A Comparison Study of Four Texture Synthesis Algorithms on Near-regular Textures

Wen-Chieh Lin* James Hays* Chenyu Wu* Vivek Kwatra† Yanxi Liu*
Carnegie Mellon University* Georgia Institute of Technology†

1 Introduction

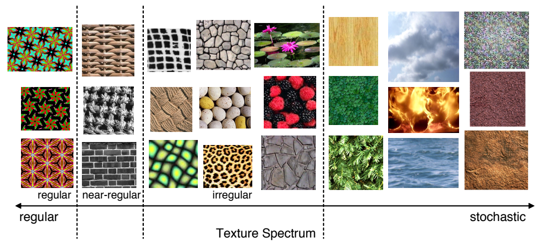

Figure 1: A texture spectrum on which textures are arranged according to their structural regularity.

Textures are conventionally classified as either regular or stochastic. However, many real-world textures fall somewhere in-between these two extremes. Most textures, along with regular and stochastic textures, form a texture spectrum on which the structural regularity varies continuously towards randomness(Figure 1). Ideally, a good texture synthesis algorithm should be able to handle all types of textures on the spectrum; however, the performance of existing texture synthesis algorithms varies on different types of textures. Moreover, the performance of a synthesis algorithm is usually judged by visually examining the results, which could be subjective and inconsistent among different people. To better evaluate the synthesis results, an objective and consistent criterion in addition to the visual inspection would be very useful.

The goal of this study is to compare the performance of texture synthesis algorithms in a more consistent way. To satisfy this goal, testing samples should cover an appropriate scope of textures on the spectrum, and more importantly, the testing samples should have a consistent property so that it is easy to verify if the property is faithfully preserved in the synthesis process. For these reasons, we consider a particular type of texture in this paper: *near-regular textures*[Liu et al. 2004b]. *Regular textures* are simply periodic patterns where the color/intensity and shape of all texture elements are repeating in equal intervals. In the real-world, however, few textures are exactly regular. Most textures we see are near-regular, e.g., cloth, basket, windows, brick walls, carpet, and blanket. A near-regular texture can be regarded as a statistical distortion of a regular, wallpaper-like congruent tiling. This statistical distortion can be synthesized and manipulated in a texture synthesis process[Liu et al. 2004a].

2 Selected algorithms

The selected algorithms[Kwatra et al. 2003; Liang et al. 2001; Liu et al. 2004b; Lin et al. 2004] and the major differences between these algorithms are summarized in Figure 2. These algorithms share a common feature in that they synthesize textures by copying image patches from a texture sample and stitching these patches into the synthesized image. The major difference of these algorithms is their methods in patch extraction and pasting, which affect their capability to preserve the structural regularity of textures in the synthesis process.

*e-mail: {wclin,jhhays,cywu,yanxi}@cs.cmu.edu
†e-mail: kwatra@cc.gatech.edu

Algorithms	Graph cut	Near-regular synthesis	Regularized patch-based	Patch-based
Patch shape/size	Partial or whole input image	Tile shape/size from translational symmetry analysis	User identified lattice, quadrilateral patch	User-defined, rectangular patch
Patch extraction	Random locations	Lattice points	Lattice points	Random locations
Patch placement	Random or maximal correlation locations	Lattice points	Lattice points	Patch grids
Patch stitching	Graph cut & blending	Dynamic programming & blending	Image-feathering	Image-feathering
Regularity preservation	53%	95%	86%	23%
Average of user scores	2.9 (0.058)	3.4 (0.039)	3.0 (0.059)	1.9 (0.075)

Figure 2: Comparison of the selected synthesis algorithms. Numbers in parentheses are the standard error of the mean.

3 Comparison results

Regularity preservation test and **user evaluation test** were used to compare the synthesis results of four algorithms. These two tests are complementary to each other. The former is more objective since it only checks if the global regularity is preserved; the latter is subjective, but it examines the overall quality of the synthesized textures, including color/intensity, statistical variations, and structures, in terms of human perception. 38 texture samples were tested (http://www.cs.cmu.edu/~wclin/nrt_comp/web.html). 10 subjects participated the user evaluation test in which the synthesized textures of each algorithm were rated using a 4-point scale (4 to 1, best to worst). We ran ANOVA to analyze the user evaluation scores. There are two major findings: (1)near-regular texture synthesis algorithm performs statistically significantly better than the other three texture synthesis algorithms (all three pair-wise comparisons have $p < 0.001$); (2)the user scores are highly correlated to the degrees of regularity, i.e. the scores for preserved regularity and violated regularity differ statistically significantly ($p < 0.001$), with averages 3.3 and 2.1 respectively.

4 Conclusion

Our comparison study shows that near-regular texture synthesis algorithm[Liu et al. 2004b] performs significantly better than the other three texture synthesis algorithms on the 38 near-regular textures tested. Besides, the user evaluation result indicates that global regularity is actually an important factor when humans evaluate the faithfulness of synthesized near-regular textures.

References

KWATRA, V., SCHÖDL, A., ESSA, I., TURK, G., AND BOBICK, A. 2003. Graphcut textures: Image and video synthesis using graph cuts. In *ACM SIGGRAPH*, 277–286.

LIANG, L., LIU, C., XU, Y., GUO, B., AND SHUM, H. 2001. Real-time texture synthesis by patch-based sampling. *ACM Transactions on Graphics 20*, 3, 127–150.

LIN, W.-C., HAYS, J., WU, C., KWATRA, V., AND LIU, Y. 2004. A comparison study of four texture synthesis algorithms on regular and near-regular textures. Tech. Rep. CMU-RI-TR-04-01, Robotics Institute, Carnegie Mellon University.

LIU, Y., LIN, W.-C., AND HAYS, J. 2004. Near-regular texture analysis and manipulation. In *ACM SIGGRAPH*. To appear.

LIU, Y., TSIN, Y., AND LIN, W.-C. 2004. The promise and perils of near-regular texture. *International Journal of Computer Vision*. To appear.