Adaptive Tobit Kalman-based tracking

Kostas Loumponias, Anastasios Dimou, Nicholas Vretos, Petros Daras
Information Technologies Institute,
Centre for Research and Technology Hellas - CERTH,
Thessaloniki, Greece, GR-54124
{*loumponias, dimou, vretos, daras* }*@iti.gr*

***Abstract*—This paper presents an online, real-time, multi-object tracking algorithm based on a novel method for data association. Tracking multiple objects in real-world scenes includes several challenges, such as a) object detectors with low detection accuracy, b) false alarms, and c) unmatched tracked objects. In this paper, we propose a novel filtering method based on the theory of censored data by utilizing an Adaptive Tobit Kalman filter to estimate the object's position with high accuracy. Furthermore, in order to deal with false alarms and unmatched tracked objects, we use the non-maximum suppression and a modified Hungarian algorithm, respectively. Experiments in public datasets show that the proposed method outperforms state of the art methods in multi-object tracking with a substantial low computational cost compared to other methods in the area.**

***Keywords*-Multi-object Tracking; Adaptive Tobit Kalman Filter; Hungarian Algorithm;**

I. Introduction

Multi-object tracking (MOT) is defined as the estimation of location and size of multiple objects in each frame of a video sequence while preserving their identity. Accurate MOT involves several challenges, especially in crowded scenes, due to multiple occlusions and changes in the object appearance (different poses) and lightning. One of the most popular methods that have been proposed for MOT is tracking-by-detection, where individual object detections are linked to form trajectories of the detected objects [1]. Tracking-by-detection can be separated in two parts, namely the object detection and therefore the detection association. The accuracy of such a tracker is highly dependant on the performance of the object detector (a.k.a. detection sparsity). On the other hand, the task of the detection association is getting tougher with missing or noisy detections.

The tracking-by-detection approaches can be categorized in: 1) batch tracking (BT), and 2) online tracking (OT). In BT approaches, a set of detections from a window (batch) of frames is exploited to build robust tracklets using, usually, graph-based techniques to cope with detection errors caused by occlusions [2] or detection sparsity. However, BT methods require detection responses of future frames and are usually computationally expensive. Due to these limitations, BT methods are not suitable for online and real-time (over 15 fps) applications [3]. On the other hand, OT approaches can used for online applications since they can track objects in a frame-by-frame manner using the information available up to the current frame. Nonetheless, OT methods tend to produce fragmented trajectories as it is more difficult to handle inaccurate detections (e.g false alarms) compared to BT methods [2].

In this paper, we propose an online method for MOT, which can robustly track multiple objects using noisy detections with a low computational cost. The proposed method relies on a previous OT method [4], however, in order to achieve better tracking accuracy, we use 1) a modified Hungarian algorithm (HA) [5], and 2) the Adaptive Tobit Kalman filter (ATKF) [6] instead of a simple Kalman filter (KF). Recently, it has been shown in [7] that ATKF outperforms other state-of-the-art methods like [8],[9] in minimizing the estimation error on noisy observations in the case of Kinect data for skeleton analysis. The proposed ATKF takes advantage of the approaches presented in [6], [7] by providing more accurate state estimations due to the accurate calculation of the variance of the censored measurements (Appendix).

In MOT, one of the major challenges is to correctly associate noisy detections (observations) with previously tracked objects. In order to achieve the latter, the prediction and the update stage of ATKF are automatically adapted to the frame rate of the examined video and to a confidence value of the detection (observation).

1) An improved version of [4] based on ATKF, which increases the predictions of human's bounded boxes in real-time applications.
2) An online and real-time human tracking approach based on a modified HA.

The rest of the paper is organized as follows. In Section 2, related works are described, while in Section 3, the proposed methodology on human tracking is presented in detail. In Section 4, experimental results are illustrated, using the 2D MOT 2015 benchmark [10]. Finally, Section 5, concludes the paper.

II. Related Work

A considerable amount of solutions have been proposed in the literature in order to solve the Multi-Object Tracking problem. Most of them are focusing on improving the performance of the data association process, including the proposed work. Others, offer both object detection and detection association. Existing data association approaches for MOT are either BT or OT ones. In this section, we focus on OT approaches since they are the most relevant to our work.

In [11], [12] two online MOT methods based on stochastic models are presented. More specifically, in [11] the authors utilize the Gaussian mixture probability model density (GM-PHD) filter [13] due to its resistance in noisy and random data. Nevertheless, this method results in many false alarms in the detection accuracy. In [12], the proposed method combines a local and a global tracker in a comprehensive two-step framework. In the local tracking step [12], a frame by frame association is used in order to generate online object trajectories. Each object trajectory is represented by a set of multimodal feature distributions modeled by General Mixture Models (GMMs) [14]. In the global tracking step, occlusions and false alarms are recovered by the tracklet bipartite association method based on the Mahalanobis metric [15]. Deep Learning approaches have been, also, proposed as in [16], where a novel online method based on RNNs is presented. Therein, the authors describe the way of addressing several challenges, which arise in training RNNs for MOT. They achieve high MOT accuracy in the 2D MOT 2015 benchmark [10], however, in corresponding testing data (including more complex scenes), the MOT accuracy is significantly lower.

The closest work to the proposed method is presented in [4], where HA and KF are used for MOT. Tracklets are formed by associating detections throughout adjacent frames, where both geometry and appearance cues are combined to associate detections with previously tracked objects. Furthermore, Faster Region CNN (FrRCNN) [17] is used in order to produce higher quality detections. The proposed method, while inspired by [4], introduces significant improvements by using a pragmatic and adaptive tracking approach based on censored data theory [18] that is especially robust to noisy detections. The method is described in detail in the following section.

Many filtering methods exist for MOT either from images, videos or depth information. In the rest of this section, we mention the most known and well established filtering methods.

One of the most known filtering method is KF. In order to overcome several drawbacks of KF (mainly due to its linear nature), the Extended Kalman Filter (EKF) was proposed in [19]. Although EKF is not an optimal estimator as its linear counterpart, it has been proved that it performs better than KF in terms of smoothing and correcting signals in problems that are non-linear. However, EKF tends to be unstable in many applications due to its local nature, leading to incorrect smoothing of a signal that exhibits a high degree of non-linearities. To overcome these problems, the Unscented Kalman Filter (UKF) was proposed in [20]. UKF uses a deterministic sampling technique known as unscented transform [21] to gather a minimal set of points around a local mean. By doing so, it provides better results than EKF when the predict and the update functions are highly non-linear, although, UKF requires more computational cost than EKF. Finally, a very successful method is the Particle Filtering (PF) [22], which is a Monte Carlo based filtering method. Though PF is generally very adaptable, it requires a high computational burden, making it practically unsuitable for many real-time and online applications.

In the area of censored statistics [23], all the above mentioned methods have their drawbacks. In [24], it is stated that the formulation of a standard KF, as an estimator for censored data, results in a biased estimation of the unknown state. EKF suffers from an undefined Jacobian at the censored region, resulting in an ill-posed Jacobian. On the other hand, it is proven, that UKF is non-robust when the measurements are close to the censored region [24]. Finally, ATKF provides unbiased, recursive estimates of the latent state variables when the measurements are close to the censored region. ATKF is completely recursive and computationally inexpensive, making it a perfect candidate for real-time and online MOT. Furthermore, the proposed ATKF provides 1) a more accurate estimation of censored variance measurement (Appendix) and 2) adaptive censoring limits at each time step, compared to Tobit Kalman Filter (TKF) given in [24].

III. Methodology

The proposed method includes three steps: 1) the rejection of detections corresponding to false alarms, 2) the association of current detections with the existing trackers, and 3) the update of the predicted bounding box position using the ATKF process.

A. Detections

Detectors, such as [10], do not provide accurate predictions of humans' detections, making the tracking problem even harder. The main issue is the big amount of multiple overlapping detections that appear in the data. In order to avoid multiple overlapping detections, we use the non maximum suppression (NMS) algorithm [25] for the detections at every frame. The NMS algorithm is responsible for merging detections that belong to the same object, through a simplistic process that is based on a greedy clustering with a fixed distance threshold T_{NMS}. Furthermore, a confidence value is associated with each bounding box, which is taken into account from our method for their rejection.

B. Data Association

As described in [4], the detections at time frame k are associated with the predicted objects' position, derived from (1)-(2) (predicted trackers). The trackers's bounding boxes' coordinates are predicted as follows (Predict function):

$$\hat{\mathbf{x}}_k^- = \mathbf{A}\hat{\mathbf{x}}_{k-1} \tag{1}$$

$$\mathbf{P}_k^- = \mathbf{A}\mathbf{P}_{k-1}\mathbf{A}^T + \mathbf{Q}, \tag{2}$$

where $\mathbf{x}_k = [x_{k,1}, x_{k,2}, x_{k,3}, x_{k,4}, \dot{x}_{k,1}, \dot{x}_{k,2}, \dot{x}_{k,3}, \dot{x}_{k,4}]$, $\hat{\mathbf{x}}_k^-$ and $\hat{\mathbf{x}}_k$ are the state vector, the a priori and the a posteriori estimation at time frame k, respectively, while $\mathbf{P}_k^-, \mathbf{P}_{k-1}$ are the covariances of a priori and a posteriori error estimation, respectively. $\mathbf{Q}$ is called the covariance of error process and for convenience we suppose that it is

constant for every video sequence in [10] since it models the equipment used to acquire the videos and is set to:

$$\mathbf{Q} = \begin{bmatrix} \frac{1}{2} & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \frac{1}{2} & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 2 \end{bmatrix} \tag{3}$$

$\mathbf{A}$ is the transition matrix of the ATKF process and takes the form:

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 & 0 & \frac{1}{fps} & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & \frac{1}{fps} & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & \frac{1}{fps} & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & \frac{1}{fps} \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{4}$$

where fps stands for frames per second at each video sequence.

In the next step, the predicted trackers as resulted by (1) are assigned with the detections. To that end, an assignment cost matrix is computed, as the intersection-over-union (IOU) [4] between each detection and all predicted bounding boxes from the existing targets. For the assignment between tracker and detection, HA is used. In HA [26], a detection can be assigned only once to an existing tracker. Thus, in many cases where two objects (humans in our case) are too close, the detector provides only one detection. This has the consequence of assigning the detection to only one tracker, thus missing the second person's tracks. In order to deal with these cases, we altered HA so as to assign the unmatched tracked objects with detections when the corresponding IOU is large enough (over a threshold).

C. Estimation of bounding box position

In the final stage, the predicted trackers are updated with the assigned detections, by using the ATKF update function. In order to be more precise with the update function of ATKF, we provide some background of the censored data theory [18]. In statistics research, censoring is a condition in which the value of a measurement or observation is only partially known. Censoring occurs when a value falls outside the range of a measuring instrument. For example, a bathroom scale might only measure up to 140 kg. If an 150 kg individual is weighed using that scale, the observer would only know that the individuals weight is at least 140 kg (partially known).

We denote by $\mathbf{z}_k^* = [z_{k,1}^*, z_{k,2}^*, z_{k,3}^*, z_{k,4}^*]$ and $\mathbf{z}_k$ the (latent) measurement-detection given by the detector and the censored detection-measurement, respectively, at time frame k. The Tobit model is called censored regression model and is characterized by the stochastic difference non-linear equation:

$$\mathbf{z}_k^* = \mathbf{H}\mathbf{x}_k + \mathbf{v}_k, \tag{5}$$

$$z_{k,i} = \begin{cases} z_{k,i}^*, & T_{lowe,k}^i < z_{k,i}^* < T_{upper,k}^i \\ T_{lower,k}^i & z_{k,i}^* \leq T_{lower,k}^i \\ T_{upper,k}^i, & z_{k,i}^* \geq T_{upper,k}^i \end{cases} \quad i = 1,2,3,4 \tag{6}$$

where $\mathbf{v}_k \sim N(\mathbf{0}, \mathbf{R}_k)$ and the adaptive censored limits $T_{lower,k}^i, T_{upper,k}^i$ are given by:

$$\mathbf{T}_{upper,k} = \mathbf{H}\hat{\mathbf{x}}_k^- + \mathbf{a} \tag{7}$$

$$\mathbf{T}_{lower,k} = \mathbf{H}\hat{\mathbf{x}}_k^- - \mathbf{a}, \tag{8}$$

where $\mathbf{a} = (a_i)_{i=1}^4$ is a vector.
The observation matrix of ATKF is defined as:

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{9}$$

The update function of ATKF is given by:

$$\mathbf{R}_{k,1} = \mathbb{E}\left((\mathbf{x}_k - \hat{\mathbf{x}}_k^-)(\mathbf{z}_k - \mathbb{E}(\mathbf{z}_k))^T | \mathbf{z}_{k-1}\right), \tag{10}$$

$$\mathbf{R}_{k,2} = \mathbb{E}\left((\mathbf{z}_k - \mathbb{E}(\mathbf{z}_k))(\mathbf{z}_k - \mathbb{E}(\mathbf{z}_k))^T | \mathbf{z}_{k-1}\right), \tag{11}$$

$$\mathbf{K}_k = \mathbf{R}_{k,1}\mathbf{R}_{k,2}^{-1}, \tag{12}$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}_k(\mathbf{z}_k - \mathbb{E}(\mathbf{z}_k)), \tag{13}$$

$$\mathbf{P}_k = \mathbf{P}_k^- - \mathbf{K}_k\mathbf{R}_{k,1}^T, \tag{14}$$

where the matrices $\mathbf{R}_{k,1}, \mathbf{R}_{k,2} = (R_{k,2}^i)_{i=1,\ldots,4}$ and the censored mean $\mathbb{E}(\mathbf{z}_k)$ are given in the Appendix. $\mathbf{K}_k$ and $\hat{\mathbf{x}}_k$ are the Kalman gain and the state vector at time frame k, respectively. Finally, $\mathbf{R}_{k,2}$ depends on the covariance matrix of measurement error, $\mathbf{R}_k$. As we mentioned before, the detector includes a confidence value, C_k, for each detection-measurement, which lies within a predefined bound. Therefore, it is reasonable for $\mathbf{R}_k$ to be inversely proportional to C_k, thus, $\mathbf{R}_k$ is defined as:

$$\mathbf{R}_k = \begin{bmatrix} 1.5 & 0 & 0 & 0 \\ 0 & 1.5 & 0 & 0 \\ 0 & 0 & 1.5 & 0 \\ 0 & 0 & 0 & 1.5 \end{bmatrix} \cdot (1 - \frac{C_k}{140}) \tag{15}$$

It is clear that the above process can provide accurate estimations only when the predicted tracker is assigned to a detection. Thus, we use the a priori estimation, $(\hat{x}_{k,i}^-)_{i=1}^4$, as the latent detection, $\mathbf{z}_k^*$, for T time frames 1) when the predicted tracker is not assigned to any detection and, 2) when it has been detected for at least $\frac{2fps}{3}$ consequent times. The number T depends on the fps of the video sequence and the velocity of the unmatched tracked object for two reasons: firstly, if the velocity of the predicted tracker is small, then the a priori error prediction by (1)-(2) is reduced. Secondly, the spatial displacement between

two consecutive bounding boxes' positions (of the same object) is reduced as fps is increased, therefore:

$$T = \begin{cases} max(3, \frac{fps}{6} + 1), \ \hat{\dot{x}}_{k,1}^{-} < 5 \quad and \quad \hat{\dot{y}}_{k,1}^{-} < 5 \\ max(3, \frac{fps}{8} + 1), \ otherwise \end{cases} \tag{16}$$

In the case where the fps is too small (<7), it is clear that the error estimation of the bounding box position may increase rapidly if the predicted tracker is not assigned to any detection, therefore, we assume that $T = 1$. In Figure 1 a framework of the proposed method for MOT is illustrated.

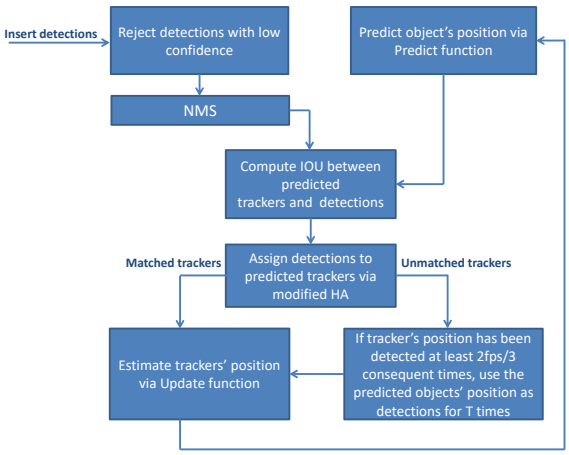

Figure 1. Framework of proposed method.

IV. Experiments

We evaluate the performance of our tracking implementation on the MotChallenge 2015 database [10], which contains both moving and static camera sequences. To initiate tracking, we assume that the minimum IOU is equal to 0.15 (as in [4]). Then, we define the minimum IOU for an unmatched tracked object (IOU_{unm}) and T_{NMS} in such a way in order to achieve the highest MOT accuracy. Using random search, the best values for IOU_{unm} and T_{NMS} are experimentally (training data [10]) found to be:

$$IOU_{unm} = 0.60, \tag{17}$$

$$T_{NMS} = 0.55 \tag{18}$$

Low values for IOU_{unm}, close to 0.40, mean that an unmatched tracked object can be matched by an inappropriate detection. On the other hand, high values, close to 0.80, mean that an unmatched tracked object cannot be matched to any nearby detection. Furthermore, low values for T_{NMS}, close to 0.35 mean that detections corresponding to different objects will be merged. On the other hand, high values, close to 0.75, mean that different detections corresponding to an object (false alarms) will not be merged (Figure 2). It is worth mentioning that a higher value for T_{NMS} can be selected, if the detector does not produce many false alarms. Finally, vector, $\mathbf{a}$, in (7)-(8), is experimentally (training data [10]) chosen to be:

$$\mathbf{a} = [40, 25, 40, 25]^T \tag{19}$$

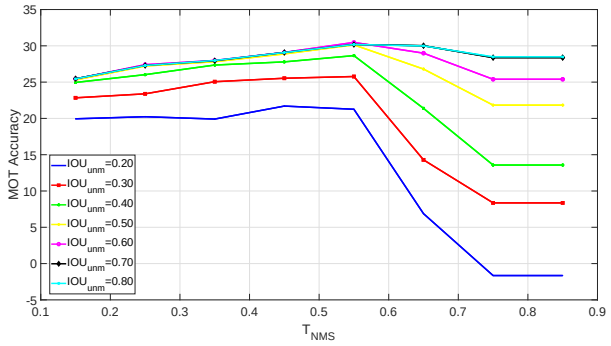

Figure 2. MOT Accuracy for various values of parameters T_{NMS} and IOU_{unm}, perspectively.

A. Tracking Performance Evaluation

We utilize the following evaluation metrics defined in [32], [33] :

- MOTA: Multi-object tracking accuracy.
- MOTP: Multi-object tracking precision.
- FA: The average number of false alarms per frame.
- MT: The ratio of ground-truth trajectories that are covered by a track hypothesis for at least 80% of their respective life span.
- ML: The ratio of ground-truth trajectories that are covered by a track hypothesis for at most 20% of their respective life span.
- FP: The total number of false positive detections.
- FN: The total number of false negative detections.
- ID sw: The total number of times an ID switches to a different previously tracked object.
- Frag: The total number of fragmentations where a track is interrupted by miss detection.
- Hz: Processing speed (in frames per second excluding the detector) on the benchmark.

The metric $MOTA$ is applicable to a wide range of tracking tasks [32] and allows for objective comparison of the main characteristics of tracking systems, such as their accuracy in recognizing object configurations and their ability to consistently track objects over time.

In Table I the proposed method, namely $ATKF$, is compared against an $ATKF$ without a modified HA and two other methods [4], [16], on the training dataset of [10]. It is clear that our method outperforms [4], in all but one metric (False Negative). Furthermore, the ATKF without modified HA is able to achieve a good performance in MOT. However, we utilized the modified HA in order to improve a little more the evaluation metrics.

As we mentioned before, in order to reduce the false alarms, we reject the detections with low confidence value and we merge the bounded boxes (by NMS) when the distance between them is small. This can lead to the

Method	MOTA ↑	MOTP ↑	FA ↓	FP ↓	FN ↓	ID sw ↓	Frag ↓
SORT [4]	26.0	72.5	1.23	6767	**21988**	780	1174
RNN_LSTM [16]	22.3	69.0	0.97	5327	25094	572	983
ATKF wihtout Mod.HA	30.2	72.7	0.72	3962	23624	254	627
ATKF (proposed)	**30.5**	**72.7**	**0.67**	**3706**	23797	**240**	**606**

Table I
PERFORMANCE OF $SORT$, $RNN\ LSTM$, $ATKF$ WITHOUT MODIFIED HA AND PROPOSED METHOD, RESPECTIVELY, ON MOT TRAINING SEQUENCES [10].

Method	MOTA ↑	MOTP ↑	FA ↓	MT ↑	ML ↓	FP ↓	FN ↓	ID sw ↓	Frag ↓	Hz ↑
CppSORT [27]	21.7	71.2	1.5	3.7%	49.1%	8422	38454	1231	2005	**1122.1**
RNN_LSTM [16]	19.0	71.0	2.0	5.5%	45.6%	11578	36706	1490	2081	165.2
OMTDFH [28]	21.2	69.9	2.3	7.1%	46.5%	13218	34657	563	1255	28.6
GSCR [29]	15.8	69.4	1.3	1.8%	61.0%	7597	43633	**514**	**1010**	28.1
LDCT [30]	4.7	**71.7**	2.4	**11.4%**	**32.5** %	14066	**32156**	12348	2918	20.7
GMPHD [11]	18.5	70.9	1.4	3.9%	55.3%	7864	41766	459	1266	19.8
TSDAOAL [31]	18.6	69.7	2.8	9.4%	42.3%	16350	32853	806	1544	19.7
MTStracker [12]	20.6	70.3	2.6	9.0%	36.9%	15161	32212	1387	2357	19.3
ATKF (proposed)	**24.8**	70.8	**1.1**	4.0%	52.0%	**6201**	39321	666	1300	205.6

Table II
PERFORMANCE OF THE PROPOSED AND OTHER ONLINE AND REAL-TIME METHODS (WITH PROCESSING SPEED OVER THAN $15fps$) ON MOT TESTING SEQUENCES [10].

Figure 3. Tracking results of ATKF on the MOTChallenge sequence PETS09-S2L1. Frames 612, 635, 653, 686, 721, 736, 776 and 795 are shown. The colour of each bounding box indicates the person identity.

rejection of a few correct detections and, therefore, FN slightly increases. In Figure 3, some tracking outputs (bounding boxes) of the proposed method are shown.

In Table II, $ATKF$ is compared against several other online and real-time methods with frequency over $15fps$ on the testing dataset of [10]. The results show that the proposed method outperforms all others methods in $MOTA$. Moreover, the proposed method achieves the highest scores in metrics FA and FP with scores 1.1 and 6201, respectively. Furthermore, the proposed method achieves a high score in metric $Frag$, with score 1300, while the highest score is achieved by $GSCR$, with score 1010.

As expected, the methods with a good performance in metrics MT and ML, present inferior performance in metrics $ID\ sw$ and Hz, respectively (see Table II). This is due to the fact that when a target is not associated with any detection (or its position is not predicted) either its trajectory is terminated or it is associated with a previously tracked object. More specifically, $LDCT$ achieves the best scores in metrics MT and ML, although it suffers from a huge number of $ID\ sw$. Finally, the proposed method is able to achieve a high score in metric $MOTP$, despite of the fact that in several cases the detections' coordinates are not provided (due to occlusions). Hence, the target's position is estimated with high accuracy (without any detection) by the proposed method.

V. CONCLUSION

In this paper, we presented a robust MOT method that focuses on frame-by-frame prediction and association. The

proposed method works on online mode and is suitable for real-time applications. To the best of our knowledge, this is the first approach that employs censored distributions to address online multi-target tracking. We have shown that the new filtering process, ATKF, can be utilized to handle noisy observations and short-term occlusions. Furthermore, in order to deal with the unmatched tracked objects, we modified HA and finally we accurately predicted the object's position in fully-occluded scenarios. The results on a public dataset show that the proposed method can achieve the highest $MOTA$ compared to other online and real-time approaches, with the second highest processing speed.

VI. Appendix

The probability distribution function of $z_{k,i}$ (6) given $z_{k-1,i}$ is given by:

$$\begin{aligned} f(z_{k,i}|z_{k-1,i}) = \frac{1}{s_{k,i}}\phi\Big(\frac{z_{k,i}-m_{k,i}}{s_{k,i}}\Big)I(z_{k,i}) \\ + \delta(z_{k,i}-T^i_{upper,k})D^i_{upper,k} \\ + \delta(z_{k,i}-T^i_{lower,k})D^i_{lower,k}, \end{aligned} \tag{20}$$

where

$$I(x) = \begin{cases} 1,\ x \in [T^i_{lower,k}, T^i_{upper,k}] \\ 0,\ otherwise \end{cases}, \tag{21}$$

δ is the Kronecker delta function, $m_{k,i} = (\mathbf{H}\hat{\mathbf{x}}^-_k)_i$, $s^2_{k,i} = (\mathbf{H}\mathbf{P}^-_k\mathbf{H}^T + \mathbf{R}_k)_i$, $\phi(x)$ is the probability distribution function of the standard normal distribution, the matrices $\mathbf{D}_{un,k}$, $\mathbf{D}_{lower,k}$ and $\mathbf{D}_{upper,k}$ have the form:

$$\mathbf{D}_{un,k} = diag\begin{bmatrix} \Phi(T_{u^1_k}) - \Phi(T_{l^1_k}) \\ ... \\ \Phi(T_{u^m_k}) - \Phi(T_{l^m_k}) \end{bmatrix}, \tag{22}$$

$$\mathbf{D}_{lower,k} = diag\begin{bmatrix} \Phi(T_{l^1_k}) \\ ... \\ \Phi(T_{l^m_k}) \end{bmatrix}, \tag{23}$$

$$\mathbf{D}_{upper,k} = diag\begin{bmatrix} 1-\Phi(T_{u^1_k}) \\ ... \\ 1-\Phi(T_{u^m_k}) \end{bmatrix}, \tag{24}$$

where

$$T_{u^i_k} = \frac{T^i_{upper} - m_{k,i}}{s_{k,i}}, \tag{25}$$

$$T_{l^i_k} = \frac{T^i_{lower} - m_{k,i}}{s_{k,i}} \tag{26}$$

and Φ is the cumulative distribution function of the standard normal distribution. For the sake of convenience, we denote $z_{k,i}|z_{k-1,i}$ as y.

The moment generating function, $M(t)$, of the random variable y with probability distribution function $f(y)$ (20) has the form:

$$\begin{aligned} M(t) = \mathbb{E}\,e^{ty} = e^{tT^i_{lower,k}}D^i_{lower,k} + e^{tT^i_{upper,k}}D^i_{upper,k} \\ + e^{m_{k,i}t + s^2_{k,i}t^2/2}\Big(\Phi\Big(\frac{T^i_{upper} - m^*_{k,i}}{s_{k,i}}\Big) - \Phi\Big(\frac{T^i_{lower} - m^*_{k,i}}{s_{k,i}}\Big)\Big), \end{aligned} \tag{27}$$

where $m^*_{k,i} = m_{k,i} + s^2_{k,i}t$ and $t \in \mathbb{R}$.
The mean of random variable y is calculated through moment generating function (27) and takes the form:

$$\begin{aligned} \mathbb{E}(y) = \frac{\mathrm{d}M(0)}{\mathrm{d}t} = T^i_{lower,k}D^i_{lower,k} \\ + T^i_{upper,k}D^i_{upper,k} \\ + D^i_{un,k}(m_{k,i} + s_{k,i}l_{k,i}), \end{aligned} \tag{28}$$

where $l_{k,i} = \frac{\phi(T_{l^i_k}) - \phi(T_{u^i_k})}{D^i_{un,k}}$.

The i^{th} component of diagonal matrix $\mathbf{R}_{k,2}$ (measurements are independent) is the variance of random variable y and via (27) takes the form:

$$\begin{aligned} \mathrm{Var}(y) = \frac{\mathrm{d}^2M(0)}{\mathrm{d}t^2} - \Big(\frac{\mathrm{d}M(0)}{\mathrm{d}t}\Big)^2 \\ = {T^i_{lower}}^2 D^i_{lower,k}(1 - D^i_{lower,k}) \\ + {T^i_{upper}}^2 D^i_{upper,k}(1 - D^i_{upper,k}) \\ + m^2_{k,i}D^i_{un,k}(1 - D^i_{un,k}) + s^2_{k,i}D^i_{un,k} + s^2_{k,i}c_{k,i}D^i_{un,k} \\ + 2m_{k,i}s_{k,i}l_{k,i}(1 - D^i_{un,k})D^i_{un,k} \\ - s^2_{k,i}{l_{k,i}}^2 {D^i}^2_{un,k} - 2T^i_{lower,k}D^i_{lower,k}T^i_{upper,k}D^i_{upper,k} \\ - 2T^i_{lower,k}D^i_{lower,k}D^i_{un,k}(m_{k,i} + s_{k,i}l_{k,i}) \\ - 2T^i_{upper,k}D^i_{upper,k}D^i_{un,k}(m_{k,i} + s_{k,i}l_{k,i}), \end{aligned} \tag{29}$$

where $c_{k,i} = \frac{T_{l^i_k}\phi(T_{l^i_k}) - T_{u^i_k}\phi(T_{u^i_k})}{D^i_{un,k}}$.
The matrix $\mathbf{R}_{k,1}$ is proved to be equal to

$$\mathbf{R}_{k,1} = \mathbf{P}^-_k\mathbf{H}^T\mathbf{D}_{un,k} \tag{30}$$

Acknowledgment

This work was supported by the European Project: SURVANT http://survant-project.eu/ Grant no. 720417 within the H2020 FTIPilot-2015.

References

[1] M. Andriluka, S. Roth, and B. Schiele, "People-tracking-by-detection and people-detection-by-tracking," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.

[2] J. Son, M. Baek, M. Cho, and B. Han, "Multi-object tracking with quadruplet convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 5620–5629.

[3] B. Benfold and I. Reid, "Stable multi-target tracking in real-time surveillance video," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 3457–3464.

[4] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking," in *Image Processing (ICIP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 3464–3468.

[5] G. A. Mills-Tettey, A. Stentz, and M. B. Dias, "The dynamic hungarian algorithm for the assignment problem with changing costs," 2007.

[6] K. Loumponias, N. Vretos, P. Daras, and Tsaklidis, "Using tobit kalman filtering in order to improve the motion recorded by microsoft kinect," in *8th International Workshop on Applied Probabilities*, 2016.

[7] K. Loumponias, N. Vretos, P. Daras, and G. Tsaklidis, "Using kalman filter and tobit kalman filter in order to improve the motion recorded by kinect sensor ii," in *Proceedings of the 29th Panhellenic Statistics Conference*, 2016, pp. 322–334.

[8] A. C. Harvey, *Forecasting, structural time series models and the Kalman filter*. Cambridge university press, 1990.

[9] A. Savitzky and M. J. Golay, "Smoothing and differentiation of data by simplified least squares procedures." *Analytical chemistry*, vol. 36, no. 8, pp. 1627–1639, 1964.

[10] "Multiple object tracking benchmark 2d 2015," https://motchallenge.net/data/2D_MOT_2015/l, accessed: 2017-07-08.

[11] Y.-m. Song and M. Jeon, "Online multiple object tracking with the hierarchically adopted gm-phd filter using motion and appearance," in *Consumer Electronics-Asia (ICCE-Asia), IEEE International Conference on*. IEEE, 2016, pp. 1–4.

[12] F. N. Nguyen Thi Lan Anh, Furqan M.Khan and F. Bremond, "Multi-object tracking using multi-channel part appearance representation," in *In International conference on Advanced video and Signal Based Surveillance*. IEEE AVSS, 2017.

[13] K. Panta, D. E. Clark *et al.*, "An efficient track management scheme for the gaussian-mixture probability hypothesis density tracker," in *Intelligent Sensing and Information Processing, 2006. ICISIP 2006. Fourth International Conference on*. IEEE, 2006, pp. 230–235.

[14] B. S. Everitt, *Mixture DistributionsI*. Wiley Online Library, 1985.

[15] R. De Maesschalck, D. Jouan-Rimbaud, and D. L. Massart, "The mahalanobis distance," *Chemometrics and intelligent laboratory systems*, vol. 50, no. 1, pp. 1–18, 2000.

[16] A. Milan, S. H. Rezatofighi, A. R. Dick, I. D. Reid, and K. Schindler, "Online multi-target tracking using recurrent neural networks." in *AAAI*, 2017, pp. 4225–4232.

[17] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.

[18] A. C. Cohen, *Truncated and censored samples: theory and applications*. CRC press, 2016.

[19] L. Ljung, "Asymptotic behavior of the extended kalman filter as a parameter estimator for linear systems," *IEEE Transactions on Automatic Control*, vol. 24, no. 1, pp. 36–50, 1979.

[20] E. A. Wan and R. Van Der Merwe, "The unscented kalman filter for nonlinear estimation," in *Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000. AS-SPCC. The IEEE 2000*. Ieee, 2000, pp. 153–158.

[21] T. Kailath, *Linear systems*. Prentice-Hall Englewood Cliffs, NJ, 1980, vol. 156.

[22] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking," *IEEE Transactions on signal processing*, vol. 50, no. 2, pp. 174–188, 2002.

[23] D. R. Helsel *et al.*, *Nondetects and data analysis. Statistics for censored environmental data.* Wiley-Interscience, 2005.

[24] B. Allik, *The Tobit Kalman filter: an estimator for censored data*. University of Delaware, 2014.

[25] A. Neubeck and L. Van Gool, "Efficient non-maximum suppression," in *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, vol. 3. IEEE, 2006, pp. 850–855.

[26] R. Jonker and T. Volgenant, "Improving the hungarian assignment algorithm," *Operations Research Letters*, vol. 5, no. 4, pp. 171–175, 1986.

[27] S. Murray, "Real-time multiple object tracking-a study on the importance of speed," *arXiv preprint arXiv:1709.03572*, 2017.

[28] J. Ju, D. Kim, B. Ku, D. K. Han, and H. Ko, "Online multi-object tracking with efficient track drift and fragmentation handling," *JOSA A*, vol. 34, no. 2, pp. 280–293, 2017.

[29] L. Fagot-Bouquet, R. Audigier, Y. Dhome, and F. Lerasle, "Online multi-person tracking based on global sparse collaborative representations," in *Image Processing (ICIP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 2414–2418.

[30] F. Solera, S. Calderara, and R. Cucchiara, "Learning to divide and conquer for online multi-target tracking," *CoRR*, vol. abs/1509.03956, 2015. [Online]. Available: http://arxiv.org/abs/1509.03956

[31] J. Jaeyong, K. Daehun, K. Bonhwa *et al.*, "Online multi-person tracking with two-stage data association and online appearance model learning," *IET Computer Vision*, vol. 11, pp. 87–95(8), February 2017. [Online]. Available: http://digital-library.theiet.org/content/journals/10.1049/iet-cvi.2016.0068

[32] Y. Li, C. Huang, and R. Nevatia, "Learning to associate: Hybridboosted multi-target tracker for crowded scene," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 2953–2960.

[33] K. Bernardin and R. Stiefelhagen, "Evaluating multiple object tracking performance: the clear mot metrics," *Journal on Image and Video Processing*, vol. 2008, p. 1, 2008.