Tracking-by-detection in a lecture hall setting

Daniel DeTone
Department of Computer Science and Electrical Engineering
University of Michigan at Ann Arbor
ddetone@umich.edu

Abstract

We present a framework for tracking a single human (person-of-interest) in a lecture hall environment. It is a tracking-by-detection framework that uses a generic person detector, a novel scoring function to solve the data association problem, and a Kalman filter that provides reliable state estimation. In our scoring function, we introduce two novel subcomponents: a subscore based on the target's width and a subscore based on the color histogram of him/her at the first time step. The primary application for such a system is the automation of recorded lectures at universities. The lecture hall environment introduces unique, challenging datasets for tracking because lecturers represent highly adaptive targets often subject to major, long duration lower-body occlusions. Our Markovian approach relies only on information from the past and is suitable for online applications. We evaluate performance of our tracker, Kalman-Scoring-Tracker (KST) on our own very challenging lecture hall dataset and show that it performs similar to or better than state-of-the-art adaptive appearance trackers.

1. Introduction

We aim to solve the problem of robustly tracking the 2D motion of a single person (person-of-interest) in a video frame from a single, stationary, color camera placed in the rear of a lecture hall or presentation room. The lecture hall presents unique challenges compared to typical human tracking scenarios such as those found in surveillience, traffic safety, or sports analysis applications. These unique challenges include major, long-term occlusion due to presentation podiums or tables and major upper-body deformation due to activities such as writing on a whiteboard and pointing around the room. The lecture hall setting also presents common human tracking problems such as handling multiple targets of the same object class due to an often viewable audience, a dynamic environment, and changes in scene illumination [23].

The major application for a robust, person-of-interest tracking system in lecture hall settings is recording professional quality video recordings of lectures, talks and speeches. Professional video quality implies proper body framing (the speaker's upper body and face are framed such that they occupy a large portion of the video frame). Using the algorithm presented in this paper, one can track a presenter within a large, very high resolution image so that a smaller subwindow or region-of-interest may be saved for future viewing, similar to [10]. In the university setting, the advantages of having recorded media for students are apparent: the videos help students with course material review prior to exams, support students that are unable to attend lecture, and allow for unlimited repetition to aid in comprehension, especially for students of which English is not their first language [19]. In the academic setting, recorded talks and meetings allow for collaboration for scientists who are unable to view streamed media due to differences in time zones and busy schedules. One example this problem is in the ATLAS Experiment at CERN which requires the successful collaboration of 3000 scientists worldwide. Professional quality recording services exist, but the high costs required for human camera operators make large-scale recordings financially infeasible [1].

The approach used in this paper is a tracking-by-detection method most similar to [6], and also found in [2], [9], [16], and [20]. The approach is outlined in Figure 1. Such approaches involve the continuous application of detections across multiple frames and the association of detections across frames. For detection, we use a generic person detector across each frame. This generic person detector yields many false positives due to other humans visible in the frame and protions of the stage which resemble a human, in addition to missed detections due to occlusion and deformation of the person-of-interest. To achieve a robust person-of-interest tracker, our algorithm adopts a scoring function similar to [6] to resolve the problem of data association. Our scoring function incorporates the person-of-interest's 2D position in the video, his/her width in the image, and a simple appearance histogram classifica-

tion. In tracking the person-of-interest, we adopt a recursive Bayesian framework model. We assume that the movement of the person-of-interest is Gaussian, thus we adopt the Kalman filter [14], which yields the optimal state estimate.

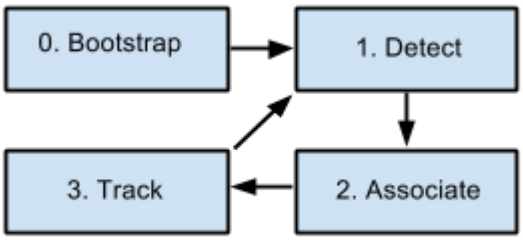

Figure 1. Framework Overview. A high-level overview of the main components of our tracking algorithm. Steps 1, 2 and 3 are repeated iteratively for each frame.

By using a recursive Bayesian framework, we model the movements of the person-of-interest in a Markovian manner, which only considers information from the current and most recent frame. Using such an approach is more suitable for time-critical, online approaches, as opposed to other methods which consider a global optimatization over the past, current and future frames as in [7], [10] and [20].

The remainder of the paper is organized as follows: in Section 2 we review various object tracking methodologies and their state-of-the-art implementations; in Section 3 we present our tracking algorithm in detail; and in Section 4 we present experimental results of our tracker run on our own challenging lecture hall dataset. We conclude in Section 5.

2. Related Work

A vast amount of work has been published on object tracking. A review of many tracking techniques can be found in [23].

Many approaches to tracking rely on background subtraction to segment the image into foreground and background pieces to simplify the tracking problem [8], [15], [22], and [10]. While this approach may be suitable in some surveillance applications, background subtraction is difficult in the lecture setting because motion in the video is frequently caused by objects other than the speaker such as students, other people on the stage, and transitions in lecture slides projected behind the speaker. Additionally, many of these techniques focus on tracking multiple targets, which presents different constrants to estimation techniques applied to various tracks, such as the inability to use a Kalman filter.

Another common approach to tracking is to use only an appearance model that is either manually defined or trained using only the first frame [11], [17], and [4]. While these methods offer generalization to objects of any appearance, they are often unable to cope with the signficant appearance changes exhibited by the typical actions of a lecturer. A detector trained on certain object class (in our case, humans) with thousands of images of different human orientations and poses can do a better job at handling occlusion and changes in pose and orientation.

Other approaches, such as [12], [18], [21], and [3] use an adaptive appearance model trained online, which means that the appearance model changes at each time step in the tracking progression. While this approach helps handle the mulitple orientations that a speaker takes during the course of a lecture, the track can suffer from drift problems caused by partial occlusions, which are very frequent in the lecture hall setting.

The approach used in this paper is most similar to [6], as both projects use a generic object detector as the primary input combined with a scoring function to solve the data association problem. In this paper, however, we are focused on tracking a single target, thus we are able to use the Kalman filter to generate a more optimal state estimation than the particle filter used in [6]. Additionally, we introduce two novel subscore components to the scoring function, as outlined in 3.4.

Some of the key contributions of this paper are:

1. We combine a generic class-specific object detector, scoring function, and Kalman filtering to address problems caused by the unreliable output from object detectors and multiple objects in view for tracking objects of specific class.
2. We introduce two novel components to the scoring function introduced in [6], namely, scoring based on object width and the object appearance histogram.

3. Method

3.1. Method Summary

As object detection has made massive improvements over recent years, a promising strategy is to employ an object detector for the observation model of the recursive Bayesian framework. However, the resulting detections are not often reliable, as shown in Figure 3, especially when applied with a low threshold, as in this paper. The detector will often generate detections which are not caused by humans (false positives), or not detect the person-of-interest due to major occlusion or large deformation (missed detections).

For many tracking applications, only past observations can be used at a certain time step to estimate the location of objects. Within this context, a recursive Bayesian framework is used, which recursively estimates the time-evolving posterier distribution of the target's location conditioned on previous observations.

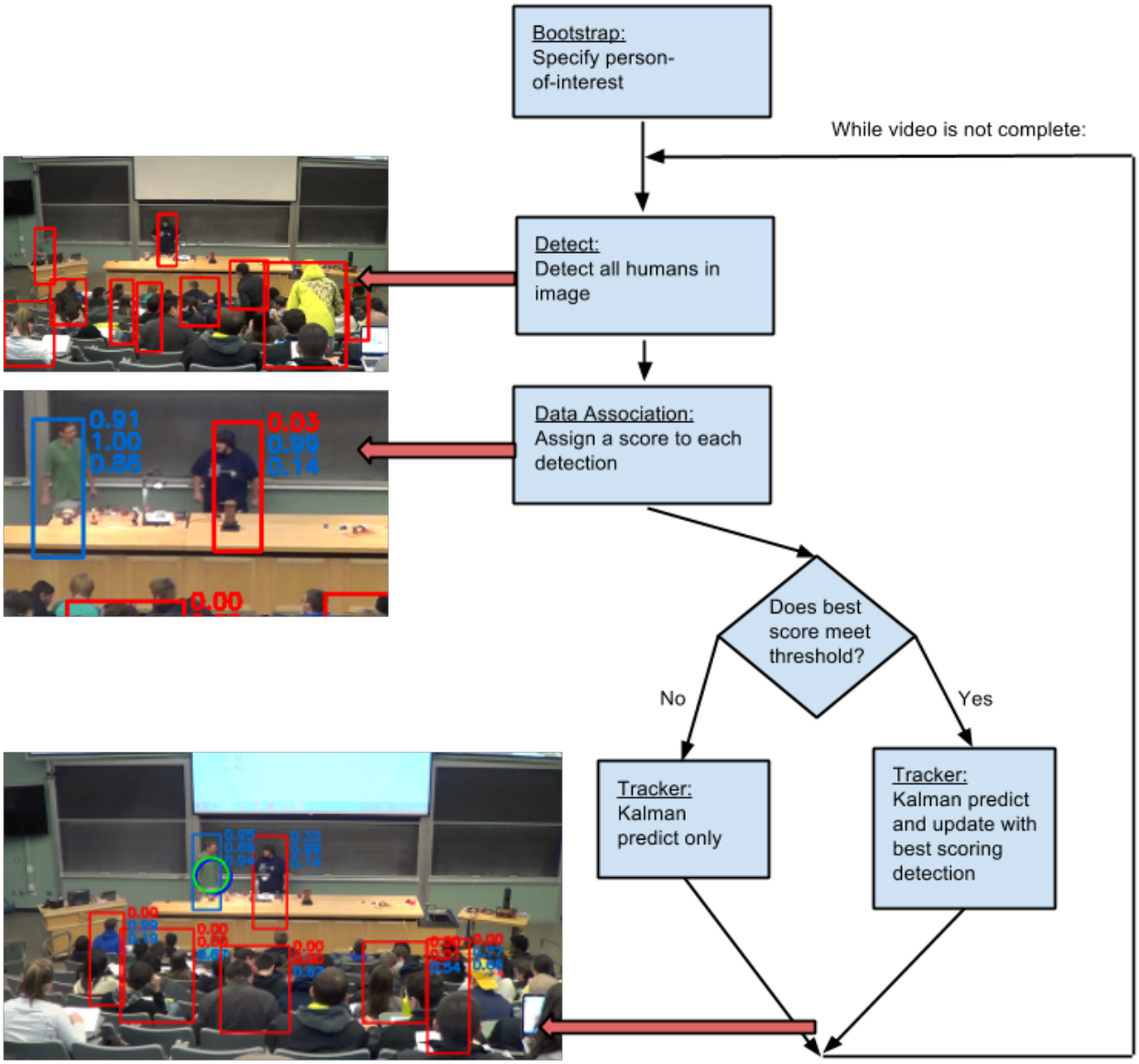

Figure 2: Algorithm Overview. After the initialization process, the algorithm detects all humans visible in the frame (top left image). The detector output typically contains detections of many humans in the image, which is apparent by the numerous red bounding boxes which represent detections. Next, each detection is assigned a score, which is a linear combination of the three scores shown next to each detction (middle image). Finally, if the best scoring detection meets a score threshold, it is assigned to the tracker. The green circle (bottom image) represents the centroid of the Kalman filter, which we use as the final output for our tracker at each time step.

Thus, our primary goal of 2D target tracking is to estimate the posterior distribution $P(X_t|Z^t)$ of the state of the 2D scene X_t at the current time step t given all the observations $Z^t = \{Z_1, \ldots, Z_t\}$ up to that time step. In our model, each observation Z_t is comprised of the set n of detections at time t, D_{tk}, where $k = \{1, \ldots, n\}$. Using such a model helps combat tracking error due to false positives, missed detections, and other humans which are visible in the image. We model the observations and corresponding error terms with a Gaussian distribution, thus the optimal state estimate is given by the Kalman filter [14].

A detailed, high-level overview of the tracking framework is shown in Figure 3.

3.2. Target Initialization

The target is first initialized by the user, who specifies the bounding box for the person-of-interest in the first frame. This initial bounding box serves as a starting point for the Kalman filter and is set as the first detection for the person-of-interest track. The size of this initial bounding box is important in the scoring function as it determines the weights given to the distance and width scoring functions. We denote the initial width of the bounding box at time $t = 0$ as W_0. This is discussed further in 3.4.

3.3. Human Detection

We use a state-of-the-art person detector as basic input to our tracking algorithm. We use the object detector introduced by Felzenszwalb et. al. [7], as implemented in OpenCV [5]. This method is a sliding-window detector based on mixtures of multiscale deformable part models. To obtain maximally possible recall, this detector is applied with a low threshold. While this introduces a number of false positives, such errors are typically corrected by assumptions made in the tracker, as it considers distance, width, and appearance in associating these detections to the track.

3.4. Data Association

In order to decide which detection should guide the tracker, we solve a data association problem: assigning a single detection to the tracker of the person-of-interest at each time step t.

The matching algorithm works as follows: at each time step, t, each detection, D_{tk}, where $k = \{0, \ldots, n\}$ is assigned a score, S. If the highest scoring detection $D_{max} = max(D_{tk})$ is greater than an empirically determined threshold S_{min}, then the detection is assigned to the tracker. Otherwise, no detection is associated to the track for the frame at time t.

The score S for each detection is a linear combination of three subscores: S_{dist}, S_{width}, S_{appear}. These three scores range from 0 to 1 and can be seen adjecent to each detection in Figure 4. The weight of the three subscores is determined by three parameters, α, β, and γ.

Figure 3. Human Detector Output. By using a low detector threshold we have fewer missed detections of the person-of-interest, but we also have more false positives.

$$S = \alpha * S_{dist} + \beta * S_{width} + \gamma * S_{appear} \quad (1)$$

Each subscore must be greater than the threshold S_{min} for the detection to be assigned the to tracker. This prevents unlikely associations from being made to the tracker at each time step, which helps mitigate tracker drift.

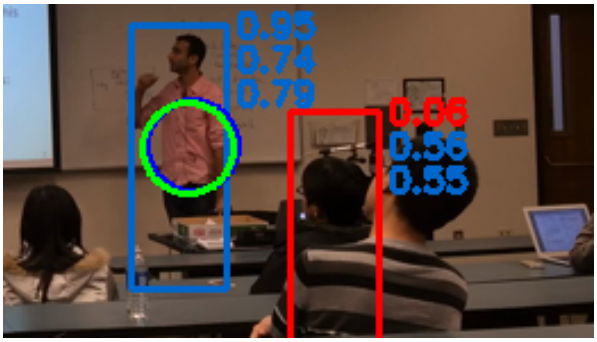

Figure 4. Scoring of Detections. Each detection is assigned three subscores, ranging from 0 to 1. The three subscores from the top to bottom represent the distance, width, and color appearance scores, respecively.

3.4.1 Distance Score

Our data association method generates a subscore, S_{dist}, based on the two-dimensional euclidean distance between the position of the track, T_{t-1}, at time $t-1$ and the detection D_{tk}.

$$S_{dist} \propto p_N(T_{t-1} - D_{tk}) \quad (2)$$

where $p_N(T_{t-1} - D_{tk}) \sim Gaussian(T_{t-1}, \sigma^2_{dist})$, which Gaussian distribution evaluated for the euclidean distance between T_{t-1} and D_{tk}, and

$$\sigma_{dist} \propto W_0 \quad (3)$$

where W_0 is the initial width of the person-of-interest from time $t = 0$. The score S_{dist} is normalized to one. By making $\sigma_{dist} \propto W_0$, we account for differences in the scale of lecture halls.

3.4.2 Width Score

Our data association method generates a subscore, S_{width}, based on the difference in widths of the track at time $t-1$, W_{t-1} and the detection D_{tk}, represented by $W_{D_{tk}}$. Because the width given by the detector for a given object can vary by a non-trivial amount due to deformations and occlusions, we keep a running average, W_t, for target width. This running average is averaged from the previous 10 frames. The width score is calcuated as

$$S_{width} \propto p_N(W_{t-1} - W_{D_{tk}}) \tag{4}$$

where $p_N(W_{t-1} - W_{D_t}) \sim Gaussian(W_{t-1}, \sigma^2_{width})$, which Gaussian distribution evaluated for the difference between T_{t-1} and D_{tk}, and

$$\sigma_{width} \propto W_0 \tag{5}$$

where W_0 is the initial width of the person-of-interest from time $t = 0$. The score S_{width} is normalized to one.

3.4.3 Appearance Score

Our data association method also generates a subscore, S_{appear}, based on the correlation of color histograms of the target in the initial frame and each detection in the current frame. As the lower half of the speaker is very often occluded in lecture hall environments, we use only the top half of the pixels enclosed by the detections as input to the appearance score. We denote the color histogram of the track at time $t = 0$ as H_0 and the color histogram for each detection as H_d.

$$S_{appear} = corr(H_0, H_{dk}) \tag{6}$$

where,

$$corr(H_1, H_2) = \frac{\sum_I (H_1(I) - \bar{H}_1)(H_2(I) - \bar{H}_2)}{\sqrt{\sum_I (H_1(I) - \bar{H}_1)^2 \sum_I (H_2(I) - \bar{H}_2)^2}} \tag{7}$$

and,

$$\bar{H}_k = \frac{1}{B} \sum_j H_k(j) \tag{8}$$

and B is the total number of histogram bins. As the correlation function ranges from 0 to 1, no normalization is necessary.

3.4.4 Kalman Filtering

To provide a reliable and relatively smooth final esimation for the target state, $P(X_t|Z^t)$, we employ a standard Kalman filter, as described in [14]. We use the mean of the estimation of $P(X_t|Z^t)$ at each time step t to represent T_t. For the predict step in the Kalman filter, we make one modification to [14]. We employ a decaying velocity model, which decreases the velocity estimation of $P(X_t|Z^t)$ by a factor of 0.9 at each time step. We make this modification because we know that the person-of-interest will not leave the image frame, thus if he/she is not detected for multiple frames, the track will not disappear off of the video. For the update step, we provide the highest scoring detection which is greater than the threshold S_{min}. If no such detection exists, we run only the predict step of the filter.

4. Experimental Results

4.1. Experimental Setup

Unfortunately there are no generally accepted benchmark video sequences for the lecture hall setting. Thus, we evalute the tracking algorithm on our own lecture hall dataset. This dataset includes 9 video sequences of classrooms at the University of Michigan. These test videos are recorded from a stationary camera mounted in the rear of each classroom. The videos have resolution of 640x360 and a frame rate of 5 fps. The sequences range from 1.5 minutes to 2.5 minutes in duration.

All parameters have been set experimentally. We chose a ratio for α:β:γ as 1:1:1. S_{min}, which is the minimum subscore required for each detection to be considered valid, is chosen as 0.1. For the histogram appearance classification, we represent the appearance of the targets in the H-S colorspace, where H is the hue and S is saturation. For the histogram comparison we use B = 50 bins with bin widths of 4.

4.2. Tracker Evaluation

To evaluate the quality of the tracker, we compare the tracker to three state-of-the-art trackers which are available as open-source projects. The three trackers used in this evaluation are: Multiple Instance Learning Tracker (MIL) [3], Online AdaBoost (OAB) [9] , and Tracking-Learning-Detection (TLD) [13]. We label the tracker introduced in this paper as Kalman-Scoring-Tracker (KST). Each tracker is run using its default settings. Because of the constant lower body occlusion in the lecture environments of our dataset, the initialization of these trackers was set to be the upper half of the bodies of the person-of-interest.

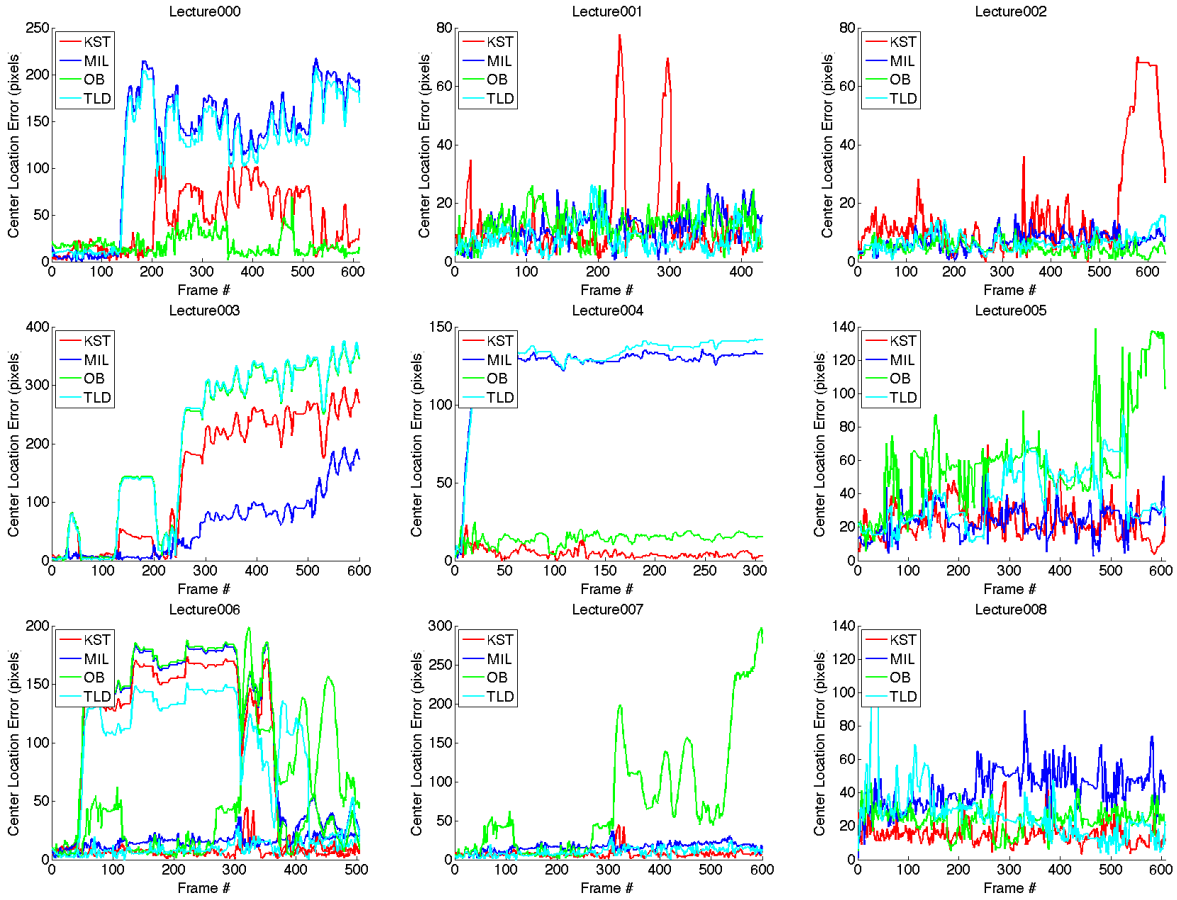

Figure 5: Tracking Error Plots. Person-of-interest tracking error per frame. See text for details.

Video Clip	MIL	OB	TLD	KST(this paper)
Lecture000	125	18	117	44
Lecture001	12	12	8	12
Lecture002	7	5	6	16
Lecture003	59	203	204	146
Lecture004	124	14	129	5
Lecture005	23	60	38	22
Lecture006	113	111	94	98
Lecture007	16	74	10	7
Lecture008	42	23	26	16

Figure 6: Average Error Table. Average tracking error for each video sequence, for each algorithm. The text in green font indicates the best performance, the red indicates second best.

In Figure 5, we plot the center euclidean position error for each frame for each of the four trackers. Qualitatively, in Lecture001 and Lecture002, when we see spurrious periods of high error, the tracker is able to recover. The KST is able to recover from moderate amounts of drift because its input is generalized to the object class. In general, the others trackers are unable to recover from high error periods (tracker drift), because of the high amount of appearance learning in each of their implementations.

In Figure 6, we display a table of the average tracking error calculated for each video frame for each tracking algorithm. When compared with this metric, the KST tracker is the best performing tracker in 4/9 datasets and second best performing in 3/9 datasets.

5. Conclusion

In this paper we presented a Tracking-by-Detection method that uses a generic person detector as basic input. We constrained the tracking problem to the lecture hall environment, because it presents an exciting application, along with some unique challenges such as highly adaptive targets and major lower body occlusions. To solve the data association problem in this framework, we presented a scoring function similar to [6], with the addition of two novel subscoring functions based on object width and histogram appearance. We presented empirical results on many challenging video clips where we measured quatitative performance of our tracker compared to a number of state-of-the-art single-object tracking algorithms. Qualitatively we observed that the KST tracker is able to recover from major drift in some cases due to its generalized input, whereas the other trackers that rely heavily on an adaptive appearance model for tracking were typically unable to do so. Quantitatively we saw that our tracker generates the minimum average tracking error in more datasets than the others it was compared to.

There are many areas for further exploration in the framework presented in the paper. The color histogram comparison component of the scoring function seems ripe for exploration. For example, as we see in Figure 5, we have high confidence that the tracker is correctly tracking the target for the first approximately 10% of the frames. We could, for instance, develop a more complex appearance model based on the these initial frames and still avoid problems with tracking drift experienced by the adaptive appearance tracking algorithms.

Finally, for use in an exciting application area such as automated lecture recording, much more robust results are required than those presented in this paper. This is because generating a recorded lecture where the person-of-interest is not centered in the video frame for the vast majority of the lecture would not be helpful to potential users. The datasets presented in this paper were very challenging, as they often included the audience in the frame. Perhaps more simple and constrained video input would provide a suitable system. Such simplification could be provided by cropping out non-salient areas from the video, such as the audience, based on prior knowledge of the stage or by estimating the geometry of the lecture hall.

References

[1] Carma: Campus automated rich media archiving, Nov. 2013. 1

[2] S. Avidan. Ensemble tracking. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(2):261–271, Feb. 2007. 1

[3] B. Babenko, M.-H. Yang, and S. Belongie. Robust object tracking with online multiple instance learning. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(8):1619–1632, Aug. 2011. 2, 5

[4] M. J. Black and A. D. Jepson. Eigentracking: Robust matching and tracking of articulated objects using a view-based representation. *Int. J. Comput. Vision*, 26(1):63–84, Jan. 1998. 2

[5] G. Bradski. *Dr. Dobb's Journal of Software Tools*, 2000. 4

[6] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. V. Gool. Robust tracking-by-detection using a detector confidence particle filter. In *IEEE International Conference on Computer Vision*, October 2009. 1, 2, 7

[7] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(9):1627–1645, Sept. 2010. 4

[8] J. B. Francois, J. Berclaz, F. Fleuret, and P. Fua. Robust people tracking with global trajectory optimization. In *In Conference on Computer Vision and Pattern Recognition*, pages 744–750, 2006. 2

[9] H. Grabner and H. Bischof. On-line boosting and vision. In *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 1*, CVPR '06, pages 260–267, Washington, DC, USA, 2006. IEEE Computer Society. 1, 5

[10] S. Halawa, D. Pang, N.-M. Cheung, and B. Girod. Classx: An open source interactive lecture streamingsystem. In *Proceedings of the 19th ACM International Conference on Multimedia*, MM '11, pages 719–722, New York, NY, USA, 2011. ACM. 1, 2

[11] M. Isard and J. MacCormick. Bramble: A bayesian multiple-blob tracker. In *ICCV*, pages 34–41, 2001. 2

[12] A. D. Jepson, D. J. Fleet, T. F. El-maraghi, I. C. Society, I. C. Society, and I. C. Society. Robust online ap-

pearance models for visual tracking. pages 415–422, 2001. 2

[13] Z. Kalal, K. Mikolajczyk, and J. Matas. Tracking-learning-detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(7):1409–1422, July 2012. 5

[14] R. E. Kalman. A new approach to linear filtering and prediction problems. 1960. 2, 4, 5

[15] O. Lanz. Approximate bayesian multibody tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28:1436–1449, 2006. 2

[16] B. Leibe, K. Schindler, and L. V. Gool. Coupled detection and trajectory estimation for multi-object tracking. In *In ICCV*, pages 1–8, 2007. 1

[17] V. Lepetit and P. Fua. Keypoint recognition using randomized trees. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(9):1465–1479, Sept. 2006. 2

[18] I. Matthews, T. Ishikawa, and S. Baker. The template update problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26:810–815, 2004. 2

[19] H. Odhabi and L. Nicks-McCaleb. Video recording lectures: Student and professor perspectives. *British Journal of Educational Technology*, 42, 2011. 1

[20] K. Okuma, A. Taleghani, N. D. Freitas, O. D. Freitas, J. J. Little, and D. G. Lowe. A boosted particle filter: Multitarget detection and tracking. In *In ECCV*, pages 28–39, 2004. 1

[21] D. A. Ross, J. Lim, R.-S. Lin, and M.-H. Yang. Incremental learning for robust visual tracking. *Int. J. Comput. Vision*, 77(1-3):125–141, May 2008. 2

[22] X. Song, J. Cui, H. Zha, and H. Zhao. Vision-based multiple interacting targets tracking via on-line supervised learning. In *Proceedings of the 10th European Conference on Computer Vision: Part III*, ECCV '08, pages 642–655, Berlin, Heidelberg, 2008. Springer-Verlag. 2

[23] A. Yilmaz, O. Javed, and M. Shah. Object tracking: A survey. *ACM Comput. Surv.*, 38(4), Dec. 2006. 1, 2