Detecting and Solving Template Ambiguities in Motion Segmentation

Pedro M. Q. Aguiar *

Instituto de Sistemas e Robótica
Instituto Superior Técnico
Lisboa, Portugal
E-mail : aguiar@isr.ist.utl.pt

José M. F. Moura

Electrical and Computer Engineering
Carnegie Mellon University
Pittsburgh, U.S.A.
E-mail : moura@ece.cmu.edu

Abstract

When the color or gray level of a moving object is very similar to that of the background, motion-based segmentation methods fail. This leads to ambiguous templates for the moving objects. In this paper, we propose a method that segments unambiguously from motion the templates of moving objects. Our method, which we call **Incremental Motion Segmentation** *integrates over time the small differences between the gray level of the moving object and that of the background. Our experiments with segmenting a robot soccer video clip show the quality of our results.*

1 Introduction

We introduced in [1] **Incremental Motion Segmentation (IMS)** and detailed its use in **Generative Video (GV)** [2] analysis. **IMS** achieves motion-based segmentation for general scenes by integrating over time the information content of an image sequence. **GV** reduces video sequences to world images and ancillary data. The world images are augmented views of the world - background world image - and complete views of moving objects - figure world images. The ancillary data registers the world images, stratifies them at each time instant, and positions the camera with respect to the layering of world images. A major task in **GV** analysis is the segmentation from motion of the (background and figures) world images.

When a moving object has color or gray-level that is very similar to the background, current motion-based segmentation methods that use only a few consecutive frames fail. To resolve these problems, approaches like in [3, 4] use statistical regularization techniques or combine motion with other attributes, like color or texture. In general, these methods lead to complex and time consuming algorithms. In contrast, the method we propose here integrates over time any existing small differences. Our results show that **IMS** has the ability to detect and solve unambiguously the templates for these low contrast objects. We present experiments with real data that construct the **GV** representation for a robot soccer video sequence.

The paper is organized as follows. Section 2 formulates the problem, leading to an ML-based cost-function. The minimization procedure is described in section 3. The **IMS** algorithm is described in section 4. Experimental results and conclusions are in sections 5 and 6.

2 Problem formulation

IMS is derived as a computationally simple approximation to *Maximum Likelihood* (ML) estimation.

Observation model. We motivate **IMS** by considering a single object moving in front of a background, the scene being captured by a moving camera. The image $\boldsymbol{I}_i$ is modeled as

$$\boldsymbol{I}_i = \left\{ \boldsymbol{B}(\boldsymbol{p}_i^{\#}) \left[1 - \boldsymbol{T}(\boldsymbol{q}_i^{\#}) \right] + \boldsymbol{O}(\boldsymbol{q}_i^{\#}) \boldsymbol{T}(\boldsymbol{q}_i^{\#}) + \boldsymbol{W}_i \right\} \boldsymbol{H} \quad (1)$$

where $\boldsymbol{B}$ and $\boldsymbol{O}$ are the background and object world images [2], $\boldsymbol{T}$ is the object template, $\boldsymbol{p}_i$ and $\boldsymbol{q}_i$ are the camera pose and the object position, and $\boldsymbol{W}_i$ is zero mean white Gauss noise. We assume that $\boldsymbol{I}_i(x, y) = 0$ for (x, y) outside the region observed by the camera. $\boldsymbol{H}$ is such that $\boldsymbol{H}(x, y) = 1$ for (x, y) in the region observed in the image and $\boldsymbol{H}(x, y) = 0$ otherwise. We denote by $\boldsymbol{I}(\boldsymbol{p})$ the registration of the image $\boldsymbol{I}$ according to the position vector $\boldsymbol{p}$. The pixel (x, y) of the registered image is denoted by $\boldsymbol{I}(\boldsymbol{p}; x, y)$. The registration of $\boldsymbol{I}(\boldsymbol{p})$ according to the position vector $\boldsymbol{q}$ is denoted by $\boldsymbol{I}(\boldsymbol{q}, \boldsymbol{p})$. We denote by $\boldsymbol{I}(\boldsymbol{p}^{\#})$ the image that registered according to $\boldsymbol{p}$ equals $\boldsymbol{I}$, so we have $\boldsymbol{I}(\boldsymbol{p}, \boldsymbol{p}^{\#}) = \boldsymbol{I}$.

Maximum likelihood estimate. The problem is to estimate from the N images $\boldsymbol{I}_1, \boldsymbol{I}_2, \ldots, \boldsymbol{I}_N$ the back-

*The work of the first author was partially supported by NATO.

ground world image $\boldsymbol{B}$, the object world image $\boldsymbol{O}$, the object template $\boldsymbol{T}$, the camera pose $\{\boldsymbol{p}_i\}$, and the object position $\{\boldsymbol{q}_i\}$. These quantities define the **GV** representation of the sequence. The ML estimate minimizes the functional J over all **GV** parameters

$$J = \sum_{x,y} \sum_{i=1}^{N} \left\{ \boldsymbol{I}_i - \boldsymbol{B}(\boldsymbol{p}_i^{\#}) \left[\boldsymbol{1} - \boldsymbol{T}(\boldsymbol{q}_i^{\#}) \right] - \boldsymbol{O}(\boldsymbol{q}_i^{\#}) \boldsymbol{T}(\boldsymbol{q}_i^{\#}) \right\}^2 \boldsymbol{H} \quad (2)$$

where the inner sum is over the full set of N images and the outer sum is over all pixels observed in each image. We omit the dependence on (x, y) to simplify the notation.

Expression (1) models explicitly the occlusion of the background by the moving object. The estimation of the parameters of (1) using N images rather than a single pair of images is a feature that distinguishes our work from other techniques which use only two or three consecutive frames.

The simultaneous minimization of the functional J over the estimation parameters $\boldsymbol{B}, \boldsymbol{O}, \boldsymbol{T}, \{\boldsymbol{p}_i\}$, and $\{\boldsymbol{q}_i\}$ is very complex. To simplify this task and motivated by our experience with real video sequences, we decouple the estimation of the motion of the camera and the motion of the moving object from the estimation of the remaining parameters. Any method that estimates motions of multiple objects can be used with our **IMS** framework. See reference [5] for a survey. We estimate motion vectors by an affine model. To cope with large displacements, we use a spatial multiresolution pyramid. To deal with multiple moving objects, we use a *quad-tree* decomposition.

3 Minimization Procedure

The position vectors $\{\boldsymbol{p}_i\}$ and $\{\boldsymbol{q}_i\}$ are estimated as explained in section 2, so that in this section we assume that they are known. The problem becomes the minimization of J, given by expression (2), with respect to the background world image $\boldsymbol{B}$, the world image $\boldsymbol{O}$ of the moving object, and its template $\boldsymbol{T}$.

We express the estimate $\hat{\boldsymbol{O}}$ of the object world image that minimizes J in terms of the template $\boldsymbol{T}$. Then we replace $\hat{\boldsymbol{O}}$ in expression (2), getting J in terms of $\boldsymbol{B}$ and $\boldsymbol{T}$. To solve for $\hat{\boldsymbol{B}}$ and $\hat{\boldsymbol{T}}$, we use a two-step iterative method. The steps involved are: i) solve for $\hat{\boldsymbol{B}}$ with fixed $\hat{\boldsymbol{T}}$, and ii) solve for $\hat{\boldsymbol{T}}$ with fixed $\hat{\boldsymbol{B}}$. We obtain closed-form solutions for both steps i) and ii) in contrast to the solution for the simultaneous minimization of J with respect to $\boldsymbol{B}$ and $\boldsymbol{T}$.

As it is usual with this kind of methods, although the value of J decreases along the iterative process, the convergence to the global minimum is not guaranteed, being necessary to provide a good initialization. We compute a first estimate of the moving object template by using the method described in [1]. It estimates $\hat{\boldsymbol{T}}$ as the union of the regions detected as being in movement over a set of frames.

Object / background world images. We consider now the estimation of the world image of the moving object and of the world image of the background, given the template $\boldsymbol{T}$.

The estimate $\hat{\boldsymbol{O}}$ of the object world image is found by setting $\nabla_{\boldsymbol{O}} J = \boldsymbol{0}$, with ∇ the gradient operator

$$\hat{\boldsymbol{O}} = \boldsymbol{T} \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{I}_i(\boldsymbol{q}_i) \quad (3)$$

This expression averages the observations $\boldsymbol{I}_i$ registered according to the motion $\boldsymbol{q}_i$ of the object in the region corresponding to the template $\boldsymbol{T}$ of the moving object.

The background world image estimate $\hat{\boldsymbol{B}}$ is found by minimizing J in (2), given the template $\boldsymbol{T}$. After algebraic manipulations, we obtain

$$\hat{\boldsymbol{B}} = \frac{\sum_{i=1}^{N} \left[\boldsymbol{1} - \boldsymbol{T}(\boldsymbol{p}_i, \boldsymbol{q}_i^{\#}) \right] \boldsymbol{I}_i(\boldsymbol{p}_i)}{\sum_{i=1}^{N} \left[\boldsymbol{1} - \boldsymbol{T}(\boldsymbol{p}_i, \boldsymbol{q}_i^{\#}) \right] \boldsymbol{H}(\boldsymbol{p}_i)} \quad (4)$$

The estimate averages the observations $\boldsymbol{I}_i$ registered with the background motion $\boldsymbol{p}_i$, in the regions $\{(x, y)\}$ not occluded by the moving object, i.e., when $\boldsymbol{T}(\boldsymbol{p}_i, \boldsymbol{q}_i^{\#}; x, y) = 0$. In the denominator, $\boldsymbol{H}(\boldsymbol{p}_i)$ weighs differently regions that, due to camera motion, are not seen in all images.

Template estimation. Replacing the object world image estimate $\hat{\boldsymbol{O}}$ given by (3) in (2), we express the functional J in terms of the background world image $\boldsymbol{B}$ and the object template $\boldsymbol{T}$. The complete derivation is given in [6]. We obtain

$$J = \sum_{x,y} \boldsymbol{T}(x, y) \boldsymbol{Q}(x, y) + \text{Constant} \quad (5)$$

$$\boldsymbol{Q}(x, y) = \boldsymbol{Q}^1(x, y) - \boldsymbol{Q}^2(x, y) \quad (6)$$

$$\boldsymbol{Q}^1(x, y) = \frac{1}{N} \sum_{i=2}^{N} \sum_{k=1}^{i-1} \left[\boldsymbol{I}_i(\boldsymbol{q}_i; x, y) - \boldsymbol{I}_k(\boldsymbol{q}_k; x, y) \right]^2 \quad (7)$$

$$\boldsymbol{Q}^2(x, y) = \sum_{i=1}^{N} \left[\boldsymbol{I}_i(\boldsymbol{q}_i; x, y) - \boldsymbol{B}(\boldsymbol{q}_i, \boldsymbol{p}_i^{\#}; x, y) \right]^2 \quad (8)$$

Consider the minimization of J given by (5) over the template $\boldsymbol{T}$, given the background world image $\boldsymbol{B}$. It is clear from (5), that the minimization of J with respect to each spatial location of $\boldsymbol{T}$ is independent

from the minimization over the other locations. The template $\hat{\boldsymbol{T}}$ that minimizes J is given by the following test evaluated at each pixel:

$$\boldsymbol{Q}^1(x,y) \underset{\hat{\boldsymbol{T}}(x,y)=1}{\overset{\hat{\boldsymbol{T}}(x,y)=0}{\gtrless}} \boldsymbol{Q}^2(x,y) \tag{9}$$

In the spatial locations where the differences between each frame $\boldsymbol{I}_i(\boldsymbol{q}_i)$ and the background $\boldsymbol{B}(\boldsymbol{q}_i, \boldsymbol{p}_i^{\#})$ are greater than the differences between each pair of co-registered frames $\boldsymbol{I}_n(\boldsymbol{q}_n)$ and $\boldsymbol{I}_k(\boldsymbol{q}_k)$, we estimate $\hat{\boldsymbol{T}}(x,y) = 1$, these regions belong to the moving object. If not, the regions belong to the background.

For regions with low contrast between the moving object and the background, the test may initially, for a small number of frames, be inconclusive ($\boldsymbol{Q}^1(x,y) \simeq \boldsymbol{Q}^2(x,y)$). As more frames are processed and small differences accumulate, the initial test ambiguity is resolved and the region is assigned to the moving object or to the background.

4 IMS Algorithm

After processing a number of images, the estimate $\hat{\boldsymbol{T}}$ does not change significantly as new images become available. To see this, we replace the image model given by (1) in (7) and (8), and compute the expected value $\mathrm{E}\{\boldsymbol{Q}\} = \mathrm{E}\{\boldsymbol{Q}_1 - \boldsymbol{Q}_2\}$ with respect to the observation noise. See [6] for the complete derivation. We obtain

$$\mathrm{E}\{\boldsymbol{Q}\} = [\boldsymbol{1} - \boldsymbol{T}] \frac{1}{N} \sum_{i=1}^{N} \sum_{k=1}^{i-1} \left[\boldsymbol{B}(\boldsymbol{q}_i, \boldsymbol{p}_i^{\#}) \boldsymbol{H}(\boldsymbol{q}_i) - \boldsymbol{B}(\boldsymbol{q}_k, \boldsymbol{p}_k^{\#}) \boldsymbol{H}(\boldsymbol{q}_k) \right]^2$$
$$- \boldsymbol{T} \sum_{i=1}^{N} \left[\boldsymbol{O} - \boldsymbol{B}(\boldsymbol{q}_i, \boldsymbol{p}_i^{\#}) \right]^2 + \text{residual independent of } \boldsymbol{T} \tag{10}$$

As we process more images, $\mathrm{E}\{\boldsymbol{Q}(x,y)\}$ becomes more negative in the spatial locations where $\boldsymbol{T}(x,y) = 1$, and more positive where $\boldsymbol{T}(x,y) = 0$ (note that the coefficients that multiply $\boldsymbol{T}$ and $[\boldsymbol{1} - \boldsymbol{T}]$ in (10) are non-negative). After processing a sufficiently large number of frames the sign of $\boldsymbol{Q}$ at each pixel (x,y) stabilizes, which in turn, through test (9), stabilizes the estimate $\hat{\boldsymbol{T}}$ of the template. These considerations motivate splitting the algorithm in two phases - template acquisition and recursive world image updating.

Template acquisition. This phase detects and solves the template ambiguities mentioned before. It implements the sequential method described in section 3 with an increasing larger set of frames. The background world image estimate $\hat{\boldsymbol{B}}$ is given by expression (4), replacing the previous estimate $\hat{\boldsymbol{T}}$. The template estimate $\hat{\boldsymbol{T}}$ is given by the test (9), replacing the previous estimate $\hat{\boldsymbol{B}}$. The template acquisition phase lasts until the test $\boldsymbol{Q}_n^1(x,y) \gtrless \boldsymbol{Q}_n^2(x,y)$ is conclusive at all spatial locations, leading to the template estimate $\hat{\boldsymbol{T}}(x,y)$.

Recursive world image generation. From expressions (3) and (4), we obtain the following recursive updates, for fixed $\boldsymbol{T} = \hat{\boldsymbol{T}}$:

$$\hat{\boldsymbol{O}}_n = \frac{n-1}{n} \hat{\boldsymbol{O}}_{n-1} + \hat{\boldsymbol{T}} \frac{1}{n} \boldsymbol{I}_n(\boldsymbol{q}_n) \tag{11}$$

$$\boldsymbol{S}_n^b = \boldsymbol{S}_{n-1}^b + \left[\boldsymbol{1} - \hat{\boldsymbol{T}}(\boldsymbol{p}_n, \boldsymbol{q}_n^{\#}) \right] \boldsymbol{H}(\boldsymbol{p}_n) \tag{12}$$

$$\hat{\boldsymbol{B}}_n = \frac{\boldsymbol{S}_{n-1}^b}{\boldsymbol{S}_n^b} \hat{\boldsymbol{B}}_{n-1} + \frac{[\boldsymbol{1} - \hat{\boldsymbol{T}}(\boldsymbol{p}_n, \boldsymbol{q}_n^{\#})]}{\boldsymbol{S}_n^b} \boldsymbol{I}_n(\boldsymbol{p}_n) \tag{13}$$

where we define the weights $\boldsymbol{S}_n^b$ as the denominator of (4), and the element $\boldsymbol{S}_n^b(x,y)$ is the number of times the pixel $\boldsymbol{B}(x,y)$ is observed in the first n frames.

5 Experimental Results

We test the algorithm with a sequence of 20 images from a robot soccer game, see [7], showing a robot pursuing a ball, see figure 1. Due to the white stripes on the field, the robot template is ambiguous during the first frames of the sequence. Only after the robot rotates, is it possible to determine its template.

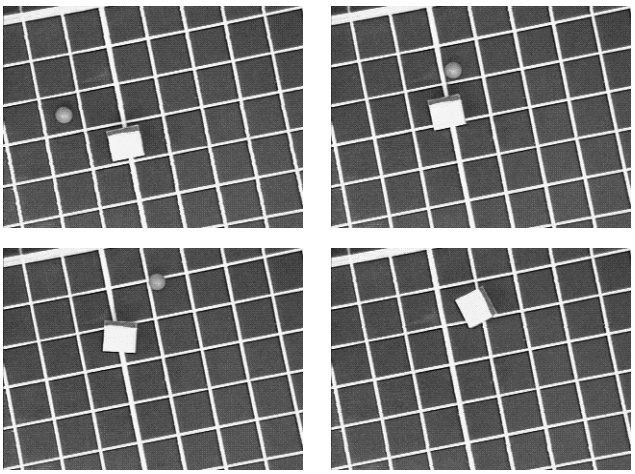

Figure 1: Image sequence. Frames $1, 4, 8, 16$.

The initialization phase takes 5 frames. It detects three motions, the static background, the moving ball, and the moving robot. According to test (9), the ball template is unambiguous after the 5 frames used in the initialization step. Figure 2 shows the evolution of the robot template. Regions where the test is inconclusive are grey, regions classified as being part of the robot template are white. The black regions are

classified as either background or as belonging to the ball template. The acquisition phase is ended after 10 frames. The final robot template estimate is shown on the right side of figure 2.

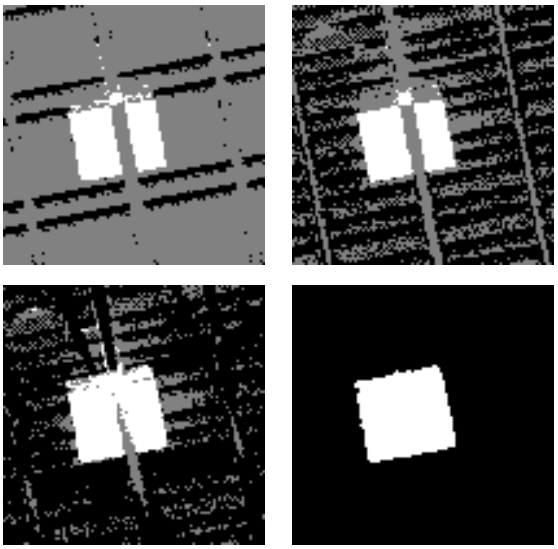

Figure 2: Robot template after frames $2, 4, 6, 10$.

Figure 3 illustrates the evolution of the matrix $\boldsymbol{Q}$. The curves on the left side represent the value of $\boldsymbol{Q}(x, y)$ for several pixels (x, y) that belong to the template of the robot. These curves start close to zero and decrease with the number of frames processed, as predicted by expression (10). The curves on the right side of figure 3 were obtained for several pixels that do not belong to the template of the robot. For these pixels, $\boldsymbol{Q}(x, y)$ increases with the number of frames.

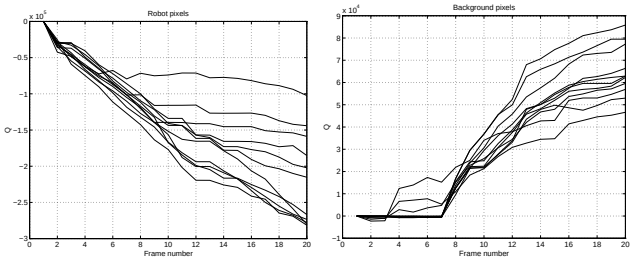

Figure 3: Evolution of $\boldsymbol{Q}$.

Figure 4 shows the world images for the two moving objects and background, after processing the entire sequence of 20 frames.

6 Conclusion

The **Incremental Motion Segmentation** algorithm described here achieves segmentation from motion of objects with low contrast. It is computationally simple, resolving the ambiguities that arise by integrating over time the information contents of the image sequence. The experimental results show that IMS is suitable for 2D object-based video analysis.

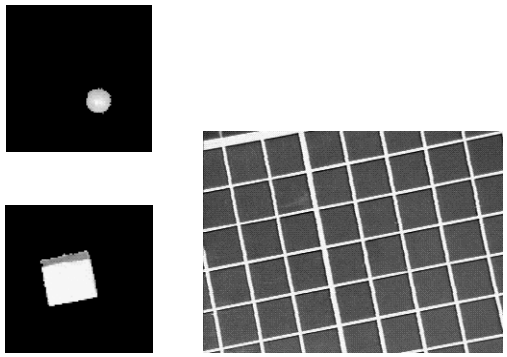

Figure 4: Ball, robot and background world images.

References

[1] P. M. Q. Aguiar and J. M. F. Moura. Incremental motion segmentation in low texture. In *IEEE Int. Conf. Image Processing*, volume I, pages 233–236, Switzerland, Sep. 1996.

[2] R. S. Jasinschi and J. M. F. Moura. Content-based video sequence representation. In *IEEE Int. Conf. Image Processing*, volume II, pages 229–232, U.S.A., Oct. 1995.

[3] P. Bouthemy and E. François. Motion segmentation and qualitative dynamic scene analysis from an image sequence. *Int. J. Computer Vision*, 10(2):157–182, 1993.

[4] M-P. Dubuisson and A. K. Jain. Contour extraction of moving objects in complex outdoor scenes. *Int. J. Computer Vision*, 14(1):83–105, 1995.

[5] S. Ayer. *Sequential and Competitive Methods for Estimation of Multiple Motions*. PhD thesis, École Polytechnique Fédérale de Lausanne, 1995.

[6] P. M. Q. Aguiar and J. M. F. Moura. Moving objects segmentation in low texture / low contrast video. To be submitted.

[7] M. Veloso, P. Stone, S. Achim, and M. Bowling. A layered approach for an autonomous robotic soccer system. In *1st Int. Conf. Autonomous Agents*, U.S.A., Feb. 1997.