EventCap: Monocular 3D Capture of High-Speed Human Motions using an Event Camera

CVPR 2020

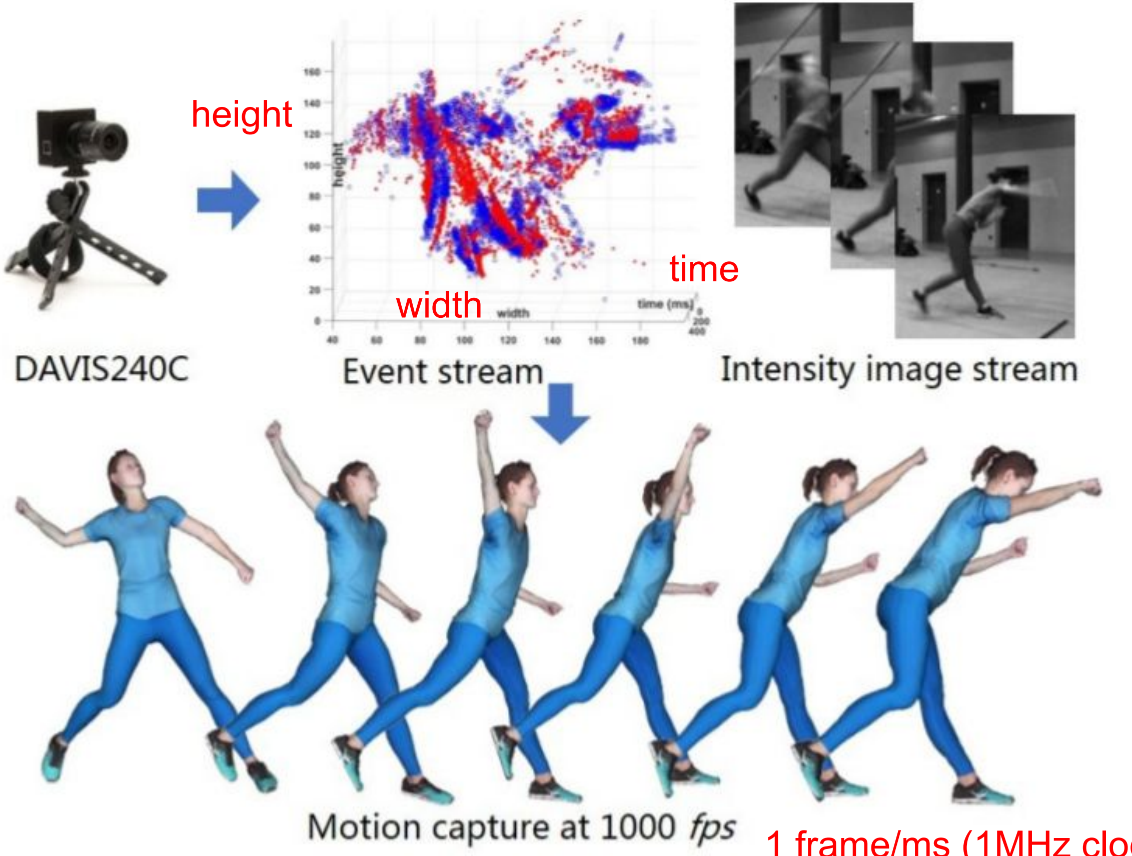
Lan Xu, Weipeng Xu, Vladislav Golyanik, Marc Habermann, Lu Fang, Christian Theobalt

Tsinghua-Berkeley Shenzhen Institute, Tsinghua University, China
Max Planck Institute for Informatics, Saarland Informatics Campus, Germany
Robotics Institute, Hong Kong University of Science and Technology, Hong Kong

Recap: Event camera

- Output: sequence of events (local brightness change on each pixel)
- An event: (x, y, p, t)
    - (x, y): pixel coordinate
    - p: brightness change, binary (+1/-1)
    - T: timestamp
- advantages
    - high **temporal** resolution (1MHz clock, 10 us)
    - low **latency** (no motion blur)
    - low **power** (only transmit brightness change)
    - high dynamic range (able to acquire information from moonlight to daylight)

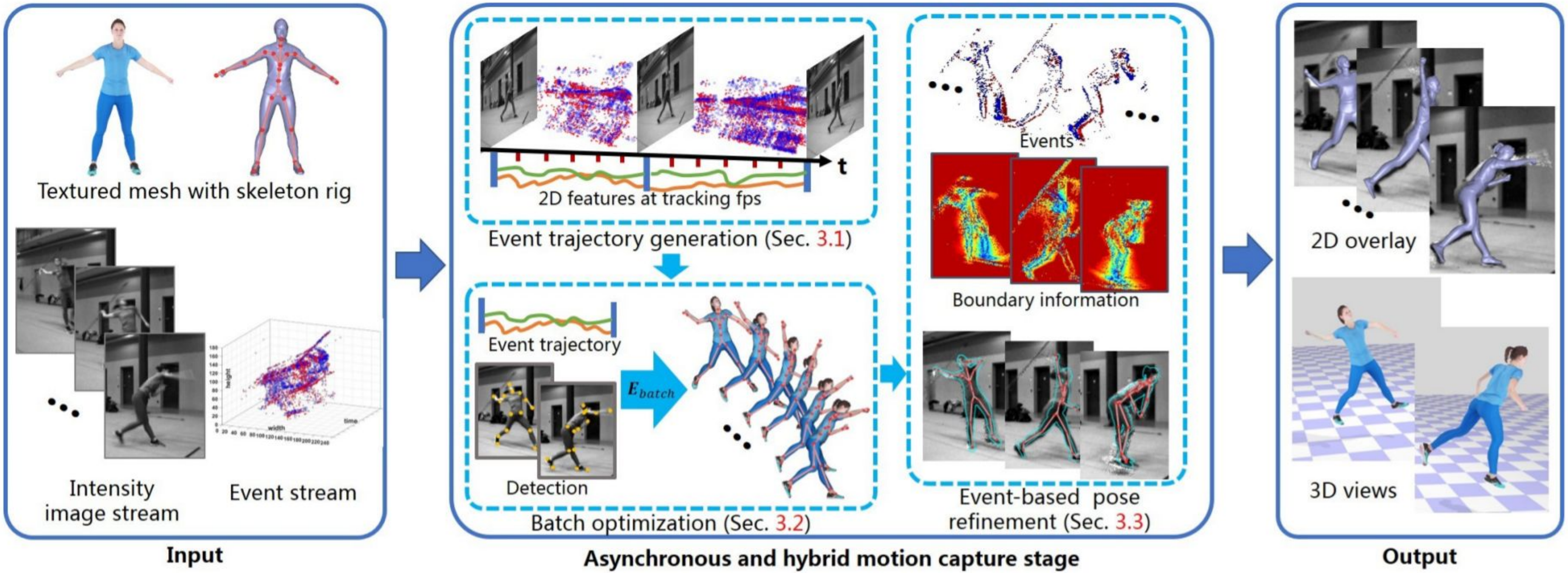
What is the paper about?



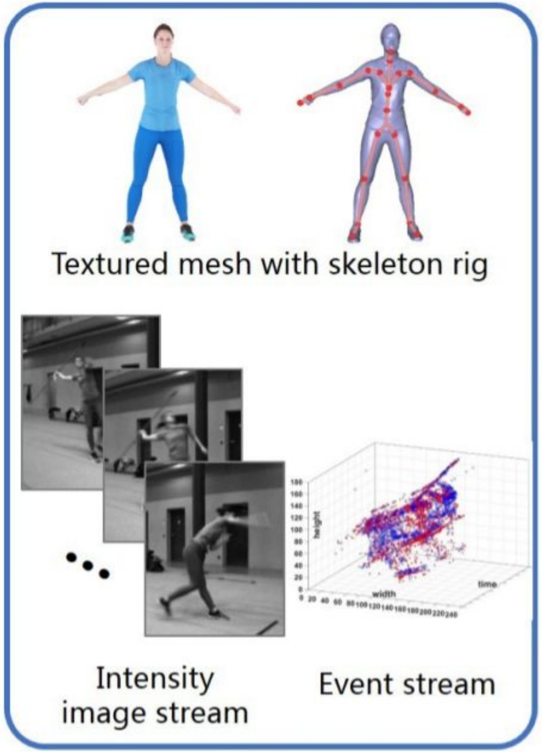
Capture high-speed human motions

1 frame/ms (1MHz clock, 10 us)

Method: three steps



Method: input



Input

Template Mesh Acquisition

1. scanner to generate the template mesh and rigging via SMPL

2. use image-based human shape estimation algorithms, HMR

Method: event trajectory generation

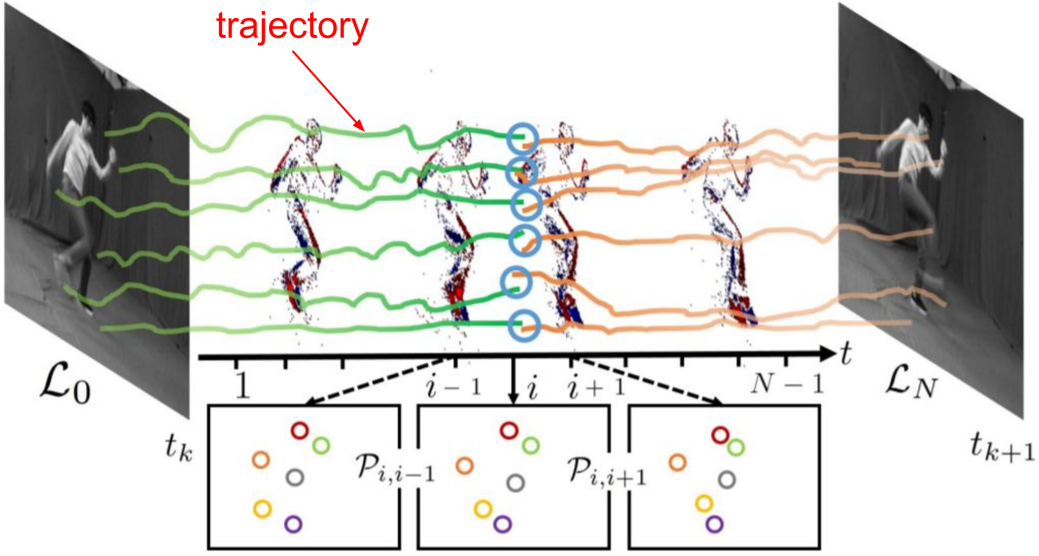
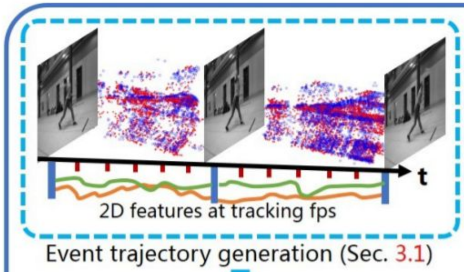


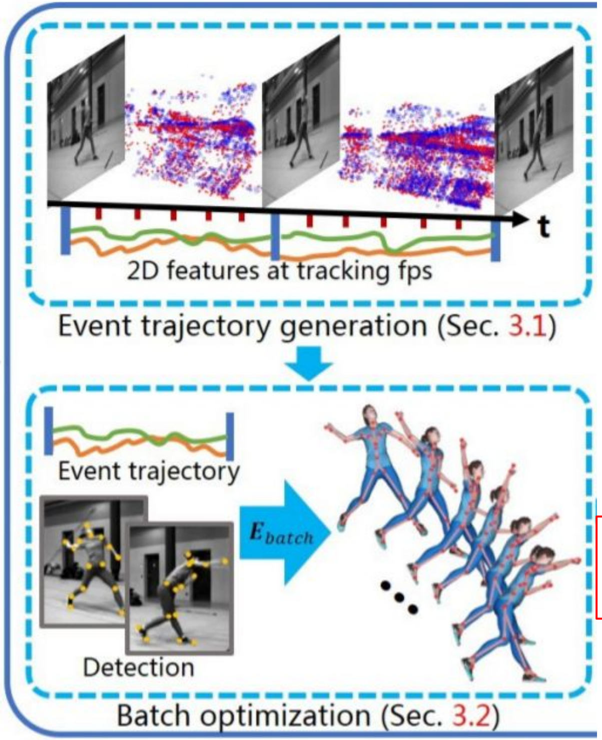
Figure 3: Illustration of asynchronous event trajectories between two adjacent intensity images. The green and orange curves represent the forward and backward event trajectories of exemplary photometric features. The blue circles denote alignment operation. The color-coded circles below indicate the 2D feature pairs between adjacent tracking frames.



- **N** tracking frame within two intensity images between **t_k** and **t_{k+1}**
- $\{\mathcal{T}(h)\}, h \in [1, H]$
  H event trajectories
  (the temporal 2D pixel locations)
- $\mathcal{P}_{i,*} = \{(p_{i,h}, p_{*,h})\}$
  event correspondences, p_{i, h} means 2D pixel for the i-th intensity image frame on the h-th trajectory

* Asynchronous, photometric feature tracking using events and frames. (ECCV 2018)

Method: batch optimization



skeleton parameters

minimize:

$$\boldsymbol{E}_{\text{batch}}(\mathcal{S}) = \lambda_{\text{adj}}\boldsymbol{E}_{\text{adj}} + \lambda_{\text{2D}}\boldsymbol{E}_{\text{2D}} + \lambda_{\text{3D}}\boldsymbol{E}_{\text{3D}} + \lambda_{\text{temp}}\boldsymbol{E}_{\text{temp}}.$$

Event Correspondence Term

projection

$$\boldsymbol{E}_{\text{adj}}(\mathcal{S}) = \sum_{i=1}^{N-1} \sum_{j\in\{i-1,i+1\}} \sum_{h=1}^{H} \tau(p_{i,h}) \|\pi(v_{i,h}(\mathbf{S}_j)) - p_{j,h}\|_2^2,$$

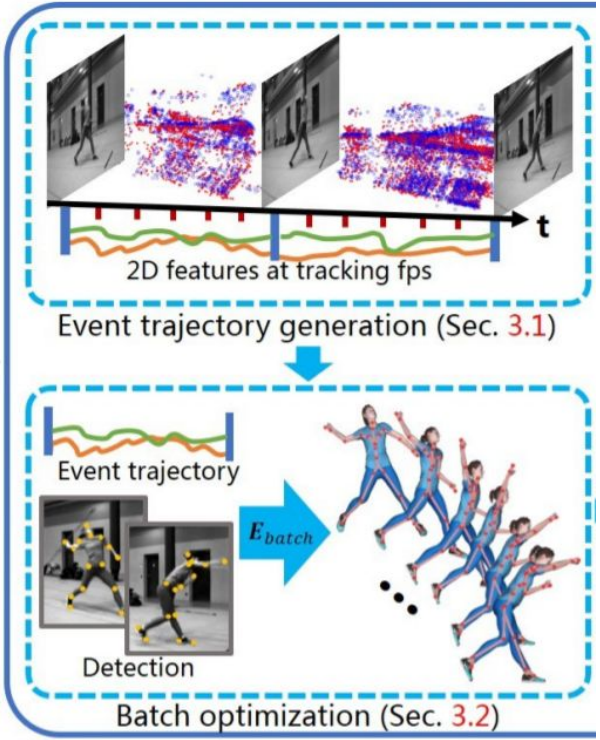
N intensity image frames

Indicator, p corresponds to a valid vertex

vertex of mesh

Method: batch optimization

skeleton parameters



minimize:

$$\boldsymbol{E}_{\text{batch}}(\mathcal{S}) = \lambda_{\text{adj}}\boldsymbol{E}_{\text{adj}} + \lambda_{\text{2D}}\boldsymbol{E}_{\text{2D}} + \lambda_{\text{3D}}\boldsymbol{E}_{\text{3D}} + \lambda_{\text{temp}}\boldsymbol{E}_{\text{temp}}.$$

2D and 3D Detection Terms (intensity image)

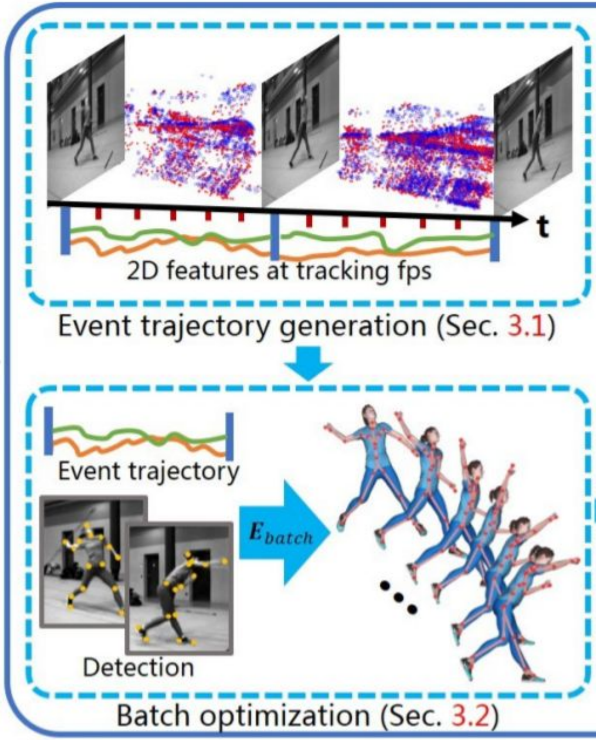
$$\boldsymbol{E}_{\text{2D}}(\mathcal{S}) = \sum_{f\in\{0,N\}} \sum_{l=1}^{N_J+4} \|\pi(J_l(\mathbf{S}_f)) - \mathbf{P}^{2D}_{f,l}\|_2^2$$

OpenPose

$$\boldsymbol{E}_{\text{3D}}(\mathcal{S}) = \sum_{f\in\{0,N\}} \sum_{l=1}^{N_J} \|J_l(\mathbf{S}_f) - (\mathbf{P}^{3D}_{f,l} + \mathbf{t}')\|_2^2$$

VNet

Method: batch optimization



skeleton parameters

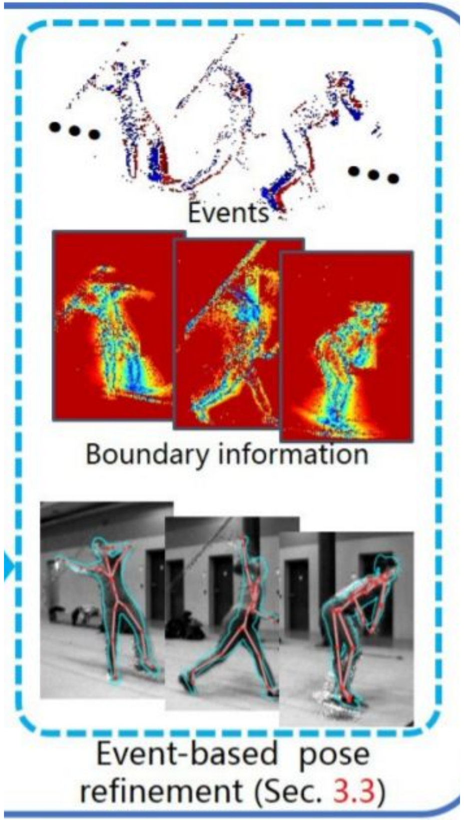
minimize:

$$\boldsymbol{E}_{\text{batch}}(\mathcal{S}) = \lambda_{\text{adj}}\boldsymbol{E}_{\text{adj}} + \lambda_{\text{2D}}\boldsymbol{E}_{\text{2D}} + \lambda_{\text{3D}}\boldsymbol{E}_{\text{3D}} + \lambda_{\text{temp}}\boldsymbol{E}_{\text{temp}}.$$

Temporal Stabilization Term

$$\boldsymbol{E}_{\text{temp}}(\mathcal{S}) = \sum_{i=0}^{N-1}\sum_{l=1}^{N_J}\phi(l)\|J_l(\mathbf{S}_i) - J_l(\mathbf{S}_{i+1})\|_2^2$$

Method: event-based pose refinement



Motivation: Most of the **events** are triggered by the moving **edges** in the image plane, which have a strong correlation with the actor's **silhouette**.

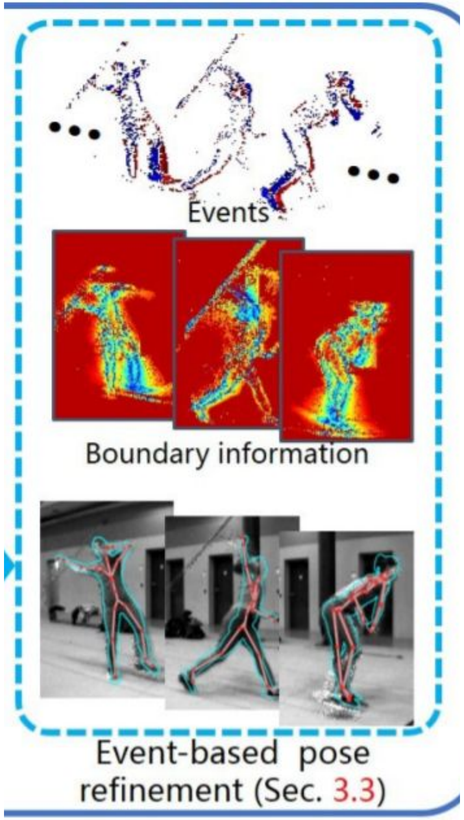
In each Iterative Closest Point (ICP) iteration, first search for the **closest event** for each **boundary pixel** of the projected mesh.

minimize:

$$\boldsymbol{E}_{\mathrm{refine}}(\mathbf{S}_f) = \lambda_{\mathrm{sil}} \boldsymbol{E}_{\mathrm{sil}}(\mathbf{S}_f) + \lambda_{\mathrm{stab}} \boldsymbol{E}_{\mathrm{stab}}(\mathbf{S}_f)$$

2D normal vector corresponding

Method: event-based pose refinement



minimize:

$$\boldsymbol{E}_{\text{refine}}(\mathbf{S}_f) = \lambda_{\text{sil}}\boldsymbol{E}_{\text{sil}}(\mathbf{S}_f) + \lambda_{\text{stab}}\boldsymbol{E}_{\text{stab}}(\mathbf{S}_f)$$

stability term

$$\boldsymbol{E}_{\text{stab}}(\mathbf{S}_f) = \sum_{l=1}^{N_J} \|J_l(\mathbf{S}_f) - J_i(\hat{\mathbf{S}}_f)\|_2^2$$

$$\boldsymbol{E}_{\text{sil}}(\mathbf{S}_f) = \sum_{b \in \mathcal{B}} \|\mathbf{n}_b^{\mathbf{T}}(\pi(v_b(\mathbf{S}_f) - u_b))\|_2^2$$

2D normal vector corresponding to boundary pixel

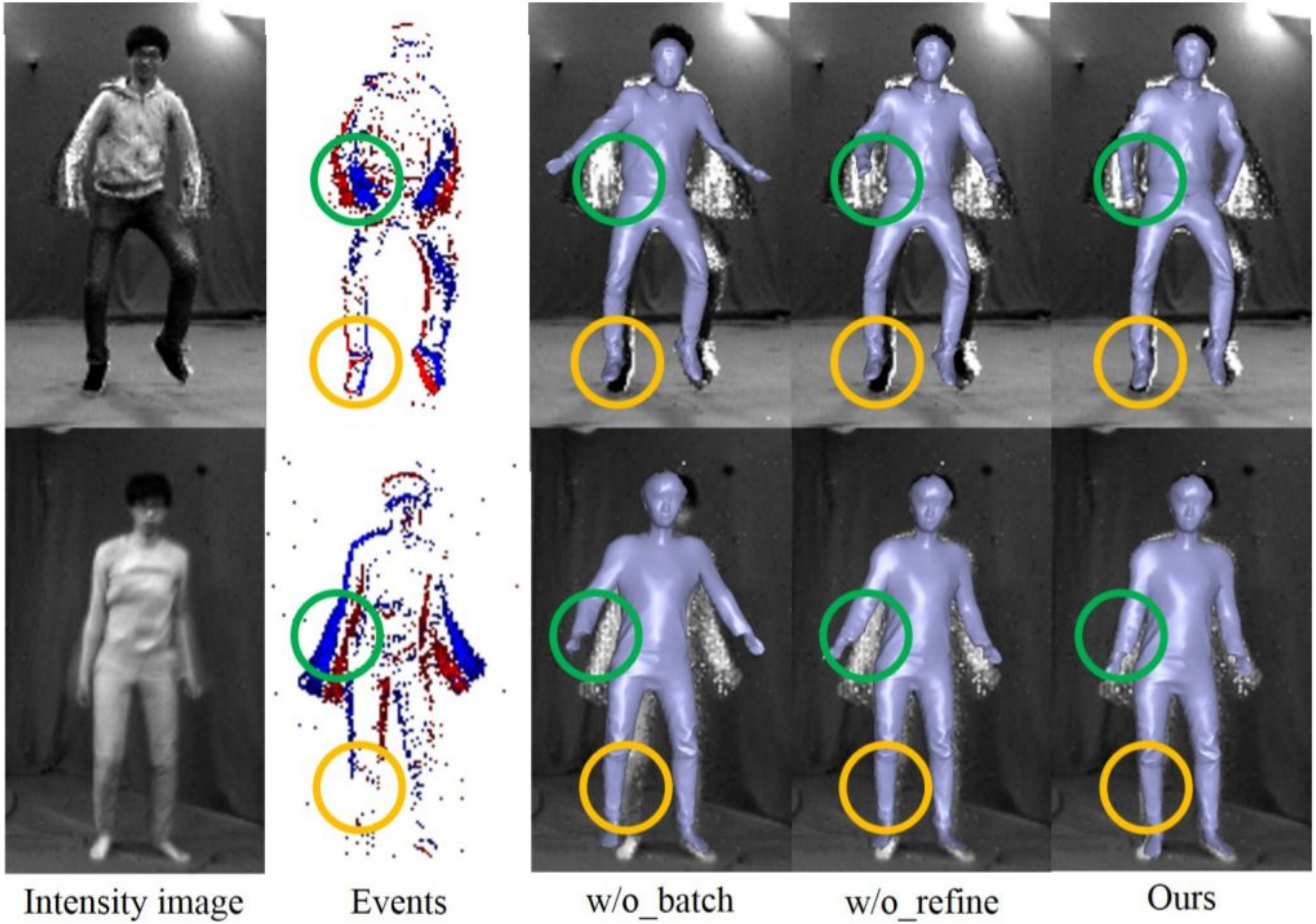
boundary vertex

target 2D position of the closet event

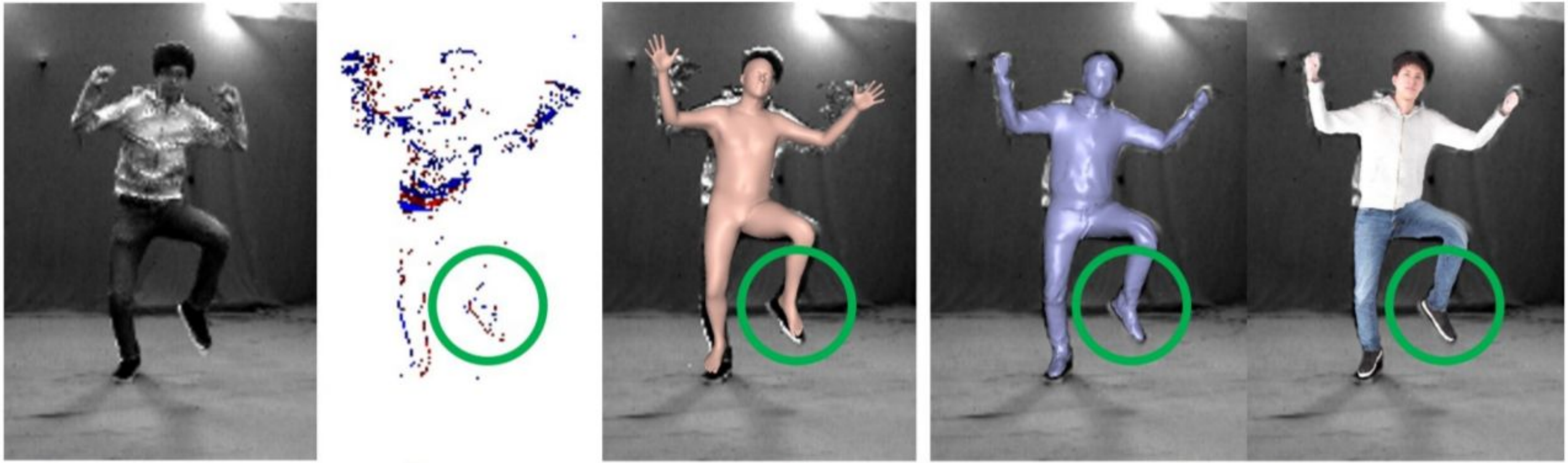
EventCap Dataset

- **12 sequences of 6 actors** performing different activities, including karate, dancing, javelin throwing, boxing, and other fast non-linear motions.
- All our sequences are captured with **a DAVIS240C event camera**, which produces an event stream and a low frame rate intensity image stream (between 7 and 25 fps) at **240×180** resolution.
- For reference, we also capture the actions with a Sony RX0 camera, which produces a high frame rate (between 250 and 1000 fps) **RGB videos at 1920 × 1080 resolution**.
- In order to perform a quantitative evaluation, **one sequence is also tracked with a multi-view markerless motion capture system** [9] at 100 fps.
- 1280X800

Results



Results



Intensity image　Events　w/o_preScan　with_preScan

Results

We run our experiments on a PC with 3.6 GHz Intel Xeon E5-1620 CPU and 16GB RAM. Our unoptimized CPU code takes **4.5 minutes for a batch (i.e. 40 frames or 40ms)**, which divides to 30 seconds for the event trajectory generation, 1.5 minutes for the batch optimization and 2.5 minutes for the pose refinement.