Motion Forecasting with Unlikelihood Training in Continuous Space: Supplementary Materials

This document contains the following sections:

- Gradient Analysis in Gaussian Mixture Model
- Original Learning Objective of Base Models
- L2 Gradient Clipping with Gaussian LaneGCN
- Derivation of Eq.3 and Eq.4
- Analysis of ϵ in Eq.5
- Hyperparameter γ
- Qualitative Results

1 Gradient Analysis in Gaussian Mixture Model

As a supplementary to Sec.3.5 of the main paper, in case we model the output distribution as a Gaussian mixture model $p_{\text{GMM}}(\boldsymbol{y}_t) = \sum_i \phi_i \mathcal{N}(\boldsymbol{y}_t; \hat{\boldsymbol{\mu}}_{i,t}, \hat{\sigma}_{i,t}\boldsymbol{I})$, the gradient of L_t w.r.t. the mean of component i is

$$\begin{aligned}\frac{\partial L_t}{\partial \hat{\boldsymbol{\mu}}_{i,t}} &= \frac{\partial}{\partial \hat{\boldsymbol{\mu}}_{i,t}}(-\log \frac{p_{\text{GMM}}(\boldsymbol{y}_{gt,t})}{p_{\text{GMM}}(\boldsymbol{y}_{neg,t})}) \\ &= -\phi_i(\frac{1}{p_{\text{GMM}}(\boldsymbol{y}_{gt,t})}\frac{\partial \mathcal{N}(\boldsymbol{y}_{gt,t}; \hat{\boldsymbol{\mu}}_{i,t}, \hat{\sigma}_{i,t}\boldsymbol{I})}{\partial \hat{\boldsymbol{\mu}}_{i,t}} - \frac{1}{p_{\text{GMM}}(\boldsymbol{y}_{neg,t})}\frac{\partial \mathcal{N}(\boldsymbol{y}_{neg,t}; \hat{\boldsymbol{\mu}}_{i,t}, \hat{\sigma}_{i,t}\boldsymbol{I})}{\partial \hat{\boldsymbol{\mu}}_{i,t}}) \\ &= -\frac{\phi_i}{\sigma_{i,t}^2}(\frac{\mathcal{N}(\boldsymbol{y}_{gt,t}; \hat{\boldsymbol{\mu}}_{i,t}, \hat{\sigma}_{i,t}\boldsymbol{I})}{p_{\text{GMM}}(\boldsymbol{y}_{gt,t})}(\boldsymbol{y}_{gt,t} - \hat{\boldsymbol{\mu}}_{i,t}) - \frac{\mathcal{N}(\boldsymbol{y}_{neg,t}; \hat{\boldsymbol{\mu}}_{i,t}, \hat{\sigma}_{i,t}\boldsymbol{I})}{p_{\text{GMM}}(\boldsymbol{y}_{neg,t})}(\boldsymbol{y}_{neg,t} - \hat{\boldsymbol{\mu}}_{i,t}))\end{aligned} \tag{1}$$

This gradient shows that the center $\hat{\boldsymbol{\mu}}_{i,t}$ of component i will be pushed towards the ground truth location $\boldsymbol{y}_{gt,t}$ and way from the negative location $\boldsymbol{y}_{neg,t}$. For $\hat{\sigma}_{i,t}$ we have

$$\begin{aligned}\frac{\partial L}{\partial \hat{\sigma}_{i,t}} &= -\phi_i(\frac{1}{p_{\text{GMM}}(\boldsymbol{y}_{gt,t})}\frac{\partial \mathcal{N}(\boldsymbol{y}_{gt,t}; \hat{\boldsymbol{\mu}}_{i,t}, \hat{\sigma}_{i,t}\boldsymbol{I})}{\partial \hat{\sigma}_{i,t}} - \frac{1}{p_{\text{GMM}}(\boldsymbol{y}_{neg,t})}\frac{\partial \mathcal{N}(\boldsymbol{y}_{neg,t}; \hat{\boldsymbol{\mu}}_{i,t}, \hat{\sigma}_{i,t}\boldsymbol{I})}{\partial \hat{\sigma}_{i,t}}) \\ &= -\frac{\phi_i}{\sigma_{i,t}^3}(\frac{\mathcal{N}(\boldsymbol{y}_{gt,t}; \hat{\boldsymbol{\mu}}_{i,t}, \hat{\sigma}_{i,t}\boldsymbol{I})}{p_{\text{GMM}}(\boldsymbol{y}_{gt,t})}(||\boldsymbol{y}_{gt,t} - \hat{\boldsymbol{\mu}}_{i,t}||^2 - \sigma_{i,t}^2) \\ &\quad - \frac{\mathcal{N}(\boldsymbol{y}_{neg,t}; \hat{\boldsymbol{\mu}}_{i,t}, \hat{\sigma}_{i,t}\boldsymbol{I})}{p_{\text{GMM}}(\boldsymbol{y}_{neg,t})}(||\boldsymbol{y}_{neg,t} - \hat{\boldsymbol{\mu}}_{i,t}||^2 - \sigma_{i,t}^2))\end{aligned} \tag{2}$$

2 Original Learning Objective of Base Models

Following is the original loss of Trajectron++ [1]. This loss is used to maximize the lower bound of ground truth's likelihood when the coefficient $\alpha = 1$. For more details, please refer to the original paper [1].

$$\begin{aligned}L = &-\mathbb{E}_{\hat{\boldsymbol{z}} \sim q_{\theta 3}(\boldsymbol{z}|\boldsymbol{X}_i, \boldsymbol{Y}_{i,gt})}[\log p_{\theta 2}(\boldsymbol{Y}_{i,gt} \mid \boldsymbol{X}_i, \hat{\boldsymbol{z}})] \\ &+ \alpha D_{\text{KL}}(q_{\theta 3}(\boldsymbol{z} \mid \boldsymbol{X}_i, \boldsymbol{Y}_{i,gt}) \| p_{\theta 1}(\boldsymbol{z} \mid \boldsymbol{X}_i)) - I_q(\boldsymbol{X}_i; \boldsymbol{z}) \\ &\geq -\log p(\boldsymbol{Y}_{i,gt} \mid \boldsymbol{X}_i, \boldsymbol{Y}_{i,gt}) - I_q(\boldsymbol{X}_i; \boldsymbol{z}), \quad \text{when } \alpha = 1\end{aligned} \tag{3}$$

The original loss of Gaussian LaneGCN, a variant of LaneGCN [2], is

$$L = L_{cls} + \alpha L_{reg} \tag{4}$$

$$L_{cls} = \frac{1}{K-1} \sum_{k \neq \hat{k}} \max(0, c_{m,k} + \epsilon - c_{m,\hat{k}}) \tag{5}$$

$$\begin{aligned} L_{reg} &= \frac{1}{T} \sum_{t=1}^{T} -\log \mathcal{N}(\boldsymbol{y}_{gt,t}; \hat{\boldsymbol{\mu}}_{t,\hat{k}}, \hat{\sigma}^2_{t,\hat{k}} \boldsymbol{I}) \\ &= \frac{1}{T} \sum_{t=1}^{T} \frac{1}{\sigma^2_{t,\hat{k}}} \frac{1}{2} (\boldsymbol{y}_{gt,t} - \hat{\boldsymbol{\mu}}_{t,\hat{k}})^2 + \log \sigma_{t,\hat{k}} + \frac{1}{2} \log 2\pi \end{aligned} \tag{6}$$

Here, $\hat{k}$ indicates the best-predicted mode that is most close to the ground truth. The classification loss L_{cls} is a max-margin loss. For more details of L_{cls}, please refer to the original paper [2].

3 L2 Gradient Clipping with Gaussian LaneGCN

The original regression loss of LaneGCN is smooth L1 distance. Given the prediction error $e = ||\boldsymbol{y}_{gt,t} - \hat{\boldsymbol{\mu}}_{t,\hat{k}}||$, The loss can be written as follows

$$L_{reg} = \begin{cases} \frac{1}{2} e^2 & \text{if } e < 1 \\ e - 0.5 & \text{if } e \geq 1 \end{cases} \tag{7}$$

The gradient of L_{reg} with respect to e in this case is

$$\frac{\mathrm{d}L_{reg}}{\mathrm{d}e} = \begin{cases} e & \text{if } e < 1 \\ 1 & \text{if } e \geq 1 \end{cases} \tag{8}$$

The $e >= 1$ part can be viewed as clipping the gradient of an L2 loss to 1 when the gradient is bigger than 1. Therefore, the smooth L1 loss can be viewed as a normal L2 loss with gradient clipping to avoid too large gradients when the prediction error is high. Since there is also an L2 loss term in the log likelihood loss in Eq.6, we apply this gradient clipping trick to the L2 loss term in Eq.6. The clipping is implemented by creating a new L2 function with modified backward propagation.

4 Derivation of Eq.3 and Eq.4

Here we show how to obtain Eq.3 and Eq.4 of the main paper

$$\begin{aligned} L_t &= -\log \mathcal{N}(\boldsymbol{y}_{gt,t}; \hat{\boldsymbol{\mu}}_t, \hat{\sigma}_t \boldsymbol{I}) + \log \mathcal{N}(\boldsymbol{y}_{neg,t}; \hat{\boldsymbol{\mu}}_t, \hat{\sigma}_t \boldsymbol{I}) \\ &= \frac{1}{2}(\log 2\pi + \log \hat{\sigma}_t^2 + \frac{||\boldsymbol{y}_{gt,t} - \hat{\boldsymbol{\mu}}_t||^2}{\hat{\sigma}_t^2}) - \frac{1}{2}(\log 2\pi + \log \hat{\sigma}_t^2 + \frac{||\boldsymbol{y}_{neg,t} - \hat{\boldsymbol{\mu}}_t||^2}{\hat{\sigma}_t^2}) \\ &= \frac{1}{2\hat{\sigma}_t^2}(||\boldsymbol{y}_{gt,t} - \hat{\boldsymbol{\mu}}_t||^2 - ||\boldsymbol{y}_{neg,t} - \hat{\boldsymbol{\mu}}_t||^2) \end{aligned} \tag{9}$$

$$\frac{\partial L_t}{\partial \hat{\boldsymbol{\mu}}_t} = \frac{\partial}{\partial \hat{\boldsymbol{\mu}}_t} \frac{1}{2\hat{\sigma}_t^2}(||\boldsymbol{y}_{gt,t} - \hat{\boldsymbol{\mu}}_t||^2 - ||\boldsymbol{y}_{neg,t} - \hat{\boldsymbol{\mu}}_t||^2) \tag{10}$$

$$= -\frac{1}{\hat{\sigma}_t^2}((\boldsymbol{y}_{gt,t} - \hat{\boldsymbol{\mu}}_t) + (\hat{\boldsymbol{\mu}}_t - \boldsymbol{y}_{neg,t})) \tag{11}$$

$$\frac{\partial L}{\partial \hat{\sigma}_t} = \frac{\partial}{\partial \hat{\sigma}_t} \frac{1}{2\hat{\sigma}_t^2}(||\boldsymbol{y}_{gt,t} - \hat{\boldsymbol{\mu}}_t||^2 - ||\boldsymbol{y}_{neg,t} - \hat{\boldsymbol{\mu}}_t||^2) \tag{12}$$

$$= -\frac{1}{\hat{\sigma}_t^3}(||\boldsymbol{y}_{gt,t} - \hat{\boldsymbol{\mu}}_t||^2 - ||\boldsymbol{y}_{neg,t} - \hat{\boldsymbol{\mu}}_t||^2) \tag{13}$$

5 Analysis of ϵ in Eq.5 of the Main Paper

$$\frac{\partial \log(x+\epsilon)}{\partial x} = \frac{1}{x+\epsilon} = \frac{x}{x+\epsilon}\frac{1}{x} = \frac{x}{x+\epsilon}\frac{\partial \log(x)}{\partial x} \tag{14}$$

Therefore, ϵ term scales the original gradient by $\frac{x}{x+\epsilon}$

6 Hyperparameter γ

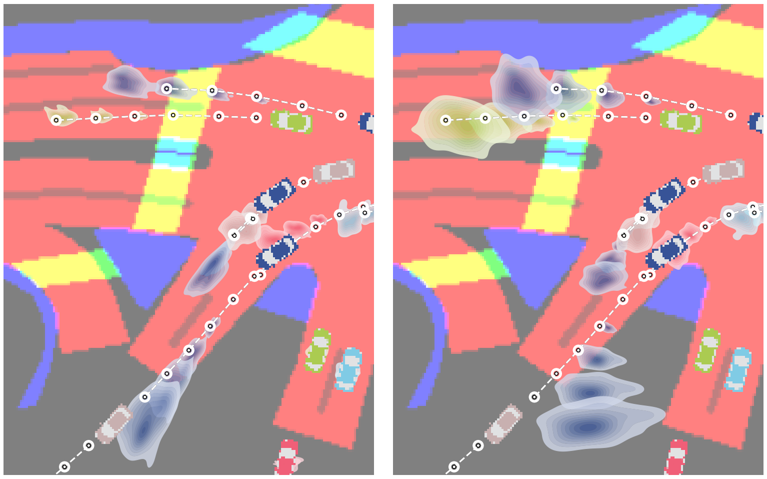
The hyperparameter γ in Eq.2 of the main paper is simply set to 1 and turned on smoothly during training. In this section, we list the performance of the model when $\gamma = 0.1, 0.3, 1, 3, 10$ in the nuScenes dataset. The new numbers reported here (0.1,0.3,3,10 cases) are average over 3 runs. The numbers of $\gamma = 1$ and the original Trajectron++ [1] are the same as the main paper and are averaged over 5 runs.

Model	Trajectron++	$\gamma = 0.1$	$\gamma = 0.3$	$\gamma = 1$	$\gamma = 3$	$\gamma = 10$
FDE-Full	2.74	2.65	2.60	2.51	2.52	2.54
Context-Vio.	10.59%	9.90%	9.29%	8.85%	9.02%	8.84%

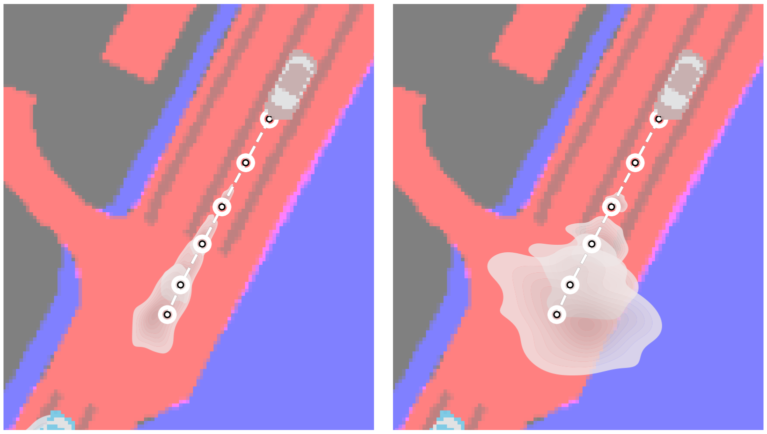
We notice that if we use a small γ, the improvement over the original Trajectron++ is decreased. This is expected since a smaller γ reduces the effect of our unlikelihood loss. The performance doesn't change much when we further increase γ from 1 to 3 and 10. However, we notice that when we set γ to 3 and 10, as the training goes, the prediction accuracy decreases after it reaches the best performance. We think this is because the model focuses too much on obeying the context and pays less attention to getting close to the ground truth in these cases. In conclusion, balancing the original training loss and our likelihood loss matters, and simply setting γ to 1 could balance the original training loss and our unlikelihood loss well.

7 Qualitative Results

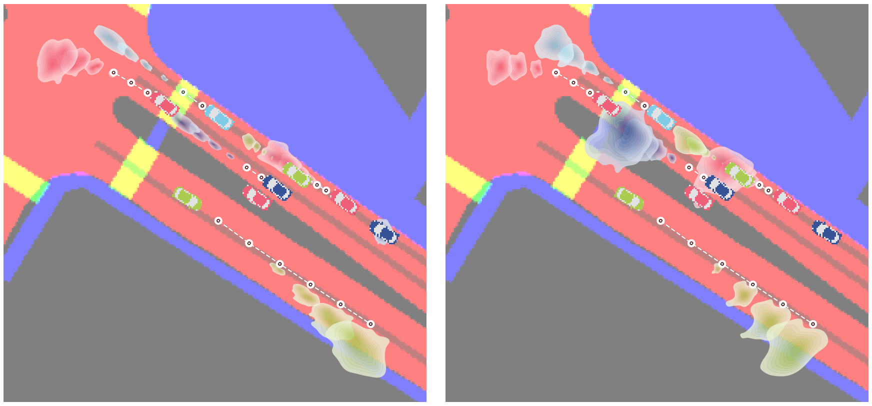
We demonstrate our method's qualitative results compared with Trajectron++ for 3 seconds prediction in nuScenes [3] and with Gaussian LaneGCN in Argoverse [4]. We randomly sample 50 trajectories from the predicted prediction, use kernel density estimation (KDE) to approximate the total output distribution from the samples, and print it out in Fig.1. White points represent the ground truth trajectories. Compared to Trajectron++ and Gaussian LaneGCN, our method complies with the contextual information more and therefore the predicted distributions are more accurate and plausible.



(a) Trajectron++ + Unlikelihood (Ours) (b) Trajectron++

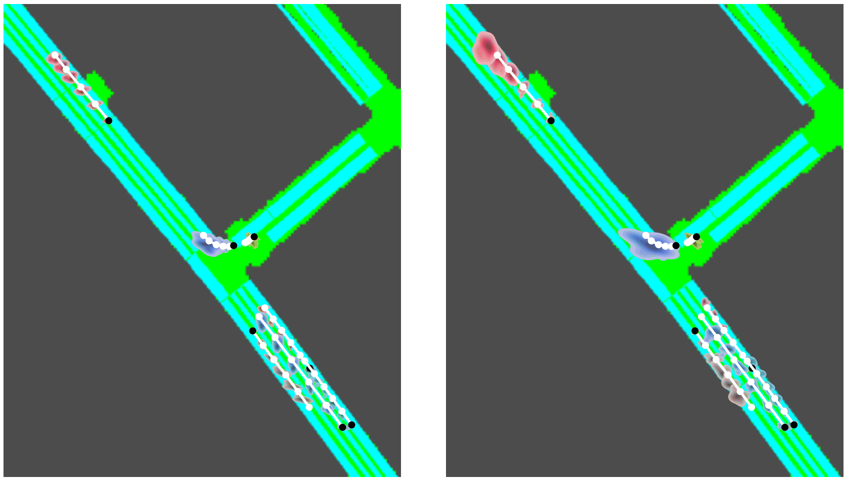


(c) Trajectron++ + Unlikelihood (Ours) (d) Trajectron++

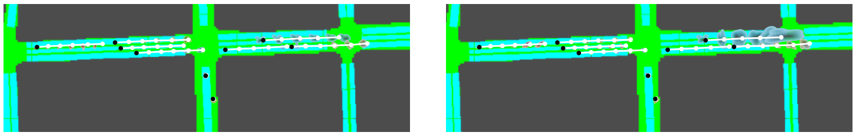


(e) Trajectron++ + Unlikelihood (Ours) (f) Trajectron++

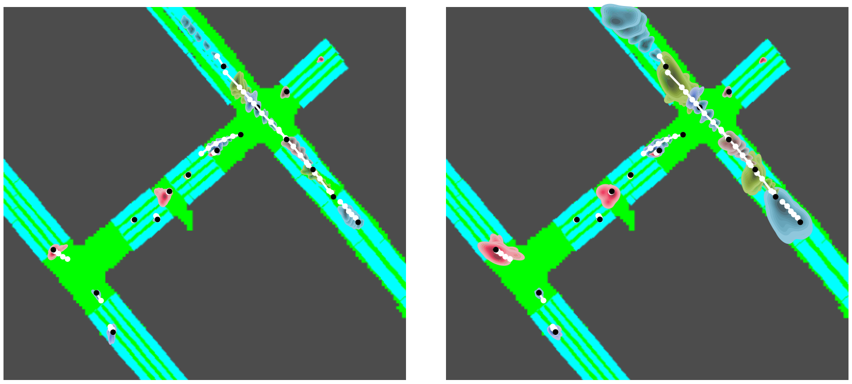
Figure 1: Qualitative results of our method and Trajectron++.



(a) Gaussian LaneGCN + Unlikelihood (Ours)   (b) Gaussian LaneGCN



(c) Gaussian LaneGCN + Unlikelihood (Ours)   (d) Gaussian LaneGCN



(e) Gaussian LaneGCN + Unlikelihood (Ours)   (f) Gaussian LaneGCN

Figure 2: Qualitative results of our method and Gaussian LaneGCN. White dots indicate the ground truth future. Black dots indicates the ground truth locations at the first step.

References

[1] T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone. Trajectron++: Multi-agent generative trajectory forecasting with heterogeneous data for control. *European Conference on Computer Vision 2020*, 2020.

[2] M. Liang, B. Yang, R. Hu, Y. Chen, R. Liao, S. Feng, and R. Urtasun. Learning lane graph representations for motion forecasting. In *European Conference on Computer Vision*, pages 541–556. Springer, 2020.

[3] 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. *arXiv preprint arXiv:1903.11027*, 2019.

[4] M.-F. Chang, J. Lambert, P. Sangkloy, J. Singh, S. Bak, A. Hartnett, D. Wang, P. Carr, S. Lucey, D. Ramanan, et al. Argoverse: 3d tracking and forecasting with rich maps. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8748–8757, 2019.