# MX-LSTM: mixing tracklets and vislets to jointly forecast trajectories and head poses

Irtiza Hasan[1,2], Francesco Setti[1], Theodore Tsesmelis[1,2,3], Alessio Del Bue[3], Fabio Galasso[2], and Marco Cristani[1]

[1] University of Verona (UNIVR) [2] OSRAM GmbH [3] Istituto Italiano di Tecnologia (IIT)

{irtiza.hasan,francesco.setti,marco.cristani}@univr.it,

{t.tsesmelis,f.galasso}@osram.com, alessio.delbue@iit.it

## Abstract

*Recent approaches on trajectory forecasting use tracklets to predict the future positions of pedestrians exploiting Long Short Term Memory (LSTM) architectures. This paper shows that adding* vislets*, that is, short sequences of head pose estimations, allows to increase significantly the trajectory forecasting performance. We then propose to use vislets in a novel framework called MX-LSTM, capturing the interplay between tracklets and vislets thanks to a joint unconstrained optimization of full covariance matrices during the LSTM backpropagation. At the same time, MX-LSTM predicts the future head poses, increasing the standard capabilities of the long-term trajectory forecasting approaches. With standard head pose estimators and an attentional-based social pooling, MX-LSTM scores the new trajectory forecasting state-of-the-art in all the considered datasets (Zara01, Zara02, UCY, and TownCentre) with a dramatic margin when the pedestrians slow down, a case where most of the forecasting approaches struggle to provide an accurate solution.*

## 1. Introduction

Anticipating the trajectories that could occur in the future is important for several reasons: in computer vision, path forecasting helps the dynamics modeling for target tracking [40, 47, 48, 59] and behavior understanding [3, 30, 33, 35, 47]; in robotics, autonomous systems should plan routes that will avoid collisions and be respectful of the human proxemics [13, 21, 31, 36, 53, 62]. Recently, path forecasting has benefited from the introduction of Long Short Term Memory (LSTM) architectures [3, 22, 26, 50, 51, 55].

All of these approaches use exclusively the $(x, y)$ position coordinates for the prediction, forgetting that humans act and react using their senses to explore the environment, in particular, through the visual information conveyed by the gaze and inferred by the head pose [8, 9, 11, 14, 15, 16, 27, 39, 46, 49, 54]. In particular, [8, 11, 14, 15, 16, 27, 39, 54] found that the head pose correlates to the person destination and pathway: these findings are also supported by a statistical analysis presented in our paper (Sec. 4).

For the first time this work considers the head pose, jointly with the positional information, as a cue to perform forecasting. In particular, tracklets (sequences of $(x, y)$ coordinates) and *vislets*, that is, reference points indicating the head pan orientation, are the input of the novel MiXing LSTM (MX-LSTM), an LSTM-based model that learns how tracklet and vislet streams are related, mixing them together in the LSTM hidden state recursion by means of cross-stream full covariance matrices, optimized during backpropagation.

MX-LSTM is able to encode how movements of the head and the people dynamics are connected. For example, it captures the fact that rotating the head towards a particular direction may anticipate a trajectory drifting with an acceleration (as in the case of a person leaving a group after a conversation). This happens thanks to a novel optimization of the LSTM parameters using a Gaussian full covariance through an unconstrained log-Cholesky parameterization in the backpropagation, securing positive semidefinite matrices. To the best of our knowledge, this is the first time Gaussian distributions with covariance matrices of order higher than two are optimized in LSTMs.

Vislet information is also used to build a scene context, i.e. where are the other people and how they are moving, by a shared state pooling as in [3, 55], that here is further improved using the head pose by discarding the people that an individual cannot see.

As a by-product, MX-LSTM predicts head orientations too, allowing to reason where people will most probably look at, providing a fine grained level of long-term prediction never reached so far in crowded scenarios.

Adopting standard protocols for trajectory forecasting [3, 34, 40] and using head poses information given by a standard head pose estimator [32], MX-LSTM defines the new state-of-the-art both in the UCY sequences (Zara01, Zara02 and UCY) and in the TownCentre dataset. In particular, MX-LSTM has the ability to forecast people when they are moving slowly, the Achille's heel of all the other approaches proposed so far.

As main contributions, in this paper:

- We show that trajectory forecasting can be dramatically ameliorated by considering head pose estimates;
- We propose a novel LSTM architecture, MX-LSTM, which exploits positional (tracklets) and orientational (vislets) information thanks to an optimization of $d$-variate Gaussian parameters including full covariances with $d > 2$;
- We motivate the need for MX-LSTM showing that head poses are related with the trajectories, even at low velocities, where most of the forecasting approaches fail;
- We define a novel type of social pooling, in the sense of [3, 55], by exploiting the vislet information;
- Thanks to MX-LSTM, we define state-of-the-art forecasting results on different datasets;
- We present MX-LSTM results of head pose forecasting, showing new long-term behavior analysis capabilities.

The rest of the paper is organized as follows. Sec. 2 reviews the related literature. Sec. 3 presents the proposed MX-LSTM while Sec. 4 motivates its design by showing how head pose and trajectories are related in the most popular forecasting datasets. We show quantitative and qualitative experiments in Sec. 5, concluding the paper in Sec. 6.

## 2. Related work

Classical forecasting approaches [38] adopted Kalman filters [29], linear [37] or Gaussian regression models [44, 45, 57, 58], autoregressive models [2] and time-series analysis [43]. These approaches ignore human-human interactions, which instead play a major role in recent literature.

**Human-human interactions.** The consideration of other pedestrians in the scene and their innate avoidance of collisions was first pioneered by [24]. This initial seed was further developed by [34], [35] and [40], which respectively introduced a data-driven, a continuous, and a game theoretical model. Notably, these approaches successfully employ essential cues for track prediction such as the human-human interaction and the people intended destination. More recent works encode the human-human interactions into a "social" descriptor [4] or propose human attributes [60] for the forecasting in crowds. More implicitly, other methods [3, 55] embed proxemic reasoning in the prediction by pooling hidden variables representing the probable location of a pedestrian in a LSTM. Our work mainly differentiates from [3, 34, 40, 55] because we only consider for interactions those people who are within the cone of attention of the person, (as also verified by psychological studies [27]).

**Destination-focused path forecast.** Path forecasting has also been framed as an inverse optimal control (IOC) problem [30]. Follow-up work adopted inverse reinforcement learning [1, 61] and dynamic reward functions [33] to address the occurring changes in the environment. We describe these approaches as destination-focused because they all require the end-point of the person track to be known, which later works have relaxed to a set of plausible path ends [13, 36]. Here we discard this information, that in our opinion undermines the reason why we may be predicting the tracks.

**The head pose and the social motivation.** The interest into the head pose stems from sociological studies such as [8, 11, 14, 15, 16, 39, 54], whereby head pose has been shown to correlate to the person destination and pathway. In this paper, we also discover that the head pose is correlated with the movement, especially at high velocities, while slowing down this correlation decreases too, but still remaining statistically significant. These studies motivate the use of the head pose as a proxy to the track forecasting. Using head pose comes with the further advantage that it can be estimated at small resolutions [5, 17, 23, 32, 46, 49, 52], thus requiring no oracle information and enabling a real-time system. Without loss of generality, for the head pose estimation we adopt the publicly available algorithm of [32].

**LSTM models.** LSTM models [26] are employed in those tasks where the output is conditioned on a varying number of inputs [20, 56], notably hand writing generation [19] and tracking [10].

As for trajectory forecasting, [3] models pedestrians as LSTMs that share their hidden states through a "social" pooling layer, avoiding to forecast colliding trajectories. This idea has been successfully adopted by [55], and further developed in [48] for modeling the tracking dynamics. A similar idea has been embedded directly in the LSTM memory unit as a regularization that models the local spatial and temporal dependency between neighboring pedestrians [22, 50]. As written above, here we modify the social pooling by considering a visibility attentional area driven by the head pose.

In most of the cases, the training of LSTMs for forecasting minimizes the negative log-likelihood over Gaussians [3, 55] or mixture of Guassians [19]. In general, when it comes to Gaussian log-likelihood loss functions, only bidimensional data (*i.e.* $(x, y)$ coordinates) have been considered so far, leading to the estimation of $2 \times 2$ covariance matrices. These can be optimized without considering

the positive semidefinite requirement [18], which is one of the most important problems for the covariances obtained by optimization [41] (see Sec. 3.4). Here for the first time, we study the problem of optimizing Gaussian parameters of higher dimensionality.

## 3. Our approach

In this section we present the *MX-LSTM*, capable of jointly forecasting positions and head orientations of an individual thanks to the presence of two information streams: Tracklets and *vislets*.

### 3.1. Tracklets and vislets

Given a subject $i$, a tracklet (see Fig. 1a) ) is formed by consecutive $(x, y)$ positions on the ground plane, $\{\mathbf{x}_t^{(i)}\}_{t=1,\dots,T}$, $\mathbf{x}_t^{(i)} = (x, y) \in \mathcal{R}^2$, while a vislet is formed by anchor points $\{\mathbf{a}_t^{(i)}\}_{t=1,\dots,T}$, with $\mathbf{a}_t^{(i)} = (a_x, a_y) \in \mathcal{R}^2$ indicating a reference point at a fixed distance $r$ from the corresponding $\mathbf{x}_t^{(i)}$, towards which the face is oriented[1]. In

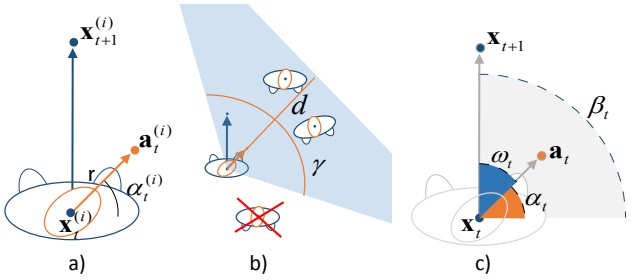


Figure 1. Graphical explanations; a) tracklets $\mathbf{x}_t^{(i)}$ and $\mathbf{x}_{t+1}^{(i)}$ and vislet anchor point $\mathbf{a}_t^{(i)}$; b) visual frustum pooling; c) angles for the correlation analysis (see Sec. 4).

practice, $\mathbf{a}_t^{(i)}$ is a fixed size vector originating from $\mathbf{x}_t^{(i)}$, whose direction implicitly indicates the pan angle $\alpha_t^{(i)}$ of the head. In principle, it would be possible to encode the head orientation directly with an angle. We prefer the vislet representation because it does not show discontinuity (between 360° and 0°) and because it is closer to the $(x, y)$ position representation and therefore more convenient for the vislet-position interplay.

In the forecasting literature [3, 53, 59] it is assumed that the prediction follows an "observation" period in which ground-truth data is fed into the machine. Here, the observation tracklets and vislets are fed into the MX-LSTM, which mixes together the two streams to understand their relationship, providing a joint prediction. In the experiments we evaluate the cases in which the past vislets are ground truth, but also the "in-the-wild" case, in which head pose is given by a real head detector. In this way, MX-LSTM will require no additional annotations in respect with former approaches.

A single MX-LSTM is instantiated for each pedestrian $i$, accepting tracklets and vislets with two separate embedding functions:

$$\mathbf{e}_t^{(x,i)} = \phi\left(\mathbf{x}_t^{(i)}, \mathbf{W}_x\right) \quad (1)$$

$$\mathbf{e}_t^{(a,i)} = \phi\left(\mathbf{a}_t^{(i)}, \mathbf{W}_a\right) \quad (2)$$

where the embedding function $\phi$ consists in a linear projection through the embedding weigths $\mathbf{W}_x$ and $\mathbf{W}_a$ into a $D$-dimensional vector, multiplied by a RELU nonlinearity, where $D$ is the dimension of the hidden space.

### 3.2. VFOA social pooling

The social pooling introduced in [3] is an effective way to let the LSTM capture how people move in a crowded scene avoiding collisions. This work considers an isotropic interest area around the single pedestrian, in which the hidden states of the the neighbors are considered, including those which are *behind* the pedestrian. In our case, we improve this module using the vislet information by selecting which individuals to consider, by building a view frustum of attention (VFOA), that is a triangle originating from $\mathbf{x}_t^{(i)}$, aligned with $\mathbf{a}_t^{(i)}$, and with an aperture given by the angle $\gamma$ and a depth $d$; these parameters have been learned by cross-validation on the training partition of the TownCentre dataset (see Sec. 5).

Our view-frustum social pooling is a $N_o \times N_o \times D$ tensor, in which the space around the pedestrian is divided into a grid of $N_o \times N_o$ cells as in [3], in which the VFOA is located, acting as the new interest region where people have to be taken into account. The pooling occurs as follows:

$$\mathbf{H}_t^{(i)}(m, n, :) = \sum_{j \in VFOA_i} \mathbf{h}_t^{(j)}, \quad (3)$$

where the $m$ and $n$ indices run over the $N_o \times N_o$ grid and the condition $j \in VFOA_i$ is satisfied when the subject $j$ is in the VFOA of subject $i$. The pooling vector is then embedded into a $D$-dimensional vector by

$$\mathbf{e}_t^{(H,i)} = \phi(\mathbf{H}_t^{(i)}, \mathbf{W}_H). \quad (4)$$

Finally, the MX-LSTM recursion equation is

$$\mathbf{h}_t^{(i)} = LSTM\left(\mathbf{h}_{t-1}^{(i)}, \mathbf{e}_t^{(x,i)}, \mathbf{e}_t^{(a,i)}, \mathbf{e}_t^{(H,i)}, \mathbf{W}_{\text{LSTM}}\right). \quad (5)$$

### 3.3. LSTM recursion

In principle (but in the next subsection we will ultimately modify the formulation), the hidden state is enforced to contain the parameters of a four dimensional Gaussian multivariate distribution $\mathcal{N}(\mu_t^{(i)}, \mathbf{\Sigma}_t^{(i)})$ as follows:

$$[\mu_t^{(i)}, \hat{\mathbf{\Sigma}}_t^{(i)}] = \mathbf{W}_o \mathbf{h}_{t-1}^{(i)}, \quad (6)$$

[1] The distance $r$ is not influent in this work, and it can be any value; in this work we set it at 0.5 for the visualization sake.

where $\hat{\mathbf{\Sigma}}_t^{(i)}$ is the vectorized version of $\mathbf{\Sigma}_t^{(i)}$. In practice $\mu_t^{(i)} = [\mu_t^{(x,i)}, \mu_t^{(y,i)}, \mu_t^{(a_x,i)}, \mu_t^{(a_y,i)}]$ and $\mathbf{\Sigma}_t^{(i)}$ contains the covariances among the $(x, y)$ coordinate distributions of the tracklets and the vislets. The distribution is then sampled to generate the joint prediction of tracklets and vislet points $[\hat{\mathbf{x}}_t, \hat{\mathbf{a}}_t]$. In other words, we are able at the same time of forecasting trajectries and head poses.

The weight parameters of the LSTM are found by minimizing the multivariate Gaussian log-likelihood for the $i-$th trajectory

$$L^i(\mathbf{W}_x, \mathbf{W}_a, \mathbf{W}_H, \mathbf{W}_{\text{LSTM}}, \mathbf{W}_o) \quad = \\ -\sum_{T_{obs}+1}^{T_{pred}} log\left(P([\mathbf{x}_t^{(i)}, \mathbf{a}_t^{(i)}], \mu_t^{(i)}, \mathbf{\Sigma}_t^{(i)})\right), \tag{7}$$

where $T_{obs}$ is the time frame until when the ground truth data is observed by the LSTM, while $T_{obs}+1, \ldots, T_{pred}$ are the time frames for which is requested the prediction. The loss of Eq. 7 is minimized over all the training sequences, and to prevent overfitting we include an $l_2$ regularization term.

### 3.4. MX-LSTM optimization

The optimization provides the weight matrices of the MX-LSTM, which in turn produce the set of Gaussian parameters, including the full covariance $\mathbf{\Sigma}$. The latter is needed to enforce the LSTM in encoding the relations among the $(x, y)$ coordinate distributions of the tracklets and the vislets, which we will further discuss in Sec. 4.

In general, the estimation of a full covariance matrix through optimization of an objective function (as the log-likelihood of Eq.(7) ) is a difficult numerical problem [41], since one must guarantee that the resulting estimate is a proper covariance, *i.e.,* a positive semi-definite (p.s.d.) matrix.

LSTMs involving log-likelihood losses over Gaussian distributions have been restricted so far to two dimensions for simple Gaussian [3] or mixture of Gaussian [19] distributions, in which the $2 \times 2$ covariance matrices have been obtained by simply optimizing the scalar correlation index $\rho_{x,y}$, which becomes the covariance term of $\mathbf{\Sigma}$ with $\sigma_{x,y} = \rho_{x,y}\sigma_x\sigma_y$ [19]. In the case of higher dimensional problems, pairwise correlation terms cannot be optimized and used to build $\mathbf{\Sigma}$, since the optimization process for each correlation term is independent from each other, while the positive-definiteness is a simultaneous constraint on multiple variables [42]. This lacks of coordination provides matrices far from being s.d.p., that in turns require a correction procedures by projecting into the closest s.d.p. matrix using, for instance, a cost function based on the Frobenious norm [7, 25]. These procedures are costly [41], and difficult to be embedded into the optimization process [12], especially in the case of the LSTM, where nonlinearities due to the embedding weights make the analytical derivation hard to formulate. So far, no LSTM loss has involved full covariances of dimension $> 2$.

Our solution involves unconstrained optimization, where an opportune parameterization of the variables to learn enforces the positive semi-definite constraint, which is easier to express, dramatically improving the convergence properties of the optimization algorithm.

In practice, we consider the Choleski family of parameterizations [42]: let $\mathbf{\Sigma}$ denote a definite positive $n \times n$ (in our case, $n = 4$) covariance matrix. Since $\mathbf{\Sigma}$ is symmetric, only $n(n+1)/2$ parameters are requested to represent it. The Choleski factorization is given by:

$$\mathbf{\Sigma} = \mathbf{L}^T\mathbf{L}, \tag{8}$$

where $\mathbf{L}$ is a $n \times n$ upper triangular matrix. In practice, the optimization process would focus on finding the $n(n+1)/2$ distinct scalar values for $\mathbf{L}$, which then solve for the covariance given Eq. (8). One problem with the Cholesky factorization is its non-uniqueness: any matrix obtained by multiplying a subset of the rows of $\mathbf{L}$ by -1 is valid; as a consequence, non-uniqueness of the solution makes the optimization process hard to converge. To make $\mathbf{L}$ unique, its diagonal elements have to be all positive. To this end, the Log-Cholesky parameterization [42] assumes that the values found by the optimizer of the main covariance diagonal are the log of the values of $\mathbf{L}$: Formally, the values found by the optimizer can be written as

$$\theta_L = \begin{bmatrix} \log l_{1,1} & l_{1,2} & l_{1,3} & l_{1,4} \\ 0 & \log l_{2,2} & l_{2,3} & l_{2,4} \\ 0 & 0 & \log l_{3,3} & l_{3,4} \\ 0 & 0 & 0 & \log l_{4,4} \end{bmatrix}$$

. In practice, after the estimation of $\mathbf{W}_x$, $\mathbf{W}_a$, $\mathbf{W}_H$, $\mathbf{W}_{\text{LSTM}}$, $\mathbf{W}_o$ parameters, the values of $\theta_L$ are extracted by

$$[\mu_t^{(i)}, \hat{\theta_{L_t}}^{(i)}] = \mathbf{W}_o\mathbf{h}_{t-1}^{(i)}, \tag{9}$$

where $\hat{\theta_L}$ is the vectorized version of $\theta_L$. Then, the diagonal values of $\theta_L$ are exponentiated to form $\mathbf{L}$ and obtaining $\mathbf{\Sigma}$ through Eq. (8).

## 4. Motivation for the MX-LSTM

So far, no quantitative studies focused on how head pose knowledge impacts on the trajectory forecasting. Here, we show a preliminary analysis of the common forecasting datasets with emphasis on the head pose, that motivated the design of the MX-LSTM.

In particular, we focus on the UCY dataset [34], composed by the Zara01, Zara02, and UCY sequences, which provides the annotations for the pan angle of the head pose of all the pedestrians. We also consider the Town Center

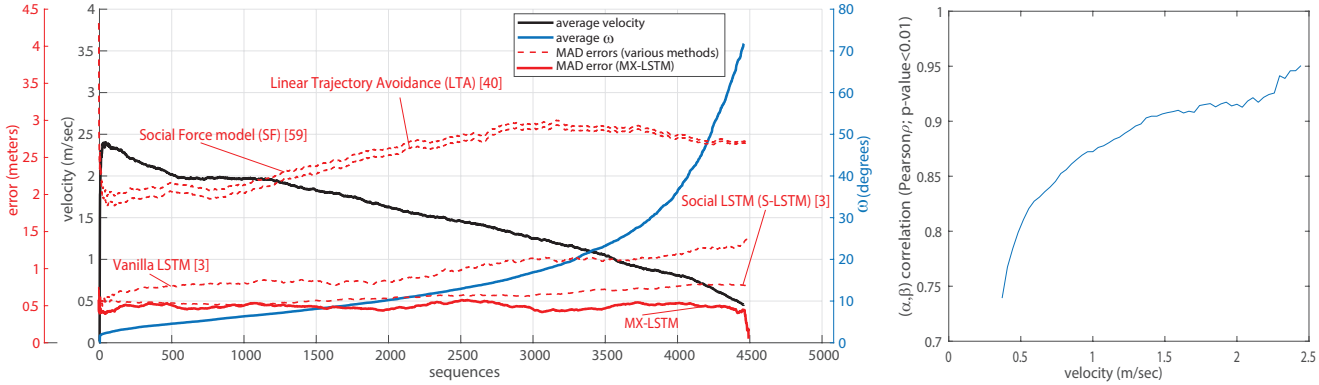


Figure 2. Motivating the MX-LSTM: a) analysis between the angle discrepancy $\omega$ between head pose and movement, the pedestrain smoothed velocity and the average errors of different approaches on the UCY sequence [34]; b) correlation between movement angle $\beta$ and head orientation angle $\alpha$ when the velocity is varying (better in color).

dataset [6], where we manually annotated the head pose, using the same annotation protocol of [34]. We discover the following facts[2]:

**1) People often do not watch their steps**. To show this fact, for each individual trajectory composed by $T$ frames (omitting the individual indices), we calculate all the $\alpha_t$, $\beta_t$ and $\omega_t$ of Fig.1c. The $\alpha_t$ is the head pose pan angle with respect to a given reference system; similarly, $\beta_t$ is the angle of motion, and $\omega_t$ is showing the discrepancy between the two. For each individual trajectory, we compute the average $\omega = 1/N \sum_{t=1...T} \omega_t$. On the multi-y-axis Fig. 2a, we show the $\omega$ value (in degrees) of all the sequences, in an increasing order (blue line and axis). From the figure, we omit those sequences where the speed is below 0.45m/sec.: in those cases the individual is essentially still and the movement vector $\overrightarrow{\boldsymbol{x_{t+1}} - \boldsymbol{x_t}}$ carry few if no meaning, and consequently the angle $\beta_t$ cannot be taken into account. The $\omega$ value ranges from $0.02°$ to $72°$. We conclude that in 25% of the video sequences the misalignment between the head pose and the step direction is larger than $20°$.

**2) Head pose and movements are (statistically) correlated**; On the same figure, we report the velocity curve (black solid line and axis), where each y-point gives the average speed of the $i-th$ ordered trajectory on the x-axis. For the sake of readability, the curve has been smoothed with moving average filter of size 10. As it shows, there is a relation of inverse proportionality between the $\omega$ and the pedestrian speed: the alignment between the head towards the direction of movement is higher when the speed is higher; when the person slows down the head pose is dramatically misaligned. The relation is statistically significant: we consider the Pearson circular correlation coefficient [28] between the angles $\alpha_t$ and $\beta_t$, computed over all the frames of the sequences considered for that figure. On the whole data, the correlation is 0.83 (p-value$<0.01$). We also investigate how the correlation changes with the speed: Fig. 2b shows the correlation values against velocity, computed by pooling the $\alpha_t$ and $\beta_t$ angles around a certain velocity value; in particular, each correlation value at velocity $\tau$ has been computed by considering all of the samples in the range $[\tau - 0.01R, \tau + 0.01R]$, where $R$ is the whole velocity range. All the reported values have statistical significance (p-value$<0.01$). The plot shows clearly that the correlation is lower at low velocities, where the discrepancy between the $\alpha_t$ and $\beta_t$ angles is in general higher. The challenge here is to investigate whether this discrepancy can be learned by the MX-LSTM to improve the forecasting. More intriguingly, MX-LSTM should learn how these relations evolve in time, which has not been investigated yet, since the analysis done so far consider each time instant as independent from each other.

**3) Forecasting errors are in general higher when the speed of the pedestrian is lower**; In Fig. 2 are reported the Mean Average Displacement (MAD) error [40] (red line and axis) of the following approaches: SF [59], LTA [53], vanilla LSTM and Social LSTM [3], together with our MX-LSTM approach. In general, lower velocities bring to higher errors, since when people are walking very slowly their behavior become less predictable, due to physical reasons (less inertia) but also behavioral (people walking slowly are usually involved into other activities, like talking with other people, looking around). On the contrary, it is shown here that MX-LSTM is performing well even at lower velocities, reaching errors very close to zero with static people (more details in Sec.5).

Summarizing, head pose is correlated with the movement, especially when people move fast. When people move slow, the correlation is weaker but significant, the prediction errors are larger, and the head pose is drastically

[2]Here is presented the analysis on the UCY sequence, which is similar to what we observed on the other sequences.

misaligned with the movement. These facts justify and motivate our objective with the MX-LSTM, to capture the head pose information jointly with the movement and use it for a better forecasting.

## 5. Experiments

We present here both quantitative and qualitative experiments. Quantitative results validate the proposed MX-LSTM model, setting the new state-of-the-art for trajectory forecasting; results are also provided for an ablation study showing the importance of the different parts of the MX-LSTM. Finally, we present the very first results on head pose forecasting. Qualitative results unveil the interplay between tracklets and vislets that the MX-LSTM has learnt.

### 5.1. Quantitative results

We evaluate our model against all the published approaches which made their code publicly available: Social Force model (SF) [59], Linear Trajectory Avoidance (LTA) [40], Vanilla LSTM and Social LSTM (S-LSTM) [3].

Experiments follow the widely-used evaluation protocol of [40], in which the algorithm first observes 8 "observation" ground truth (GT) frames of a trajectory, predicting the following 12 ones. For the three UCY sequences three models have been trained: for each one we used two sequences as training data and then we tested on the third sequence. For Town Centre dataset the model has been trained and tested on the respective provided sets. The grid for the social pooling (Eq. (3)) has $N_o \times N_o$ cells with $N_o = 32$. The view frustum aperture angle has been cross-validated on the training partition of the TownCenter and kept fixed for the remaining trials ($\gamma = 40°$), while the depth $d$ is simply bounded by the social pooling grid. Trajectory prediction performances are analyzed with the Mean Average Displacement (MAD) error (euclidean distance between predicted and GT points, averaged over the sequence), and Final Average Displacement (FAD) error (distance between the last predicted point and the corresponding GT point) [40].

Results are reported in Table 1. The MX-LSTM outperforms the state-of-the-art methods in every single sequence and with both metrics, with an average improvement of 32.7%. The highest relative gain is achieved in Zara02 dataset, where complex non-linear paths are mostly caused by standing conversational groups and people that walk close to them, avoiding collisions. People slowing down and looking at the window shops pose also a challenge. As shown in Fig. 2, slow moving and interacting pedestrians cause troubles to the competing methods, while MX-LSTM clearly overcomes such shortcomings denoting a better model. Qualitative motivations will follow in Sec. 5.2.

Please note that different methods rely on different input data: both SF and LTA require the destination point of each individual, while SF also requires annotations about social groups; MX-LSTM requires the head pose of each individual for the first 8 frames, but this can be estimated by a head pose estimator. This motivates our next experiment: we automatically estimate the head bounding box given the feet positions on the floor plane, assuming people being 1.80m tall. Then, we apply the head pose estimator of [32] which gives continuous angles that can be used as input of our approach now named "MX-LSTM-HPE". As shown by the scores in Table 1, MX-LSTM-HPE does not suffer about small errors in the input head pose, with an average drop in performances of only 5%. Note that MX-LSTM-HPE still outperforms all competing methods on all dataset even with the noisy estimated head pose information.

How accurate should the head pose estimation be, for the MX-LSTM-HPE to have convincing performances, for example outperforming the Social LSTM? We answer this question by corrupting the true head pose estimate with additive Gaussian noise$\sim \mathcal{N}(\alpha_t, \hat{\sigma})$, where $\alpha_t$ is the correct head pose and $\hat{\sigma}$ the standard deviation. MX-LSTM-HPE outperforms social-LSTM up to a noise of $\hat{\sigma} = 24°$.

#### 5.1.1 Ablation study

Aside with the models in the literature, we investigate three variations of the MX-LSTM to capture the net contributions of the different parts that characterize our approach.

**Block-Diagonal MX-LSTM (BD-MX-LSTM)**: it serves to highlight the importance of estimating full covariances to understand the interplay between tracklets and vislets. Essentially, the approach estimates two bidimensional covariances[3] $\mathbf{\Sigma}_x$ and $\mathbf{\Sigma}_a$ for the trajectory and the vislet modeling respectively, without capturing the cross-stream covariances.

**NoFrustum MX-LSTM**: this variation of the MX-LSTM uses social pooling as in [3], in which the interest area where people hidden states $\{\mathbf{h}_t^j\}$ are pooled into the social tensor all around the individual. In other words, no frustum selecting the people that have to be considered is used here.

**Individual MX-LSTM**: In this case, no social pooling is taken into account, therefore the embedding operation of Eq. (4) is absent, and the weight matrix $\mathbf{W}_H$ vanishes. In practice, this variant learns independent models for each person, each one considering the tracklet and vislet points.

Table 1, last three columns, reports numerical results for all the MX-LSTM simplifications on all the datasets. The main facts that emerge are: 1) the highest variations are with the Zara02 sequence, where MX-LSTM doubles the performances of the worst approach (Individual MX-LSTM); 2) the worst performing is in general Individual MX-LSTM,

[3]The $2 \times 2$ covariance is estimated employing two variances $\sigma_1, \sigma_2$ and a correlation terms $\rho$ as presented in [19] Eq.(24) and (25).

Table 1. Mean and Final Average Displacement errors (in meters) for all the methods on all the datasets. The first 5 columns are the comparative methods and our proposed model trained and tested with GT annotations. MX-LSTM-HPE is our model tested with the output of a real head pose estimator [32]. The last 3 columns are variations of our approach trained and tested on GT annotations.

| Metric | Dataset | SF [59] | LTA [40] | Vanilla LSTM [3] | Social LSTM [3] | MX-LSTM | *MX-LSTM-HPE* | Individual MX-LSTM | NoFrustum MX-LSTM | BD-MX-LSTM |
|---|---|---|---|---|---|---|---|---|---|---|
| MAD | Zara01 | 2.88 | 2.74 | 0.90 | 0.68 | **0.59** | ***0.66*** | 0.63 | 0.63 | 0.60 |
| | Zara02 | 2.32 | 2.23 | 1.09 | 0.63 | **0.35** | ***0.37*** | 0.72 | 0.36 | 0.41 |
| | UCY | 2.57 | 2.49 | 0.67 | 0.62 | **0.49** | ***0.55*** | 0.53 | 0.51 | 0.54 |
| | TownCentre | 9.35 | 9.14 | 4.62 | 1.96 | **1.15** | ***1.21*** | 2.09 | 1.70 | 1.40 |
| FAD | Zara01 | 5.55 | 5.55 | 1.85 | 1.53 | **1.31** | ***1.43*** | 1.37 | 1.40 | 1.51 |
| | Zara02 | 4.35 | 4.35 | 2.15 | 1.43 | **0.79** | ***0.82*** | 1.56 | 0.84 | 1.00 |
| | UCY | 4.62 | 4.66 | 1.39 | 1.40 | **1.12** | ***1.20*** | 1.16 | 1.15 | 1.23 |
| | TownCentre | 16.01 | 16.08 | 8.26 | 3.96 | **2.30** | ***2.38*** | 4.00 | 3.40 | 2.90 |

Table 2. Mean angular error (in degrees) for the state-of-the-art head pose estimator [32], and the MX-LSTM model fed with GT annotations and estimated values (MX-LSTM-HPE).

| Metric | Zara01 | Zara02 | UCY | Town Centre |
|---|---|---|---|---|
| HPE [32] | 14.29 | 20.02 | 19.90 | 25.08 |
| MX-LSTM | 12.98 | 20.55 | 21.36 | 26.48 |
| MX-LSTM-HPE | 17.69 | 21.92 | 24.37 | 28.55 |

showing that social reasoning is indeed necessary; 3) social reasoning is systematically improved with the help of the vislet-based view-frustum; 4) full covariance estimation has a role in pushing down the error which is already small with the adoption of vislets.

Summarizing the results so far, having vislets as input allows to definitely increase the trajectory forecasting performance, even if vislets are estimated with noise. Vislets should be used to understand social interactions with social pooling, by building a view frustum that tells which are the people currently observed by each individual. All of these features are done efficiently by the MX-LSTM: in fact the training time is the same with having an LSTM with social pooling.

#### 5.1.2 Head pose forecasting

As done with trajectories, we are also providing a forecast of the head pose of each individual at each frame which is a distinctive attribute of our method. We evaluate the performances of this estimation in terms of mean angular error $e_\alpha$, which is the mean absolute difference between the estimated pose (angle $\alpha_{t,\cdot}$ in Fig. 1c) and the annotated GT.

Table 2 shows numerical results of the static head pose estimator [32] (HPE), the MX-LSTM using GT head poses, and the MX-LSTM fed with the output of HPE during the observation period (MX-LSTM-HPE). In all the cases our forecast output is comparable with the one of HPE, but in our case we do not use appearance cues – *i.e.* we do not look at the images at all. In case of Zara01, the MX-LSTM is even better that the static prediction showing the forecasting power of our model. In our opinion this is due to the fact that in this sequence trajectories are mostly very linear and fast, and heads are mostly aligned with the direction of motion. When we provide estimations to the MX-LSTM model during the observation period, angular error increases, as expected. Despite this, the error is surprisingly limited.

### 5.2. Qualitative results

Fig. 3 shows qualitative results on the Zara02 dataset, which has been shown to be the most complex scenario in the quantitative experiments Fig. 3a presents MX-LSTM results: a group scenario is taken into account, with the attention focused on the girl in the bottom-left corner. In the left column, the green ground-truth prediction vislets show that the girl is conversing with the group members, with a movement close to zero and the pan head angle which oscillates. In magenta, the behavior of the S-LSTM, predicting erroneously the girl leaving the group. This error confirms the problem of competing methods in forecasting the motion of people slowly moving or static as discussed in Sec.4, and further confirmed by the results of the quantitative experiments. In the central column, the observation sequence given to the MX-LSTM is shown in orange (almost static with oscillating vislets). The related prediction (in yellow) shows oscillating vislets, and almost no movement, confirming that the MX-LSTM has learnt this particular social behavior. If we provide to MX-LSTM an artificial observation sequence with the annotated positions (real trajectory) but vislets oriented toward west (third column, orange arrows), where no people are present, the MX-LSTM predicts a trajectory departing from the group (cyan trajectory and arrows).

The two rows of Fig. 3b) analyze the Individual MX-LSTM, in which no social pooling is taken into account. Therefore, here each pedestrian is not influenced by the surrounding people, and the relationship between the tracklets and the vislets in the prediction can be observed without any confounding factor. Fig. 3b) first row shows three situations in which the vislets of the observation sequence are artificially made pointing north (orange arrows), resulting not aligned with the trajectory. In this case the Individual MX-LSTM predicts a decelerating trajectory drifting toward north (magenta trajectory and vislets), especially in

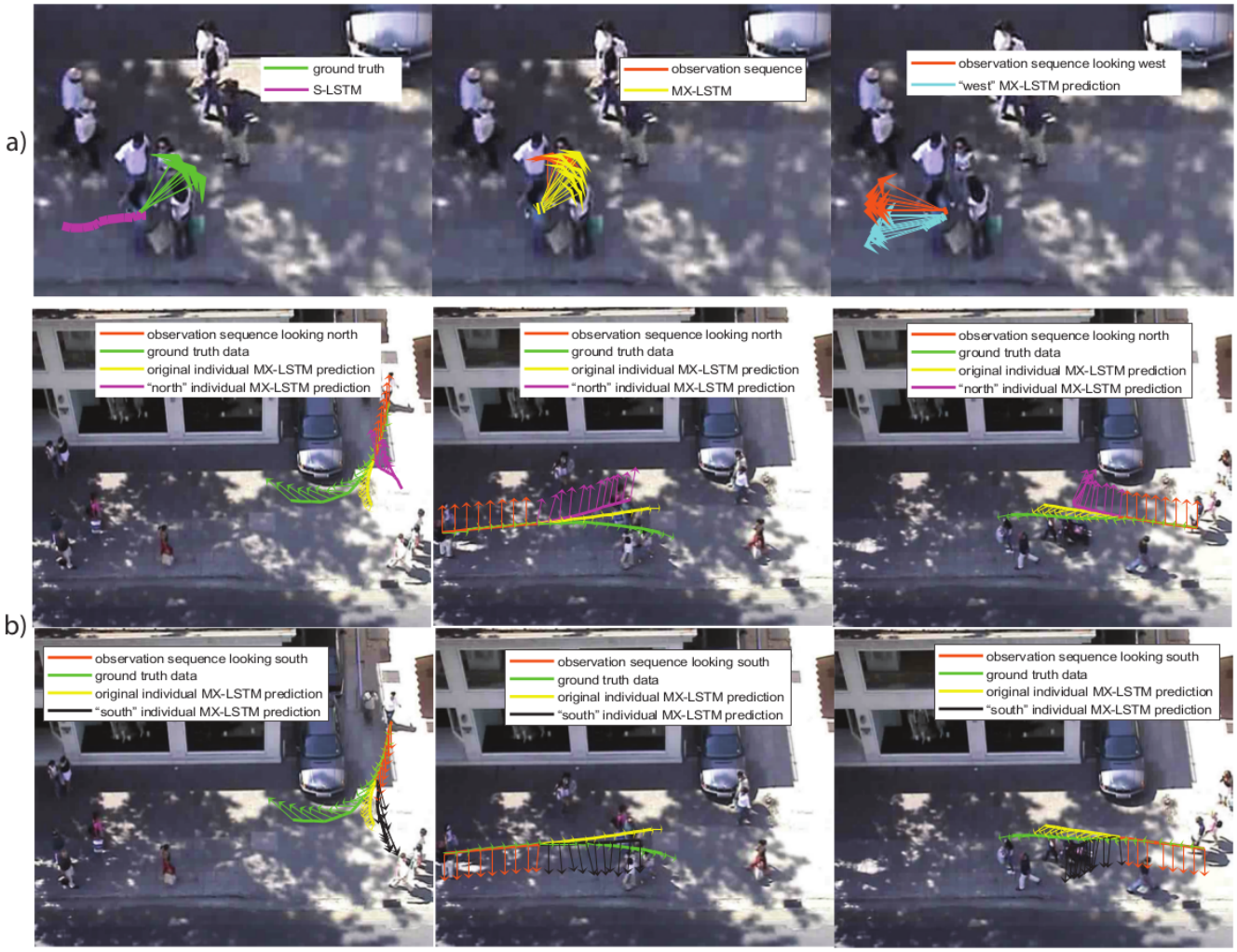


Figure 3. Qualitative results: a) MX-LSTM b) Ablation qualitative study on Individual MX-LSTM (better in color).

the second and third pictures. If the observation has the legit vislets (green arrows, barely visible since they are aligned with the trajectory), the resulting trajectory (yellow trajectory and vislets) has a different behavior, closer to the GT (green trajectory and vislets). The second row is similar, with the observation vislets made pointing to south. The prediction with the modified vislets is in black. The only difference is in the bottom left picture: here the observation vislets pointing south are in accord with the movement, so that the resulting predicted trajectory is not decelerating as in the other cases, but accelerating toward south.

## 6. Conclusion

This paper showed that sequences of consecutive head poses, *i.e.*, the vislets, are of great help for trajectory forecasting. We introduced a model to incorporate vislets and tracklets, the MX-LSTM, which mixes together the two streams of information providing cross-stream $4 \times 4$ covariances, that explain how head poses and positions on the plane are correlated, providing accurate forecasting prediction for both of them. This has been possible thanks to an optimization process embedded into the LSTM backpropagation which uses a log-Cholesky parameterization, leading to unconstrained optimization. We believe that consideration of vislets would allow us, in future work, to also encode specific areas of interest into the trajectory forecasting.

**Acknowledgements**: This work has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie Grant Agreement No. 676455, and has been partially supported by the POR FESR 2014-2020 Work Program (Action 1.1.4, project No.10066183).

## References

[1] P. Abbeel and A. Y. Ng. Apprenticeship learning via inverse reinforcement learning. In *ICML*, 2004. 2

[2] H. Akaike. Fitting autoregressive models for prediction. *Annals of the institute of Statistical Mathematics*, 21(1):243–247, 1969. 2

[3] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese. Social LSTM: Human trajectory prediction in crowded spaces. In *CVPR*, 2016. 1, 2, 3, 4, 5, 6, 7

[4] A. Alahi, V. Ramanathan, and L. Fei-Fei. Socially-aware large-scale crowd forecasting. In *CVPR*, 2014. 2

[5] S. O. Ba and J.-M. Odobez. A probabilistic framework for joint head tracking and pose estimation. In *ICPR*, 2004. 2

[6] B. Benfold and I. Reid. Stable multi-target tracking in real-time surveillance video. In *CVPR*, 2011. 5

[7] S. Boyd and L. Xiao. Least-squares covariance matrix adjustment. *SIAM Journal on Matrix Analysis and Applications*, 27(2):532–546, 2005. 4

[8] J. F. Caminada and W. J. M. van Bommel. Philips engineering report 43, 1980. 1, 2

[9] C. Chen and J.-M. Odobez. We are not contortionists: Coupled adaptive learning for head and body orientation estimation in surveillance video. In *CVPR*, 2012. 1

[10] H. Coskun, F. Achilles, R. Di Pietro, N. Navab, and F. Tombari. Long short-term memory kalman filters: Recurrent neural estimators for pose regularization. In *ICCV*, 2017. 2

[11] N. Davoudian and P. Raynham. What do pedestrians look at at night? *Lighting Research and Technology*, 44(4):438–448, 2012. 1, 2

[12] J. E. Dennis Jr and R. B. Schnabel. *Numerical methods for unconstrained optimization and nonlinear equations*. SIAM, 1996. 4

[13] A. D. Dragan, N. D. Ratliff, and S. S. Srinivasa. Manipulation planning with goal sets using constrained trajectory optimization. In *ICRA*, 2011. 1, 2

[14] S. Fotios, J. Uttley, C. Cheal, and N. Hara. Using eye-tracking to identify pedestrians' critical visual tasks, Part 1. Dual task approach. *Lighting Research & Technology*, 47(2):133–148, 2015. 1, 2

[15] S. Fotios, J. Uttley, and B. Yang. Using eye-tracking to identify pedestrians' critical visual tasks. part 2. fixation on pedestrians. *Lighting Research & Technology*, 47(2):149–160, 2015. 1, 2

[16] T. Foulsham, E. Walker, and A. Kingstone. The where, what and when of gaze allocation in the lab and the natural environment. *Vision research*, 51(17):1920–1931, 2011. 1, 2

[17] N. Gourier, J. Maisonnasse, D. Hall, and J. L. Crowley. Head pose estimation on low resolution images. In *CLEAR*, 2006. 2

[18] A. Graves. *Supervised sequence labelling with recurrent neural networks*, volume 385. Springer, 2012. 3

[19] A. Graves. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*, 2013. 2, 4, 6

[20] K. Gregor, I. Danihelka, A. Graves, D. J. Rezende, and D. Wierstra. DRAW: A recurrent neural network for image generation. *arXiv preprint arXiv:1502.04623*, 2015. 2

[21] E. T. Hall. *The hidden dimension*. Doubleday & Co, 1966. 1

[22] Y. D. B. Z. Hang Su, Jun Zhu. Forecast the plausible paths in crowd scenes. In *IJCAI*, 2017. 1, 2

[23] I. Hasan, T. Tsesmelis, F. Galasso, A. Del Bue, and M. Cristani. Tiny head pose classification by bodily cues. In *ICIP*, 2017. 2

[24] D. Helbing and P. Molnar. Social force model for. *Physical review E*, 51(5):4282, 1995. 2

[25] N. J. Higham. Computing a nearest symmetric positive semidefinite matrix. *Linear algebra and its applications*, 103:103–118, 1988. 4

[26] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997. 1, 2

[27] J. Intriligator and P. Cavanagh. The spatial resolution of visual attention. *Cognitive psychology*, 43(3):171–216, 2001. 1, 2

[28] S. R. Jammalamadaka and A. Sengupta. *Topics in circular statistics*, volume 5. World Scientific, 2001. 5

[29] R. E. Kalman et al. A new approach to linear filtering and prediction problems. *ASME Journal of Basic Engineering*, 1960. 2

[30] K. Kitani, B. Ziebart, J. Bagnell, and M. Hebert. Activity forecasting. In *ECCV*, 2012. 1, 2

[31] M. Kuderer, H. Kretzschmar, C. Sprunk, and W. Burgard. Feature-based prediction of trajectories for socially compliant navigation. In *Robotics: science and systems*, 2012. 1

[32] D. Lee, M.-H. Yang, and S. Oh. Fast and accurate head pose estimation via random projection forests. In *ICCV*, 2015. 2, 6, 7

[33] N. Lee and K. M. Kitani. Predicting wide receiver trajectories in american football. In *WACV*, 2016. 1, 2

[34] A. Lerner, Y. Chrysanthou, and D. Lischinski. Crowds by example. In *Computer Graphics Forum*, 2007. 2, 4, 5

[35] W.-C. Ma, D.-A. Huang, N. Lee, and K. M. Kitani. Forecasting interactive dynamics of pedestrians with fictitious play. In *CVPR*, 2017. 1, 2

[36] J. Mainprice, R. Hayne, and D. Berenson. Goal set inverse optimal control and iterative replanning for predicting human reaching motions in shared workspaces. *IEEE Trans. on Robotics*, 32(4):897–908, 2016. 1, 2

[37] P. McCullagh and J. A. Nelder. Generalized linear models, no. 37 in monograph on statistics and applied probability, 1989. 2

[38] B. T. Morris and M. M. Trivedi. A survey of vision-based trajectory learning and analysis for surveillance. *IEEE Trans. on Circuits and Systems for Video Technology*, 18(8):1114–1127, 2008. 2

[39] A. E. Patla and J. N. Vickers. How far ahead do we look when required to step on specific locations in the travel path during locomotion? *Experimental brain research*, 148(1):133–138, 2003. 1, 2

[40] S. Pellegrini, A. Ess, K. Schindler, and L. Van Gool. You'll never walk alone: Modeling social behavior for multi-target tracking. In *ICCV*, 2009. 1, 2, 5, 6, 7

[41] J. C. Pinheiro and D. M. Bates. Unconstrained parametrizations for variance-covariance matrices. *Statistics and Computing*, 6(3):289–296, 1996. 3, 4

[42] M. Pourahmadi. Covariance estimation: The GLM and regularization perspectives. *Statistical Science*, pages 369–387, 2011. 4

[43] M. B. Priestley. *Spectral analysis and time series*. Academic press, 1981. 2

[44] J. Quiñonero-Candela and C. E. Rasmussen. A unifying view of sparse approximate gaussian process regression. *Journal of Machine Learning Research*, 6(12):1939–1959, 2005. 2

[45] C. E. Rasmussen. Gaussian processes for machine learning. In *Adaptive Computation and Machine Learning*, 2006. 2

[46] N. M. Robertson and I. D. Reid. Estimating gaze direction from low-resolution faces in video. In *ECCV*, 2006. 1, 2

[47] A. Robicquet, A. Sadeghian, A. Alahi, and S. Savarese. Learning social etiquette: Human trajectory understanding in crowded scenes. In *ECCV*, 2016. 1

[48] A. Sadeghian, A. Alahi, and S. Savarese. Tracking the untrackable: Learning to track multiple cues with long-term dependencies. *arXiv preprint arXiv:1701.01909*, 2017. 1, 2

[49] R. Stiefelhagen, M. Finke, J. Yang, and A. Waibel. From gaze to focus of attention. In *VISUAL*, 1999. 1, 2

[50] H. Su, Y. Dong, J. Zhu, H. Ling, and B. Zhang. Crowd scene understanding with coherent recurrent neural networks. In *IJCAI*, 2016. 1, 2

[51] L. Sun, Z. Yan, S. M. Mellado, M. Hanheide, and T. Duckett. 3DOF pedestrian trajectory prediction learned from long-term autonomous mobile robot deployment data. *arXiv preprint arXiv:1710.00126*, 2017. 1

[52] D. Tosato, M. Spera, M. Cristani, and V. Murino. Characterizing humans on riemannian manifolds. *IEEE TPAMI*, 35(8):1972–1984, 2013. 2

[53] P. Trautman and A. Krause. Unfreezing the robot: Navigation in dense, interacting crowds. In *IROS*, 2010. 1, 3, 5

[54] P. Vansteenkiste, G. Cardon, E. D'Hondt, R. Philippaerts, and M. Lenoir. The visual control of bicycle steering: The effects of speed and path width. *Accident Analysis & Prevention*, 51:222–227, 2013. 1, 2

[55] D. Varshneya and G. Srinivasaraghavan. Human trajectory prediction using spatially aware deep attention models. In *NIPS*, 2017. 1, 2

[56] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. In *CVPR*, 2015. 2

[57] J. M. Wang, D. J. Fleet, and A. Hertzmann. Gaussian process dynamical models for human motion. *IEEE TPAMI*, 30(2):283–298, 2008. 2

[58] C. K. I. Williams. Prediction with gaussian processes: From linear regression to linear prediction and beyond. In *Learning in graphical models*, pages 599–621. Springer, 1998. 2

[59] K. Yamaguchi, A. C. Berg, L. E. Ortiz, and T. L. Berg. Who are you with and where are you going? In *CVPR*, 2011. 1, 3, 5, 6, 7

[60] S. Yi, H. Li, and X. Wang. Understanding pedestrian behaviors from stationary crowd groups. In *CVPR*, 2015. 2

[61] B. D. Ziebart, A. Maas, J. A. Bagnell, and A. K. Dey. Maximum entropy inverse reinforcement learning. In *AAAI*, 2008. 2

[62] B. D. Ziebart, N. Ratliff, G. Gallagher, C. Mertz, K. Peterson, J. A. Bagnell, M. Hebert, A. K. Dey, and S. Srinivasa. Planning-based prediction for pedestrians. In *IROS*, 2009. 1