Proceedings of International Joint Conference on Neural Networks, Orlando, Florida, USA, August 12-17, 2007

Fuzzy weighted support vector regression for multiple linear model estimation : application to object tracking in image sequences

Franck Dufrenois, Denis Hamad

***Abstract*— In this paper, we present a new Support Vector Regression (SVR) based strategy for simultaneously extracting multiple linear structures in a training data set. As in Fuzzy c-prototypes algorithms [17], [18], [10], we introduce fuzzy weights in the SVR formulation which assign to each data point a membership value according to c-structures. We propose to solve the corresponding dual problem under an iterative strategy with an initialization step. Experiments show the benefits of robustness properties of SVR in comparison with the standard Fuzzy c-prototypes algorithm. Next, the motion estimation problem is used to illustrate its applicability and relevance in respect of real-world applications.**

I. Introduction

In scientific data analysis, it is frequent to observe the presence of multiple structures and outliers in the data set. Often, a parametric modelization of these structures is needed to easily extract them from the data set. For example, focusing on problems of computer vision and pattern recognition, straigth lines and planes are of utmost importance because they efficiently describe large parts of manmade objects. Two current examples are the extraction of step and crease surfaces in depth measurements [9], [25] or the extraction of multiple moving objects under an affine displacement assumption in video sequences [4]. In scientific data analysis, they are of similar relevance because they model linear phenomena. The problem of extracting these structures can be formally stated as a multiple estimation problem where the training data can be described by several identical/different models and the data partition is not known a priori. Generally, this problem is tackled following two ways :

- The first family of methods assumes the existence of a dominant structure or model in the data set, i.e. a structure that describes the majority of the data samples [5], [25], [3]. The other structures are considered as residual structures. In this case, the extraction of the different structures results of a general methodology based on successive applications of a robust regression algorithm [22], [15] so that during each (successive) iteration, a single dominant model is estimated and next a data partition step allows to remove the data belonging to it.
- The second family assumes the presence of several structures and the goal is to simultaneously extract them from a single data set. These methods use different learning strategies that combine both data clustering and model estimation.

Traditional clustering algorithms can be classified into two main categories : hierachical and partitional. Hierarchical clustering are collectively known as multiple learner system [13] (classification and regression trees (CART), mixture of experts (MoE), adaptive regression splines (MARS), etc.). Partitional clustering methods include crisp clustering, Fuzzy clustering [17], Gaussian mixture decomposition [6],etc.. The main difference between these two approaches is the way to analyse the data set. In Hierarchical clustering, the data set is iteratively analysed over a only local neighbourhood while, in partitional clustering, the data set is considered as a whole and the partition is based on the use of a membership value of a data according to a given cluster.

In this paper, we propose a new robust partitional clustering algorithm based on support vector regression. Although SVR is initially based on single model formulation, we show here the ability of the standard approach to extract simultaneously multiple linear structures from a degraded data set. In this first version, for simplicity, we assume that the number of clusters or structures is known. Of course, this question may be solved via a process of competitive agglomeration as in [11]. We modify the standard SVR formulation by introducing both Fuzzy weights in the SVR penalization term and an additional Fuzzy constraint. Then, the influence of a datum in regression is weigthed according to its relative importance with a given structure in the data set. This concept has been already introduced in SVM based classification problem [26], [19] and in least squares support vector regression [23], [16]. This methodology called "Fuzzy Weighted SVR" (FW-SVR) is similar to linear c-prototypes clustering algorithms [17], [18], [10] which are derived from the Fuzzy c-Means algorithm (FCM) [2]. However, we show that the FW-SVR algorithm is much more robust and less sensitive to initialization than the standard Fuzzy c-prototypes algorithm and can be an interesting alternative to other robust Fuzzy clustering algorithms [12], [20].

The rest of the paper is organized as follows. Section 2 briefly introduces standard SVR principles. Section 3 presents the FW-SVR. Next, the performances of the proposed approach is also studied and compared with the standard Fuzzy c-prototypes algorithm. Next, the proposed approach is applied for rigid object tracking in image sequence. The paper closes with a conclusion.

II. Linear Support Vector Regression

Consider a training data set $S = \{(\mathbf{x_1}, y_1), (\mathbf{x_2}, y_2), ..., (\mathbf{x_n}, y_n)\}$, where $\mathbf{x}_i \in R^d$ denotes an

F. Dufrenoi and D. Hamad are with the **L**aboratoire d'**A**nalyse des **S**ystèmes du **L**ittoral, Université du Littoral Côte d'Opale, 62228 Calais, France (phone: 33.03.21.46.56.57; fax: 03.21.46.06.86; email: Franck.Dufrenois/Johan.Colliez/Denis.Hamad@lasl.univ-littoral.fr).

input vector and $y_i \in R$ its corresponding target value. The generic SVR builds a linear function : $f(\mathbf{x}) = (\mathbf{w}, \mathbf{x}) + b$, such as the regression vector $\mathbf{w} \in R^d$ and the bias term $b \in R$ are the solutions of the following convex optimization problem [24], [1]:

$$\underset{\mathbf{w},\xi_i,\xi_i^*}{min} \left(\frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) \right) \tag{1}$$

subject to:

$$\begin{cases} y_i - (\mathbf{w}, \mathbf{x_i}) \leq \varepsilon + \xi_i \\ -y_i + (\mathbf{w}, \mathbf{x_i}) \leq \varepsilon + \xi_i^* \qquad \forall\ i\ \epsilon\ 1, ..., n \\ \xi_i, \xi_i^* \geq 0 \end{cases}$$

In (1), the parameter ε controls the "flatness" of the solution and so is related to the accuracy of the approximation. The presence of *errors* in the data set is measured by other internal parameters ξ_i and ξ_i^* called "*slack variables*", which characterize the deviation of training samples outside the ε-margin. The control of the global deviation is managed by the parameter C in (1). It turns out that this optimization problem can be solved more easily in its dual formulation:

$$\begin{array}{ll} \underset{\alpha_i,\alpha_i^*}{max} & -\frac{1}{2}\sum_{i,j=1}^{n}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*).(\mathbf{x_i}, \mathbf{x_j}) \\ & -\varepsilon\sum_{i,j=1}^{n}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{n} y_i(\alpha_i - \alpha_i^*) \end{array} \tag{2}$$

subject to: $\sum_{i=1}^{n}(\alpha_i^* - \alpha_i) = 0$ and $\alpha_i^*, \alpha_i \in [0, C]$.

Where the dual variables α_i and α_i^* are determined by Quadratic Programming techniques [1]. Then, the vector solution $\widehat{\mathbf{w}}$ and the estimated function $\widehat{f}$ are obtained from the following expressions:

$$\widehat{\mathbf{w}} = \sum_{i=1}^{n}(\alpha_i - \alpha_i^*)\mathbf{x_i} \tag{3}$$

$$\widehat{f}(x) = \sum_{i=1}^{n}(\alpha_i - \alpha_i^*)(\mathbf{x_i}, \mathbf{x}) + \mathbf{b} \tag{4}$$

In the next section, we introduce a Fuzzy weighted version of SVR to extract simultaneously several structures in data sets.

III. FW-SVR FOR SIMULTANEOUSLY EXTRACTING MULTIPLE LINEAR STRUCTURES

Given a training data set $S = \{(\mathbf{x_1}, y_1), (\mathbf{x_2}, y_2), ..., (\mathbf{x_n}, y_n)\}$ where $\mathbf{x}_i \in R^d$ and $y_i \in R$, we assume the existence of c structures $s_1, s_2, .., s_c$ in S. Our goal is to simultaneously estimate a regression vector $\mathbf{w}_k^{(t)}$ for each structure. To solve this problem, we introduce Fuzzy weights $\chi_{ki}^{(t)} \in [0, 1]$ that verify the following properties :

$$\textstyle\sum_{k=1}^{c} \chi_{ki} = 1 \quad \forall i \in [1, .., n] \tag{5}$$

$$0 < \textstyle\sum_{i=1}^{n} \chi_{ki} < n \quad \forall k \in [1, .., c] \tag{6}$$

$\chi_{ki}^{(t)}$ denotes the membership value of a data vector i with a structure s_k. The primal optimization problem of Fuzzy weighted SVR is formulated as :

$$\underset{\mathbf{w_i},,\xi_i,\xi_i^*}{min} \left(\frac{1}{2}\sum_{k=1}^{c}\|\mathbf{w}_k\|^2 + C\sum_{k=1}^{c}\sum_{i=1}^{n}\chi_{ki}^{m}(\xi_{ki} + \xi_{ki}^*) \right)$$

$$sc : \begin{cases} r_{ki} = y_i - (\mathbf{w}_k, \mathbf{x_i}) - b_k \leq \varepsilon + \xi_{ki} \\ -r_{ki} = -y_i + (\mathbf{w}_k, \mathbf{x_i}) + b_k \leq \varepsilon + \xi_{ki}^* \\ \xi_{ki}, \xi_{ki}^* \geq 0 \\ \sum_{k=1}^{c} \chi_{ki} = 1 \end{cases} \tag{7}$$

here m >1 is a weighting exponent on each Fuzzy membership and determines the amount of fuzziness of the resulting partition. We notice, as in FCM, that the clustering is all the more fuzzy since m is high. When the value of m tends towards 1, we are very close to a crisp partition. In FCM, most applications consider the value of m = 2 [23], [17]. The Lagrangian function is then expressed by :

$$\begin{aligned} L = & \frac{1}{2}\sum_{k=1}^{c}\|\mathbf{w}_k\|^2 + C\sum_{k=1}^{c}\sum_{i=1}^{n}\chi_{ki}^{m}(\xi_{ki} + \xi_{ki}^*) \\ & - \sum_{i=1}^{n}\lambda_i(\sum_{k=1}^{c}\chi_{ki} - 1) - \sum_{k=1}^{c}\sum_{i=1}^{n}\alpha_{ki}\,(\varepsilon + \xi_{ki} - r_{ki}) \\ & - \sum_{k=1}^{c}\sum_{i=1}^{n}\alpha_{ki}^*\,(\varepsilon + \xi_{ki}^* + r_{ki}) - \sum_{k=1}^{c}\sum_{i=1}^{n}(\eta_{ki}\xi_{ki} + \eta_{ki}^*\xi_{ki}^*) \end{aligned} \tag{8}$$

where $\alpha_{ki}^{(*)}$, $\eta_{ki}^{(*)}$ and λ_i are nonnegative Lagrange multipliers. First, we differentiate L with respect to the primal variables $w_k, b_k, \xi_{ki}^{(*)}$ and χ_{ki}. We obtain for a given structure l and a given data j :

$$\partial L_{b_l} = \sum_{i=1}^{n}(\alpha_{li}^* - \alpha_{li}) = 0 \tag{9}$$

$$\partial L_{\mathbf{w}l} = \mathbf{w}_l - \sum_{i=1}^{n}(\alpha_{li} - \alpha_{li}^*) = 0 \tag{10}$$

$$\partial L_{\xi_{lj}^{(*)}} = \chi_{lj}^{m}\ C - \alpha_{lj}^{(*)} - \eta_{lj}^{(*)} = 0 \tag{11}$$

$$\partial L_{\chi_{lj}} = m\ C\ \chi_{lj}^{m-1}(\xi_{lj} + \xi_{lj}^*) - \lambda_j = 0 \tag{12}$$

By substituting the Fuzzy constraint (5) into (12), we eliminate the variables λ_j and the following expression for the fuzzy weight χ_{lj} is obtained:

$$\chi_{lj} \ = \left[\sum_{k=1}^{c} \left(\frac{\xi_{lj} + \xi_{lj}^*}{\xi_{kj} + \xi_{kj}^*} \right)^{\frac{1}{m-1}} \right]^{-1} \tag{13}$$

The direct substitution of (9), (10), (11) and (13) into (8) makes the resolution of the dual problem difficult. Instead of that, we consider that the weigths χ_{lj} are beforehand estimated before the resolution of (8). In (13), the slack variables ξ_{lj} and ξ_{lj}^* measure the deviation of the data vector

$\mathbf{x}_j$ "above" and "below" the ε-tube of the *l-th* structure, respectively. These variables are connected with the residuals by the following relations : $\xi_{lj} = \max(0, r_{lj} - \varepsilon)$ and $\xi^*_{lj} = \max(0, -r_{lj} - \varepsilon)$. As we can note, these variables are **non-negative** and **if one of both is positive the other is 0**. Then, we deduce that the sum $\xi_{lj} + \xi^*_{lj}$ can be replaced by max$(0, r_{lj} - \varepsilon, -r_{lj} - \varepsilon)$ which is always ≥ 0, and the expression of χ_{lj} becomes :

$$\chi_{lj} = \left[\sum_{k=1}^{c} \left(\frac{max(0, r_{lj} - \varepsilon, -r_{lj} - \varepsilon)}{max(0, r_{kj} - \varepsilon, -r_{kj} - \varepsilon)} \right)^{\frac{1}{m-1}} \right]^{-1} \quad (14)$$

As the residual r_{lj} represents the distance between the *j-th* data with the *l-th* model, Equation (14) is similar with the updating of Fuzzy weights in Fuzzy c-means ([2]). Thus, the weights χ_{lj} can be estimated independently of the other variables.

Then, substituting equations (9), (10) and (11) into (8) yields the following dual problem :

$$\max_{\alpha_{ki}, \alpha^*_{ki}} D = \max_{\alpha_{ki}, \alpha^*_{ki}} (\sum_{k=1}^{c} D_k) \quad (15)$$

$$\text{subject to : } \sum_{k=1}^{c} \sum_{i=1}^{n} (\alpha^*_{ki} - \alpha_{ki}) = 0$$
$$\text{and } \alpha^*_{ki}, \alpha_{ki} \in [0, \chi^m_{ki}.C]$$

where

$$D_k = \sum_{i,j=1}^{n} (\alpha_{ki} - \alpha^*_{ki})(\alpha_{kj} - \alpha^*_{kj}).(\mathbf{x_i}, \mathbf{x_j}) \quad (16)$$
$$-\varepsilon \sum_{i,j=1}^{n} (\alpha_{ki} + \alpha^*_{ki}) + \sum_{i=1}^{n} y_i(\alpha_{ki} - \alpha^*_{ki})$$

is the dual quadratic expression of the *k-th* structure.

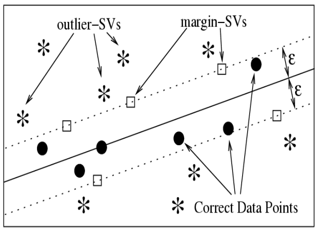
In our implementation, instead of maximizing the global quadratic form (15), we maximize independently each quadratic form D_k (Eq. 16) subjected to the constraints of Eq. (15).

Consequently, as we can notice, the introduction of the Fuzzy weights in the SVR formulation do not change the dual formulation problem with the exception that the upper bounds of Lagrange multipliers $\alpha_i^{(*)}$are modified by dynamical boundaries (the upper bounds become $\chi^m_{ki}C$ instead of C). To explain the effect of the weight factors, the values of the dual variables $\alpha_i^{(*)}$ and their corresponding slack variables $\xi_i^{(*)}$ are analysed from Figure 1. Figure 1-a shows a typical result of the linear SVR where the optimal solution is illustrated by a solid line and the ε-margin by dotted lines. Then, the data set can be divided into 3 subsets: data points inside the margin ('•'), data points on the margin ('□') and data points outside the margin ('*'). In practice, a simple test on the pair (α_i, α^*_i) (Fig. 1-b) makes it possible to classify the data in these three groups. As we can see on this table, only the subsets being on and outside the margin give a non-zero contribution to SVM solution (see Equation 3). Data which are inside the margin are considered as not support vectors and therefore have no contribution to SVM solution. This property is defined by $\alpha_i^{(*)} = 0$ and $\xi_i^{(*)} = 0$. Then, by weighting the value of a slack variable, we can modify the "weight" of a data vector $\mathbf{x}_i$ according to a structure c.

Then, following the previous remarks, we propose to solve the supervised Fuzzy Weighted SVR algorithm by the following steps (c is known):

1. Randomly initialization of $\mathbf{w}_k$ and b_k $(1 \leq k \leq c)$.
2. For each model k :
   - 2.1. Compute the residuals $r_{ki}, 1 \leq k \leq c, 1 \leq i \leq n$.
   - 2.2. Update $\chi_{ki}$with (14).
   - 2.3 estimate $\mathbf{w}_k$ and b_k with the solution of (15).
3. Apply 2 until stabilization of the χ_{ki} term.

In the next section, the performances of this algorithm are analysed in the linear case.



(a)

●	Not SV	$\xi_i^{(*)} = 0$	$\alpha_i^{(*)} = 0$
□	Is SV	$\xi_i^{(*)} = 0$	$\alpha_i^{(*)} \in]0,C[$
*	Is SV	$\xi_i^{(*)} > 0$	$\alpha_i^{(*)} = C$

(b)

Fig. 1. (a) Location of SVs and non SVs with respect to the margin. (b) Values of slack variables and dual variables for different subsets.

IV. Performance and comparative results

In this section, the performances of our algorithm in line fitting will be demonstrated and its tolerance to large percentages of outliers will be compared with the standard Fuzzy c-lines of [17]. The performance measure is based on the following expression :

$$S = \frac{1}{n * c} \sum_{k=1}^{c} \sum_{i=1}^{n} (y_{ki} - \widehat{y}_{ki})^2 \quad (17)$$

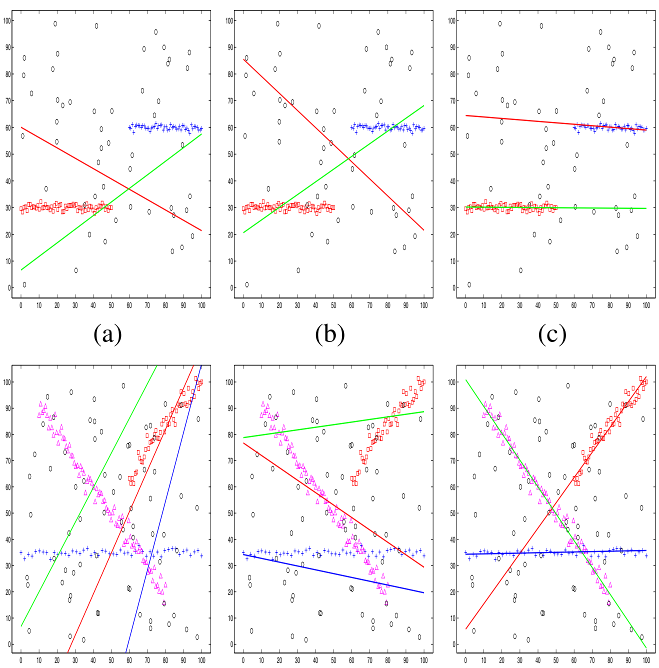
where $y_{ki} = (\mathbf{w}_k, \mathbf{x_i}) + b_k$ (resp $\widehat{y}_{ki}$) is the true (estimated) y-value for the i-th data point and the k-th structure.

In the following tests, we fixed ε=0.001, C=10 for the FW-SVR. In [5], it has been shown that the value of the ε parameter plays the most important role for linear SVM regression, whereas SVM solutions are rather insensitive to the regularization parameter C. Thus, we consider a small fixed ε- value in order to obtain good initial estimates. The coefficient m of Fuzzy weights is fixed to 2 for both methods. These values will be used in the rest of this paper if no specific value is explicitly stated. We assume that the number of linear structures is known. Every line is corrupted by a gaussian noise with zero mean and different variance σ. The i^{th} line has n_i samples. We add n_O uniformly distributed outliers in the range of (0,100). The training data sets are defined as follows:

A step : x=(0 50), y=30, $\mathbf{n_1}$=60, σ=1.5 ('□'); x=(60,100), y=60, $\mathbf{n_2}$=50, σ=2 ('+'); $\mathbf{n_0}$=50 ('o') (Fig. 2)).

Three crossed lines : x=(60,100), y=x+3, $\mathbf{n_1}$=50, σ=1 ('□'); x=(0,100), y=35, $\mathbf{n_2}$=50, σ=1 ('+'); x=(10,80), y=-x+100, $\mathbf{n_3}$=80, σ=4 ('△'); $\mathbf{n_0}$=70 ('o') (Fig. 3)).

The first example (Fig. 2) illustrates, for example, a pair of range surfaces forming a step discontinuity. Despite a poor initialization (Fig. 2-a) and a highly outliers percentage (31%), FW-SVR succeeds in extracting the linear structures (Fig. 2-c) whereas the standard approach completely fails (Fig. 2-b). The second example (Fig. 2-d) illustrates three crossed lines with different noise variances. Again, FW-SVR confirms its high robustness to correctly extract the three linear structures (Fig. 2-f) while the Fuzzy c-prototypes algorithm again fails (Fig. 2-e). For information, table 1 provides the value of S and the iteration number N returned by the two approaches. It clearly shows that FW-SVR is more performant and reach faster the solution than the standard Fuzzy c-lines. The stopping criterion for the two methods is based on the stabilization of the fuzzy weights with the same threshold.



(a) (b) (c)

(d) (e) (f)

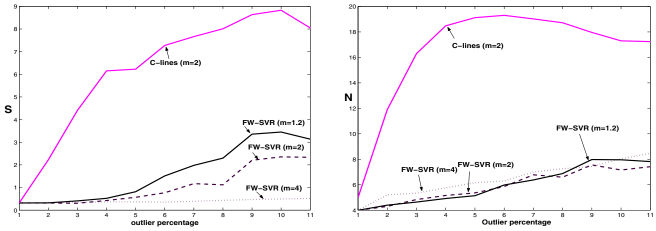
Fig. 2. Multiple linear model estimation : **step edge** (top row) - **three crossed lines** (bottow row). Initialization (a) and (d). Results of the Fuzzy c-lines algorithm (b) and (e). Results of the Fuzzy SVR algorithm (c) and (f).

In this experiment, we consider again the data set of figure 2-a with the same initialization. To illustrate the robustness of the proposed approach, the percentage of added outliers changes from 0% to 60%. Outliers are randomly distributed in the range [0,100]. The performances of the proposed

TABLE I

COMPARATIVE RESULTS

Methods	Step edge	Crossed lines
FW-SVR	$S = 1.02$ $N = 5$	$S = 1.04$ $N = 9$
C-lines	$S = 16.96$ $N = 22$	$S = 7.92$ $N = 68$



(a) (b)

Fig. 3. Evolution of S and N with respect to the percentage of outliers : **three crossed lines**. Result of the Fuzzy c-prototypes algorithm (grey curve, m=2). Results of the Fuzzy SVR algorithm (black curves : solid (m=1.2), dashed (m=2) and dotted (m=4)).

method are also studied using three different values of m (m=1.2, 2 and 4). We repeat this experiment 100 times and the average results are shown in Figure 3-a which illustrates the evolution of S with respect to the contamination rate. The solid, dashed and dotted black curves represent the results of the Fuzzy-SVR for m =1.5, 2 and 4, respectively. The grey curve corresponds to the results of the standard Fuzzy-c prototypes (m =2). As we can note first, FW-SVR approach benefits of the inherent robustness properties of standard SVR and outperforms standard Fuzzy c-prototypes algorithm in a degraded situation. Secondly, the obtained results seem to depend on the given value of m. It seems that insensitivity to outliers is higher (on average) when m increases. these recent results will have to be confirmed later. Figure 3-b shows the evolution of the average iteration number (N) with respect to the contamination rate. As previously, these results confirm that FW-SVR reachs faster the solution than standard Fuzzy-c prototypes algorithm.

V. APPLICATION TO VIDEO OBJECT TRACKING

In this section, we illustrate the performance of the proposed approach for rigid object tracking in image sequences. Here, the tracking problem is formulated as discovering the geometric transforms of object images between frames according to the extracted feature correspondences. In order to obtain a valid estimate of the transforms, a correct detection of feature correspondences is essential, which is, however, not easy in practice due to three factors: (1) the similarities, such as similar intensities and shapes, shared among features; (2) the occlusions and (3) the noise, which might also drive the data away from where they should be. Therefore, developing a robust feature matching technique is especially important.

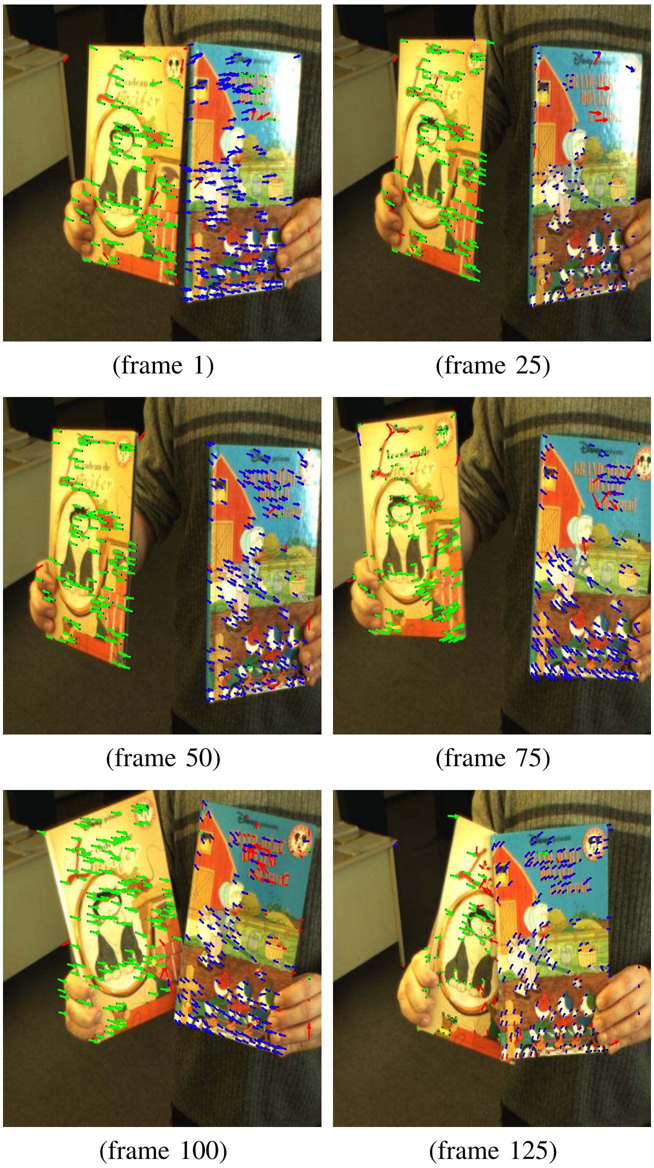


Fig. 4. Motion segmentation of two books in a video sequence (125 frames). Grey and black vector fields illustrate the segmented motions of the two books.

Fig. 5. Tracking of two figures in the video sequence The grey and black cercles represent the tracking results.

In this study, the matching problem is solved with a two step algorithm. The first one computes the ”corners” on the objects of interest in two consecutive frames. These feature points are extracted according to the Harris operator [14]. Then, the best correspondences are selected in the sense of both shape similar and intensity agreeing. The intensity based similarity measure is defined by the similarity of the histograms computed in the neighborhood of feature points. We use Bhattacharyya coefficient as a measure of similarity between two histograms [8]. The shape similarity measure is based on the maximum-likelihood edge template matching technique of [21]. For simplicity, we assume the object motions to be locally affine. The affine transform of object images is formulated as follows:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} a_1 & b_1 \\ a_2 & b_2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} \tag{18}$$

where coefficients (a_1, b_1, c_1, a_2, b_2, c_2) are the affine parameters. (x’,y’) and (x,y) are the points in the target template images. The estimation of affine parameters can be reformulated as a regression problem. Under SVR formalism, if we put **w**=(a_1, b_1, c_1, a_2, b_2, c_2), we obtain two constraint sets : one constraint over x’ :

$$\left\{ \begin{array}{l} r_{x'_i} = x'_i - \Big(\mathbf{w}, \phi_{x'_i}(x_i, y_i)\Big) - b \leq \varepsilon + \xi_{x'_i}, \\ -r_{x'_i} = -x'_i + \Big(\mathbf{w}, \phi_{x'_i}(x_i, y_i)\Big) + b \leq \varepsilon + \xi^*_{x'_i} \end{array} \right. ,$$

and one constraint over y’ :

$$\begin{cases} r_{y'_i} = y'_i - \Big(\mathbf{w}, \phi_{y'_i}(x_i, y_i)\Big) - b \leq \varepsilon + \xi_{y'_i}, \\ -r_{y'_i} = -y'_i + \Big(\mathbf{w}, \phi_{y'_i}(x_i, y_i)\Big) + b \leq \varepsilon + \xi^*_{y'_i} \end{cases}$$

with $\phi_{x'_i}(x_i, y_i) = (x_i, y_i, 1, 0, 0, 0)$ and $\phi_{y'_i}(x_i, y_i) = (0, 0, 0, x_i, y_i, 1)$. Thus, the estimation consists of extracting independently two planes in the data space. Next, $\mathbf{w}$ and b are estimated with the Fuzzy weighted SVR as previously presented. Note that, the parameters c_1 and c_2 have to be added to the bias term b.

The application concerns a real image sequence of about 125 frames which represents two independently moving books. The goal here is to accurately separate the two corresponding motions (Figure 4) and track two preselected figures on these books (cercles on Figure 5). In the first frame, two motion vectors are randomly generated. After estimation, motion vectors thus obtained serve as initial motion vectors for the next frame, and so on... Feature correspondences are computed in the whole image. Figure 4 displays 6 selected frames of the sequence. In each frame, we show a vector field which illustrates corner matches between the current frame and the next frame. We clearly distinguish two dominant vector fields (target books) corrupted by the presence of outlier motions (false matches). As we can see on these figures, the proposed approach accurately separates the two motions (black and grey vectors) from outliers matching. These two motion clusters correspond to the hardening of each membership value χ_{ki} with a threshold fixed to 0.8. On the other hand in Figure 5, figures are consistenly tracked throughout the sequence with the proposed approach (black and grey cercles).

VI. Conclusion

This paper is an attempt to addressing the problem of simultaneously extracting multiple linear structures with SVR. It is based on a fuzzy weighting strategy as in Fuzzy c-means. Our approach benefits of the well known robustness properties from SVR [7] and therefore outperforms the standard Fuzzy c-prototypes algorithm in a noisy real situation. This advantage is illustrated in a variety of synthetic data sets. Our method is also applied for motion estimation and object tracking in a real video sequence.

Acknowledgment

This work is supported by région Nord Pas-de-Calais under AutoRIS project.

References

[1] R. W. B. Schölkopf, A.J. Smola and P. Bartlett. New support vector algorithms. *Neural Computation*, 12:1083–1121, 2000.

[2] J. Bezdek. *Pattern recognition with fuzzy objective function algorithms*. Plenum Press, New York, 1981.

[3] H. Chen and P. Meer. Robust computer vision through kernel density estimation. *European Conference on Computer Vision*, pages 236–250, 2002.

[4] V. Cherkassky and Y. Ma. Selection of meta-parameters for support vector regression. In *Int. Conf. Artificial Neural Networks*, pages 687–693, 2002.

[5] V. Cherkassky and Y. Ma. Multiple model regression estimation. *IEEE Transactions on Neural Networks*, 16:785–798, 2005.

[6] Y. Choi and R. Krishnapuram. Fuzzy and robust formulations of maximum-likelihood-based gaussian mixture decomposition. *Proc. Int'l Conf. Fuzzy Systems, New Orleans*, pages 1899–1905, 1996.

[7] A. Christmann and I. Steinwart. On robustness properties of convex risk minimization methods for pattern recognition. *Journal of Machine Learning*, 5:1007–1034, 2004.

[8] V. R. D. Comaniciu and P. Meer. Real-time tracking of non rigid objects using mean shift. *IEEE Inter. Conf. On Computer Vision and Pattern Recognition*, 2:142–149, 2000.

[9] G. Danuser and M. Stricker. Parametric model fitting : from inlier characterization to outlier detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 20(2):263–279, 1998.

[10] H. Frigui and R. Krishnapuram. A robust algorithm for automatic extraction of an unknown number of clusters from noisy data. *Pattern Recognition Letters*, 17:1223–1232, 1996.

[11] H. Frigui and R. Krishnapuram. Clustering by competitive agglomeration. *Pattern recognition*, 30(7):1223–1232, 1997.

[12] H. Frigui and R. Krishnapuram. A robust competitive clustering algorithm with applications in computer vision. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 21(5):450–465, 1999.

[13] J. Ghosh. Multiclassifier systems : Back to the future. *Proc. Multiple Classifier Systems (MCS'02)*, 2364:1–15, 2002.

[14] C. Harris and M. Stephens. A combined corner and edge detector. *Proceedings Alvey conference*, pages 189–192, 1988.

[15] P. Huber. *Robust Statistics*. Wiley, New York, 1981.

[16] R. Jayadeva and S. Chandra. Regularized least squares fuzzy support vector regression for time series forecasting. *International Joint Conference on Neural Networks*, pages 593–598, 2006.

[17] R. G. J.C. Bezdek, C. Coray and J. Watson. Detection and characterization of cluster substructure. i. linear structure, fuzzy c-lines. *SIAM J. appl. Math*, 40:339–357, 2005.

[18] R. Krishnapuram and J. Keller. *Fuzzy and Possibilistic clustering methods for computer vision.*, pages 133–159. 1994.

[19] C. Lin and S. Wang. Fuzzy support vector machines. *IEEE Transaction Neural Networks*, 13(2):464–471, 2002.

[20] O. Nasraoui and R. Krishnapuram. A genetic algorithm for robust clustering based on a fuzzy least median of squares criterion. *Proc. Amer. Fuzzy Information Processing Soc. Conf., New York*, pages 217–221, 1997.

[21] C. Olson. Maximum-likelihood template matching. *Proceedings of the International Conference on Computer Vision*, 2:52–57, 2000.

[22] P. Rousseeuw and A. Leroy. *Robust regression and outlier detection*. John Wiley, 1987.

[23] Z. Sun and Y. Sun. Fuzzy support vector machine for regression estimation. *IEEE International Conference SMC*, pages 3336–3341, 2003.

[24] V. Vapnik. *The Nature Of Statistical Learning Theory*. 2nd edition. Springer Verlag, 1999.

[25] H. Wang and D. Suter. Variable bandwidth qmdpe and its application in robust optical flow estimation. In *ICCV*, pages 178–183, Nice, France, 2003.

[26] K. Wu and K. Yap. Fuzzy svm for content-based image retrieval: a pseudo-label support vector machine framework. *IEEE Computational Intelligence Magazine*, 1(2):10–16, 2006.