ODANet: Online Deep Appearance Network for Identity-Consistent Multi-Person Tracking

Guillaume Delorme, Yutong Ban, Guillaume Sarrazin, Xavier Alameda-Pineda

ODANet: Online Deep Appearance Network for Identity-Consistent Multi-Person Tracking

Guillaume Delorme[1], Yutong Ban[2], Guillaume Sarrazin[1]
and Xavier Alameda-Pineda[1]

[1]Inria, Univ. Grenoble-Alpes, LJK, CNRS, [2]MIT CSAIL Distributed Robotics Lab

Abstract. The analysis of effective states through time in multi-person scenarii is very challenging, because it requires to *consistently* track all persons over time. This requires a robust visual appearance model capable of re-identifying people already tracked in the past, as well as spotting newcomers. In real-world applications, the appearance of the persons to be tracked is unknown in advance, and therefore on must devise methods that are both discriminative and flexible. Previous work in the literature proposed different tracking methods with fixed appearance models. These models allowed, up to a certain extent, to discriminate between appearance samples of two different people. We propose an online deep appearance network (ODANet), a method able to simultaneously track people and update the appearance model with the newly gathered annotation-less images. Since this task is specially relevant for autonomous systems, we also describe a platform-independent robotic implementation of ODANet. Our experiments show the superiority of the proposed method with respect to the state of the art, and demonstrate the ability of ODANet to adapt to sudden changes in appearance, to integrate new appearances in the tracking system and to provide more identity-consistent tracks.

1 Introduction

Traditionally, the analysis of the affective states of people has been done using individual data: meaning that only one person is present in the data stream. In the recent past, various papers on group emotion recognition, meaning the ability of extracting both grouped and individual affective states from still images has gained some popularity [29,30,17,31,33]. However, the recognition *throught time* of the affective state of individuals involved in group interactions has not yet been thoroughly addressed. We believe that one of the main reasons for this is the lack of robust and adaptive online person tracking methods. Indeed, a system able to perform group and individual behavior analysis in the wild has several prerequisites. First, the ability to tracking multiple persons, known as multiple object tracking [10,25,2,28,3]. Second, the ability to maintain an appearance model capable of recognising people seen in the past, known as person

re-identification [22,32,12]. Finally, the capacity to update both the tracking and the re-identification models incrementally, as new data is gathered by the system. This last step is particularly difficult, because it must be done at test time, and therefore is purely unsupervised.

In this paper, we are interested in endowing an autonomous system with the ability of tracking multiple persons in an online setting, and in particular to update its appearance description model to the persons in the scene. This is challenging because of four main reasons: (i) the method can only use causal information, since the system does not have access to future images and detections; (ii) the method must be computationally light, in the sense that the system must track people using consumer technology; (iii) the model update must be done online in an unsupervised manner, since the system does not have access to annotations of the tracked people; and (iv) the overall system must account for visual clutter e.g. occlusions and ego-motion.

To that aim, we propose a probabilistic model combined with a deep appearance model. While the probabilistic model sets the relationship between the latent variables (e.g. people's position) and the observations (bounding boxes), the appearance model based on a deep siamese neural network allows to robustly discriminate images belonging to different people. Most importantly, and this is the main contribution of the paper, the probabilistic-deep siamese combination allows to update the deep appearance model with the supervision generated by the probabilistic model, avoiding the necessity of annotated data. This combination is the key that allows the update of supervised discriminative models in unsupervised settings, such as the task at hand.

Up to our knowledge, we are the first to propose a method able to simultaneously track multiple people and update a deep appearance architecture, and light enough to be running in an autonomous robot. Indeed, we benchmarked our method with the state-of-the-art on standard datasets under two different settings: moving surveillance camera and robot navigation in crowded scenes. In addition, we provide qualitative results. The reported experiments validate our initial thoughts and confirm that updating the appearance model with the supervision from the probabilistic model is a good strategy. Indeed, the proposed method exhibits a significant performance increase when compared to the use of a fixed deep appearance model, and to the state-of-the-art. Our strategy appears to be effective for learning a discriminative appearance model on the fly, while tracking multiple people and accounting for clutter at the same time.

2 Related Work

Tracking by detection is the most popular paradigm in the MOT community. Causal tracking is generally performed by elaborating robust similarity measures between known tracks and current detections, and by using data association methods to obtain optimal track-to-detection assignments. The major differences

rely in the similarity measure used, which strongly depends on the visual cues that are used (e.g. spatio-temporal, appearance, interaction models).

Spatio-temporal similarity generaly assumes linear motion model [2,10,6]. The recent introduction of deep learning has leveraged the use of reliable appearance descriptors, allowing causal tracking models to robustly evaluate appearance similarities [23,3]. The introduction of person Re-ID models [24,28,7] has allowed tracking algorithms to take advantage of external dataset to improve their generalization ability. Several strategies are developped to aggregate those similarity measures: [2] takes advantage of the probabilistic formulation to merge different cue information, while some [28,11] propose a deep formulation where cues are merged early in the network which is directly trained to output the desired similarity measure. Single object tracking strategies [11,3,15] have also been exploited to perform multiple object tracking, especially [11,3] using the siamese formulation which allows an online finetuning of the appearance model to the tracking experiment at hand.

While the progress in MOT is quite significant, methods able to perform online MOT on autonomous robots are much more scarce. Computational complexity and moving cameras are two of the main difficultes most methods have trouble overcoming. One example of a MOT method fully adapted to robotic platforms is [4], where the authors propose a probabilistic model and a variational approximation to solve the tracking problem, while using the motor position to improve the tracking results. However, the appearance model used in [4] is based on color histogram descriptors, which lack or robustness and description power, specially in challenging visual conditions and for unseen identities.

We propose to exploit the same tracking formulation, to provide supervision for training a deep appearance models. Indeed, we exploit a Siamese deep network and use a soft-label formulation within the deep metric learning paradigm, to update the deep appearance model while tracking multiple people.

3 Joint Tracking and Appearance Modeling

In this section we introduce the propose online update strategy for our deep appearance model. To do so, we first briefly discuss the variational probabilistic tracker used in [4,5]. We refer the reader to these articles for a detailed description of the method. After, we discuss how metric learning can be used in the classical settings for person re-identification. This is then used to present the main methodological contribution of the manuscript, namely the use of the tracking results as a soft supervision to update the deep appearance model for re-identification.

3.1 Variational Multiple Object Tracking in a Nutshell

The probabilistic formulation used in this paper is inspired by [4,5], and lies within the tracking-by-detection paradigm, meaning that we assume the exis-

tence of a person detector (e.g. [1]) providing K_t bounding boxes at every frame t. These bounding boxes consist of the coordinates $\mathbf{y}_{tk}$ and of the content $\mathbf{h}_{tk}$. Very importantly, the detections are not associated to any of the N persons being tracked, and we face a double latent variable problem. Indeed, we must concurrently estimate the oposition of the persons as well as the bounding box-to-person assignment, in an unsupervised manner. To do so, the probabilistic formulation and the variational approximation proposed in [4,5] lead to an alternate optimisation algorithm that roughly speaking switches between running N Kalman filters and inferring the latent assignments. These assignments are computed using both the geometric similarity, with a Gaussian distribution on the bounding box coordinates, and the appearance similarity, with a Bhattacharyya distribution on the color histograms of the bounding box contents. The main limitations of color histograms is the low adaptability to new lighting conditions, to occlusions and to discriminate new appearances. This is mainly because color histograms are a fixed representation that cannot be trained/updated. Ideally, one would like to use new bounding boxes to update the appearance model.

3.2 Deep Probabilistic Appearance Model

In order to incorporate a learnable appearance model within the probabilistic formulation, we propose to use a feature extractor $\phi_{\boldsymbol{w}}(\mathbf{h})$ on the bounding box contents $\mathbf{h}$ and with trainable parameters $\boldsymbol{w}$. In practice, $\phi_{\boldsymbol{w}}$ will be instantiated by a convolutional neural network (CNN), that will be merged with the probabilistic formulation by assuming the following distribution on the extracted features:

$$\phi_{\boldsymbol{w}}(\mathbf{h}) \sim \mathcal{N}(\phi_{\boldsymbol{w}}(\mathbf{h}); \boldsymbol{m}_n, s_n^2 \mathbf{I}_A), \tag{1}$$

if $\mathbf{h}$ is associated to the n-th source and being $\boldsymbol{m}_n \in \mathbb{R}^A$ and $s_n > 0$ the mean and variance of the deep probabilistic model for source n, where A is the dimension of the extracted features. While this is a very natural way of modeling the appearance within a probabilistic framework for tracking, up to our knowledge we are the first to holistically include a deep appearance model within a probabilistic framework, and to provide an online update algorithm for training the deep appearance model in unsupervised settings.

The parameters of the probabilistic model, $\boldsymbol{m}_n$ and s_n, together with the network parameters, $\boldsymbol{w}$ need to be estimated. To do so, we propose an alternate learning procedure based on the variational expectation maximisation proposed in [4,5]. Indeed, the same variational approximation can also be applied here and leads to a VEM algorithm with three steps: the E-position step, the E-assignment step, and the M step. As for any VEM, these three steps are alternated, and therefore one needs to first initialise two of these steps, and start the alternating procedure with the third step. In our case, we will initialise the VEM algorithm at time t, by using the same parameters (M-step) as in the previous time frame, and uniformly assign the detections (E-assignment) to the N persons. We can therefore start the alternate optimisation procedure with the E-position step,

which provides the mean vector $\boldsymbol{\mu}_{tn}$ and the covariance matrix $\boldsymbol{\Gamma}_{tn}$ of the n-th source at time frame t. In the present paper, we use the exact same formulae of [4,5], leading to N separate Kalman filters, weighted by the soft-assignments.

Once the posterior distribution of the position is computed, and $\boldsymbol{\mu}_{tn}$ and $\boldsymbol{\Gamma}_{nt}$ are obtained, we update the posterior distribution of the assignment variables:

$$\alpha_{tkn} = \frac{\eta_{tkn}}{\sum_m \eta_{tkm}}, \tag{2}$$

where α_{tkn} denotes the posterior probability of the observation $(\mathbf{y}_{tk}, \mathbf{h}_{tk})$ to be generated from the n-th person, and:

$$\eta_{tkn} = \underbrace{\mathcal{N}(\mathbf{y}_{tk}, \boldsymbol{\mu}_{tn} - \mathbf{E}_t, \boldsymbol{\Sigma}) \mathrm{e}^{-\frac{1}{2}\mathrm{tr}(\boldsymbol{\Sigma}^{-1}\boldsymbol{\Gamma}_{tn})}}_{\text{Coordinates}} \underbrace{\mathcal{N}(\phi_{\boldsymbol{w}}(\mathbf{h}_{tk}), \boldsymbol{m}_n, s_n^2 \mathbf{I}_A)}_{\text{Appearance}}, \tag{3}$$

where $\mathbf{E}_t$ is the impact of the ego-motion of the system in the image plane (see Section 4.3 for a detailed description on how tom compute $\mathbf{E}_t$ from e.g. optical flow). The previous equation shows how the posterior distribution of an assignment variable is updated taking into account both the coordinate similarity (left) and the appearance similarity (right). It is therefore of utmost importance to update the model online so as to obtain a discriminative feature extractor $\phi_{\boldsymbol{w}}$ and the associated mean $\boldsymbol{m}_n$ and variance s_n for each of the tracked sources. This corresponds to the M-step, presented in the next Section.

3.3 Unsupervised Deep Metric Learning

The update of the parameters of the deep probabilistic appearance model, that is $\boldsymbol{w}$, $\boldsymbol{m}_n$ and s_n, is done within the framework of unsupervised deep metric learning. In order to motivate this choice, we first discuss standard supervised deep metric learning, and then integrate this within our probabilistic formulation.

Supervised metric learning consists in learning a distance such that similar elements are close, and dissimilar ones are far apart. Siamese networks became a common framework for metric learning, also in the tracking community [21,9,23]. Differently from classification problems, training Siamese networks requires a data set of triplets $(\mathbf{h}_i, \mathbf{h}_j, c_{ij})$, where $i, j \in \{1, \ldots, I\}$. The two bounding box images are feed-forwarded with the same weights $\boldsymbol{w}$, thus obtaining $\phi_{\boldsymbol{w}}(\mathbf{h}_i)$ and $\phi_{\boldsymbol{w}}(\mathbf{h}_j)$ respectively. The label is $c_{ij} = 1$ if the two images $\mathbf{h}_i$, $\mathbf{h}_j$ belong to the same person, and $c_{ij} = -1$ otherwise. A popular loss for training Siamese networks is a variant of the contrastive loss [18], introduced in [20]:

$$J(\boldsymbol{w}) = \frac{1}{2} \sum_{i,j=1}^{I} g(c_{ij}(\tau - \|\phi_{\boldsymbol{w}}(\mathbf{h}_i) - \phi_{\boldsymbol{w}}(\mathbf{h}_j)\|^2)), \tag{4}$$

with $g(x) = \max(0, 1 - x)$ and $\tau > 0$ is a parameter.

Traditionally, (4) is optimized with stochastic gradient descent, thus forcing squared distances of elements from a negative pair further than $\tau+1$, and those of a positive pair closer than $\tau-1$. At test time, one can use the distance between the embedding vectors to gauge whether two appearances belong to the same person or not.

The standard (supervised) metric learning framework has proven to be useful for a variety of tasks, but requires an annotated dataset. In our case, that is at test time of a multiple object tracker, the bounding box-to-person assignments are not annotated. In this sense, we face an unsupervised deep metric learning task. As previously discussed, the E-assignment step of the VEM algorithm, see (2), provides an approximation of the posterior distribution of these assignment variables. In the following we introduce a new methodology that allows to exploit the posterior distribution of the assignment variables to update the parameters $\boldsymbol{w}$ of the feature extractor $\phi_{\boldsymbol{w}}$.

For any given two past observations k_i and k_j, from time frames t_i and t_j, we can compute the probability to be both generated by the same person:

$$\gamma_{ij} = \sum_{n=1}^{N} \alpha_{t_i k_i n} \alpha_{t_j k_j n}. \tag{5}$$

Intuively, if γ_{ij} is close to 1 or to 0, it means that the pair $(\mathbf{h}_i, \mathbf{h}_j)$ is a true positive (the appearance images belong to the same person) or a true negative (the appearance images belong to different persons) with high confidence. If γ_{ij} is not close to 1 or 0, then the image pair assignment is uncertain.

Once the γ_{ij} are computed, we sample pseudo-positive pairs from those with $\gamma_{ij} > \eta$ and pseudo-negative pairs from those with $\gamma_{ij} < 1-\eta$, respectively $\mathcal{H}^+$ and $\mathcal{H}^-$ sets. We optimize the following soft-weighted contrastive loss:

$$\begin{aligned} J(\boldsymbol{w}) = &\frac{1}{2} \sum_{ij \in \mathcal{H}^+} g(\gamma_{ij}(\tau - \|\phi_{\boldsymbol{w}}(\mathbf{h}_i) - \phi_{\boldsymbol{w}}(\mathbf{h}_j)\|^2)) \\ &- \frac{1}{2} \sum_{ij \in \mathcal{H}^-} g((1-\gamma_{ij})(\tau - \|\phi_{\boldsymbol{w}}(\mathbf{h}_i) - \phi_{\boldsymbol{w}}(\mathbf{h}_j)\|^2)). \end{aligned} \tag{6}$$

The weights of the feature extractor $\phi_{\boldsymbol{w}}$ will therefore be updated using the soft-weighted contrastive loss detailed above. This is important for two main reasons. First, the soft-weighted contrastive loss does not require any annotation, since the weights of the loss are provided by the unsupervised VEM algorithm. Second, those weights provide implicitly information about the confidence of the image pair when training the network. Indeed, the weights γ_{ij} are used not only to sample image pairs, but also to weight the loss of that image pair. In this way, we force the network to exploit more intensively the pairs with low uncertainty (γ_{ij} close to 1 or to 0).

Simultaneously to the training of the Siamese network, one needs to update the appearance parameters of the probabilistic model, i.e. $\boldsymbol{m}_n$ and s_n^2, as defined

in (3). As is the case for the parameters of the Siamese network, there is no annotated dataset, and this update must be done in an unsupervised manner. However, we still have access to the posterior probability of the bounding box-to-person assignments, and easily obtain the following updates:

$$\boldsymbol{m}_n = \frac{1}{S_n} \sum_{t'=t-u}^{t} \sum_{k=1}^{K_{t'}} \alpha_{knt'} \phi_{\boldsymbol{w}}(\mathbf{h}_{t'k}) \tag{7}$$

$$s_n^2 = \frac{1}{S_n A} \sum_{t'=t-u}^{t} \sum_{k=1}^{K_{t'}} \alpha_{knt'} \|\phi_{\boldsymbol{w}}(\mathbf{h}_{t'k}) - \boldsymbol{m}_n\|^2 \tag{8}$$

where $S_n = \sum_{t'=t-u}^{t} \sum_{k=1}^{K_{t'}} \alpha_{knt'}$ and u is a windowing parameter. Updating these parameters at each time step allows the appearance model to be more flexible to sudden appearance variations and to better adapt the internal track appearances to the observations. The optimisation of (6) together with (7)-(8) form the M-step of the VEM algorithm for joint tracking and update of the deep appearance model under unsupervised settings.

4 Overall Tracking System

In this section we describe the implementation of the method for joint tracking and appearance update, including the details on how to update the deep appearance model, and on how to initialise new tracks. Additionally, we provide training details and a software architecture for a generic robotic platform.

4.1 Deep Appearance Model Update

We instantiate our appearance model with a generic CNN backbone, consisting on several convolutional layers (see details below). This backbone is trained by means of mini-batch stochastic gradient descent techniques. To construct batches, we randomly sample a track, one positive pair (from $\mathcal{H}^+$) and two negative pairs (from $\mathcal{H}^-$) from this track. This is done so as to respect the positive-negative balance [3,11]. This strategy is repeated B times, obtaining a batch annotated with γ_{ij}.

$\phi_{\boldsymbol{w}}$ is pretrained to perform an ID classification similarly to [24]. Thus, the appearance model update can now be seen as a domain adaptation problem, where we need to learn the appearance shift (different people, background and illumination changes) between the pre-training dataset and the tracking data. To achieve this adaptation, only the top layers of $\phi_{\boldsymbol{w}}$ are updated during tracking. The amount of layers to be trained depends on the computation power of the system, allowing the best trade-off between generalization ability and computation complexity. In our case, we perform an update of the last 2 layers.

4.2 Birth and Visibility Processes

New tracks (e.g. people coming in the field of view) are initialized using a birth process that assesses the consistency of the observations previously assigned to none of the visual tracks. We compare two hypothesis: (i) the previous L geometric observations $\mathbf{y}_{tk_0}, \ldots, \mathbf{y}_{t-Lk_L}$ assigned to clutter correspond to an undetected track, and (ii) the very same observations belong to clutter and are uniformly distributed. If the first hypothesis wins, a new track is initialized using the last detected bounding box, and its appearance model is initialized with the content of the bounding boxes using (7) and (8).

We used an additional hidden Markov model visibility process to determine whether or not a track has been lost. The observation of this binary visibility process arise from the output of the variational EM algorithm, in particular we set: $\nu_{tn} = \sum_k \alpha_{tkn}$, representing whether a given track n is assigned to a current detection or not. The estimation of the latent variable probability $p(V_{tn}|\nu_{1:t})$ is done using standard HMM inference algorithms, and informs us about the visibility state of the considered track. Non-visible tracks are not output by the tracker, and their associated variables and parameters are not updated until they are visible again.

4.3 Implementation and Training Details

The overall tracking algorithm is presented in Algorithm 1. While tracking, the algorithm uses the various updates derived from the variational EM algorithm (see Section 3.2). Every T frames, the system updates the appearance model with the equation updates of Section 3.3 and the sampling strategies and implementation details of Section 4.1. The birth and visibility processes of Section 4.2 are then used to set up new tracks and freeze non-visible tracks.

The appearance model is updated every $T = 5$ frames using a training set created by sampling 50 images per identity, and then sampling image pairs as described in Section 4.1. We update the appearance models using stochastic gradient descent with RMS [16] with a learning rate of $\lambda = 0.001$ and a weight decay with a factor $\lambda = 10^{-4}$, for two epochs.The appearance model CNN is instantiated by a ResNet [19] architecture, where the last layer was replaced by a two layer perceptron with 500 and 100 units activated with ReLu. This CNN is pre-trained for the person re-identification task on the Market-1501 [35] and DukeMTMC [36] datasets, following [37].

Optical flow (OF) is extracted [13] and used to estimate the ego-motion vector $\mathbf{E}_t$, by averaging the OF over the entire image. In addition, the average optical flow within one detection bounding box, provides an estimate of the velocity of the track (after the ego-motion vector is subtracted). In order to get stable results we regularise s_n^2 with a prior around $\tau - 1$, since the cost (6) is supposed to induce spherical clusters of radius $\sqrt{\tau - 1}$.

while *tracking* **do**
  $t \leftarrow t + 1$;
  Compute the ego-motion vector $\mathbf{E}_t$, see Section 4.3;
  Update $\{\alpha_{tkn}\}_{k\in[1,K_t],n\in[1,N]}$ with (2);
  Update $\{\boldsymbol{\mu}_{tn}, \boldsymbol{\Gamma}_{tn}\}_{n\in[1,N]}$ as in Section 3;
  Update $\{\boldsymbol{m}_n, s_n\}_{n\in[1,N]}$ with (7) and (8;
  if $t_u = T$ **then**
    $t_u \leftarrow 0$ // Reset frame counter;
    while $\phi_{\boldsymbol{w}}$ *not converged* **do**
      $\gamma_{ij} \leftarrow$ compute with (5);
      $\mathcal{H}^+, \mathcal{H}^- \leftarrow$ sample pair sets, see Section 4.1;
      Optimise (6) with RMS;
    end
  end
  $t_u \leftarrow t_u + 1$;
  Birth/visibility update, see Section 4.2;
  Output $\{\boldsymbol{\mu}_{tn}\}_{n\in[1,N]}$;
end

Algorithm 1: Overall tracking algorithm. Updates are performed with the various equations and strategies already described. The frame update counter t_u allows to update the $\phi_{\boldsymbol{w}}$ every T frames. The algorithm outputs the position of all tracks at every frame t.

In addition to the offline quantitative experiments, we also run online quantitative experiments (no ground truth is available) on a robotic platform. To do so, the proposed algorithm is implemented on a robotic platform using the ROS middleware. ROS does not only allow a platform-independent implementation, but also provides a unified framework to distribute the computation when and where needed. We use this property to exploit the computational power of an external GPU, devoted to execute the face detector [1] and to extract CNN appearance features. The face detector replace here the person detectors used in standard tracking datasets that will be used in our offline quantitative experiments. This demonstrates the ability of our tracking and appearance update method to utilise different kind of detectors. All other computations, including the update of the Siamese network, are ran on the CPU of the robot. We use a Intel(R) Xeon(R) CPU E5-2609 and a NVIDIA GeForce GTX 1070, and exploit the native camera of the robot. The system runs under Ubuntu 16.04 and ROS Kinetic version. Thanks to these implementation choices, our online tracker runs at 10 FPS. A schematic representation of the overall tracking system is shown in Figure 1.

ROS makes use of a distributed network of Nodes (ROS processes), which use topics to communicate. Typically, the drivers nodes control the low level communication with robot's sensors and produce Topics containing images and motor positions. Camera's topics are processed by the face detection and feature extraction Nodes to produce detections and deep appearance descriptors transmitted to the tracker Node. The tracker Node takes advantage of the mo-

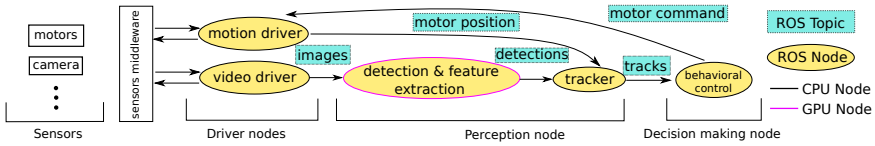


Fig. 1: The robotic software architecture is composed of several nodes: an image is produced by the video driver, fed to the face detector which produces both face detections and appearance features, then transmitted to the tracker node (alongside motor position information). The tracking results are exploited by the robot control to move the robot's head exploiting the motors drivers.

tor information and detections to update the appearance model and produce track information that is transmitted to the behavioral control, which uses it to update the motors' position, depending on the predefined policy.

5 Experiments

We evaluate our joint tracking and appearance model update robotic architecture in three different settings. First, we provide quantitative evaluation under the "active surveillance camera" scenario. Second, we provide quantitative evaluation under the "robot navigation in crowded scenes" scenario. Finally, we provide qualitative evaluation under the "social conversation" scenario. Our aim is to evaluate whether or not the proposed deep appearance model online update provides more consistent tracks than the state-of-the-art.

5.1 Quantitative Evaluation

Dataset For the sake of reproducibility and in order to be able to compare different tracking systems in the exact same conditions, we use the well-known MOT17 dataset [25]. This dataset is composed of a dozen of videos taken in various conditions, and provides detections obtained with various detectors (DPM [14], FR-CNN [26] and SDP [34]) as well as ground truth for evaluation. The standard protocol is to report the results averaged over the three detectors.

Importantly, two kinds of videos are available: recorded with a surveillance camera, and with a camera mounted on an autonomous robot navigating in crowded scenes. Both scenarios are of interest for us. Indeed, the first scenario allows us to simulate the motion of a surveillance camera, and to gauge the robustness of the proposed tracking system against ego-motion noise. The second scenario provides the opposite case in which the ego-motion is completely unknown and must be inferred from visual information.

The first scenario (*moving surveillance camera*) consists on emulating that the surveillance camera only sees half of the image width, and then moving the

Table 1: Results on the *moving surveillance camera* setting .

η	Model	Detection		Tracking		Identities		
		Rcll	Prcn	MOTA	MOTP	IDP	IDR	IDF1
0	CH [5]	49.4	88.2	42.5	84.5	70.3	39.4	50.5
	ODA-FR	49.5	**88.7**	43.0	**84.8**	66.7	37.2	47.8
	ODA-UP	**54.7**	86.7	**45.6**	84.0	**75.4**	**45.7**	**56.0**
0.8	CH [5]	49.6	88.0	42.5	84.4	69.9	39.4	50.4
	ODA-FR	49.7	**88.7**	43.1	**84.7**	67.1	37.6	48.2
	ODA-UP	**54.4**	86.3	**45.0**	83.8	**71.2**	**44.9**	**55.1**
1.6	CH [5]	49.1	88.2	42.2	84.2	70.3	39.1	50.2
	ODA-FR	49.5	**88.6**	42.8	**84.5**	66.3	37	47.5
	ODA-UP	**54.5**	86.4	**45.3**	83.7	**73.3**	**46.2**	**56.7**
3.2	CH [5]	49.2	88.2	42.3	83.2	68.1	38.0	48.8
	ODA-FR	49.1	**88.4**	42.4	**83.3**	66.8	37.1	47.7
	ODA-UP	**54.2**	86.1	**44.8**	82.8	**71.5**	**45.0**	**55.2**

emulated field of view accordintly to a pre-defined trajectory. The ego-motion vector $\mathbf{E}_t$ is then contaminated with Gaussian noise with a standard deviation of η pixels in a uniformly sampled direction. The second scenario (*robot navigating in the crowd*) consists on using the full field of view of the camera, and estimate the ego-motion vector $\mathbf{E}_t$ with an optical-flow-based strategy.

Evaluation Protocol We compare our *online deep appearance update* (ODA-UP) based method with the state-of-the-art in multi-person tracking for social robotics [5]. While the tracking model is very similary, the appearance model previously used in the literature is based in *color histograms* (CH). Additionally, and in order to provide a full evaluation of the necessity of the on-line appearance model update, we compare the proposed tracker with the extact same architecture without updating the weights of the deep appearance model, and refer to it as ODA-FR, for frozen. In that case, the appearance likelihood is provided by computing the cosine similarity between appearance templates and current detections. For the *moving surveillance camera* scenario, we evaluate under different values of $\eta \in \{0, 0.8, 1.6, 3.2\}$. We report standard multiple object tracking metrics in three groups. The quality of the detections is evaluated with the recall (Rcll) and the precision (Prcn). The quality of the tracks is evaluated with the multiple object tracking accuracy (MOTA) and precision (MOTP), see [8]. The consistency of the track identities is evaluated with the identity recall (IDR), precision (IDP) and F1 measure (IDF1), see [27].

Discussion Table 1 and 2 report the results in the two scenarii. Regarding the *moving surveillance camera* setting in Table 1, we first observe that our approach significantly outperforms both the frozen (FR) and the color histogram

Table 2: Results on the *robot navigating in the crowd* settings.

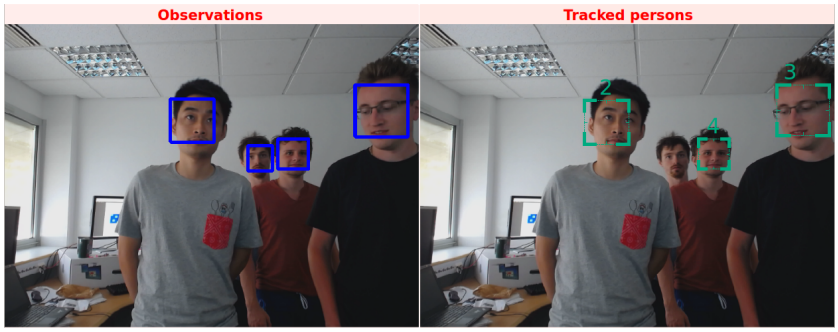
Model	Detection		Tracking		Identities		
	Rcll	Prcn	MOTA	MOTP	IDP	IDR	IDF1
CH [5]	45.8	91.8	41.2	80.7	74.1	37.0	49.3
ODA-FR	45.8	**93.1**	42.0	81.0	73.8	36.3	48.6
ODA-UP	**52.3**	90.5	**46.2**	**81.5**	**79.0**	**45.7**	**57.9**

(CH) models, by more than +3% and +2% respectively in MOTA. Unsurprisingly, the pretrained appearance model outperforms color histogram based model by roughly +0.5% in MOTA. While different levels of ego-motion noise lead to different scores, the ranking between the methods stays the same. The difference in MOTP is quite small, meaning that the quality of the output bounding boxes (only the tracked ones) is roughly the same. The slight decrease for ODA-UP is due to the fact that ODA-UP is able to track people that are harder to track, and for which estimating good bounding boxes is more challenging. This is supported by the relative position of the methods in the other metrics. Indeed, the recall and precision metrics are another proof that ODA-UP is able to track significantly more people. Regarding the identify measures, we can see that ODA-UP exhibits by far the highest performance, putting forward the advantage of the adaptive strategy. Indeed, the ODA-UP model outperforms the other two. Interestingly, ODA-FR is outperformed (in identity measures) by the color histograms, demonstrating that compelx deep models are useful only if trained in relevant data or, as we propose in this paper, if they are adapted online.

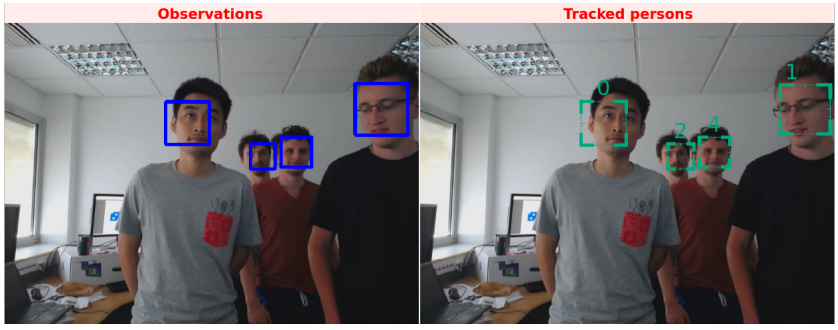
A similar situation is found in Table 2 for *robot navigating in the crowd* setting: our approach outperforms the histogram model and the pretrained model by respectively +5.0% and +4.4% in MOTA. The rest of the metrics follow the same ranking as in the previous setting. Very importantly, the findings in Table 1 are further confirmed by larger advantage margins in Table 2. In both experiments, we observe how that update of the deep appearance model brings two main advantages. First, the recall increases because the number of false negative (or missed detections) decreases. Second, and more important, the consistency of the tracks' ID exhibits a significant increase when updating the model online.

5.2 Qualitative Results

We qualitatively evaluate the performance of the tracker on a real robotic platform, as described in Section 4.3, and provide them as videos alongside the paper. More results are available at https://team.inria.fr/perception/research/oda_track/. In that case, the procedure in [4] is used to compute the ego-motion vector from the motors velocity. Only faces' bounding boxes are extracted from the images using [1]. Example of the tracking results are displayed in Figure 2, using CH and ODA-UP. We note that since our deep metric formulation has a



(a) Tracking result using CH [4]



(b) Tracking result using ODA-UP

Fig. 2: Tracking qualitative results using CH and ODA-UP. Detections are displayed on the left panel (blue), and tracking results are available on the right panel (green) in 2 settings.

higher discriminative power than color histogram based appearance model, it is able to better distinguish ID 2 and ID 4, even if they are close and that both detections could be generated by a unique ID. Also, we note that the ID labels of the tracks differ significantly when comparing both methods, which is caused by the high number of identity switches in the CH setting.

6 Conclusions

In this paper, we adressed the problem of online multiple object tracking using a joint probabilistic and deep appearance model, that allow the update of the appearance embedding simultaneously to tracking multiple people, while accounting for the robot ego-motion. We demonstrate its performance quantitatively in two settings, and qualitatively onboard of a consumer robot. The proposed model exhiits superior tracking performance ot the state of the art Very importantly, the identify consitency of the tracks over time is significantly better thanks to the proposed online update strategy. We hope that the proposed system will open the door to the affective analysis of multi-person scenarios, and will foster more research efforts on leveraging online information to conceive better appearance models.

References

1. High quality face recognition with deep metric learning, http://blog.dlib.net/2017/02/high-quality-face-recognition-with-deep.html
2. Ba, S., Alameda-Pineda, X., Xompero, A., Horaud, R.: An on-line variational bayesian model for multi-person tracking from cluttered scenes. Computer Vision and Image Understanding **153**, 64–76 (2016)
3. Bae, S., Yoon, K.: Confidence-based data association and discriminative deep appearance learning for robust online multi-object tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence (2018)
4. Ban, Y., Alameda-Pineda, X., Badeig, F., Ba, S., Horaud, R.: Tracking a varying number of people with a visually-controlled robotic head. In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 4144–4151. IEEE (2017)
5. Ban, Y., Alameda-Pineda, X., Girin, L., Horaud, R.: Variational bayesian inference for audio-visual tracking of multiple speakers. IEEE Transactions on Pattern Analysis and Machine Intelligence (2019)
6. Ban, Y., Ba, S., Alameda-Pineda, X., Horaud, R.: Tracking Multiple Persons Based on a Variational Bayesian Model. In: Computer Vision – ECCV 2016 Workshops
7. Bergmann, P., Meinhardt, T., Leal-Taixe, L.: Tracking without bells and whistles. In: IEEE International Conference on Computer Vision. pp. 941–951 (2019)
8. Bernardin, K., Stiefelhagen, R.: Evaluating multiple object tracking performance: The clear mot metrics. EURASIP Journal on Image and Video Processing (2008)
9. Bertinetto, L., Valmadre, J., Henriques, J.F., Vedaldi, A., Torr, P.H.: Fully-convolutional siamese networks for object tracking. In: European conference on computer vision. pp. 850–865. Springer (2016)
10. Breitenstein, M.D., Reichlin, F., Leibe, B., Koller-Meier, E., Gool, L.V.: Robust tracking-by-detection using a detector confidence particle filter. In: 2009 IEEE ICCV (Sep 2009)
11. Chu, Q., Ouyang, W., Li, H., Wang, X., Liu, B., Yu, N.: Online multi-object tracking using cnn-based single object tracker with spatial-temporal attention mechanism. In: IEEE International Conference on Computer Vision, ICCV (10 2017)
12. Delorme, G., Xu, Y., Lathuilière, S., Horaud, R., Alameda-Pineda, X.: Canu-reid: A conditional adversarial network for unsupervised person re-identification. In: International Conference on Pattern Recognition (2020)
13. Farnebäck, G.: Two-frame motion estimation based on polynomial expansion. In: Bigun, J., Gustavsson, T. (eds.) Image Analysis. pp. 363–370. Springer Berlin Heidelberg, Berlin, Heidelberg (2003)
14. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. IEEE TPAMI **32**(9) (2010)
15. Feng, W., Hu, Z., Wu, W., Yan, J., Ouyang, W.: Multi-object tracking with multiple cues and switcher-aware classification. arXiv (2019)
16. Geoffrey Hinton, Nitish Srivastava, K.S.: Lecture 6a: Overview of mini-batch gradient descent, http://www.cs.toronto.edu/tijmen/csc321/slides/lecture_slides_lec6.pdf
17. Gupta, A., Agrawal, D., Chauhan, H., Dolz, J., Pedersoli, M.: An attention model for group-level emotion recognition. In: Proceedings of the 20th ACM International Conference on Multimodal Interaction. pp. 611–615 (2018)
18. Hadsell, R., Chopra, S., LeCun, Y.: Dimensionality reduction by learning an invariant mapping. In: null. pp. 1735–1742. IEEE (2006)

19. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. CoRR **abs/1512.03385** (2015)
20. Hu, J., Lu, J., Tan, Y.P.: Discriminative deep metric learning for face verification in the wild. In: IEEE Conf. on Computer Vision and Pattern Recognition (2014)
21. Hu, J., Lu, J., Tan, Y.P.: Deep metric learning for visual tracking. IEEE Transactions on Circuits and Systems for Video Technology **26**(11), 2056–2068 (2016)
22. Kalayeh, M.M., Basaran, E., Gökmen, M., Kamasak, M.E., Shah, M.: Human semantic parsing for person re-identification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1062–1071 (2018)
23. Leal-Taixe, L., Canton-Ferrer, C., Schindler, K.: Learning by tracking: Siamese cnn for robust target association. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops (June 2016)
24. Long, C., Haizhou, A., Zijie, Z., Chong, S.: Real-time multiple people tracking with deeply learned candidate selection and person re-identification. In: ICME (2018)
25. Milan, A., Leal-Taixé, L., Reid, I., Roth, S., Schindler, K.: MOT16: A benchmark for multi-object tracking. arXiv:1603.00831 (Mar 2016)
26. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence
27. Ristani, E., Solera, F., Zou, R., Cucchiara, R., Tomasi, C.: Performance measures and a data set for multi-target, multi-camera tracking. In: European Conference on Computer Vision. pp. 17–35. Springer (2016)
28. Sadeghian, A., Alahi, A., Savarese, S.: Tracking the untrackable: Learning to track multiple cues with long-term dependencies. In: IEEE International Conference on Computer Vision, ICCV (2017)
29. Sun, B., Wei, Q., Li, L., Xu, Q., He, J., Yu, L.: Lstm for dynamic emotion and group emotion recognition in the wild. In: Proceedings of the 18th ACM International Conference on Multimodal Interaction. pp. 451–457 (2016)
30. Tan, L., Zhang, K., Wang, K., Zeng, X., Peng, X., Qiao, Y.: Group emotion recognition with individual facial emotion cnns and global image based cnns. In: Proceedings of the 19th ACM International Conference on Multimodal Interaction. pp. 549–552 (2017)
31. Wang, K., Zeng, X., Yang, J., Meng, D., Zhang, K., Peng, X., Qiao, Y.: Cascade attention networks for group emotion recognition with face, body and image cues. In: Proceedings of the 20th ACM International Conference on Multimodal Interaction. pp. 640–645 (2018)
32. Wei, L., Zhang, S., Gao, W., Tian, Q.: Person transfer gan to bridge domain gap for person re-identification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 79–88 (2018)
33. Wei, Q., Zhao, Y., Xu, Q., Li, L., He, J., Yu, L., Sun, B.: A new deep-learning framework for group emotion recognition. In: Proceedings of the 19th ACM International Conference on Multimodal Interaction. pp. 587–592 (2017)
34. Yang, F., Choi, W., Lin, Y.: Exploit all the layers: Fast and accurate cnn object detector with scale dependent pooling and cascaded rejection classifiers. In: CVPR (2016)
35. Zheng, L., Shen, L., Tian, L., Wang, S., Wang, J., Tian, Q.: Scalable person re-identification: A benchmark. In: IEEE ICCV (2015)
36. Zheng, Z., Zheng, L., Yang, Y.: Unlabeled samples generated by gan improve the person re-identification baseline in vitro. In: IEEE ICCV (2017)
37. Zhong, Z., Zheng, L., Zheng, Z., Li, S., Yang, Y.: Camera style adaptation for person re-identification. In: IEEE CVPR (2018)