EURASIP Journal on Applied Signal Processing 2005:12, 1845–1853

An Integrated Dynamic Scene Algorithm for Segmentation and Motion Estimation

Ikhlas Abdel-Qader
Electrical and Computer Engineering Department, College of Engineering and Applied Sciences, Western Michigan University, Kalamazoo, MI 49008-5314, USA
Email: abdelqader@wmich.edu

Tomislav Bujanovic
Electrical and Computer Engineering Department, College of Engineering and Applied Sciences, Western Michigan University, Kalamazoo, MI 49008-5314, USA
Email: tombujanovic@yahoo.com

Received 8 June 2004; Revised 3 December 2004; Recommended for Publication by Luciano da F. Costa

Segmentation and motion estimation are two problems that require accurate estimation for many applications in computer vision and image analysis. This work presents a solution to these two problems simultaneously. Both the segmentation and motion fields are integrated and estimated in parallel to reduce computation time. The presented algorithm is based on producing motion estimates and restored pixel intensity values through an optimization process that uses deterministic mean-field annealing (MFA) framework. The MFA results at different temperature values are used to run a segmentation process using the concept of region-growing-based algorithm. The segmentation process starts at high temperatures and continues in parallel to the annealing process to refine the segmentation process at lower temperatures. The algorithm results are good and dependent on the annealing parameters. Several experimental results from synthetic and real-world sequences are presented.

Keywords and phrases: segmentation, motion estimation, mean-field annealing.

1. INTRODUCTION

Accurate estimation of motion information and scene segmentation is the focus of investigators in many discipline areas for a variety of applications. Visual motion analysis is necessary for applications such as target tracking, video coding, automatic surveillance, remote sensing, image compression, and many other real-life applications. The estimation of the motion (displacement) fields is the first step in many applications. The methods of estimating the motion fields can be categorized into three groups: the gradient methods (known also as the optical flow); the correspondence methods; and block-matching methods. Each group has its own advantages, disadvantages, and limitations. In this work, the algorithm incorporates the methods of the first category. One of the main advantages of this category is the ability to provide dense displacement fields at subpixel accuracy as opposed to the algorithms in the other two categories that produce displacement vectors for blocks or for predetermined tokens (sparse motion fields) [1, 2]. Segmentation on the other hand aims to segment the scene into different objects or into objects and background. In general terms, image segmentation can be described as the process of generating pixel labels at each pixel. These labels are intended to group pixels into different segments, objects, or partitions. Algorithms can be categorized as spatial segmentation, temporal segmentation, or combination of temporal-spatial segmentation. Some algorithms are based on simple tools such as the histograms and others are very computationally expensive such as MRF-label modeling. Optimization algorithms such as mean-field annealing (MFA), stochastic simulated annealing (SSA), and iterative conditional mode (ICM) [3, 4, 5] among others are used to solve segmentation problems in general. MFA has proven its superiority on others in two ways, computational complexity and optimality of the solution [6], and thus was chosen for this work. In all, algorithms are based on energy functions that are using image attributes in one part or two and iterative minimization is used to reach an optimum global solution. A review of segmentation literature can be found in [7, 8].

To combine these two problems, segmentation of dynamic scenes and motion estimation algorithms have been investigated. In some algorithms, the segmentation would be a preprocess for motion estimation while in others motion estimation would be a preprocess for segmentation [9, 10, 11]. Most algorithms used edge-based motion detection to

decide on the different segments or vice versa such as [12, 13]. Others do not estimate motion as much as use some motion information to perform the segmentation [14, 15] or to identify moving objects in a sequence of frames [16]. Detecting boundaries and simultaneously computing motion have been reported in [7]. The work presented in [9, 10] is based on Bayesian decision associated with MRF models, but their models need line processors to represent motion discontinuity. These algorithms depend totally on line processors that are a significant disadvantage due to the increase of computational complexity.

MFA was previously used to produce dense motion estimates without considering any segmentation models [17]. In this work, MFA is used to generate estimates of motion vectors and restored pixels at different temperature values. At each temperature value, the estimates are used to generate a segmented scene by integrating the segmentation model within the MFA framework. Further, the optical flow computation assumes that the displacement field is smooth over segments that belong to one object. Hence, the merging of a segmentation process with the displacement estimation will enhance the results and reduce the computations. It is important to point out the major differences between those models and the work presented here. All had the segmentation as a by-product resulting from motion computation as opposed to simultaneously estimating both. Moreover, the noise effects on the algorithm's performance were not considered nor were any noise models investigated. Furthermore, algorithms reported are heavily dependent on initial values of segments or motion vectors. This algorithm does not require any prior information. Indeed, this algorithm begins with zero motion for every pixel and each pixel is a region by itself.

The integrated framework is presented in the next section over three subsections, the motion model, the annealing process describing the process of estimating motion, and the segmentation model. Section 3 introduces the experimental results, while the concluding remarks are presented in Section 4.

2. MOTION AND SEGMENTATION INTEGRATED FRAMEWORK

This framework attempt to employ the local interactions between a pixel and its neighbors and the temporal information over two frames to estimate each pixel's displacement and true intensity values, and further, to identify which segments would a pixel belong to, all in a parallel manner. The displacement variable and the true intensity values are modeled as a MRF. Then, it uses the equivalence of Gibbs distribution to MRF to implement the solution [18]. A maximum a posteriori (MAP) is determined by solving for a global minimum of the energy function instead of solving for the most probable state in the a posteriori probability distribution, which is usually a very hard task to achieve. In the following section, a description of the algorithm and the different parameters considered for both models (motion and segmentation) is presented. The fact that this model [17] accounts for the noise is very useful to the segmentation since it will allow the segmentation process to use the restored pixel values. The motion estimation process is summarized in the following section for the completion of this paper.

2.1. Displacement fields model

The displacement field is modeled under the assumptions of smooth and piecewise constant fields. The posterior distribution of the displacement field and the true underlying image intensities of the two frames under consideration are modeled explicitly. In these, the displaced pixel differences were strictly required to have exact numerical identity between displaced pixels wherever possible. In this work, the displaced pixel differences are modeled as observation noise in the pixels themselves. This is achieved explicitly by modeling the noise observation in the noise fields, which will significantly improve the results. Also, in this work, the motion field is expected to vary smoothly spatially. Thus one would expect to find smooth displacement fields in the image plane. In other words, pixels close to each other and within the same object tend to have the same displacement, that is, a piecewise constant displacement field.

Mathematically, the displacement energy model $H_d(d)$ can be written as

$$H_d(d) = \lim_{\tau\to 0} \sum_i \sum_{j\in\aleph} \frac{-\alpha_i}{\sqrt{2\pi\tau}} e^{-\|d_i - d_j\|^2/2\tau}, \tag{1}$$

where α_i is a weighting factor, and d_i and d_j are the displacement vectors for pixel i and j, respectively, such that $d_i = [d_{xi} \;\; d_{yi}]^t$, where d_{xi} is the horizontal component of the displacement vector of pixel i, and d_{yi} is the vertical component of the displacement of pixel i. $\aleph$ is the neighborhood of pixel i (denoting the image as $N^2 \times 1$ vector). The $\|\cdot\|$ designates the norm of the difference between the two displacement vectors. Taking the limit of H_d in (1) as τ approaches zero would make the energy function approximates a differentiable Dirac delta function (notice that it is a Dirac function at the correct displacement). Also, it shows that a minimum can be approached when $d_i = d_j$ which satisfies the constant displacement field.

To capture the temporal attributes in a model suitable for annealing algorithms, this model is generated which captures the intensity gradient,

$$H_g(d) = \lim_{\tau\to 0} \sum_i \frac{-\beta_i}{\sqrt{2\pi\tau}} e^{-(I_{t_1}[i] - I_{t_2}[i+d_i])^2/2\tau}, \tag{2}$$

where $I_{t_1}(i)$ is the intensity at location i in frame t_1, $I_{t_2}(i + d_i)$ is the intensity at location $i + d_i$ in frame t_2, and β_i is the weighting coefficient for this part of the energy function. This function assumes that the intensity is preserved under motion [17]. This model leads to the minimization of the following function using Taylor's series expansion to express the model as a function of the displacement directly:

$$H_g(d) = \sum_i \frac{-\beta_i}{\sqrt{2\pi\tau}} e^{-(dI/dt + (dI_{t_2}/dx)dx_i + (dI_{t_2}/dy)dy_i)^2/2\tau}. \tag{3}$$

Notice that from (2), one can conclude that noise will have strong impact on the results. It is worth to note that the model developed is using the restored data, that is, the contribution of the restoration model and not the actual noisy image data is used simultaneously with the segmentation model. The noisy data will be denoted by g_1 and g_2, for frame 1 and frame 2, respectively. The noise model adapted in this framework is the Gaussian additive noise. This is the most common type of independent noise [19] and it is typical in video frames, leading to the following model:

$$g = I + n, \tag{4}$$

where I is the restored signal, g is the noisy data, and n is the additive Gaussian random noise with variance of σ_n^2. The noise energy function is modeled as

$$H_n(I) = \frac{1}{2\sigma_n^2} \sum_i \left[(I_{1i} - g_{1i})^2 + (I_{2i} - g_{2i})^2 \right]. \tag{5}$$

2.2. The annealing process

After joining all parts of the energy function, the algorithm seeks an estimate of the following vector for each pixel in the image:

$$f_k = \left[d_{xk} \quad d_{yk} \quad I_{1k} \quad I_{2k} \right], \tag{6}$$

where d_{xk} is the horizontal component of the displacement vector of pixel k; d_{yk} is the vertical component of the displacement vector of pixel k; I_{1k} is the true intensity value of pixel k; I_{2k} is the true intensity value of pixel k. The noisy data will be denoted by g_{t1} and g_{t2}, for frame 1 and frame 2.

The mean-field vector μ is given by

$$\mu_k = \left[\mu_{xk} \quad \mu_{yk} \quad \mu_{I_1k} \quad \mu_{I_2k} \right]^t, \tag{7}$$

where μ_{xk} and μ_{yk} are the mean-field parameters for the horizontal and the vertical components of the displacement field, respectively. The parameters μ_{I_1k} and μ_{I_2k} are the mean-field parameters for the true intensity value of pixel k in both frames. The energy function consists of three parts, the displacement function, H_d, the intensity function, H_g, and the noise function, H_n. The energy function is now a function of the vector f:

$$H(f) = H_d(f) + H_g(f) + H_n(f). \tag{8}$$

The energy function $H(f)$ is in general a function that is rich in local minima and may be ill-behaved in other ways as well. Instead of minimizing $H(f)$, MFA approximates H by another function H_0, called mean-field energy function. H_0 is assumed to resemble H, but it is simpler in form and easier to minimize. Therefore, the first step in using MFA is to choose H_0. For many image processing problems, a quadratic function for H_0 has been shown to be suitable [20] and Gibbs distribution has been used in similar optimization. These are

$$\begin{aligned} H_0(f,\mu) &= \sum_k \|f_k - \mu_k\|^2, \\ p_0(f,\mu) &= \frac{1}{Z_0} e^{-H_0(f,\mu)/T}. \end{aligned} \tag{9}$$

Gradient descent is used for its simplicity only. The algorithm performance is dependent on many factors and parameters and not just on the choice of energy function models. The choice of parameters values α and β is very crucial to the performance. The different choices for their ratios shift the emphasis in the optimization on the different components of the energy function model. Annealing schedule is important as the simulations results show in the following section. The results are the displacement fields displayed as optical flow over the scene and the different segments displayed with different gray values. Also, the algorithm continues for a final partitioning process based on the estimates of the vector f_k.

2.3. Segmentation fields model

Visually, it is agreed on that different segments in an image are of different colors or of different gray-scale tones, sizes, shapes, textures, patterns, and/or shadows [7]. The segmentation process developed in this work is based on employing spatial (gray-scale values) and motion information at different temperatures in a region-growing manner to generate the segmented scene. Other image characteristics that stem from certain applications may be incorporated in the models. In [21, 22], a framework is generated using MFA based on homogeneity measure for the purpose of segmenting images only. In general, models may include any characteristics such as homogenous regions in the image, objects of certain shapes, speeds, or of specific texture. The more constraints are imposed, the more the algorithm is application dependent. Thus, this work employed general characteristics in order for the algorithm to be suitable for many applications with motion.

The algorithm is initiated by adapting the results of the MFA algorithm at low temperatures to identify objects or segments that are larger than one pixel. All parts of the Hamiltonian model will merge their computational contributions to identify those segments that are most likely to belong to the same object. At the end of each iteration (temperature reduction), the segmentation process utilizes current displacement and intensity information of a pixel. The algorithm proceeds in parallel to the MFA computations as described in Figure 1.

(1) Initially, assume that each pixel is a separate partition. Assign labels for them from 1 to $N^2 \times 1$ (the image size).

(2) Compute displacement fields and apply a selective filter to increase homogeneity of displacement fields at the very first iteration. Such a process can be achieved by using

$$\mu_{ke} = \frac{8\mu_k + \sum_{l \in \aleph_{ke}} \mu_l}{8 + n_{ke}}, \tag{10}$$

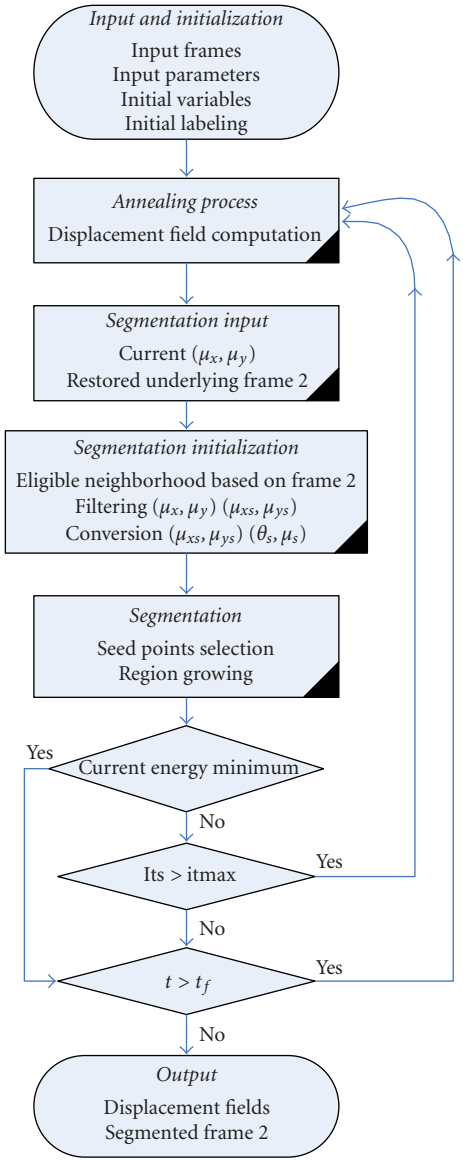


Figure 1: Description of the integrated algorithm.

where $\aleph_{ke}$ represents the eligible neighborhood of pixels k and n_{ke} represents the number of its eligible neighbors. The measure for eligibility is the threshold of the restored true intensity of two neighbors in the underlying frame.

(3) Segmentation decision is made in the region-growing procedure based on both current motion information (magnitude and direction) and underlying restored intensity of each pixel. Once a cycle of iterations is performed, a pixel will take over the least label, which is appearing in its eligible neighborhood. The regions are growing from the pixels with the lower current label toward the pixels with the higher one. Pixel k motion information is switched form μ_{xk} and μ_{yk} (the horizontal and vertical displacement values) to the phasor form of magnitude μ_k and direction θ_k.

(4) Regions can be labeled as scene background objects (stationary) or scene moving objects. When the energy function reaches its first minimum, the pixels with motion information under a given magnitude level (seed threshold) in it and its neighborhood will be assigned as the seeds of the background (no intensity change between the frames).

(5) Identifying different segments is performed using different divergence formulas for the scene background and moving objects. These are as follows.

(i) *Scene background* is recognized as the partitions with "small or zero" motion magnitude. Background partition is growing from the seed points defined by

$$\frac{8\mu_k + \sum_{l\in\aleph_k}\mu_l}{16} \leq \text{seed\_threshold}, \tag{11}$$

where μ_k is the magnitude in the seed point (pixel k) and μ_l are the magnitudes over all 8 neighbors of the seed point (the neighborhood $\aleph_k$ of the pixel k).

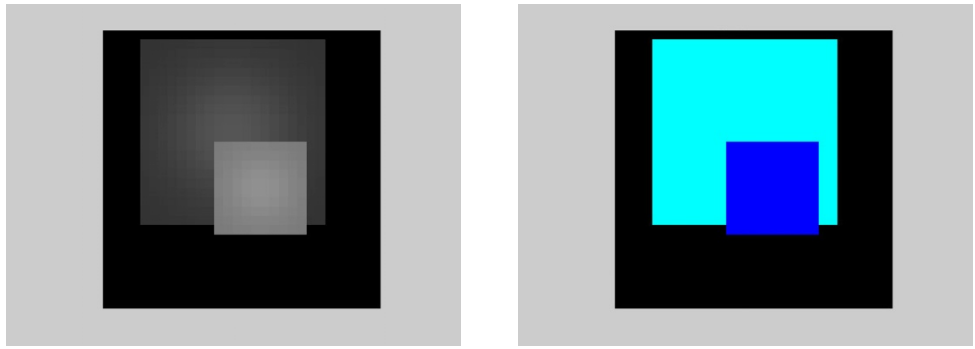
The distance $D^b_{k,l}$ between any two neighbor-pixels k and l, using the formula applicable for small motion magnitude, is defined as

$$\begin{aligned} D^b_{k,l} &= \gamma^b_\mu (\Delta\vec{\mu}_{k,l})^2 + \gamma^b_I (\Delta I_{k,l})^2 \\ &= \gamma^b_\mu (\mu_k^2 + \mu_l^2 - 2\gamma^b_\theta \mu_k\mu_l \cos(\theta_k - \theta_l)) + \gamma^b_I (I_k - I_l)^2, \end{aligned} \tag{12}$$

where $D^b_{k,l}$ is the distance between two pixels, $\Delta\vec{\mu}_{k,l}$ and $\Delta I_{k,l}$ are motion and restored underlying intensity distance, and γ^b_μ, γ^b_θ, and γ^b_I are parameters that enable equal or similar contributions of both motion information and underlying intensity in the formula. Also, these neighborhood parameters are associated with the displacement magnitude μ, the displacement direction θ, and the restored data μ_k and μ_I for pixels k and l. The parameters w_1 and w_2 decrease contribution of motion magnitude and underlying intensity in the distance formula since this is for nonmoving objects, respectively, as follows:

$$\begin{aligned} D^{b1}_{k,l} &= \frac{\gamma^b_\mu}{w_1}(\Delta\vec{\mu}_{k,l})^2 + \gamma^b_I (\Delta I_{k,l})^2 \\ &= \frac{\gamma^b_\mu}{w_1}(\mu_k^2 + \mu_l^2 - 2\gamma^b_\theta \mu_k\mu_l \cos(\theta_k - \theta_l)) + \gamma^b_I (I_k - I_l)^2, \\ D^{b2}_{k,l} &= \gamma^b_\mu (\Delta\vec{\mu}_{k,l})^2 + \frac{\gamma^b_I}{w_2}(\Delta I_{k,l})^2 \\ &= \gamma^b_\mu (\mu_k^2 + \mu_l^2 - 2\gamma^b_\theta \mu_k\mu_l \cos(\theta_k - \theta_l)) + \frac{\gamma^b_I}{w_2}(I_k - I_l)^2. \end{aligned} \tag{13}$$

Applying a weighing formula, we choose a set of the neighbors that are similar to the pixel k: $D^{b1}_{k,l} \leq$ Threshold_1, and $D^{b2}_{k,l} \leq$ Threshold_1. Among the eligible neighbors we assign labels $L^1_k = \min_{\aleph_k}(L_l)$ and $L^2_k = \min_{\aleph_k}(L_l)$. Finally, the partition label is chosen using $L^b_k = \max(L^1_k, L^2_k)$, which generally follows both motion and underlying intensity conditions.



(a) (b)

Figure 2: (a) Frame 2 of synthetic sequence with two overlapping objects moving in the same direction on a stationary background with different speeds. (b) The segmented image.

(ii) *Moving objects* divergence formula is employed to enable growing of a partition in motion. In the case of the objects in motion, the motion magnitude is larger and relative motion magnitude is applicable:

$$D_{k,l}^{o1} = \gamma_\mu^o \frac{(\Delta\vec{\mu}_{k,l})^2}{\exp(|\mu_k|)} + \gamma_I^o (\Delta I_{k,l})^2 = \gamma_\mu^o \frac{\mu_k^2 + \mu_l^2 - 2\gamma_\theta^o \mu_k \mu_l \cos(\theta_k - \theta_l)}{\exp(|\mu_k|)} + \gamma_I^o (I_k - I_l)^2. \tag{14}$$

The parameters k_3 and k_4 decrease contribution of motion magnitude and underlying intensity in the divergence formula, respectively, as follows:

$$\begin{aligned} D_{k,l}^{o1} &= \frac{\gamma_\mu^o}{w_3} \frac{(\Delta\vec{\mu}_{k,l})^2}{\exp(|\mu_k|)} + \gamma_I^o (\Delta I_{k,l})^2 \\ &= \frac{\gamma_\mu^o}{w_3} \frac{\mu_k^2 + \mu_l^2 - 2\gamma_\theta^o \mu_k \mu_l \cos(\theta_k - \theta_l)}{\exp(|\mu_k|)} + \gamma_I^o (I_k - I_l)^2, \\ D_{k,l}^{o2} &= \gamma_\mu^o \frac{(\Delta\vec{\mu}_{k,l})^2}{\exp(|\mu_k|)} + \frac{\gamma_I^o}{w_4} (\Delta I_{k,l})^2 \\ &= \gamma_\mu^o \frac{\mu_k^2 + \mu_l^2 - 2\gamma_\theta^o \mu_k \mu_l \cos(\theta_k - \theta_l)}{\exp(|\mu_k|)} + \frac{\gamma_I^o}{w_4} (I_k - I_l)^2. \end{aligned} \tag{15}$$

Among the eligible neighbors, we assign $L_k^3 = \min_{\aleph_k}(L_l)$ and $L_k^4 = \min_{\aleph_k}(L_l)$, respectively.

Finally, $L_k^o = \max(L_k^3, L_k^4)$, which follows both motion and underlying intensity conditions. Following both conditions is important particularly at the higher temperatures of the cooling process, when there is no significant motion information (algorithm starts with zero motion for every pixel).

(6) Repeat region-growing procedure for every annealing iteration up to the freezing point. One can speed up the process and reduce the computational complexity and time by processing background pixels and moving-object pixels in parallel and independent from each other. Also, both processes are ongoing in parallel, leading to reduction of computation time in half.

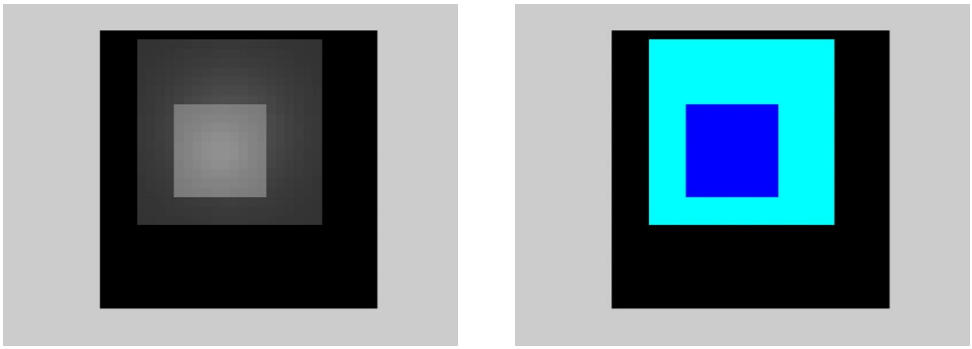
It is clear that the algorithm forces the math to rely on the motion information more than on intensity changes over the moving segments while it does the opposite for the stationary segments.

3. EXPERIMENTAL RESULTS AND DISCUSSION

Synthetic and real scenes are used in the simulations. Also, simulations were executed to cover several possibilities of the motion estimation parameter values, annealing schedules, number of iterations, and the segmentation parameters. The simulations were performed using Matlab and did not consider real-time applications. The code was not optimized. However, fast annealing implementations of MFA or hardware are available in literature [23, 24, 25], and an investigation of such implementations may be an interesting extension of this work. The following is a sample of the simulation results.

3.1. Experimenting with synthetic images

Simulated sequences were generated with different motion magnitudes and motion directions. In Figure 2a, two overlapping objects in motion in similar directions but with different motion magnitudes are shown. Figure 3a is showing objects in different directions. The resulted segmented scene is shown in Figures 2b and 3b, respectively. Parameter values used in both simulations are $\alpha = 6.8$, $\beta = 5.4$, temperatures are $t_i = 20$, $t_f = 1$. Segmentation process parameters for stationary objects are Threshold_1 = 0.05, Threshold_seed = 0.1, $w_1 = 1000$, $w_2 = 2$, $\gamma_\mu^b = 30$, $\gamma_\theta^b = 1$, and $\gamma_I^b = 0.001$. Moving objects parameters are Threshold_2 = 800, $w_3 = 1000$, $w_4 = 10$, $\gamma_\mu^o = 1200$, $\gamma_\theta^o = 1$, and $\gamma_I^o = 12$. Both synthetic sequences produced correct results for both motion and segmentation.



(a) (b)

Figure 3: (a) Frame 2 of synthetic sequence with two overlapping objects moving in different directions and different speeds on a stationary background. (b) The segmented image.

Figure 4: Frame 1 of the real-world scene (moving object(s)).

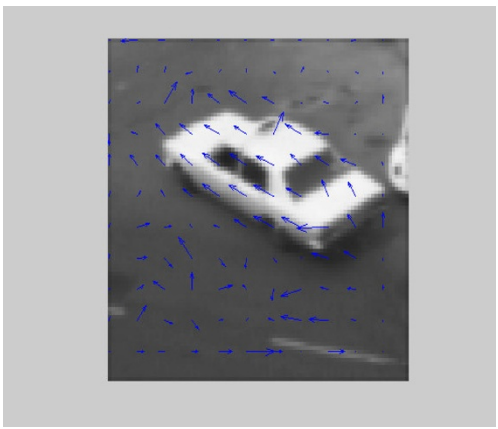


Figure 5: Zoom into the scene with the motion vectors superimposed on the scene (optical flow vectors with block size of 8 and $\sigma = 0.5$).

3.2. Experimenting with real images

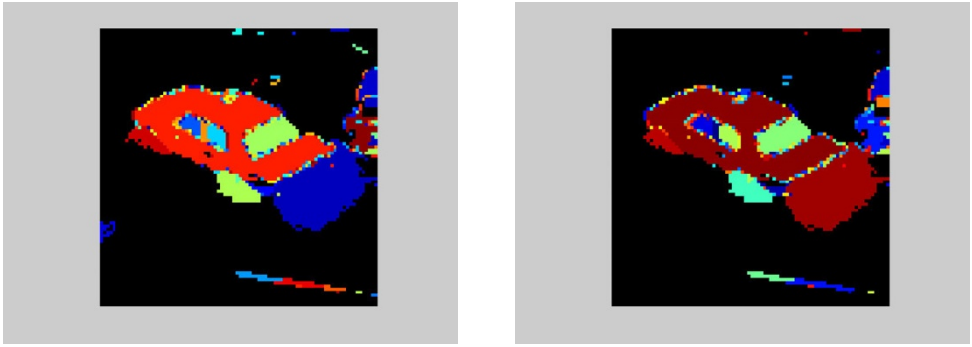
Figure 4 is showing the real-scene sequences used in simulation of this algorithm. Results are presented in the following figures with the zoom into the different scene parts to show the segmentation results clearly. Figure 5 is showing the middle car with the motion vectors superimposed on the scene.

The algorithm parameters values used are as follows: $\alpha = 6.8$, $\beta = 5.4$, and $t_f = 1$. The stationary parameters are Threshold_1 = 0.1, Threshold_seed = 0.1, $w_1 = 1000$, $w_2 = 2$, $\gamma_\mu^b = 30$, $\gamma_\theta^b = 1$, and $\gamma_I^b = 0.001$. Moving object parameters are Threshold_2 = 800, $\gamma_\mu^o = 1000$, $\gamma_\theta^o = 1$, $\gamma_I^o = 4$, $w_3 = 1000$, and $w_4 = 10$; real scene segmentation results are shown in Figures 6, 7, and 8. It is worth noting that the thresholding parameters were selected based on the values of the neighborhood intensity averages, which make these thresholding values have dependency on the texture of the object.

Also, to show robustness of the algorithm, several simulations on real data were used. Figure 9 shows two segmented images for the same scene using different parameter values. In Figure 10, the famous ping-pong ball sequence is used in simulations. Figure 10a is showing a zoom in on the first quarter of the frame while the rotating ball segmentation is shown in Figure 10b.

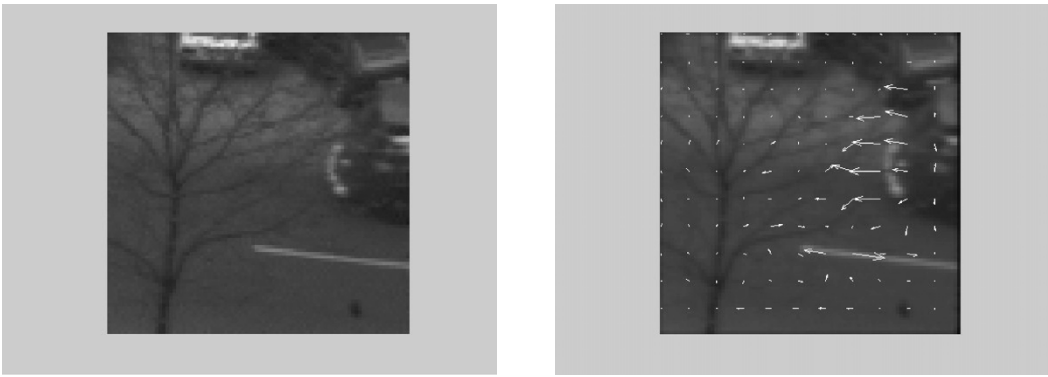
4. CONCLUSION

A general framework is generated to accomplish more than one result. The framework can be tailored to target any specific data by adding more constraints to the energy function. The algorithm is based on Markov random fields (MRFs) modeling and Gibbs distribution equivalence that allowed the use of MFA algorithm to estimate motion fields and restored pixel values. To generate a segmented scene of the image, the mean-field values were incorporated into growing-region segmentation process over several iterations. The algorithm produces accurate segmentations and displacement fields of the moving objects. Generally, procedure of computation of the displacement fields of two (or more) different objects may be done independently of each other, because total energy function of the motion of the image is a



(a) (b)

Figure 6: Segmentation results for different annealing schedules for the real scene shown in Figure 5. The algorithm is robust against parameter value changes. In (b), the final temperature value is 0.1.



(a) (b)

Figure 7: (a) The original scene and (b) the optic flow results superimposed on the car leaving the scene on the right-hand side of the frame as shown in Figure 4.

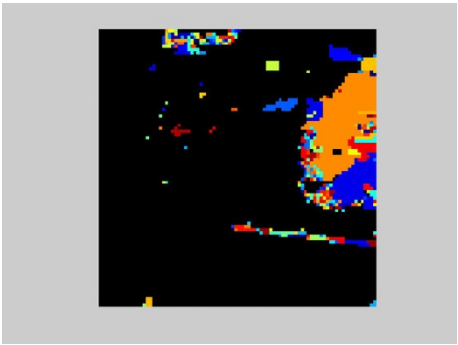
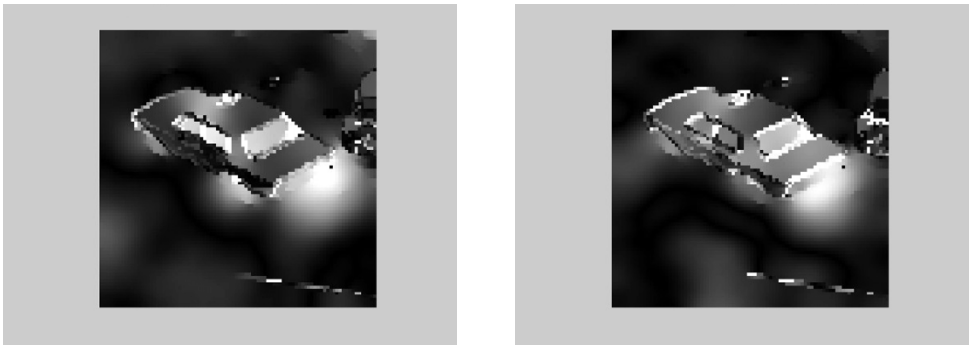


Figure 8: Segmentation results of the scene in Figure 7a.

combination of the energy functions of each object individually and may not be with just one maximum. It is significant to emphasize that there was no preprocessing of any data nor there is any postprocessing of results. This algorithm is efficient when both motion estimates and segmentations are needed simultaneously. That is when the application requires that both motion information and segmentation be calculated, then an integrated algorithm as presented in this paper would be the best because of the savings in time and computations. Future extension of this work is to incorporate the segmentation labels in the MRF model and use the same annealing schedule.

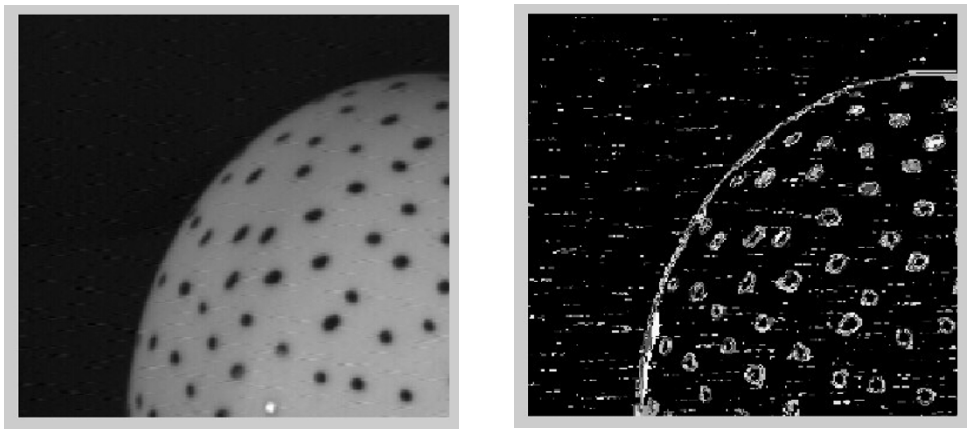
ACKNOWLEDGMENTS

This work was funded by the National Science Foundation Grant MRI- 0215356. The authors would like also to acknowledge Western Michigan University for its support and contributions to the Information Technology and Image Analysis (ITIA) Center. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation or the Western Michigan University. The authors also acknowledge the Journal Editors and the anonymous reviewers for their valuable comments and suggestions.



(a) (b)

Figure 9: Two segmentation results for the same real scene. (a) Object displacement abs(μ_{xs}), $t_f = 10$, $K = 10$. (b) Object displacement abs(μ_{ys}), $t_f = 10$, $K = 10$.



(a) (b)

Figure 10: (a) A zoom-in on the rotating ball [26] and (b) the segmentation results.

REFERENCES

[1] B. Horn and B. Schunck, "Determining optical flow," *Artificial Intelligence*, vol. 17, no. 1–3, pp. 185–203, 1981.

[2] E. Hildreth, "Computations of the velocity fields," *Proceedings of the Royal Society of London*, no. 221, pp. 189–220, 1984.

[3] D. Miller, P. Bunyaratavej, and Q. Zhao, "A sequence-based generalization of mean-field annealing using the forward/backward algorithm: application to image segmentation," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP '02)*, vol. 1, pp. I-969–I-972, Orlando, Fla, USA, May 2002.

[4] M. Goktepe, V. Atalay, N. Yalabik, and C. Yalabik, "Unsupervised texture based image segmentation by simulated annealing using Markov Random field and potts models," in *Proc. 14th International Conference on Pattern Recognition (ICPR '98)*, vol. 1, pp. 820–822, Brisbane, Queensland, Australia, August 1998.

[5] J. Fwu and P. M. Djuric, "Unsupervised vector image segmentation by the ICM method," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP '96)*, vol. 4, pp. 2235–2238, Atlanta, Ga, USA, May 1996.

[6] J. Konrad and E. Dubois, "Comparison of stochastic and deterministic solution methods in Bayesian estimation of 2D motion," *Image and Visual Computing*, vol. 8, no. 4, pp. 304–317, 1990.

[7] N. R. Pal and S. K. Pal, "A review on image segmentation techniques," *Pattern Recognition*, vol. 26, no. 9, pp. 1277–1294, 1994.

[8] Y. Zhang, "Evaluation and comparison of different segmentation algorithms," *Pattern Recognition Letters*, vol. 18, no. 10, pp. 963–997, 1997.

[9] T. Tian and M. Shah, "Motion estimation and segmentation," *Machine Vision and Applications*, vol. 9, no. 1, pp. 32–42, 1996.

[10] J. Fan, J. Yu, X. Xue, and L. Wu, "Efficient motion estimation algorithm based on structure segmentation and compensability analysis," *Journal of Optical Engineering*, vol. 37, no. 5, pp. 1563–1570, 1998.

[11] S. Lee, J. Choi, and S. Kim, "Scene segmentation using combined criterion of motion and intensity," *Journal of Optical Engineering*, vol. 36, no. 8, pp. 2346–2352, 1997.

[12] A. Spoerri and S. Ullman, "The early detection of motion boundaries," in *Proc. 1st International Conference on Computer Vision (ICCV '87)*, pp. 209–218, London, UK, June 1987.

[13] I. Overington, "Gradient-based flow segmentation and location of the focus of expansion," in *Proc. 3rd Alvey Vision Conference (AVC '87)*, pp. 169–177, Cambridge, UK, September 1987.

[14] M. Ben-Ezra, S. Peleg, and B. Rousso, "Motion segmentation using convergence properties," in *Proc. APRA Image Understanding Workshop (IUW '94)*, vol. 2, pp. 1233–1235, Monterey, Calif, USA, November 1994.

[15] O. Shakernia, R. Vidal, and S. Sastry, "Multibody motion estimation and segmentation from multiple central panoramic views," in *Proc. IEEE International Conference on Robotics and Automation (ICRA '03)*, vol. 1, pp. 571–576, Taipei, Taiwan, September 2003.

[16] D. Cremers, "A variational framework for image segmentation combining motion estimation and shape regularization," in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '03)*, C. Dyer and P. Perona, Eds., vol. 1, pp. 53–58, Madison, Wis, USA, June 2003.

[17] I. Abdel-Qader, "Computation of displacement fields in noisy images," *International Journal of Machine Graphics and Vision*, vol. 6, no. 3, pp. 363–380, 1997.

[18] S. Geman and D. Geman, "Stochastic relaxation, gibbs distribution, and Bayesian restoration of images," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 6, no. 6, pp. 721–741, 1984.

[19] T. Chen, K.-K. Ma, and L.-H. Chen, "Tri-state median filter for image denoising," *IEEE Trans. Image Processing*, vol. 8, no. 12, pp. 1834–1835, 1999.

[20] G. Bilbro, W. E. Snyder, S. Garnier, and J. W. Gault, "Mean field annealing: a formalism for constructing GNC-like algorithms," *IEEE Trans. Neural Networks*, vol. 3, no. 1, pp. 131–138, 1992.

[21] C. Bouman and M. Shapiro, "A multiscale random model for Bayesian image segmentation," *IEEE Trans. Image Processing*, vol. 3, no. 2, pp. 162–177, 1994.

[22] T. Hofmann, J. Puzicha, and J. M. Buhmann, "Unsupervised texture segmentation in a deterministic annealing framework," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 20, no. 8, pp. 803–818, 1998.

[23] M. Betke and N. C. Makris, "Fast object recognition in noisy images using simulated annealing," in *Proc. 5th International Conference on Computer Vision (ICCV '95)*, pp. 523–530, Cambridge, Mass, USA, June 1995.

[24] T. Mikami, A. Yamaguchi, and M. Wada, "Fast annealing schedule by landscape enhancing for function optimization," in *Proc. International Conference on Soft Computing*, pp. 605–610, Iizuka, Japan, October 2000.

[25] S. T. Acton, J. Ghosh, and A. C. Bovik, "Fast combinatorial optimization using generalized deterministic annealing," in *Science of Artificial Neural Networks II*, vol. 1966 of *Proceedings of SPIE*, pp. 402–413, Orlando, Fla, USA, April 1993.

[26] CMU/VASC Image Database, http://vasc.ri.cmu.edu/idb/html/motion/pingpong/.

Ikhlas Abdel-Qader is currently an Assistant Professor of electrical and computer engineering, and is the Founder and Director of the Information Technology and Image Analysis (ITIA) Center in the College of Engineering and Applied Sciences, Western Michigan University. She earned her Ph.D. degree in electrical and computer engineering from North Carolina State University, Raleigh, in 1992. Dr. Abdel-Qader's research expertise and interests include feature extraction, motion estimation, and image compression with applications in medical imaging, nondestructive testing, and intelligent transportation systems. Dr. Abdel-Qader is a Member of the Institute of Electrical and Electronic Engineers (IEEE); the IEEE Acoustics, Speech, and Signal Processing Society; and the IEEE Engineering in Medicine and Biology Society. She is also a Member of the Society of Women Engineers, a Member of the Honor Society of Phi Kappa Phi, and an Associate Member of the Sigma Xi Scientific Research Society. Dr. Abdel-Qader has been a registered professional engineer in the State of Minnesota since 1996.

Tomislav Bujanovic received his M.S. degree in electrical engineering from Western Michigan University, Kalamazoo, Michigan. Mr. Bujanovic's research areas and interests include motion estimation and image segmentation from time-varying images. Tomislav Bujanovic has been a registered professional engineer in Serbia and Montenegro (former Yugoslavia) and he is certified for dealing with anti-explosion defense of electrical appliances on overground places imperiled by explosive blends. Prior to this, he was a Senior Engineer for power capacitors (including design, research, and development) and their applications (including control design, construction, installation, and final examination) in former Yugoslavia. Tomislav Bujanovic is an author or coauthor of several research articles and reports in image processing and in power capacitors and its applications. Tomislav Bujanovic was a Member of Yugoslav Technical Committee, IEC Commission no. 31 (networks, distribution systems).