Single Object Long-term Tracker for Smart Control of a PTZ camera

Antonio González, Rafael Martín-Nieto, Jesús Bescós, José M. Martínez
Video Processing and Understanding Lab (VPULab), Escuela Politécnica Superior, Universidad Autónoma de Madrid
C/ Francisco Tomás y Valiente, 11,28049 Madrid (Spain)
antonio.gonzalezh@titulado.uam.es,rafael.martin@uam.es,j.bescos@uam.es,josem.martinez@uam.es

ABSTRACT

In this paper, we present a single-object long-term tracker that supports high appearance changes in the tracked target, occlusions, and is also capable of recovering a target lost during the tracking process. The initial motivation was real time automatic speaker tracking by a static camera in order to control a PTZ camera capturing a lecture. The algorithm consists of a novel combination of state-of-the-art techniques. Subjective evaluation, over existing and newly recorded sequences, shows that the tracker is able to overcome the problems and difficulties of long-term tracking in a real lecture. Additionally, in order to further assess the performance of the proposed approach, a comparative evaluation over the VOT2013 dataset is presented.

Categories and Subject Descriptors

I.4.8 [**Image Processing and Computer Vision**]: Scene Analysis *–tracking*

General Terms

Algorithms, Performance, Design, Experimentation.

Keywords

Object tracking, feature points, template matching, lectures production, PTZ control.

1. INTRODUCTION

Lecture production is an important task in the field of online teaching, conference streaming, etc. High quality lecture capturing depends on human camera operators, thus resulting in expensive solutions. Automatic capturing dramatically reduces the cost of these online initiatives.

The original motivation behind the work presented in this paper is to simulate the behavior of a human camera operator capturing a lecture for online transmission. The setup consists of a PTZ camera, which is the camera whose signal is transmitted; and a fixed wide-angle and high-resolution camera whose output is analyzed to track the speaker position and to consequently command the PTZ camera to follow her/him as if it were a human operator. As both cameras are closely anchored and the target object (see Fig. 1), the speaker, is considered to be quite far from them, a simple homography is obtained to relate their positions. This set up could be adapted to other scenarios (e.g., surveillance, tv reporters), where close-up monitoring of selected targets by a PTZ camera that can be controlled by mid-range tracking of a general view static camera is a solution.

This paper focuses on the development of an object tracker designed for real-time long-term automatic speaker tracking with the fixed camera. Additionally, thanks to a simple frame stabilization module, the proposed tracker shows its capability to operate in more generic situations, like scenarios where the fixed camera cannot be guaranteed to be fully stable (e.g., mounted on a pole), hence improving the results obtained by the state of the art trackers on such sequences.

After this introduction, some references from the state of the art are described. Then, the video object tracking algorithm is presented and each one of its modules is explained. Afterwards, the selected evaluation framework is presented. The experiments and results are described after that. Main conclusions and future work conclude the paper.

2. RELATED WORK

2.1 Tracking for lecture capturing

In the work presented by Rui et al. [1], the design of a complete system that automatically captures and broadcasts lectures is reported. They also describe how the system can be generalized to a variety of lecture room environments differing in room size and number of cameras. The goal is to share their experience building the system with the practitioners in the field to facilitate the construction of similar systems, and to identify unsolved problems requiring further research.

Similar to the previous system is the work presented by Zhang et al. [2]. This system also considers two cameras (one for the lecturer and one for the audience). This work presents a complete automated end-to-end system that supports capturing, broadcasting, viewing, archiving and searching of presentations. They describe a system architecture that minimizes the pre- and post-production time, and a fully automated lecture capturing system. Zhang et al. [3] also present another automated lecture capturing system scheme based only on a single PTZ camera. This system has certain limitations: it is difficult to port the system to another lecture room; analog cameras do not only require a lot of wiring work, but also need multiple computers to digitize and process the captured videos. During the system setup stage, a detection region and a screen region should be manually specified.

Chou et al. [4] propose an automatic lecture recording system based on a PTZ camera shooting in a lecture. In the system, the PTZ camera is controlled to make the recording video similar to one shot by a real cameraman. In this case, the camera should be mounted on the central back hall and at the eye level of the audience. This system includes lecturer detection and screen detection, which are applied to locate the position of the lecturer

ICDSC '14, November 04 - 07 2014, Venezia Mestre, Italy
Copyright 2014 ACM 978-1-4503-2925-5/14/11 ...$15.00.
http://dx.doi.org/10.1145/2659021.2659043

and of the screen. For the lecturer tracking, the Mean Shift method is used with two features: color and edge orientation histogram. One main restriction of this method is that assumes that there is only one person standing in the front of the lecture room, the lecturer.

ClassX [5] is an interactive online system developed at Stanford University by Pang et al. In this work they propose a new learning algorithm to automatically generate a professional virtual camera view by learning the behavior of a human camera operator, based on the consideration that a tracking-based camera view does not mimic a human operator naturally. In the project, a simple virtual camera is considered with two degrees of freedom: horizontal position and a bi-level zoom level (for simplicity). This system obtains an input video recorded from a static HD camera, as well as the locations of the writing boards. For lecturer tracking, conventional background subtraction and template matching techniques are used to detect, whilst the position and velocity of the lecturer are the features used for prediction.

Another work on this topic is described by Yokoi et al. [6], where a method for generating a dynamic lecture video from the high resolution images recorded by a HDV camcorder is proposed. The lecture video is generated by cropping the recorded high resolution images. The method used for detecting the positions to crop from the high resolution images are calculated with frame temporal differencing.

A general weakness of these systems is the absence of a real-time tracking algorithm able to operate in a long-term fashion, which is required to automate the event capturing process. This paper focuses on this specific aspect, and the next section presents a brief state-of-the-art on this topic.

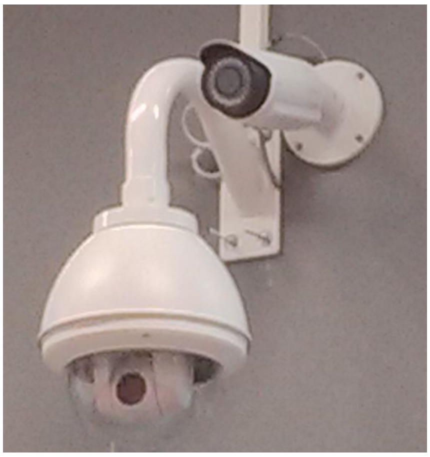

Figure 1. Cameras positioning. The cameras are placed 5 metres over the floor level

2.2 Tracking using fixed cameras

This section presents a brief review of recent schemes similar to those applied in the proposed algorithm; exhaustive reviews of state-of-the-art object tracking can be found in [7].

In the object tracking field, Varcheie et al. describe some solutions [8][9][10]to the tracking problem: a KLT based tracking algorithm [8], a tracking algorithm using motion detection with fuzzy classifiers [9], and an adaptive color based particle filter tracking algorithm [10]. In [11], Xie et al. propose a particle filter tracking algorithm applied to the omega shape instead of based on color, using the Viola-Jones method to set the initial object position. In this field of study, Chang et al. [12] propose a Mean-Shift tracking algorithm on the HSV color space. In [13], Xie et al. expose another algorithm based on shape from the multi-part perspective, where the object is divided into pieces, using a HOG descriptor for each one

3. PROPOSED OBJECT TRACKER

3.1 Overview

Initially the image stabilization module extracts the feature points between consecutive frames and estimates a homography between them to somehow compensate possible small camera motion (see section 3.3).

After this, a two phases algorithm has been designed for the proposed single-target object tracker. The first phase uses the Kanade Lukas Tomasi approach (KLT)[14] to choose the object features (using color and motion coherence) in order to track relatively large object displacements. The second phase uses mean shift gradient descent [15] to take the bounding box to the exact position of the object model, using the object features provided by the KLT learning model. Figure 2 shows the block diagram of the algorithm.

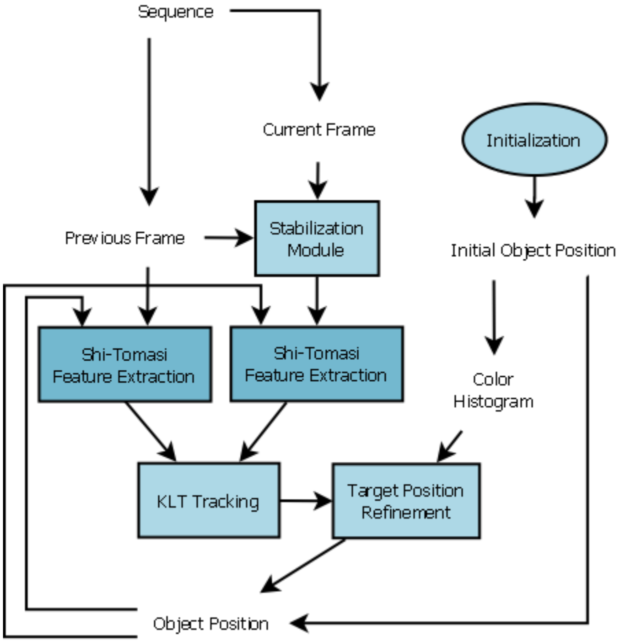

Figure 2. Algorithm modules

3.2 Initialization

Only the position and size of the object in the first frame of the video is given to this block. This position can be obtained in different ways, depending on the target application: it can be selected manually or by using an automatic object detector (e.g., a people detector) able to get the initial position of an object.

The object model is based on the RGB color and on the luminance gradient. The model consists of a one-dimensional histogram including the quantized values of the color components -16 bins per color in our experiments-, and an edge binary flag. The histogram is generated with all the pixels of this first frame located inside the object image patch. All pixels in this patch contribute with the same weight to the histogram, regardless of their position/location relative to the center of the bounding box. After that, using the CBWH method [16], the histogram is corrected reducing the effect caused by the background pixels in the initial bounding box.

3.3 Video stabilization

The first approach to the tracking algorithm was designed to be used in fixed camera scenarios. However, due to the possible vibration of *fixed* cameras, and also to extend the application domains moving cameras, a stabilization module was developed.

Camera stabilization is applied to every input frame, via fitting a set of matching feature points to an eight parameter—an homography—camera motion model. When addressing the problem of spatial correspondence for subsequent video analysis, there are several options to consider, as total stabilization or partial stabilization. For the proposed algorithm, spatial correspondence with the previous frame has considered to be enough, thus allowing elimination of motion of the moving camera and discrimination with respect to other moving objects.

The homography between two consecutive frames is estimated (via the RANSAC method) using the Shi-Tomasi features obtained for tracking (see section 3.4) and the current frame and the tracked object position (previously obtained) are corrected accordingly.

3.4 Feature points

The features used in a video object tracker should try to address the characteristics:

- Numerous: having the greatest number of features to discriminate the object
- Descriptive enough: to be located accurately, without ambiguities.
- As repeatable (between frames) as possible
- As invariant (between frames) as possible
- Low computational cost: to achieve real time applications

Given these assumptions, the best reported options include: the Harris and Stephens corner detector [17], the Shi-Tomasi corner detector [14], the fast FAST corner detector [18] and the ORB points [19]. SIFT [20] and SURF [21] have been initially discarded as they have a high computational cost and are also worse detectors for the tracking objective due to its low repeatability.

In the proposed algorithm, the objective is to quickly track as many features as possible. Considering the options, the Shi-Tomasi corner detector is conceptually optimal as it is based on how the trackers work [14]. These features present high repeatability and low computational cost, allowing the algorithm to track many features and, in the case of a sequence (consecutive frames without major changes between them), in a more robust way. The selected features are also non-parametric, avoid the spurious corner points on smooth curves and are invariant to typical image transformations.

These feature points are additionally used for image stabilization, in order to reduce the computational cost of this phase.

3.5 KLT tracking

The KLT feature tracker is originally based on the work done by Lucas-Kanade for calculating the optical flow [22], subsequently completed by Tomasi-Kanade [23], and finally presented and clarified by Shi-Tomasi [14]. This technique is based on characteristic points tracking, using the equations developed by Lucas-Kanade for calculating the optical flow, and also implements the iterative Newton-Raphson method for searching the object position. Starting with the extraction of the feature points obtained from the object patch for each frame by the method of Shi-Tomasi, this tracking method calculates the target displacement with respect to the next frame with the KLT minimizing error process (Newton-Raphson) using all the detected feature points and its displacements. A weight is defined according to the distance between each point and the center of the bounding box, so that each feature point contributes in a different way to the total displacement of the target.

This approach is only valid if the spatial displacement is small enough so that the gradient does not change its direction. In the case of a 25 fps video sequence, the changes presented between two consecutive frames are small enough, so this approximation is valid. For large displacements, this procedure should be applied in a pyramidal way [24][25].

3.6 Position refinement via template matching

The output of the KLT tracker is an estimate of the target position in the current frame. Our experiments indicated that this position might be quite noisy or imprecise in many situations; hence, we chose to refine the target position via a template matching approach. In this direction, Mean Shift [15] is one of the most used techniques to find model matches in object tracking. Here we apply it to find the maximum in the location confidence map resulting from comparing the maintained object model—based on the RGB color and on the luminance gradient—to a searching area around the KLT position estimate.

3.7 Recovery process

In sequences from practical applications all trackers get lost at some point. Furthermore, the object tracked may momentarily disappear from the scene. As the aim of this work is to design a long-term tracker, a recovery method is necessary.

If during of a certain number of frames—30 in our experiments—the algorithm does not match any characteristic point of the tracked object and the similarity between histograms is low—0.6 in our experiments—, the recovery process is activated. This process consists of: first, all feature points in the frame are obtained (from the association between two consecutive frames, as discussed above); then, a bounding box, with the size of the initial object model obtained from the first frame, is centered in each feature point, and the model histogram is calculated; finally, if the similarity between this histogram and that of the initial object model is higher than a threshold—0.1 in our experiments—the object is considered to be recovered and the tracking process continues with the object centered in that point; in case the histogram similarity is not above the threshold for any obtained point the recovery process continues in the following frame.

The computational cost of the recovery process is greater than the computational cost of the tracker when the target is located, but as the recovery process is executed just for exceptional cases, the processing time of the complete algorithm is not increased significantly.

4. EVALUATION FRAMEWORK

4.1 Datasets

4.1.1 VPULab-Lectures dataset

Some sequences have been designed and recorded in order to check the long-term performance of the designed algorithm.

These sequences are public and available online[1]. Some sample frames of these video sequences are presented in Figure 3.

There are two types of sequences: real-lecture videos and challenges videos. The real-lecture category consists of two long duration videos (30 minutes each) that have been recorded in a real lecture; they present some difficulties as target disappearing and reappearing on the scene, target appearance changes and scene illumination changes.

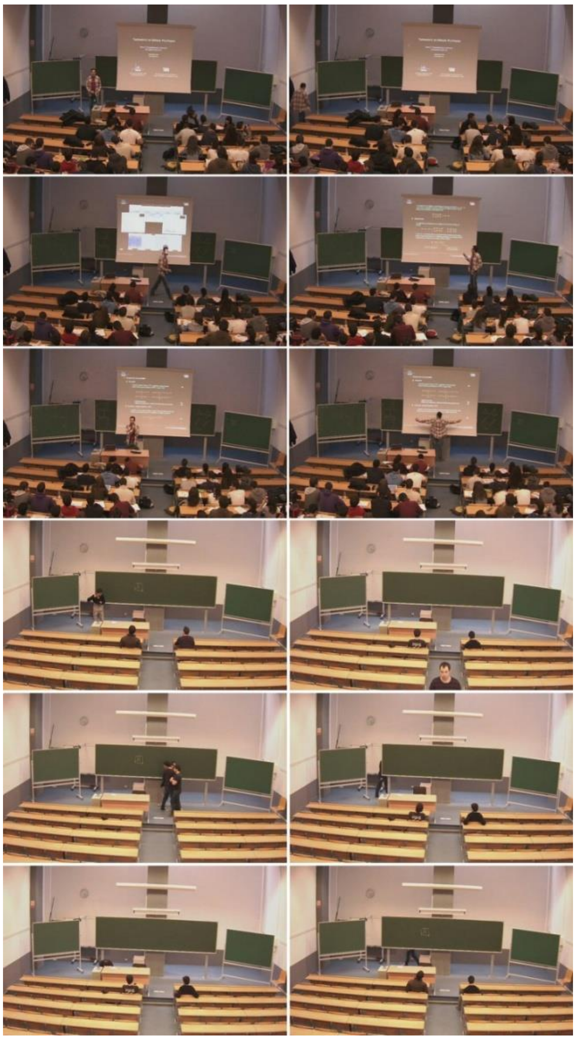

Figure 3. Example frames of the lecture recorded sequences. The six top examples correspond to the real lecture videos and the six bottom examples correspond to the challenge videos

In addition to the two long videos, 10 short different challenges videos designed to include difficult situations that a tracking algorithm might face during a lecture have been recorded with the aim of testing the algorithm in these situations. C1 presents the target taking off a sweater, which represents a high appearance change. In C2 and C3, the tracked target leaves the scene and then comes back to it. In C2, the exit and entry are produced in the same place, different than C3 in which they are produced at different locations. A significant occlusion is shown in C4, where the target moves and hides behind the blackboard coming out on the opposite end of it. In C5, the target squats (for example, to turn on a computer) behind a table. The challenge presented in C6 is similar than the one presented in C4, but in this case the target comes in on the same place where he came out. Both C7 and C8 present people and target crossing situations with people wearing similar (dark) clothes. C9 shows a situation where the target sits down and waits while two people with similar appearance go on stage sequentially. Finally C10 shows a target spinning.

4.1.2 *VOT2013 Dataset*

This is a *typical* short-sequences dataset, aimed to compare the algorithm performance with that of other state-of-the-art trackers, usually optimized for short-term situations. The content set used to generate it was provided by the VOT2013 challenge[2] trying to independently address the different problems that a tracker can face. The main criteria for dataset selection were that the dataset should represent various realistic scenes and conditions, including occlusions, illumination changes, scale changes, etc. Figure 2 (obtained from the VOT2013 challenge website) shows some frames of the different dataset videos (16 sequences summing up over 5000 frames). The VOT2013 dataset also provides the ground truth files. The associated ground-truth consists of only one target in each sequence that has been manually annotated by various authors of the dataset placing a bounding box over the object in each frame.

The evaluation protocol of VOT2013 was not used as it recovers manually the target (using ground truth) whenever it is lost, and we believe that that is not the natural functioning of these systems, especially for evaluating short sequences such as those presented in the dataset.

Figure 4. Example frames of the VOT2013 dataset.

4.2 Evaluation metric

The metrics correlation study in [26] demonstrates the high redundancy that exists among several state-of-the-art metrics (correlation above 0.9). Therefore, only the Sequence Frame Detection Accuracy (SFDA) metric [27] has been used in our evaluation. SFDA was chosen for two main reasons: its correlation with respect to other metrics is one of the highest, and it also considers and penalizes both false positives and false negatives.

[1] http://www-vpu.eps.uam.es/publications/TeacherTrackingForAutomaticLecturesProduction/

[2] http://www.votchallenge.net/vot2013/

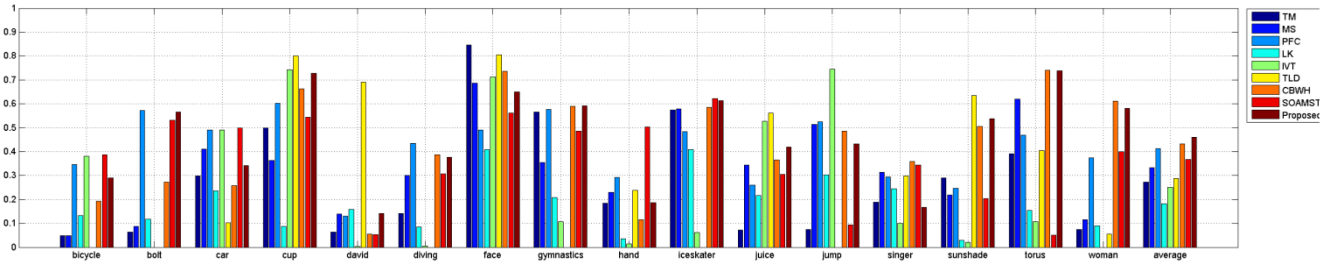

Figure 5. SFDA results

SFDA calculates in each frame the spatial overlap between the estimated target location and the ground-truth annotation. It contains information regarding the missed detections, false positives and spatial overlap. SFDA ranges from 0 to 1; the higher the value, the better.

$$SFDA = \frac{\sum_{t=1}^{N\,frames} FDA(t)}{\sum_{t=1}^{N\,frames} \exists \left(N_G^{(t)} OR\ N_D^{(t)}\right)}$$

$$FDA(t) = \frac{overlap\_ratio}{\frac{N_G^{(t)} + N_D^{(t)}}{2}}$$

$$overlap\_ratio = \sum_{i=1}^{N_{mapped}^{(t)}} \frac{\left|G_i^{(t)} \cap D_i^{(t)}\right|}{\left|G_i^{(t)} \cup D_i^{(t)}\right|}$$

where

$G_i^{(t)}$ denotes the i-th ground-truth object in frame t.

$G_i^{(t)}$ denotes the i-th ground-truth object in frame t.

$D_i^{(t)}$ denotes de i-th detected object in frame t.

$N_G^{(t)}$ and $N_D^{(t)}$ denote the number of ground truth objects and the number of detected objects in frame t, respectively.

$N\ frames$ is the number of frames in the sequence.

$N_{mapped}^{(t)}$ is the number of mapped ground truth and detected object pairs in frame t.

4.3 EXPERIMENTAL VALIDATION

4.4 Long-term qualitative validation

The sequences in the VPULab Lectures dataset (both real-lecture videos and challenges videos) have been used for the algorithm development and configuration. In the absence of ground-truth data for these sequences results have been visually generated by superimposing the obtained target bounding box to each frame. The results are available on the website mentioned in subsection 4.1.1. There are five object recovery situations (in sequences L1, C2, C3, C4 and C6) and in all of them the algorithm has operated correctly. If the object returns to the scene by the same place at which it left the scene (L1, C2, C6) the recovery is immediate. In the cases where it returns in a different location (C3, C4), the recovery takes longer, especially in the cases where changes in the target appearance occur, resulting in (at most) a few seconds.

4.5 Short-term quantitative comparative results

For this section the used metric is SFDA (see section 4.2) and the content set is the VOT2013 Dataset (see section 4.1.2).

The comparison is done against the following trackers: Template Matching (TM) [28], Mean-Shift (MS) [15], Particle Filter-based Colour tracking (PFC) [29], Lucas-Kanade tracking (LK) [30], Incremental learning for robust Visual Tracking (IVT) [31], Tracking Learning Detection tracking (TLD) [32], Corrected Background Weighted Histogram tracker (CBWH) [16] and Scale and Orientation Adaptive Mean-Shift Tracking (SOAMST) [33]. The first four tracking algorithms have been selected because they are classical and general tracking algorithms. The last four have been chosen because they are modern trackers with contrasted and remarkable results.

Figure 5 shows the SFDA score of the eight previously presented state-of-the-art trackers, and of the tracker proposed in this paper. Table 1 presents the numerical average scores of all the trackers to facilitate the comparison. The values presented in the table are those shown in the last set of bars in figure 3.

Table 1. Average SFDA scores

TM	MS	PFC	LK	IVT
0,27	0,33	0,41	0,18	0,25

TLD	CBWH	SOAMST	**Proposed**
0,29	0,43	0,37	**0,46**

The results of individual trackers present high variations depending on the evaluated sequence. The mean score for the proposed algorithm presents the best figure, slightly above the best of the state of the art tracker (CBWH).

Regarding the execution time for each sequence, table 2 shows the frames per second (fps) obtained after the execution over each of the 16 videos in the VOT dataset. The processing time is variable and depends on the frame size and on the elapsed time when an object is lost and recovered. The characteristics of the computer on which the times have been obtained are: Intel(R) Core(TM) 2 Duo @2,93GHz, 4GB RAM, Windows 7, 32 Bits.

Table 2. Execution times (fps)

bicycle	bolt	car	cup	david	diving
22,6	10,6	22,0	21,6	18,8	33,0

face	gym.	hand	ice.	jump	juice
14,8	34,5	24,4	15,6	25,3	9,9

singer	sun.	torus	woman	**average**
7,3	24,6	24,0	21,3	**20,6**

5. CONCLUSIONS

This paper presents an algorithm for long-term real-time tracking of single objects. According to the ideas presented in the introduction, using appropriate calibration between the fixed camera and a PTZ camera, this algorithm would command the PTZ camera to focus on a target during a long period (e.g. during a lecture, in order to allow for a proper lecture video production using the most adequate zoom if necessary).

The paper is focused on the description and evaluation of a novel tracking approach which uses a relatively simple but well founded combination of existing methods in the state-of-the-art (i.e., stabilization, feature point tracking, and template matching refinement) capable of tracking an arbitrary object for a long period of time and able to recover it once it has been lost during the tracking.

The experiments qualitatively show that the algorithm operates as expected in long-term sequences with challenging situations; and quantitatively demonstrate that its performance in the challenging scenarios proposed by the scientific community (VOT) is higher than state-of-the-art trackers with which it has been compared.

It is important to remark that while having obtained the best average score, there are many evident improvements applicable to the algorithm that have not yet been included, e.g.: model update, in order to cope with target changes in scale and orientation, inclusion of additional information (e.g., motion) in the object model; consideration of other objects in the scene to help solving occlusions and to avoid confusion with similar objects.

ACKNOWLEDGMENTS

This work has been partially supported by the Spanish Government (TEC2011-25995).

REFERENCES

[1] Y. Rui, A. Gupta, J. Grudin, and L. He, “Automating lecture capture and broadcast: technology and videography,” ACM Multimedia Systems Journal, 10:3-15, 2004.

[2] C. Zhang, Y. Rui, J. Crawford, and L. He, “An automated end-toend lecture capturing and broadcasting system,” in ACM Multimedia, pp. 808-809, 2005.

[3] C. Zhang, Y. Rui, L. He, and M. Wallick, “Hybrid speaker tracking in an automated lecture room,” in International Conference on Multimedia and Expo, pp. 1-4, 2005.

[4] H.P. Chou, J.M Wang, C.S. Fuh, S.C. Lin, and S.W. Chen, “Automated lecture recording system,” in Conference on System Science and Engineering, pp. 167-172, 2010.

[5] D. Pang, S. Madan, S. Kosaraju, and T. Vir Singh, “Automatic virtual camera view generation for lecture videos,” Tech. Rep., Stanford Universit, 2010.

[6] T. Yokoi and H. Fujiyoshi, “Virtual camerawork for generating lecture video from high resolution images,” in Conference on Multimedia and Expo, pp. 1-4, 2005.

[7] A.W.M. Smeulders, et al., "Visual Tracking: An Experimental Survey", IEEE Transactions on Pattern Analysis and Machine Intelligence, 36(7):1442-1468, 2014

[8] P.D.Z. Varcheie and G.A. Bilodeau, “Active people tracking by a ptz camera in ip surveillance system,” in Workshop on Robotic and Sensors Environments, pp. 98-103, 2009.

[9] P.D.Z. Varcheie and G.A. Bilodeau, “Human tracking by ip ptz camera control in the context of video surveillance” in Image Analysis and Recognition, vol. 5627 LNCS, pp. 657-667, 2009.

[10] P.D.Z. Varcheie and G.A. Bilodeau, “Adaptive fuzzy particle filter tracker for a ptz camera in an ip surveillance system,” Instrumentation and Measurement, 60(2):354-371, 2011.

[11] Y. Xie, M. Pei, G. Yu, X. Song, and Y. Jia, “Tracking pedestrians with incremental learned intensity and contour templates for ptz camera visual surveillance,” in International Conference on Multimedia and Expo, pp. 1-6, 2011.

[12] F. Chang, G. Zhang, X. Wang, and Z. Chen, “Ptz camera target tracking in large complex scenes,” in Congress on Intelligent Control and Automation, pp. 2914-2918, 2010.

[13] Y. Xie, L. Lin, and Y. Jia, “Tracking objects with adaptive feature patches for ptz camera visual surveillance,” in Conference on Pattern Recognition, pp. 1739-1742, 2010.

[14] J. Shi and C. Tomasi, “Good features to track,” in Computer Vision and Pattern Recognition, 1994,” in Computer Vision and Pattern Recognition, pp. 593-600, 1994.

[15] D. Comaniciu, V. Ramesh, and P. Meer, “Real-time tracking of non-rigid objects using mean shift,” in Computer Vision and Pattern Recognition, pp. 142–149, 2000.

[16] J. Ning, L. Zhang, D. Zhang, and C. Wu, “Robust meanshift tracking with corrected background-weighted histogram,” IET Computer Vision, 6(1):62-69, 2012.

[17] C. Harris and M. Stephens, “A combined corner and edge detector,” in Alvey Vision Conference, pp. 147-151, 1988.

[18] E. Rosten and T. Drummond, “Machine learning for high-speed corner detection,” in European Conference on Computer, vol. 3951 LNCS, pp. 430-443, 2006.

[19] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, “Orb: An efficient alternative to sift or surf,” in International Conference onComputer Vision, pp. 2564-2571, 2011.

[20] D.G. Lowe, “Distinctive image features from scale-invariant keypoints”. International Journal of Computer Vision, 60(2):91-110, 2004.

[21] H. Bay, T. Tuytelaars , L. Van Gool, “Speeded-up robust features (SURF)”, Computer vision and image understanding, 110(3):346-359, 2008.

[22] B.D. Lucas and T. Kanade, “An iterative image registration technique with an application to stereo vision,” in DARPA Image Understanding Workshop, pp. 121-130, 1981.

[23] C. Tomasi and T. Kanade, “Detection and tracking of point features”, Tech. Rep., Journal of Computer Vision, 1991.

[24] J. Bouguet, “Pyramidal implementation of the Lucas Kanade feature tracker,” Intel Corp., Micro. Research Labs, 2000.

[25] K.S. Kim, D.S. Jang, and H. Choi, “Real time face tracking with pyramidal Lucas-Kanade feature tracker,” in Computational Science and Its Applications, vol. 4705 LNCS, pp. 1074–1082, 2007.

[26] R. Martin and J.M. Martinez, “Correlation study of video object trackers evaluation metrics,” IET Electronics Letters, 50(5):361-363, 2014.

[27] R. Kasturi, et al., “Framework for performance evaluation of face, text, and vehicle detection and tracking in video: Data, metrics, and protocol,” IEEE Trans. on Pattern Analysis and Machine Intelligence, 31(2):319–336, 2009.

[28] R. Brunelli, Template Matching Techniques in Computer Vision: Theory and Practice, Wiley Publishing, 2009.

[29] K. Nummiaro, E. Koller-Meier, and L.J. Van Gool, “An adaptive colour-based particle filter,” Image and Vision Computing, 21(1):99-110, 2003.

[30] S. Baker and I. Matthews, “Lucas-kanade 20 years on: A unifying framework,” International Journal of Computer Vision, 56(3):221-255, 2004.

[31] D. A. Ross, J. Lim, R. S. Lin, and M. H. Yang, “Incremental learning for robust visual tracking,” International Journal of Computer Vision, 77(1-3):125-141, 2008.

[32] Z. Kalal, K. Mikolajczyk, and J. Matas, “Tracking-learningdetection,” IEEE Trans. on Pattern Analysis and Machine Intelligence, 34(7):1409-1422, 2011.

[33] J. Ning, L. Zhang, D. Zhang, and C.Wu, “Scale and orientation adaptive mean shift tracking,” IET Computer Vision, 6(1):52-61, 2012.