Electronics Letters 51.3 (2015): 226 – 228

DOI: http://dx.doi.org/10.1049/el.2014.3405

Covariance-based online validation of video tracking

Juan C. SanMiguel and A. Calvo

A novel approach is proposed for online evaluation of video tracking without ground-truth data. The temporal evolution of covariance features is exploited to detect the stability of the tracker output over time. A model validation strategy performs such detection without learning the failure cases of the tracker under evaluation. Then, the tracker performance is estimated by a finite state machine determining whether the tracker is on target (successful) or not (unsuccessful). The experimental results over a heterogeneous dataset show that the proposed approach outperforms related state-of-the-art in terms of performance and computational cost.

Introduction: Video tracking approaches (trackers) are widely used in many multimedia applications although the existing visual challenges are not simultaneously addressed by current trackers [1]. Online tracker validation is therefore needed to select the best tracker for each application or to improve tracking performance via self-tuning. Such validation is complex as only the tracker result is available at runtime and ground-truth data (ideal result) can not be used. Current approaches for online validation are based on multi-hypothesis trackers, target motion reversibility and feature reliability. *Multi-hypothesis* approaches measure the dispersion of hypothesis in the state-space and, albeit effective, they require specific architectures such as Particle Filters (PFs) [2]. *Reversibility*-based approaches apply tracking in reverse direction to check similarities between the tracker to validate and the reverse one [3]. However, reverse tracking has high computational cost which depends on the video sequence length. *Feature*-based approaches estimate the reliability of features extracted from the tracker output such as the change of target size [4]. They often use standard tracker outputs (e.g. bounding box) so they can be applied to many trackers. Feature validation can be also cast as a classification problem between successful (on-target) and unsuccessful (off-target) tracker cases [5], often solved by the maximum likelihood criterion. Feature-based approaches usually have low performance as the training data availability is limited for the unsuccessful case and feature values are similar for both cases due to wrong target model updates or distractors (objects similar to the target). Selecting an optimal feature and classification strategy are key for efficient online validation in terms of performance and computational cost.

To overcome the above-mentioned problems, this Letter presents an approach for online evaluation of single-object trackers without ground-truth data. It focuses on the temporal evolution of covariance features only requiring a bounding box as tracker output. Unlike previous work assuming prior knowledge on the unsuccessful tracker case, the proposed approach only models the successful case and presents a model acceptance strategy to identify model deviations. Then, a two-state machine uses the detected deviations to determine the successful tracker results.

Proposed approach: An overview of the proposed approach is shown in Fig. 1. It starts from the target location estimated by the tracker at time t:

$$\mathbf{x_t} = [x_t, y_t, w_t, y_t, o_t], \tag{1}$$

where the tracker output (bounding box) is described by its center location (x_t, y_t), width w_t, height h_t and orientation o_t. The proposed approach can be used for most of existing trackers as they fit Eq. 1. Then, we measure the structure of the target appearance using $\mathbf{x_t}$ and the covariance feature:

$$\Sigma_t = \frac{1}{N} \sum_{r \in \mathcal{R}_t} (\phi^{\mathbf{r}}_{\mathbf{t}} - \bar{\phi}_{\mathbf{t}})(\phi^{\mathbf{r}}_{\mathbf{t}} - \bar{\phi}_{\mathbf{t}})^T \tag{2}$$

where $\mathcal{R}_t$ is the set of N pixels contained in $\mathbf{x_t}$; r is the pixel index; $\phi^{\mathbf{r}}_{\mathbf{t}}$ is a vector describing the r^{th} pixel by its location (x^r_t, y^r_t) and RGB values (R^r_t, G^r_t, B^r_t); and $\bar{\phi}_t = \frac{1}{N} \sum_{r \in \mathcal{R}_t} \phi^{\mathbf{r}}_{\mathbf{t}}$ is the mean descriptor value. The covariance feature allows to represent any tracker output with a low-dimensional 5×5 matrix. Furthermore, it provides a robust descriptor to match regions across different target changes such as appearance or pose.

We assume short-term stability of target features over time and we exploit the temporal evolution of the covariance feature to determine whether the tracker is on target. First, we use the proposal of [6] to compute the distance between covariance features in consecutive time-steps:

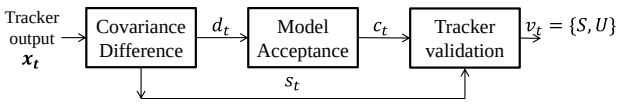

Fig. 1 *Block diagram of the proposed approach. For each time-step, the tracker output $\mathbf{x_t}$ is validated as successful (S) or unsuccessful (U).*

$$d_t(\Sigma_t, \Sigma_{t-1}) = \sqrt{\sum_{i=1}^{5} ln^2 \lambda_i(\Sigma_t, \Sigma_{t-1})} \tag{3}$$

where $\lambda_i(\Sigma_t, \Sigma_{t-1})$ are the Eigenvalues obtained by solving the problem $|\lambda\Sigma_t - \Sigma_{t-1}| = 0$. Therefore, $d_t(\Sigma_t, \Sigma_{t-1}) \geq 0$ where values close to 0 indicate similar covariance features.

We propose to detect dissimilar covariance features over time via a model acceptance strategy. We consider a model D to define the variability of d_t during successful tracker operation, which follows the probability density function (pdf) $p(d_t)$. We perfom hypothesis testing for model acceptation where the null hypothesis H_0 indicates that the covariance change d_t is consistent with the model D. Let H_1 be the hypothesis that an unknown change of d_t has occurred. Model acceptance is formulated as:

$$c_t = \begin{cases} H_0 & if \quad p(d_t) < \tau \\ H_1 & otherwise \end{cases} \tag{4}$$

where c_t indicates is covariance difference is consistent with the model D and τ is a threshold defining the tolerance to deviations from the model. Note that c_t is not sufficient to determine the successful tracker operation and additional reasoning is needed. For example, the tracker output may remain locked on a background region after a target loss, thus having low d_t values (i.e. H_0 hypothesis) albeit the tracker is not on target anymore.

We employ a finite state machine to validate the tracker operation (see Fig. 2) where two states are defined for the successful (S) and unsuccessful (U) cases. Starting from the S state, the $S \rightarrow U$ transition is triggered when the H_1 hypothesis is detected due to tracker failures (target loss). The $U \rightarrow S$ transition is when the tracker recovers to the correct target after a failure. It is activated when H_1 hypothesis is accepted and the new tracker output is similar to the previously tracked target. Inspired by [7], we compute the similarity between the last successful output and the new tracker output:

$$s_t = \begin{cases} 1 & if \quad d_t(\Sigma_{t_{ref}}, \Sigma_t) < \beta \\ 0 & otherwise \end{cases} \tag{5}$$

where t_{ref} is the last time-step for the successful tracker case and β is a threshold to accept the similarity between both covariance descriptors. Due to the use of covariance features, the check of Eq. 5 allows variations in scale, pose and appearance of the target between t and t_{ref}.

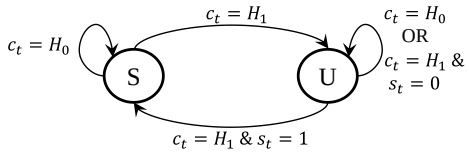

Fig. 2 *Finite state machine to validate the tracker output using two states: successful (S) and unsuccessful (U). c_t and s_t are variables for model acceptance and tracker recovery checks, respectively.*

Experimental data: We use the SOVTds dataset [8] for evaluation, which contains 126 sequences (~23000 annotated frames) covering common problems in visual tracking such as occlusions, similar objects and appearance changes. The 126 sequences are grouped in three random sets for training the feature model D (pdf $p(d_t)$) (76), for choosing the optimal value for τ in Eq. 4 (25) and for testing the proposed approach (25). We validate the results of six trackers: Mean-Shift [9], Color-based Particle Filter [10], Incremental Visual Tracking [11], Tracking-Learning-Detection [12], SuperPixel Tracking [13] and Locally Orderless Tracking [14]. The code of the original authors is used to analyze the dataset and get the tracker results for validation (~138000 in total). We heuristically set the parameter to check tracker recovery (Eq. 5) as $\beta = 2.3$.

Performance metrics: We measure the model acceptance performance by using standard measures of Precision (P), Recall (R) and F-score (F) as in [2]. We also assess the tracker validation accuracy via the *True Positive Rate* (TPR) and *False Positive Rate* (FPR) [1] that account for the number of correct and wrong tracker validations.

Experimental results: Table 1 compares common features in video tracking against the covariance feature, all applied within the proposed approach. For each feature, $p(d_t)$ is modeled as the best fitting of popular distributions using the Kolmogorov-Smirnov statistic over the training set. The results show low performance for features based on contour (shape and area), motion (speed and direction) and color (gray, RGB histograms and texture) information, demonstrating their low discriminative power between the successful and unsuccessful cases. Structure-based features (HOG, CLD and Covariance) present the best results showing that the target appearance structure exhibits short-term stability. Fig. 3 shows an example of the proposed approach where the three tracker errors (frames 90, 131-164 and 195-214) are correctly identified.

Table 1: Performance (mean results) of the proposed approach using common features for video tracking. Bold indicates best results.

Feature employed	Fitted pdf for $p(d_t)$	Model acceptance			Tracker validation		
		P	R	F	TPR	FPR	AUC
Shape ratio [4]	Beta	.107	.177	.099	.929	.587	.672
Area ratio [5]	Beta	.159	.397	.187	.905	.412	.747
Direction smoothness [5]	Normal	.077	.241	.100	.913	.451	.726
Speed smoothness [5]	Rayleigh	.039	.422	.069	.885	.429	.729
Texture difference [5]	Gamma	.069	.164	.089	.967	.734	.617
Gray level [5]	Gamma	.253	.150	.081	**.968**	.834	.568
Color hist. (RGB) [10]	Exponential	.571	.166	.150	.967	.831	.568
Gradient hist. (HOG) [1]	Exponential	.297	.367	.309	.958	.518	.720
Color layout (CLD) [15]	Exponential	.415	.363	.349	.937	.629	.754
Covariance (Proposed)	Exponential	**.462**	**.549**	**.489**	.935	**.359**	**.788**

Table 2 compares the results of the proposed approach against the related state-of-the-art in terms of accuracy and computational cost. For feature-based approaches, the proposal clearly improves the accuracy of [5] (and its modification using the best feature), showing the benefits of model validation over a two-model Bayesian classifier for successful and unsuccessful cases. Moreover, the computational cost is reduced as only covariance feature is employed instead of multiple features in [5]. Compared to reverse validation [3], the proposed approach reduces the computation cost around $50\times$ as compared to [3]. Moreover, the computations of [3] depend on the sequence length whereas the proposed approach has a bounded computation. This limitation of [3] prevents its use for many applications where execution time is critical and for long sequences as the computational cost is not affordable. Therefore, the proposed approach allows a broader application of online validation as compared to [3], offering a trade-off between accuracy and cost. Note that we do not compare with [2] as it is for PF-based approaches and [4] as it uses low-performing features (motion speed and smoothness, see Table 1).

Table 2: Comparative results (mean) for online tracker validation. The symbol '*' is for [5] using only the best feature. $\triangle\%$ shows the difference (in percentage) between the proposed and each selected approach.

Reference	Type	Tracker validation				Execution time (ms/frame)		
		TPR	FPR	AUC	$\triangle\%$	Train	Test	$\triangle\%$
[5]	Feature	.941	.773	.584	+34.7	4578	4230	-87.2
[5]*	Feature	.940	.739	.601	+31.1	4299	3970	-86.3
[3]	Reversibility	.931	.185	.886	-11.1	-	26681	-97.7
Proposed	Feature	.935	.359	.788	-	567	542	-

Conclusions: An approach to validate trackers is presented in this Letter based on short-term evolution of covariance features. The results show that focusing on temporal consistency of features is more effective than the traditional two-model classification. Moreover, the structure of target appearance (covariance) performs better than common features to determine tracker errors. Finally, the proposed approach outperforms competitive feature-based approaches and provides a generic cost-bounded validation that can be applied for long-term and time-critical applications.

Acknowledgment: This work was supported by the Spanish Government (TEC2011-25995, EventVideo).

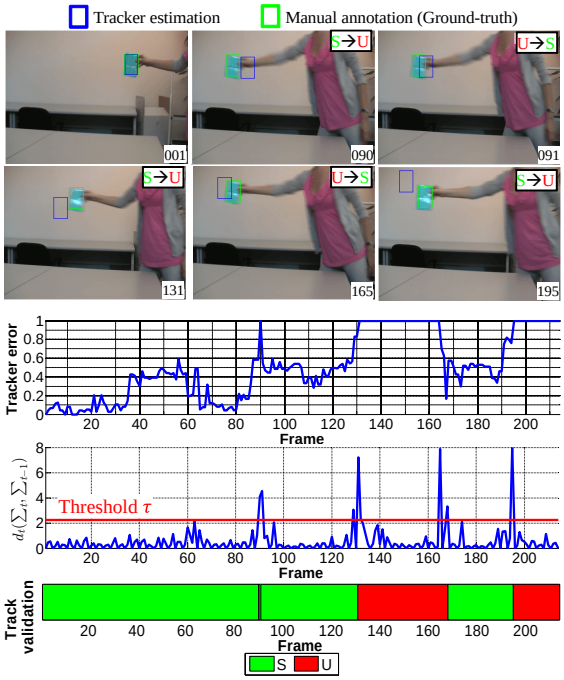

Fig. 3 *Example for online validation of tracking results between successful (S) and unsuccessful (U) for the Mean-Shift (MS) tracker. From top to bottom graphs: error as the spatial overlap between the estimation and ground-truth data [3], covariance difference d_t (Eq. 1) and final tracker validation.*

Juan C. SanMiguel and A. Calvo (*Video Processing and Understanding Lab, Universidad Autónoma de Madrid, Madrid, Spain*)

E-mail: juancarlos.sanmiguel@uam.es

References

1 Smeulders, A., Chu, D., Cucchiara, R., Calderara, S., Dehghan, A., and Shah, M.: 'Visual Tracking: An Experimental Survey', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2014, **36**, (7), pp.1442-1468

2 SanMiguel, J., and Cavallaro, A.: 'Temporal validation of Particle Filters for video tracking', *Comput. Vis. Image Understand.*, 2014, http://dx.doi.org/10.1016/j.cviu.2014.06.016 (last accessed Oct. 2014)

3 Hao, W., Sankaranarayanan, A., and Chellappa, R.: 'Online Empirical Evaluation of Tracking Algorithms', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2010, **32**, (8), pp.1443-1458

4 Chau, D., Thonnat, M., Brémond, F., and Corvée, E.: 'Online parameter tuning for object tracking algorithms', *Image Vis. Comput.*, 2014, **32**, (4), pp. 287-302.

5 Spampinato, C., Palazzo, S., and Giordano, D.: 'Evaluation of tracking algorithm performance without ground-truth data', *Proc. of IEEE Conf. on Image Process.*, Orlando (USA), Oct. 2012, pp.1345-1348

6 Förstner, W., and Moonen, B.: 'A metric for covariance matrices', *Geodesy-The Challenge of the 3rd Millennium*, 2003, pp. 299-309

7 Matthews, I., Ishikawa, T., and Baker, S.: 'The template update problem', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2004, **26**, (6), pp. 810-815

8 VPULab datasets: 'SOVTds: A single-object video tracking dataset', http://www-vpu.eps.uam.es/SOVTds/ (last accessed Oct. 2014)

9 Comaniciu, D., Ramesh, V., and Meer, P.: 'Kernel-based object tracking', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2003, **25**, (5),pp. 564-577

10 Nummiaro, K., Koller-Meier, E., and Gool, L.V.: 'An adaptive colour-based particle filter', *Image Vis. Comput.*, 2002, **21**, (1), pp. 99-110

11 Ross, D.A., Lim, J., Lin, R.S., and Yang, M.H.: 'Incremental learning for robust visual tracking', *Int. J. Comput. Vis.*, 2008, **77**, 1, pp. 125-141

12 Kalal, Z., Mikolajczyk, K., and Matas, J.: 'Tracking-learning-detection', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2011, **34**, (7),pp. 1409-1422.

13 Fan, Y., Huchuan, L., and Ming-Hsuan, Y.: 'Robust Superpixel Tracking', *IEEE Trans. Image Process.*, 2014, **23**, 4, pp.1639-1651

14 Oron, S., Bar, A., Levi, D., and Avidan, S.: 'Locally Orderless Tracking', *Int. J. Comput. Vis.*, 2014, http://dx.doi.org/10.1007/s11263-014-0740-6 (last accessed Oct. 2014)

15 Manjunath, B., Ohm, J., Vasudevan, V., and Yamada, A.: 'Color and texture descriptors', *IEEE Trans. Circ. Syst. Video Technol.*, 2001, **11**, 6, pp.703-715