Adaptive Neuro-Fuzzy Controller for Multi-Object Tracker

Duc Phu CHAU(1), K. SUBRAMANIAN(2), and François BREMOND(1)
Duc-Phu.Chau@inria.fr, kartick1@e.ntu.edu.sg,
Francois.Bremond@inria.fr

(1)STARS team, INRIA Sophia Antipolis, France
(2)School of Computer Engineering, Nanyang Technological University, Singapore

Abstract. Sensitivity to scene such as contrast and illumination intensity, is one of the factors significantly affecting the performance of object trackers. In order to overcome this issue, tracker parameters need to be adapted based on changes in contextual information. In this paper, we propose an intelligent mechanism to adapt the tracker parameters, in a real-time and online fashion. When a frame is processed by the tracker, a controller extracts the contextual information, based on which it adapts the tracker parameters for successive frames. The proposed controller relies on a learned neuro-fuzzy inference system to find satisfactory tracker parameter values. The proposed approach is trained on nine publicly available benchmark video data sets and tested on three unrelated video data sets. The performance comparison indicates clear tracking performance improvement in comparison to tracker with static parameter values, as well as other state-of-the art trackers.

1 Introduction

The research in field of object tracking [1, 2, 6, 17] is seeing growing interest due to its importance in the area such as video surveillance, motion-based recognition, human computer interaction. These trackers aim to accurately associate multiple objects across multiple frames. However, there are various challenges in this field. One of the main challenges is real-time object association by trackers. Moreover, the change in scene context might also affect the tracking performance drastically. In order to overcome this issue, various real time trackers as well as tracker parameter tuning mechanisms have been developed in literature.

An online learning scheme based on computationally expensive Adaboost is proposed in [3], to calculate a discriminative appearance model for each mobile object. Whereas, computationally expensive genetic algorithm based approach is employed in [5] to search for best tracker parameters. Moreover, genetic algorithm suffers from the tendency to get stuck in local optimal solution space. In literature, work employing multiple trackers to find the best parameters based on contextual information has also been proposed [6]. In these works, there are strong limitations on the online processing ability and self-adaptability to scene variations. In [7], an online parameter tuning approach to adapt the tracking parameters of [4] to scene variations has been proposed, and code-books are utilized to store the learned parameters. The parameter tuning is however done using a nearest neighborhood search, which is not accurate when the training set is not large enough. To summarize, all these approaches have some limitations in genericity, efficiency or performance.

In literature of soft computing, various prediction/ forecasting techniques have been proposed, such as support vector regression [8] or artificial neural networks [11] which can learn efficiently even over small data sets. With the development of evolving/ online learning algorithms, the training data is learned when it appears without the need to be stored. This results in significantly reduced space and time complexity in learning. The above mentioned learning mechanisms can learn any given data efficiently. However, they fail to generalize the learned knowledge over unseen data. Recently, in [9], a Meta-Cognitive sequential learning algorithm for evolving Neuro-Fuzzy Inference System (McFIS) has been proposed which can self-regulate its learning to attain better training as well as generalization accuracy.

In this paper, we propose a generic, efficient et robust controller which relies on McFIS to adapt online tracker parameters for scene context variations. The proposed approach brings the following contributions:

- Evolving Learning
  - One-shot meta-cognitive learning for faster training.
  - Evolving/adaptive learning to estimate the functional relationship between tracking contextual features and tracking parameters.
- Online Control
  - A generic controller for tuning parameters of any tracker to cope with video content variation.
  - Accurate estimation of tracker parameters utilizing all rules.
  - Fast inference of tracker parameters.

The proposed controller is trained on nine publicly available video data sets and is tested on three unrelated video data sets. The results indicate a significant improvement in comparison to other recent state-of-the-art trackers.

The rest of the paper is organized as follows. In Section 2, we present the proposed object tracker controller mechanism. The performance of the proposed system is evaluated in section 3. The paper is concluded and future work discussed in section 4.

2 Tracker Control Mechanism

In this section, an overview of the proposed tracker control mechanism is described. The control mechanism consists of two phases, namely, a learning phase and an online control phase (as shown in Figure 1). The aim of the learning phase is to find a functional relationship (F) between the contextual features of detected objects ($\mathbf{C}$) and the best tracker parameters ($\mathbf{P}$), for a given video chunk (v). The online control phase determines the satisfactory tracker parameters ($\mathbf{P}$) for a given contextual feature set ($\mathbf{C}$) based on the learned functional relationship. Since the proposed mechanism is not dependent on video, but only on video meta-data (contextual information), it is applicable to unknown/unseen video sequences. The generalization ability of the controller to unseen video depends on how efficient the learned functional relationship $\mathbf{C} \to \mathbf{P}$. In this

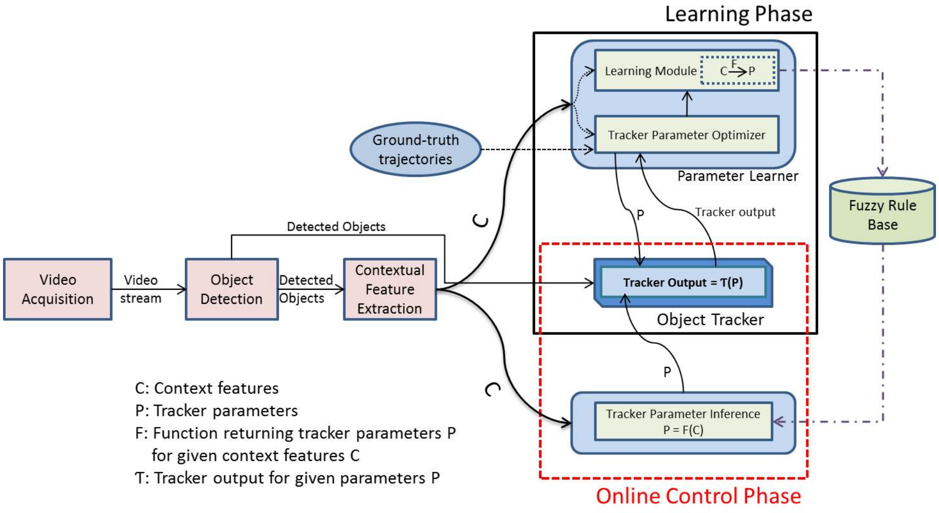


Fig. 1. Working of Proposed Tracker Control Mechanism

work, we employ a meta-cognitive neuro-fuzzy inference system [9] for learning the underlying functional relationship between $\mathbf{C}$ and $\mathbf{P}$.

Two practical assumptions made in realizing this control mechanism:

- There is no drastic change in the observed scene over a short period of time (1-2 seconds).
- The controlled tracker has a set of important parameters which could be adjusted online to improve tracking performance.

Six scene contextual parameters are employed in this study, and they will be described in detail in the next section. The video chunk size (v) is decided based on tracking context variation. Fast variation requires a small chunk size and vice-versa. Also, the learning phase and the online control phase are mutually independent and hence they could be run in parallel.

Next, we describe the contextual features employed in this study. To control the parameters of the tracker, a fuzzy controller is employed. The control rules for this controller are learned in an adaptive fashion, employing a meta-cognitive learning algorithm. A detailed description on the fuzzy controller and its learning algorithm, together referred to as Meta-Cognitive Neuro-Fuzzy Inference System is detailed in section 2.2.

2.1 Contextual Features

In this work, we use the six following features to define a context (or tracking context). These features are selected based on their effect on the tracker performance. The features employed are: density, occlusion, contrast, contrast variance, 2D area and 2D area variance. The calculation of each of these features and their effect on tracking is described next.

Density of Mobile Objects: Density of mobile objects affects the detection and tracking. More the mobile objects appear in scene, more the tracking issues can happen such as switch or lost of object identity. In this work, density of mobile objects at time t is defined as their 2D area occupancy over 2D camera view area.

Occlusion Level of Mobile Objects: Occlusion renders an object more difficult to detect and to track. Moreover, the variation of occlusion level results in loss of coherence of object appearance and causes the object tracking errors as consequence. The occlusion level of mobile objects at instant t is defined as the ratio between the 2D overlap area of objects and the object 2D areas.

Contrast of Mobile Objects: The contrast of an object is defined as the color intensity difference between this object and its surrounding background. Objects with low contrast imply a low discrimination between them, in particular for color descriptor-based trackers. In this work, contrast of an object at time t is calculated as the Earth Mover's Distance between the normalized intensity histograms of this object and its surroundings. The contrast value of mobile objects at frame t is defined as the mean value of all object contrasts at this frame.

Contrast Variance of Mobile Objects: In a video, when different contrast values exist, mean value cannot be representative of the multi-object contrast accurately. As a result, the variance of contexts across different objects in a scene is considered. The contrast variance at a frame is defined as the standard deviation of the contrast values computed at this frame.

2D Area of Mobile Objects: The 2D area characterizes the reliability of the object appearance for tracking. Greater the object area, higher object appearance reliability. The 2D area feature value at a frame is defined as the mean value of the 2D areas of mobile objects at the same frame.

2D Area Variance of Mobile Objects: Similar to contrast variance of mobile objects, 2D area variance of mobile objects is also defined as the standard deviation of the 2D areas of objects at the same frame.

McFIS based tracker controller infers the tracker parameters based on the above given scene contextual information and the knowledge stored in its knowledge base.

2.2 Meta-Cognitive Neuro-Fuzzy Inference System

In this section, we describe the employed fuzzy controller and its learning algorithm. The aim of the learning algorithm is to find the functional relationship between the contextual features $\mathbf{C}$ and tracker parameter values $\mathbf{P}$. The learned functional relationship is stored in form of Gaussian fuzzy rules. A fuzzy rule r is represented by three values: $[\boldsymbol{\mu}_r, \sigma_r, \boldsymbol{\alpha}_r]$ where $\boldsymbol{\mu}_r, \sigma_r$ denote the center and spread of the rule (representing information of the six above contextual features) and $\boldsymbol{\alpha}_r$ denotes its weightage (representing the tracker parameter values). For any given contextual input features set, McFIS calculates its similarity with existing fuzzy rules (membership). The tracker parameters are inferred as the weighted sum of these memberships. For example at any given time, let the controller consist of R fuzzy rules. The predicted tracker parameter values $\hat{\mathbf{P}}^t$ for a given scene contextual feature set $\mathbf{C}^t$ at time t, are then given as:

$$\hat{\mathbf{P}}^t = \frac{\sum_{r=1}^{R} \boldsymbol{\alpha}_r \phi_r}{\sum_{p=1}^{R} \phi_p} \tag{1}$$

where ϕ_r is a membership function and it represents the distance of context C to rule r and is defined in form of Gaussian function as follows:

$$\phi_r = \exp\left(-\frac{\|\mathbf{C^t} - \boldsymbol{\mu}_\mathbf{r}\|^\mathbf{2}}{2\sigma_r^2}\right) \tag{2}$$

During sequential learning, the aim of the meta-cognitive algorithm is to adapt/evolve the knowledge (fuzzy rules) such that the error $\mathbf{e}^t$ between the optimal tracker parameter values $\mathcal{P}^t$ (obtained using ground-truth data) and predicted tracker parameter values $\hat{\mathbf{P}}^t$ (computed by equation 1) is minimized.

Using this error value, the meta-cognitive learning algorithm employs a set of strategies, viz., sample deletion strategy, sample learning strategy and sample reserve strategy, to adapt/evolve the fuzzy control rules. The sample deletion strategy deletes the current input/output ($\mathbf{C}^t, \mathcal{P}^t$) without being learned, if the prediction error $\mathbf{e}^t$ is significantly low $(\mathbf{e}^t < E_d)$. The sample learning strategy results in either a new rule being evolved or the parameters of the existing rules being adapted.

A new rule is added if prediction error is significantly high $(\mathbf{e}^t > E_a)$ and the existing rules do not sufficiently cover the current sample $(\psi^t < E_S)$. Here, ψ^t is a measure of rule coverage given by:

$$\psi^t = \frac{\sum_{r=1}^{R} \phi_r}{R} \tag{3}$$

During rule evolution, the $(R+1)^{th}$ rule is added as

$$\boldsymbol{\mu}_{R+1} = \mathbf{C}^t \tag{4}$$

$$\sigma_{R+1} = \kappa \|\mathbf{C}^t - \boldsymbol{\mu}_{nr}\| \tag{5}$$

$$\boldsymbol{\alpha}_{R+1} = \mathbf{P}^t - \frac{\sum_{r=1}^{R} \phi_r \boldsymbol{\alpha}_r}{1 + \sum_{p=1}^{R} \phi_p} \tag{6}$$

where nr denotes the nearest rule to the context feature and κ is a predefined rule overlap factor (chosen in interval [0.5, 0.9]). The parameters of existing rules are updated using an extended Kalman filtering scheme if prediction error is higher $(\mathbf{e}^t > E_d)$.

3 Performance Evaluation

In the previous sections, we have presented McFIS based controller for object tracking. In this section, we evaluate the performance of the proposed controller. The parameters E_a, E_s and E_d of the proposed approach are fixed for all following experiments.

We select a baseline tracker using different object appearance descriptors [4] to experiment the proposed controller. We present the tracking results in three cases: baseline tracker (with fix parameters) [4], baseline tracker with a parameter tuner based on code-book [7], and baseline tracker with the controller proposed in this paper. The performance of the tracker with the proposed controller is compared against other state-of-the-art tracking algorithms. The performance comparison shows improved performance of a tracker with the proposed controller over state-of-the-art trackers.

3.1 Appearance based Tracker

We employ the appearance-based tracker proposed in [4] to study the proposed control mechanism. The principle of this tracker is similar to many different appearance-based trackers in state of the art. This tracker relies on the computation of object similarity across different frames using different object appearance descriptors (e.g. 2D area, color histogram, color covariance). Since the object descriptor reliability is influenced by context, the descriptor weights w_k need to be set and tuned along changes in context. The approach [4] proposes a scheme to learn offline the values of these weights but cannot adjust them online.

In this work, we aim to predict the values of these weights all along the current video. We have selected the weights of the five following descriptors for tuning as they have a significant influence to tracker performance: 2D shape ratio, 2D area, color histogram, color covariance and dominant color. Therefore the set of parameters w_k corresponding to these descriptors represent the control parameters $\mathbf{P}$ in section 2.

3.2 Training Phase

Initially, the proposed controller is trained on nine video sequences: six videos from CAVIAR dataset[1] and three from ETISEO dataset[2]. The videos are selected such that they represent a large of tracking contextual information (e.g.. low/high density of objects in the scene, strong/weak object contrast). The video chunk size v for controller (Section 2) is set to 50 frames. A training sample is the value set of six contextual features over a video chunk of 50 frames. The offline training phase requires the ground-truth of object tracking as input. The best descriptor weight values are found using grid-search technique, such that the F1-score is maximized.

At the end of the training phase, 91 rules are created after training 260 samples. This shows that the proposed learning scheme is convergent. Using these learned rules, we can define a mapping function to link the contextual features of the given 50 frames to the best tracking descriptor weights to maximize the tracking performance on the next video chunk.

3.3 Online Control Phase

The proposed controller is evaluated on two public datasets (PETS 2009 and TUD), and the third one from Vanaheim European project (recorded in a subway station). For all these videos, the observed scenes are different from the ones of training videos. As a

[1] homepages.inf.ed.ac.uk/rbf/CAVIAR/

[2] www-sop.inria.fr/orion/ETISEO

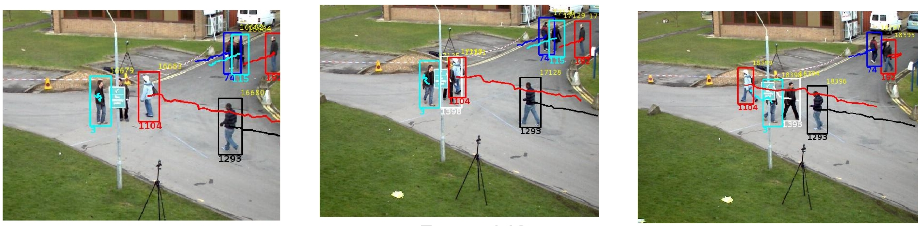
new frame is presented to the controller, it extracts the six contextual values from HOG-based detector output. Upon concatenating the contextual features over 50 frames, the controller adapts the tracking descriptor weights to the change in contextual information using the mapping function.

PETS dataset 2009 In this test, the sequence S2_L1, camera view 1, time 12:34 is selected for testing because this sequence is used for evaluation in several state of the art trackers. It consists of 794 frames with 21 mobile objects with different degrees of inter-person and person-object occlusion. The performance of the tracker is compared with respect to two metrics defined in [14]: Multi-object tracking precision (MOTP) and multi-object tracking accuracy (MOTA). Illustrations of tracking performance for frames 146, 149 and 158 are shown in Fig. 2.

The controller adapts the tracking parameters from frame 100 to frame 200 as follows: $w_1 = 0$ (2D shape ratio weight), $w_2 = 0.15$ (2D area weight), $w_3 = 0.3$ (color histogram weight), $w_4 = 0.4$ (color covariance weight) and $w_5 = 0.15$ (dominant color weight). This parameter tuning is reasonable. The color covariance descriptor can handle occlusion, so its weights is more important than the others. The color histogram performance is better than the dominant color when object resolution is small. The 2D shape ratio descriptor (defined as the ratio between the 2D width and 2D height) is not used as most of persons in the scene have quite similar shape ratio. Results show that all person trajectories are correct, in particular the person of ID 1104 (red trajectory) even when this person is heavily occluded in continuous frames.

Table 1 gives the performance analysis on PETS video compared with other state-of-the-art trackers. The result of [12] is provided by [2]. In order to highlight the advantage of the proposed controller, the performance is also compared against the employed tracker [4] without the proposed controller and with a code-book controller [7]. It could be observed that the tracker performance of [4] is improved significantly thanks to the proposed McFIS controller. The MOTA value increases from 0.62 to 0.86, and the MOTP value increases from 0.63 to 0.73.

The obtained tracking performance also outperforms all the other considered trackers, except [2, 1]. However, these two approaches are fully offline. In [2], the tracker requires the whole video for maximizing their results. In [1], the approach is optimized



Frame 146 Frame 149 Frame 158

Fig. 2. Tracking results for persons in PETS data set. The tracking performance is correct even in the presence of high static and dynamic occlusion.

Approaches	#Ground truth	MOTA	MOTP
Amir et al. [2] **(offline tracker)**	21	*0.90*	*0.69*
Izadinia et al. [1] **(offline tracker)**	21	*0.90*	*0.76*
Berclaz et al. [15]	21	0.80	0.58
Shitrit et al. [12]	21	0.81	0.58
Chau et al., 2011 [4]	21	0.62	0.63
Chau et al., 2013 [7] ([4] with code-book based controller)	21	0.85	0.71
Proposed approach ([4] with McFIS based controller)	21	**0.86**	**0.73**

Table 1. Performance comparison for PETS 2009 video. The best values are printed in **red** (not taking into account the two offline approaches: [2] and [1]).

iteratively, rendering it offline. The proposed approach is fully online, wherein a frame is processed by the tracker one and only once. This makes the controller ideal for any real-time situations. Moreover, the controller achieves this precision based on training on other video sequences.

TUD-Stadtmitte The second test is conducted with the the TUD-Stadtmitte sequence. This video is very challenging due to heavy and frequent object occlusions. Figure 3 shows a snapshot of the tracking performance of the proposed Tracker-Controller pair.

Table 2 presents the tracking results of the proposed approach and three recent trackers from the state of the art. In this experiment, the following tracking evaluation metrics are used to easily compare with other approaches. Let GT be the number of trajectories in the ground-truth of the test video. The first metric MT (mostly tracked) computes the percentage of trajectories correctly tracked for more than 80% of length. This metric represents the true positive value. The second metric PT (partially tracked) computes the percentage of trajectories correctly tracked between 20% and 80% length. The last metric ML (mostly lost) is the percentage of the remaining trajectories. The ML metric represents the false negative value.

For this video, the proposed controller helps to increase the MT value from 50% to 70% and to decrease the ML value from 20% to 0%. Compared to [7], we obtain the same MT value (always 70%), but much lower ML value (0% compared to 20%). We obtain the best MT and ML values compared to the other trackers.

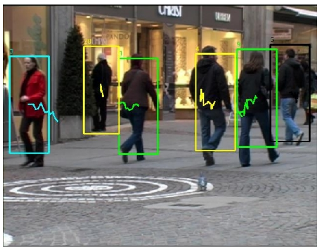


Fig. 3. Illustration of object tracking for the TUD-Stadtmitte video

Methods	GT	$MT(\%)$	$PT(\%)$	$ML(\%)$
Kuo et al. [13]	10	60	30.0	10.0
Andriyenko et al. [16]	10	60.0	30.0	10.0
Chau et al., 2011 [4]	10	50.0	30.0	20.0
Chau et al., 2013 [7] ([4] with code-book based controller)	10	**70.0**	10.0	20.0
[4] with McFIS based controller	10	**70.0**	**30.0**	**0.0**

Table 2. Tracking results for the TUD-Stadtmitte sequence. The best values are printed in **red**.

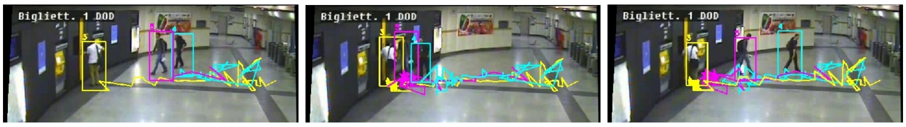


Fig. 4. Tracking of the motion of 3 persons in the subway video over time. The tracking is correct even with low person resolutions and low person contrast (in second image).

Subway video The video of the third test belongs to an European project (anonymity). The test sequence contains 36006 frames and lasts 2 hours. Figure 4 illustrates the correct tracking of three persons with low resolutions over time. In the second image, the contrast with respect to the surrounding background of the two persons with tracker ID 4 and 5 (corresponding to cyan and pink trajectories) are very low. They occlude each other at several frames but they are still correctly tracked (see the right image).

Table 3 presents the performance of the proposed approach and three recent trackers from state of the art. For this sequence, the proposed controller improves the performance of the tracker [4]. The MT value increases from 55.26% to 65.79%. The ML value decreases significantly from 13.16% to 2.63%. The tracking result with the proposed controller gets the best quality among the trackers presented in table 3.

4 Conclusion

An online generic, efficient and robust controller, based on meta-cognitive neuro-fuzzy inference system, for an appearance-based tracker is proposed. It monitors the scene context over a chunk of frames to compute the satisfactory tracker parameter values for

Approaches	GT	$MT(\%)$	$PT(\%)$	$ML(\%)$
Souded et al. [10]	38	44.74	42.11	13.15
Chau et al. 2011 [4]	38	55.26	31.58	13.16
Chau et al., 2013 [7] ([4] with code-book based controller)	38	60.53	36.84	**2.63**
[4] with the McFIS based controller	**38**	**65.79**	**31.58**	**2.63**

Table 3. Tracking results on the subway video. The proposed controller improves significantly the tracking performance. The best values are printed in **red**.

the next chunk of frames. The use of meta-cognitive learning strategies for the controller improves its generalization over unseen test data, in addition to reducing training time. The performance comparison on three untrained video sequences clearly highlights the tracking improvement with the proposed controller. In the future work, we will propose a method to determine automatically the most appropriate context scene features to characterize more accurately videos depending on the selected tracker.

Acknowledgments

This work is supported by The Panorama and Centaur European projects as well as Movement French projects.

References

1. Izadinia, H., Saleemi, I., Li, W. and Shah, M.: $(MP)^2T$: Multiple People Multiple Parts Tracker. In ECCV (2012)
2. Zamir, A.R., Dehghan, A. and Shah, M.: GMCP-Tracker: Global Multi-object Tracking Using Generalized Minimum Clique Graphs. In ECCV (2012)
3. Kuo, C.H., Huang, C. and Nevatia, R.: Multi-target tracking by online learned discriminative appearance models. In CVPR (2010)
4. Chau, D.P., Bremond, F., Thonnat, M.: A multi-feature tracking algorithm enabling adaptation to context variations. In: ICDP (2011)
5. Hall, D.: Automatic parameter regulation of perceptual system. In: J. of Image and Vision Computing, vol 24, 870 – 881 (2006)
6. Yoon, J.H., Kim, D.Y., Yoon, K.J.: Visual Tracking via Adaptive Tracker Selection with Multiple Features. In: ECCV (2012)
7. Chau, D.P. and Badie, J. and Bremond, F. and Thonnat, M.: Online Tracking Parameter Adaptation based on Evaluation. In: AVSS (2013)
8. Drucker, H. and Durges, C.J., Kaufman, L., Smola, A. and Vapnik, V.: Support vector regression machines. In book: Advances in neural information processing systems, vol 9 (1997)
9. Subramanian, K. and Suresh, S.: A meta-cognitive sequential learning algorithm for neuro-fuzzy inference system. In: J. of Applied Soft Computing, vol. 12, 3603 – 3614 (2012)
10. Souded, M. ,Giulieri, L., and Bremond, F.: An Object Tracking in Particle Filtering and Data Association Framework, using SIFT Features. In: ICDP (2011)
11. Psaltis, D., Sideris, A. and Yamamura, A.: A multilayered neural network controller. In: IEEE Control Systems Magazine, vol. 8, 17 – 21 (1988)
12. Shitrit, J., Berclaz, J., Fleuret, F., Fua, P.: Tracking multiple people under global appearance constraints. In: ICCV (2011)
13. Kuo, C. and Nevatia, R.: How does person identity recognition help multi-person tracking?. In CVPR (2011)
14. Kasturi, R., Soundararajan, P., Garofolo, J., Bowers, R. and Korzhova, V.: How does person identity recognition help multi-person tracking? In CVPR (2011)
15. Berclaz, J., Fleuret, F., Turetken, E., Fua, P.: Multiple object tracking using k-shortest paths optimization. In: TPAMI, vol 33, 1806–1819 (2011)
16. Andriyenko, A. and Schindler, K.: Multi-target tracking by continuous energy minimization. In CVPR (2011)
17. Chen, S., Fern, A. and Todorovic, S.: Multi-Object Tracking via Constrained Sequential Labeling. In CVPR (2014)