On Exposing the Challenging Long Tail in Future Prediction of Traffic Actors

Osama Makansi   Özgün Çiçek   Yassine Marrakchi   Thomas Brox
University of Freiburg
`makansio,cicek,marrakch,brox@cs.uni-freiburg.de`

Abstract

Predicting the states of dynamic traffic actors into the future is important for autonomous systems to operate safely and efficiently. Remarkably, the most critical scenarios are much less frequent and more complex than the uncritical ones. Therefore, uncritical cases dominate the prediction. In this paper, we address specifically the challenging scenarios at the long tail of the dataset distribution. Our analysis shows that the common losses tend to place challenging cases sub-optimally in the embedding space. As a consequence, we propose to supplement the usual loss with a loss that places challenging cases closer to each other. This triggers sharing information among challenging cases and learning specific predictive features. We show on four public datasets that this leads to improved performance on the challenging scenarios while the overall performance stays stable. The approach is agnostic w.r.t. the used network architecture, input modality or viewpoint, and can be integrated into existing solutions easily. Code is available at github.

1. Introduction

Future prediction in traffic scenarios aims to foresee the future location of dynamic actors based on their current and previous locations and possibly other information about the environment. For an actor in interaction with others, reasoning about possible future locations of the other actors is necessary for path planning and to avoid collisions. Given enough data, some recent prediction methods [48, 60, 47] also not just predict a single location of the actor in the future but a multimodal distribution over possible future locations.

The average prediction errors of such methods look promising, but they hide that the training and test data is dominated by simple scenarios, where the trajectory can be smoothly propagated into the future. Such scenarios can be handled with a simple Kalman filter or other autoregressive models. However, the most safety-critical scenarios are those that involve close-by dynamic obstacles and require
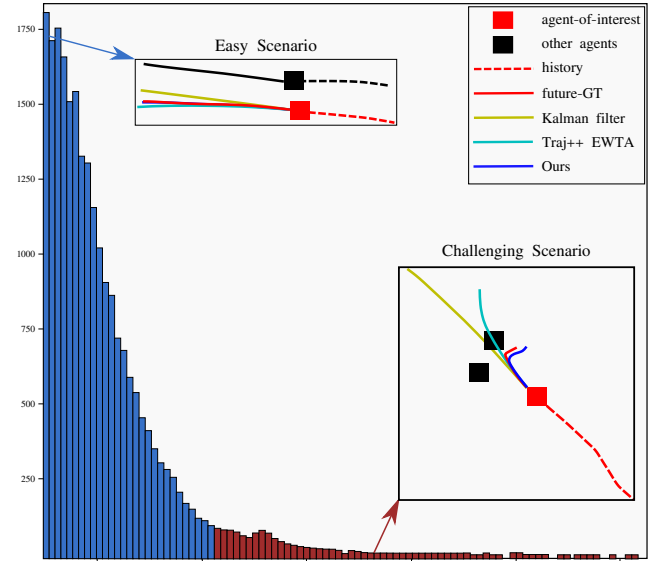

Figure 1. Histogram of the ETH-UCY dataset based on the difficulty of the sample (based on displacement error of a Kalman filter [31]). An easy scenario (belongs to the head blue class) and a challenging scenario (belongs to the tail red class) are shown along the prediction of the state-of-the-art (Traj++ EWTA) and our approach. Our approach targets those challenging scenarios (from the tail) and improves their performance while maintaining a good performance on the easy scenarios.

an evasive maneuver. Such scenarios are rare in both the training and the test data. The more complex and safety-critical they are, the less frequent they are. Fatal cases with a collision are not part of the dataset at all.

As an example, the ETH-UCY dataset is often used to benchmark methods for future trajectory prediction. It is considered a challenging dataset, as it includes interacting pedestrians in crowded scenes. Figure 1 shows the histogram of samples in this dataset based on their difficulty approximated by the prediction error of a Kalman filter. The large majority of scenarios can be well modeled by linear extrapolation, whereas scenarios that require more complex modeling are rare. The depicted challenging scenario showcases a pedestrian (red box), who will turn right in the future to avoid a collision with the stationary pedestrians (black boxes) in front of them.

1

In this paper, we explicitly address the long-tailed data distribution in future prediction and focus on the rare but important cases rather than the average case. Straightforward ideas to re-distribute the dataset by undersampling the frequent scenarios [20, 27] or by reweighting the loss for these samples [13] are not viable solutions, since it would reduce the (effective) data size dramatically. One can also oversample the challenging scenarios during training [26, 36], yet this repetition of the same rare samples leads to overfitting and does not perform well, as we show in our experiments. Some works have tried to simulate rare cases [43, 42]. However, to-date, even the most realistic simulations suffer from the domain gap between the simulated and the real world. An interesting direction for dealing with imbalanced data has been presented by Cao et al., who proposed a loss that ensures larger margins for the minority [5].

We pick up this general idea and propose to reshape the feature embedding of the predictor. We show in a detailed analysis of the feature space that, with the usual loss, the challenging examples get placed next to many normal cases. Consequently, the relevant information of these samples gets smoothed out. As we push the challenging scenarios to be in proximity in the embedding, more of these samples that share a similar scenario build a small cluster and are no longer ignored. With this approach we can predict the future trajectory of interacting pedestrians better; see blue trajectory in Figure 1.

Our contributions can be briefly summarized as follows. (1) We analyze the problem of long-tailed data distributions in future prediction for the first time. (2) We propose a novel joint optimization of the regular regression loss for predicting the future location and a loss that reshapes the feature embedding in favor of the long-tail samples. (3) We show that multi-headed networks outperform cVAEs in addressing the multimodal nature of the future.

The proposed approach is easy to integrate into existing approaches, since it is agnostic to the network architecture, viewpoint, and input modalities. We demonstrate this by evaluating on four diverse public datasets. On each of them, the method improves the prediction quality of the challenging cases, while maintaining the quality on simple cases.

2. Related Work

Future prediction. Deep learning methods dominate future prediction. LSTMs [1, 71, 75, 2, 52, 59, 16] were mostly used to model the states of the agents over time, while graph-based approaches [67] were used to model the interactions between agents. However, these methods cannot handle the multimodal nature of the future. Meanwhile, several works addressed the multimodality in future prediction by cVAEs [39, 51], GANs [22, 58, 77, 37, 66], non-parametric approaches [43, 10] or a sampling-fitting framework [48]. Recently, graph neural networks [53, 60, 40, 6,

44] and transformers [74] have become popular to model the agent interactions. All aforementioned works assume that the scene is static and is observed from a bird's-eye view. Among these, Trajectron++ [60] currently performs the best.

In automotive settings, the observation is typically from an egocentric view (e.g, with camera(s) or LiDAR mounted on the vehicle). This introduces new challenges due to the large egomotion of the vehicle and the narrow field of view. Multiple works project the data to the bird's-eye view using expensive 3D sensors [12, 17, 15, 63, 46, 57, 11]. Some recent approaches work directly on the egocentric view. Deterministic approaches [64, 65] modeled the motion of the scene via optical flow. Yao et al. [73] proposed to use the planned egomotion to improve the predictions. TraPHic [7] exploited the interaction between nearby heterogeneous objects via LSTMs. Some works also tackled the multimodality in future prediction by using Bayesian RNNs to sample multiple futures with uncertainties [3, 49]. Titan [50] modeled the future as a bi-variate Gaussian and conditioned the learning process on a set of labelled prior actions to further improve the prediction. Makansi et al. [47] proposed a three-staged framework FLN-RPN, which currently performs the best in the egocentric view.

None of the above approaches addressed the long tail of the data distribution. We base our method on Trajectron++ [60] in the bird's-eye setting and FLN-RPN [47] for the egocentric setting, and specifically address the challenging cases in the long tail of the dataset distribution for the first time.

Learning on imbalanced datasets. Issues with the long tail of a dataset have been well studied for classification tasks. Many works tackled the issue from the data side. A common approach is oversampling of rare classes [61, 56, 14]. Another option is undersampling of the most frequent classes [20, 27]. Several works follow the idea of generating more samples of the minority classes by simulation, which can be considered a more sophisticated version of oversampling [8, 24, 54, 36]. Instead of changing the number of samples, samples can also be reweighted in the loss [29, 13, 45, 62]. Some works proposed to learn these weights [33, 30]. Recently, Li et al. [41] group classes of similar sizes and learn group-wise classifiers.

Another idea is to design loss functions that affect the feature space by increasing the inter-class distance and reducing the intra-class distance [76, 29]. This concept of enlarging the margin between minority classes leads to a larger margin between classes and, thus, better generalization [18, 5, 34, 25]. Similarly, contrastive learning has become very popular due to promising results on self-supervised feature learning with noise-contrastive learning [23, 9, 69]. Noisy versions of a sample (positives) are forced to be separated from other samples (negatives) [19]. Recently, con-

trastive learning enabled learning stronger feature extractors for classifying long-tail datasets [72].

All these methods were applied to classification tasks, where there is an explicit distinction between frequent and rare classes. Our approach also augments the loss to reshape the data distribution in the embedding space, yet we do not rely on predetermined clusters, since we have a regression task. Given the flexibility of contrastive learning in defining losses based on the definition of positive and negative samples, we adopt a novel way of embedding the samples based on their difficulty as measured by the performance of a Kalman filter and combine the reshaping of the embedding space with the regular regression loss. For sake of fair comparison, we also adapt previous methods tailored for classification and use them in conjunction with the regression loss as detailed in Section 6.5.

3. Future Prediction

Given current and past observations $(\mathbf{x}_{t-h}, ..., \mathbf{x}_t)$, where h is the length of the history, future prediction aims to predict the true state $\mathbf{y}$ of the actor of interest at times $(t + \Delta t, ..., t + M\Delta t)$ in the future. An observation $\mathbf{x}$ at a single time step t can consist of the 2D location $\mathbf{p}^t = (p_x, p_y)$, a map $\mathbf{Q}$ of the environment, a bounding box $\mathbf{b}^t = (b_x, b_y, b_w, b_h)$ of the actor of interest, an RGB image $\mathbf{I}^t$, semantic segmentation $\mathbf{S}^t$, or future egomotion $\mathbf{e}^{t\Rightarrow t+\Delta t}$. For future trajectory prediction, the state $\mathbf{y}$ is defined as the future trajectory (p_x, p_y) at $(t + \Delta t, ..., t + M\Delta t)$ and for future localization prediction as the future bounding box $\mathbf{b}$ at $t + \Delta t$.

We address the issue with the long tails of the data distribution in both bird's-eye view and egocentric settings. As backbone for these scenarios, we use the Trajectron++ [60] and FLN-RPN [47], respectively.

3.1. Bird's-Eye View - Trajectron++

Trajectron++ [60] is the state-of-the-art method for future trajectory prediction in bird's-eye view. It takes the dynamic actors, the static environment, and heterogeneous input data into account. Given the past trajectories $[(p_x^{t-h}, p_y^{t-h}), ..., (p_x^t, p_y^t)]$, and optionally a map $\mathbf{Q}$ of the scene, Trajectron++ builds a directed spatiotemporal graph for a scene based on its topology. It predicts future trajectories $\mathbf{y} = [(p_x^{t+\Delta t}, p_y^{t+\Delta t}), ..., (p_x^{t+M\Delta t}, p_y^{t+M\Delta t})]$. The nodes of the graph represent the actors, and the edges represent their interactions. The actors' histories are modeled by LSTMs, features of interacting actors are aggregated via point-wise summation, and GRUs are used to decode the future trajectories. The original architecture employs a cVAE to produce multiple future trajectories.

Since cVAEs require multiple runs of the decoder to obtain multiple predictions, we replace the cVAE by the multi-hypotheses networks trained with EWTA (Evolving

Winner-Takes-All) [48]. The EWTA loss for every sample i in the batch is defined as:

$$L_i^{\text{EWTA}} = \sum_{m=1}^{M} \sum_{k=1}^{K} w_{k,m} ||\mathbf{p}_k^{t+m\Delta t} - \hat{\mathbf{p}}^{t+m\Delta t}||, \quad (1)$$

$$w_{j,m} = \mathbb{1}_{j \in \underset{k}{\operatorname{argmin}} ||\mathbf{p}_k^{t+m\Delta t} - \hat{\mathbf{p}}^{t+m\Delta t}||}, \quad (2)$$

where K is the number of estimated hypotheses. $\mathbb{1}_{cond}$ is the indicator function that returns 1 if the condition $cond$ returns true and 0 otherwise. $\mathbf{p}_k^{t+m\Delta t}$ and $\hat{\mathbf{p}}^{t+m\Delta t}$ denote the kth predicted future state and the ground truth at future time step $(t + m\Delta t)$, respectively. The argmin returns the k hypotheses closest to the ground truth, where k gradually decreases from K to 1 during training. While all hypotheses are penalized in the beginning of the training, only the best one would be penalized at the end of the training. The Trajectron++ augmented by EWTA (Figure 3 (top)) produces multiple future trajectories in a single network pass and outperforms the standard Trajectron++, as we show in Section 6.5.

3.2. Egocentric View - FLN-RPN

FLN-RPN [47] is the state-of-the-art method for future localization prediction in the egocentric setting. FLN-RPN predicts the multimodal distribution of the future localization of an actor in three steps. First, it predicts where an actor is most likely to be in the current image (*Reachability Prior*). Second, it transfers the reachability prior from the current frame to the future frame using the future egomotion. Finally, past bounding boxes of the actor $\mathbf{b}$, images $\mathbf{I}$, semantic segmentations $\mathbf{S}$ at time steps $(t - h, ..., t)$, the future egomotion $\mathbf{e}^{t\Rightarrow t+\Delta t}$ and the predicted future reachability prior are given to the network to predict the future localization of the actor of interest. The prediction has the form of set of bounding boxes $\mathbf{b}$ at $t+\Delta t$. The two key components of FLN-RPN are the reachability prior, which helps overcoming mode collapse, and the EWTA loss function (Eq. 1 with $M = 1$ and $\mathbf{p}$ is replaced by $\mathbf{b}$) that can learn diverse multiple states of the future in a single forward pass. Figure 3 (bottom) illustrates the FLN-RPN framework.

3.3. Difficulty Ranking

Before we explore the effects of the distribution of the challenging scenarios in the feature space on the final prediction, we need to know how challenging each scenario is. Since manual labeling is not a viable option, we use a common and simple metric to measure the difficulty of cases: the displacement error made by the Kalman Filter [73, 47, 31] on this sample. Small errors indicate good approximation with linear extrapolation, whereas large errors indicate a challenging scenario that requires complex nonlinear prediction.
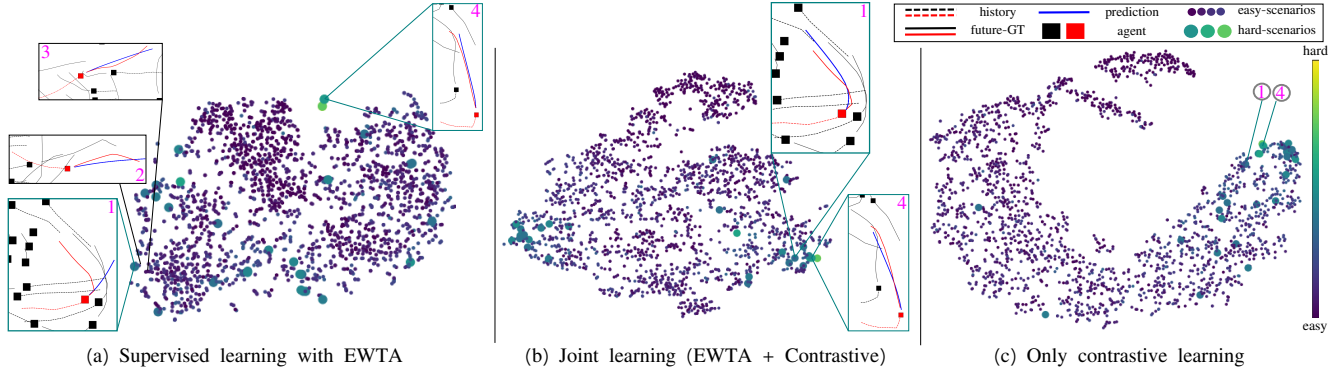
3

Figure 2. Feature space for the UNIV scene from ETH-UCY dataset using t-SNE [70]. **(a)** Training only with the supervised objective for future prediction (e.g, EWTA). Rare challenging scenarios (large green bright circles) are scattered among the frequent easy scenarios (small dark blue circles). We zoom into two challenging (1,4) and two easy scenarios (2,3). **(b)** Joint learning with the supervised (EWTA) and the contrastive loss. The challenging scenarios form two sub-spaces where they can share relevant features. The two challenging examples (1,4) are close and benefit each other, which improves their future predictions considerably (particularly for 1). **(c)** Only contrastive learning is used, where all challenging scenarios are strictly mapped to the same location. This destroys the task relevant cues and cannot provide any future prediction.

4. Why are Hard Cases Ignored by the Model?

To understand the cause of the problem with samples from the long tail of the data distribution, we visualized the feature embedding of the data from a network trained with a supervised future prediction objective and analyzed particular cases in detail. Figure 2 (left) shows the feature space for the UNIV scene in the ETH-UCY dataset projected to 2D with t-SNE [70]. Each dot is a sample from the scene mapped to the feature space by the network trained with EWTA loss (Eq. 1). Hard cases are sprinkled among the easy cases in the feature space without any structure. A closer look at a hard case (1) reveals that it shares some similarity with corresponding easy cases (2,3), which explains its position in the embedding, but the relevant cues, in which it is different from the easy cases, get ignored with normal training. The sample should rather be close to another challenging example (4) to capture the social interaction, where pedestrians walk in groups and follow other groups. We believe that challenging scenarios being alone in a manifold full of easy scenarios causes the network to ignore them and base its decisions on shortcuts learned from the dominant easy scenarios. The network does not get a chance to learn dedicated features to solve challenging cases by reusing some common cues among them (1,4), as long as they get mixed up with the easy cases. Indeed, the prediction for case (1) is quite wrong since it is similar to the prediction of cases (2,3), where social interaction is missing.

5. Reshaping the Embedding with Contrastive Learning

The analysis from the previous section triggers the idea to push hard samples away from the easy ones, such that

the relevant cues of similar hard samples get the chance to be no longer ignored during training. We implement this idea with an additional contrastive loss. Contrastive learning enforces certain training samples (positives) to be closer in the embedding to a sample (anchor) i than others (negatives). There are multiple ways to express this in a loss. The most popular is

$$
L_i^{\text{Contr}} = -\frac{1}{N_{\mathbf{po}_i} - 1} \sum_{j=1}^{N} \mathbb{1}_{i \neq j} \cdot \mathbb{1}_{j \in \mathbf{po}_i}
$$
$$
\cdot \log \frac{\exp(\mathbf{z}_i \cdot \mathbf{z}_j / \tau)}{\sum_{k=1}^{N} \mathbb{1}_{i \neq k} \cdot \exp(\mathbf{z}_i \cdot \mathbf{z}_k / \tau)} , \tag{3}
$$

where z is the learned feature vector at the bottleneck of the network (see Figure 3), $\mathbf{po}_i$ is the positive set of anchor i. $\mathbb{1}_{cond}$ is the indicator function that returns 1 if the condition $cond$ returns true and 0 otherwise. N is the total number of samples in the batch. $N_{\mathbf{po}_i}$ is the total number of positive samples for the anchor i. $\tau > 0$ is the temperature parameter. Positive samples are often defined as augmented versions of the same image [9] or samples belonging to the same class [35]. Negative samples, on the other hand, are other samples in the batch that do not satisfy the positive criterion by some safe margin. Since our goal is to distribute the features based on the difficulty, we define the positive set $\mathbf{po}_i$ as the set of samples j in the batch which has a difficulty score s_j satisfying $|s_i - s_j| < \theta_p$, where θ_p is a hyper-parameter defining the positivity threshold. Similarly, the negatives samples are defined as all samples with a difficulty score satisfying $|s_i - s_j| > \theta_n$. Note that we use different thresholds $\theta_p \neq \theta_n$ implying that many samples in the batch are neither positive nor negative. In order to minimize this loss, the network must maximize the nominator and minimize the denominator. Doing so, it learns to

map the positive samples close in the feature space and the negative ones apart. The result of training with such a loss is shown in Figure 2 (right).

While having the hard cases being pushed together is good for them to share relevant cues and learn prediction models for less common scenarios, there is much diversity among hard cases, and not all of them should be pushed to the same space. In particular, we should not destroy cues shared with the easy examples, which are necessary for the network to solve the actual task. The contrastive loss alone can not predict the future state. To this end, we jointly optimize for the supervised future prediction loss L^{EWTA} and the self-supervised contrastive loss L^{Contr} as:

$$L_i = L_i^{\text{EWTA}} + \lambda \cdot L_i^{\text{Contr}}, \qquad (4)$$

where λ controls the importance of the contrastive loss, hence the strength of the attraction that pulls hard cases together.

Figure 2 (middle) shows the effect of this combination. Cases (1) and (4) fall into the same sub-space resulting in a much better prediction for (1). Other hard cases rather stay with similar easy samples as they have no other hard cases to share information with.

Due to its simplicity, this difficulty-based contrastive learning can be added to any existing method as long as the difficulty can be defined explicitly on the training set.

6. Experiments

6.1. Datasets

The **ETH-UCY** dataset is the combination of the ETH [55] and the UCY [38] pedestrian datasets. Both include videos from bird's-eye view of the pedestrians, where the trajectories are manually annotated. The challenges in these datasets are the frequent interactions between pedestrians, as the scenes are very crowded, and the lack of visual information due to the viewpoint, i.e, the actors are small and uninformative. We present 5-fold cross-validation results on the five scenes of the dataset.

nuScenes [4] is a large autonomous driving dataset with 1000 scenes, where each is 20 seconds long. It provides HD semantic maps with 11 different layers and accurate bounding box annotations in time. It provides scenarios from bird's-eye view and egocentric view, and we experiment with each of them.

Waymo [68] is the most recent autonomous driving dataset with 1000 scenes, where each is 20 seconds long. We use the validation part of the dataset (202 scenes) to show zero-shot transfer of our approach in egocentric view (i.e, without retraining the model).
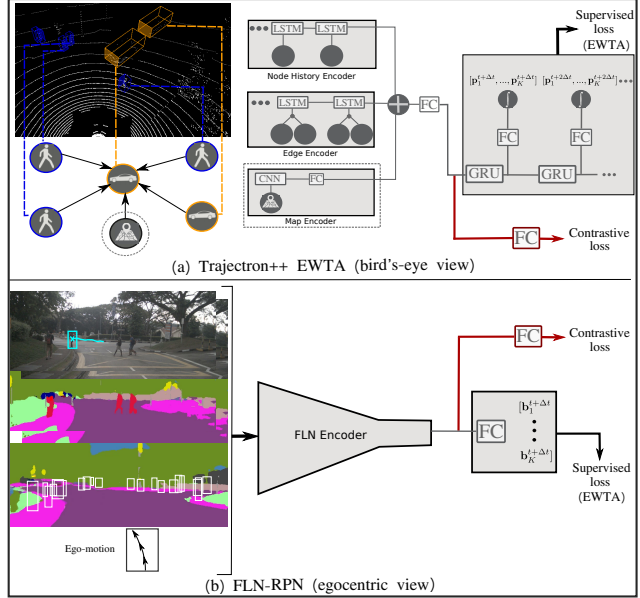

(a) Trajectron++ EWTA (bird's-eye view)

(b) FLN-RPN (egocentric view)

Figure 3. Schematic that shows how we flexibly integrate the contrastive loss (red) in existing future prediction frameworks. (a) Bird's-eye view (Trajectron++[60]). (b) Egocentric view (FLN-RPN[47]). Independent of the contrastive loss, we modified Trajectron++ by replacing the cVAE with the EWTA [48] framework to better capture the multimodality of the predicted future and for faster inference time. The map encoder (dashed gray) is optional and only used for the nuScenes dataset.

6.2. Evaluation Metrics

min-ADE is the minimum average displacement error. It computes the mean L_2 distance between all predicted trajectories and the ground truth and reports the error of the closest one. This is sometimes also referred to as *oracle* (or *best-of-many*), since the selection of the minimum error depends on the ground truth.

min-FDE is the minimum final displacement error. It computes the L_2 distance between the final locations of the predicted trajectories and the ground truth at the end of the predicted time horizon ($t + M\Delta t$) and, like min-ADE, reports the minimum.

6.3. Training Details

In our experiments for bird's-eye view, we followed the original training schedule for Trajectron++ [60]. We trained the Trajectron++ (EWTA) with batch size 256 for 100 epochs in every EWTA stage ($k = K, ..., 1$) for ETH-UCY and for 5 epochs in every EWTA stage for nuScenes. For the experiments in egocentric view, we used ResNet34 [28] as the encoder of FLN-RPN [47] and trained with batch size of 32. Following [60, 47], we set M to 12, 6, 1 and Δt to 0.4, 0.5, 3.0 for ETH-UCY, nuScenes (bird's-eye view) and nuScenes/Waymo (egocentric view), respectively. The remaining design choices were kept as in the original papers

[60, 47]. For our joint optimization, λ was chosen based on the validation set as 1, 50, 150 for nuScenes (bird's-eye view), ETH-UCY, and nuScenes (egocentric view), respectively. We used the recommended value of 0.5 for τ [9]. θ_p and θ_n were set such that the ratio of positives and negatives over the batch size are 10% and 40% for Trajectron++ and 33% and 33% for FLN-RPN, respectively. z had the dimensions of 232 and 256 for Trajectron++ and FLN-RPN, respectively. A study about the effect of the hyper-parameter λ is presented in the supplemental material.

6.4. Baselines

Bird's-eye view (ETH-UCY). We selected a set of recent methods addressing the future trajectory prediction: Graph-based approaches: RSBG[67], S-STGCNN[53], and Trajectron++[60] (referred as Traj++); transformer-based approach: STAR[74]; multi-stage networks: TPNet [21] and PECNet [51].

Bird's-eye view (nuScenes). We compare against a set of baselines including deterministic LSTM-based approaches: S-LSTM [1], CSP [16], and CAR-Net [59]; multimodal graph-based approaches: SpAGNN [6] and Trajectron++ [60].

Egocentric view. We compare against the multimodal state-of-the-art FLN-RPN [47].

Moreover, for all settings and datasets, we implemented the common approaches for imbalanced data: resampling [61], reweighting using the inverse class frequency [29], and reweighting using the effective number of samples [13]. We also adapt sophisticated long-tail classification methods[5, 41] to the considered task by defining classes based on the discretization of Kalman filter scores. Then, the network is jointly trained on the regression loss and the considered classification loss (more details are provided in the supplementary). Notice that recent methods: cRT, τ-norm and LWS introduced by Kang et al. [32] can not be adapted to regression tasks since they do not affect the feature extractor and fully rely on post-processing the classifier which is not needed at test time in our scenario.

6.5. Results & Discussion

To show the validity of the proposed approach, we selected strong baselines and state-of-the-art methods for comparison. Tables 1, 2 and 3 summarize our results on the four different datasets. Since we are interested in improving the quality of the predictions of the rare cases, we report min-ADE and min-FDE for all samples, as well as the top 1-3% challenging cases.

EWTA vs cVAE. Tables 1 and 2 show that our base method, where we use the Trajectron++ as the backbone with the EWTA objective, clearly outperforms the previous state-of-the-art Trajectron++. This shows that EWTA-based sampling for possible future trajectories works better than

	All	Top 3%	Top 2%	Top 1%
RSBG [67]	0.48/0.99	-/-	-/-	-/-
Reciprocal [66]	0.44/0.90	-/-	-/-	-/-
TPNet [21]	0.42/0.90	-/-	-/-	-/-
S-STGCNN [53]	0.44/0.75	-/-	-/-	-/-
STAR [74]	0.26/0.53	-/-	-/-	-/-
PEC-NET [51]	0.29/0.48	-/-	-/-	-/-
Traj++ [60]	0.21/0.41	0.65/1.42	0.71/1.51	0.58/1.23
Traj++ EWTA (ours)	0.16/0.32	0.47/1.07	0.51/1.13	0.42/0.87
+ LDAM [5]	0.17/0.33	0.47/1.04	0.50/1.08	0.42/0.83
+ LDAM-DRW [5]	0.17/0.33	0.47/1.04	0.51/1.08	0.43/0.83
+ BAGS [41]	0.17/0.32	0.48/1.08	0.51/1.10	0.42/0.85
+ contrastive (ours)	**0.16/0.32**	**0.46/1.03**	**0.48/1.03**	**0.38/0.71**

Table 1. Average error on the **ETH-UCY benchmark** over all test samples and over the 1-3% most challenging scenarios in the format of (min-ADE/min-FDE). Joint learning with the contrastive loss yields large improvements on the challenging scenarios while not harming the overall average accuracy.

cVAE-based sampling.

	All	Top 3%	Top 2%	Top 1%
S-LSTM [1]	– /1.61	– / –	– / –	– / –
CSP [16]	– /1.50	– / –	– / –	– / –
CAR-Net [59]	– /1.35	– / –	– / –	– / –
SpAGNN [6]	– /1.23	– / –	– / –	– / –
Traj++ [60]	0.22/0.39	0.55/0.98	0.60/1.04	0.72/1.21
Traj++ EWTA (ours)	0.19/0.32	0.48/0.88	0.50/0.88	0.59/1.02
+ LDAM [5]	0.18/0.32	0.48/0.88	0.51/0.93	0.60/1.10
+ LDAM-DRW [5]	0.18/0.32	0.50/0.93	0.52/0.96	0.63/1.14
+ BAGS [41]	0.18/0.31	0.48/0.88	0.51/0.94	0.61/1.11
+ contrastive (ours)	**0.18/0.30**	**0.44/0.73**	**0.46/0.72**	**0.54/0.85**

Table 2. Average error on the **nuScenes dataset (bird's eye view)** over all test samples and over the 1-3% most challenging scenarios in the format of (min-ADE/min-FDE). Joint learning with the contrastive loss yields large improvements on the challenging scenarios and even improves the overall average accuracy a little.

Large improvements on the challenging cases. Results on all datasets show that our approach yields large improvements on the challenging cases (particularly for the top 1%) while maintaining the overall average error. In particular, on the most challenging cases (top 1%), our approach improves by 18%, 17%, 23% and 12% on the ETH-UCY, nuScenes (bird's eye-view), nuScenes (egocentric view) and Waymo open dataset, respectively. The challenging training samples, as hypothesized, help each other when they are in proximity in the feature space. Notably, the studied datasets differ in their input modalities (additional semantic maps for nuScenes), viewpoint (bird's-eye vs egocentric views), and prediction output (2D points in bird's-eye view while bounding boxes for egocentric view). This indicates that the approach is agnostic to different input modalities and generalizes well.

Comparison to long-tail classification baselines. Tables 1, 2 and 3 show a comparison against recent methods addressing the long-tail problem in classification. Our method based on the contrastive loss outperforms all these techniques on all metrics.

6

	nuScenes Egocentric View				Waymo Egocentric View			
	All	Top 3%	Top 2%	Top 1%	All	Top 3%	Top 2%	Top 1%
FLN-RPN [47]	7.10	29.98	31.13	36.16	**6.39**	24.87	25.49	27.32
+ LDAM [5]	8.04	25.23	26.02	31.13	7.61	23.00	23.09	25.05
+ LDAM-DRW [5]	8.01	26.63	27.85	34.58	8.05	25.23	25.98	29.32
+ BAGS [41]	7.28	29.54	30.38	35.74	6.67	24.45	24.88	26.66
+ contrastive (ours)	**7.04**	**25.05**	**25.26**	**27.49**	6.49	**22.36**	**22.72**	**24.09**

Table 3. Results on **egocentric datasets (nuScenes and Waymo)**. We show the min-FDE over all scenarios and over the top 1-3% challenging scenarios. Our approach yields an improvement on the challenging scenarios while maintaining the performance on average.

Zero-shot transfer. Results on the Waymo dataset (Tab. 3) show promising zero-shot transfer to unseen dataset, where models were trained on the nuScenes dataset and tested on the validation split of the Waymo dataset.

Avoids bias. In Table 4 we compare our method against the common approaches for imbalanced data: resampling and reweighting. We report across all datasets the performance over all samples and over the most challenging samples (top 1%). As expected, these baselines tend to bias the challenging cases. Hence, the average performance drops significantly (66%, 16%, 44% and 64% for ETH-UCY, nuScenes bird's, nuScenes egocentric and Waymo). Our method, on the other hand, maintains the average performance over all samples. Detailed results on all metrics and difficulties are provided in the supplementary.

	ETH-UCY	nuScenes-B	nuScenes-E	Waymo
	All/Top 1%	All/Top 1%	All/Top 1%	All/Top 1%
Baseline	0.32/0.87	0.32/1.02	7.10/36.16	**6.39**/27.32
+ resample [61]	0.53/1.22	0.37/1.33	10.20/21.62	10.48/19.69
+ reweight [29]	0.56/0.76	0.58/1.67	14.47/16.20	14.00/**16.44**
+ reweight [13]	0.56/0.78	0.60/1.71	16.54/**15.46**	17.43/18.79
+ contrastive	**0.32/0.71**	**0.30/0.85**	**7.04**/27.49	6.49/24.09

Table 4. Comparison to the common resampling/reweighting techniques on the four datasets. For each method, we show the min-FDE over all samples and over top 1% challenging samples. Our method yields large improvements on the challenging ones while maintaining the average. This is in contrast to the reweighting/resampling baselines, which lead to much worse performance on average. Baseline indicates Traj++ EWTA for bird's eye view and FLN-RPN for egocentric view.

6.6. Qualitative Results

In Figure 4 (a), we show three challenging examples from ETH-UCY dataset. In all the cases, the future trajectory of the pedestrian (red) is not trivial, and the network must model the interaction between pedestrians to generate a plausible future trajectory. Our approach (blue) generates trajectories that are much closer to the ground truth than Trajectron++ EWTA (cyan). In Figure 4 (b), we show three challenging examples for vehicles from the nuScenes dataset (bird's-eye view). In these examples, the vehicle changes direction, which requires interpretation of the maps. Our approach succeeds on these examples, whereas Trajectron++ EWTA misses these cues and predicts the simple continuation of the trajectory.

Figure 5 shows four different examples from the egocentric setting. Figure 5 (a) shows a child crossing the street in front of the vehicle. Figure 5 (b) shows a vehicle that will turn right to go down the street, which is rarely encountered. Figure 5 (c) shows an example that is difficult because of the uncommon egomotion of the car moving to the opposite lane to overtake the bus. Figure 5 (d) shows a vehicle that turns right to exit the round-about. In all these examples, our approach makes predictions close to the ground truth (both in scale and location), whereas the baseline fails.

Limitations and failure cases. We also analyzed the limits of our approach to identify room for further improvements. We found that some challenging cases continue to stay in a manifold for easy cases because of missing similarity to other hard cases Figure 6 (b), or easy cases moved wrongly to a manifold of challenging cases Figure 6 (c). Consequently, our approach yields wrong prediction. We also found that our method, like other methods, cannot model unexpected behavior, such as suddenly stopping and turning in the opposite direction Figure 6 (a). We also provide the feature embedding before and after application of our approach for all datasets in the supplementary.

7. Conclusions

We addressed the long-tailed data distributions by acting on the feature embedding. We showed that pulling the rare challenging samples together in the feature embedding via contrastive learning helps improve their final predictions while preserving the performance over the whole dataset. We validated our approach qualitatively and quantitatively on four different datasets, two different viewpoints and different combinations of input and output modalities. The proposed loss can be integrated easily into existing approaches to improve their performance on critical challenging cases. We hypothesize that the concept is generic and could be integrated into other regression tasks with an unbalanced sample distribution, as long as there is a way to identify the underrepresented samples during training.

8. Acknowledgments

	other agent	⬛⬛⬛ history	Traj++ EWTA
	agent-of-interest	—— future-GT	—— (Traj++ EWTA)+Contrastive

(a) Three challenging pedestrian examples from ETH-UCY dataset   (b) Three challenging vehicle examples from nuScenes dataset (bird's-eye view)
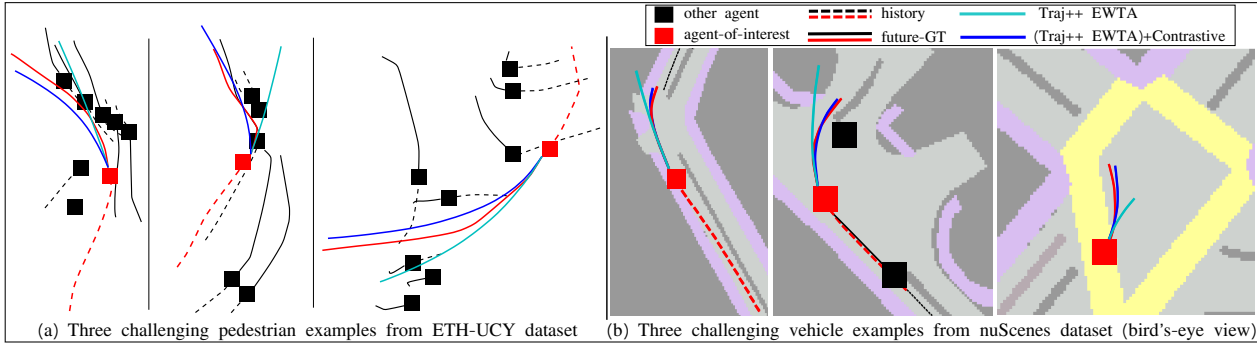
Figure 4. Qualitative challenging examples for pedestrians from ETH-UCY dataset (a) and vehicles from nuScenes bird's-eye view dataset (b). Note how our approach outperforms the SOTA (Trajectron++ EWTA) by generating a future trajectory closer to the ground truth. We visualize the best hypothesis for each method. For the examples from nuScenes (b), we show the underlying map on which the method need to reason about.

(a) A running child crossing the street   (b) A vehicle turning right to go down the street

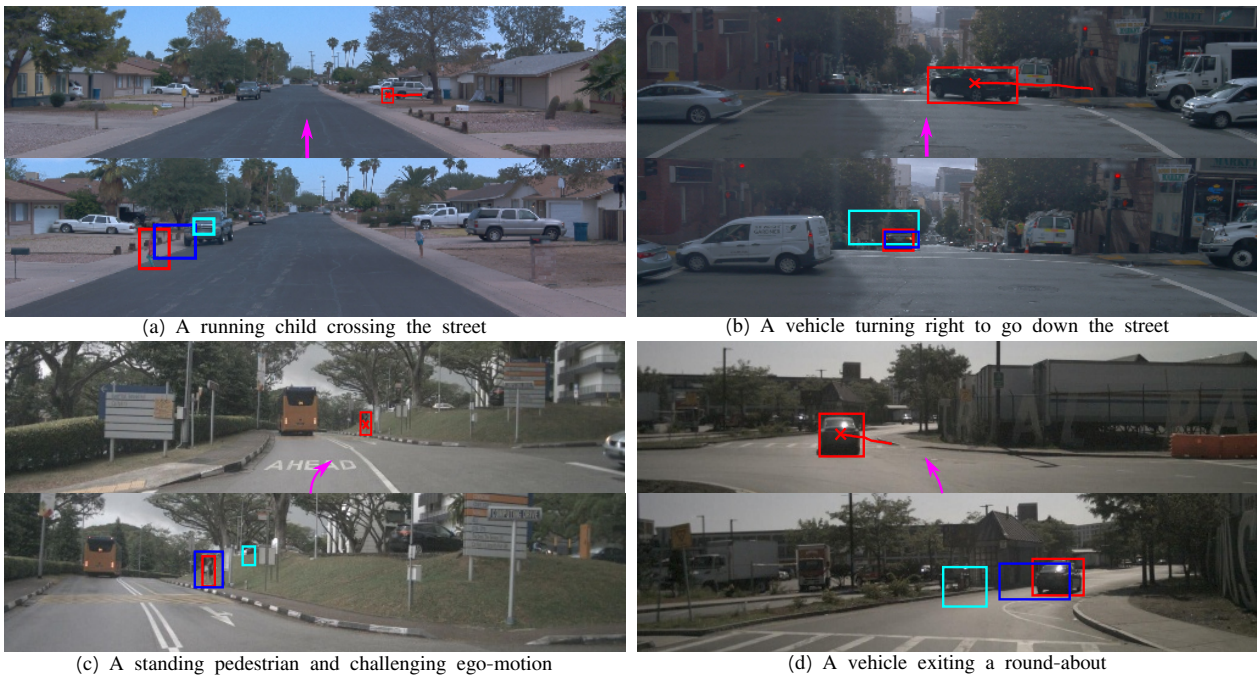(c) A standing pedestrian and challenging ego-motion   (d) A vehicle exiting a round-about

Figure 5. Qualitative challenging examples from Waymo open dataset (a-b) and nuScenes egocentric view (c-d). For each example, we show both the last observed image (top) and the future image (bottom) along with the predictions (FLN-RPN [47] and Ours) and the ground truth. We visualize the best hypothesis for each method. The future egomotion is also shown as arrow indicating the motion of the ego-car.

(a) A very challenging example of an unusual pedestrian behavior   (b) A challenging example stayed within its manifold for easy cases   (c) A less challenging example moved to a manifold of challenging cases
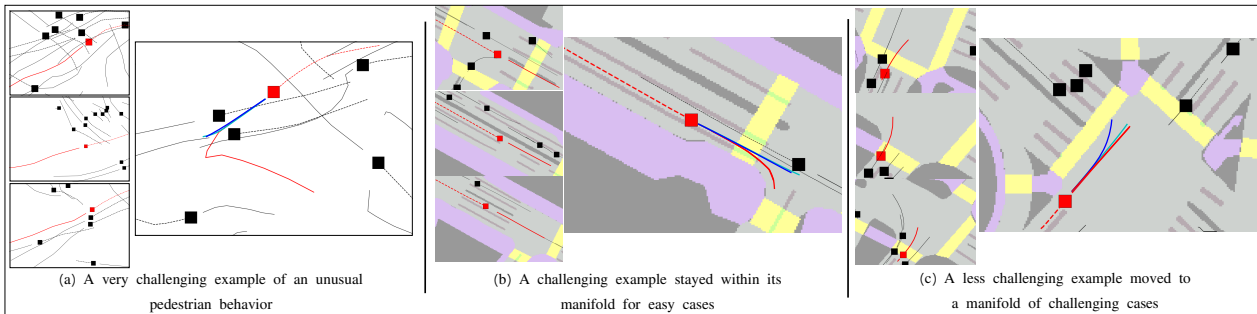
Figure 6. Three examples for different categories of failures from our method. Each example is shown together with three other examples on its left from its manifold resulting from our approach. (a) An example of a pedestrian from ETH-UCY dataset who unexpectedly decided to turn back and go left. Such an unexpected future behavior is very hard to model. (b) A vehicle from nuScenes (top view) dataset that decided to turn right and our approach in unable to change its manifold. (c) An example of a less challenging example from nuScenes (bird's-eye view) dataset which our approach mistakenly moves to a challenging manifold.

8

References

[1] 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. 2, 6

[2] Federico Bartoli, Giuseppe Lisanti, Lamberto Ballan, and Alberto Del Bimbo. Context-aware trajectory prediction. In *ICPR*, 2018. 2

[3] A. Bhattacharyya, M. Fritz, and B. Schiele. Long-term on-board prediction of people in traffic scenes under uncertainty. In *CVPR*, 2018. 2

[4] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *CVPR*, 2020. 5

[5] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Aréchiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. In *NeurIPS*, 2019. 2, 6, 7, 12

[6] Sergio Casas, Cole Gulino, Renjie Liao, and Raquel Urtasun. Spagnn: Spatially-aware graph neural networks for relational behavior forecasting from sensor data. In *ICRA*, 2020. 2, 6

[7] Rohan Chandra, Uttaran Bhattacharya, Aniket Bera, and Dinesh Manocha. Traphic: Trajectory prediction in dense and heterogeneous traffic using weighted interactions. In *CVPR*, 2019. 2

[8] Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 2002. 2

[9] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020. 2, 4, 6

[10] Chiho Choi and Behzad Dariush. Looking to relations for future trajectory forecast. In *ICCV*, 2019. 2

[11] Chiho Choi, Abhishek Patil, and Srikanth Malla. DROGON: A causal reasoning framework for future trajectory forecast. *CoRR*, abs/1908.00024, 2019. 2

[12] Henggang Cui, Thi Nguyen, Fang-Chieh Chou, Tsung-Han Lin, Jeff Schneider, David Bradley, and Nemanja Djuric. Deep kinematic models for physically realistic prediction of vehicle trajectories. *CoRR*, abs/1908.00219, 2019. 2

[13] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge J. Belongie. Class-balanced loss based on effective number of samples. In *CVPR*, 2019. 2, 6, 7, 13

[14] Yin Cui, Yang Song, Chen Sun, Andrew Howard, and Serge J. Belongie. Large scale fine-grained categorization and domain-specific transfer learning. In *CVPR*, 2018. 2

[15] N. Deo, A. Rangesh, and M. M. Trivedi. How would surround vehicles move? a unified framework for maneuver classification and motion prediction. *T-IV*, 2018. 2

[16] Nachiket Deo and Mohan M. Trivedi. Multi-modal trajectory prediction of surrounding vehicles with maneuver based lstms. In *Intelligent Vehicles Symposium*, 2018. 2, 6

[17] Nemanja Djuric, Vladan Radosavljevic, Henggang Cui, Thi Nguyen, Fang-Chieh Chou, Tsung-Han Lin, Nitin Singh, and Jeff Schneider. Uncertainty-aware short-term motion prediction of traffic actors for autonomous driving. In *WACV*, 2020. 2

[18] Qi Dong, Shaogang Gong, and Xiatian Zhu. Imbalanced deep learning by minority class incremental rectification. *IEEE TPAMI*, 2019. 2

[19] Alexey Dosovitskiy, Philipp Fischer, Jost Tobias Springenberg, Martin A. Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with exemplar convolutional neural networks. *IEEE TPAMI*, 2016. 2

[20] Chris Drummond and Robert Holte. C4.5, class imbalance, and cost sensitivity: Why under-sampling beats oversampling. *ICML Workshop*, 2003. 2

[21] Liangji Fang, Qinhong Jiang, Jianping Shi, and Bolei Zhou. Tpnet: Trajectory proposal network for motion prediction. In *CVPR*, 2020. 6

[22] Agrim Gupta, Justin Johnson, Li Fei-Fei, Silvio Savarese, and Alexandre Alahi. Social gan: Socially acceptable trajectories with generative adversarial networks. In *CVPR*, 2018. 2

[23] Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In Yee Whye Teh and D. Mike Titterington, editors, *AISTATS*, 2010. 2

[24] Hui Han, Wenyuan Wang, and Binghuan Mao. Borderline-smote: A new over-sampling method in imbalanced data sets learning. In *ICIC*, 2005. 2

[25] Munawar Hayat, Salman Khan, Syed Waqas Zamir, Jianbing Shen, and Ling Shao. Gaussian affinity for max-margin class imbalanced learning. In *ICCV*, 2019. 2

[26] Haibo He, Yang Bai, Edwardo A. Garcia, and Shutao Li. ADASYN: adaptive synthetic sampling approach for imbalanced learning. In *IJCNN*, 2008. 2

[27] H. He and E. A. Garcia. Learning from imbalanced data. *IEEE TKDE*, 2009. 2

[28] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 5

[29] Chen Huang, Yining Li, Chen Change Loy, and Xiaoou Tang. Learning deep representation for imbalanced classification. In *CVPR*, 2016. 2, 6, 7, 13

[30] Muhammad Abdullah Jamal, Matthew Brown, Ming-Hsuan Yang, Liqiang Wang, and Boqing Gong. Rethinking class-balanced methods for long-tailed visual recognition from a domain adaptation perspective. In *CVPR*, 2020. 2

[31] R. E. Kalman. A new approach to linear filtering and prediction problems. *ASME Journal of Basic Engineering*, 1960. 1, 3

[32] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. In *ICLR*, 2020. 6

[33] Salman H. Khan, Munawar Hayat, Mohammed Bennamoun, Ferdous Ahmed Sohel, and Roberto Togneri. Cost-sensitive learning of deep feature representations from imbalanced data. *IEEE Trans. Neural Networks Learn. Syst.*, 2018. 2

[34] Salman H. Khan, Munawar Hayat, Syed Waqas Zamir, Jianbing Shen, and Ling Shao. Striking the right balance with uncertainty. In *CVPR*, 2019. 2

[35] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *CoRR*, abs/2004.11362, 2020. 4

[36] Jaehyung Kim, Jongheon Jeong, and Jinwoo Shin. M2m: Imbalanced classification via major-to-minor translation. In *CVPR*, 2020. 2

[37] Vineet Kosaraju, Amir Sadeghian, Roberto Martín-Martín, Ian D. Reid, Hamid Rezatofighi, and Silvio Savarese. Social-bigat: Multimodal trajectory forecasting using bicycle-gan and graph attention networks. In *NeurIPS*, 2019. 2

[38] Laura Leal-Taixé, Michele Fenzi, Alina Kuznetsova, Bodo Rosenhahn, and Silvio Savarese. Learning an image-based motion context for multiple people tracking. In *CVPR*, 2014. 5

[39] Namhoon Lee, Wongun Choi, Paul Vernaza, Christopher B Choy, Philip HS Torr, and Manmohan Chandraker. Desire: Distant future prediction in dynamic scenes with interacting agents. In *CVPR*, 2017. 2

[40] Jiachen Li, Fan Yang, Masayoshi Tomizuka, and Chiho Choi. Evolvegraph: Multi-agent trajectory prediction with dynamic relational reasoning. In *NeurIPS*, 2020. 2

[41] Yu Li, Tao Wang, Bingyi Kang, Sheng Tang, Chunfeng Wang, Jintao Li, and Jiashi Feng. Overcoming classifier imbalance for long-tail object detection with balanced group softmax. In *CVPR*, 2020. 2, 6, 7, 12

[42] Junwei Liang, Lu Jiang, and Alexander G. Hauptmann. Simaug: Learning robust representations from 3d simulation for pedestrian trajectory prediction in unseen cameras. In *ECCV*, 2020. 2

[43] Junwei Liang, Lu Jiang, Kevin Murphy, Ting Yu, and Alexander G. Hauptmann. The garden of forking paths: Towards multi-future trajectory prediction. In *CVPR*, 2020. 2

[44] Ming Liang, Bin Yang, Rui Hu, Yun Chen, Renjie Liao, Song Feng, and Raquel Urtasun. Learning lane graph representations for motion forecasting. In *ECCV*, 2020. 2

[45] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, 2017. 2

[46] Yuexin Ma, Xinge Zhu, Sibo Zhang, Ruigang Yang, Wenping Wang, and Dinesh Manocha. Trafficpredict: Trajectory prediction for heterogeneous traffic-agents. In *AAAI*, 2019. 2

[47] O. Makansi, Ö. Çiçek, K. Buchicchio, and T. Brox. Multimodal future localization and emergence prediction for objects in egocentric view with a reachability prior. In *CVPR*, 2020. 1, 2, 3, 5, 6, 7, 8, 13, 15

[48] Osama Makansi, Eddy Ilg, Ozgun Cicek, and Thomas Brox. Overcoming limitations of mixture density networks: A sampling and fitting framework for multimodal future prediction. In *CVPR*, 2019. 1, 2, 3, 5

[49] Srikanth Malla and Chiho Choi. NEMO: future object localization using noisy ego priors. *CoRR*, abs/1909.08150, 2019. 2

[50] Srikanth Malla, Behzad Dariush, and Chiho Choi. TITAN: future forecast using action priors. In *CVPR*, 2020. 2

[51] Karttikeya Mangalam, Harshayu Girase, Shreyas Agarwal, Kuan-Hui Lee, Ehsan Adeli, Jitendra Malik, and Adrien Gaidon. It is not the journey but the destination: Endpoint conditioned trajectory prediction. In *ECCV*, 2020. 2, 6

[52] Huynh Manh and Gita Alaghband. Scene-lstm: A model for human trajectory prediction. *CoRR*, abs/1808.04018, 2018. 2

[53] Abduallah Mohamed, Kun Qian, Mohamed Elhoseiny, and Christian Claudel. Social-stgcnn: A social spatio-temporal graph convolutional neural network for human trajectory prediction. In *CVPR*, 2020. 2, 6

[54] Sankha Subhra Mullick, Shounak Datta, and Swagatam Das. Generative adversarial minority oversampling. In *ICCV*, 2019. 2

[55] Stefano Pellegrini, Andreas Ess, Konrad Schindler, and Luc Van Gool. You'll never walk alone: Modeling social behavior for multi-target tracking. In *ICCV*, 2009. 5

[56] Junran Peng, Xingyuan Bu, Ming Sun, Zhaoxiang Zhang, Tieniu Tan, and Junjie Yan. Large-scale object detection in the wild from imbalanced multi-labels. In *CVPR*, 2020. 2

[57] Nicholas Rhinehart, Rowan McAllister, Kris Kitani, and Sergey Levine. Precog: Prediction conditioned on goals in visual multi-agent settings. In *ICCV*, 2019. 2

[58] Amir Sadeghian, Vineet Kosaraju, Ali Sadeghian, Noriaki Hirose, Hamid Rezatofighi, and Silvio Savarese. Sophie: An attentive gan for predicting paths compliant to social and physical constraints. In *CVPR*, 2019. 2

[59] Amir Sadeghian, Ferdinand Legros, Maxime Voisin, Ricky Vesel, Alexandre Alahi, and Silvio Savarese. Car-net: Clairvoyant attentive recurrent network. In *ECCV*, 2018. 2, 6

[60] Tim Salzmann, Boris Ivanovic, Punarjay Chakravarty, and Marco Pavone. Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data. In *ECCV*, 2020. 1, 2, 3, 5, 6

[61] Li Shen, Zhouchen Lin, and Qingming Huang. Relay backpropagation for effective learning of deep convolutional neural networks. In *ECCV*, 2016. 2, 6, 7, 13

[62] Abhinav Shrivastava, Abhinav Gupta, and Ross B. Girshick. Training region-based object detectors with online hard example mining. In *CVPR*, 2016. 2

[63] Shashank Srikanth, Junaid Ahmed Ansari, Karnik Ram R., Sarthak Sharma, J. Krishna Murthy, and K. Madhava Krishna. INFER: intermediate representations for future prediction. In *IROS*, 2019. 2

[64] Olly Styles, Tanaya Guha, and Victor Sanchez. Multiple object forecasting: Predicting future object locations in diverse environments. In *WACV*, 2020. 2

[65] O. Styles, A. Ross, and V. Sanchez. Forecasting pedestrian trajectory with machine-annotated training data. In *IV*, 2019. 2

[66] Hao Sun, Zhiqun Zhao, and Zhihai He. Reciprocal learning networks for human trajectory prediction. In *CVPR*, 2020. 2, 6

[67] Jianhua Sun, Qinhong Jiang, and Cewu Lu. Recursive social behavior graph for trajectory prediction. In *CVPR*, 2020. 2, 6

[68] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine, V. Vasudevan, W. Han, J. Ngiam, H. Zhao, A. Timofeev, S. Ettinger, M. Krivokon, A. Gao, A. Joshi, Y. Zhang, J. Shlens, Z. Chen, and D. Anguelov. Scalability in perception for autonomous driving: Waymo open dataset. In *CVPR*, 2020. 5

[69] Aäron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *CoRR*, abs/1807.03748, 2018. 2

[70] L.J.P. van der Maaten and G.E. Hinton. Visualizing high-dimensional data using t-sne. *Journal of Machine Learning Research*, 2008. 4, 12, 14

[71] Y. Xu, Z. Piao, and S. Gao. Encoding crowd interaction with deep neural network for pedestrian trajectory prediction. In *CVPR*, 2018. 2

[72] Yuzhe Yang and Zhi Xu. Rethinking the value of labels for improving class-imbalanced learning. In *NeurIPS*, 2020. 3

[73] Y. Yao, M. Xu, C. Choi, D. J. Crandall, E. M. Atkins, and B. Dariush. Egocentric vision-based future vehicle localization for intelligent driving assistance systems. In *ICRA*, 2019. 2, 3

[74] Cunjun Yu, Xiao Ma, Jiawei Ren, Haiyu Zhao, and Shuai Yi. Spatio-temporal graph transformer networks for pedestrian trajectory prediction. In *ECCV*, 2020. 2, 6

[75] Pu Zhang, Wanli Ouyang, Pengfei Zhang, Jianru Xue, and Nanning Zheng. Sr-lstm: State refinement for lstm towards pedestrian trajectory prediction. In *CVPR*, 2019. 2

[76] X. Zhang, Z. Fang, Y. Wen, Z. Li, and Y. Qiao. Range loss for deep face recognition with long-tailed training data. In *ICCV*, 2017. 2

[77] Tianyang Zhao, Yifei Xu, Mathew Monfort, Wongun Choi, Chris Baker, Yibiao Zhao, Yizhou Wang, and Ying Nian Wu. Multi-agent tensor fusion for contextual trajectory prediction. In *CVPR*, 2019. 2

Supplementary Material for:
On Exposing the Challenging Long Tail in Future Prediction of Traffic Actors

1. Visualization Plots

Figure 7 and 8 show the comparison between our method and different baselines where each circle indicates the performance of one method. These figures illustrate better the improvements gained by our method (dashed arrows).

2. Feature Space Visualization

Figure 9 shows the projection of the feature space using tSNE [70] on three different datasets with different input modalities and views. For each dataset, we show the feature space embedding without our joint optimization (i.e, only the supervised loss) and with our joint optimization (i.e, additionally utilizing the contrastive loss). Note how our approach reshapes the feature space by pushing the challenging scenarios to be closer so that they can benefit each other as also shown in our quantitative results.

3. Effect of the Strength of the Contrastive Loss

In Table 5 we show a study for the importance of the contrastive loss (λ) used in our approach (Eq. (4)). Using a small factor leads to small improvements on the challenging scenarios as the force of reshaping the feature space is rather weak. On the other hand, using a very large factor yields worse results as the network focuses more on reshaping the feature space and ignores the important cues for the actual task which are learned from the supervised loss. Note that this study is used only to show the effect of the weight of the contrastive loss. In our main results, we use the validation set to select the best value for λ.

4. More Qualitative Results

We provide more qualitative results from our approach in Figure 10, Figure 11 and Figure 12 for the ETH-UYC, nuScenes (bird's-eye view) and nuScenes/Waymo (egocentric view) datasets, respectively.

5. Detailed Quantitative Results

Table 6 show a detailed comparison between our method and the resampling/reweighting baselines across all datasets on all metrics and difficulties. This support our findings that

	ETH-UCY (AVG)			
	All	Top 3%	Top 2%	Top 1%
Traj++ EWTA (ours)	**0.16/0.32**	0.47/1.07	0.51/1.13	0.42/0.87
+ contrastive ($\lambda = 20$)	0.17/0.33	0.47/1.04	0.50/1.07	0.43/0.84
+ contrastive ($\lambda = 50$)	**0.16/0.32**	**0.46/1.03**	**0.48/1.03**	**0.38/0.71**
+ contrastive ($\lambda = 100$)	0.17/0.32	0.48/1.04	0.52/1.10	0.50/0.97

Table 5. Study of the hyper-parameter λ on the ETH-UCY dataset. While small λ yields small improvement on the challenging scenarios, large λ yields larger errors on the challenging scenarios.

these baselines tend to bias the challenging cases (overfitting) while our approach maintain the average performance and improves largely on the challenging cases.

6. Baselines Implementation Details

In order to use state-of-the-art methods for long-tail classification, we map the regression task to a classification task by assigning classes to training samples based on the error of the Kalman filter. In particular, we group the errors into bins and assign the same class to all samples in each bin. To alleviate the issue of having classes with only one sample, we group all samples with a score greater than a specific threshold into the same bin. This yields 13, 36, 331 classes for ETH-UCY, nuScenes bird's eye view and nuScenes egocentric view, respectively. For all baselines (including our method), we use the same joint training scheme where two heads (classification and regression) are trained on top of the feature embedding. For the LDAM baseline [5], we experiment with different scaling factors and use the best setting $s = 1$. Following BAGS [41], we split the classes into 4 homogeneous groups to ensure that all classes from the same group have roughly the same number of items and use a sampling ration of 8 to ensure that all groups contribute to the mini-batch during training.
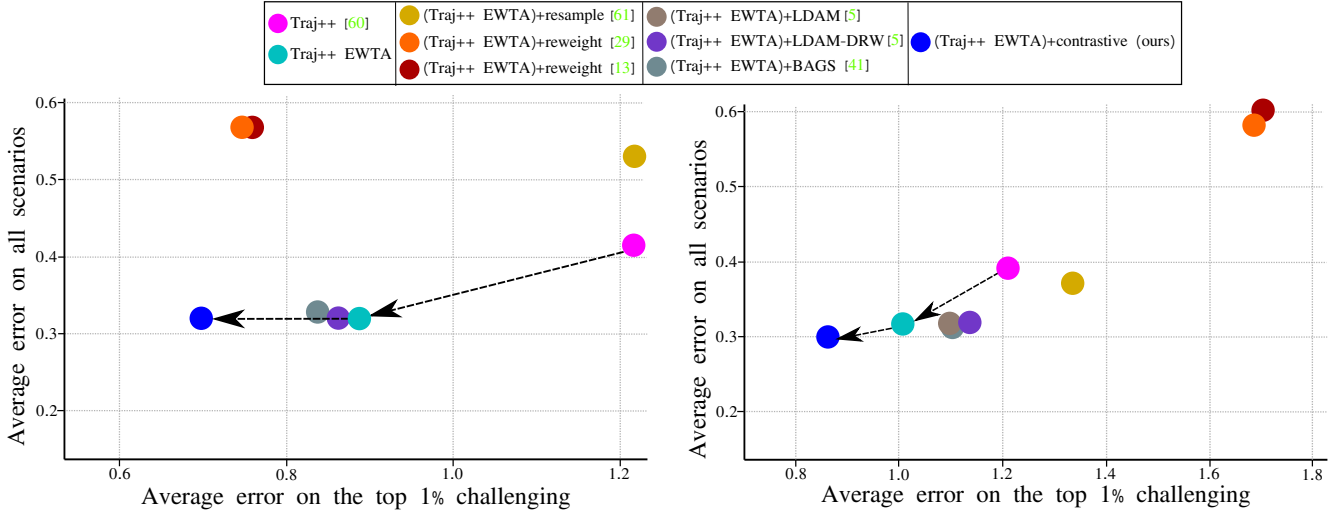
Figure 7. Average vs. Top 1% error comparison on the **ETH-UCY dataset** (left) and the **nuScenes bird's eye view** (right). Our base method of integrating EWTA with the backbone of Trajectron++ (cyan) outperforms the previous state-of-the-art (magenta). Joint learning with the contrastive loss (blue) yields large improvements on the challenging scenarios while not reducing the overall average accuracy. The improvements are indicated by dashed arrows. While the resampling/reweighting baselines also improve on the hard cases, they increase the average error a lot (overfitting). The model-based baselines for long-tailed (LDAM and BAGS) yield only small improvements on ETH-UCY or worse performance on nuScenes bird's eye view.

Figure 8. Average vs. Top 1% error comparison on the **nuScenes egocentric view dataset** (left) and the **Waymo open dataset** (right). Our approach utilizing the contrastive loss (blue) yields a significant improvement on the challenging scenarios while not reducing the overall average accuracy. The improvements are indicated by dashed arrows. While the resampling/reweighting baselines also improve on the hard cases, they increase the average error a lot (overfitting). The model-based baselines for long-tailed (LDAM and BAGS) yield smaller improvements than our method.

	ETH-UCY				nuScenes-Bird's Eye View				nuScenes Egocentric View				Waymo Open Dataset			
	All	Top 3%	Top 2%	Top 1%	All	Top 3%	Top 2%	Top 1%	All	Top 3%	Top 2%	Top 1%	All	Top 3%	Top 2%	Top 1%
Baseline	0.16/0.32	0.47/1.07	0.51/1.13	0.42/0.87	0.19/0.32	0.48/0.88	0.50/0.88	0.59/1.02	7.10	29.98	31.13	36.16	6.39	24.87	25.49	27.32
+ resample [61]	0.25/0.53	0.56/1.16	0.61/1.24	0.61/1.22	0.21/0.37	0.55/0.98	0.61/1.07	0.78/1.33	10.20	18.90	19.37	21.62	10.48	19.46	18.91	19.69
+ reweight [29]	0.28/0.56	**0.41/0.78**	**0.44/0.81**	0.43/0.76	0.33/0.58	0.74/1.28	0.80/1.38	0.99/1.67	14.47	15.33	15.42	16.20	14.00	**17.01**	**16.80**	**16.44**
+ reweight [13]	0.28/0.56	0.43/0.83	0.45/0.86	0.44/0.78	0.34/0.60	0.75/1.33	0.80/1.42	0.99/1.71	16.54	**15.29**	**15.34**	**15.46**	17.43	20.34	19.40	18.79
+ contrastive	**0.16/0.32**	0.46/1.03	0.48/1.03	**0.38/0.71**	**0.18/0.30**	**0.44/0.73**	**0.46/0.72**	**0.54/0.85**	**7.04**	25.05	25.26	27.49	6.49	22.36	22.72	24.09

Table 6. Comparison to the common resampling/reweighting techniques on the four datasets. For each method, we show the min-FDE/min-ADE over all samples and over top 1-3% challenging samples. Our method yields large improvements on the challenging ones while maintaining the average. This is in contrast to the reweighting/resampling baselines, which lead to much worse performance on average (see the error increase on the 'All' columns). Baseline indicates Traj++ EWTA for bird's eye view and FLN-RPN [47] for egocentric view.

(a) ETH   (b) HOTEL   (c) nuScenes (bird's-eye view) (d) nuScenes (ego-centric view)
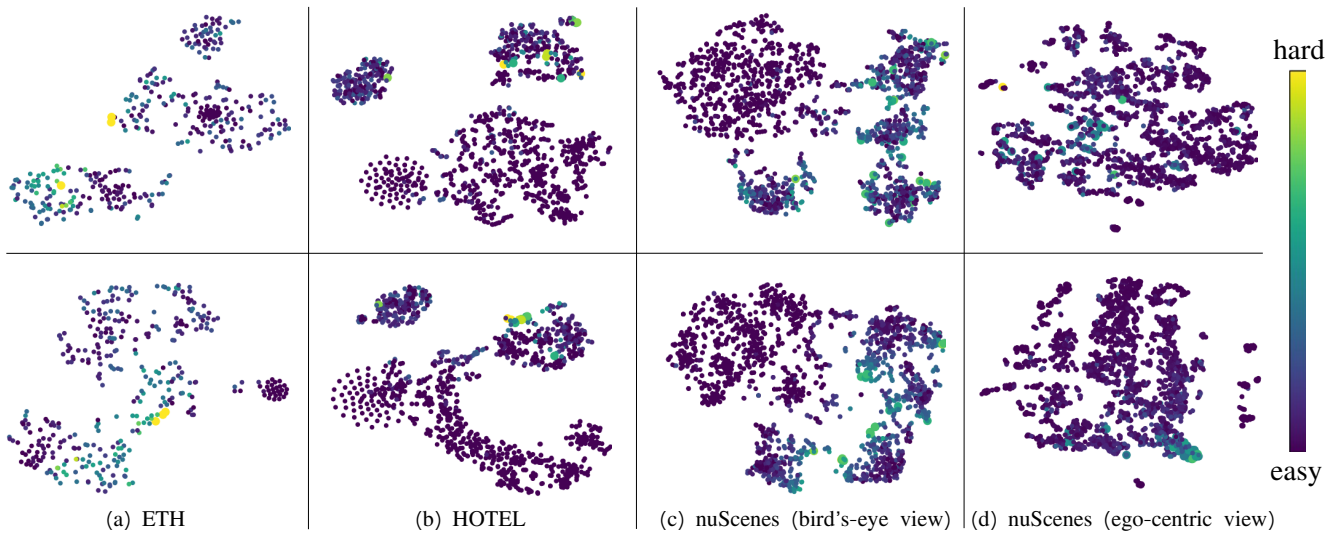
Figure 9. Plot of the feature space using tSNE [70] on three different datasets (a and b are different scenes from the ETH-UCY dataset). **Top.** Training only with the supervised regression loss. **Bottom.** The resulting feature space when trained jointly with the contrastive loss. Large brighte circles indicate the top 1% challenging scenarios. The darker the color of the sample, the easier it is.
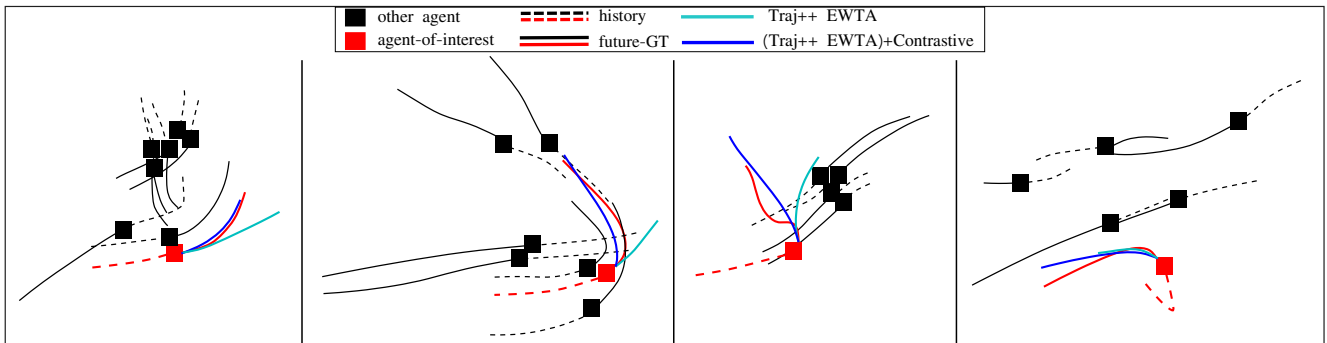

Figure 10. More results from our approach on the ETH-UCY dataset. For all these challenging scenarios, our approach reasons successfully about the social relations to other pedestrians and yields better prediction than the baseline.
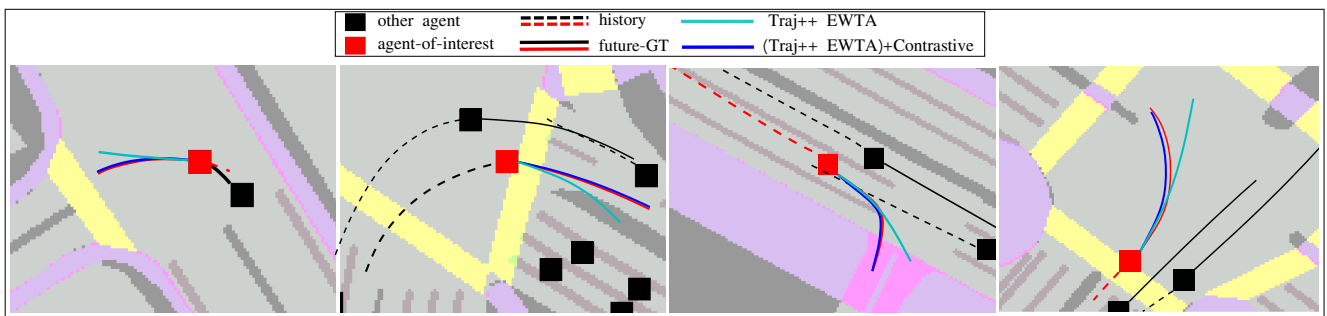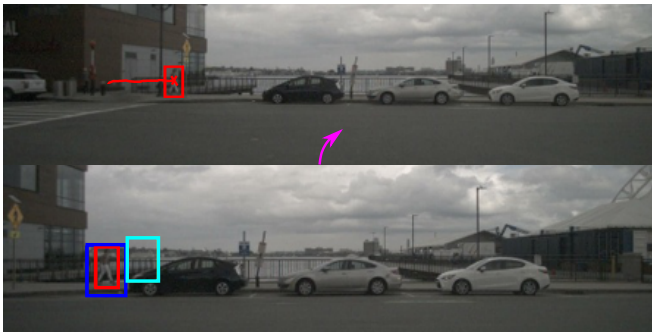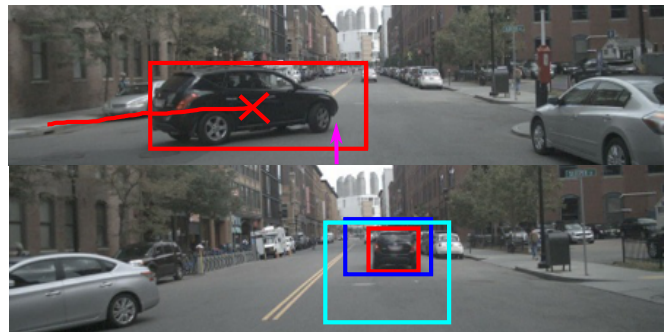

Figure 11. More results from our approach on the nuScenes dataset (bird's-eye view). For all these challenging scenarios, our approach reasons successfully about the semantic cues and predicts the correct trajectory.
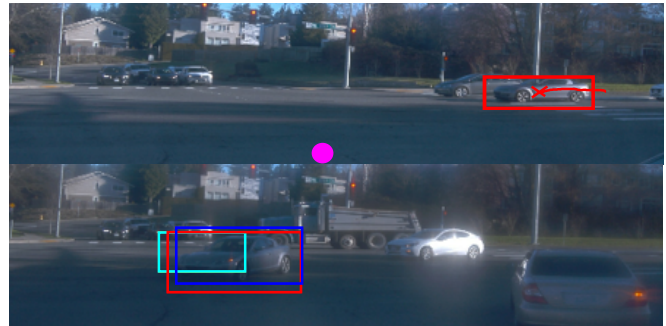
14

(a) A running pedestrian and challenging egomotion

(b) A vehicle turning left and the ego-car is slowing down

(c) A walking pedestrian and challenging egomotion

(d) A vehicle turning right to pass-by the ego-car

Figure 12. More results from our approach on both egocentric view datasets: nuScenes (a-b) and Waymo (c-d). For each example, we show both the last observed image (top) and the future image (bottom) along with the predictions (FLN-RPN [47] and Ours) and the ground truth. We visualize the best hypothesis for each method. The future egomotion is also shown as arrow indicating the motion of the ego-car.