Supplementary Material for Event-based Non-Rigid Reconstruction of Low-Rank Parametrized Deformations from Contours

Yuxuan Xue[1,2,3,4*], Haolong Li[1], Stefan Leutenegger[2] and Jörg Stückler[1]

[1*]Max Planck Institute for Intelligent System, Max-Planck-Ring 4, Tübingen, 72076, Baden-Württemberg, Germany.
[2]Technical University of Munich, Boltzmannstr. 3, Munich, 85748, Bavaria, Germany.
[3]University of Tübingen, Maria-von-Linden Str. 6, Tübingen, 72076, Baden-Württemberg, Germany.
[4]Tübingen AI Center, Maria-von-Linden Str. 6, Tübingen, 72076, Baden-Württemberg, Germany.

*Corresponding author(s). E-mail(s): yuxuan.xue@uni-tuebingen.de;
Contributing authors: haolong.li@tuebingen.mpg.de; stefan.leutenegger@tum.de; joerg.stueckler@tuebingen.mpg.de;

1 Expectation Maximization Framework

1.1 Mathematical Proof

Here we provide the derivation of Eq. 7 to Eq. 10 in the main paper. Note, that in the following the derivation will be for maximum likelihood estimation with EM. We omit the additional log prior term on the pose parameters for maximum a-posterior (MAP) estimation for easier readability.

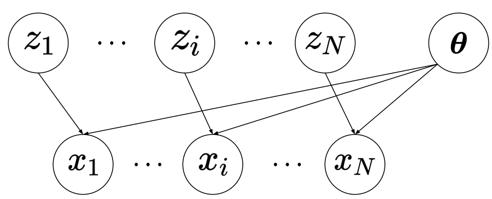

Fig. 1: Bayesian network graph with variables x_i, z_i, and $\boldsymbol{\theta}$. We note the association of each event z_i can be instantiated w.r.t. all mesh faces, namely $z_i = 0$ to $z_i = F$, where F denotes the total number of mesh faces. Each event x_i is only dependent on its parents: the association variable z_i and the mesh pose parameters $\boldsymbol{\theta}$.

We start by showing the example of the expectation of the marginal log likelihood (Eq. 9-30 in [2]) for single event measurement x_0 association and the initialized model parameters $\overline{\boldsymbol{\theta}}$. The expectation is stated in Eq. 1.

In practice, we assume that N events $\{e_1, ..., e_N\}$ in an event buffer are caused by the same mesh with pose $\boldsymbol{\theta}$. This can be represented as in Eq. 2. The z_i is the association situation of event x_i. To be more intuitive on the dependency between variables, we generate a Bayesian network in Fig. 1 to interpret the relationship between events $\{x_1, ..., x_N\}$, association situation for each event $\{z_1, ..., z_N\}$, and desired mesh pose $\boldsymbol{\theta}$. We assume that events are independent from each other. We can apply the chain rule on equation 2 and rewrite it into Eq. 3.

Eq. 3 proves that maximizing the expectation of logarithmic association likelihood of multiple events caused by same pose $\boldsymbol{\theta}$ is equivalent to maximizing the sum of the expectation of all events in an event buffer. In fact, multiple events could accumulate at the same location in an event buffer. The derivation shows that each event can be processed individually.

$$\mathcal{Q}\left(x_0, \boldsymbol{\theta}, \overline{\boldsymbol{\theta}}\right) = \sum_{j=0}^{F} p\left(z_0 = j \mid x_0, \overline{\boldsymbol{\theta}}\right) \ln p(x_0, z_0 = j \mid \boldsymbol{\theta}), \tag{1}$$

$$\begin{aligned} \sum_{z_1} \cdots \sum_{z_i} \cdots \sum_{z_N} (p\left(z_1, \cdots, z_i, \cdots, z_N \mid x_1, \cdots x_i, \cdots, x_N, \overline{\boldsymbol{\theta}}\right) \cdot \\ \ln p\left(x_1, \cdots, x_i, \cdots, x_N, z_1, \cdots, z_i, \cdots, z_N \mid \boldsymbol{\theta}\right)), \end{aligned} \tag{2}$$

$$\begin{aligned}
& \sum_{z_1} \cdots \sum_{z_N} (p\left(z_k, \cdots, z_N \mid x_k, \cdots, x_N, \overline{\boldsymbol{\theta}}\right) \cdot \ln p\left(x_1, \cdots, x_N, z_1, \cdots, z_N \mid \boldsymbol{\theta}\right)) \\
= & \sum_{z_1} \cdots \sum_{z_N} (p\left(z_1 \mid x_1, \cdots x_N, \overline{\boldsymbol{\theta}}\right) \cdots p\left(z_N \mid x_1, \cdots x_N, \overline{\boldsymbol{\theta}}\right) \cdot \ln\left((p(x_1 \mid z_1, \boldsymbol{\theta}) \cdots p\left(x_N \mid z_N, \boldsymbol{\theta}\right)\right)) \\
= & \sum_{z_1} \cdots \sum_{z_N} (p\left(z_1 \mid x_1, \overline{\boldsymbol{\theta}}\right) \cdots p\left(z_N \mid x_N, \overline{\boldsymbol{\theta}}\right) \cdot \left(\ln\left(p\left(x_1 \mid z_1, \boldsymbol{\theta}\right)\right) + \cdots + \ln\left(p\left(x_N \mid z_N, \boldsymbol{\theta}\right)\right)\right)) \\
= & \sum_{z_1} \cdots \sum_{z_N} (p\left(z_1 \mid x_1, \overline{\boldsymbol{\theta}}\right) \underbrace{\cdots p\left(z_N \mid x_N, \overline{\boldsymbol{\theta}}\right)}_{\text{sums up to 1}} \cdot \ln\left(p\left(x_1 \mid z_1, \boldsymbol{\theta}\right)\right)) \\
+ & \sum_{z_1} \cdots \sum_{z_N} \cdots \\
+ & \sum_{z_1} \cdots \sum_{z_N} (\underbrace{p\left(z_1 \mid x_1, \overline{\boldsymbol{\theta}}\right) \cdots}_{\text{sums up to 1}} P\left(z_N \mid x_N, \overline{\boldsymbol{\theta}}\right) \cdot \ln\left(p\left(x_N \mid z_N, \boldsymbol{\theta}\right)\right)) \\
= & \underbrace{\sum_{z_1} (p\left(z_1 \mid x_1, \overline{\boldsymbol{\theta}}\right) \cdot \ln\left(p\left(x_1 \mid z_1, \boldsymbol{\theta}\right)\right))}_{\text{expectation of log-likelihood for } x_1} \\
+ & \cdots \\
+ & \underbrace{\sum_{z_N} (p\left(z_N \mid x_N, \overline{\boldsymbol{\theta}}\right) \cdot \ln\left(p\left(x_N \mid z_N, \boldsymbol{\theta}\right)\right))}_{\text{expectation of log-likelihood for } x_N} \\
= & \sum_{i=0}^{N} \mathcal{Q}\left(x_i, \boldsymbol{\theta}, \overline{\boldsymbol{\theta}}\right)
\end{aligned} \tag{3}$$

2 Hyperparameters

We provide the hyperparameters we used in the experiments. The hyperparameters have been found using the Optuna [1] hyperparameter framework for which we used

$$\sum_j \sum_i \mathrm{MPJPE}\left(M(f(\theta^{i,j}, \mathcal{H})) - M(GT^{i,j})\right) \tag{4}$$

as the objective function for optimization, where $\mathcal{H}$ is the combination of optimized hyperparameters, $f(\cdot)$ is our approach which results in optimized pose θ, GT denotes the ground truth pose, and M represents the parametric model. We sum the MPJPE for each event buffer i in a sequence to formulate the final objective function. For each scenario, we generate 3 training sequences with random motion for the hyperparameter optimization which have different trajectories than in the test set used for evaluation. The training sequences are indexed by j. Due to the differences of the scenarios, we obtained different settings of hyperparameters for the MANO and SMPL-X sequences.

2.0.1 MANO

- $\alpha = 3.104415774078913 \times 10^{-6}$
- $\beta = 0.2766735382305266$
- $\gamma = 11.398513831616794$
- ADAM learning rate: 0.00065505427907943
- $\tau_{lateral} = 0.011249351598402\,15$

2.0.2 SMPL-X

- $\alpha = 1.0304767369667457 \times 10^{-7}$
- $\beta = 0.10679963391728674$
- $\gamma = 0.10986587419562713$
- ADAM learning rate: 0.003150912997278243
- $\tau_{lateral} = 0.0022776835235587$

2.0.3 Nehvi's method on MANO [3]

- $w = 500.1304611210178$
- $C = 0.02955036287711992$
- ADAM learning rate: 0.0054869951307353024

References

[1] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019.

[2] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2006.

[3] Jalees Nehvi, Vladislav Golyanik, Franziska Mueller, Hans-Peter Seidel, Mohamed Elgharib, and Christian Theobalt. Differentiable event stream simulator for non-rigid 3d tracking. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1302–1311, 2021.