A Survey on Leveraging Deep Neural Networks for Object Tracking

Sebastian Krebs
Daimler AG
Research and Development
Ulm, Germany
Email: sebastian.krebs@daimler.com

Bharanidhar Duraisamy
Daimler AG
Research and Development
Ulm, Germany
Email: bharanidhar.duraisamy@daimler.com

Fabian Flohr
Daimler AG
Research and Development
Ulm, Germany
Email: fabian.flohr@daimler.com

***Abstract*—Object tracking is the task of estimating over time the state of a single or multiple objects based on noisy measurements received from one or several sensors. The field of object tracking spans over several application domains ranging from military radar systems and sensor fusion approaches, to today's computer vision tracking methods employed in consumer electronics and surveillance systems. It also plays a substantial role in autonomous driving. In recent years, the use of deep neural networks has spiked in various fields, due to their impressive performance in detection and classification tasks. This aspect also makes these methods applicable to object tracking. Therefore, the aim of this survey is to give the reader a brief yet comprehensive start into the widespread field of object tracking with a focus on the latest deep-learning based extensions and approaches. At first, traditional non-deep tracking systems are briefly reviewed and a generic model of the individual components of such systems is introduced. Based on this structure the representative deep-based tracking applications in the literature are classified and presented.**

I. Introduction

Starting in the early 1960's with the development of aerospace applications the research field of modern object tracking has grown steadily [1], [2]. In its simplest form, object tracking can be defined as the task of continuously estimating the trajectory of an object (also referred to as target) given the noisy measurements received from a sensor. However, over the past decades driven by the vast amount of different use cases and scenarios object tracking has evolved into various form [3], [4].

While single-target tracking focuses on the precise estimation of the targets state based on noisy measurements, the simultaneous tracking of multiple targets provides an additional challenge. During multi-object tracking the correspondences between new measurements and the whole set of already tracked targets need to be identified [5]. This NP-hard combinatorial problem is generally further complicated by the unknown number of targets, false measurements from the sensor caused by clutter, potentially new occurring objects, as well as noise.

The tracking of objects in video data has given rise to the research field of visual target tracking, which incorporates additional techniques from computer vision and machine learning [6], [7]. Lately the tracking-by-detection approach has become popular. Based on the results of an object detector, it tries to solve the spatio-temporal tracking problem by finding the correspondence between detections and already tracked targets. Using image frames as the input for the tracking task poses new challenges such as illumination changes, non-ridged objects, a changing appearance and/or pose, as well as occlusion. Therefore, some approaches in this field aim to learn the appearance of the tracked object in an online-manner.

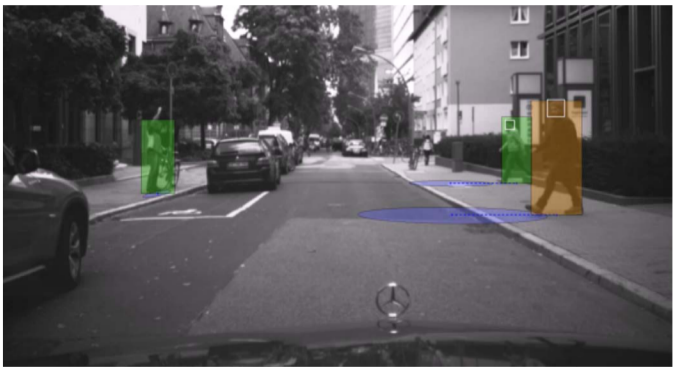


Fig. 1. A scene example encountered by an autonomous vehicle in an urban scenario. The detected pedestrians are marked by the colored bounding boxes. Based on the detections a tracking method obtains the needed kinematic information. The kinematic information of each target is used to predict the most likely position of the pedestrian in a one second time horizon, indicated by the blue ellipsoids.

The use of a robust tracking method is crucial especially in the context of intelligent vehicles and autonomous driving. Thereby, the output of object detectors can be robustified, which permits a more reliable perception of the environment including obstacles and other traffic participants. Furthermore, the tracking of targets over time renders the estimation of otherwise not directly observable variables possible. In the case of a vision-based pedestrian detection and tracking system the velocity and acceleration can be inferred over time solely from the detections. This kinematic information is particularly vital for a prediction of the future movement, as shown in Figure 1. By incorporating such predictions into modern intelligent vehicles, critical situations can be identified earlier.

Most recently, the use of deep learning based approaches has yielded an enormous performance increase in vision-based tasks such as image classification [8]. This caused the em-

978-1-5386-1526-3/17/$31.00 ©2017 IEEE

ployment of deep learning based techniques in various fields, including many of which are essential for the development of an autonomous vehicle: object detection [9], target tracking [10], and also action planning [11].

Therefore the aim of this work is to give the reader a comprehensive review of current state of the art tracking methods which utilize deep neural networks. After introducing a general component structure of traditional target tracking methods, numerous recent approaches employing deep learning are presented. These range from early attempts which use the networks as a black box to extract robust visual features, over networks used as a similarity measure to solve the data association, to the use of networks to perform the prediction. Also, several end-to-end tracking approaches are further described. All the above mentioned approaches have in common that the features are learned in a bottom-up manner directly from data that instead of using heuristically hand-crafted features.

II. Related Work

Although the aim of this survey is to give the reader an introduction into the topic of *leveraging deep learning methods for the task of object tracking*, a comprehensive survey on the topic of deep learning itself is beyond the scope of this work. However, there are several excellent recent surveys [12]–[15] which review deep learning in general.

Several other publications [3]–[6], [10], [16] focus on an introduction and summary to the challenging problem of tracking. Both of the surveys [3], [6] focus on visual tracking methods using different low-level features or statistical learning techniques. The work from Yilmaz et al. [3] is an extensive survey of object tracking methods up to the year 2006. The authors categorize the tracking methods based on the employed object and motion representations, provide detailed descriptions of representative methods in each category, and examine their pros and cons.

Li et al. [7] focus on a more intensive analysis of various 2D appearance models for visual object tracking. The literature for visual representations is reviewed from a feature-construction viewpoint, by a hierarchical classification into global and local features. Further the model-construction mechanisms are distinguished into generative, discriminative and hybrid ones.

In the more recent work from Vo et al. [5], different methods for online multi-target tracking are reviewed. The authors thereby focus on three major techniques used most widely to resolve the measurement to track correspondence problem, namely the Joint Integrated Probabilistic Data Association, Multi Hypothesis Tracking and Random Finite Sets. Besides an introduction including Bayesian Estimation and the Kalman and Particle Filter, the article from Vo et. al. especially reviews the mathematical background of the methods as well as different method variations.

In the experimental survey work of Smeulders et al. [4] 19 different visual single-target trackers were systematically and experimentally evaluated. The test dataset included 315 different video fragments covering various challenging scenarios such as illumination changes, occlusion, and clutter.

The work of Feng et al. [10] is one of the most recent and most closely related to this paper. The authors introduce the utilization of deep learning to perform the task of visual tracking. After a brief introduction of the basic concepts of deep learning based visual tracking the authors focus on three representative applications, namely the Deep Learning Tracker (DLT) [17], the Fully Convolutional Network Based Tracker (FCNT) [18], and lastly the Multi-Domain Network (MDNet) [19].

Finally the work from Wang et al. [16] divides a common tracking system into five constituent parts and investigates the influence of each part. The authors state that each tracker consists of the motion model, feature extractor, observation model, model updater, and ensemble post-processing. One of the main findings of the paper is the outcome, that the feature extractor plays the most crucial role, whereas the observation model has practically no influence on the tracker performance.

One major building block in todays research in the field of machine learning and computer vision is the ability to draw a quantitative comparison between different state of the art methods based on standardized datasets and evaluation criterion. Furthermore, one of the most indispensable components of each deep learning based method is the data which is available to train the network. Thus, the following presents the most important and prevalent tracking datasets and benchmarks.

The Visual Object Tracking (VOT) challenge focuses on a precisely defined and repeatable way of comparing short-term trackers on a yearly basis [1]. However, the employed tracking methods should not utilize pre-trained models, which renders the benchmark unsuitable for deep learning based tracking methods.

On the contrary, the Multi-Object Tracking Challenge (MOT Challenge) [2] also provides a separate set of sequences which can be used to train model-based approaches a-priori. The results of the MOT Challenge are also reviewed on a yearly basis, focusing on multi-object trackers for surveillance and automotive use-cases.

Another dataset and benchmark which focuses on the automotive use case is the comprehensive Kitti Vision Benchmark Suite [20] which also includes an object tracking part for cars and pedestrians. The dataset includes the images in conjunction with additional information, such as lidar data, and the ego motion of the recording vehicle.

To summarize, there are already several tracking surveys available. Some focus on general object tracking methods developed and publicized [3], [6], while others review one particular part like the data association or appearance models in more detail [5], [7]. The focus on deep learning based tracking approaches differentiates this work from the that mentioned above. Compared to the review of Feng et al. [10] a wider variety of different approaches is examined in a more general way.

III. Traditional Object Tracking

The first target tracking systems emerged primarily from military applications such as radar or sonar tracking [21]. In general, the aim of these algorithms is to estimate the current state x_i^t at time t of the i-th object, by using a set of noisy measurements $Z^t = \{z_1, \ldots, z_n\}$. These tracking methods can be separated roughly into the four coherent parts of: Data Association, State Update, State Prediction, and Track Management, as illustrated in Figure 2. Yet these parts are often interconnected, dependent on additional parts, or are bundled, making it non-trivial or impossible to draw explicit lines.

The **Data Association** describes the problem of finding the correct correspondences between the set of already known tracked targets and the ensemble of new possible noisy measurements. One of the first and still primary methods for performing the data association in a general multi-object tracking scenario includes the Multi Hypothesis Tracking (MHT) [22] and the Joint Integrated Probabilistic Data Association (JIPDA) [21]. In the case of visual tracking-by-detection approaches, these measurements are already preprocessed signals, such as the bounding box of the desired objects in the image. Especially for visual tracking the use of an additional appearance model, which encapsulates the visual information of the tracked object has yielded significant performance increases during data association, e.g. see [23]–[26].

Once the correspondences between measurements and targets is calculated the **State Update** can be performed. During the update of the objects state the uncertainty of the current state as well as the potential sensor noise have to be taken into account. One of the most utilized and popular methods to solve this estimation in a recursive manner is the Kalman Filter (KF) [27]. The Kalman Filter is a closed-form solution of the Bayes Estimation Theorem usually using Gaussian distributions, which is optimal for linear problems. In the presence of non-linear problems, several extensions of this approach have been presented, namely the Extended Kalman Filter (EKF) and the Unscented Kalman Filter (UKF). For non-linear and/or non-Gaussian systems the Particle Filter (PF) has become the method of choice. PF refers to a set of sequential Monte Carlo methods to approximate a numerical solution to the Bayes Recursion. The estimated probability distribution is approximated using a set of random particles. Besides using solely the PF to estimate the state distribution of the object, plenty of published work exists which incorporates the data association into the filtering process [28], [29]. These Random Finite Set (RFS) based tracking techniques model the whole world state space as a multi-modal probability distribution, which is approximated by the PF. The data association is implicitly performed during the sampling process, by using the measurements as new particles.

Closely related and usually entangled in the State Update is the **State Prediction**. Most trackers work in discretized time cycles. Since the world is not static and usually either the object position, the sensor position, or both change over time, these changes need to be countervailed between the tracking cycles. Therefore, the targets state is predicted from the latest update time-stamp to the time the new measurements arise. These predictions are mostly derived from the kinematic information of the object along with some a-priori knowledge about the targets movement [30]. The choice of the employed motion model heavily depends on the targeted use-case and the object type to be tracked. Predicting the state from a later time entails an increase in the overall state uncertainty.

Lastly, the **Track Management** embraces the logic applied to create new tracks and delete obsolete ones. In general, the track management exerts an influence on the data association process [31], [32], since measurements that cannot be linked to an existing track should give rise to a new one (without creating false positives from clutter). Furthermore, heuristics such as finite-state machines based on the track age are used to determine the track-state, and finally the point of deletion.

IV. Deep Learning for Tracking

A. General

Using deep learning for the task of object tracking can take various forms and approaches. Based on the model presented in Section III, the remainder of this section reviews state of the art deep learning based tracking approaches and classifies them into: feature, data association, prediction, and end-to-end learning.

Emerging from the field of visual tracking, the first deep-based approaches leveraged the robust feature representations learned by the networks to better model the tracked objects. These approaches [17], [33]–[36] often use deep neural networks as a black-box feature extractor and perform the subsequent classification (same-object, other-object), association and filtering with more traditional methods.

To overcome the challenges caused by the non-deep successive classification and association parts, some approaches [37]–[39] aim to solve the correspondences problem in a deep learning manner. These methods mostly employ Siamese networks [40] to directly learn a similarity measure given two input patches containing the possible object match candidates. Based on the calculated match probabilities, well-known methods such as the Hungarian algorithm [41] can calculate the optimal correspondences.

Further leveraging the power of deep neural networks to capture non-linear dependencies directly from the data, some methods [42]–[44] focus on learning to predict the movement of objects. The knowledge of the networks which was gathered during the offline training is used to estimate the most likely future positions of all objects given their history.

Lastly, by formulating tracking as an end-to-end problem some approaches introduce networks which jointly optimize the whole tracking process. These works [45]–[49] unite all above-mentioned traditional components of a traditional tracking pipeline into one network.

Recently, most research groups have been trying to focus on the modeling and integration of the temporal aspects and

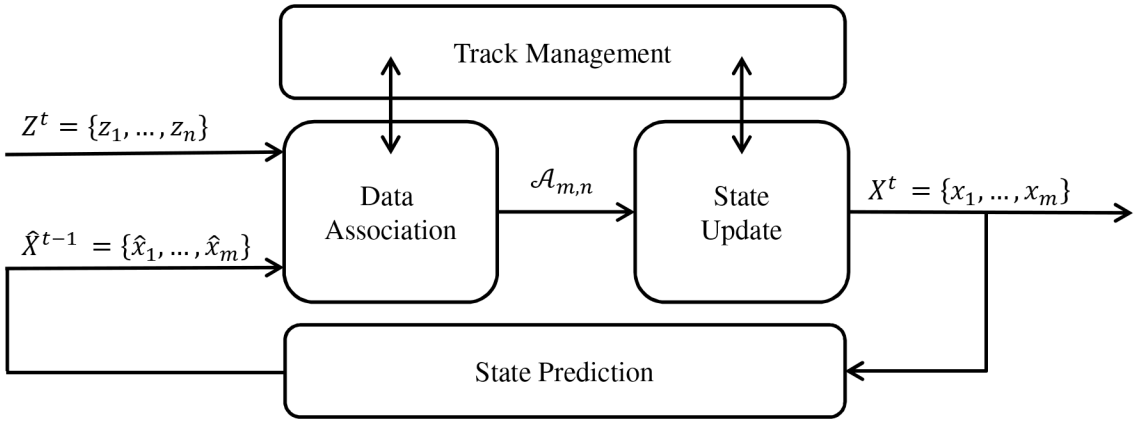


Fig. 2. Schematic illustration of the single components of a generic traditional multi-object tracking framework. The set of n new measurements Z^t at each time t is used to update the predicted states $\hat{X}^{t-1}$ of the m already tracked objects, based on the calculated association matrix $\mathcal{A}_{m,n}$.

dependencies. Recurrent neural networks, which have a built-in feedback loop acting as network memory, have therefore become popular. Especially Long Short Term Memory (LSTM) [50] is utilized as an advanced recurrent network type to further improve the tracking tasks.

B. Using Deep Learning for Feature Learning

First approaches are aiming at leveraging the powerful and robust image features learned by deep neural networks to generate a meaningful model of the tracked object.

One of the first approaches into using deep learning for object tracking is the Deep Learning Tracker (DLT) introduced by Wang and Yeung [17] in 2013. In their work the authors train a stacked Denoising Autoencoder (DAE) offline on a large dataset of auxiliary natural images to obtain a generic object feature representation which is transferred to the online tracking process. The use of a DAE increases the robustness of the learned image features, compared to a standard Auto Encoder (AE). For the online tracking process, the encoder part of the stacked DAE is used as a pre-trained feature extractor extended by a sigmoid classification layer. The DLT uses a particle filter as the motion model and to perform the state estimation. During tracking for each frame, a set of particles is drawn from the currently estimated patch. The confidence of each particle is calculated by making a simple forward pass through the network. If the sum of the calculated confidences for a frame falls below a pre-defined threshold, the network needs to be re-tuned again to cope with the changing appearance of the tracked object.

However, the employed network structure limits the input image size, causing blurry features, and the linear classification layer does not yield optimal results.

To overcome these issues, Wang et al. presented the Structured Outcome Deep Learning Tracker (SO-DLT) [33] which is an enhancement of the DLT. Instead of the stacked DAE a novel structured output CNN is used during pre-training to distinguish between objects and background. The CNN outputs a pixel-wise probability map indicating whether the pixel belongs to the bounding box of an object. Especially due to the classification performance of the CNN, the results of the SO-DLT are superior to those of the DAE-based DLT approach.

In their work, Ma et al. also use a CNN [36] which is trained offline on the large-scale ImageNet dataset with category-level labels. The authors state in their paper that higher level convolutional layer in the CNN encodes the semantic object information. However, their spatial resolution is too coarse to precisely infer the target location. The feature maps from lower level layers provide more precise position information, but lack the semantic information needed for appearance-invariance. Thus, reasoning with multiple layers of different levels of CNN features is essential for target tracking. In [36], three different convolutional layers from the pre-trained VGG-net [51] are used. The outputs of these layers are implement as multi-channel features to learn an adaptive linear correlation filter per layer. During the online tracking the generated feature maps of each layer are convolved with the respective learned correlation filter to obtain a response map. Using a coarse-to-fine search approach the object location is inferenced based on the calculated multi-level response maps.

Similarly, Wang et al. conduct an in-depth feature analysis on their ImageNet pre-trained VGG-based CNN in their work [34] with the same findings. Further, the authors state that only a subset of the neurons is relevant for tracking a specific target. Therefore, a tracking framework is introduced with a feature map selection method that avoids irrelevant feature maps as well as over-fitting on noisy ones. The selected feature maps are jointly considered at two different levels, by two separate networks (the general network GNet and the specific network SNet). Both networks perform a heat map regression to infer the object location.

In the work of Hong et al. [52] a pre-trained CNN is used for target localization and feature extraction. The extracted features are classified by an online learned Support Vector Machine (SVM). The features of each positive classified

sample are back-propagated through the network to obtain a saliency map. By combining the saliency maps produced by each sample, an object-specific spatial saliency map can be calculated. Tracking is performed by sequential Bayesian filtering using the object-specific map as observation along with an online learned appearance model, resulting in the final object probability map. Based on the calculated track, new samples are extracted which are employed for the online-training of the SVM.

C. Using Deep Learning for Data Association

One of the major challenges of multi-object tracking is the task of data association. Deep-based approaches can be applied to learn a generic matching function directly from the data, thus bypassing the creation of heuristic hand-crafted features. The most popular approach to learn such a similarity metric from the data are Siamese Networks. These twin-networks consist of two identical sub-networks joined at the output.

Toa et al. introduce the Siamese INstance Search Tracker (SINT) [37]. The main concept of the SINT is the use of a two-streamed Siamese deep neural network, which is explicitly designed for tracking. During the training phase, the network is provided with two data streams: the search and the query stream. The query stream provides a video frame with the exact position of the tracked object, whereas another - not necessarily adjacent - frame is given by the search stream with randomly sampled object locations. The random object proposals are used as positive training samples if they exceed a certain overlap-threshold with the desired object, otherwise they are used as negative samples. Thus, the Siamese network learns a generic measure, which for a given pair of input images expresses the probability that the patches contain the same object. During tracking, the position of the object in the first frame is given. This initial position is used to extract a template of the tracked object. During the entire tracking sequence, this template is utilized to calculate the similarity of the new candidate regions. The candidate regions are created using a radius sampling approach. Finally, the region with the highest similarity is considered the new object position and the box positions are further refined by four Ridge regressors.

Leal-Taixe et al. [53] also leverage the ability of a twin-architecture Siamese network to learn a similarity measure directly from data. In their work, the authors focus on pedestrian tracking. The network is trained to learn local spatio-temporal features to distinguish people, by aggregating pixel values with optical flow. Further, the output of the learned network is extended by a set of contextual features - which capture the relative position and size change of the objects, as well as their relative velocity - using gradient boosting. The learned likelihood function is used by a modified linear programming approach to obtain the final multi-object pedestrian tracker.

Instead of using the optical flow to model the spatio-temporal dependencies, Varior et al. use a recurrent version of the Siamese network [38]. The introduced LSTM-based Siamese network can memorize spatial dependencies over time, and is further able to selectively propagate relevant information. This enhances the discriminate capabilities of the network in the context of human re-identification, and optimizes the employed contractive loss function (inter-pair distance increases, whereas intra-pair distance declines).

D. End-to-End Learning for Tracking

The superior results of most deep-learning based visual detection methods can be attributes to the fact, that instead of using hand-crafted features, the features are learned directly from the data. By using an end-to-end approach this could also enable a performance boost for the task of object tracking, since the full tracking pipeline - object representation, object extraction and location prediction is jointly tuned to maximize the overall tracking quality.

By using a recurrent neural network Gan et al. [45] present an anonymous model-free visual object tracking approach. The network is completely trained offline with synthesized data which simulates a moving object with changing velocity, acceleration and direction. The recurrent neural network is used to capture the spatio-temporal dependencies and fuse past predictions with their corresponding visual features. During the training phase the artificial raw video frames are used as input, along with the ground-truth location of the desired object in the first frames. This implicitly enforces a model-free anonymous tracking, since no further knowledge of the object is given. The network calculates the estimated bounding box for the tracked target for each frame as its output.

A similar approach is presented by Held et al. with the Generic Object Tracking Using Recurrent Networks (GO-TURN) [46]. The network is trained entirely offline to learn a generic relationship between motion and appearance of novel objects. As stated by the authors, the network is capable of tracking an unknown object with up to 100 frames per second (fps), making it the fastest deep-based tracking method for single-object tracking today present.

Another end-to-end based approach is presented by Nam and Han, called the Multi-Domain Network (MDNet) [19]. The essence of the work is to separate domain-specific knowledge from domain-independent one. Therefore, the MDNet is composed of one shared layer branch which is utilized to learn a generic feature representation employable to tracking, and several domain-specific branches. Each domain-specific binary classification branch is trained only with a set of videos containing objects of the same class. To obtain the final tracking network, all binary domain branches are removed and replaced by a new classification layer, which is trained online during tracking to adapt to the desired domain.

Ning et al. extend the object detection network You Only Look Once (YOLO) [54] in a recurrent fashion, creating the Recurrent YOLO (ROLO) [48]. Their proposed network concatenates the high-level features produced by the general object detection network with a Long-Short Term Memory (LSTM) to model the temporal relation along with the spatial information. Thus ROLO directly allows end-to-end spatio-temporal regression of either object coordinates or heat maps. The authors state that the network is ”double deep” (temporal

and spatial) since it examines the history of all previous object locations as well as the former robust visual features. The detection part of the network is pre-trained on the ImageNet dataset, whereas the training of the tracking module is performed in a second stage with the ADAM stochastic optimization method.

The online multi-target tracking framework presented by Milan et al. [49] is closely related to traditional target tracking and can solve the data association, state estimation and prediction, as well as the creation and deletion of tracks. The network consists of a traditional recurrent neural network part which is employed for state update, state prediction, and existence probability estimation. The state space is modeled as multi-dimensional with continuous and discrete variables. While the recurrent neural network performs well for the above mentioned tasks it cannot properly handle the combinatorial problem of data association. Thus, an LSTM-based architecture capable of solving the highly complex one-to-one assignment problem entirely from offline training is designed.

Contrary to most approaches Posner and Ondruska assume a complete yet uninterpretable state of the world for their Deep Tracking [47] framework. By leveraging RNNs the authors are able to train the network end-to-end with partial observations of dynamic scenes. The offline training is performed using a novel dropout method, in which observations are dropped spatially across the entire scene, as well as temporally across multiple time steps. Based on the raw sensor measurements obtained from a 2D laser-sensor, the network is able to predict the unoccluded state of the entire scene as an occupancy grid map based representation. The approach has proven applicable to synthetic data.

The Deep Tracking architecture was extended by Onduska et al. [55]. To increase the performance especially for challenging real world scenarios, multi-scale convolutions are used to cope with objects of different sizes. Further static memory is employed to learn spatial information, and dynamic memory is applied so that information can be used for a longer period of time. Finally, due to inductive transfer, the tracking features learned during training are used to extend the network to be able to classify the tracked objects. The resulting recurrent neural network, tailored to tracking in complex and dynamic scenarios, is empirically evaluated with real-world data collected at a busy road intersection.

Finally, the Deep Tracking framework was further extended by Dequaire et al. to be applicable to tracking objects from an autonomous vehicle [56]. Since the vehicle itself moves in the world, the ego motion needs to be corrected and decoupled from the objects' motion. This is achieved by introducing a Spatial Transformer Module (STM) in the hidden state of the network, which performs a transformation on all feature maps based on an external received ego motion.

E. Using Deep Learning for Prediction

Traditional object tracking systems usually rely on an empirically or heuristically created motion model to predict the target's position between time stamps. Since the motion model is highly dependent on the object class (e.g.: intercontinental ballistic missiles require another motion model than humans) it seems straightforward to learn those motion models directly from the data.

To this extent, Li et al. propose the Behavior-CNN [43] to model the kinematic properties of pedestrians in crowded scenes. The employed CNN is trained with real-world data from a surveillance scenario involving a static camera. The authors perform an in-depth investigation on different aspects of the learned Behavior-CNN. They state that the learned location maps represent the location awareness of the network for the learned scene (note that this is only possible due to the static sensor pose). Furthermore, the feature maps of the trained layers are examined, resulting in the low-level layers capturing simple motion patterns such left or up movements, while higher-level layers capture more semantic object movements which also influence other objects. Lastly, the impact of the receptive field size is reviewed, which is crucial to capture inter-object dependencies.

Alahi et al. also focus on the prediction of human trajectories in multi-person scenarios. Therefore, they introduce the Social-LSTM [42] which interprets the task of predicting the future trajectories of multiple persons as a sequence generation problem. For each person present in the scene a single LSTM is utilized to predict the future trajectory based on the pre-processed trajectory history. To also be able to capture inter-class dependencies, a novel social pooling layer is introduced, which enables the Social-LSTM to capture connections between multiple spatially proximal persons. This renders the LSTM-based approach feasible to jointly predict multiple human trajectories precisely.

Lastly, Hoermann et al. focus in their work on the prediction of a Dynamic Occupancy Grid Map (DOGMa) for an autonomous vehicle in an urban driving scenario using deep learning [44]. The DOGMa is used as an intermediate time-filtered environment representation obtained from the raw sensor measurements from multiple sensors. The map is a probabilistic grid-based representation of the surrounding static and dynamic objects, with probability of allocation for each cell. Using an image-like representation of the map as input for the employed neural network allows predicting the perceivable scene. During training, a fully automatic label generation is used to overcome the need for annotating the recorded real-world data manually. It is shown that the trained network can correctly predict complex scenarios at an urban intersection involving multiple road users (pedestrians, cars, cyclists) with different maneuver classes.

V. Conclusion

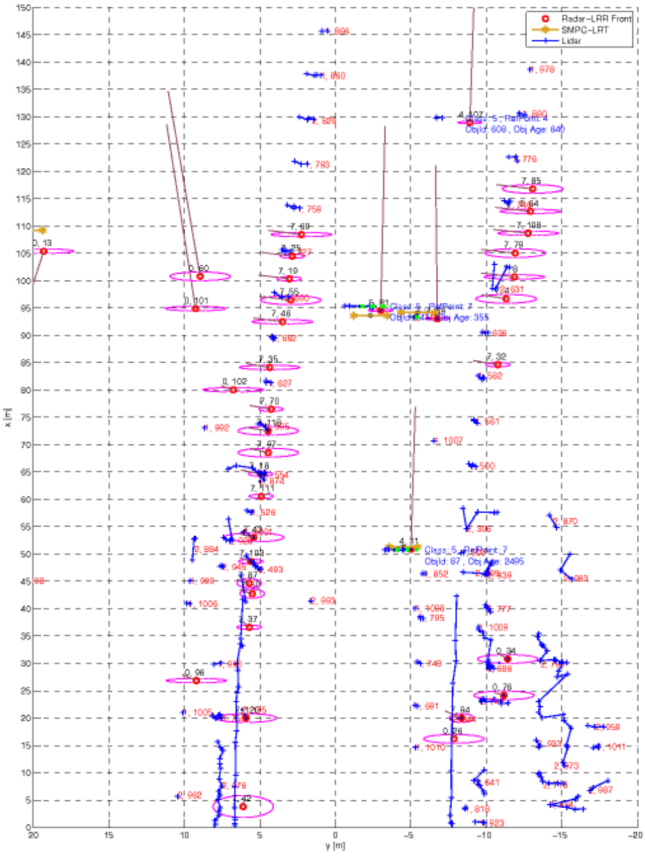
This article covered different and various representative work in the field of object tracking using deep-learning based techniques. It was shown that a substantial amount of early deep-based tracking approaches were tailored by their historically-related vision-based classification and detection tasks. These deep visual trackers lately often cast the problem of tracking a target as a spatio-temporal detection

in a sequence of multiple adjacent frames [19], [46], [48]. However, although yielding state-of-the-art results in various benchmarks, in the context of autonomous driving these methods lack the explicit modeling of the kinematic state of the tracked objects, that is crucial for environment perception and subsequent motion planning. In contrast, in the work presented by Milan et al. [49] the state space is explicitly designed like in more traditional tracking methods.

Another remaining challenge is the integration into the networks of non-image based input from possible multiple sensors with different physical measurement methods. An example of real world data gathered from three different sensors is shown in Figure 3. As presented by Posner et al. [47] and Hoermann et al. [44] using an intermediate pre-filtered occupancy grid map representation that can be processed in an image-like manner appears promising.

Overall, compared to datasets such as the ImageNet [57] for detection and classification tasks, or the Cityscapes dataset [58] for semantic scene understanding of urban scenarios, it was shown that the tracking community lacks a dataset offering a huge amount of training data tailored for deep learning tracking methods. Finally, most work currently focuses on the adoption of recurrent neural networks to allow implicitly learning the temporal dependencies during tracking directly in a bottom-up fashion from the data itself.

(a) Camera image corresponding to the scene in Figure 3b



(b) Birdseye view of the tracked objects in the scene shown in Figure 3a. The red dots are radar objects, the blue objects are measurements received from the lidar sensor, and the gold objects are detected by a stereo camera. The green lines represent the associated, fused and tracked targets which belong to the class *car*.

Fig. 3. Real world data recorded with an autonomous prototype vehicle. Figures from [23]

References

[1] M. Kristan, R. Pflugfelder, A. Leonardis, J. Matas, L. Čehovin, G. Nebehay, T. Vojívr, G. Fernandez, and Others, "The visual object tracking VOT2014 challenge results," in *European Conference on Computer Vision (ECCV)*, 2014, pp. 1–27.

[2] A. Milan, L. Leal-Taixe, I. Reid, S. Roth, and K. Schindler, "MOT16: A Benchmark for Multi-Object Tracking," *arXiv preprint arXiv:1603.00831*, mar 2016.

[3] A. Yilmaz, O. Javed, and M. Shah, "Object tracking," *ACM Computing Surveys*, vol. 38, no. 4, pp. 13–es, 2006.

[4] A. W. M. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah, "Visual tracking: An experimental survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 7, pp. 1442–1468, 2014.

[5] B.-n. Vo, S. Singh, A. Doucet, and M. Carlo, "Multi-Target Tracking," *Wiley Encyclopedia of Electrical and Electronics Engineering*, 2015.

[6] K. Cannons, "A Review of Visual Tracking," Tech. Rep., 2008. [Online]. Available: http://www.cse.yorku.ca/techreports/2008/CSE-2008-07.pdf

[7] X. Li, W. Hu, C. Shen, Z. Zhang, A. Dick, and A. van den Hengel, "A Survey of Appearance Models in Visual Object Tracking," *ACM Transactions on Intelligent Systems and Technology*, vol. 4, no. 4, pp. 1–42, 2013.

[8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," pp. 1097–1105, 2012. [Online]. Available: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks

[9] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single Shot MultiBox Detector," in *European Conference on Computer Vision (ECCV)*, dec 2015.

[10] X. Feng, W. Mei, and D. Hu, "A Review of Visual Tracking with Deep Learning," *Advances in Intelligent Systems Research*, vol. 133, pp. 231–234, 2016.

[11] M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, X. Zhang, J. Zhao, and K. Zieba, "End to End Learning for Self-Driving Cars," *arXiv preprint arXiv:1604.07316*, apr 2016.

[12] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, 2015.

[13] D. Li, "A tutorial survey of architectures , algorithms „," *APSIPA Transactions on Signal and Information Processing*, vol. 3, no. 2014, pp. 1–29, 2014.

[14] L. Deng and D. Yu, "Deep Learning: Methods and Applications," *Foundations and Trends in Signal Processing*, vol. 7, no. 3-4, pp. 197–387, 2014.

[15] J. Schmidhuber, "Deep Learning in neural networks: An overview,"

Neural Networks, vol. 61, pp. 85–117, 2015.

[16] N. Wang, J. Shi, D. Y. Yeung, and J. Jia, "Understanding and diagnosing visual tracking systems," in *Proceedings of the IEEE International Conference on Computer Vision*, vol. 11-18-Dece. IEEE, dec 2016, pp. 3101–3109.

[17] N. Wang and D.-Y. Yeung, "Learning a Deep Compact Image Representation for Visual Tracking," in *Proceedings of the 26th International Conference on Neural Information Processing Systems*, 2013, pp. 809–817.

[18] L. Wang, W. Ouyang, X. Wang, and H. Lu, "Visual Tracking with Fully Convolutional Networks," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3119–3127, 2015.

[19] H. Nam and B. Han, "Learning Multi-domain Convolutional Neural Networks for Visual Tracking," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 4293–4302.

[20] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the KITTI vision benchmark suite," *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3354–3361, 2012.

[21] Y. Bar-Shalom, F. Daum, and J. Huang, "The Probabilistic Data Association Filter," *IEEE Control Systems Magazine*, no. December, pp. 82—-100, 2009.

[22] S. Blackman, "Multiple hypothesis tracking for multiple target tracking," *IEEE Aerospace and Electronic Systems Magazine*, vol. 19, no. 1, pp. 5–18, jan 2004.

[23] B. Duraisamy, T. Schwarz, and C. Woehler, "On track-to-track data association for automotive sensor fusion," in *Information Fusion (Fusion), 2015 18th International Conference on*, 2015.

[24] W. Choi, "Near-Online Multi-target Tracking with Aggregated Local Flow Descriptor," *IEEE International Conference on Computer Vision (ICCV)*, pp. 3029–3037, apr 2015.

[25] S.-H. Bae and K.-J. Yoon, "Robust Online Multi-object Tracking Based on Tracklet Confidence and Online Discriminative Appearance Learning," in *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, jun 2014, pp. 1218–1225.

[26] C.-H. Kuo, C. Huang, and R. Nevatia, "Multi-target tracking by on-line learned discriminative appearance models," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, jun 2010, pp. 685–692.

[27] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of basic Engineering*, vol. 82, pp. 35—-45, 1960.

[28] D. Clark, K. Panta, and B.-n. Vo, "The GM-PHD Filter Multiple Target Tracker," in *9th International Conference on Information Fusion*. IEEE, jul 2006, pp. 1–8.

[29] S. Reuter, Ba-Tuong Vo, Ba-Ngu Vo, and K. Dietmayer, "The Labeled Multi-Bernoulli Filter," *IEEE Transactions on Signal Processing*, vol. 62, no. 12, pp. 3246–3260, jun 2014.

[30] X. R. Li and V. P. Jilkov, "Survey of maneuvering targettracking . part I: dynamic models," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1333–1364, oct 2003.

[31] K. Panta, D. E. Clark, and B.-N. Vo, "Data Association and Track Management for the Gaussian Mixture Probability Hypothesis Density Filter," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 45, no. 3, pp. 1003–1016, jul 2009.

[32] S.-H. Bae and K.-J. Yoon, "Robust online multiobject tracking with data association and track management." *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society*, vol. 23, no. 7, pp. 2820–33, jul 2014.

[33] N. Wang, S. Li, A. Gupta, and D.-Y. Yeung, "Transferring Rich Feature Hierarchies for Robust Visual Tracking," *arXiv preprint arXiv:1501.04587*, jan 2015.

[34] L. Wang, W. Ouyang, X. Wang, and H. Lu, "Visual Tracking with Fully Convolutional Networks," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 3119–3127.

[35] Z. Chi, H. Li, H. Lu, and M.-H. Yang, "Dual Deep Network for Visual Tracking," *IEEE Transactions on Image Processing*, vol. 26, no. 4, pp. 2005–2015, dec 2017.

[36] C. Ma, J. B. Huang, X. Yang, and M. H. Yang, "Hierarchical convolutional features for visual tracking," in *Proceedings of the IEEE International Conference on Computer Vision*, vol. 11-18-Dece, 2016, pp. 3074–3082.

[37] R. Tao, E. Gavves, and A. W. M. Smeulders, "Siamese Instance Search for Tracking," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[38] R. R. Varior, B. Shuai, J. Lu, D. Xu, and G. Wang, "A Siamese Long Short-Term Memory Architecture for Human Re-Identification," *European Conference on Computer Vision (ECCV)*, pp. 135–153, jul 2016.

[39] B. Li, C. Yang, and G. Xu, "Multi-pedestrian tracking based on feature learning method with lateral inhibition," in *IEEE International Conference on Information and Automation*, 2015.

[40] S. Chopra, R. Hadsell, and L. Y., "Learning a similiarty metric discriminatively, with application to face verification," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 349–356, 2005.

[41] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, no. 1-2, pp. 83–97, mar 1955.

[42] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social LSTM: Human Trajectory Prediction in Crowded Spaces," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2016, pp. 961–971.

[43] S. Yi, H. Li, and X. Wang, "Pedestrian Behavior Understanding and Prediction with Deep Neural Networks," in *European Conference on Computer Vision (ECCV)*, 2016, pp. 263–279.

[44] S. Hoermann, M. Bach, and K. Dietmayer, "Dynamic Occupancy Grid Prediction for Urban Autonomous Driving: A Deep Learning Approach with Fully Automatic Labeling," *arXiv:1705.08781*, may 2017.

[45] Q. Gan, Q. Guo, Z. Zhang, and K. Cho, "First Step toward Model-Free, Anonymous Object Tracking with Recurrent Neural Networks," *arXiv preprint arXiv:1511.06425*, nov 2015.

[46] D. Held, S. Thrun, and S. Savarese, "Learning to track at 100 FPS with deep regression networks," *Lecture Notes in Computer Science*, vol. 9905 LNCS, pp. 749–765, 2016.

[47] I. Posner and P. Ondruska, "Deep Tracking: Seeing Beyond Seeing Using Recurrent Neural Networks," *Proceedings of the 30th Conference on Artificial Intelligence (AAAI 2016)*, pp. 3361–3367, 2016.

[48] G. Ning, Z. Zhang, C. Huang, Z. He, X. Ren, and H. Wang, "Spatially Supervised Recurrent Convolutional Neural Networks for Visual Object Tracking," *arXiv preprint arXiv:1607.05781*, jul 2016.

[49] A. Milan, S. H. Rezatofighi, A. Dick, K. Schindler, and I. Reid, "Online Multi-target Tracking using Recurrent Neural Networks," *Arxiv*, 2016.

[50] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, nov 1997.

[51] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," sep 2014.

[52] S. Hong, T. You, S. Kwak, and B. Han, "Online Tracking by Learning Discriminative Saliency Map with Convolutional Neural Network," in *International Conference on International Conference on Machine Learning (ICML)*, 2015, pp. 597–606.

[53] L. Leal-Taixé, C. Canton-Ferrer, and K. Schindler, "Learning by tracking: Siamese CNN for robust target association," *Arxiv*, apr 2016.

[54] D. Impiombato, S. Giarrusso, T. Mineo, O. Catalano, C. Gargano, G. La Rosa, F. Russo, G. Sottile, S. Billotta, G. Bonanno, S. Garozzo, A. Grillo, D. Marano, and G. Romeo, "You Only Look Once: Unified, Real-Time Object Detection," *Nuclear Instruments and Methods in Physics Research*, vol. 794, pp. 185–192, jun 2015.

[55] P. Ondruska, J. Dequaire, D. Z. Wang, and I. Posner, "End-to-End Tracking and Semantic Segmentation Using Recurrent Neural Networks," *arXiv*, 2016.

[56] J. Dequaire, D. Rao, P. Ondruska, D. Wang, and I. Posner, "Deep Tracking on the Move: Learning to Track the World from a Moving Vehicle using Recurrent Neural Networks," *arXiv preprint arXiv:1609.09365*, vol. arXiv prep, 2016.

[57] Jia Deng, Wei Dong, R. Socher, Li-Jia Li, Kai Li, and Li Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2009, pp. 248–255.

[58] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The Cityscapes Dataset for Semantic Urban Scene Understanding," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 3213–3223.