# Supplementary material: Spatiotemporal Registration for Event-based Visual Odometry

Daqi Liu Álvaro Parra Tat-Jun Chin
School of Computer Science, The University of Adelaide
{daqi.liu, alvaro.parrabustos, tat-jun.chin}@adelaide.edu.au

## A. Geometry consistency of the full rigid motion (Sec. 3.5 in the main text)

Here we present the justification of the geometry consistency under full (6 Dof) rigid motion. We start from Eq. (35) in the main text

$$\mathbf{M}_{a,b} = \begin{bmatrix} \mathbf{R}_{a,b} & -a\mathbf{R}_{a,b}\mathbf{v}_{a,b} + b\mathbf{v}_{a,b} \\ \mathbf{0} & 1 \end{bmatrix}, \tag{1}$$

where the translation part $-a\mathbf{R}_{a,b}\mathbf{v}_{a,b}+b\mathbf{v}_{a,b}$ between time $a, b \in \mathcal{T}$ not only depend on the difference $b-a$, but also on the times $a$ and $b$. In this case, $\mathbf{M}_{a,b} \neq \mathbf{M}_{c,d}$ such that $a \leq b \leq c \leq d$ and $b-a = c-d$. However, $\mathbf{M}_{a,b}$ can be calculated by solving $\mathbf{R}_{a,b}$ and $\mathbf{v}_{a,b}$ seperately.

We will rewrite **Lemma 1** in the main text with the linear constant velocity assumption

$$\mathbf{v}_{a,b} = (b-a) \times \mathbf{v} \tag{2}$$

in the following lemma.

**Lemma 1** *Assuming that the camera undergoes a full rigid motion with constant linear velocity in the period $\mathcal{T}$, both the linear velocity $\mathbf{v}_{a,b}$ and rotational motion $\mathbf{R}_{a,b}$ between any $a, b \in \mathcal{T}$ with $a \leq b$ depends only on the difference $b-a$.*

Under the motion model in Lemma 1, the geometry consistency of full rigid motion is described by the following equation:

$$d'\hat{\mathbf{u}}' + t'\mathbf{v}_{a,b} = d\mathbf{R}_{a,b}\hat{\mathbf{u}} + t\mathbf{R}_{a,b}\mathbf{v}_{a,b}. \tag{3}$$

Since $t\mathbf{R}_{a,b}\mathbf{v}_{a,b}$ is nonlinear, both $\mathbf{R}_{a,b}$ and $\mathbf{v}_{a,b}$ cannot be solved using SVD directly. However, a gradient-based optimisation approach such as Levenberg Marquardt can be used to calculate $\mathbf{R}_{a,b}$ and $\mathbf{v}_{a,b}$.

## B. Feature tracking over consecutive frames (Sec. 4 in the main text)

Our feature tracking process is illustrated in Fig. 1. Firstly, we generate event correspondences $\{\langle \mathbf{e}_j, \mathbf{e}_{n_j} \rangle\}_{j=1}^{K}$ using full batch $\mathcal{E}$ in time window $\mathcal{T} = [\alpha, \beta]$. Secondly, we obtain $\bar{\mathcal{E}}$ in time window $\mathcal{T}' = [\alpha', \beta]$ using Eq. (41) in the main text. Finally, the reduced batch $\bar{\mathcal{E}}'$ can be computed using Eq. (42) in the main text and new correspondences in time window $\mathcal{T}' = [\alpha', \beta']$ are generated on $\bar{\mathcal{E}}'$.

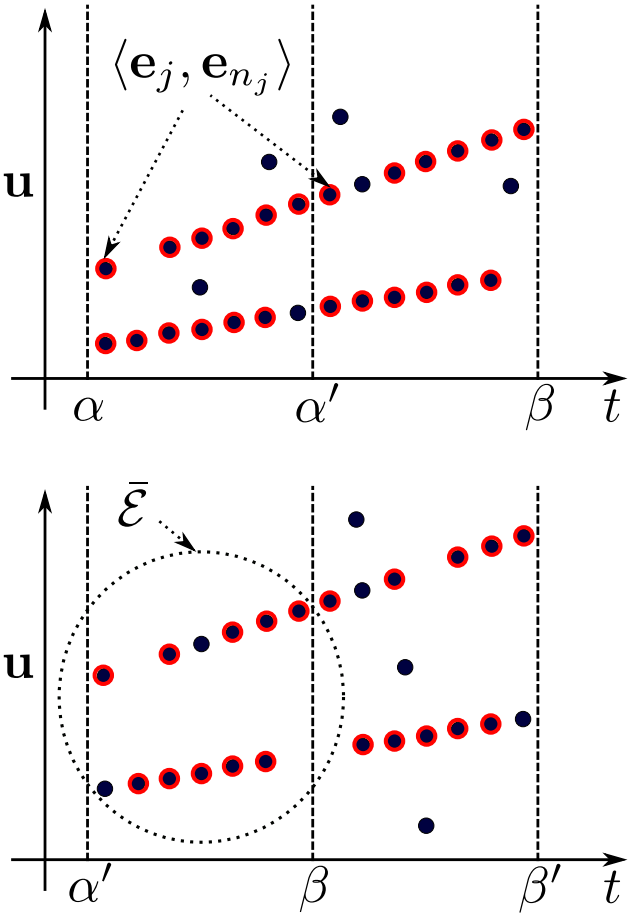



Figure 1. Feature trackes in time windows $\mathcal{T}$ and $\mathcal{T}'$. points with red circles are valid tracks

| Sequences | $\text{VSTR}_\text{C}$ | $\text{VSTR}_\text{A}$ | VCM | VEM | ZHU |
|---|---|---|---|---|---|
| shapes | 6.57 | **6.52** | 17.62 | 9, 19 | 128.84 |

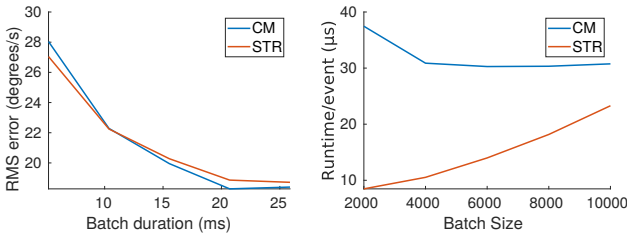
Table 1. Average absolute orientation error (deg) per batch over all instances in shapes sequence

## C. Result of varying batch sizes (Sec. 3.4 in the main text)

Fig. 2 depicts RMS angular velocity errors versus batch duration $|\mathcal{T}|$, and runtime at different batch sizes $N$ for shapes and the sequences in RobotEvt dataset that are not presented in the main text. Since the pattern of the shapes sequence is simpler than the sequences presented in the main text, we reduce batch sizes from $N = 2,000$ to $10,000$.

## D. Result of VO (Sec. 4.1 in the main text)

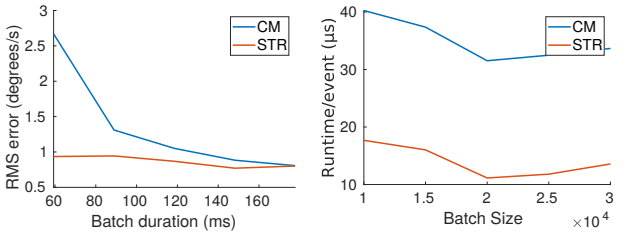
Table 1 depicts the average absolute orientation errors of the shapes sequence. We used a different setting for shapes for which we run all methods with batches of size $N = 6000$ and $\epsilon_k = 1000$ for $\text{VSTR}_\text{A}$ and $\text{VSTR}_\text{C}$. Fig. 3, 4, 5 and 6 plot the absolute orientation trajectories and absolute orientation errors for all the remaining sequences (not included in the main text) on both RobotEvt and Uth dataset.




(a) Error vs batch duration $|\mathcal{T}|$.



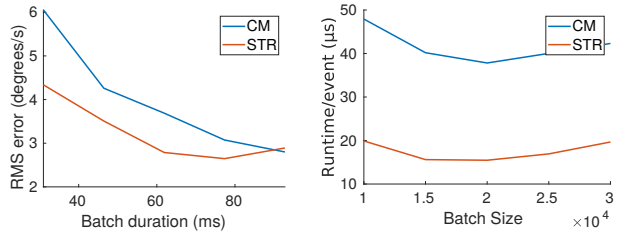

(b) Runtime vs batch size $N$.




(c) Error vs batch duration $|\mathcal{T}|$.



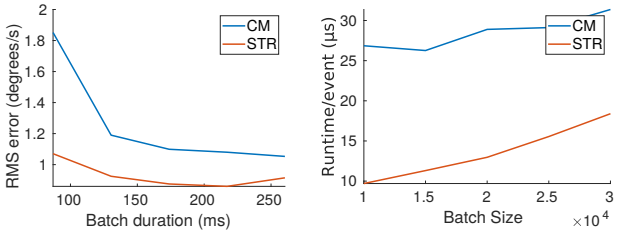

(d) Runtime vs batch size $N$.




(e) Error vs batch duration $|\mathcal{T}|$.



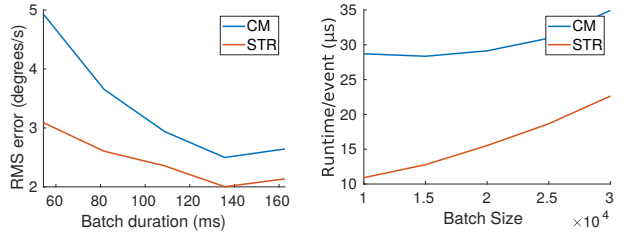

(f) Runtime vs batch size $N$.




(g) Error vs batch duration $|\mathcal{T}|$.




(h) Runtime vs batch size $N$.




(i) Error vs batch duration $|\mathcal{T}|$.




(j) Runtime vs batch size $N$.

Figure 2. Motion estimation error and runtime of CM and STR on shapes, PureRot_slow_On, PureRot_Fast_On, PureRot_slow_Off and PureRot_Fast_Off.

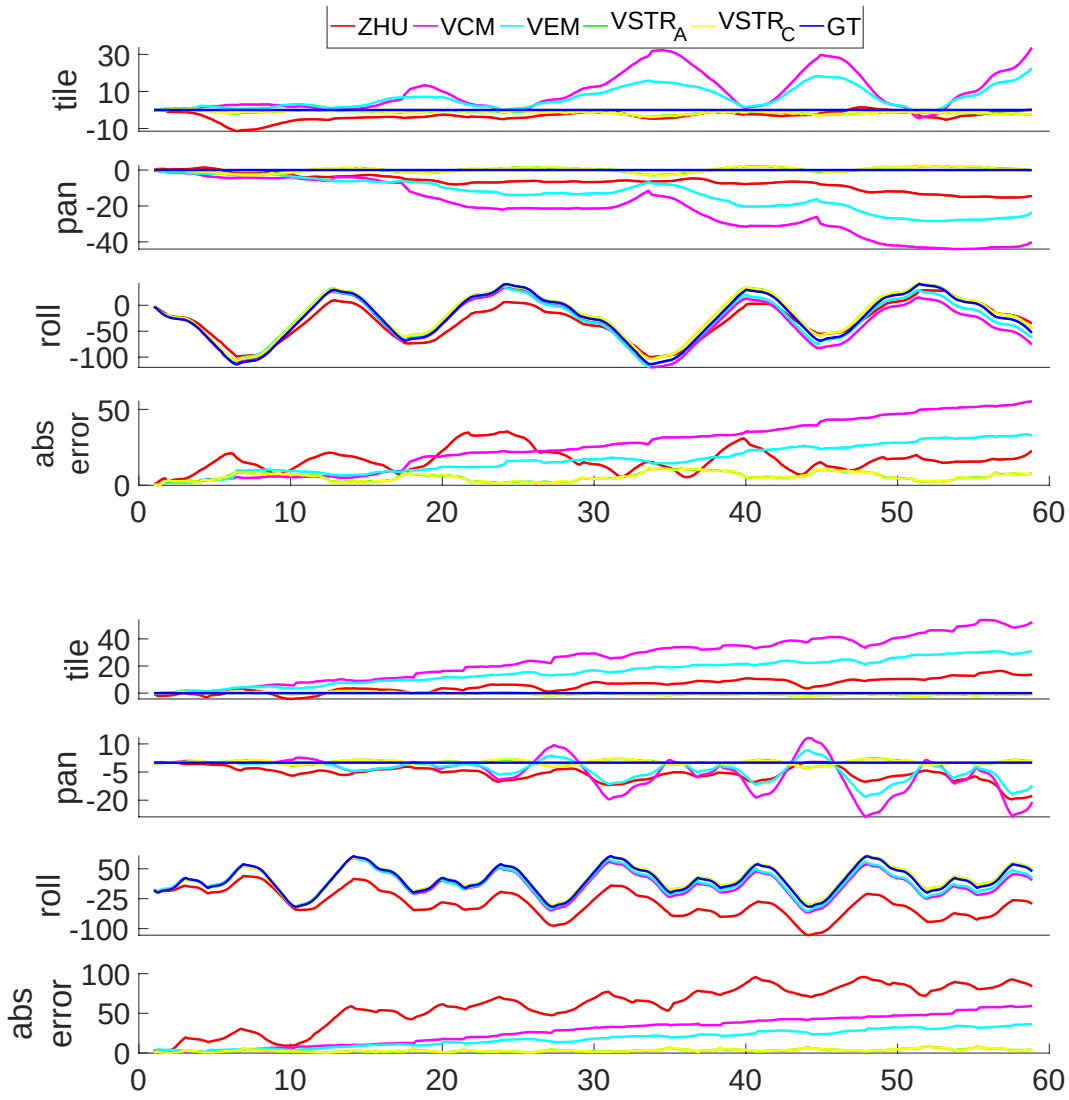



Figure 3. Absolute orientation trajectories (plotted as Euler angles) and absolute orientation error over time. From top to bottom: `PureRot_Slow_On` and `PureRot_Mid_On`.

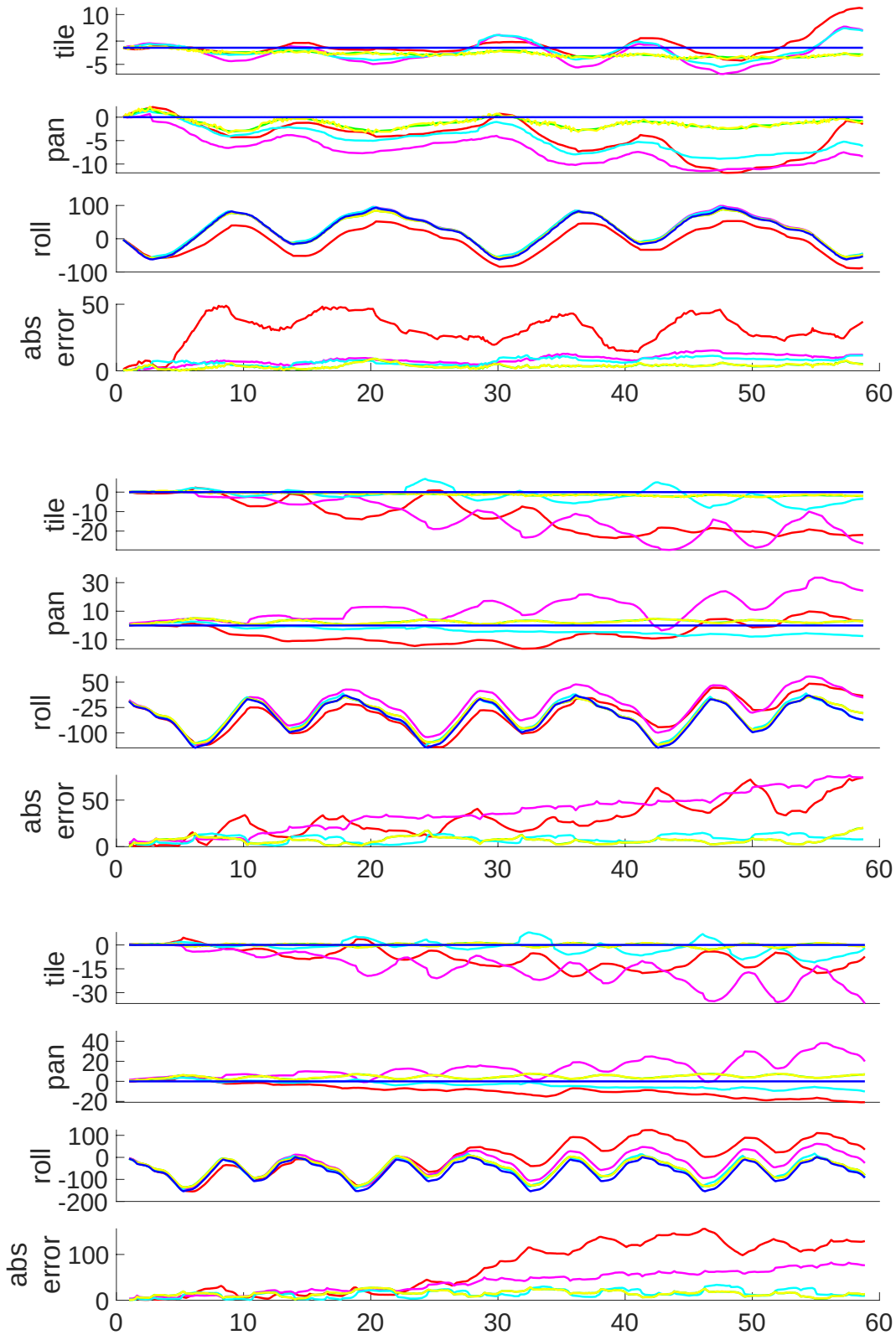



Figure 4. Absolute orientation trajectories (plotted as Euler angles) and absolute orientation error over time. From top to bottom: `PureRot_Slow_Off`, `PureRot_Mid_Off` and `PureRot_Fast_Off`.

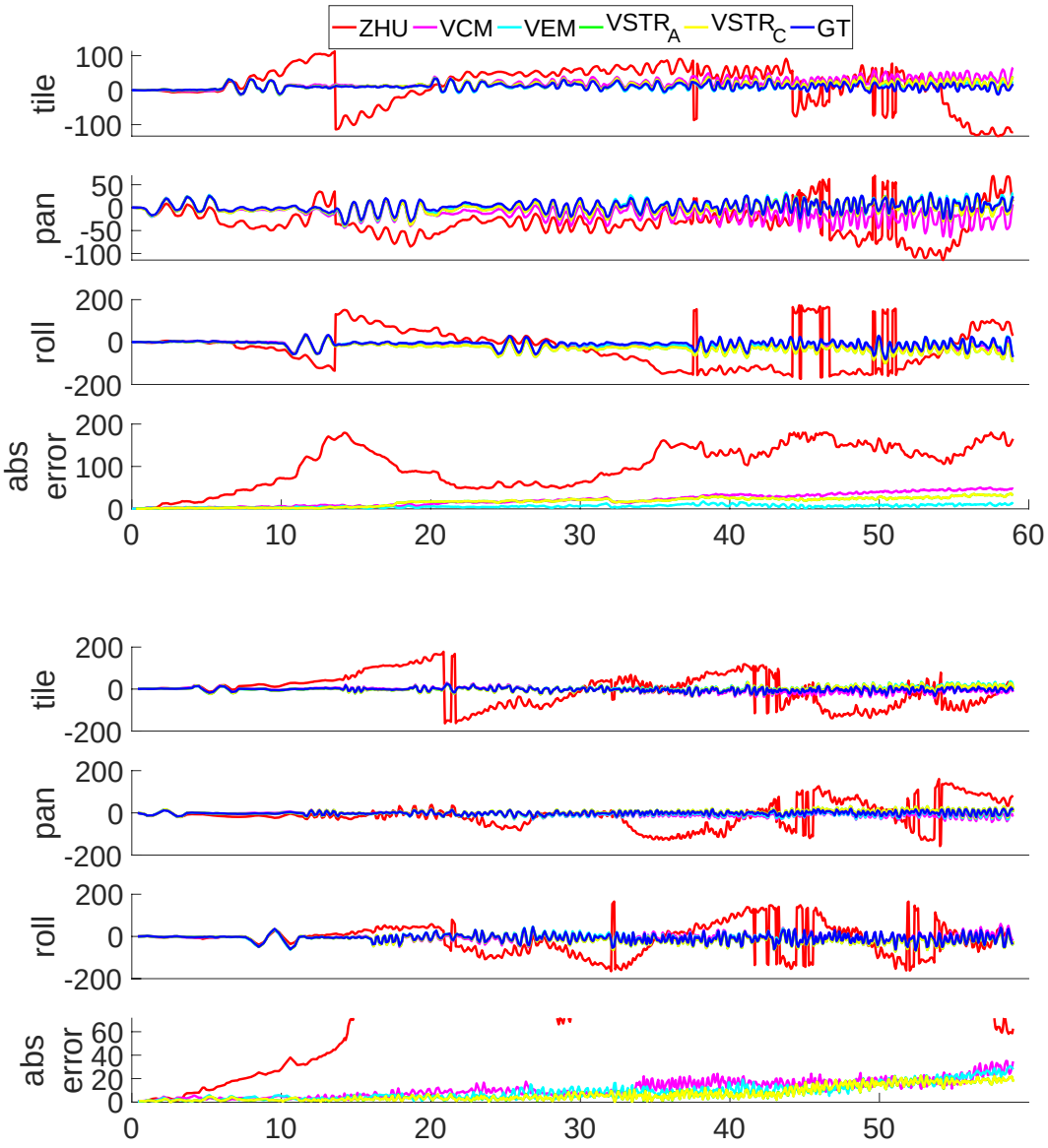



Figure 5. Absolute orientation trajectories (plotted as Euler angles) and absolute orientation error over time. From top to bottom: `dynamic` and `shapes`.