Motion Segmentation by quasi-simultaneous multi-parametric Estimation

R. Montoliu and F. Pla

Dept. Lenguajes y Sistemas Informáticos
Jaume I Univerisity
Campus Riu Sec s/n 12071 Castellón, Spain
email: `[montoliu,pla]@uji.es`, `http://www.vision.uji.es`

Abstract

This paper presents a framework for the motion segmentation and estimation task on sequences of two grey images without a priori information of the number of moving regions present in the sequence. The proposed algorithm combines temporal information, by using an accurate Generalized Least-Squares Motion Estimation process and spatial information by using an inlier/outlier classification process which classifies regions of pixels, in a first step, and the pixels directly, in a second step, into the different motion models present in the sequence. The performance of the algorithm has been tested on synthetic and real images with multiple objects undergoing different types of motion.

1. Introduction

Segmentation of moving objects in a video sequence is basic task for several applications of computer vision, e.g. a video monitoring system, intelligent-highway system, tracking, airport safety, surveillance tasks and so on. In this paper, Motion Segmentation, also called spatial-temporal segmentation, refers to labelling pixels which are associated with different coherently moving objects or regions in a sequence of two images. Motion Estimation refers to assigning a motion vector to each region (or pixel) in an image.

Although the Motion Segmentation and Estimation problem can be formulated in many different ways ([**?**], [**?**], [**?**], [**?**]), we choose to approach this problem as a multi-structural parametric fitting problem. In this context, the segmentation problem is similar to robust statistical regression. The main difference is that robust statistical regression usually involves statistics for data having one target distribution and corrupted with random outliers. Motion segmentation problems usually have more than one population with distinct distributions and not necessarily with a population having absolute majority.

The problem of fitting an a priori known model to a set of noisy data (with random outliers) was studied in the statistical community for a number of decades. One important contribution was the Least Median of Squares (LMedS) robust estimator but it has the break down point of 50%. This means that LMedS technique needs the population recovered to have at least a majority of 50% (plus 1). Other robust estimators have been developed in order to overcome this problem, which is frequently encountered in different computer vision tasks. They are Adaptive Least k-th Order residual (ALKS) [**?**] and Minimum Unbiased Scale Estimator (MUSE) [**?**]. These techniques minimize the k-th order statistic of the square residuals where the optimum value for the k is determined from the data. The problem of both techniques is the estimation of the correct value of k suffers high computation effort. Bab-Hadiashar and Suter presented a method named Selective Statistical Estimator (SSE) [**?**] which is a variation of the Least K-th order statistic data regression where the user proposes the value k as the lower limit of the size populations one is interested in. All the Motion Segmentation LKS-based algorithms start selecting an initial model using random sampling, and classifying all the pixels into this model using a scale measure. With the remaining pixels the process is repeated until all the pixel have been classified. The main problem of these algorithms is that there are frequently pixels that can be more suitable to belong to a model but they have been classified in an earlier model.

Danuser and Stricker [**?**] presented a similar framework for parametric model fitting. Their algorithm has a fitting step that is one component of the algorithm which also collect model inliers, detects data outliers and determines the a priori unknown total number of meaningful models in the data. They apply a quasi simultaneous application of a general Least Squares fitting while classifying observations in the different parametric data models. They applied their algorithm to multiple lines and planes fitting tasks. The most important advantages with respect to LKS-based algorithms are the use of an exchange step, that permits change of observations among models, and the use of a inliers/outliers classification process, which increases the accuracy of the segmentation.

In [**?**] a quasi-simultaneous motion segmentation and estimation method based on a parametric model fitting algorithm was presented. The method accurately estimates the affine motion parameters using a generalized least squares fitting process. It also classifies the pixels into the motion models present in two consecutive frames. This algorithm uses each pixel of the image as observation. It suffers from problems of isolated points because it does not use neighbourhood information and need given good initial models to obtain the final motion segmentation. Nevertheless, it indicates that the quasi-simultaneous application of the inliers/outliers classification algorithm and the accurate motion estimator can be useful to be applied in Motion Segmentation tasks.

This paper presents a Motion Segmentation and Estimation algorithm that, in a first step uses regions of pixels as observations in order to obtain good initial models that in a second step will be improved using each pixel as observation. The use of regions in the first step makes the segmentation more spatial consistent. In addition, the algorithm uses neighbourhood constraints to collect new inliers to the model, only regions (or pixels) that are neighbour of the model are considered to be inliers. This algorithm overcomes the need of a good enough previous segmentation of the models (they are obtained in the first step) and allows extracting the models without a priori information of the number of moving regions present in the sequence.

The rest of the paper is organized as follows: Section 2 explains the complete Motion Segmentation and Estimation algorithm. Section 3 presents a set of experiments in order to verify the results obtained with our approach. Finally, some conclusions drawn from this work are described.

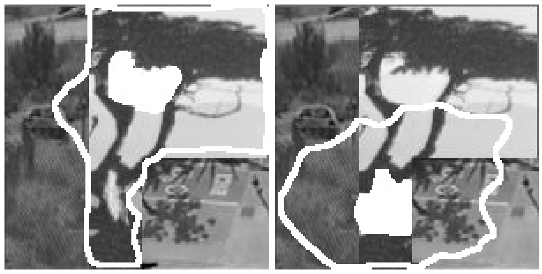
2 Algorithm Outline

In this paper we use the term **Model** as a structure with two elements, the first is a parametric motion vector and the second is a list of regions of the image that support the parametric motion vector. We refer as **Region** to a set of pixels with grey-level coherence.

The input of the algorithm are two consecutive images of a sequence, the first one I_1 captured at time t and the second one I_2 captured at time $t+1$. The output of the algorithm are a motion-based segmentated image I_s and a list of motion parameters corresponding at each region in I_s. For the sake of clarity, we describe the first part of the proposed algorithm in 6 steps:

1. **Preliminaries**: In this step, I_2 is segmented using a given grey level segmentation algorithm. The regions obtained are used as input of the algorithm. An adjacency graph of the previous segmentation is created. In addition the spatial derivates of the images I_1 and I_2 are estimated.

   The purpose of the grey-level segmentation process is to classify the pixels into regions. Our Motion Segmentation algorithm requires that each segmented region should not have pixels belonging to more than one final motion models. Any grey level segmentation algorithm that wherever possible tries to fulfil the previous constraint can be used.



(a) (b)

Figure 1: Two examples of initial models

2. **Get Initial Model**: The aim of this process is find the best possible start point to the global Motion Segmentation and Estimation algorithm. A good initial model is make up of a set of regions that have a high likelihood to belong to the same model. The process starts selecting a region randomly. A model with this region and its neighbours is formed. The motion is estimated for this model using the process in subsection 2.1. A goodness measure GM is calculated for this model. The previous step is repeated q times. The model with the best goodness measure is selected as the initial model.

   The goodness measure is calculated using the following expression: $GM = ((1-l_{avg})*2+(l_{best}-l_{worst}))$ where l_{avg} is the average of the likelihood $L_{M_n}(R)$ for each region R using the motion model M_n (see point 3), l_{best} is the highest likelihood of the regions and l_{worst} is the lowest likelihood of the regions. Therefore, the best initial model is the one which has the less GM.

   Figure 1 shows an illustrative example of two possible initial models for a sequence with three different motion models: static (left part of the image) and two translational motion (the part of the image showing a tree and the bottom right part). The pixels belonging to the region that have been selected randomly have been painted using white color. The limits of the model made up with the previous region and its possible neighbours are drawn with a continuous white line. Note that in the left image the majority of the pixels perform the same motion (the model of the three) and only a small area performs a different motion. Therefore, its GM will have a very small value. In addition, its GM will be lower than in the case of the right image where there is not a majority of pixels performing the same motion.

3. **Improve the model**: An iterative classification process is started in order to find the inliers and to reject outliers between the k regions that make up the initial model. With the set of resulting regions, we start another classification process with the neighbours of the last inserted regions not yet processed. This process continues until there are not more new neighbour regions to be processed.

   The loop of the inliers/outliers classification consists of:

   (a) Estimate the motion parameters using all the pixels belonging the regions of the model (see subsection 2.1).

   (b) Look for outliers into the regions of the model, if there are outliers, improve the motion parameters. A region R is considered outlier (with respect to model M_n) if the likelihood of region R belonging to a model M_n is lower than a threshold.

   (c) Test each outlier if it can be now considered inlier according the new estimated parameters. If there are new inliers, the parameters are improved again. A region R is considered inlier (with respect to model M_n) if the likelihood of the region R belonging to a model M_n is higher than a threshold.

   (d) Go to step b and repeat until there are not changes in the set of regions of the model.

   In order to estimate a likelihood of a region R belonging to a model M_n, the next expressions are used:

   $$L_{M_n}(R) = (\sum_{p_i \in R} L_{M_n}(p_i))/N_R$$
   $$L_{M_n}(p_i) = e^{-0.5 * \frac{F^2_{M_n}(p_i)}{\sigma_2}} \quad (1)$$

   where N_R is the number of pixels of the region R. For each pixel p_i belonging to the region R the likelihood $L_{M_n}(p_i)$ of the pixel belonging to a model M_n is calculated. This likelihood ([**?**]) has been modelled as a Gaussian like function where $F_{M_n}(p_i)$ is the residual for the pixel p_i of the objective function using the motion parametric vector of the model M_m.

4. **Exchange of regions**: If a valid model M_n has been extracted, then a region exchange procedure is started. The goal of this procedure is to reclassify regions that have been captured by an early model M_m where $m < n$. A region is moved if it lies closer to the new extracted model and there is a neighbour relationship between the region and the new model. If all the regions of the model M_m lie closer to the new Model M_n then the model M_m is deleted. When for each region of model M_m we can not decide if it lies closer to the model M_m or to the model M_n, then the models are merged, that is, it is considered both models have similar motion parameters.

5. **Repeat**: Go to step 2 and repeat the same process with another initial model if any. If there is any problem estimating the motion of some model, e.g. not enough texture information, not enough number of observations, etc., the regions of this model are moved to a set called *regions with problems* (RWP).

6. **End**: When all possible models have been extracted, the models that only have one region are tested in order to try to merge them with their neighbour models. In addition, each region in the RWP set is tested in order to move it into some of the models in its neighbourhood.

At the end of the first part of the algorithm, a set of NM motion models have been extracted. Each motion model is made up of a vector of parametric motion models and a set of regions which support the motion. Our Motion Segmentation algorithm requires that each region from the given grey-level segmentation should not have pixels belonging to more than one final motion model. It is very likely that some regions will not fulfill this constraint. The second part of the algorithm is performed in order to improve motion segmentation in these regions. In this step, instead of using a region of pixels as observation, each pixel is considered as observation. This process consists of:

1. **Find Outliers**: For each extracted model M_n ($n = 1 \ldots NM$), find all the pixels that can be considered as outliers. They are the pixels p_i which their likelihood respect to the model M_n, $L_{M_n}(p_i)$ is less than a threshold. All the outlier pixels are included in a set, together with the pixels belonging to the region which have been considered outliers in the previous part.

2. **Improve parameters**: The motion parameters for the motion models that have new outliers are improved (see subsection 2.1).

3. **Find Inliers**: For each outlier, test if it can be included in some of the motion models. A pixel p_i will be included in the model with the greatest likelihood $L_{M_n}(p_i)$, $n = 1 \dots NM$, if it is bigger than a threshold and there is a neighbourhood relationship between the pixel p_i and the model M_n.

4. **Improve parameters**: The motion parameters for the motion models that have new inliers are improved (see subsection 2.1).

5. **Repeat**: Repeat 1 to 4 while there are changes in the set of pixels.

At the end of the two parts of the algorithm the pixels have been classified into the different motion models corresponding to the moving objects in the scene. The pixels that could not be included in any model will be considered as outliers.

2.1 Motion Estimation

The Generalized Least Squares (GLS) algorithm is used in order to obtain the motion parameters of a model. The GLS algorithm [**?**] is based on minimizing an objective function O over a set S of r observation vectors, $S = \{L_1, \dots, L_r\}$.

$$O = \sum_{L_i \in S} (F_i(\chi, L_i))^2 \qquad (2)$$

where $\chi = (\chi^1, \dots, \chi^p)$ is a vector of p motion parameters and L_i is a vector of n observation $L_i = (L_i^1, \dots, L_i^n)$, $i = 1 \dots r$.

The equation (2) is non-linear, but it can be linearized using the Taylor expansion and neglecting higher order terms. This implies that an iterative solution has to be found. At each iteration, the algorithm estimates $\Delta\chi$, that improves the parameters as follows: $\chi_{t+1} = \chi_t + \Delta\chi$. The increment $\Delta\chi$ is calculated (see [**?**]) using the following expressions:

$$\begin{aligned}
\Delta X &= \left(A^T(BB^T)^{-1}A\right)^{-1} A^T(BB^T)^{-1}W \\
B &= \begin{pmatrix} B_1 & 0 & 0 & 0 \\ 0 & B_2 & 0 & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & B_r \end{pmatrix}_{(r \times (r \times n))} \\
A &= \begin{pmatrix} A_1 \\ A_2 \\ \dots \\ A_r \end{pmatrix}_{(r \times p)} \qquad W = \begin{pmatrix} w_1 \\ w_2 \\ \dots \\ w_r \end{pmatrix}_{(r \times 1)} \\
B_i &= \left(\frac{\partial F_i(\chi_t, L_i)}{\partial L_i^1}, \frac{\partial F_i(\chi_t, L_i)}{\partial L_i^2}, \dots, \frac{\partial F_i(\chi_t, L_i)}{\partial L_i^n}\right)_{(1 \times n)} \\
A_i &= \left(\frac{\partial F_i(\chi_t, L_i)}{\partial \chi^1}, \frac{\partial F_i(\chi_t, L_i)}{\partial \chi^2}, \dots, \frac{\partial F_i(\chi_t, L_i)}{\partial \chi^p}\right)_{(1 \times p)} \\
w_i &= -F_i(\chi_t, L_i)
\end{aligned} \qquad (3)$$

In motion estimation problems ([**?**]) the objective function is based on the assumption that the grey level of all the pixels of a region remains constant between two consecutive images. The motion parameters vector, χ, depends on the motion model being used. For each point i, the vector of observation L_i has three elements: column, row and grey level of second image at these coordinates. The objective function is expressed as follows:

$$O = \sum_{L_i \in S} (F_i(\chi, L_i))^2 = \sum_{L_i \in S} (I_1(x_i', y_i') - I_2(x_i, y_i))^2 \qquad (4)$$

where $I_1(x_i', y_i')$ is the grey level of the first image in the sequence at the transformed point x_i', y_i', and $I_2(x_i, y_i)$ are the grey level of the second image in the sequence at point x_i, y_i. Here, $L_i = (x_i, y_i, I_2(x_i, y_i))$.

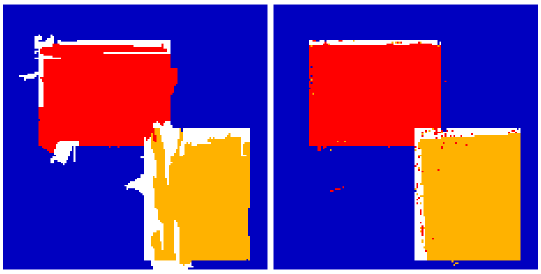
The affine motion model is used in this work, which is able to cope with translations, scaling, rotation and shear of images and is defined with a vector of $\chi = (a_1, b_1, c_1, a_2, b_2, c_2)$.

3 Experimental Results

In order to show the performance of the approach presented, two types of experiments have been carry out. In the first experiment, synthetic sequences have been used, where the final motion segmentation and the motion parameters of each model are known. In the second experiment real scenes are used, where the final motion segmentation and the motion parameters are unknown.

Figures 2(a,b) show both images of an example of synthetic sequence. In this synthetic sequence three different

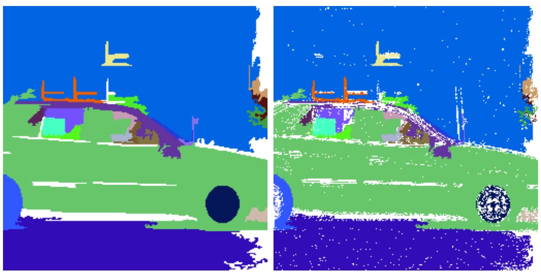
(a) 1st image (b) 2nd image



(c) Initial gray segmentation (d) Final segmentation

Figure 2: Both images of the synthetic sequence and results

(a) 1st image (b) 2nd image



(c) Initial gray segmentation (d) Final segmentation

Figure 3: Both images of the real sequence and results

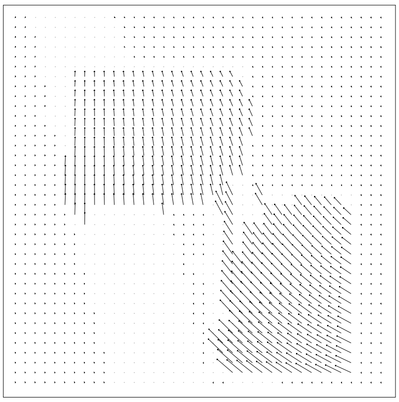


Figure 4: Optic Flow computed from results of the synthetic sequence

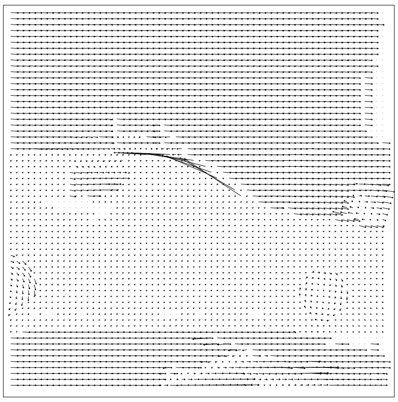


Figure 5: Optic Flow computed from results of the real sequence

motion models can be found. The first one is the background, which performs a null-motion. The second motion model performs a change of scale and the third one corresponds to a rotational motion. Figure 3(a,b) show both images of an example of real sequence.

Figures 2(c,d) and 3(c,d) show the result after the first step of the algorithm and the final results for both sequences. The white pixels in figures 2c and 3c, are the ones that have not been classified in any model. These regions correspond mainly to regions belonging to occluded areas due to the motion and to regions that do not fulfill the requirement of belonging only to a model, i.e. some pixels belong to a model and some other belong to a different model.

Figures 2d and 3d show the segmentation performed after the second step showing how segmentation has been improved in previous regions. Now, white pixels are the ones considered as outliers. They are mainly pixels belonging to occluded areas due to the motion and pixels where our algorithm could not estimate the motion due to lack of texture or to the presence of too large motions.

Figures 4 and 5 show the optic flow for both sequences. They have been computed using the motion parameters of each model in all the pixel belonging to them. They are presented in order to illustrate the motion models estimated.

4 Conclusions

In this paper, a motion segmentation and estimation algorithm has been presented, which can extract different moving regions present in the scene quasi-simultaneously and without a priori information of the number of moving objects. The main properties of our approach are:

- A GLS Motion Estimation is used, which produces accurate estimation of the motion parameters.
- A classification process which collects inliers, rejects outliers and exchanges regions among models allows to improve motion segmentation.
- It uses, in the first step, regions of pixels and neighbourhood information, that improves the spatial consistency and provides a good initial point to start the second step of the algorithm, which using pixels as observations improves the segmentation in the regions.
- The pixels considered as outliers are mainly pixels belonging to occluded areas due to the motion, thus, detection of outliers provides valuable information about occluded areas.

Future work must study hierarchical techniques in order to improve the speed of the algorithm and to cope with larger motion. The possibility of using sequences with more than two images will be also studied.