One-Click Lattice Extraction from Near-Regular Texture

Kevin Sookocheff David Mould

University of Saskatchewan

Abstract

We present a method for extracting a lattice from near-regular texture. Our method demands minimal user intervention, needing a single mouse click to select a typical texton. The algorithm follows a four-step approach. First, an estimate of texton size is obtained by considering the spacing of peaks in the auto-correlation of the texture. Second, a sample of the image around the user-selected texton is correlated with the image. Third, the resulting correlation surface is converted to a map of potential texton centres using non-maximal suppression. Finally, the maxima are formed into a graph by connecting potential texton centres. We have found the method robust in the face of significant changes in pixel intensity and geometric structure between textons.

CR Categories: I.3.8 [Computer Graphics]: Applications I.4.8 [Image Processing and Computer Vision]: Scene Analysis—Object Recognition

1 Introduction

A *regular texture* is formed from a regular congruent tiling of textons. If the tiling statistically deviates from regularity, either by texton structure, colour, or size, we call the texture *near-regular* [Liu et al. 2004b]. If we continue to perturb the tiling, the texture becomes *stochastic*. The set of possible textures that lie between regular and stochastic make up the *texture spectrum*: regular, near-regular, irregular, near-stochastic, and stochastic (see Figure 1).

Given a near-regular texture, we propose a method of extracting a lattice defining the placement and structure of textons. The texton centres will be connected by edges in the lattice following the logical structure of texton placement. More precisely, for a texture T, we wish to create a graph $G = (V, E)$ dependent on T, where V is a set of texton centres, and $E = (v_i, v_j)$ is a set of edges, where $v_i, v_j \in V$. Each edge $e \in E$ connects texton centre $v \in V$ to its k-nearest perceptually logical neighbours, where k varies depending on the local texture structure surrounding v.

2 Related Work

Texture regularity is difficult to quantify. Statistical methods using co-occurrence matrices [Starovoitov et al. 1998; Haralick 1979; Connors and Harlow 1980] have proven difficult to extend over the texture spectrum. Auto-correlation based techniques [Chetverikov 2000; Leu 2001; Lin et al. 1997] have led to the similar notions of *auto-correlation profile* [Leu 2001] and *contrast function* [Chetverikov 2000]. The insight behind the two ideas is to look at a profile of a texture's auto-correlation surface along one or more axes. The degree of regularity in the spacing and height of maxima along a profile is used to quantify texture regularity. Clustering of auto-correlation surfaces performed over a set of Gabor filtered images has been used to classify texture according to regularity for the current iteration of the MPEG-7 image standard [Manjunath et al. 2000].

Previous and concurrent work on identifying textons follows three general approaches: *colour thresholding*, *filtering*, and *probabilistic learning*. Filtering has shown promise in identifying repeating textons [Malik et al. 1999; Blostein and Ahuja 1989; Benke 2000]. Malik et al. [1999] use a battery of 36 filters of different size and orientation to create a vector of filter responses for each location in the texture. Clusters in this high-dimensional feature space represent frequently occurring elements in the texture. This work shows promise in identifying textons, though it is a computationally demanding process. Machine learning may be used to discover texture model parameters when viewing texture as an observation of some underlying random process, such as a Markov or Gibbs random field [Gimel'farb 1999; Guo et al. 2003; Zhu et al. 2005].

Hamey [Hamey 1989] proved that, for regularly repeating patterns, the shortest pair of vectors defining the frequency of texton occurrence can translate a texton over the plane, creating a tiling or regular texture. In the sequel, we call this pair *translation vectors*. Lattice extraction for regularly repeating textures is close to becoming a solved problem using this proof and a generalised Hough transform over translation vector intersection points [Liu et al. 2004a; Tuytelaars et al. 2003]. Unfortunately, as a texture deviates from regularity, the translation vector proof fails to hold and the generalised Hough transform fails to connect translation vector intersections. The work presented here overcomes these hurdles to extend lattice extraction from regular to near-regular texture.

3 Lattice Extraction

We have established an algorithm that successfully extracts a lattice from regular and near-regular texture. The algorithm can be divided in four parts: texton size estimation, sample correlation, peak extraction, and lattice construction.

3.1 Texton Size Estimation

Lattice extraction, under our algorithm, requires a single measure of texton size. This measure will necessarily be an estimate; the definition of near-regular texture implies

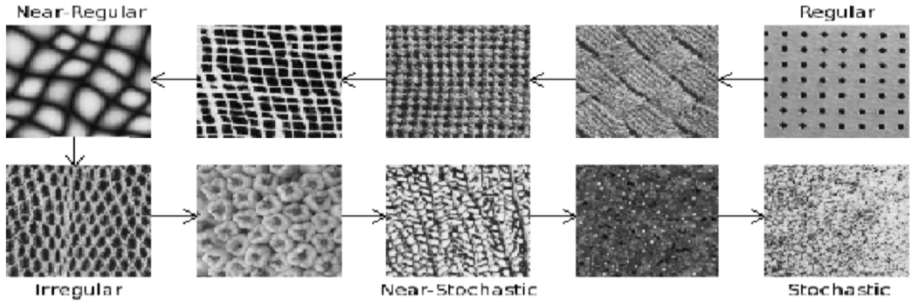
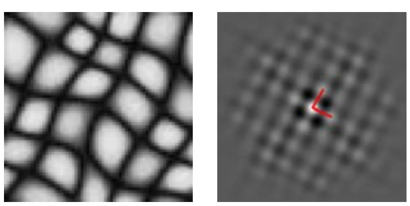


Figure 1: The texture spectrum.

texton size variation. We obtain our size estimate using auto-correlation. The texture's auto-correlation surface is computed by multiplying the Fourier transform of the image by its matrix transposition, and inverse Fourier transforming the result.

The auto-correlation surface is unfortunately noisy, and peaks in the surface are not easily extracted. Fortunately, the work of Lin provides a convenient way to smooth an auto-correlation surface using a Gaussian filter, where the spread parameter, σ, is iteratively increased until the number of peaks in the surface does not decrease between iterations [Lin et al. 1997]. We leave the size of the filter as a parameter of the algorithm, though we have achieved good results with a constant 5×5 filter. Figure 2(a) depicts a near-regular texture, and Figure 2(b) depicts a smoothed correlation surface.



(a) (b)

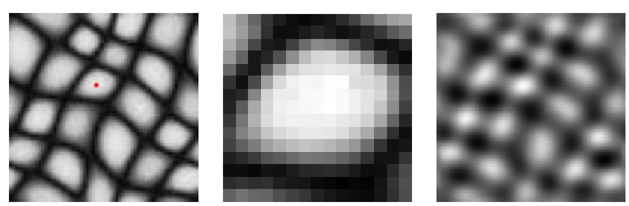
Figure 2: A near-regular texture with texton size estimate gained from correlation. 2(a): the original image, 2(b): smoothed auto-correlation surface and texton size estimate.

Having computed a smooth auto-correlation surface, the direction of the translation vectors can be computed using the method of Leu [Leu 2001]. The largest peak in the auto-correlation surface is the location where the image is exactly correlated with itself, which serves as the origin of the translation vectors. The next largest peak in the surface is the maximum correlation of the image with a translated version of itself. The vector from the origin to this point is the the *primary* translation vector. Similarly, the vector from the origin to the third largest peak in the auto-correlation surface is the *secondary* translation vector. Figure 2(b) depicts a texton size estimate determined by the primary and secondary translation vectors.

3.2 Sample Correlation

The next stage of lattice extraction produces a surface with maxima corresponding to texton centres. Such a surface may be computed using frequency space analysis, correlation, co-occurrence matrices, or object recognition; we choose correlation. We have found, and recent literature confirms [Liu et al. 2004a; Tuytelaars et al. 2003], that correlation is a robust and effective method for detecting repetitions in regular and near-regular texture.

To produce a correlation surface with peaks corresponding to texton centres, a sample texton is correlated with the input texture. Texture locations displaying characteristics similar to the sample texton produce peaks in the correlation surface. Techniques have been developed that isolate texton samples in certain contexts [Malik et al. 1999; Leung and Malik 1996; Schaffalitzky and Zisserman 1999; Malik et al. 2001], but no general purpose method has been found. We therefore require user intervention to select a texton sample; the user clicks on the centre of a "typical" texton. The texton sample is created by extracting a window from the texture with size specified by the approximate texton size. The window is rectangular, with axes parallel to the image axes, and centred at the user-selected point. This sample is correlated with the texture to produce a surface. The correlation surface is then smoothed using Lin's algorithm, which serves a dual purpose: eliminating unnecessary peaks, and improving peak location estimates. Figure 3 depicts the sample correlation process.



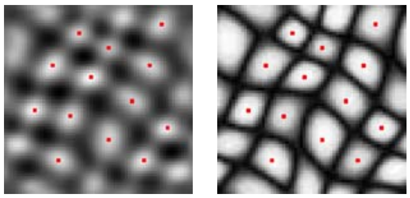
(a) (b) (c)

Figure 3: Correlation of a texture with a texton sample. 3(a): the user selected texton centre, 3(b): sample window derived from texton centre and size estimate, 3(c): smoothed correlation surface.

3.3 Peak Extraction

Having found a surface with peaks closely corresponding to texton centres, we can extract peaks to obtain locations of *potential texton centres*. The peaks are termed potential texton centres because the extraction process will sometimes return a peak in the surface that is not located at a perceptual texton centre.

We use non-maximal suppression to extract peaks. The results of non-maximal suppression are sensitive to the choice of neighbourhood, and care must be taken to choose a neighbourhood appropriate to the task. In our algorithm, the neighbourhood is dependent on approximate texton size, so, in the ideal case, one potential texton centre will reside within each texton of the image. Figure 4 shows the non-maximal suppression result.

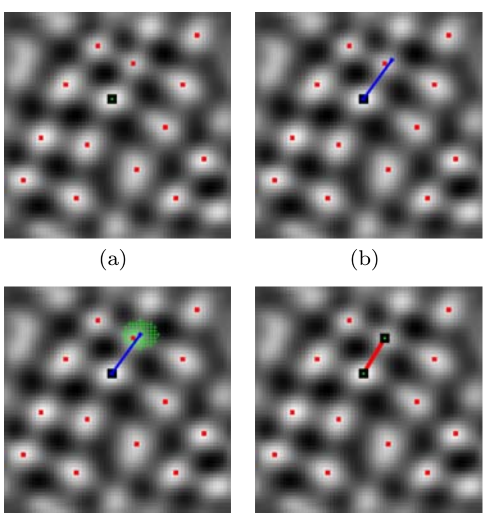


(a) (b)

Figure 4: Non-maximal suppression of the partial correlation surface extracts texton centres. 4(a): peaks isolated through non-maximal suppression, 4(b): partial correlation peaks as potential texton centres.

3.4 Lattice Construction

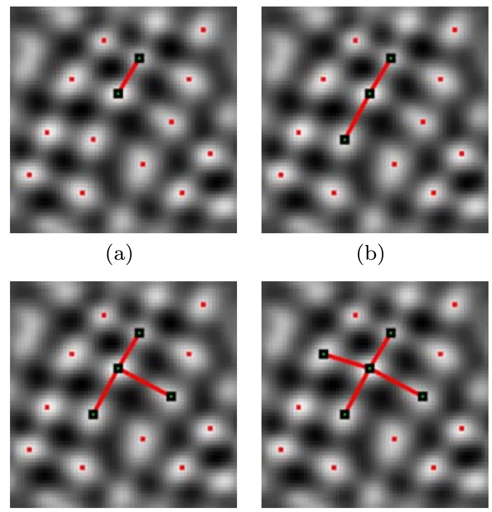
We have developed an iterative lattice construction algorithm. The algorithm makes use of a queue, storing *actual texton centres*, which are potential texton centres included in the final lattice. Each texton centre in the queue is processed by extending the lattice from this centre to a potential texton centre not in the lattice. Potential texton centres newly attached to the lattice are inserted into the queue to be processed at a later time. The processing stops when the queue becomes empty, signalling that no other potential texton centres can be added to the lattice.



(a) (b) (c) (d)

Figure 5: Lattice construction. 5(a): the user selected texton centre, 5(b): a translation vector extended from a known texton centre, 5(c): a search centred at the translation vector endpoint, 5(d): an edge added to the lattice.

In more detail, the algorithm first inserts the point initially selected by the user into the queue (Figure 5(a)). A translation vector is extended from this point (Figure 5(b)), and a search is performed in an elliptical region centred at the translation vector's endpoint (Figure 5(c)). If a potential texton centre lies within this ellipse, an edge is created and the potential centre added to the lattice and inserted in the queue (Figure 5(d)). A potential texton centre is added to the lattice for each translation vector and negated translation vector. Figure 6 depicts a complete iteration of the lattice extraction algorithm. Overlaying the lattice on the original texture solves the problem of lattice extraction from near-regular texture.



(a) (b) (c) (d)

Figure 6: One iteration of the lattice extraction algorithm.

A few notes about our algorithm. First, the search ellipse is defined using the ratio of the size of the translation vectors. We initially create a small search ellipse, and iteratively increase the major and minor axes, until a potential texton centre is found. If more than one centre lies in the search ellipse, the centre with largest correlation value is added to the lattice. Second, if any part of the search ellipse ever lies outside of the boundary of the image, we cancel the search. Third, if a known texton centre lies within the search ellipse, we automatically connect to this centre regardless of the contents of the remainder of the search ellipse.

4 Results and Conclusion

Figure 7 depicts some results from our lattice extraction algorithm. Figure 7(a) shows a perceptually logical lattice extracted from a texture with significant texton variation in geometry, scale, and pixel intensity. Figure 7(b) shows a real-world example of lattice extraction. Figure 7(c) shows how the dominant texton is connected in the lattice. Figure 7(d) depicts a successful lattice extraction under varying texton geometry. Finally, the honeycomb geometry of Figure 7(e) poses no problems for our algorithm.

Although regularity quantification and texton size estimation have previously been studied, the application of the two areas in the same work provides a novel algorithm for lattice extraction from near-regular texture. Furthermore, the

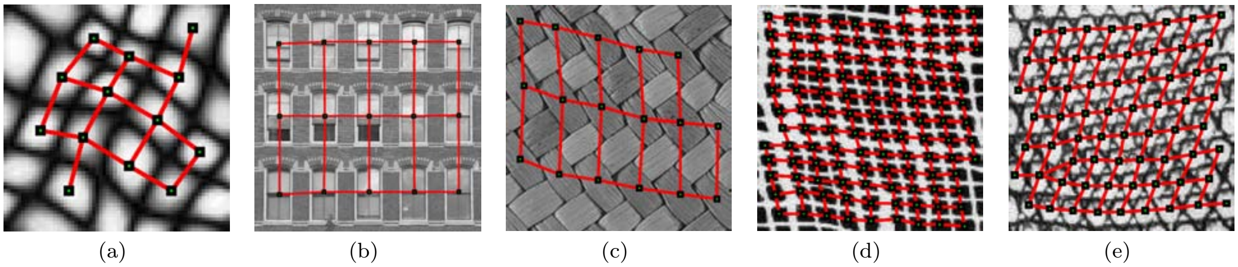


Figure 7: Lattice extraction results. See Section 4 for discussion. Figure 7(a) is from the database of DeBonet. Figure 7(b) is from the database of Simoncelli. The remaining figures are from the Brodatz texture album [Brodatz 1966].

graphing algorithm we have developed is a new contribution to computer graphics, providing efficient construction of a perceptually logical lattice. The result may be used to further automate texture synthesis, or be applied in novel ways to problems such as texture blending.

Though observed failure cases are few, we wish to eliminate inaccuracies attributed to improper texton size estimation. In the future, we hope to extend our algorithm to accommodate texture from a wider range of the texture spectrum. Specifically, we wish to extract a lattice from irregular texture.

References

BENKE, K. 2000. Adaptation of spatial filtering theory to the estimation of texture element size and regularity. *Computers and Electronics in Agriculture 28*, 3, 187 – 194.

BLOSTEIN, D., AND AHUJA, N. 1989. A multi-scale region detector. *Computer Graphics Vision and Image Processing 45*, 1 (January), 22 – 41.

BRODATZ, P. 1966. *Textures: A Photographic Album for Artists and Designers*. Dover, New York.

CHETVERIKOV, D. 2000. Pattern regularity as a visual key. *Image and Vision Computing 18*, 975 – 985.

CONNORS, R., AND HARLOW, C. 1980. Toward a structural analyzer based on statistical methods. *Computer Graphics and Image Processing 12*, 224–256.

GIMEL'FARB, G. L. 1999. Modeling image textures by gibbs random fields. *Pattern Recognition Letters 20*, 11 - 13, 1123 – 1132.

GUO, C.-E., ZHU, S.-C., AND WU, Y. N. 2003. Modeling visual patterns by integrating descriptive and generative methods. *International Journal of Computer Vision 53*, 1, 5 – 29.

HAMEY, L. G. C. 1989. Computer analysis of regular repetitive textures. In *Proceedings of DARPA Image Understanding Workshop*, 1076 – 1088.

HARALICK, R. 1979. Statistical and structural approaches to texture. *Proceedings of the IEEE 67*, 786 – 804.

LEU, J. 2001. On indexing the periodicity of image textures. *IVC 19*, 13 (November), 987–1000.

LEUNG, T. K., AND MALIK, J. 1996. Detecting, localizing and grouping repeated scene elements from an image. In *ECCV '96: Proceedings of the 4th European Conference on Computer Vision-Volume I*, Springer-Verlag, 546–555.

LIN, H.-C., WANG, L.-L., AND YANG, S.-N. 1997. Extracting periodicity of a regular texture based on autocorrelation functions. *Pattern Recognition Letters 18*, 5, 433 – 443.

LIU, Y., COLLINS, R., AND TSIN, Y. 2004. A computational model for periodic pattern perception based on frieze and wallpaper groups. *IEEE Transactions on Pattern Analysis and Machine Intelligence 26*, 3 (March), 354 – 371.

LIU, Y., LIN, W.-C., AND HAYS, J. H. 2004. Near regular texture analysis and manipulation. *ACM Transactions on Graphics (SIGGRAPH 2004) 23*, 3 (August), 368 – 376.

MALIK, J., BELONGIE, S., SHI, J., AND LEUNG, T. 1999. Textons, contours and regions: Cue integration in image segmentation. In *ICCV '99: Proceedings of the International Conference on Computer Vision-Volume 2*, IEEE Computer Society, 918.

MALIK, J., BELONGIE, S., LEUNG, T., AND SHI, J. 2001. Contour and texture analysis for image segmentation. *Internation Journal of Compututer Vision 43*, 1, 7–27.

MANJUNATH, B., WU, P., NEWSAM, S., AND SHIN, H. 2000. A texture descriptor for browsing and image retrieval. *International Communications Journal 16* (September), 33 – 43.

SCHAFFALITZKY, F., AND ZISSERMAN, A. 1999. Geometric grouping of repeated elements within images. In *Shape, Contour and Grouping in Computer Vision*, Springer-Verlag, 165–181.

STAROVOITOV, V., JEONG, S.-Y., AND PARK, R.-H. 1998. Texture periodicity detection: features, properties, and comparisons. *IEEE Transactions on Systems, Man, and Cybernetics 28*, 6 (November), 839 – 849.

TUYTELAARS, T., TURINA, A., AND GOOL, L. V. 2003. Non-combinatorial detection of regular repetitions under perspective skew. *IEEE Transactions on PAMI 25*, 4 (April), 418 – 432.

ZHU, S.-C., EN GUO, C., WANG, Y., AND XU, Z. 2005. What are textons? *International Journal of Computer Vision 62*.