

Trainable Estimators for Indirect People Counting: A Comparative Study

Giovanni Acampora and Vincenzo Loia
Dept. of Computer Science
University of Salerno
Fisciano, Italy 84084
Email: {gacampora, loia}@unisa.it

Gennaro Percannella and Mario Vento
Dept. of Electronic Engineering and Information Engineering
University of Salerno
Fisciano, Italy 84084
Email: {pergen, mvento}@unisa.it

Abstract—Estimating the number of people in a scene is a very relevant issue due to the possibility of using it in a large number of contexts where it is necessary to automatically monitor an area for security/safety reasons, for economic purposes, etc. The large number of people counting approaches available in the literature can be roughly ascribed to two categories: direct approaches and indirect ones. In the first category there are methods that first detect people and then count them; differently, the indirect methods face the counting problem by establishing a relation between some scene features and the estimated number of people. Some recent comparative evaluations carried out in the framework of the PETS initiative have demonstrated that the indirect methods tends to be more robust than direct ones, above all when they are used in very crowded conditions. In this paper, we analyze the behavior of an indirect approach that is based on a trainable estimator that does not require an explicit formulation of a priori knowledge about the perspective and density effects present in the scene at hand. In particular, we investigate on the way the counting accuracy in different crowding conditions is affected by the choice of the trainable estimator.

Index Terms—Indirect People Counting, Trainable Estimators, ANFIS, ϵ -SVR.

I. INTRODUCTION

Knowing the number of people present in an area is a so important issue in the framework of video surveillance applications, that an increasing number of papers on this topic have been proposed in the recent past.

Despite of the fact that recently some pioneering systems have been made available, further improvements are still necessary, especially concerning their generality and flexibility. Many aspects make the problem really challenging: systems are required to work in real time on general purpose computers, possibly in parallel on different video streams coming from megapixel cameras, so as to supply up to date information on crowd density. Of course, the computational charge is a crucial but a less important issue with respect to the expected accuracy: as the output is mainly used for safety issues, the estimation accuracy of the number of people must be sufficiently high, even in presence of dense crowds. To this concern, it is worth pointing out that in these situations only parts of people's body appear in the image; the occluded parts generally cause significant underestimation in the counting process; it

means that the partial occlusions must be forecasted and suitably taken into account starting from the information about the crowd density. Another crucial point is the unavoidable presence of perspective distortions: the optical equipment of the cameras, especially those with wide angle views, cause that people far from the camera appear small while the close ones significantly bigger. So, methods must deal with this perspective issue, so as to obtain an estimation independent from the local scale of the image. Moreover, it is convenient that the system works with uncalibrated cameras, as calibration is generally time consuming and demands for suitable technical skills, not owned by the end user. Consequently, the availability of simple training procedures, not requiring knowledge about the internal organization of the computational model, but possibly depending on simple geometric properties derivable from the scene, is an extremely desirable feature.

The literature on people counting presents two conceptually different ways to face this task. In the *direct approach* (also called *detection-based*), each person in the scene is individually detected, using some form of segmentation and object detection; the number of people is then trivially obtainable. In the *indirect approach* (also called *map-based* or *measurement-based*), instead, counting is performed using the measurement of some features that do not require the separate detection of each person in the scene; these features then have to be put somehow in relation to the number of people.

The direct approach has the advantage that people detection is often already performed on a scene for other purposes (e.g. detecting events based on a person's position or trajectory), and as long as people are correctly segmented, the count is not affected by perspective, different people densities and, to some extent, partial occlusions. On the other hand, correct segmentation of people is a complex task by itself, and its output is often unreliable, especially in crowded conditions (which are of primary interest for people counting). Recent examples of the direct approach can be found in [1], [2] and [3].

The indirect approach, instead, is more robust, since it is based on features that are simpler to detect, but it is often not easy to find an accurate correspondence between these features and the number of people, especially if people may appear in the scene at different distances from the camera, and in groups

This research has been partially supported by A.I.Tech s.r.l., a spin-off society of the University of Salerno (www.aitech-solutions.eu).

with diverse densities. For the indirect approach, methods such as the use of measurements such as the amount of moving pixels [4], blob size [5], fractal dimension [6] or other texture features [7] have been proposed.

However, one of the most recent indirect approaches has been presented in [8] where the estimation of the count is obtained through a trainable estimator that takes into account perspective effects and the influence of people density on the detection of corner points, and consequently, can be easily adapted to the characteristics of a new scene. This paper is devoted to analyze the behavior of the mentioned approach when it is used with two different estimators: ANFIS and ϵ -SVR.

As will be shown in the experimental results, even though the two approaches yield similar global performance, the neuro-fuzzy based estimator is more efficient when the scenes are characterized by a high density crowd, whereas, the approach based on ϵ -SVR works in a better way when the scenes are characterized by a low density crowd. This comparative analysis opens new scenarios for the development of novel indirect approaches for people counting able to adaptively select the most suitable estimator that reduces the output error measure under particular scene conditions.

II. AN INDIRECT PEOPLE COUNTING APPROACH BASED ON A TRAINABLE ESTIMATOR

In this section, a description of an indirect approach based on a trainable estimator is provided in order to evaluate its performance when different regression methods are used to estimate the number of people. The approach uses as its features the moving interest points, where the interest points are first detected using a feature detector from the state of the art, and then the static ones are filtered out on the basis of a motion vector estimation.

An outline of the considered method is described by the following steps: Moving SURF points detection, SURF point Clustering and Feature extraction and regression. More in details, in the first step, SURF algorithm proposed by H. Bay et al. in 2006 [9] is adopted in order to detect interest points; in the second step, the detected points are partitioned into clusters corresponding to separated groups of persons, as to be able to compute for each group its distance from the camera and its density; in the last step, a trainable function estimator is exploited to estimate the number of people in a scene.

This paper focuses on the last step. Precisely, the aim of this work is to compare two different regression methods, ANFIS and ϵ -SVR, when they are used to implement the trainable function estimator of an indirect people counting approach.

However, before presenting the description of the explored methods, in the following subsections, a description of the architecture (see Fig. 1) of the considered approach is given.

A. Moving SURF points detection

In order to detect interest points associated to people, two basic assumptions are made: persons within the scene are not static and there are no other moving elements in the

scene. Thus, if we compute the interest points of the image and the associated motion information, the above assumptions guarantee that only the interest points with a non null motion vector must be associated to people. As proposed in [10], the interest points associated to people are extracted in two steps. First, we determine all the interest points within the frame under analysis. Then, we prune the points not associated to persons by taking into account their motion information. In this approach, interest points are determined by using the SURF algorithm.

In order to remove the static interest points (that are not associated to people), for each point detected by the SURF algorithm we estimate the motion vector with respect to the previous frame by using a block-matching technique. Then the algorithm distinguishes between static and moving interest points on the basis of the following rule:

$$p(x, y) = \begin{cases} \text{moving point} & \text{if } |\vec{v}(x, y)| > \beta \\ \text{static point} & \text{if } |\vec{v}(x, y)| \leq \beta \end{cases} \quad (1)$$

where $p(x, y)$ is the interest point at the x, y coordinates, $|\vec{v}(x, y)|$ is the magnitude of the motion vector calculated in x, y with respect to the previous frame; β is a bias value (in our experiments we set $\beta = 0.0$).

B. SURF point Clustering

In order to compensate for changes in the number of points due to perspective and to partial occlusions, the algorithm needs to partition the detected points into clusters corresponding to separated groups of persons, so as to be able to compute for each group its distance from the camera and its density.

The faced clustering problem is characterized by the fact that there is no priori knowledge about the number and the

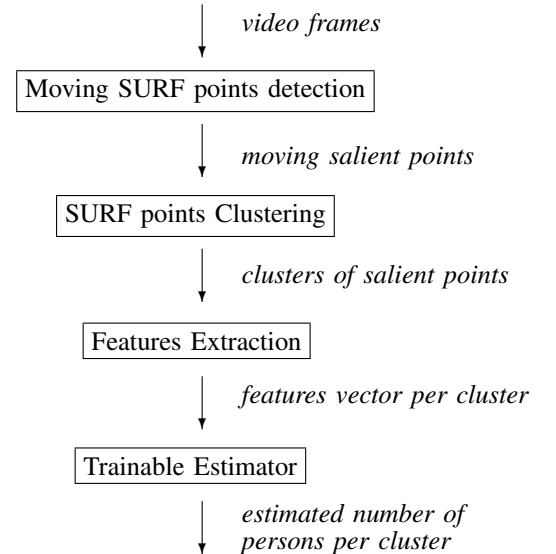


Fig. 1. System architecture.

shape of the clusters to be found. As observed in [11], the clustering algorithms based on graph theory are well suited to face clustering problems where no assumptions can be made about the clusters. In particular, the considered algorithm adopts the technique presented in [12], since (differently from other algorithms in the graph-based clustering family) it requires no parameters to be tuned or adapted to the particular application.

This algorithm represents the set of points as a graph in which each point corresponds to a node and each edge is labeled with the distance between its endpoints. The minimum spanning tree (MST) of the graph is computed; this tree will contain some edges that are between nodes in the same cluster (intra-cluster edges) and other edges between nodes of different clusters (inter-cluster edges). Assuming that the clusters are well separated, it can be expected that the intra-cluster edges are shorter than the inter-cluster edges. So the algorithm uses a thresholding to divide the edges in two sets (the ones below the threshold, say it λ , and the ones above the threshold λ). The edges in the second set are deleted, and the remaining connected components are the clusters output by the algorithm.

The use of a fixed value for the threshold λ would be problematic, since the threshold would need to be adjusted depending on the resolution, the distance from the camera and so on. Instead, this approach used a threshold proportional to the average edge length, computed as:

$$\lambda = \gamma \cdot \frac{1}{N} \sum_{i=1}^N x_i \quad (2)$$

where γ is the proportionality factor, N is the number of edges of the spanning tree, while x_i is the weight of the i -th edge of the tree. We have experimentally found that the choice $\gamma = 2.0$ works adequately for all the considered scenes.

C. Feature extraction and regression

In this stage of the algorithm, a feature vector is computed from each cluster detected in the previous step, and is fed into a regressor. The output of the regressor is the estimated number of persons in the group represented by the cluster.

The considered approach reasonably assumes that when the density of people increase, the detected points get closer to each other. So, the density of the points as related to people density is considered, and indirectly it is possible to take into account people density by establishing a relation between the average number of points per person and the point density.

The relation between the number of interest points and the number of people appears more complex than a direct proportionality, because it is possible to take into account also the distance of the people from the camera and the point density. This relation can be formulate as:

$$n_{\text{people}} = f(n_{\text{points}}, \rho, d) \quad (3)$$

where:

- n_{people} is the estimated number of people;
- n_{points} is the number of interest points within the cluster;

- ρ is the average density of the points in the cluster: the value is obtained as the ratio between the number of points into the cluster and the area of the bounding box. Note that the area of the bounding box is computed with respect to real world coordinates. This allows us to normalize the average density of the points to the value it would have if the cluster were moved to a predefined distance from the camera;
- d is the distance of the cluster from the camera: assuming that the bottom points of the bounding box lie on the ground plane, the calculation is done by applying an Inverse Perspective Mapping and is referred to the center of the bottom edge of the cluster's bounding box.

Since the analytical form of f is not known a priori, the considered approach chooses to learn this function from a set of labeled examples by using a regression method.

In particular, this paper aims to compare two different techniques to implement the aforementioned function estimator: the ϵ -SVR regressor and ANFIS. Both explored methods are able to learn a non linear relation and show a good generalization ability. In details, they receive as inputs the number of points of a cluster, the distance from the camera and the point density of the cluster, and are trained (using a set of training frames) to output the estimated number of people in the cluster. However, as will be shown in section IV, ANFIS obtains better results when the people counting is applied to high density crowd, whereas the ϵ -SVR regressor provides more benefits when the indirect people counting method is used to estimate the number people in low density crowd.

III. TRAINABLE ESTIMATORS: ANFIS AND ϵ -SVR

In this section, a description of the two methods useful to implement the trainable function estimator for indirect people counting is provided. In particular, a brief description about ANFIS and ϵ -SVR is given.

A. Adaptive Neuro Fuzzy Inference System

Adaptive Neuro-Fuzzy Inference Systems (ANFIS) [13] is a class of adaptive networks that are functionally equivalent to conventional fuzzy inference systems. An adaptive network is a multilayer feed forward network in which each node performs a particular function (node function) on incoming signals using a set of parameters specific to this node [14]. The form of node functions may vary from node to node, and the choice of each node function depends on the overall function, which the adaptive network is designed to implement. Adaptive networks are composed by two different classes of nodes: *adaptive nodes* and *fixed nodes*. Adaptive nodes are characterized by a collection of modifiable parameters, named *parameter set*, whereas, the fixed nodes only deal with unmodifiable parameters. Adaptive networks work in order to achieve a desired input-output relationships by updating the parameter sets according to given training data and a gradient-based update procedure. ANFIS exploits this updating feature to learn and adapt the parameters of a given Takagi-Sugeno-Kang (TSK) fuzzy inference system.

As an example, Fig. 2 shows an adaptive network that models a first-order TSK fuzzy inference system composed by two rules:

- 1) If x is A_1 and y is B_1 Then $f_1 = p_1 \cdot x + q_1 \cdot y + r_1$
- 2) If x is A_2 and y is B_2 Then $f_2 = p_2 \cdot x + q_2 \cdot y + r_2$

where the symbols x and y are the system's inputs, A_1 , A_2 , B_1 and B_2 are linguistic labels determined by fuzzy sets defined on the universe of discourse related to the variables x and y whereas, the f_1 and f_2 are linear functions computing the output value of the TSK inference system.

To reflect the ANFIS adaptive capabilities, in Fig. 2 the square nodes represent the adaptive nodes, whereas the circle nodes represent the fixed nodes.

Each ANFIS layer implements a particular fuzzy inference subtask. In detail:

Layer 1 Every node i in this layer implements the node function:

$$O_i^1 = \mu_{A_i}(x) \quad (4)$$

where x is the input to node i , and A_i is a linguistic label. From a fuzzy inference point of view, O_i^1 is the membership function related to the fuzzy set A_i , and it specifies the degree to which the input x satisfies A_i . Typically, A_i is a bell-shaped function characterized by a set of three parameters $\{a, b, c\}$ corresponding to the parameter set of the node i .

Layer 2 The nodes labeled with Π multiply their input signals:

$$w_i = \mu_{A_i}(x) \times \mu_{B_i}(y) \text{ with } i = 1, 2; \quad (5)$$

where w_i represents the firing strength of a rule.

Layer 3 The node in this layer calculate the ratio of the i^{th} rules firing strength to the sum of all rules firing strengths:

$$\bar{w}_i = \frac{w_i}{w_1 + w_2} \text{ with } i = 1, 2. \quad (6)$$

The value w_i represents the *normalized firing strength*.

Layer 4 The adaptive node i in this layer computes the following function:

$$O_i^4 = \bar{w}_i f_i = \bar{w}_i (p_i x + q_i y + r_i) \quad (7)$$

where the collection $\{p_i, q_i, r_i\}$ is the parameter set of the node i and corresponds to the TSK consequent parameters.

Layer 5 The single and fixed node in this layer, labeled Σ , computes the overall network output as the summation of all incoming signals:

$$O_i^5 = \sum_i \bar{w}_i f_i = \frac{\sum_i w_i f_i}{w_i} \quad (8)$$

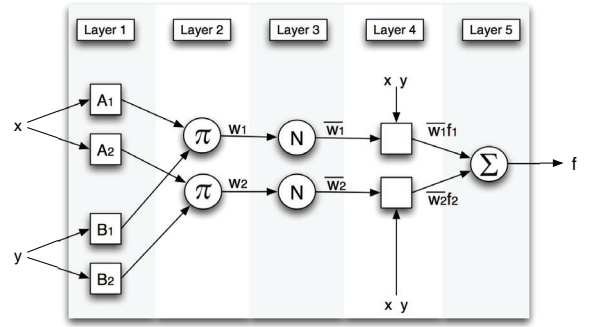


Fig. 2. The Adaptive Neuro Fuzzy Inference System.

B. ϵ -Support Vector Regression

The ϵ -SVR is a variant of the Support Vector Machines (SVM) designed to face the regression estimation task.

SVM is a class of learning algorithms originally developed for classification task, where in order to find a decision rule with good generalization ability one selects some (small) subset of the trained data called *Support Vectors*. Optimal separation of the Support Vectors is equivalent to optimal separation the entire data. In order to deal with the SV Regression, Vapnik [15] devised the so-called ϵ -insensitive loss function which does not penalize errors below some $\epsilon > 0$, chosen a priori [16]. Hereafter, a formal description of ϵ -SVR method is given.

As described in [17], let $G(x)$ be the unknown function where the vector $\mathbf{x} = [x_1, x_2, \dots, x_d]$ is the *input space* characterized by a dimensionality equal to d . Let $F = (\mathbf{x}, \mathbf{w})$ be a family of functions parameterized by $\mathbf{w}$ and $\hat{\mathbf{w}}$ be the value of $\mathbf{w}$ which minimizes the error between $G(\mathbf{x})$ and $F = (\mathbf{x}, \hat{\mathbf{w}})$. The objective is to estimate $\mathbf{w}$ with $\hat{\mathbf{w}}$ by observing N training instances $\mathbf{v}_j$, $j = 1, \dots, N$.

In order to achieve this aim, the approximation function F , employed by the SVR method, is defined:

$$F(\mathbf{x}, \hat{\mathbf{w}}) = \sum_{i=1}^N (\alpha_i^* - \alpha_i) (\mathbf{v}_i \cdot \mathbf{x} + 1)^p + b$$

This function represents an expansion which explicitly uses the N training vectors. Indeed, different from other approximation functions, F depends on $2N+1$ coefficients α_i^* , α_i and b that must be determined from the training examples. Now, let α represent the $2N$ values of α_i^* and α_i , the optimum values for the components of $\hat{\mathbf{w}}$ or α depends on the definition of the objective function. Formally, the objective function is defined as:

$$\|\mathbf{w}\|^2 + C \sum_{j=1}^N L[y_j - F(\mathbf{v}_j, \hat{\mathbf{w}})]$$

where L is a general loss function, F is the approximation function, y_j is the observation of $G(\mathbf{x})$ in the presence of noise, and the first term, i.e., the 2-norm of the weight vector $\hat{\mathbf{w}}$, is a regularizer. Finally, the constant C represents a further regularization.

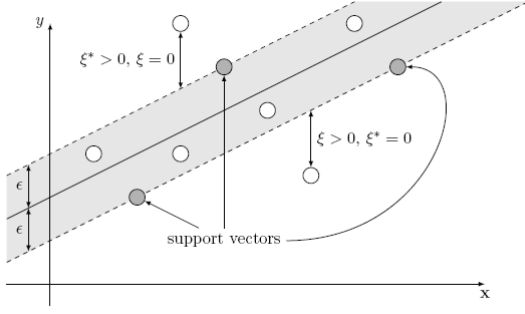


Fig. 3. The ϵ -SVR representation

In particular, the ϵ -SVR exploits as loss function the so-called ϵ -insensitive loss function defined as:

$$L = \begin{cases} 0 & \text{if } |y_j - F(\mathbf{x}_i, \hat{\mathbf{w}})| < \epsilon \\ |y_j - F(\mathbf{x}_i, \hat{\mathbf{w}})| - \epsilon & \text{otherwise} \end{cases} \quad (9)$$

This function defines an ϵ tube [18] (see Figure 3) so that if the predicted value is within the tube the loss is zero, while if the predicted point is outside the tube, the loss is the magnitude of the difference between the predicted value and the radius ϵ of the tube.

In this way, the objective function becomes to minimize this specific function:

$$\frac{1}{2} \cdot \mathbf{w}^2 + C \left(\sum_{i=1}^N \xi_i + \xi_i^* \right)$$

where ξ_i and ξ_i^* represents the so-called *slack variables* for each input point x_i . Precisely, ξ_i measures positive deviation of more than ϵ , while ξ_i^* indicates negative deviations. So, if a data point is situated within the tube, both slack variables are 0. However, since no point can be on both sides simultaneously, at least one of the two slack variables is always 0. This is mathematically expressed by means of the ϵ -insensitive loss function defined in the equation 9.

Finally, in order to complete the description of the ϵ -SVR method, the constraints on the objective function are described:

$$\begin{aligned} y_i - (\mathbf{w} \cdot \mathbf{v}_i) - b &\leq \epsilon + \xi_i \\ (\mathbf{w} \cdot \mathbf{v}_i) + b - y_i &\leq \epsilon + \xi_i^* \\ \xi_i &\geq 0 \\ \xi_i^* &\geq 0 \end{aligned} \quad (10)$$

IV. EXPERIMENTAL RESULTS

In this section, the performances of the indirect people counting approach are evaluated on a specific dataset by using two different estimators, ANFIS and ϵ -SVR, in order to compare the results provided by these regression methods.

In particular, the performance of the proposed methods has been assessed using the PETS2009 dataset¹. The dataset is organized in four sections, but our attention is primarily

¹<http://www.cvg.rdg.ac.uk/PETS2009/>

focused on the section named S1 that was used to benchmark algorithms for the "Person Count and Density Estimation" PETS2009 contest. The main characteristics of the subset of video sequences of the PETS 2009 dataset used for assessing the performance of the proposed method are summarized in the Table I in terms of their length, number of people in the scene (minimum, maximum and average number) and other elements as density of the crowd, illumination conditions, etc.

For our experimentations, four videos of the view 2, namely S1.L1.13-57, S1.L2.14-06, S1.L2.14-31 and S3.MF.12-43, are used. For all the sequences the number of people in the whole frame is computed.



(a)



(b)

Fig. 4. Examples of the frames of the video sequences (a) S1.L2.14-06 and (b) S3.MF.12-43 used for the test

In order to exploit the proposed function estimators for people counting, a phase of training is necessary. The training set was built by manually collecting some samples of people groups from a subset of the test frames. For each selected box, the feature vector and the associated ground truth, i.e. the true number of persons that are inside the box, are calculated. Samples were carefully selected in order to guarantee that all the possible combinations in terms of number of persons in

TABLE I
RELEVANT CHARACTERISTICS OF THE FOUR SEQUENCES OF THE PETS 2009 DATASETS USED FOR ASSESSING THE PERFORMANCE OF THE PROPOSED METHOD.

Video sequence View 2	Length (frames)	Conditions	Number of people		
			Min	AVG	Max
S1.L1.13-57	221	medium density crowd, overcast	8	34.19	46
S1.L2.14-06	201	high density crowd, overcast	3	37.10	46
S1.L2.14-31	131	high density crowd, overcast	10	35.19	43
S3.MF.12-43	108	very low density crowd, overcast	1	4.99	7

TABLE II
PERFORMANCE OF THE ANFIS AND ϵ -SVR. IN EACH CELL THERE ARE REPORTED THE VALUES OF THE MAE AND OF THE MRE (IN PARENTHESIS).

Video view 2	ϵ -SVR	Anfis
S1.L1.13-57	11.76 (30.0%)	11.02 (28.32%)
S1.L2.14-06	18.03 (43.0%)	11.61 (30.02%)
S1.L2.14-31	5.64 (18.8%)	6.99 (20.52%)
S3.MF.12-43	0.63 (18.8%)	1.79 (41.67%)

TABLE III
PERFORMANCE OF THE ANFIS AND ϵ -SVR. IN EACH CELL THERE ARE REPORTED THE VALUES OF THE MAE AND OF THE MRE (IN PARENTHESIS).

Video (view)	ϵ -SVR	Anfis
medium density crowd	11.76 (30.0%)	11.02 (28.32%)
high density crowd	11.84 (30.9%)	9.3 (25.27%)
low density crowd	0.63 (18.8%)	1.79 (41.67%)

the group, points density and distance from the camera were adequately represented in the training set. In our tests, about 30-40 training frames, which correspond to about 5% of the frames within the whole dataset, have been taken into account.

Testing has been carried out by comparing the actual number of people in the video sequences and the number of people calculated by the two considered function estimators, ANFIS and ϵ -SVR. The indices used to report the performance are the Mean Absolute Error (MAE) and the Mean Relative Error (MRE) defined as:

$$MAE = \frac{1}{N} \cdot \sum_{i=1}^N |G(i) - T(i)| \quad (11)$$

$$MRE = \frac{1}{N} \cdot \sum_{i=1}^N \frac{|G(i) - T(i)|}{T(i)} \quad (12)$$

where N is the number of frames of the test sequence and $G(i)$ and $T(i)$ are the guessed and the true number of persons in the i -th frame, respectively.

This index is very useful to exactly quantify the error in the estimation of the number of person which are in the focus of the camera, but it does not relate this error to number of people; in fact, the same absolute error can be considered negligible if the number of persons in the scene is high while it becomes significant if the number of person is of the same order of magnitude. For this reason, we introduced also the MRE index which takes into account the estimation error related to the true people number.

The performances of the ANFIS and ϵ -SVR algorithms applied on the adopted dataset are reported in Table II.

Starting from the Table II, it is possible to summarize that the two tested function estimators show a different behavior depending on the scene features. In particular, ANFIS method is more suitable to estimate number of people in scenes characterized by a high density crowd, instead, the ϵ -SVR is

more adequate for scene with a low density crowd. The results are summarized in the table III.

V. CONCLUSIONS

In this paper, we have analyzed and evaluated the performance of an indirect method for counting moving people in video-surveillance scenes by considering two different trainable estimators, ANFIS and ϵ -SVR. As highlighted in experiments both methods achieve good performances when applied to two different kinds of scenes. Precisely, the neuro-fuzzy based estimator obtains good performance when scenes are characterized by a high density crowd, whereas, the approach based on ϵ -SVR works in a better way when the scenes are characterized by a low density crowd. As a future work, we may use these different estimation behaviors to design an adaptive people counting method capable of identifying the scene features and dynamically selecting the most suitable estimator to deal with these features.

ACKNOWLEDGMENT

The authors would like to thank the PhD student Autilia Vitiello for her precious support to MatlabTM programming.

REFERENCES

- [1] J. Rittscher, P. Tu, and N. Krahnstoeber, "Simultaneous estimation of segmentation and shape," in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2005, pp. 486-493.
- [2] G. J. Brostow and R. Cipolla, "Unsupervised bayesian detection of independent motion in crowds," in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2006, pp. 594-601.
- [3] T. Zhao, R. Nevatia, and B. Wu, "Segmentation and tracking of multiple humans in crowded environments," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 7, pp. 1198-1211, 2008.
- [4] S.-Y. Cho, T. W. S. Chow, and C.-T. Leung, "A neural-based crowd estimation by hybrid global learning algorithm," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 29, no. 4, pp. 535-541, 1999.
- [5] D. Kong, D. Gray, and H. Tao, "A viewpoint invariant approach for crowd counting," in *International Conference on Pattern Recognition*, 2006, pp. 1187-1190.

- [6] A. N. Marana, L. da F. Costa, R. A. Lotufo, and S. A. Velastin, "Estimating crowd density with mikowski fractal dimension," in *Int. Conf. on Acoustics, Speech and Signal Processing*, 1999.
- [7] H. Rahmalan, M. S. Nixon, and J. N. Carter, "On crowd density estimation for surveillance," in *The Institution of Engineering and Technology Conference on Crime and Security*, 2006.
- [8] P. Foggia, G. Percannella, F. Tufano, and M. Vento, "A method for counting people in crowded scenes," in *2010 Seventh International Conference on Advanced Video and Signal Based Surveillance*, 2010.
- [9] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, "Surf: Speeded up robust features," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008.
- [10] A. Albiol, M. J. Silla, A. Albiol, and J. M. Mossi, "Video analysis using corner motion statistics," in *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*, 2009, pp. 31–38.
- [11] S. Theodoridis and K. Koutroumbas, *Pattern Recognition, Third Edition*. Academic Press, February 2006.
- [12] P. Foggia, G. Percannella, C. Sansone, and M. Vento, "A graph-based algorithm for cluster detection," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 22, no. 5, pp. 843–860, 2008.
- [13] J.-S. Jang, "Anfis: adaptive-network-based fuzzy inference system," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 23, no. 3, pp. 665–685, may/jun 1993.
- [14] E. Purwanto, S. Arifin, and B.-S. So, "Application of adaptive neuro fuzzy inference system on the development of the observer for speed sensor less induction motor," in *Electrical and Electronic Technology, 2001. TENCON. Proceedings of IEEE Region 10 International Conference on*, vol. 1, 2001, pp. 409–414 vol.1.
- [15] V. Vapnik, *The Nature of Statistical Learning Theory (Information Science and Statistics)*, 2nd ed. Springer, November 1999. [Online]. Available: [http://www.cs.orst.edu/~bulatov/papers/Vapnik, V. \(2000\) The Nature of Stastical Learning Theory, 167p.pdf](http://www.cs.orst.edu/~bulatov/papers/Vapnik, V. (2000) The Nature of Stastical Learning Theory, 167p.pdf)
- [16] B. Schölkopf, P. Bartlett, A. Smola, and R. Williamson, "Support vector regression with automatic accuracy control," in *Proceedings of ICANN'98*, ser. Perspectives in Neural Computing, L. Niklasson, M. Bodén, and T. Ziemke, Eds. Berlin: Springer Verlag, 1998, pp. 111–116. [Online]. Available: [/papers/SchBarSmoWil98.ps.gz](#)
- [17] H. Drucker, C. J. Burges, L. Kaufman, C. J. C, B. L. Kaufman, A. Smola, and V. Vapnik, "Support vector regression machines," 1996.
- [18] S. Klement, "Incremental Support Vector Regression for Steering Hot Rolling Mills," Diploma Thesis, Universität zu Lübeck, 2006.