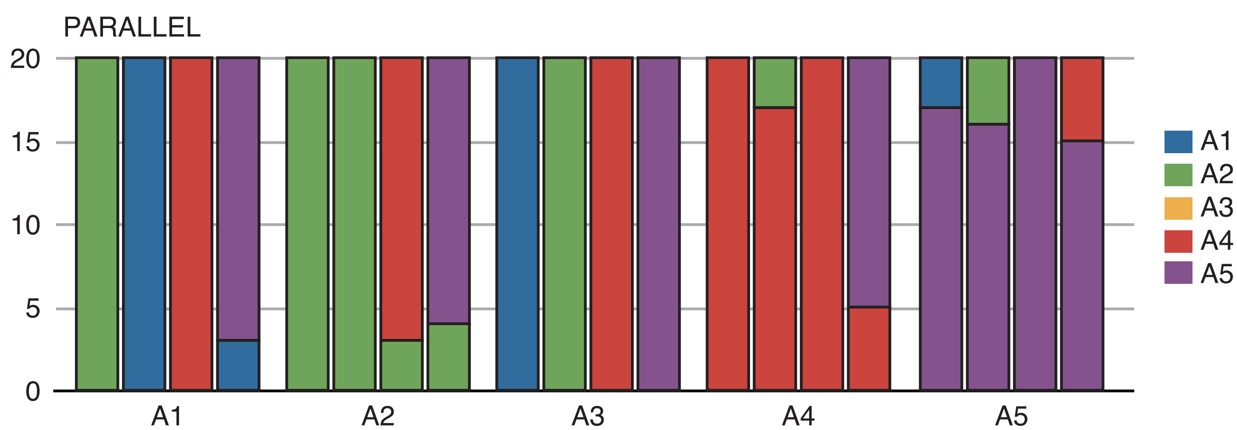


Figure B.3: *Parallel comparison selections.*

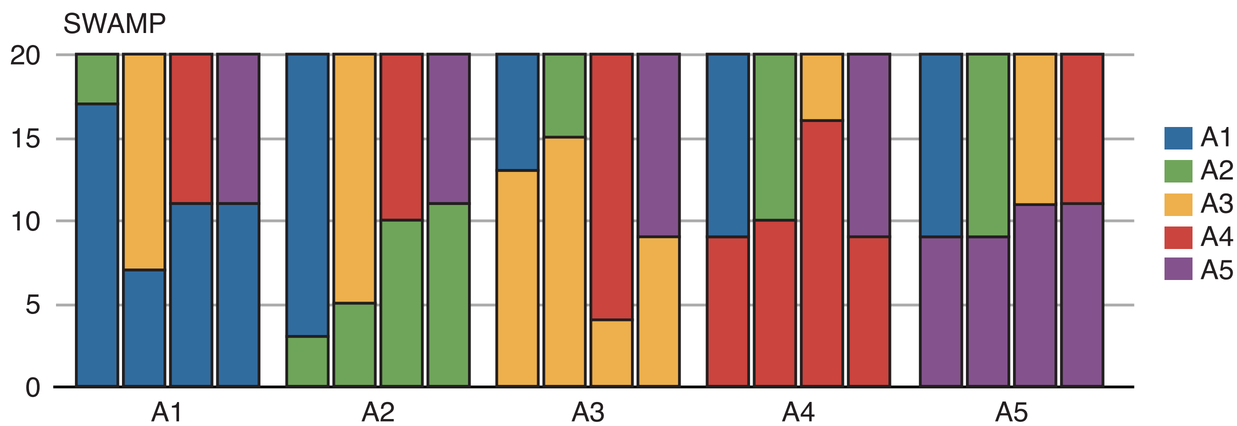


Figure B.4: *Swamp comparison selections.*

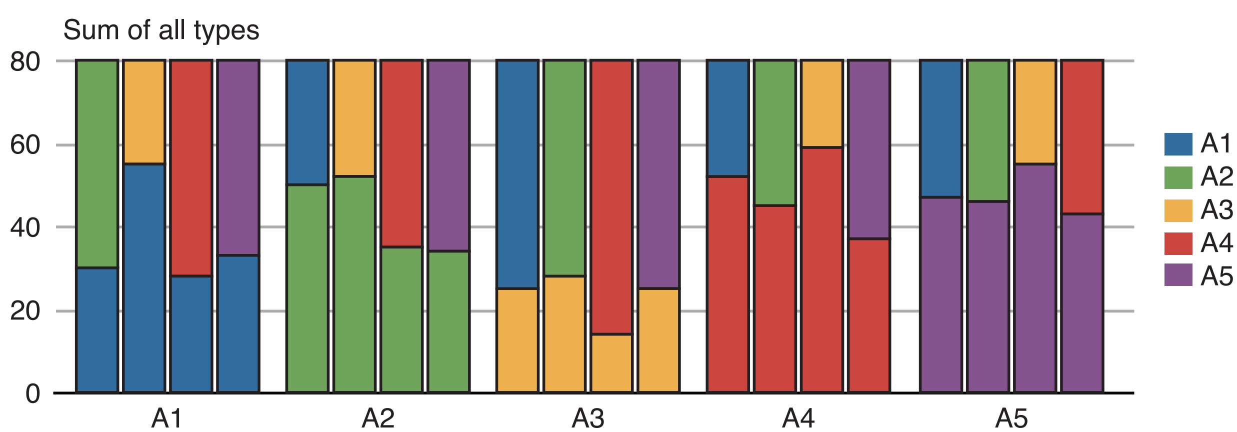


Figure B.5: *Sum of all comparison selections shown in the figures above (Figures B.1–B.4)*

Appendix C

Additional Expert Designer arrangements

In Chapter 5, I presented an analysis on a pile-sorting study conducted using four expert designer synthesized results amongst a group of arrangements synthesized by other sources. After the study was complete two other designers volunteered. Both expert designer submitted new sets of synthesized results. The fifth expert, identified as **H5**, has 15 years of experience and works under the title "Graphics/Interaction Designer". The sixth expert identified as **H6**, has more than six years of experience as a "Graphics Designer and teacher". I subsequently added these arrangements to the dataset shown in Figure C.1.

While preparing the synthesis results for the pile-sorting study, I cropped the dataset expert arrangements to a slightly smaller square boundary so that they match the sizes of arrangements gathered from the authors of GTS algorithms. The arrangements in Figure C.1 are all the original uncropped arrangements received from expert designers in the template. The arrangement set by expert designer **H2** was the only one effected most by this cropping since they were synthesized directly across borders.

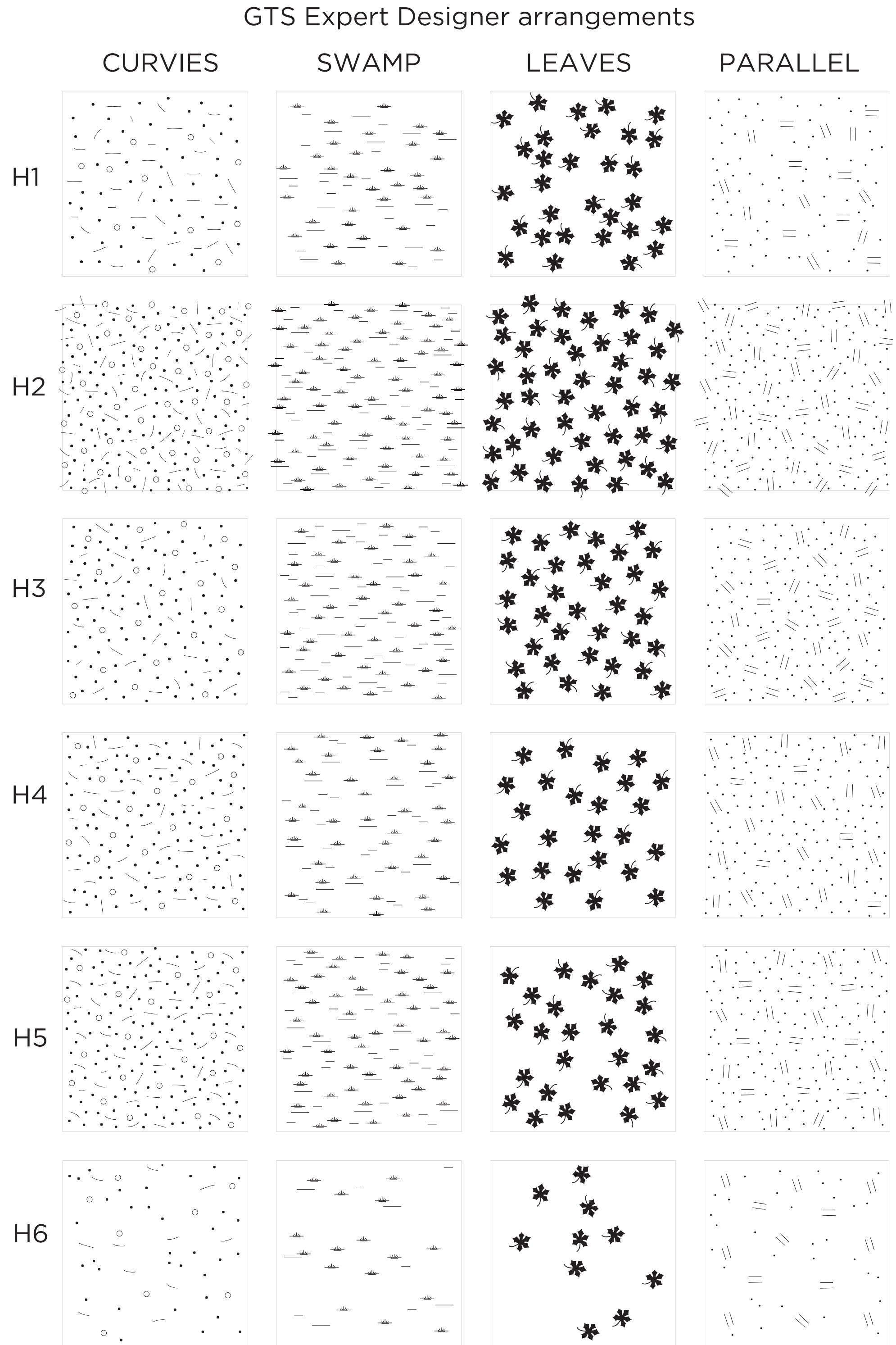


Figure C.1: *Expert Designer synthesized arrangements according to pattern source exemplars as collected from the provided user study template.*

Appendix D

Continued GTS quantitative measures

The following tables list the quantitative measures calculated for all arrangements in the GTs dataset proposed in Chapter 5, which are shown again in Figure D.1 with their corresponding Fourier power and phase spectra.

Measure	input sample	H1	H2	H3	H4	H5	H6	A1	A2	A3	A4	A5	R1	R2
Geometric density	0.02882	0.0129	0.03309	0.02015	0.02755	0.02755	0.0083	0.02288	0.02953	0.0051	0.0300	0.0300	0.0255	0.0250
Pixel density	0.0296	0.0131	0.0339	0.0210	0.0239	0.0289	0.0079	0.0228	0.0293	0.0216	0.0299	0.0299	0.0255	0.249
Entropy	0.1916	0.1005	0.2137	0.1470	0.1624	0.1887	0.0652	0.1566	0.1906	0.1499	0.1937	0.1937	0.1711	0.1681
Mean	0.9706	0.9864	0.9661	0.9790	0.9762	0.9711	0.9923	0.9773	0.9708	0.9765	0.9701	0.9701	0.9745	0.9751
Standard dev.	0.1558	0.1136	0.1810	0.1424	0.1525	0.1675	0.0875	0.1491	0.1625	0.1451	0.1702	0.1702	0.1575	0.1558
Cross Corr. Coeff.	1	0.0738	0.0673	0.0905	0.0756	0.0922	0.1298	0.3022	0.0793	0.1023	0.0824	0.0880	0.0960	0.0776
ACE	0	355.8	260.75	539.31	107.292	63.542	332.72	125.97	14.083	99.722	25.986	27.986	86.029	315.81

Table D.1: *Basic statistical image-based measures for* CURVIES *arrangements.*

Measure	input sample	H1	H2	H3	H4	H5	H6	A1	A2	A3	A4	A5	R1	R2
Geometric density	0.04525	0.02177	0.0430	0.03174	0.02537	0.03305	0.0097	0.0372	0.02123	0.00724	0.0397	0.0367	0.0285	0.0292
Pixel density	0.0307	0.0195	0.0393	0.0283	0.0216	0.0294	0.0089	0.0362	0.0211	0.0245	0.0374	0.0350	0.0356	0.0288
Entropy	0.1972	0.1383	0.2389	0.1856	0.1500	0.1911	0.0734	0.2240	0.1471	0.1160	0.2298	0.02185	0.2257	0.1883
Mean	0.9694	0.1383	0.2384	0.1856	0.1500	0.9707	0.9911	0.9639	0.9790	0.9755	0.9627	0.9651	0.9636	0.9712
Standard dev.	0.1721	0.1381	0.1942	0.1657	0.1452	0.1688	0.0939	0.1865	0.1434	0.1546	0.1897	0.1836	0.1874	0.1672
Cross Corr. Coeff.	1	0.0608	0.1412	0.1283	0.1764	0.2774	0.1778	0.3249	0.1593	0.1578	0.2178	0.1664	0.1595	0.1800
ACE	0	427.847	372.083	406.014	418.167	391.32	516	192.181	371.82	277.61	201.833	216.42	249.85	281.54

Table D.2: *Basic statistical image-based measures for* SWAMP *arrangements.*

Measure	input sample	H1	H2	H3	H4	H5	H6	A1	A2	A3	A4	A5	R1	R2
Geometric density	0.0262	0.0126	0.0271	0.0222	0.0210	0.0219	0.0074	0.0175	0.0205	0.0138	0.0256	0.0255	0.0283	0.0294
Pixel density	0.0261	0.0121	0.0259	0.0213	0.0201	0.0211	0.0072	0.0173	0.0200	0.0505	0.0254	0.0246	0.0280	0.0291
Entropy	0.1739	0.0943	0.1731	0.1483	0.1416	0.1475	0.0618	0.1259	0.1421	0.2882	0.1702	0.1664	0.1841	0.1900
Mean	0.9740	0.9879	0.9742	0.9788	0.9800	0.9789	0.9928	0.9827	0.9799	0.9496	0.9746	0.9754	0.9720	0.9709
Standard dev.	0.1591	0.1093	0.1587	0.1442	0.1401	0.1436	0.0847	0.1303	0.1404	0.2188	0.1573	0.1548	0.1649	0.1682
Cross Corr. Coeff.	1	0.0848	0.0801	0.1013	0.0918	0.1221	0.1423	0.2619	0.1253	0.0899	0.1234	0.0840	0.0891	0.0794
ACE	0	172.24	78.333	66.514	106.028	90.417	234.597	291.33	277.19	543.153	76.055	71.917	222.15	248.37

Table D.3: *Basic statistical image-based measures for* PARALLEL *arrangements.*

Measure	input sample	H1	H2	H3	H4	H5	H6	A1	A2	A3	A4	A5	R1	R2
Geometric density	0.1316	0.1398	0.2098	0.1637	0.1170	0.1210	0.0466	0.1045	0.1443	0.0917	0.1645	0.1610	0.1404	0.1363
Pixel density	0.1433	0.1391	0.2097	0.1629	0.1164	0.1210	0.0466	0.1039	0.1435	0.1308	0.1647	0.1608	0.1395	0.1381
Entropy	0.5910	0.5812	0.7404	0.6406	0.5184	0.5317	0.2713	0.4807	0.5926	0.5588	0.6447	0.6355	0.5825	0.5809
Mean	0.8574	0.8612	0.7906	0.8374	0.8837	0.8792	0.9535	0.8963	0.8568	0.8695	0.8356	0.8395	0.8607	0.8613
Standard dev.	0.3497	0.3458	0.4069	0.3690	0.3205	0.3259	0.2105	0.3049	0.3503	0.3369	0.3706	0.3671	0.3463	0.3457
Cross Corr. Coeff.	1	0.4813	0.3317	0.3561	0.3351	0.3289	0.2498	0.6184	0.3842	0.5989	0.3611	0.3197	0.2619	0.2495
ACE	0	+2000	+2000	+2000	+2000	+2000	+2000	+2000	+2000	858.86	640.28	683.72	+2000	+2000

Table D.4: *Basic statistical image-based measures for* LEAVES *arrangements.*

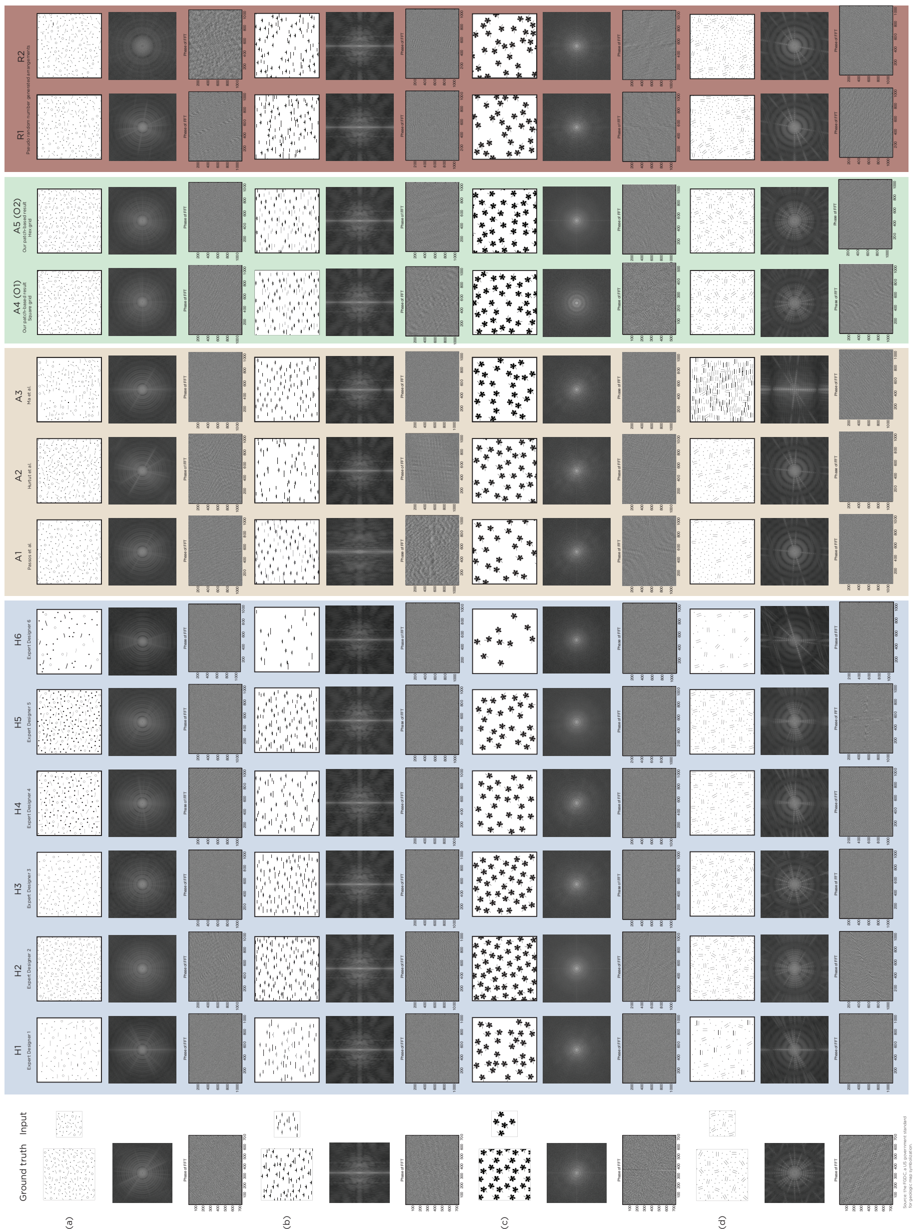


Figure D.1: *Fast-Fourier Power transforms and phase plots for the GTS dataset*

Bibliography

[1] Zainab AlMeraj, Craig S. Kaplan, and Paul Asente. Patch-based geometric texture synthesis. In *Proceedings of the Symposium on Computational Aesthetics*, CAE '13, pages 15–19, New York, NY, USA, 2013. ACM.

[2] Zainab AlMeraj, Craig S. Kaplan, and Paul Asente. Towards effective evaluation of geometric texture synthesis algorithms. In *Proceedings of the Symposium on Non-Photorealistic Animation and Rendering*, NPAR '13, pages 5–14, New York, NY, USA, 2013. ACM.

[3] Zainab AlMeraj, Craig S. Kaplan, Paul Asente, and Edward Lank. Towards ground truth in geometric textures. In *NPAR*, pages 17–26, New York, NY, USA, 2011. ACM.

[4] Zainab AlMeraj, Brian Wyvill, Tobias Isenberg, Amy A. Gooch, and Richard Guy. Automatically mimicking unique hand-drawn pencil lines. *Computers & Graphics*, 33(4):496 – 508, 2009.

[5] Vladimir Alves dos Passos, Marcelo Walter, and Mario Costa Sousa. Sample-based synthesis of illustrative patterns. In *Pacific Graphics '10*, pages 109–116, Washington, DC, USA, 2010. IEEE Computer Society.

[6] Paul J. Asente. Folding avoidance in skeletal strokes. In *SBIM '10: Proceedings of the 7th Eurographics Symposium on Sketch-Based Interfaces and Modeling*, pages 33–40. Eurographics Association, 2010.

[7] Michael Ashikhmin. Synthesizing natural textures. In *I3D '01: Proceedings of the 2001 symposium on Interactive 3D graphics*, pages 217–226, New York, NY, USA, 2001. ACM.

[8] A. Baddeley and R. Turner. `spatstat` website. URL: `www.spatstat.org`.

[9] A. Baddeley and R. Turner. Spatstat: an R package for analyzing spatial point patterns. *Journal of Statistical Software*, 12(6):1–42, 2005. URL: `www.jstatsoft.org`, ISSN: 1548-7660.

[10] A. Baddeley and R. Turner. Modelling spatial point patterns in R. In A. Baddeley, P. Gregori, J. Mateu, R. Stoica, and D. Stoyan, editors, *Case Studies in Spatial Point Pattern Modelling*, number 185 in Lecture Notes in Statistics, pages 23–74. Springer-Verlag, New York, 2006. ISBN: 0-387-28311-0.

[11] Benjamin Balas. Attentive texture similarity as a categorization task: Comparing texture synthesis models. *Pattern Recognition*, 41(3):972–982, March 2008.

[12] Benjamin J. Balas. Texture synthesis and perception: Using computational models to study texture representations in the human visual system. *Vision Research*, 46(3):299 – 309, 2006.

[13] Pascal Barla, Simon Breslav, Lee Markosian, and Joëlle Thollot. Interactive hatching and stippling by example. *CoRR*, abs/cs/0607050, 2006.

[14] Pascal Barla, Simon Breslav, Joëlle Thollot, François X. Sillion, and Lee Markosian. Stroke pattern analysis and synthesis. *Computer Graphics Forum*, 25(3):663–671, 2006.

[15] William Baxter and Ken-ichi Anjyo. Latent doodle space. *Computer Graphics Forum*, 25(3):477–485, 2006.

[16] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(4):509–522, 2002.

[17] Pierre Bénard, Joëlle Thollot, and Francois Sillion. Quality assessment of fractalized npr textures: a perceptual objective metric. In *Proceedings of the 6th Symposium on Applied Perception in Graphics and Visualization*, APGV '09, pages 117–120, New York, NY, USA, 2009. ACM.

[18] James R. Bergen and Michael S. Landy. Computational modeling of visual texture segregation. In *in Computational Models of Visual Processing*, pages 253–271. MIT Press, 1991.

[19] Pravin Bhat, Stephen Ingram, and Greg Turk. Geometric texture synthesis by example. In *SGP '04: Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, pages 41–44, New York, NY, USA, 2004. ACM.

[20] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[21] V. Bruce, P. Green, and M.A. Georgeson. *Visual Perception: Physiology, Psychology, and Ecology*. Psychology Press, 1996.

[22] Meru Brunn, Mario Costa Sousa, and Faramarz F. Samavati. Capturing and re-using artistic styles with reverse subdivision-based multiresolution methods. *Int. J. Image Graphics*, 7(4):593–615, 2007.

[23] James Michael Coggins. *A framework for texture analysis based on spatial filtering*. PhD thesis, East Lansing, MI, USA, 1983. AAI8315444.

[24] Michael F. Cohen, Jonathan Shade, Stefan Hiller, and Oliver Deussen. Wang tiles for image and texture generation. *ACM Trans. Graph.*, 22(3):287–294, July 2003.

[25] Anthony C. Copeland, Gopalan Ravichandran, and Mohan M. Trivedi. Texture synthesis using gray-level co-occurrence models: algorithms, experimental analysis, and psychophysical support. *Optical Engineering*, 40(11):2655–2673, 2001.

[26] John W. Creswell. *Qualitative Inquiry & research Design*. Sage publications; 2nd edition, 2007.

[27] Ketan Dalal, Allison W. Klein, Yunjun Liu, and Kaleigh Smith. A spectral approach to NPR packing. In *Proceedings of the 4th international symposium on Non-photorealistic animation and rendering*, NPAR, pages 71–78, New York, NY, USA, 2006. ACM.

[28] Jeremy S. De Bonet. Multiresolution sampling procedure for analysis and synthesis of texture images. In *SIGGRAPH '97: Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 361–368, New York, NY, USA, 1997. ACM Press/Addison-Wesley Publishing Co.

[29] Douglas DeCarlo and Matthew Stone. Visual explanations. In *NPAR*, pages 173–178, 2010.

[30] M.Fatih Demirci. Retrieving 2d shapes using caterpillar decomposition. *Machine Vision and Applications*, 24(2):435–445, 2013.

[31] Oliver Deussen, Stefan Hiller, Cornelius van Overveld, and Thomas Strothotte. Floating points: A method for computing stipple drawings. *Computer Graphics Forum*, 19:40–51, 2000.

[32] Jean-Michel Dischler and Djamchid Ghazanfarpour. A procedural description of geometric textures by spectral and spatial analysis of profiles. *Comput. Graph. Forum*, 16(3):129–140, 1997.

[33] Jean-Michel Dischler, Karl Maritaud, Bruno Lévy, and Djamchid Ghazanfarpour. Texture particles. *Comput. Graph. Forum*, 21(3), 2002.

[34] N. A. Dodgson. Computational aesthetics 2008: Balancing the expected and the surprising in geometric patterns. *Comput. Graph.*, 33:475–483, August 2009.

[35] David S. Ebert, F. Kenton Musgrave, Darwyn Peachey, Ken Perlin, and Steven Worley. *Texturing and Modeling: A Procedural Approach.* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2002.

[36] Alexei A. Efros and William T. Freeman. Image quilting for texture synthesis and transfer. In *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 341–346, New York, NY, USA, 2001. ACM.

[37] Alexei A. Efros and Thomas K. Leung. Texture synthesis by non-parametric sampling. In *ICCV '99: Proceedings of the International Conference on Computer Vision-Volume 2*, page 1033, Washington, DC, USA, 1999. IEEE Computer Society.

[38] Cheng en Guo, Song-Chun Zhu, and Yingnian Wu. Visual learning by integrating descriptive and generative methods. *Computer Vision, IEEE International Conference on*, 1:370, 2001.

[39] Matthew Fisher, Daniel Ritchie, Manolis Savva, Thomas Funkhouser, and Pat Hanrahan. Example-based synthesis of 3d object arrangements. *ACM Trans. Graph.*, 31(6):135:1–135:11, November 2012.

[40] William T. Freeman, Joshua B. Tenenbaum, and Egon C. Pasztor. Learning style translation for the lines of a drawing. *ACM Trans. Graph.*, 22(1):33–46, 2003.

[41] A.E. Gelfand, P. Diggle, P. Guttorp, and M. Fuentes. *Handbook of Spatial Statistics.* Chapman & Hall/CRC Handbooks of Modern Statistical Methods. Taylor & Francis, 2010.

[42] Alan E. Gelfand. *Handbooks of Modern Statistical Methods: Handbook of Spatial Statistics.* Chapman & Hall/CRC, Taylor and Francis, 2010.

[43] G. Gilet and J.-M. Dischler. Procedural descriptions of anisotropic noisy textures by example. In *Eurographics (Short)*, 2010.

[44] G. Gilet and J.-M. Dischler. Procedural texture particles. In *SI3D*, 2010.

[45] Ian E. Gordon. *Theories of Visual Perception*. Psychology Press, December 2004.

[46] R.M. Haralick. Statistical and structural approaches to texture. *Proceedings of the IEEE*, 67(5):786–804, 1979.

[47] David J. Heeger and James R. Bergen. Pyramid-based texture analysis/synthesis. In *SIGGRAPH '95: Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 229–238, New York, NY, USA, 1995. ACM.

[48] Hermann Helmholtz. *Selected Writings of Hermann Helmholtz*. Wesleyan University Press, 1878.

[49] Aaron Hertzmann. Non-photorealistic rendering and the science of art. In *NPAR*, pages 147–157, New York, NY, USA, 2010. ACM.

[50] Aaron Hertzmann, Charles E. Jacobs, Nuria Oliver, Brian Curless, and David H. Salesin. Image analogies. In *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 327–340, New York, NY, USA, 2001. ACM.

[51] Aaron Hertzmann, Nuria Oliver, Brian Curless, and Steven M. Seitz. Curve analogies. In *EGRW '02: Proceedings of the 13th Eurographics workshop on Rendering*, pages 233–246, Aire-la-Ville, Switzerland, Switzerland, 2002. Eurographics Association.

[52] H.P. Hsu. *Schaum's outline of theory and problems of signals and systems*. Schaum's outline series. McGraw-Hill, 1995.

[53] S. C. Hsu, I. H. H. Lee, and N. E. Wiseman. Skeletal strokes. In *UIST '93: Proceedings of the 6th annual ACM symposium on User interface software and technology*, pages 197–206, New York, NY, USA, 1993. ACM.

[54] T. Hurtut, P.-E. Landes, J. Thollot, Y. Gousseau, R. Drouillhet, and J.-F. Coeurjolly. Appearance-guided synthesis of element arrangements by example. In *NPAR*, pages 51–60, New York, NY, USA, 2009. ACM.

[55] Thomas Hurtut and Pierre-Edouard Landes. Synthesizing structured doodle hybrids. In *SIGGRAPH Asia 2012 Posters*, SA '12, pages 43:1–43:1, New York, NY, USA, 2012. ACM.

[56] Takashi Ijiri, Radomír Měch, Takeo Igarashi, and Gavin Miller. An example-based procedural system for element arrangement. *Computer Graphics Forum.*, 27(2):429–436, 2008.

[57] Tobias Isenberg. Evaluating and validating non-photorealistic and illustrative rendering. In Paul Rosin and John Collomosse, editors, *Image and Video-Based Artistic Stylisation*, volume 42 of *Computational Imaging and Vision*, pages 311–331. Springer London, 2013.

[58] Tobias Isenberg, M. Sheelagh T. Carpendale, and Mario Costa Sousa. Breaking the Pixel Barrier. In László Neumann, Mateu Sbert Casasayas, Bruce Gooch, and Werner Purgathofer, editors, *Proceedings of the First Eurographics Workshop on Computational Aesthetics in Graphics, Visualization and Imaging 2005 (May 18–20, 2005, Girona, Spain)*, pages 41–48, Aire-la-Ville, Switzerland, 2005. Eurographics Association.

[59] Tobias Isenberg, Petra Neumann, Sheelagh Carpendale, Mario Costa Sousa, and Joaquim A. Jorge. Non-Photorealistic Rendering in Context: An Observational Study. In *NPAR*, pages 115–126, New York, NY, USA, 2006. ACM.

[60] Robert Jagnow, Julie Dorsey, and Holly Rushmeier. Stereological techniques for solid textures. In *SIGGRAPH '04: ACM SIGGRAPH 2004 Papers*, pages 329–335, New York, NY, USA, 2004. ACM.

[61] Bernhard Jenny, Ernst Hutzler, and Lorenz Hurni. Point pattern synthesis. *The Cartographic Journal*, 47(3):257–261, 2010.

[62] Pierre-Marc Jodoin, Emric Epstein, Martin Granger-Piché, and Victor Ostromoukhov. Hatching by example: a statistical approach. In *NPAR*, pages 29–36, New York, NY, USA, 2002. ACM.

[63] B. Julesz. Visual pattern discrimination. *IT*, 8(2):84–92, February 1962.

[64] B. Julesz. Texton gradients: The texton theory revisited. *Biological Cybernetics*, 54:245–251, 1986.

[65] B. Julesz, E N Gilbert, L A Shepp, and Frisch HL. Inability of humans to discriminate between visual textures that agree in second-order statistics – revisited. 2:391–405, 1973.

[66] Bela Julesz. Textons, the elements of texture perception, and their interactions. *Nature*, 290(5802):91–97, March 1981.

[67] Robert D. Kalnins, Lee Markosian, Barbara J. Meier, Michael A. Kowalski, Joseph C. Lee, Philip L. Davidson, Matthew Webb, John F. Hughes, and Adam Finkelstein. Wysiwyg npr: drawing strokes directly on 3d models. In *SIGGRAPH '02: Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 755–762, New York, NY, USA, 2002. ACM.

[68] Evangelos Kalogerakis, Siddhartha Chaudhuri, Daphne Koller, and Vladlen Koltun. A probabilistic model for component-based shape synthesis. *ACM Trans. Graph.*, 31(4):55:1–55:11, July 2012.

[69] Samuel Kaski, Janne Nikkil, Merja Oja, Jarkko Venna, Petri Trnen, and Eero Castrn. Trustworthiness and metrics in visualizing similarity of gene expression. *BMC Bioinformatics*, 4(1):1–13, 2003.

[70] Johannes Kopf, Daniel Cohen-Or, Oliver Deussen, and Dani Lischinski. Recursive wang tiles for real-time blue noise. In *ACM SIGGRAPH 2006 Papers*, SIGGRAPH '06, pages 509–518, New York, NY, USA, 2006. ACM.

[71] Ilona Kovacs and Bela Julesz. Perceptual sensitivity maps within globally defined visual shapes. In *Nature*, volume 370, pages 644–646. Nature Publishing Group, 1994.

[72] Vivek Kwatra, Irfan Essa, Aaron Bobick, and Nipun Kwatra. Texture optimization for example-based synthesis. *ACM Trans. Graph.*, 24(3):795–802, 2005.

[73] Vivek Kwatra, Arno Schödl, Irfan Essa, Greg Turk, and Aaron Bobick. Graphcut textures: image and video synthesis using graph cuts. *ACM Trans. Graph.*, 22(3):277–286, 2003.

[74] Ares Lagae, Olivier Dumont, and Philip Dutré. Geometry synthesis by example. In *SMI '05: Proceedings of the International Conference on Shape Modeling and Applications 2005*, pages 176–185, Washington, DC, USA, 2005. IEEE Computer Society.

[75] Ares Lagae and Philip Dutré. A procedural object distribution function. *ACM Trans. Graph.*, 24(4):1442–1461, 2005.

[76] Ares Lagae, Sylvain Lefebvre, Rob Cook, Tony DeRose, George Drettakis, D.S. Ebert, J.P. Lewis, Ken Perlin, and Matthias Zwicker. State of the art in procedural noise functions. In Helwig Hauser and Erik Reinhard, editors, *EG 2010 - State of the Art Reports*. Eurographics, Eurographics Association, May 2010.

[77] Ares Lagae, Sylvain Lefebvre, George Drettakis, and Philip Dutré. Procedural noise using sparse gabor convolution. In *SIGGRAPH '09*, pages 1–10, New York, NY, USA, 2009. ACM.

[78] Pierre-Edouard Landes, Bruno Galerne, and Thomas Hurtut. A shape-aware model for discrete texture synthesis. In *Computer Graphics Forum (Proceedings of EGSR 2013)*, volume 32, 2013.

[79] Pierre-Edouard Landes and Cyril Soler. Content-Aware Texture Synthesis. Research Report RR-6959, INRIA, 2009.

[80] Sylvain Lefebvre and Hugues Hoppe. Parallel controllable texture synthesis. *ACM Trans. Graph.*, 24(3):777–786, July 2005.

[81] Sylvain Lefebvre and Hugues Hoppe. Appearance-space texture synthesis. In *SIGGRAPH '06: ACM SIGGRAPH 2006 Papers*, pages 541–548, New York, NY, USA, 2006. ACM.

[82] J. P. Lewis. Generalized stochastic subdivision. *ACM Trans. Graph.*, 6(3):167–190, 1987.

[83] J. P. Lewis. Algorithms for solid noise synthesis. In *SIGGRAPH '89: Proceedings of the 16th annual conference on Computer graphics and interactive techniques*, pages 263–270, New York, NY, USA, 1989. ACM.

[84] M. Leyton. A process-grammar for shape. *Artif. Intell.*, 34(2):213–247, March 1988.

[85] Wen-Chieh Lin, James Hays, Chenyu Wu, Yanxi Liu, and Vivek Kwatra. Quantitative evaluation of near regular texture synthesis algorithms. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 1*, CVPR '06, pages 427–434, Washington, DC, USA, 2006. IEEE Computer Society.

[86] Aristid Lindenmayer. Mathematical models for cellular interactions in development ii. simple and branching filaments with two-sided inputs. *Journal of Theoretical Biology*, 18(3):300 – 315, 1968.

[87] Dongwei Liu, Junsong Zhang, and Changle Zhou. Perceptually-based Stroke Pattern Synthesis. In Bing-Yu Chen, Jan Kautz, Tong-Yee Lee, and Ming C. Lin, editors, *Pacific Graphics 2011*, pages 13–17, Kaohsiung, Taiwan, 2011. Eurographics Association.

[88] Yanxi Liu, Robert T. Collins, and Yanghai Tsin. A computational model for periodic pattern perception based on frieze and wallpaper groups. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(3):354–371, 2004.

[89] Yanxi Liu, Wen-Chieh Lin, and James Hays. Near-regular texture analysis and manipulation. In *ACM Trans. Graph.*, volume 23, pages 368–376, New York, NY, USA, 2004. ACM.

[90] Chongyang Ma, Li-Yi Wei, Sylvain Lefebvre, and Xin Tong. Dynamic element textures. In *SIGGRAPH 2013*, page to appear, 2013.

[91] Chongyang Ma, Li-Yi Wei, and Xin Tong. Discrete element textures. *ACM Trans. Graph.*, 30(4):62:1–62:10, August 2011.

[92] Ross Maciejewski, Tobias Isenberg, William M. Andrews, David S. Ebert, and Mario Costa Sousa. Aesthetics of hand-drawn vs. computer-generated stippling. In *Computational Aesthetics '07*, pages 53–56, Aire-la-Ville, Switzerland, Switzerland, 2007. Eurographics Association.

[93] David Marr. Vision: A computational investigation into the human representation and processing of visual information, 1982.

[94] David Marr. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information.* W. H. Freeman, March 1983.

[95] Domingo Martín, Germán Arroyo, M. Victoria Luzón, and Tobias Isenberg. Scale-dependent and example-based grayscale stippling. *Computers & Graphics*, 35(1):160 – 174, 2011.

[96] Radomír Měch and Gavin Miller. The *Deco* framework for interactive procedural modeling. *Journal of Computer Graphics Techniques (JCGT)*, 1(1):43–99, Dec 2012.

[97] Niloy J. Mitra, Leonidas J. Guibas, and Mark Pauly. Partial and approximate symmetry detection for 3d geometry. In *ACM SIGGRAPH 2006 Papers*, SIGGRAPH '06, pages 560–568, New York, NY, USA, 2006. ACM.

[98] Niloy J. Mitra, Michael Wand, Hao Zhang, Daniel Cohen-Or, and Martin Bokeloh. Structure-aware shape processing. In *EUROGRAPHICS State-of-the-art Report*, 2013.

[99] Fanya S. Montalvo. Human vision and computer graphics. In *SIGGRAPH '79: Proceedings of the 6th annual conference on Computer graphics and interactive techniques*, pages 121–125, New York, NY, USA, 1979. ACM.

[100] Liangliang Nan, Andrei Sharf, Ke Xie, Tien-Tsin Wong, Oliver Deussen, Daniel Cohen-Or, and Baoquan Chen. Conjoining gestalt rules for abstraction of architectural drawings. *ACM Trans. Graph.*, 30(6):185:1–185:10, December 2011.

[101] U Neisser. Gibson's ecological optics: Consequences of a different stimulus description. In *Journal for the Theory of Social Behaviour*, volume 7, pages 17 – 28. Blackwell Publishing Ltd, 1977.

[102] A. Cengiz Öztireli and Markus Gross. Analysis and synthesis of point distributions based on pair correlation. *ACM Transaction on Graphics*, 31(6):170:1–170:10, November 2012.

[103] Stephen E. Palmer. *Vision science : photons to phenomenology.* MIT Press, Cambridge, Mass., 1999.

[104] Wai-Man Pang. An intuitive texture picker. In *IUI*, pages 365–368, 2010.

[105] Darwyn R. Peachey. Solid texturing of complex surfaces. In *SIGGRAPH '85: Proceedings of the 12th annual conference on Computer graphics and interactive techniques*, pages 279–286, New York, NY, USA, 1985. ACM.

[106] Ken Perlin. An image synthesizer. In *SIGGRAPH '85: Proceedings of the 12th annual conference on Computer graphics and interactive techniques*, pages 287–296, New York, NY, USA, 1985. ACM.

[107] Nico Pietroni, Paolo Cignoni, Miguel Otaduy, and Roberto Scopigno. A survey on solid texture synthesis. *IEEE Computer Graphics and Applications*, 99, 2009.

[108] Javier Portilla and Eero P. Simoncelli. A parametric texture model based on joint statistics of complex wavelet coefficients. In *Int. J. Comput. Vision*, volume 40, pages 49–70, Hingham, MA, USA, 2000. Kluwer Academic Publishers.

[109] P. Prusinkiewicz and Aristid Lindenmayer. *The algorithmic beauty of plants*. Springer-Verlag New York, Inc., New York, NY, USA, 1990.

[110] Lincoln Ritter, Wilmot Li, Maneesh Agrawala, Brian Curless, and David Salesin. Painting With Texture. In *Rendering Techniques 2006, Proceedings of the 17th Eurographics Symposium on Rendering (EGSR 2006, June 26–28, 2006, Nicosa, Cyprus)*, pages 371–376. EuroGraphics Publishers, 2006.

[111] Abdelmounaime Safia and Dong-Chen He. New brodatz-based image databases for grayscale color and multiband texture analysis. *ISRN Machine Vision*, 2013, 2013.

[112] Michael P. Salisbury, Sean E. Anderson, Ronen Barzel, and David H. Salesin. Interactive pen-and-ink illustration. In *SIGGRAPH '94: Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 101–108, New York, NY, USA, 1994. ACM.

[113] Amir Semmo, Jan Eric Kyprianidis, Matthias Trapp, and Jürgen Döllner. Real-time rendering of water surfaces with cartography-oriented design. *Proceedings International Symposium on Computational Aesthetics in Graphics, Visualization, and Imaging (CAe)*, 2013.

[114] C. E. Shannon. Prediction and entropy of printed English. *Bell Systems Technical Journal*, 30:50–64, 1951.

[115] Jie Shao, Heng Tao Shen, Zi Huang, and Xiaofang Zhou. Exploring distributional discrepancy for multidimensional point set retrieval. *Multimedia, IEEE Transactions on*, 13(1):71–81, 2011.

[116] Kaleigh Smith, Yunjun Liu, and Allison Klein. Animosaics. In *Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*, SCA '05, pages 201–208, New York, NY, USA, 2005. ACM.

[117] Pedro Sousa and Manuel J. Fonseca. Sketch-based retrieval of drawings using spatial proximity. *J. Vis. Lang. Comput.*, 21(2):69–80, April 2010.

[118] D. Stoyan and H. Stoyan. Estimating pair correlation functions of planar cluster processes. In *Biometrical journal*, volume 38, pages 259–271, 1996.

[119] Maria Stylianou-Korsnes, Miriam Reiner, Svein Magnussen, and Marcus Feldman. Visual recognition of shapes and textures: an fmri study. In *Brain Structure and Function*, volume 214, pages 355–359. Springer Berlin / Heidelberg, 2010.

[120] Mihran Tuceryan and Anil K. Jain. *Handbook of pattern recognition & computer vision*. World Scientific Publishing Co., Inc., River Edge, NJ, USA, 1993.

[121] Greg Turk. Generating textures on arbitrary surfaces using reaction-diffusion. In *Proceedings of the 18th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '91, pages 289–298, New York, NY, USA, 1991. ACM.

[122] Greg Turk. Texture synthesis on surfaces. In *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 347–354, New York, NY, USA, 2001. ACM.

[123] Christopher W Tyler. Beyond fourth-order texture discrimination: generation of extreme-order and statistically-balanced textures. *Vision Research*, 44(18):2187 – 2199, 2004.

[124] R. Ulichney. *Digital Halftoning*. MIT Press, 1987.

[125] R.A. Ulichney. Dithering with blue noise. *Proceedings of the IEEE*, 76(1):56–79, 1988.

[126] United States Federal Geographic Data Committee, Geological Data Subcommittee. *FGDC Digital Cartographic Standard for Geologic Map Symbolization*. United States Geological Survey, 2006. Viewable online at http://pubs.usgs.gov/tm/2006/11A02/FGDCgeostdTM11A2_web_all.pdf.

[127] Li-Yi Wei. Texture synthesis from multiple sources. In *SIGGRAPH '03: ACM SIGGRAPH 2003 Sketches & Applications*, pages 1–1, New York, NY, USA, 2003. ACM.

[128] Li-Yi Wei, Sylvain Lefebvre, Vivek Kwatra, and Greg Turk. State of the art in example-based texture synthesis. In *Eurographics, State of the Art Report, EG-STAR*. Eurographics Association, 2009.

[129] Li-Yi Wei and Marc Levoy. Fast texture synthesis using tree-structured vector quantization. In *SIGGRAPH '00: Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 479–488, New York, NY, USA, 2000. ACM Press/Addison-Wesley Publishing Co.

[130] Li-Yi Wei and Rui Wang. Differential domain analysis for non-uniform sampling. In *ACM SIGGRAPH 2011 papers*, SIGGRAPH '11, pages 50:1–50:10, New York, NY, USA, 2011. ACM.

[131] Susan C. Weller and Kimball Romney. *Systematic data collection*. Sage Publications, 1988.

[132] Max Wertheimer. *A source book of Gestalt psychology.* Translation published by Ellis, W. (1938). Routledge & Kegan Paul, London, 1923.

[133] Georges Winkenbach and David H. Salesin. Computer-generated pen-and-ink illustration. In *SIGGRAPH '94: Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 91–100, New York, NY, USA, 1994. ACM.

[134] Michael T. Wong, Douglas E. Zongker, and David H. Salesin. Computer-generated floral ornament. In *SIGGRAPH '98: Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 423–434, New York, NY, USA, 1998. ACM.

[135] Steven Worley. A cellular texture basis function. In *SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 291–294, New York, NY, USA, 1996. ACM.

[136] Qing Wu and Yizhou Yu. Feature matching and deformation for texture synthesis. In *ACM Trans. Graph.*, volume 23, pages 364–367, New York, NY, USA, 2004. ACM.

[137] B. Wyvill, P. G. Kry, R. Seidel, and D. Mould. Determining an aesthetic inscribed curve. In *Proceedings of the Eighth Annual Symposium on Computational Aesthetics in Graphics, Visualization, and Imaging*, CAe, pages 63–70, Aire-la-Ville, Switzerland, Switzerland, 2012. Eurographics Association.

[138] Kai Xu, Hao Zhang, Daniel Cohen-Or, and Baoquan Chen. Fit and diverse: set evolution for inspiring 3d shape galleries. *ACM Trans. Graph.*, 31(4):57:1–57:10, July 2012.

[139] Yi-Ting Yeh and Radomír Měch. Detecting symmetries and curvilinear arrangements in vector art. *Comput. Graph. Forum*, 28(2):707–716, 2009.

[140] Li yi Wei and Marc Levoy. Order-independent texture synthesis. Technical report, Technical report, 2002.

[141] Semir Zaki. *Inner Vision: an exploration of art and the brain*. Oxford University Press, 1998.

[142] Semir Zeki. *A Vision of the Brain*. Wiley-Blackwell, August 1993.

[143] Kun Zhou, Xin Huang, Xi Wang, Yiying Tong, Mathieu Desbrun, Baining Guo, and Heung-Yeung Shum. Mesh quilting for geometric texture synthesis. *ACM Trans. Graph.*, 25(3):690–697, July 2006.