

Simulating Geological Structures Based on Training Images and Pattern Classifications

P. Switzer, T. Zhang, A. Journel

Department of Geological and Environmental Sciences
Stanford University
CA, 94305, U.S.A

Ph. +01 650 7232879; Fax +01 650 7250979

E-mail: switzer@stanford.edu

Abstract

Local two-dimensional spatial structure, as represented by a training image, can be summarized by a system of filter scores. Local patterns are then classified according to these scores. Sequential point-support simulation proceeds by selecting the score class most resembling the local data and then patching a pattern from this class at the simulation location. This procedure can handle both categorical and continuous variable training images. In addition, because the score space has low dimension, computation is efficient. Examples show spatial simulations derived from training images of sand channels and lithofacies.

Keywords: training images, filters, filter scores, simulations, spatial uncertainty

1 Training Images

Multiple point (mp) simulation aims at capturing patterns from a training image and anchoring them to actual data. Training image patterns are described as repetitive or representative features. Training images are purely conceptual models of patterns. They are typically disassociated from the spatial sample conditioning data or other location-specific detailed information. A training image reflects only general aspects of spatial structure or texture. It should display stationary patterns, which can be transported to actual simulation space by reproducing higher order statistics from the training image. Interpreted photographs or remote sensing data or hand-drawn sketches of patterns can be used to build training images. Figures 1a-c show training image examples of categorical and continuous variables.

A template of fixed size is passed over the training image. At each position in the training image the template records the local pattern as an array of values. The array is reduced to a

low-dimensional set of scores. Thus, the local structure at each position in the training image is reduced to a set of scores, see Figure 4. Patterns with similar scores are then grouped together into pattern classes.

To capture features at more than one scale, the template can be varied. For example, we have used templates at three different scales, with a corresponding set of local scores for each scale, all derived from the training image, see Figure 2.

2 Pattern scoring

Let $X(i, j)$ denote the value at location (i, j) in the training image. A score $S_f(i, j)$ for the pattern in the neighborhood of (i, j) is defined by a filter $f(u, v)$ as follows:

$$S_f(i, j) = \sum_{v=-n}^n \sum_{u=-n}^n X(i+u, j+v) f(u, v)$$

where the dimension of the local neighborhood or template is $(2n+1) \times (2n+1)$. We define six different filters f_1, \dots, f_6 as follows.

(1) f_1 : N-S average

$$f_1(u, v) = 1 - \frac{|v|}{n}, v = -n, \dots, n$$

see figure 3a.

(2) f_2 : E-W average: It is obtained by rotating f_1 by 90 degrees.

$$f_2(u, v) = 1 - \frac{|u|}{n}$$

See figure 3b.

(3) f_3 : N-S gradient:

$$f_3(u, v) = v/n$$

see figure 3c.

(4) f_4 : E-W gradient: It is obtained by rotating f_3 by 90 degrees.

$$f_4(u, v) = u/n$$

see figure 3d.

(5) f_5 : N-S curvature:

$$f_5(u, v) = \frac{2|v|}{n} - 1$$

see figure 3e.

(6) f_6 : E-W curvature: It is obtained by rotating f_5 by 90 degrees.

$$f_6(u, v) = \frac{2|u|}{n} - 1$$

See figure 3f.

Each of these six filters is used to scan the TI. At each pixel location, the template of neighborhood data is weighted by the filters to produce a series of 6 scores. If the six scores are assigned to the pixel at the center of the template, we thus obtain score maps of the training image itself. In Figure 5, we see the score maps for the training image in Figure 2. The size of the template used is 15×15 pixels ($n=7$), while the training image is 250×250 .

The first two score maps S_1 and S_2 are weighted moving averages of the $15 \times 15 = 225$ template values. They highlight the channel center locations. The S_3 score provides N to S edge detection, hence that score highlights the channel boundaries. The S_4 gives E to W edge detection, reflecting any deviation of the channel from the dominant EW channel direction.

3 Pattern classification using scores

Scanning the TI with each of the six filters produces a frequency distribution for each of the scores S_1, \dots, S_6 . Each of these marginal frequency distributions is discretized into 5 equal frequency bins according to their respective quintiles. This results in a partition of the 6-dimensional score space into $5^6 = 15625$ cells. [For binary data on a fine grid it may happen that many template neighborhoods are all zeros or all ones. Therefore it is possible for some quintiles to be the same, resulting in fewer effective bins].

Even though each of the six scores has been divided into equal frequency bins, the 6-component joint cell frequencies are not equal. Many cells are empty because there are no local training patterns having such filter score combinations. Training patterns whose filter scores fall into the same cell are thus grouped into pattern classes. Figures 6a-6c show three groups of channel patterns derived from the training image in Figure 2 at different scales.

For each non-empty score cell, a prototype is obtained by averaging all patterns falling into that class, which can be seen as the aggregate of similar local patterns. For example the prototypes corresponding to the pattern classes in Figures 6a-6c are shown at the bottom of each figure, respectively. Figure 7 shows the first 20 prototypes with the most replicates of patterns for 20 pattern classes derived from the channel training image of Figure 2 at the finest scale.

4 Stochastic pattern simulation

Based on the classification of local patterns from the training image, sequential simulation with multiple-grids can be utilized to generate pattern simulations that incorporate structural features of the training image. For simplicity, we first discuss pattern simulations at single grid scale, then move on to simulation using multi-grids.

4.1 Simulation at a single grid scale

We use a sequential simulation approach visiting each node of the simulation grid along a random path. At each node to be simulated, we extract the configuration of conditioning data or previously simulated values from a local template centered at this node. This local information template has the same dimensions as the local pattern prototypes obtained from the training image.

We calculate the distance between this template information TEM and each training prototype PROT. [If there are no conditioning data within the data template, choose one template prototype closest to the target mean and pick a training pattern from this prototype whose mean is close to the target mean.] Distance between the current template information and each prototype is calculated by a weighting average of absolute pixel wise differences. There are three types of conditioning information:

- (1) hard original data
- (2) previously simulated nodes
- (3) pattern patches from previously simulated nodes

The distance expression is written as

$$d(TEM, PROT) = \sum_{k=1}^3 \omega(k) \frac{\sum_{i_k=1}^{n_k} |x^{(k)}(i_k) - y^{(k)}(i_k)|}{n_k}$$

where i_k are the pixel locations of information of type k and $\omega(k)$ are weights for the three respective information types with $\omega(1) \geq \omega(2) \geq \omega(3)$.

Once we identify the local template prototype closest to the conditioning template information, we then sample a specific pattern from the prototype pattern class. We patch the sampled pattern at the current simulation node. The "inner" part of the patch is frozen and is not revisited in the sequential simulation. A larger inner patch area makes simulation faster, but may cause discontinuity.

4.2 Simulation using multi-grids

We use multi-grids to capture pattern structure and texture at different scales. The training image is scanned using local templates at several grid scales. Separately, at each grid scale, local patterns are converted to a 6-dimensional scores using the filters described earlier. Thus,

at each grid scale we get a classification of local patterns at that scale. In our examples, we used coarser scales that are 2 times and 4 times the dimension of the finest grid scale, see Figure 2.

Simulation proceeds, as described in section 4.1, from the coarsest grid to the finest grid. Simulated values from the preceding coarser grid are used as conditioning information at the finer grid simulation. However, coarser grid simulated values are themselves revisited and resimulated at the finer grid.

The two flowcharts for pattern classification and pattern simulation with multi-grids are displayed in Figures 11 and 12.

5 Illustrations

The illustrations shown here exhibit the ability of local filter scores to capture spatial pattern when used to simulate categorical and continuous patterns from training images. First, we investigate structural simulation using categorical training images, such as the 250×250 pixel binary training image (sand/shale) with sand proportion $p = 0.28$; see Figure 2.

Three grid scales are used. The local template size is 15×15 pixels and the patch size is 11×11 . The medium grid scale uses a 2 times spacing of pixels while the coarse grid uses 4 times spacing. 50 locations represent the hard data as shown in Figure 8b. At each simulation grid scale, local patterns from the training image are classified in score space using filters. Figures 6a-6c show examples of pattern classes at different scales. After training image classification, sequential simulation proceeds by randomly visiting nodes (pixels) in turn and identifying local pattern prototypes that match the currently available information in the neighborhood of the simulation node. Figures 8c-8e display conditional simulation results at successively finer grid scales. It can be seen that channel structures are well reproduced from the coarse grid to the fine grid. Unfreezing values simulated at the previous grid scale can lead to better pattern reproduction because mismatched patterns may be corrected at the finer scale simulation.

A 3 facies training image is shown in Figure 9a. The facies are sand channel, levee and shale background. Figures 9b-d are 3 non-conditional simulations, i.e. based only on the statistics of the training image, with no hard data. Once again, we use a 15×15 data template and 11×11 patch. Our test runs indicate that recoding facies did change training pattern classifications, but the simulations are qualitatively similar. We coded the 3 facies as follows: channel=0, levee=2, background=3.

A continuous variable training image is displayed in Figure 10a. It is a mosaic cross section of packed stones. It contains visible gray scale textures with sharp boundaries. We used 21×21 pixel templates to classify local training patterns over 3 grid scales. Figures 10b-d show 3 non-conditional simulations based on the training image of Figure 10a.

6 Conclusions

In this paper, we apply a set of filters to scan training images. Local patterns and textures in training images are classified by a set of filter scores. This leads to dimension reduction of training patterns. This training pattern classification allows us to simulate patterns derived from training images. The simulation proceeds by sequentially visiting each simulation node and identifying the closest training pattern class to the local template data centered at the simulation node. We sample a specific pattern from the identified pattern class, and patch the sampled pattern at the simulation node. Freezing the inner part of patched pattern not only makes the simulation faster but also ensure better pattern reproduction. Multi-grid simulation is implemented, allowing for pattern reproduction at different scales. Examples show that our approach works well for both categorical variable training images and continuous variable training images. Flowcharts for pattern classification and simulation are shown.

References

- Journel, A. G., 1993, Geostatistics: Roadblocks and challenges, in Soares, A.,ed,Geostatistics-Troia, Vol. 1: Kluwer Academic, Dordrecht, p.213-224.
- Journel, A.G., 2003, Multiple-point Geostatistics: A state of the Art, Report no. 16, Stanford Center for Reservoir Forecasting, Stanford University
- Strebelle, S., 2002, *Sequential simulation drawing structures from training images*: PhD thesis, Stanford University, Stanford, CA.
- Strebelle, S., 2002, Conditional simulation of complex geological structures using multiple-point statistics: *Math. Geol.*, v.34, no1, p.1-21.
- Tran, T. 1994, Improving variogram reproduction on dense simulation grids, *Computers & Geosciences*, vol.25, no.7, pp.1161-68

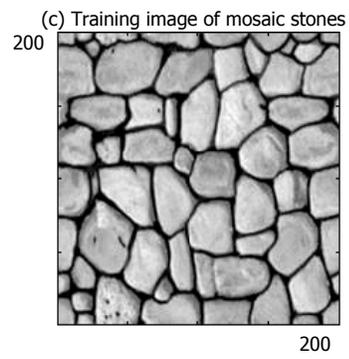
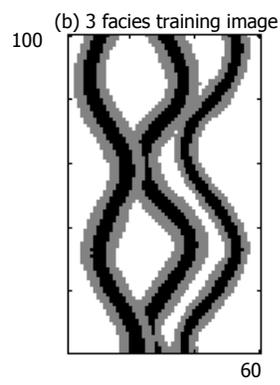
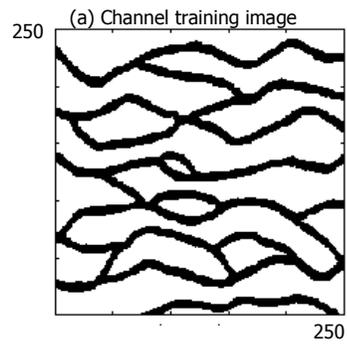


Figure 1: Training images

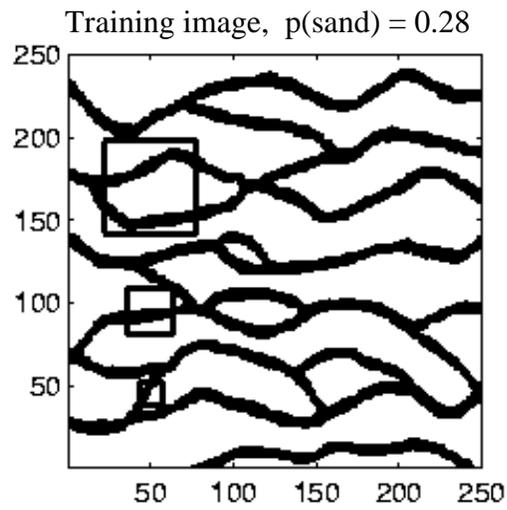
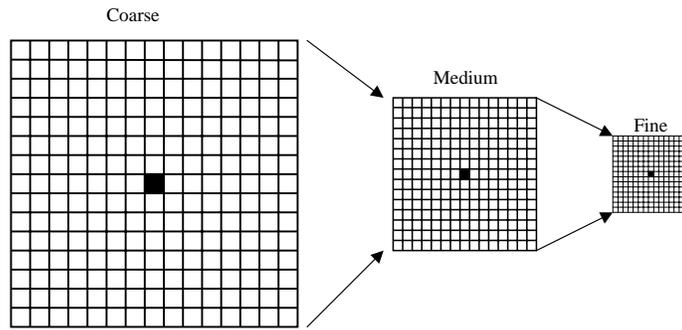


Figure 2: Channel training image and three templates (15x15) at different scales

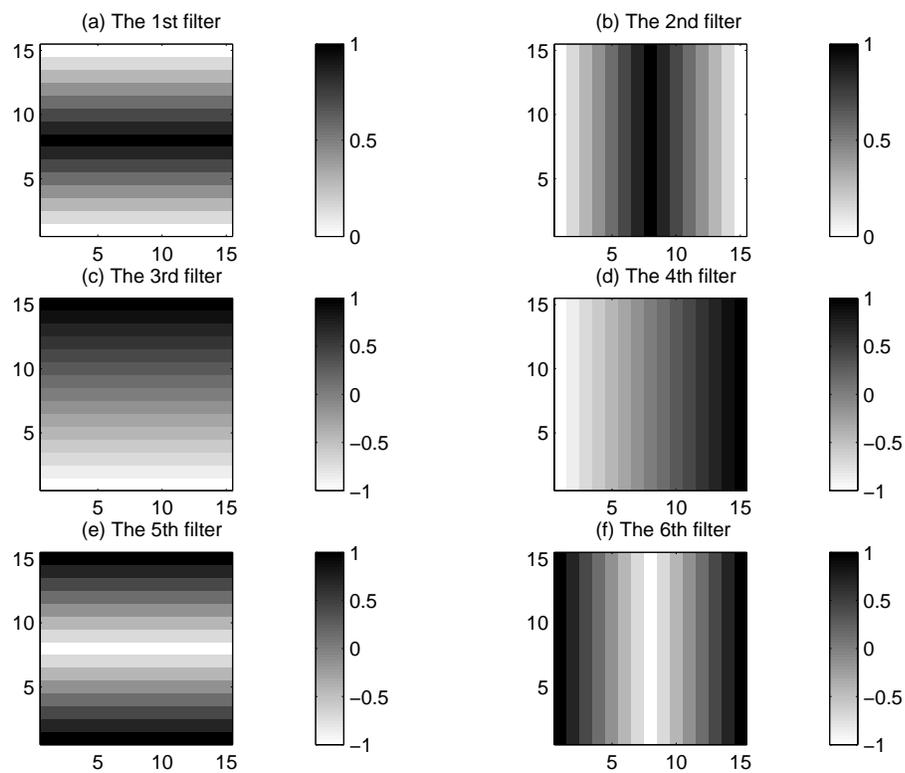


Figure 3: Six local filters: the weights are given in gray scale

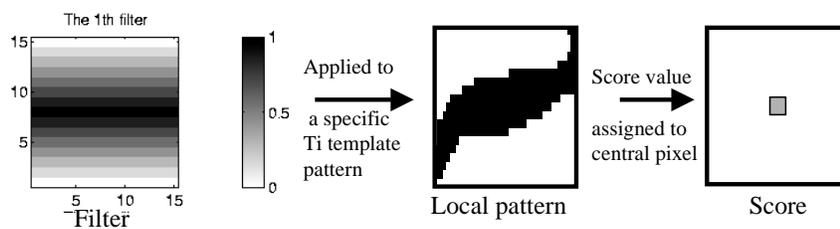


Figure 4: Pattern scoring

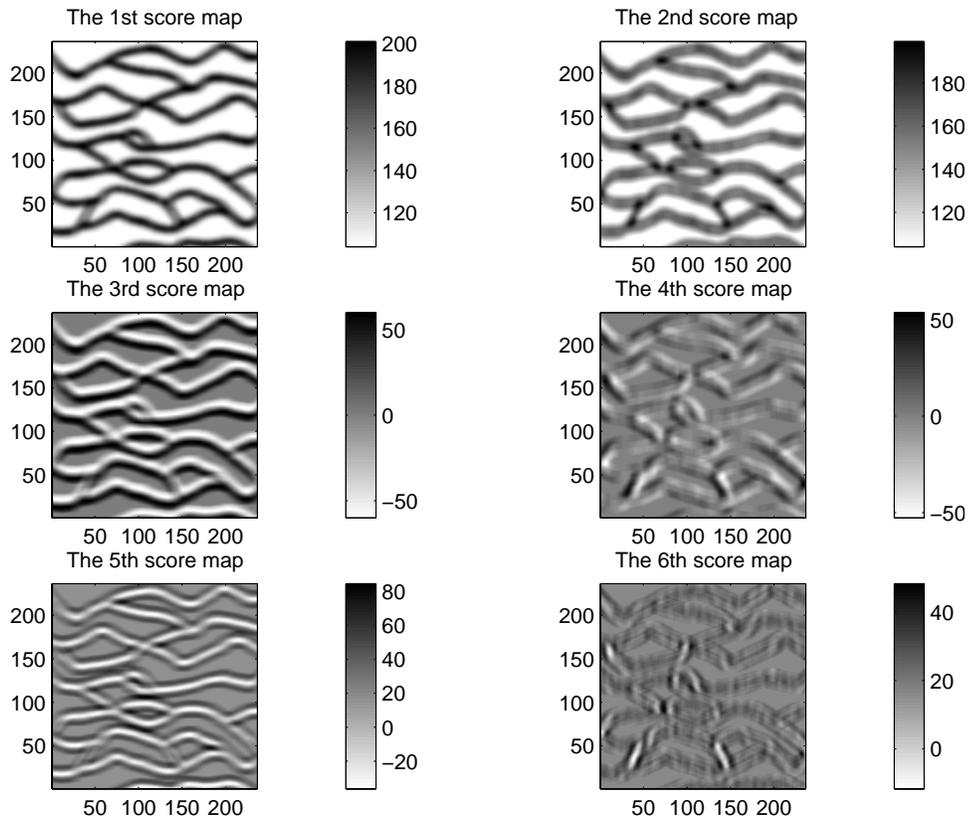
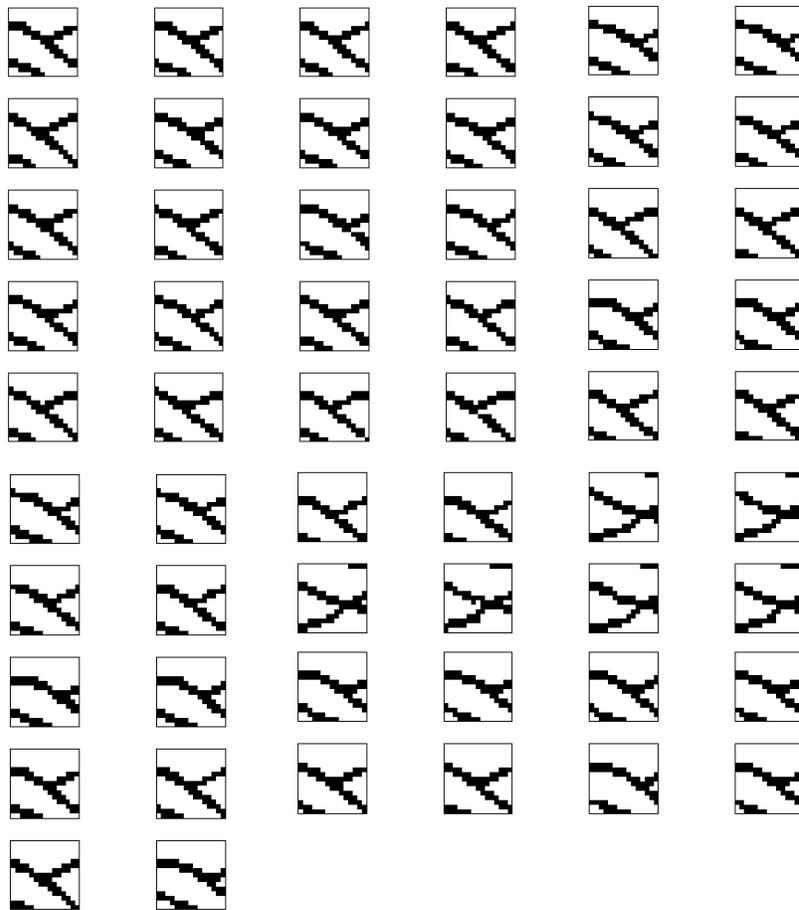
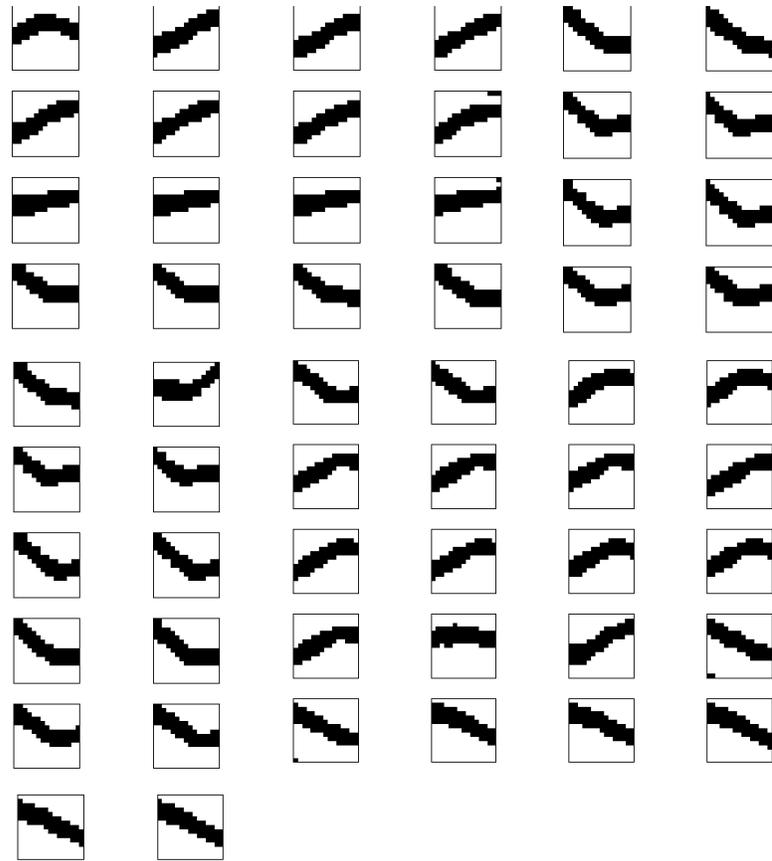


Figure 5: Six score maps of channel TI on the finest grid. (The template size is 15x15)



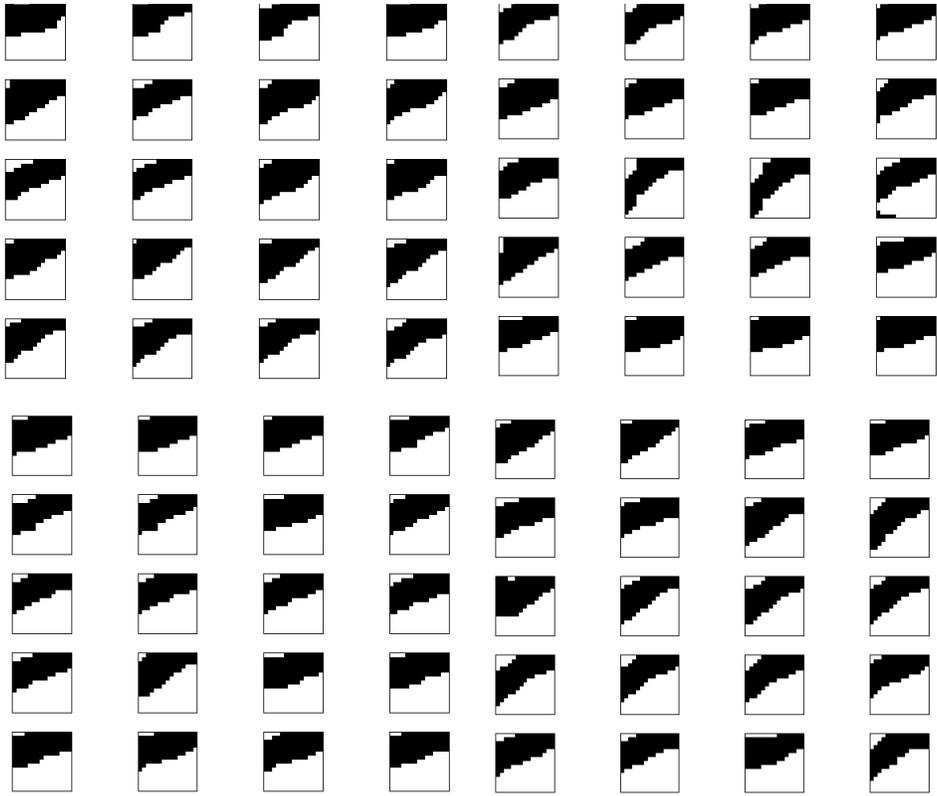
Prototype

Figure 6a: Patterns falling into a single cell in score space. Coarse grid 15x15 template. Use training image Figure 2



Prototype

Figure 6b: Patterns falling into a single cell in score space. Medium grid 15x15 template. Use training image Figure 2



Prototype

Figure 6c: Patterns falling into a single cell in score space. Fine grid 15x15 template. Use training image Figure 2

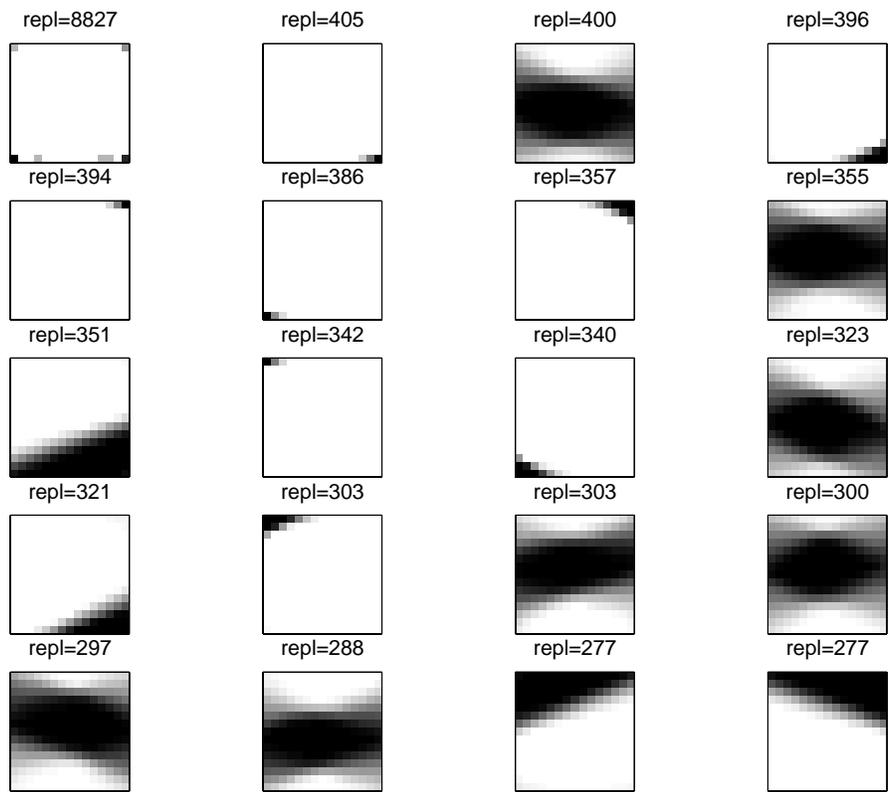


Figure 7: Class prototypes based on training image from Figure 2 and 15x15 template at the finest grid

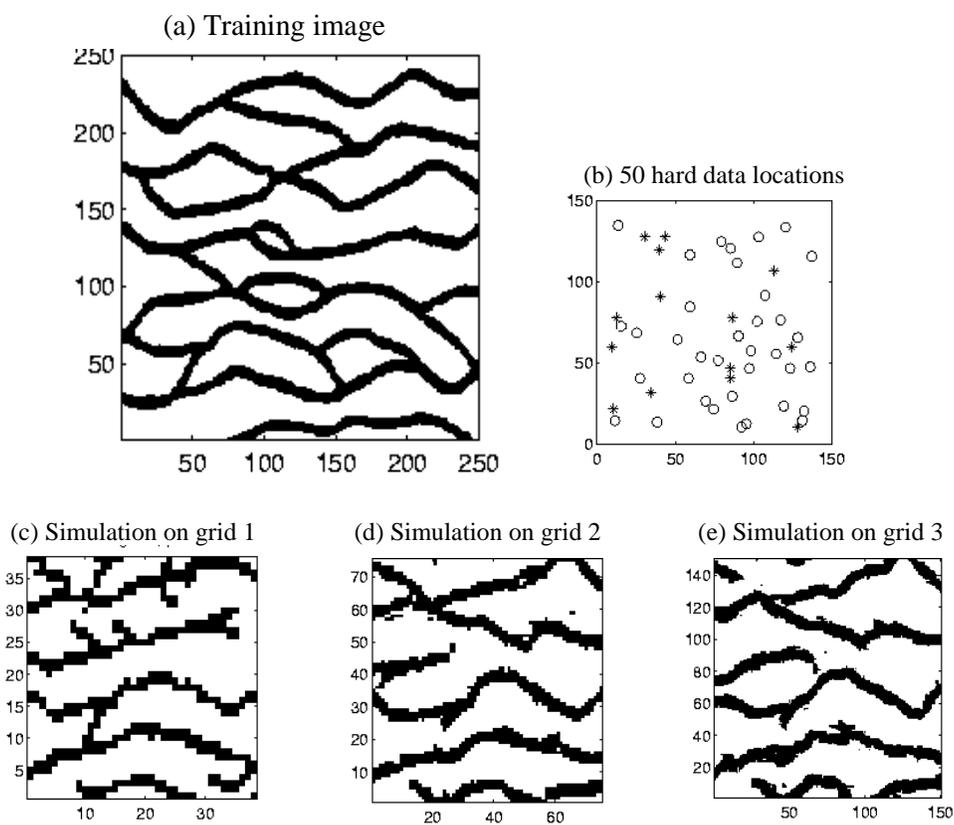


Figure 8: (a) Channel training image; (b) hard data [star indicates sand; circle indicates shale; (c)-(e) conditional simulations at different scales; [Template 15x15; patch 11x11]

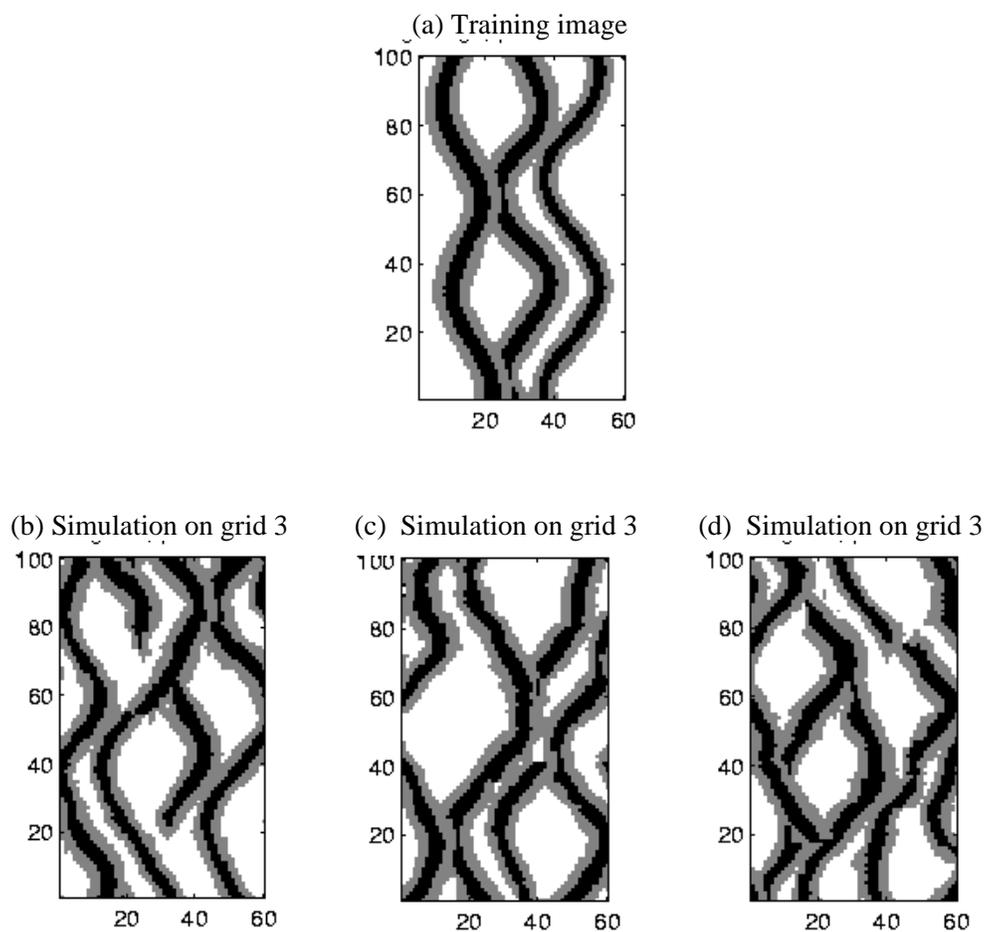


Figure 9: Three non-conditional simulations for a 3-facies channel training image [Template 15x15; patch 11x11]

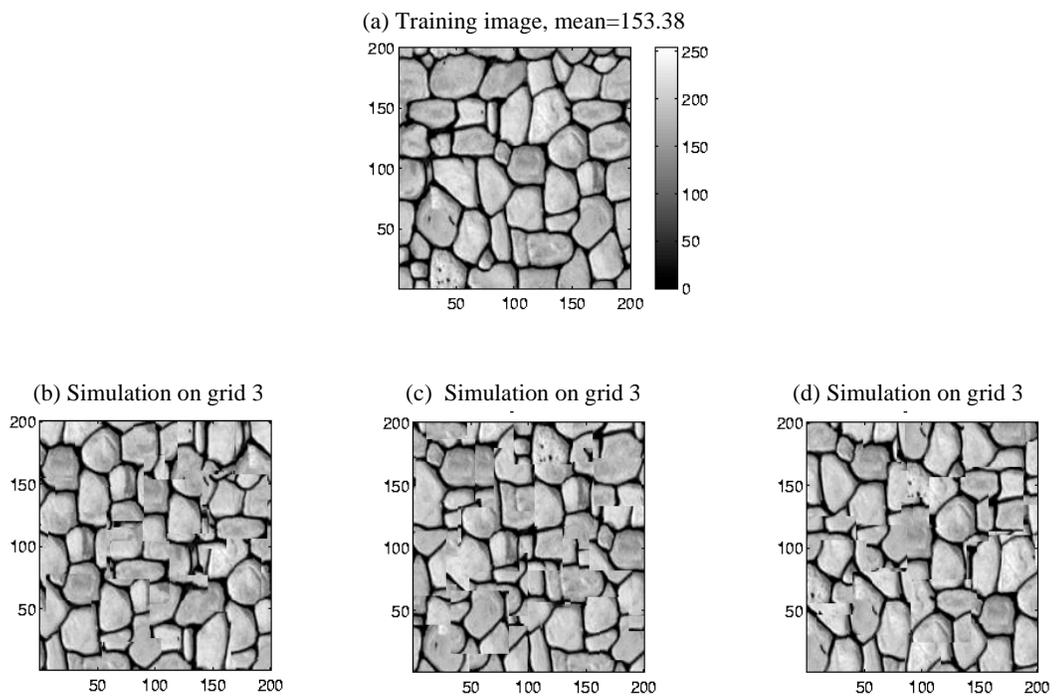


Figure 10: (a) Stone training image; (b)-(d) three non-conditional simulations at the finest grid [Template 21x21; patch 15x15]

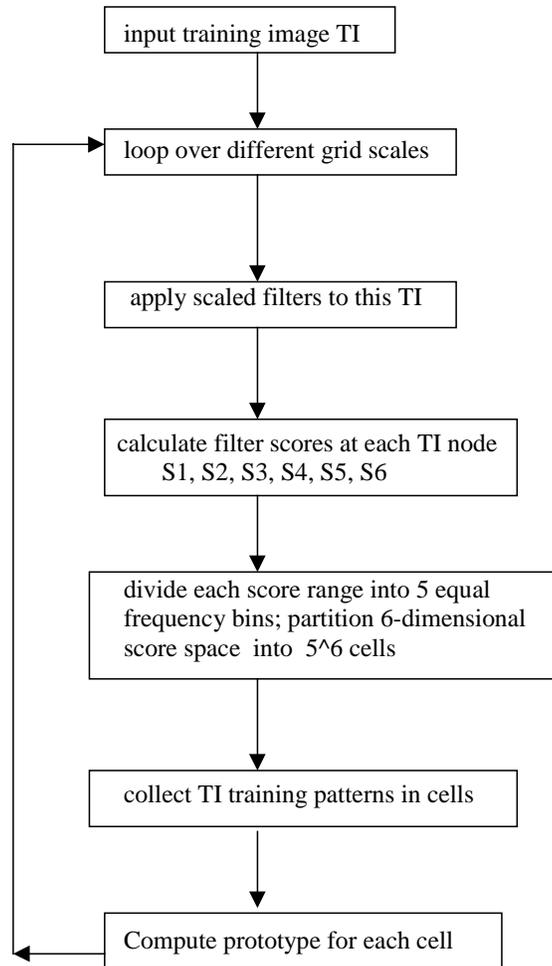


Figure 11: Classification flowchart

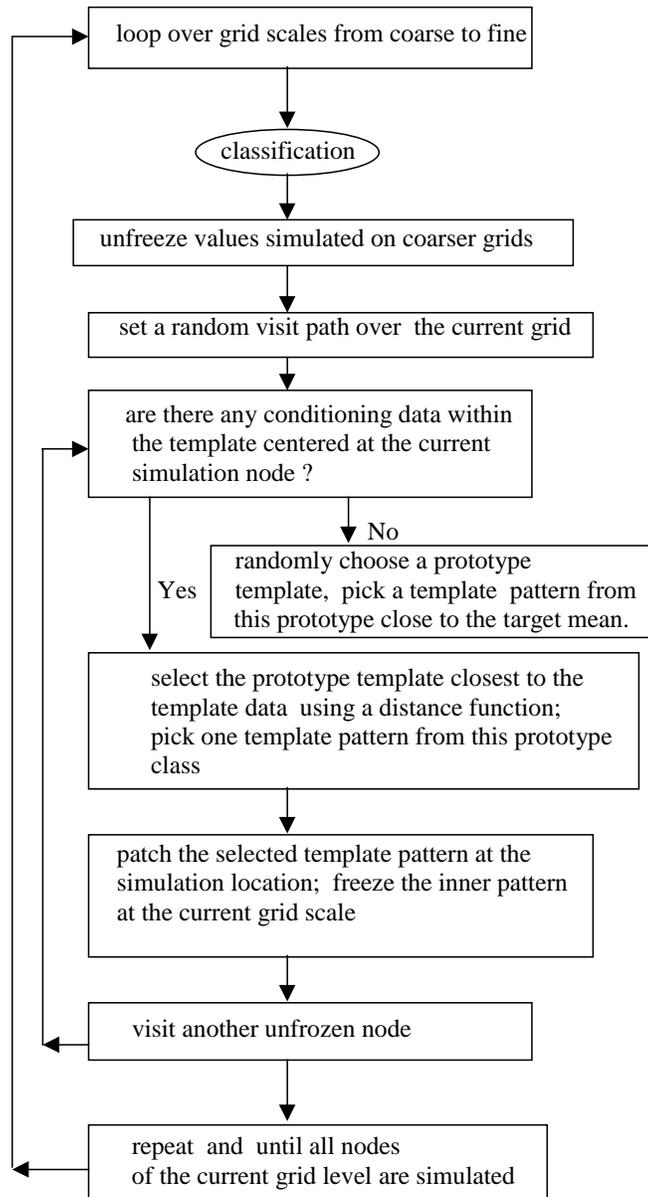


Figure 12: Simulation flowchart