Monte Carlo methods for optimizing the piecewise constant Mumford–Shah segmentation model

Hiroshi Watanabe\textsuperscript{1}, Satoshi Sashida\textsuperscript{1}, Yutaka Okabe\textsuperscript{1} and Hwee Kuan Lee\textsuperscript{2,3}

\textsuperscript{1}Department of Physics, Tokyo Metropolitan University, Hachioji, Tokyo 192-0397, Japan
\textsuperscript{2}Bioinformatics Institute, 30 Biopolis Street, No. 07-01, Matrix, Singapore 138671, Singapore
E-mail: leehk@bii.a-star.edu.sg

\textit{New Journal of Physics} 13 (2011) 023004 (14pp)
Received 24 August 2010
Published 1 February 2011
Online at http://www.njp.org/
doi:10.1088/1367-2630/13/2/023004

Abstract. Natural images are depicted in a computer as pixels on a square grid and neighboring pixels are generally highly correlated. This representation can be mapped naturally to a statistical physics framework on a square lattice. In this paper, we developed an effective use of statistical mechanics to solve the image segmentation problem, which is an outstanding problem in image processing. Our Monte Carlo method using several advanced techniques, including block-spin transformation, Eden clustering and simulated annealing, seeks the solution of the celebrated Mumford–Shah image segmentation model. In particular, the advantage of our method is prominent for the case of multiphase segmentation. Our results verify that statistical physics can be a very efficient approach for image processing.

Online supplementary data available from stacks.iop.org/NJP/13/023004/mmedia

\textsuperscript{3}Author to whom any correspondence should be addressed.
1. Introduction

Effective statistical physics theories such as renormalization group, scaling laws, theories of phase transitions and advanced Monte Carlo methods have been developed based on isotropic lattice models. There have been some efforts to extend these theories to complex, non-isotropic lattice models by casting image processing problems in a statistical physics framework \cite{1}–\cite{3}. Generally, an image consists of discrete pixels arranged on a square grid. This maps naturally to the familiar concept of a square lattice in statistical physics. Such notion of thought can hold much potential for new breakthroughs as it is radically different from the approaches of traditional image processing; with this notion of thought, theories of statistical mechanics may be refined and used for image processing.

One of the earliest prominent papers that have used the Monte Carlo method for image processing is the one by Geman and Geman of 1984 \cite{4}. The focus of computer vision has since moved away from Monte Carlo techniques because it was generally believed that Monte Carlo methods are too slow in computation. However, Monte Carlo techniques have significantly improved in the last two decades. Sampling methods experienced a speedup of one or more orders of magnitude \cite{5,6}. We believe that Monte Carlo techniques have advanced enough to be entitled a bigger role in computer vision once again.

In this paper, we illustrate an effective use of Monte Carlo by formulating the image segmentation problem in a statistical physics framework. Image segmentation is one of the most important fundamental tasks of image processing \cite{7}, whereby the image is partitioned into segments. Segmentation results in a simpler representation of the original image and this representation is often used for other higher-order cognitive tasks such as objection recognition. As an example, in figure 1, we show a river scene image segmented into three regions, the river, land and sky. Details of the segmentation procedure will be given later in this paper.

There are many thousands of publications on image segmentation, but generally the image segmentation problem is considered unsolved. In 1989, Mumford and Shah \cite{8} developed a segmentation model that has become the hallmark of image segmentation. Many variants of this model were developed \cite{9}–\cite{14} to segment almost any kind of image. A more popular variant
Figure 1. The original image (left) and the segmented image (right). The original size of the image is 864 × 624, \( q = 3, \mu = 1000, T = 0 \) and 100 × (864 × 624) single spin flips at different levels of block-spin transformations. A movie showing the segmentation process is given in supplementary video 1 (available from stacks.iop.org/NJP/13/023004/mmedia).

is one that approximates the image with a cartoon-like piecewise constant intensity image. We consider the energy functional \( F \), defined as

\[
F[C, c_1, \ldots, c_q] = \sum_{i=1}^{q} \int_{\Omega_i} (u(x) - c_i)^2 \, dx + \mu |C|,
\]

(1)

where \( q \) is the number of constant intensities and is also commonly called the number of phases of the segmentation; \( u(x) \) is the original image and \( C \) is a set of curves that partition the image into exclusive segments \( \Omega_i, i = 1, \ldots, q; c_i \) is a set of constants, \( \mu \) is a tuning parameter and \( |C| \) represents the length of the curve measured in Euclidean distance. The actual calculation of \( |C| \) will be explained later in equation (2). The first term accounts for the deviation from constant intensities \( c_i \) and the second term accounts for the regularization of curves partitioning different segments. The objective is to seek the partitioning curve \( C \) and constants \( c_i, i = 1, \ldots, q \), that minimize the functional \( F \). We use figure 1 to illustrate the physical meaning of \( F \) more clearly.

The original image shows three regions of approximately constant intensities, the sky, river and land. The optimal energy functional seeks three constants \( c_1 \) (sky), \( c_2 \) (river) and \( c_3 \) (land), and a curve \( C \) that partitions the image into three regions. In this case, \( c_1, c_2 \) and \( c_3 \) are the average pixel intensity values of the sky, river and land, respectively. We would also like to point out that the number of segmentation phases is given \textit{a priori}.

Much effort has been devoted to solving the Mumford–Shah model. The more popular approaches are level-set, threshold dynamics and graph cut [15]–[21]. One of the most cited among all proposed Mumford–Shah solutions is that of Chan and Vese [15]. However, this method suffers from local minimum trap although techniques have been developed to obviate this problem [22, 23]. Furthermore, in most approaches, the number of phases of segmentation is always given by \( 2^k \), where \( k \) is the number of level-set functions. This limits the freedom of choosing an arbitrary number of phases.

Some attempts to treat multiphases were reported in [24].

We organize the rest of the paper as follows. In section 2, we describe the statistical physics representation of the Mumford–Shah segmentation model. The optimization process with the Monte Carlo is explained in section 3. We present the numerical results in section 4, and a comparison with other methods in section 5. The final section is devoted to the discussion.

New Journal of Physics 13 (2011) 023004 (http://www.njp.org/)
2. Statistical physics representation of the Mumford–Shah model

We rewrite the Mumford–Shah model in a statistical physics framework, in particular in terms of a $q$-state Potts model [25] on a discretized square lattice as

$$F[S] = \sum_{i=1}^{q} \sum_{x} (u(x) - c_i)^2 \delta_{S(x),i} + \mu \sum_{\langle x,y \rangle} (1 - \delta_{S(x),S(y)}),$$

$$c_i = \frac{\sum_{x} u(x) \delta_{S(x),i}}{\sum_{x} \delta_{S(x),i}}. \tag{2}$$

$u(x)$ is the pixel intensity at the pixel location $x$; $c_i$ is a set of constants described in equation (1); $S(x) \in 1, \ldots, q$ is the Potts spin value at $x$; $\mu$ is the tuning parameter described in equation (1). The first term sums over all pixel (lattice) points; this term corresponds to the first term in equation (1). The second term sums over all nearest neighbors on the lattice; this term corresponds to the length term $|C|$ in equation (1).

This representation is also called Markov random fields in the image processing literature. In the $q$-state Potts model representation, we superimpose the image $u(x)$ with a lattice containing Potts spins $S(x) \in 1, \ldots, q$. One can immediately observe several advantages of Potts’ representation of the Mumford–Shah model. Firstly, the number of phases of segmentation, $q$, can be chosen arbitrarily. Next, the second term in equation (2) is the same form as the Potts model. Lastly, the machinery of statistical mechanics and advanced Monte Carlo methods can be used directly to solve the Mumford–Shah model. We use a combination of statistical physics and Monte Carlo methods to develop a very fast solution for the Mumford–Shah model. Our method includes single spin flip, block-spin transformation, Eden clustering and simulated annealing.

3. Optimization

Single spin flip is performed with the standard Metropolis algorithm [5]. This single spin flip is very fast because the change in Mumford–Shah functional can be calculated locally. Details of this calculation are given in appendix A. Figure 2 gives an illustration of a single spin flip move.

Figure 2. A schematic diagram of the optimization steps for two-phase segmentation with pixels labeled as $q = 0$ or 1. (a, b) Original labeling with a randomly chosen pixel shown in dark gray. After the single spin flip step, the label for this pixel is changed to 0 with a probability given in appendix A. (c) For Eden clustering, only pixels at the region boundary (shown in light gray) are randomly chosen. In this figure, the pixel in dark gray is chosen by chance and its label changed to 0.
Figure 3. Intermediate segmentation results at different scales. Block-spin transformation with $12 \times 12$ pixels (top row), $24 \times 24$ pixels (middle row) and $192 \times 192$ (bottom row) pixels for the original image. $q = 4$, $\mu = 1000$ at $192 \times 192$ pixels, annealing scheme starts at $T = 1000$ and $T \to 0.96 T$ for every ten sweeps through the lattice. Fifty sweeps through the lattice were performed at $T = 0$ toward the end of the simulation. A total of $334 \times 192 \times 192$ single-spin-flip updates were performed at different scales. The final Mumford–Shah functional value is $9.353 \times 10^6$. A movie of the segmentation process is given in supplementary video 2 (available from stacks.iop.org/NJP/13/023004/mmedia).

Block spin transformation originates from renormalization group theory [26], where the lattice is coarse grained at several levels. Its purpose is to speed up the numerical computations. For two dimensions, each operation of block-spin transformation reduces the number of pixels by fourfold. Block-spin transformation is performed by taking four neighboring pixels and replacing them by one pixel with the average intensity value of all four pixels. In three dimensions, eight pixels are averaged over. Monte Carlo optimization is performed across all levels of block-spin transformations starting from the coarsest level. Figure 3 demonstrates the process of optimization across levels of block-spin transformations. Take for example the top row with $12 \times 12$ pixels and the middle row with $24 \times 24$ pixels. Optimization is first performed on the image with $12 \times 12$ pixels; after a suboptimal segmentation is obtained, the algorithm switches to optimize the $24 \times 24$ pixel image. The segmented result of $12 \times 12$ pixels was kept...
as the initial condition for segmenting the $24 \times 24$ pixel image, with the $12 \times 12$ pixel segmented image being resized to $24 \times 24$ pixels.

It is important that the Mumford–Shah model (equation (2)) remains invariant under block-spin transformation. A theorem given by Law et al [22] ensures this invariant and gives the rule for setting the parameter $\mu$ across different scales. Essentially, $\mu \to \mu/2$ across each level of block-spin transformation.

At the later stage of segmentation, only spins near the segment boundary have a high chance of flipping. Hence, we employ the method of Eden clustering [27] to take advantage of this observation. For this Monte Carlo move, we choose the pixel at segment boundary with equal probability and perform single spin flip on them. As a parameter in our simulation, we assign an Eden clustering probability $p_{\text{Eden}}$ and, for each Monte Carlo step, perform Eden clustering with probability $p_{\text{Eden}}$ and single spin flip with probability $1 - p_{\text{Eden}}$. Although Eden clustering violates detailed balance, we performed an extensive check and found that, to within error bars, Eden clustering does not affect the final minimized Mumford–Shah functional value. Figure 2 gives an illustration of an Eden clustering move.

Simulated annealing can be used to get away from a local minimum, which enhances ergodicity. Annealing is achieved by adjusting either the temperature $T$ or the Mumford–Shah parameter $\mu$. For initial conditions, all simulations were performed with each pixel assigned to a random initial phase. An outline of our algorithm is as follows:

1. Perform $n$ block-spin transformations on the image so that we have an image pyramid of $n$ levels.
2. At the coarsest level ($n$ level) of block-spin transformation, initialize the segmented image with random phases.
3. Set the initial temperature $T_i$. $T_i$ is a tuning parameter.
4. Starting from the coarsest level ($n$ level) of the image pyramid, perform Monte Carlo optimization:
   (a) Draw a random number $\eta$ from a uniform distribution $U[0, 1]$.
   (b) If $\eta < p_{\text{Eden}}$, perform one Eden clustering move, else perform a single spin flip move.
5. Annealing and changing the level of block-spin transformation are performed concurrently,
   (a) **Annealing**: Repeat steps 4a and 4b $10N$ times and then set the temperature $T \to \alpha T$. $N$ is the number of pixels in the current block-spin transformation level. $\alpha < 1$ is a tuning parameter.
   (b) **Block-spin transformation**: For the $i$th level of block-spin transformation, repeat steps 4a and 4b $\beta_i \times N$ times and then set the level of block-spin transformation to $i - 1$. We use $\beta_i \propto 2^{-i}$ so that higher-level images in the image pyramid undergo exponentially smaller number of Monte Carlo optimization steps. An example of the values of $\beta_i$ is given in table 1. Steps 5a and 5b are performed whichever comes first.
6. Repeat steps 4 and 5 until a predetermined Monte Carlo step is reached.

**4. Results**

Figure 1 shows the segmentation result for a river scene. The number of phases was set to 3, $q = 3$, demonstrating one of the advantages of the $q$-state Potts representation. Three-phase segmentation is not achieved for most level-set approaches.
Figure 4. Six-phase segmentation of a pencil drawing (608 \times 816 pixels) demonstrates that our method can handle multiphase segmentation without running into numerical instabilities or getting trapped in local minima. The figures on the bottom row show zoomed-in images of the house at the top middle of the original image. A good segmentation is obtained with $\mu = 200$ (middle), whereas with $\mu = 0$ (right), segmentation results in single pixel noise. An equivalent of 1800 sweeps through the lattice ($1800 \times 608 \times 816$ single Monte Carlo steps) were performed. Annealing was carried out by starting with $T = 800$ and $T \to 0.99 T$ for every ten sweeps through the lattice.

Figure 3 shows the segmentation of a brain MRI image (http://www.umassmed.edu/bmp/faculty/ross.cfm) at different levels of block-spin transformation. The segmentation can identify important structures such as the tumor in the middle even at the very coarse grained level of $12 \times 12$ pixels (four levels of block-spin transformations). The segmentation iteratively refines itself as more details in the original image are presented.

Figure 4 demonstrates the robustness of our method to numerical instabilities and local minimum traps. The pencil drawing of the house and boat (608 \times 816 pixels) is very complex and requires six phases to segment. It was reported that the level-set method [15] faces a serious local minimum problem even for four-phase segmentation [22]. One notices that the segmented image is visually a good representation of the original image. Zoomed-in images of the house on the top middle of the image show details of the segmentation. The house and background are very well segmented. As a comparison, we present the result for segmentation with $\mu = 0$ (bottom right of figure 4). At $\mu = 0$, segmentation results in single pixel noise.

Figure 5 shows the effectiveness of block-spin transformation and Eden clustering. Convergence is much faster with block-spin transformation as indicated by arrows. Note that the
Figure 5. Convergence rate of segmentation for the brain MRI image (figure 3). Arrows indicate that convergence with block-spin transformation is very effective. Block-spin transformation was performed as described in figure 3. Different symbols are used to represent various values of Eden clustering probability, $p_{\text{Eden}} = 0$ (circles), $p_{\text{Eden}} = 0.2$ (squares), $p_{\text{Eden}} = 0.5$ (diamonds) and $p_{\text{Eden}} = 0.8$ (triangles). The inset shows an enlarged image with block-spin transformation toward the end of the simulation. The annealing parameters used to generate these plots are the same as those used in figure 3.

Table 1. Schedule for block-spin transformation used in figure 3. The third column shows the normalized number of sweeps, e.g. $142 \times 96 \times 96 \approx 35.49 \times 192 \times 192$. These reflect the true algorithm speed that is plotted in figure 5.

| Image size | Number of sweeps through the image | Number of sweeps normalized w.r.t. the 192 × 192 image |
|------------|-----------------------------------|-----------------------------------------------|
| 12 × 12    | 30                                | 0.12                                          |
| 24 × 24    | 35                                | 0.55                                          |
| 48 × 48    | 66                                | 4.12                                          |
| 96 × 96    | 142                               | 35.49                                         |
| 192 × 192  | 298                               | 298                                           |

$x$-axis, showing the equivalent number of Monte Carlo steps per sweep through the 192 × 192 lattice, is plotted in log scale. The inset shows an enlarged region of plots without block-spin transformation. This plot shows the convergence rate for different values of Eden clustering probability $p_{\text{Eden}}$. Convergence is slowest without Eden clustering ($p_{\text{Eden}} = 0$, represented by circles). The block-spin transformation schedule is given by table 1.

Ten independent simulations started with different random initial conditions were performed to generate each data point in figure 5. We obtained very small error bars (when not shown they are smaller than the size of the symbols). Hence, our Monte Carlo method is very robust to initial conditions and statistical fluctuations.
Figure 6. Two-phase segmentation of the tree and sky (624 × 608 pixels) illustrates that convergence rate depends on the length parameter $\mu$. For $\mu = 100$, segmentation converges at 50 × 624 × 608 single spin flips. When $\mu = 1 \times 10^6$, slower convergence can be enhanced by annealing in $\mu$. $\mu$ increases from $\mu = 43$ to $\mu = 1 \times 10^6$ with an increment rate of $\mu \rightarrow \mu / 0.99$ per sweep through the lattice. A movie showing the segmentation process is given in supplementary video 3 (available from stacks.iop.org/NJP/13/023004/mmedia).

Figure 7. Optimized two-phase images of the tree-and-sky image using Monte Carlo and graph-cut. Optimized Mumford–Shah functional values with $\mu = 1000$ are given below the image.

Figure 6 shows two-phase segmentation results for different tuning parameters, $\mu$. Segmentation converges quickly and shows fine details for small $\mu$ ($\mu = 100$). The length of the boundaries of segment is much shorter for large $\mu$ and convergence is slower. It is also observed that in the level-set method [15], convergence is slower for larger $\mu$. We found, as in the right most image of figure 6 ($\mu = 1 \times 10^6$), that annealing in $\mu$ can enhance convergence.

5. Comparisons with other methods

The graph-cut method can guarantee optimization to global minimum for two-phase segmentation [19]–[21]. We can use graph-cut as a benchmark to check how well our Monte Carlo method optimizes the two-phase image. We found that the optimized Mumford–Shah functional value for the two-phase tree-and-sky image (figure 7) is $6.3294 \times 10^8$ using graph-cut and $6.3417 \times 10^8$ using our method, with the parameter $\mu$ as 1000. This shows that our Monte Carlo method optimized the image to within 0.19% of the global minimum. The segmented
Figure 8. Segmented images of the brain MRI, breast cancer cells and zebra fish intestine using Monte Carlo and multiphase graph-cut (El-Zehiry et al [21]). The numbers below the images represent the optimized Mumford–Shah energy functional values.

tree-and-sky image by graph-cut is shown in figure 7(a), and graph-cut segmentation looks visually similar to Monte Carlo segmentation (figure 7(b)).

We also use graph-cut to compare our results for more than two-phase segmentation. First we compared our segmentation algorithm to the graph-cut method of El-Zehiry et al [21]. In this case, the graph-cut does not guarantee finding the global minimum. Figure 8 shows Monte Carlo (a, c, e) and graph-cut segmentation (b, d, f) for the brain MRI, breast cancer cells and zebra fish intestine images. For the brain MRI image, both methods can segment the tumor. But graph-cut could not segment out the gray matter (dark gray area on the surface of the brain), whereas the Monte Carlo method can distinguish between the white matter and the gray matter. The Mumford–Shah functional with $\mu = 1000$ is $4.539 \times 10^7$ for graph-cut and $9.353 \times 10^6$ for Monte Carlo. The optimized value by graph-cut is about 5 times higher. For the breast cancer cells image, the Mumford–Shah functional energy for graph-cut is $3.720 \times 10^8$ and $6.489 \times 10^7$ for Monte Carlo. Graph-cut cannot segment the background of the zebra fish intestine very well. Parameters for the Monte Carlo method are $\mu = 1000$, starting with $T = 2500$, anneal with $T \rightarrow 0.99 \, T$. Four block-spin transformations are performed. An equivalent of 1502 sweeps through the lattice (1502 \times 640 \times 480 single Monte Carlo steps) were performed. For the zebra fish intestine image, the Mumford–Shah functional energy for graph-cut is $4.075 \times 10^7$ and $1.567 \times 10^7$ for Monte Carlo. Graph-cut cannot segment the background of the zebra fish intestine very well. Parameters for the Monte Carlo method are

New Journal of Physics 13 (2011) 023004 (http://www.njp.org/)
\( \mu = 1000 \), starting with \( T = 1000 \), anneal with \( T \rightarrow 0.99 \, T \). Two block-spin transformations are performed. An equivalent of 2550 sweeps through the lattice \((2550 \times 188 \times 156 \) single Monte Carlo steps\) was performed.

We remark here on the Chan–Vese algorithm [15]. Similar test images of the brain MRI, breast cancer cells and zebra fish intestine were used by Law et al [22]. Since the test images in [22] are resized to different sizes, we do not make a direct comparison of energy here. But Law et al showed unambiguously that the Chan–Vese [15] algorithm cannot handle four-phase segmentation well.

As another method of graph-cut, recently Bae and Tai [28] applied graph-cut optimization to multiphase image segmentation. They constructed a graph in a similar way to Ishikawa [29] and Ishikawa and Geiger [30]. They claimed that their algorithm can efficiently minimize the energy, and the computation time is dramatically reduced compared to the gradient descent solution of the level-set approach. The acceleration rate is about 200–400 times depending on the images. We compared the performances of our Monte Carlo method and the graph-cut method of Bae and Tai [28]. As a preliminary work, we tested on the MRI image of brain, figure 3, for four phases. The Monte Carlo method with block-spin transformation arrived at the minimum energy only 0.6% higher than that of graph-cut [28]. Average computation time of Monte Carlo differs by less than 5%. We should note that the graph-cut is strongly dependent on the initial conditions. They cannot reach the minimum solution from some initial conditions. The Monte Carlo is robust to initial conditions. We can start from random initial conditions. Moreover, there is a restriction of the form of the energy functional when the graph-cut method of Bae and Tai [28] is applied. Actually, the second term in equation (2) should be replaced by \( \mu \sum_{(x, y)} |S(x) - S(y)| \). Thus, our method of Monte Carlo is very efficient for multiphase segmentation, and the performance is compatible with the sophisticated graph-cut method. The Monte Carlo has the advantages of generality and robustness. Detailed comparison is left for a future work.

The level-set-based method developed by Song and Chan [17] is similar in essence to our method. It was claimed that their method is very fast but does not guarantee a global minimum solution. The method by Song and Chan does a raster scan on the image to flip pixels if energy decreases as a result. Our method holds several advantages. We use Eden clustering that is much faster than a raster scan; our block-spin transformations can effectively speed up the segmentation as well as enhance ergodicity. If the same annealing scheme is imposed on the method of Song and Chan, it should perform comparably to the plot representing \( p_{\text{Eden}} = 0 \) in the inset of figure 5 (represented by circles).

In addition, our method is robust to statistical fluctuations and does not require a special initial condition. We are able to treat any image, whereas other methods have problems; our method can be trivially generalized to three-dimensional images without sacrificing computational efficiency.

6. Discussions

In this paper, we showed that statistical physics representation for image processing can be effective, when the advanced Monte Carlo method is utilized. We emphasize that the advantage of our method is prominent in the case of multiphase segmentation. The performance of the present method is shown by the movie given in Supporting information. The Java applets for sample images are available at http://ccmp1.phys.se.tmu.ac.jp/ccmp/demo/. Although it takes
time for graphics in the Java applet, the net computation speed is much faster. Timings for all figures are presented in appendix B. In our future work, we would like to extend our statistical physics approach to include topological dependence [31]–[33] and the subspace Mumford–Shah model [9]–[11]. Eden clustering can be used to include a topological dependence constraint developed recently to segment crowded objects [31, 32].

Acknowledgments

This work was supported by a Grant-in-Aid for Scientific Research from the Japan Society for the Promotion of Science.

Appendix A. Details of the calculation for single spin flips

We present the algorithm for performing single spin flips using the mathematical notation defined in equation (2). A pixel \( x_0 \) and a new Potts state \( n \in [1, q] \) are chosen at random. Propose to flip the spin at \( x \) such that \( S(x) \rightarrow S'(x) \), with \( S'(x) = S(x) \) for \( x \neq x_0 \) and \( S'(x) = n \) for \( x = x_0 \). The Mumford–Shah functional changes as \( F[S(x)] \rightarrow F[S'(x)] \) due to this spin flip. Accept the proposed flip with the probability \( \min(1, \exp[-(F[S'] - F[S])/T]) \), where \( T \) is a temperature variable that can be used in a simulated annealing scheme.

A very fast procedure for calculating the change in the Mumford–Shah functional due to single spin flip Monte Carlo moves is as follows. If we write the Mumford–Shah functional in terms of the variables \( N_i = \sum_x \delta S(x), i \), \( U_i = \sum_x u(x) \delta S(x), i \) and \( c_i = U_i / N_i \) and if the pixel at \( x_0 \), \( S(x_0) = m \) is changed to \( S'(x_0) = n \), then the new values \( N'_i, U'_i \) and \( c'_i \) are given by

\[
\begin{align*}
N'_m &= N_m - 1, & N'_n &= N_n + 1, \\
U'_m &= U_m - u(x_0), & U'_n &= U_n + u(x_0), \\
c'_m &= U'_m / N'_m, & c'_n &= U'_n / N'_n.
\end{align*}
\]

Finally, the change in the Mumford–Shah functional is calculated as

\[
F[S'] - F[S] = -c'_m U'_m + c_m U_m - c'_n U'_n + c_n U_n + \mu \Delta |C|,
\]

where \( \Delta |C| \) is the local change in the length of the partitioning curve. We note that all Monte Carlo updates presented in this paper have been performed by the local calculation.

Appendix B. Computational timings

All segmentations in this paper have been performed using a 2.2 GHz Intel Core 2 Duo MacBook. The following table shows the details of computational times.

| Figure   | Time (s) | Figure   | Time (s) |
|----------|----------|----------|----------|
| Figure 1 | 12       | Figure 8(b) | 2       |
| Figure 3 | 3        | Figure 8(c) | 96      |
| Figure 4 | 235      | Figure 8(d) | 18      |
| Figure 6 (\( \mu = 100 \)) | 14 | Figure 8(e) | 14      |
| Figure 6 (\( \mu = 1 \times 10^6 \)) | 124 | Figure 8(f) | 2       |

New Journal of Physics 13 (2011) 023004 (http://www.njp.org/)
Appendix C. Supporting information

QuickTime movies demonstrating figures 1, 3 and 6 are available from stacks.iop.org/NJP/13/023004/mmedia.

References

[1] Tanaka K 2002 Statistical–mechanical approach to image processing J. Phys. A: Math. Gen. 35 R81–150
[2] Ruderman D and Bialek W 1994 Statistics of natural images: scaling in the woods Phys. Rev. Lett. 73 814–17
[3] Pryce J M and Bruce A D 1995 Statistical mechanics of image restoration J. Phys. A: Math. Gen. 28 511–32
[4] Geman S and Geman D 1984 Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images IEEE Trans. Pattern Anal. Mach. Intell. 6 721–41
[5] Landau D P and Binder K 2005 A Guide to Monte Carlo Simulations in Statistical Physics (Cambridge: Cambridge University Press)
[6] Liu S J 2008 Monte Carlo Strategies in Scientific Computing (Berlin: Springer)
[7] Pal N R and Pal S K 1993 A review on image segmentation techniques. Pattern Recognit. 26 1277–94
[8] Mumford D and Shah J 1989 Optimal approximations by piecewise smooth functions and associated variational problems Commun. Pure Appl. Math. XLII 577–685
[9] Law Y N, Lee H K and Yip A M 2009 Supervised texture segmentation using the subspace Mumford–Shah model Proc. 2009 Int. Conf. on Image Processing, Computer Vision and Pattern Recognition (IPCV’09) (Las Vegas, NV: CSREA Press) pp 554–60
[10] Law Y N, Lee H K and Yip A M 2010 Semi-supervised subspace learning for Mumford–Shah model based texture segmentation Opt. Express 18 4434–448
[11] Law Y N, Yip A M and Lee H K 2010 Automatic measurement of volume percentage stroma in endometrial images using texture segmentation J. Microscopy 241 171–8
[12] Yap C K and Lee H K 2008 Identification of cell nucleus using a Mumford–Shah ellipse detector Lecture Notes Comput. Sci. 5358 582–93
[13] Cremers D, Tischhäuser F, Weickert J and Schnörr C 2002 Diffusion Sankes: introducing statistical shape knowledge into the Mumford–Shah functional Int. J. Comput. Vis. 50 295–313
[14] Chan T F, Sandberg B Y and Vese L A Active contours without edges for vector-valued images J. Vis. Commun. Image. Represent. 11 130–41
[15] Chan T F and Vese L A 2001 Active contours without edges IEEE Tran. Image Process. 10 266–77
[16] Merriman B, Bence J K and Osher S J 1994 Motion of multiple junctions: a level set approach J. Comput. Phys. 112 334–63
[17] Song B and Chan T A 2002 Fast algorithm for level set based optimization UCLA CAM Report 02-68
[18] Esedoglu S and Tsai Y R Threshold dynamics for the piecewise constant Mumford–Shah functional J. Comput. Phys. 211 367–84
[19] Zeng X, Chen W and Peng Q 2006 Efficiently solving the piecewise constant Mumford–Shah model using graph cuts Technical Report Department of Computer Science, Zhejiang University, People’s Republic of China
[20] El-Zehiry N, Xu S, Sahoo P and Elmaghriby A 2007 Graph cut optimization for the Mumford–Shah model Proc. IASTED Int. Conf. on Visualization, Imaging and Image Processing
[21] El-Zehiry N and Elmaghriby A 2007 Brain MRI tissue classification using graph cut optimization of the Mumford–Shah functional Proc. Image Vision Comput. pp 321–6
[22] Law Y N, Lee H K and Yip A M H 2008 A multiresolution stochastic level set method for Mumford–Shah image segmentation IEEE Trans. Image Process. 17 2289–300
[23] Chan T F, Esedoglu S and Nikolova M 2006 Algorithms for finding global minimizers of denoising and segmentation models SIAM J. Appl. Math. 66 1632–48
[24] Brox T and Weickert J 2004 Level set based image segmentation with multiple regions *Pattern Recognit.* **31**75 415–23
[25] Wu F Y 1982 The Potts model *Rev. Mod. Phys.* **54** 235–68
[26] Fisher M E 1998 Renormalization group theory: its basis and formulation in statistical physics *Rev. Mod. Phys.* **70** 653–81
[27] Eden M 1961 A two-dimensional growth process *Proc. 4th Berkeley Symp. on Mathematics, Statistics and Probability* vol 4 (Berkeley, CA: University of California Press) pp 223–39
[28] Bae E and Tai X C 2009 Graph cut optimization for the piecewise constant level set method applied to multiphase image segmentation *Scale Space and Variational Methods in Computer Vision (Lecture Notes in Computer Science* vol 5567) (Berlin: Springer) pp 1–13
[29] Ishikawa H 2003 Exact optimization for Markov random fields with convex priors *IEEE Trans. Pattern Anal. Mach. Intell.* **25** 1333–6
[30] Ishikawa H and Geiger D 1998 Segmentation by grouping junctions *CVPR 1998: Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (Washington, DC) (Los Alamitos, CA: IEEE Computer Society) pp 125–31
[31] Yu W M, Lee H K, Hariharan S, Bu W Y and Ahmed S 2008 Level set segmentation of cellular images based on topological dependence *Proc. ISVC 2008 (Lecture Notes in Computer Science* vol 5358) (Berlin: Springer) pp 540–51
[32] Yu W M, Lee H K, Hariharan S, Bu W and Ahmed S 2008 Quantitative neurite outgrowth measurement based on image segmentation with topological dependence *Cytometry A* **75** 289–97
[33] Yu W M, Lee H K, Hariharan S, Bu W and Ahmed S 2010 Evolving generalized Voronoi diagram of active contours for accurate cellular image segmentation *Cytometry A* **77** 379–86