1. Additional Details of our “Best of Many” Sample Objective

Here we provide additional details of our “Best of Many” samples objective and include additional qualitative results. We begin with the formal statement of the First Mean Value Theorem of Integration [1]. The First Mean Value Theorem of Integration states that, if \( f : [a, b] \rightarrow \mathbb{R} \) is continuous and \( f_2 \) is an integrable function that does not change sign on \([a, b]\), then \( \exists z' \in (a, b) \) such that,

\[
\int_a^b f_1(z) f_2(z) \, dz = f_1(z') \int_a^b f_2(z) \, dz \tag{S1}
\]

The data log-likelihood Equation (3) in the main paper, estimated using importance sampling using a recognition network \( q_\phi \) is given by,

\[
\log(p_\theta(y|x)) = \log \left( \int_a^b p_\theta(y|z, x) \frac{p(z|x)}{q_\phi(z|x, y)} q_\phi(z|x, y) \, dz \right). \tag{S2}
\]

We apply the First Mean Value Theorem of Integration to derive Equation (4) in the main paper, which is,

\[
\log(p_\theta(y|x)) = \log \left( \int_a^b p_\theta(y|z, x) q_\phi(z|x, y) \, dz \right) + \log \left( \frac{p(z'|x)}{q_\phi(z'|x, y)} \right), \tag{S3}
\]

To do this, we set \( f_1(z) = \frac{p(z|x)}{q_\phi(z|x, y)} \) and \( f_2(z) = p_\theta(y|z, x) \times q_\phi(z|x, y) \) (from the data log-likelihood in (S2)). The integral in (S2) can be very well approximated on a large enough bounded interval \([a, b]\). This leads to,

\[
\left( \int_a^b p_\theta(y|z, x) \frac{p(z|x)}{q_\phi(z|x, y)} q_\phi(z|x, y) \, dz \right) = \frac{p(z'|x)}{q_\phi(z'|x, y)} \left( \int_a^b p_\theta(y|z, x) q_\phi(z|x, y) \, dz \right). \tag{S4}
\]

Taking log on both sizes of (S4) leads to (S3). We can further lower bound (S3), leading to Equation (5) in the main paper, which is,

\[
\log(p_\theta(y|x)) \geq \log \left( \int_a^b p_\theta(y|z, x) q_\phi(z|x, y) \, dz \right) + \min_{z' \in (a, b)} \log \left( \frac{p(z'|x)}{q_\phi(z'|x, y)} \right) \tag{S5}\]

However, as mentioned in the main paper, the minimum in (S5) is difficult to estimate. Therefore, we use the following approximation. From (S3), we know that \( \exists z' \in (a, b) \) which lower bounds the data log-likelihood. To maximize this data log-likelihood, we would like to maximize \( \log(f_1(z')) \). However, as we do not know \( z' \), we instead choose to maximize it for a set of \( N \) points in \( (a, b) \),

\[
\log \left( \int_a^b p_\theta(y|z, x) q_\phi(z|x, y) \, dz \right) + \log \left( \frac{p(z_1'|x)}{q_\phi(z_1'|x, y)} \right) + \ldots + \log \left( \frac{p(z_N'|x)}{q_\phi(z_N'|x, y)} \right). \tag{S6}
\]

As values of both \( p \) and \( q_\phi \) are bounded above by 1, the value of the function \( f_2(z_i') = \frac{p(z_i'|x)}{q_\phi(z_i'|x, y)} \) is likely to be low when \( p \) low and \( q_\phi \) is high. Therefore, to give more importance to such points \( z_i' \), we weight each point by \( q_\phi(z_i'|x, y) \),

\[
\log \left( \int_a^b p_\theta(y|z, x) q_\phi(z|x, y) \, dz \right) + q_\phi(z_1'|x, y) \times \log \left( \frac{p(z_1'|x)}{q_\phi(z_1'|x, y)} \right) + \ldots + q_\phi(z_N'|x, y) \times \log \left( \frac{p(z_N'|x)}{q_\phi(z_N'|x, y)} \right). \tag{S7}
\]
We can estimate the data log-likelihood term in (S10) using

\[
\log \left( \int_a^b p_\theta(y|z, x) q_\phi(z|x, y) \, dz \right)
\]

As mentioned in the main paper, we use the re-

\[
-q_\phi(z_1'|x, y) \times \log \left( \frac{q_\phi(z_1'|x, y)}{p(z_1'|x)} \right)
\]

The second integral in (S9) is the KL divergence between

\[
\ldots - q_\phi(z_N'|x, y) \times \log \left( \frac{q_\phi(z_N'|x, y)}{p(z_N'|x)} \right).
\]

If we choose a sufficiently large set of points \(z_i' \in (a, b)\), we can collect the terms in the second part of (S8) and replace them with a single integral,

\[
\log \left( \int_a^b p_\theta(y|z, x) q_\phi(z|x, y) \, dz \right)
\]

The second integral in (S9) is the KL divergence between

\[
\log \left( \int_a^b p_\theta(y|z, x) q_\phi(z|x, y) \, dz \right)
\]

The second integral in (S9) is the KL divergence between

\[
-D_{KL}(q_\phi(z|x, y) \parallel p(z|x)).
\]

We can estimate the data log-likelihood term in (S10) using Monte-Carlo integration. This leads to the “Many Sample” objective from the main paper,

\[
\hat{L}_{MS} = \log \left( \frac{1}{T} \sum_{i=1}^T p_\theta(y|\hat{z}_i, x) \right)
\]

As mentioned in the main paper, we use the re-

\[
-D_{KL}(q_\phi(z|x, y) \parallel p(z|x)), \quad \hat{z}_i \sim q_\phi(z|x, y).
\]

As mentioned in the main paper, we use the re-

\[
-LSTM_{rec} \text{ Dense} \quad 64 \quad ReLU \quad \text{In}_2 \quad LSTM_{rec} \quad \{D_2, D_2\}
\]

\[
D_1 \quad \text{Dense} \quad 64 \quad LSTM_{rec} \quad \mu
\]

\[
D_2 \quad \text{Dense} \quad 64 \quad LSTM_{rec} \quad \sigma
\]

\[
\text{Table 2: Details of the recognition network used during training of our model for Structured Trajectory Prediction.}
\]

2.2. Extension with Visual Input

This model is similar to the model for Structured Trajectory Prediction, expect that the LSTM \(_{dec}\) is additionally conditioned on the output of an CNN encoder. The details are in Table 3 and Table 4. We use the same recognition network as described previously in subsection 2.1.

\[
\text{Table 3: Details of the CNN encoder used with the extended Structured Trajectory Prediction model with Visual Input.}
\]

2. Additional Details of our Models

Here, we include details of each layer of our models.

2.1. Model for Structured Trajectory Prediction

We provide the details of our structured trajectory prediction model in Table 1. Followed by the details of the recognition network \((q_\phi)\) in Table 2. We refer to fully connected layers as Dense and Size refers to the number of neurons in the layer.

\[
\text{Table 4: Details of the LSTM decoder used during training of our model for Structured Trajectory Prediction.}
\]
Table 4: Details our model for extended Structured Trajectory Prediction model with Visual Input. The details of the recognition network $q_{\phi}$ used during training follows in Table 5.

<table>
<thead>
<tr>
<th>Layer</th>
<th>Type</th>
<th>Filters</th>
<th>Size</th>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>In</td>
<td>Input</td>
<td>x</td>
<td>EMB₂</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMB₁</td>
<td>Dense</td>
<td>64</td>
<td>ReLU</td>
<td>In₁</td>
<td>LSTM₂</td>
</tr>
<tr>
<td>LSTM_rec</td>
<td>LSTM</td>
<td>48</td>
<td>tanh</td>
<td>EMB₁</td>
<td>EMB₂</td>
</tr>
<tr>
<td>EMB₂</td>
<td>Dense</td>
<td>64</td>
<td>ReLU</td>
<td>[LSTM_rec, FC₂]</td>
<td>EMB₃</td>
</tr>
<tr>
<td>LSTM_rec</td>
<td>LSTM</td>
<td>64</td>
<td>tanh</td>
<td>EMB₃</td>
<td>Out₁</td>
</tr>
<tr>
<td>Out₁</td>
<td>Dense</td>
<td>2</td>
<td>LSTM_rec</td>
<td>y</td>
<td></td>
</tr>
</tbody>
</table>

Table 5: Details of the recognition network used during training of our extended Structured Trajectory Prediction model with Visual Input.

<table>
<thead>
<tr>
<th>Layer</th>
<th>Type</th>
<th>Filters</th>
<th>Size</th>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>In</td>
<td>Input</td>
<td>y</td>
<td>EMB₂</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMB₁</td>
<td>Dense</td>
<td>64</td>
<td>ReLU</td>
<td>In₁</td>
<td>LSTM₂</td>
</tr>
<tr>
<td>LSTM_rec</td>
<td>LSTM</td>
<td>128</td>
<td>tanh</td>
<td>EMB₂</td>
<td>{D₁, D₂}</td>
</tr>
<tr>
<td>D₁</td>
<td>Dense</td>
<td>64</td>
<td>LSTM_rec</td>
<td>µ</td>
<td></td>
</tr>
<tr>
<td>D₂</td>
<td>Dense</td>
<td>64</td>
<td>LSTM_rec</td>
<td>σ</td>
<td></td>
</tr>
</tbody>
</table>

Table 6: Details our model for Structured Image Sequence Prediction. CLSTM stands for 2D Convolutional LSTM, Conv stands for 2D convolution, MaxPool stands for 2D max pooling and UpSample stands for 2D upsampling operations. The details of the recognition network $q_{\phi}$ used during training follows in Table 7.

<table>
<thead>
<tr>
<th>Layer</th>
<th>Type</th>
<th>Filters</th>
<th>Size</th>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>In</td>
<td>Input</td>
<td>x</td>
<td>CEMB₂</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CEMB₁</td>
<td>Conv</td>
<td>32</td>
<td>3×3</td>
<td>In₁</td>
<td>P₁</td>
</tr>
<tr>
<td>P₁</td>
<td>MaxPool</td>
<td>2×2</td>
<td>CEMB₁</td>
<td>CLSTM rec₁</td>
<td></td>
</tr>
<tr>
<td>CLSTM rec₁</td>
<td>CLSTM</td>
<td>32</td>
<td>3×3</td>
<td>P₁</td>
<td>P₂</td>
</tr>
<tr>
<td>P₂</td>
<td>MaxPool</td>
<td>2×2</td>
<td>CLSTM rec₂</td>
<td>CLSTM rec₁</td>
<td></td>
</tr>
<tr>
<td>CLSTM rec₂</td>
<td>CLSTM</td>
<td>64</td>
<td>3×3</td>
<td>P₂</td>
<td>CEMB₂</td>
</tr>
<tr>
<td>CEMB₂</td>
<td>Conv</td>
<td>32</td>
<td>3×3</td>
<td>{CLSTM rec₂, φ₀}</td>
<td>CLSTM rec₁</td>
</tr>
<tr>
<td>U₁</td>
<td>UpSample</td>
<td>2×2</td>
<td>CLSTM rec₂</td>
<td>CLSTM rec₂</td>
<td></td>
</tr>
<tr>
<td>CLSTM rec₂</td>
<td>CLSTM</td>
<td>64</td>
<td>3×3</td>
<td>U₁</td>
<td>U₂</td>
</tr>
<tr>
<td>U₂</td>
<td>UpSample</td>
<td>2×2</td>
<td>CLSTM rec₂</td>
<td>Out₁</td>
<td></td>
</tr>
<tr>
<td>Out₁</td>
<td>Conv</td>
<td>32</td>
<td>3×3</td>
<td>Out₁</td>
<td>y</td>
</tr>
<tr>
<td>Out₂</td>
<td>Conv</td>
<td>1</td>
<td>3×3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 7: Details of the recognition network used during training of our model for Structured Image Sequence Prediction. CLSTM stands for 2D Convolutional LSTM, Conv stands for 2D convolution, MaxPool stands for 2D max pooling and UpSample stands for 2D upsampling operations.

3. Additional Results

We show additional qualitative results on the HKO dataset in Figure 1 at $t + 5$, $t + 10$ and $t + 15$. We generate $T = 50$ samples and show the sample closest to the groundtruth (Best), the mean of all the samples and the per-pixel variance in the samples. As in the main paper, the qualitative examples demonstrate that our model produces samples which are close to the groundtruth (comparing the Best sample and the groundtruth) and diverse samples (comparing the difference between the mean of the samples and the Best sample).

References

<table>
<thead>
<tr>
<th>Time-step</th>
<th>Groundtruth</th>
<th>Best</th>
<th>Mean</th>
<th>Variance</th>
<th>Groundtruth</th>
<th>Best</th>
<th>Mean</th>
<th>Variance</th>
</tr>
</thead>
<tbody>
<tr>
<td>(t + 5)</td>
<td><img src="image1.jpg" alt="Image" /></td>
<td><img src="image2.jpg" alt="Image" /></td>
<td><img src="image3.jpg" alt="Image" /></td>
<td><img src="image4.jpg" alt="Image" /></td>
<td><img src="image5.jpg" alt="Image" /></td>
<td><img src="image6.jpg" alt="Image" /></td>
<td><img src="image7.jpg" alt="Image" /></td>
<td><img src="image8.jpg" alt="Image" /></td>
</tr>
<tr>
<td>(t + 10)</td>
<td><img src="image9.jpg" alt="Image" /></td>
<td><img src="image10.jpg" alt="Image" /></td>
<td><img src="image11.jpg" alt="Image" /></td>
<td><img src="image12.jpg" alt="Image" /></td>
<td><img src="image13.jpg" alt="Image" /></td>
<td><img src="image14.jpg" alt="Image" /></td>
<td><img src="image15.jpg" alt="Image" /></td>
<td><img src="image16.jpg" alt="Image" /></td>
</tr>
<tr>
<td>(t + 15)</td>
<td><img src="image17.jpg" alt="Image" /></td>
<td><img src="image18.jpg" alt="Image" /></td>
<td><img src="image19.jpg" alt="Image" /></td>
<td><img src="image20.jpg" alt="Image" /></td>
<td><img src="image21.jpg" alt="Image" /></td>
<td><img src="image22.jpg" alt="Image" /></td>
<td><img src="image23.jpg" alt="Image" /></td>
<td><img src="image24.jpg" alt="Image" /></td>
</tr>
</tbody>
</table>

Figure 1: Statistics of samples generated by our LSTM-BMS model on the HKO dataset.