Long-Term On-Board Prediction of People in Traffic Scenes under Uncertainty (Supplementary Material)

Apratim Bhattacharyya, Mario Fritz, Bernt Schiele
Max Planck Institute for Informatics, Saarland Informatics Campus, Saarbrücken, Germany
{abhattac, mfritz, schiele}@mpi-inf.mpg.de

1. Additional Details of Training Objective

As derived in [2, 1], in Bayesian Regression, the KL divergence between a approximate variational posterior $q(\omega)$ and the true posterior $p(\omega|X,Y)$ distribution of models likely to have generated our data is given by,

$$\mathrm{KL}(q(\omega)\ ||\ p(\omega|X,Y)) \propto \mathrm{KL}(q(\omega)\ ||\ p(\omega)) - \int q(\omega)\log p(Y|X,\omega)d\omega. \tag{A1}$$

In our case, as we train our model to predict future bounding box sequences given the past bounding box sequence, past and future vehiche odometry, we have $X = \{\mathrm{B_p}, \mathrm{O_f}, \mathrm{O_p}\}$ and $Y = \{\mathrm{B_f}\}$. Therefore, the KL divergence is given by,

$$\mathrm{KL}(q(\omega)\ ||\ p(\omega|X,Y)) \propto \mathrm{KL}(q(\omega)\ ||\ p(\omega)) - \int q(\omega)\log p(\mathrm{B_f}|\mathrm{B_p}, \mathrm{O_f}, \mathrm{O_p}, \omega)d\omega. \tag{A2}$$

As the bounding box at time $t+n$ in $\mathrm{B_f}$ is predicted conditioned on the bounding box at time $t+n-1$ and the past bounding box sequence, past and future vehiche odometry, by our Bayesian RNN Encoder-Decoder, the KL divergence is given by,

$$\mathrm{KL}(q(\omega)\ ||\ p(\omega|X,Y)) \propto \mathrm{KL}(q(\omega)\ ||\ p(\omega)) - \sum_t \int q(\omega)\log p(b_t^{t+n}|b_t^{t+n-1}, \mathrm{B_p}, \mathrm{O_p}, \mathrm{O_f}, \omega)d\omega. \tag{A3}$$

During training (as mentioned in subsection 3.5 of the main paper), we use Monte-Carlo integration to estimate the integral in (A3) (using N samples),

$$\mathrm{KL}(q(\omega)\ ||\ p(\omega|X,Y)) \propto \mathrm{KL}(q(\omega)\ ||\ p(\omega)) - \frac{1}{N}\sum_t \sum_{i=0}^{N} \log p(b_t^{t+n}|b_t^{t+n-1}, \mathrm{B_p}, \mathrm{O_p}, \mathrm{O_f}, \hat{\omega}_i), \quad \hat{\omega}_i \sim q(\omega). \tag{A4}$$

The probability term $p(b_t^{t+n}|b_t^{t+n-1}, \mathrm{B_p}, \mathrm{O_p}, \mathrm{O_f}, \hat{\omega}_i)$ takes the form $e^{-\|\hat{b}_i^{t+j} - b_i^{t+j}\|_2^2 (\hat{\Sigma}_i^{t+j})^{-2}}$. Therefore, replacing the log probability term with the exponential squared error term and introducing additional regularization as mentioned in subsection 3.5 of the main paper leads to the training objective used,

$$\frac{1}{4N}\sum_{i=1}^{N}\sum_{j=1}^{n}\|\hat{b}_i^{t+j} - b_i^{t+j}\|_2^2(\hat{\Sigma}_i^{t+j})^{-2} + \lambda\sum_{\mathcal{W}}\|W_k\|_2 + \log\hat{\sigma}_i^2$$

2. Additional Details of Two Stream Model

Here, we include details of each layer of our Two Stream Model. We refer to fully connected layers as Dense and Size refers to the number of neurons in the layer.

Bayesian Bounding Box Prediction Stream. We provide the details of the Bayesian Bounding Box prediction stream in Table 1.

Layer	Type	Size	Activation	Input	Output
In_1	Input			B_{past}	EMB_1
In_2	Input			O_{past}	EMB_1
EMB_1	Dense	64	*ReLU*	$\{\mathrm{In}_1, \mathrm{In}_2\}$	LSTM_{enc1}
LSTM_{enc1}	LSTM	128	*tanh*	EMB_1	EMB_2
EMB_2	Dense	64	*ReLU*	$\left\{\mathrm{LSTM}_{enc1}, \hat{\mathrm{O}}_\mathrm{f}\right\}$	LSTM_{dec1}
LSTM_{dec1}	LSTM	128	*tanh*	EMB_2	Out_1
Out_1	Dense	4		LSTM_{dec}	$\hat{\mathrm{B}}_\mathrm{f}$

Table 1: Details of the Bounding Box Prediction Stream. Note that, the weights of all the layers are sampled from the approximate posterior $q(\omega)$.

Odometry Prediction Stream. We provide the details of the odometry prediction stream in Table 2. We then provide details of the CNN encoder.

3. Database Statistics

In Figure 1 we plot the number of pedestrian tracks of lengths from 6 to 30. The track length distribution is consistent across training and test sets. We observe that there are

Layer	Type	Size	Activation	Input	Output
In_3	Input			O_{past}	LSTM_{enc2}
LSTM_{enc2}	LSTM	128	*tanh*	In_3	LSTM_{dec2}
LSTM_{dec2}	LSTM	128	*tanh*	$\{\text{LSTM}_{enc1}, \text{FC}_3\}$	Out_1
Out_2	Dense	2		LSTM_{dec2}	$\hat{\text{O}}_\text{f}$

Table 2: Details of the Odometry Prediction Stream. Details of the CNN encoder (with output FC_3) follows in Table 3

Layer	Type	Filters	Size	Activation	Input	Output
In_4	Input					C_1
C_1	Conv	32	3×3	*ReLU*	In_2	C_2
C_2	Conv	32	3×3	*ReLU*	C_1	P_1
P_1	MaxPool		2×2		C_2	C_3
C_3	Conv	64	3×3	*ReLU*	P_1	C_4
C_4	Conv	64	3×3	*ReLU*	C_4	P_2
P_2	MaxPool		2×2		C_4	C_5
C_5	Conv	128	3×3	*ReLU*	P_2	C_6
C_6	Conv	128	3×3	*ReLU*	C_5	P_3
P_3	MaxPool		2×2		C_6	C_7
C_7	Conv	256	3×3	*ReLU*	P_3	C_8
C_8	Conv	256	3×3	*ReLU*	C_7	C_8
P_4	MaxPool		2×2		C_8	C_9
C_9	Conv	512	3×3	*ReLU*	P_4	C_{10}
C_{10}	Conv	512	3×3	*ReLU*	C_9	P_5
P_5	MaxPool		2×2		C_{10}	FC_1
FC_1	Dense		1024	*ReLU*	P_5	FC_2
FC_2	Dense		256	*ReLU*	FC_1	FC_3
FC_3	Dense		128	*tanh*	FC_2	LSTM_{dec2}

Table 3: Details of the CNN encoder used to condition the output of the Odometry prediction stream. Conv stands for 2D convolution, MaxPool stands for 2D max pooling and UpSample stands for 2D upsampling operations.

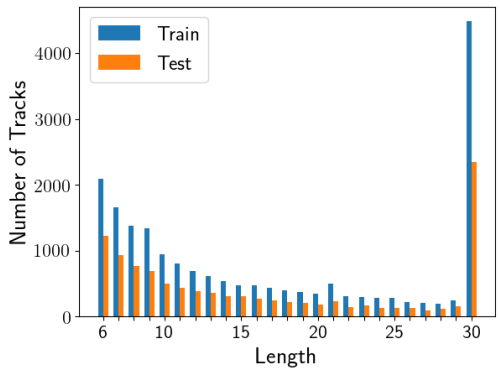

Figure 1: Length of recovered pedestrian tracks in Cityscapes.

many long tracks which stretch over the entire length (30) of the sequence.

4. Evaluation with Varying Size of LSTM

Method	LSTM size	Odometry	MSE	$\mathcal{L}$
LSTM	128	None	650	7.77
LSTM	512	None	705	8.15
LSTM-Bayesian	128	None	**618**	**4.13**
LSTM-Bayesian	512	None	619	4.16

Table 4: Evaluation with varying size of LSTM ($|\text{B}_\text{p}| = 8$).

In the main paper, we evaluate all models constant LSTM vector size of 128. Here, we report results for the (unconditioned) one stream homoscedastic LSTM encoder-decoder model and the one stream Bayesian LSTM encoder-decoder model using a vector size of 512 In Table 4. We see that the homoscedastic version with 512 neurons performs worse than the version with 128 neurons. This is because the larger LSTM over-fits to the bounding box estimation noise in dataset. However, the Bayesian versions have comparable performance, due to dropout which prevents overfitting.

5. Visualization of Odometry Prediction

Visual examples of odometry prediction in Figure 2.

6. Additional Evaluation of our Two-stream Model

Method	Streams	Visual	MSE	$\mathcal{L}$
LSTM	Two	RGB	516	5.15
LSTM-Aleatoric	Two	RGB	618	4.92
LSTM-Bayesian	Two	RGB	**505**	**3.92**

Table 5: Evaluation of Two-stream models ($|\text{B}_\text{p}|, |\text{O}_\text{p}| = 8$).

Here, we compare our Bayesian Two-stream model (Figure 2, of main paper) to, 1. A homoscedastic Two-stream LSTM encoder-decoder model (LSTM). 2. A heteroscedastic Two-stream LSTM encoder-decoder (LSTM-Aleatoric). Note that, both models have the same odometry prediction stream as our Bayesian Two-stream LSTM model (LSTM-Bayesian). The results mirror the evaluation of only the bounding box prediction stream. We see that the heteroscedastic LSTM (LSTM-Aleatoric, 2nd row) outperforms the homoscedastic LSTM (2nd row) with respect to the $\mathcal{L}$ metric. This means that the heteroscedastic Two-stream LSTM learns to capture uncertainty and assigns higher probability to the true bounding box sequence. However, when epistemic uncertainty is not modelled, aleatoric uncertainty tried to compensate and this leads to poorer MSE. Finally, our Bayesian Two-stream LSTM (3rd row) outperforms all other methods.

Figure 2: Odometry prediction: We show predicted odometry for 15 time-steps as points (bottom to top) over-layed on the last visual observation. The distance and angle between subsequent points is the predicted (proportional) speed and steering angle. Color codes: Blue: Ground-truth, Red: Kalman Filter, Yellow: Our LSTM without visual input, Green: Our LSTM with visual input.

7. Additional Analysis of the Quality of our Uncertainty Metric

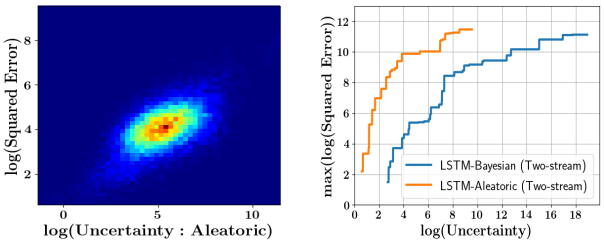

Figure 3: Plot 1 - uncertainty versus squared error, plot 3 - uncertainty versus *maximum* observed squared error.

We compare the quality of the uncertainty metric obtained with our Two-stream LSTM-Bayesian model (Figure 3, of main paper) to that of the Two-stream LSTM-Aleatoric (the heteroscedastic Two-stream LSTM encoder-decoder in the previous section, which models only aleatoric uncertainty). In plot 1 of Figure 3 the aleatoric uncertainty to the log squared error of the mean of the predictive distribution of the Two-stream LSTM-Aleatoric model is shown. We see that the distribution is more spread-out with more outliers compared to our Two-stream LSTM-Bayesian model (plot 1, Figure 3, of main paper). In plot 2 of Figure 3 the *maximum* log squared error (of the mean of the predictive distribution) observed at a certain predicted uncertainty in the test test is shown for both our Two-stream Bayesian model and Two-stream LSTM-Aleatoric. We see that the correlation is poor compared to our Two-stream LSTM-Bayesian model (also in plot 3, Figure 3, of main paper). In particular, the maximum observed log squared error rises very sharply. Therefore, for a robust error bound it is essential to model both epistemic and aleatoric uncertainty.

8. Additional Video Results

We include video results of prediction in **video.mp4**. We include examples of both point estimates and predictive distributions. We include point estimates for comparison against the Kalman Filter and One-stream baselines. The examples show accurate prediction by our Two-stream model over 15 time-steps into the future.

References

[1] Y. Gal and Z. Ghahramani. Bayesian convolutional neural networks with Bernoulli approximate variational inference. In *ICLR workshop track*, 2016. 1

[2] Y. Gal and Z. Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *ICML*, 2016. 1