Motion Compensated Color Video Classification Using Markov Random Fields*

Zoltan Kato, Ting-Chuen Pong, John Chung-Mong Lee

Hong Kong University of Science and Technology, Computer Science Dept.,
Clear Water Bay, Kowloon, Hong Kong, Tel: +852 2358 7000 — Fax:+852 2358 1477,
email: `kato@cwi.nl`, `tcpong@cs.ust.hk`, `cmlee@cs.ust.hk`

Abstract. This paper deals with the classification of color video sequences using Markov Random Fields (MRF) taking into account motion information. The theoretical framework relies on Bayesian estimation associated with MRF modelization and combinatorial optimization (Simulated Annealing). In the MRF model, we use the CIE-**luv** color metric because it is close to human perception when computing color differences. In addition, intensity and chroma information is separated in this space. The sequence is regarded as a stack of frames and both intra- and inter-frame cliques are defined in the label field. Without motion compensation, an inter-frame clique would contain the corresponding pixel in the previous and next frame. In the motion compensated model, we add a displacement field and it is taken into account in inter-frame interactions. The displacement field is also a MRF but there are no inter-frame cliques. The Maximum A Posteriori (MAP) estimate of the label and displacement field is obtained through Simulated Annealing. Parameter estimation is also considered in the paper and results are shown on color video sequences using both the simple and motion compensated models.

1 Introduction

Image classification is an important early vision task where pixels with similar features are grouped into homogeneous regions. Many high level processing tasks (surface description, object recognition, for example) are based on such a preprocessed image. Using color information can considerably improve capabilities of image classification algorithms compared to purely intensity-based approaches. However, we need a good color space in order to use color information in the same way as humans perceive color differences. There are several metrics proposed for computer vision [5]. We use the CIE-**luv** [5] color space here because it separates luminance and chroma information and it is easy to compute color differences in this metric.

The visual motion derived from a sequence of time-varying images [11, 4] is also a valuable source of information. Basically, it can be used to detect motion in the scene but it is also possible to derive more detailed information such as

* This research was supported by Hong Kong Research Grants Council under grant *HKUST616/94E*

position, orientation of a visible surface or 3D reconstruction of a scene. Herein, we are interested in computing displacement vectors [11, 13, 4] in order to build a motion compensated Markov Random Field (MRF) image classification model.

When we have a sequence of color images, still image MRF models [2, 9, 12, 7] can be easily extended to take into account the information in the previous and next frames [13, 11] (see Section 2.1). Instead of a 2D neighborhood system, we can use a 3D one with inter-frame cliques. If the camera or the objects in the scene are not moving then this model yields good segmentations. In the case of moving objects, however, this static model can fail.

To overcome the problem caused by moving objects, we introduce a displacement field (DF) [13] in Section 2.2 in order to take into account motion information in the label field. For simplicity, the DF is defined over the same lattice as the label field by placing a new lattice between two neighboring frames. DF is a vector-valued MRF giving the displacement vector at each site between two frames in the sequence. The estimation of the DF is done in parallel with the label field and no external algorithm or initialization is needed. The energy function of the so-defined system is minimized by the Metropolis algorithm [10]. The result is the classification of the input frames *and* the displacement vectors between frames.

Usually, MRF-based segmentation methods suffer from a lack of parameter estimation. The majority of the proposed methods are supervised, which limits their practical use because a human intervention is needed to compute the model parameters. Herein, we are interested in completely *data driven* algorithms since in real-life applications, these parameters are usually unknown and one has to estimate them without human intervention. In Section 2.3, we consider parameter estimation of the proposed model. Finally, some results are presented in Section 3.

2 MRF model

In this section, we describe a spatio-temporal MRF model for color video classification. First, we define a model which uses only color information and then we extend our model to take into account motion information.

2.1 Color video sequence classification (Label Field)

Let us suppose that the observed images consist of three spectral component values (**luv**) at each pixel denoted by the vector $\mathbf{f}_s^t$, where $s \in \mathcal{S}$ is the spatial index and $t \in \mathcal{T}$ is the temporal index. We are looking for the labeling $\hat{\omega}$, which maximizes the a posteriori probability $P(\omega \mid \mathcal{F})$, that is the *maximum a posteriori* (MAP) estimate. Bayes theorem tells us that:

$$P(\omega \mid \mathcal{F}) = \frac{1}{P(\mathcal{F})} P(\mathcal{F} \mid \omega) P(\omega). \tag{1}$$

Actually $P(\mathcal{F})$ does not depend on the labeling ω and we make the assumption that:

$$P(\mathcal{F} \mid \omega) = \prod_{t \in \mathcal{T}} \prod_{s \in \mathcal{S}} P(\mathbf{f}_s^t \mid \omega_s^t). \tag{2}$$

It is then easy to see that the global labeling, which we are trying to find, is given by:

$$\hat{\omega} = \arg\max_{\omega \in \Omega} \prod_{t \in \mathcal{T}} \prod_{s \in \mathcal{S}} P(\mathbf{f}_s^t \mid \omega_s^t) \prod_{C \in \mathcal{C}_{\mathcal{S}}} \exp(-V_C(\omega_C)) \prod_{C \in \mathcal{C}_{\mathcal{T}}} \exp(-V_C(\omega_C)) \ , \tag{3}$$

where $\mathcal{C}_{\mathcal{S}}$ is the set of spatial (or intra-frame) cliques and $\mathcal{C}_{\mathcal{T}}$ is the set of temporal (or inter-frame) cliques. It is obvious from this expression that the a *posteriori* probability also derives from a MRF. The energies of cliques of order 1 directly reflect the probabilistic modeling of labels without context, which could be used for labeling the pixels independently. This item ties the resulting segmentation to the original input.

A natural assumption is that $P(\mathbf{f}_s^t \mid \omega_s^t)$ is Gaussian, the classes $\lambda \in \Lambda = \{0, 1, \ldots, L-1\}$ are represented by the mean vectors $\boldsymbol{\mu}_\lambda$ and the covariance matrices Σ_λ. It is then clear that

$$P(\mathbf{f}_s^t \mid \omega_s^t) = \frac{1}{\sqrt{(2\pi)^3 \mid \Sigma_{\omega_s^t} \mid}} \exp\left(-\frac{1}{2} (\mathbf{f}_s^t - \boldsymbol{\mu}_{\omega_s^t}) \Sigma_{\omega_s^t}^{-1} (\mathbf{f}_s^t - \boldsymbol{\mu}_{\omega_s^t})^T \right) . \tag{4}$$

We get the following energy function:

$$U(\omega, \mathcal{F}) = U_1(\omega, \mathcal{F}) + U_2(\omega) + U_3(\omega) \ , \tag{5}$$

$$U_1(\omega, \mathcal{F}) = \sum_{t \in \mathcal{T}} \sum_{s \in \mathcal{S}} \left(\ln(\sqrt{(2\pi)^3 \mid \Sigma_{\omega_s^t} \mid}) + \frac{1}{2} (\mathbf{f}_s - \boldsymbol{\mu}_{\omega_s^t}) \Sigma_{\omega_s^t}^{-1} (\mathbf{f}_s^t - \boldsymbol{\mu}_{\omega_s^t})^T \right) \tag{6}$$

$$U_2(\omega) = \sum_{C \in \mathcal{C}_{\mathcal{S}}} V_2(\omega_C) \tag{7}$$

$$\text{where } V_2(\omega_C) = V_{\{s,r\}}(\omega_s^t, \omega_r^t) = \begin{cases} 0 \text{ if } \omega_s^t = \omega_r^t \\ \beta \text{ if } \omega_s^t \neq \omega_r^t \end{cases} \tag{8}$$

$$U_3(\omega) = \sum_{C \in \mathcal{C}_{\mathcal{T}}} V_3(\omega_C) \tag{9}$$

$$\text{where } V_3(\omega_C) = V_{\{t,t+1\}}(\omega_s^t, \omega_s^{t+1}) = \begin{cases} 0 \text{ if } \omega_s^t = \omega_s^{t+1} \\ \gamma \text{ if } \omega_s^t \neq \omega_s^{t+1} \end{cases} \tag{10}$$

where $\beta > 0$ and $\gamma > 0$ are model parameters controlling the homogeneity of the regions and the importance of spatial and temporal interactions. As they increase, the resulting regions become more homogeneous.

2.2 Using motion information (Displacement Field)

To further elaborate our model, we introduce a new field called the *displacement field* (DF). In this way, we can take into account motion information when doing classification of a video sequence. The DF could be defined over a different lattice than the label field (one could use a lower resolution, for instance) but, for simplicity, we define it over the same lattice, placing a new lattice between each neighboring frames $(t, t+1)$. DF is a vector-valued field, $\boldsymbol{\phi}_s^t \in \Phi$ denotes the displacement vector at site s between frames t and $t+1$.

The energy function of the DF is defined as follows:

$$U^{DF} = U_1^{DF}(\omega, \boldsymbol{\phi}) + U_2^{DF}(\boldsymbol{\phi})$$

$$U_1^{DF}(\omega, \boldsymbol{\phi}) = \sum_{t \in \mathcal{T}} \sum_{s \in \mathcal{S}} V_1^{DF}(\omega_s^t, \boldsymbol{\phi}_s^t) \tag{11}$$

$$\text{where } V_1^{DF}(\omega_s^t, \boldsymbol{\phi}_s^t) = \begin{cases} 0 \text{ if } \omega_s^t = \omega_{s+\boldsymbol{\phi}_s^t}^{t+1} \\ \alpha \text{ if } \omega_s^t \neq \omega_{s+\boldsymbol{\phi}_s^t}^{t+1} \end{cases} \tag{12}$$

$$U_2^{DF}(\boldsymbol{\phi}) = \sum_{C \in \mathcal{C}_{DF}} V_2^{DF}(\boldsymbol{\phi}_C) \tag{13}$$

$$\text{where } V_2^{DF}(\boldsymbol{\phi}_C) = \sum_{r \in C} \|\boldsymbol{\phi}_s^t, \boldsymbol{\phi}_r^t\|_2. \tag{14}$$

Unlike conventional approaches, herein we use the label field ω instead of the color value in the first order potential (Equation (12)). The second order potential (Equation (14)) is a smoothing constraint favoring similar displacement vectors in neighboring sites. Note that we have only intra-frame cliques here. In our tests, we have used a first order neighborhood system.

For motion compensated classification, we have to take into account the DF in the energy function of the label field. For this purpose, we will redefine $V_3(\omega_C)$ (see Equation (10)) in the following way:

$$V_3'(\omega_C) = V_{\{t,t+1\}}'(\omega_s^t, \omega_s^{t+1}) = \begin{cases} 0 \text{ if } \omega_s^t = \omega_{s+\boldsymbol{\phi}_s^t}^{t+1} \\ \gamma \text{ if } \omega_s^t \neq \omega_{s+\boldsymbol{\phi}_s^t}^{t+1} \end{cases} \tag{15}$$

The energy function of the motion compensated model is then given by the following equation:

$$U(\omega, \boldsymbol{\phi}, \mathcal{F}) = U_1(\omega, \mathcal{F}) + U_2(\omega) + U_3'(\omega) + U_1^{DF}(\omega, \boldsymbol{\phi}) + U_2^{DF}(\boldsymbol{\phi}) \tag{16}$$

where $U_3'(\omega)$ is the motion compensated energy function of inter-level cliques (see Equation (10) and Equation (15)). The MAP estimate of the label and displacement field is obtained trough the minimization of $U(\omega, \boldsymbol{\phi}, \mathcal{F})$:

$$(\hat{\omega}, \hat{\boldsymbol{\phi}}) = \min_{\omega, \boldsymbol{\phi}} U(\omega, \boldsymbol{\phi}, \mathcal{F}). \tag{17}$$

Since the energy function has many local minima, we use the Metropolis algorithm [10] to find the global minima. At each iteration, the label field is updated first followed by the DF.

2.3 Parameter Estimation

Our goal is to propose a completely data-driven, unsupervised classification algorithm. Thus, we have to estimate the mean vector $\boldsymbol{\mu}_\lambda$ and the covariance matrix Σ_λ for each class, and the hyper-parameters α, β and γ. The mean vectors and covariance matrices can be obtained from the first frame using an unsupervised classification algorithm (for more details, see [7]). The hyper-parameters are less sensitive. We have found in practice that $\alpha = 15.0$, $\beta = 2.5$ and $\gamma = 2.0$ give good results. Of course, one could also use an estimation algorithm (see [3, 8, 6]) to obtain the right values depending on the input video sequence. However, we found that these algorithms need a huge computing power and the obtained values were very close to our *ad hoc* estimates. The mean vectors and covariance matrices could also be re-estimated during the classification using an adaptive classification algorithm similar to [7] but experiments show that the one frame estimates are good enough to obtain a reasonably good classification.

3 Experiments

The proposed algorithm has been tested on a variety of color video sequences. Herein, we present a few of our results obtained on a variety of color video sequences and also compare the motion compensated and static models. In all cases, the optimization algorithm has been stopped when the number of changed sites was less than 0.01% of the sites.

In Table 1, we give the computing times for the presented video sequences. One can see, that motion compensated classification needs more iterations and more computing time because of the additional displacement field. However, the quality of these results is also better. The computing time depends also on the optimization method. ICM [1] is a deterministic algorithm which converges in a few iterations but it finds only a local minima. This may not be as good as the one given by a stochastic method, like the Metropolis algorithm [10].

In Figure 1 and Figure 2, we compare the results obtained by the static and motion compensated model on two color video sequences. The results in the second (resp. third) column has been obtained by the static (resp. motion compensated) model. One can see that the results are better in the case of motion compensation.

The proposed model can be easily applied to gray-level images, only the first order clique-potentials have to be changed in Equation (6): Instead of a 3-variate Gaussian distribution, we use here a univariate one. In Figure 3, we show the results obtained on the "tennis" sequence using only gray-values. The result clearly shows that color information can improve considerably the final results. The computing time is only slightly lower than in the case of color images (see Table 1).

In Figure 4, we give the classification and displacements obtained on the "tennis" sequence using the Metropolis algorithm. The displacements are displayed over 16×16 blocks. The displacement field is noisy inside homogeneous regions but it is reasonably good over region boundaries. This is good enough for the purpose of motion compensated classification. More accurate displacement field could be obtained through a more elaborated homogeneity constraint in Equation (14).

4 Conclusion

We have proposed an unsupervised, motion compensated color video classification algorithm. The classification model is defined in a Markovian framework and uses a first order potential derived from a three-variate Gaussian distribution in order to tie the final classification to the observed images. The label field has spatio-temporal cliques and the displacement vectors are taken into account by inter-frame (or temporal) cliques. In the DF's energy function we use the label field instead of computing the color differences of corresponding pixels in order to reduce computing time. The energy function is minimized through a Metropolis algorithm [10] and we obtain the classification of the frames *and* the displacement vectors at the same time. The algorithm is unsupervised; only the number of classes is supplied by the user. The method has been tested on a variety of color video sequences and the results are encouraging.

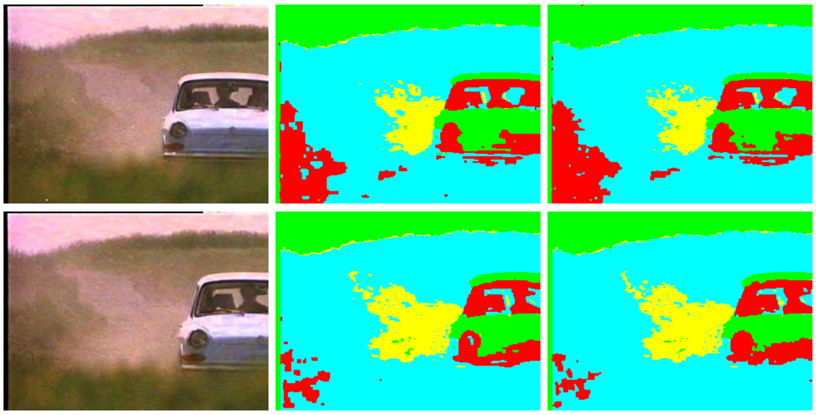
References

1. J. Besag. On the statistical analysis of dirty pictures. *Jl. Roy. Statis. Soc. B.*, 1986.
2. M. J. Daily. Color Image Segmentation Using Markov Random Fields. In *Proc. DARPA Image Understanding*, 1989.
3. D. Geman. Bayesian Image Analysis by Adaptive Annealing. In *Proc. IGARSS'85*, pages 269–277, Amherst, USA, Oct. 1985.
4. F. Heitz and P. Bouthemy. Multimodal Estimation of Discontinuous Optical Flow Using Markov Random Fields. *IEEE-PAMI*, 15(12):1217–1232, Dec. 1993.
5. A. K. Jain. *Fundamentals of Digital Image Processing.* Prentice Hall, 1989.
6. Z. Kato, M. Berthod, J. Zerubia, and W. Pieczynski. Unsupervised Adaptive Image Segmentation. In *ICASSP'95*, Detroit, USA, May 1995.
7. Z. Kato, T. C. Pong, and J. C. M. Lee. Motion Compensated Color Image Classification and Parameter Estimation in a Markovian Framework. Technical Report HKUST-CS97-04, The Hong Kong University of Science and Technology, July 1997.
8. S. Lakshmanan and H. Derin. Simultaneous Parameter Estimation and Segmentation of Gibbs Random Fields Using Simulated Annealing. *IEEE–PAMI*, 11(8):799–813, Aug. 1989.
9. J. Liu and Y. H. Yang. Multiresolution Color Image Segmentation. *IEEE-PAMI*, 16(7):689–700, July 1994.
10. N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller. Equation of state calculations by fast computing machines. *J. of Chem. Physics, Vol. 21, pp 1087-1092*, 1953.
11. D. W. Murray and B. F. Buxton. Scene Segmentation from Visual Motion Using Global Optimiziation. *IEEE-PAMI*, 9(2):220–228, Mar. 1987.

12. D. K. Panjwani and G. Healey. Markov Random Field Models for Unsupervised Segmentation of Textured Color Images. *IEEE - PAMI*, 17(10):939–954, Oct. 1995.
13. M. I. Sezan and R. L. Lagendijk, editors. *Motion Analysis and Image Sequence Processing*, chapter E. Dubois and J. Konrad: Estimation of 2-D Motion Fields from Image Sequences with Application to Motion-Compensated Processing, pages 53–88. Kluwer Academic Publishers, 1993.

Model	Method	Num. of iterations	CPU time
color "tennis" sequence (23 frames, 6 classes)			
Static	ICM	9	0.61 hours
Static	Metropolis	58	1.94 hours
Motion compensated	Metropolis	400	26.4 hours
gray-level "tennis" sequence (23 frames, 6 classes)			
Motion compensated	Metropolis	400	19.6 hours
color "car" sequence (21 frames, 4 classes)			
Static	ICM	12	0.53 hours
Static	Metropolis	67	2.02 hours
Motion compensated	Metropolis	400	21.8 hours

Table 1. Computing times on a SPARC station 1000.



Original frames Static Motion compensated

Fig. 1. Results obtained by the static and motion compensated model (21 frames, 4 classes) using the Metropolis algorithm.

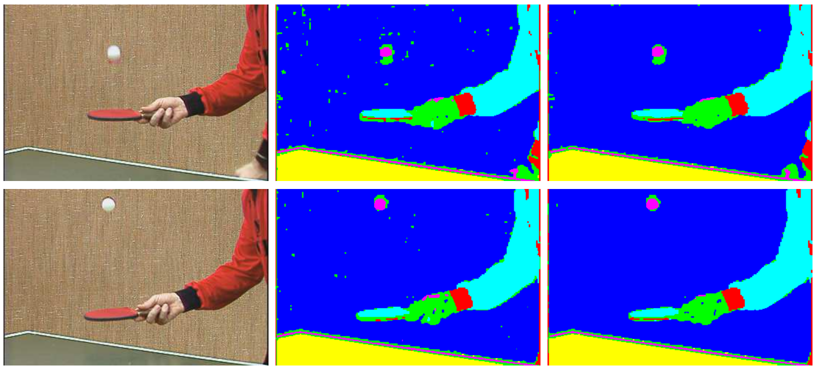


Fig. 2. Results obtained by the static and motion compensated model using the Metropolis algorithm on the “tennis” sequence (23 frames, 6 classes).

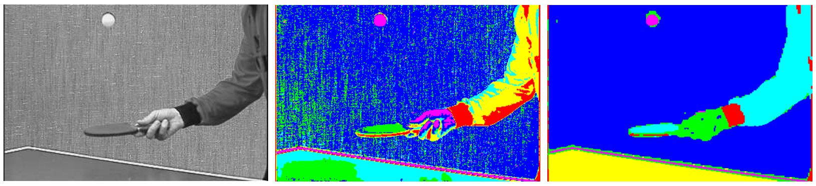


Fig. 3. Comparison of intensity- and color-based classification results on the “tennis” sequence (23 frames, 6 classes) using the motion compensated model with the Metropolis algorithm.

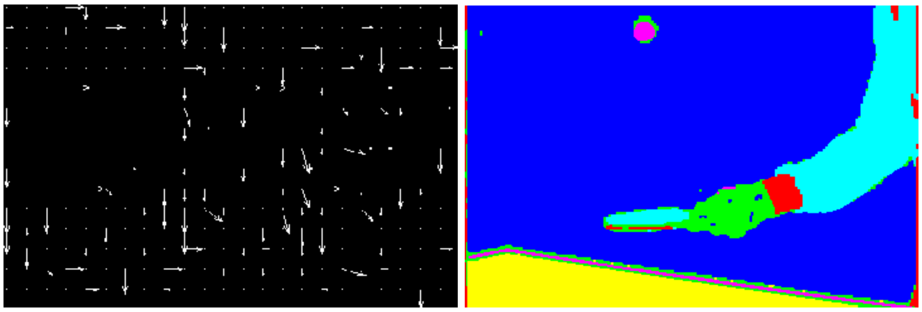


Fig. 4. Classification and displacements obtained on the “tennis” sequence (23 frames, 6 classes) using the Metropolis algorithm.