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

Lan Xu[1,3] Weipeng Xu[2] Vladislav Golyanik[2] Marc Habermann[2] Lu Fang[1] Christian Theobalt[2]

[1]Tsinghua-Berkeley Shenzhen Institute, Tsinghua University, China
[2]Max Planck Institute for Informatics, Saarland Informatics Campus, Germany
[3]Robotics Institute, Hong Kong University of Science and Technology, Hong Kong
lxuan@connect.ust.hk {wxu,golyanik,mhaberma,theobalt}@mpi-inf.mpg.de
fanglu@sz.tsinghua.edu.cn

Abstract

The high frame rate is a critical requirement for capturing fast human motions. In this setting, existing markerless image-based methods are constrained by the lighting requirement, the high data bandwidth and the consequent high computation overhead. In this paper, we propose EventCap — the first approach for 3D capturing of high-speed human motions using a single event camera. Our method combines model-based optimization and CNN-based human pose detection to capture high frequency motion details and to reduce the drifting in the tracking. As a result, we can capture fast motions at millisecond resolution with significantly higher data efficiency than using high frame rate videos. Experiments on our new event-based fast human motion dataset demonstrate the effectiveness and accuracy of our method, as well as its robustness to challenging lighting conditions.

1. Introduction

With the recent popularity of virtual and augmented reality (VR and AR), there has been a growing demand for reliable 3D human motion capture. As a low-cost alternative to the widely used marker- and sensor-based solutions, markerless video-based motion capture alleviates the need for intrusive body-worn motion sensors and markers. This research direction has received increased attention over the last years [13, 21, 53, 63, 69].

In this paper, we focus on markerless motion capture for high-speed movements, which is essential for many applications such as training and performance evaluation for gymnastics, sports and dancing. Capturing motion at a high frame rate leads to a very high data bandwidth and algorithm complexity for the existing methods. While the current *marker-* and *sensor-based* solutions can support more than 400 frames per second (*fps*) [62, 65, 43], the literature on *markerless* high frame rate motion capture is sparse.

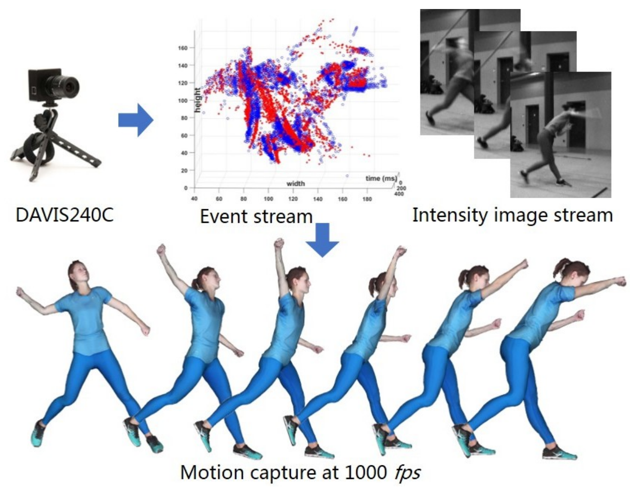


Figure 1: We present the first monocular event-based 3D human motion capture approach. Given the event stream and the low frame rate intensity image stream from a single event camera, our goal is to track the high-speed human motion at 1000 frames per second.

Several recent works [30, 72] revealed the importance of the high frame rate camera systems for tracking fast motions. However, they still suffer from the aforementioned fundamental problem — the high frame rate leads to excessive amounts of raw data and large bandwidth requirement for data processing (*e.g.,* capturing RGB stream of VGA resolution at 1000 *fps* from a single view for one minute yields 51.5GB of data). Moreover, both methods [30, 72] assume 1) well-lit scenarios for compensating the short exposure time at high frame rate, and 2) indoor capture due to the limitation of the IR-based depth sensor.

In this paper, we propose a rescue to the problems outlined above by using an event camera. Such bio-inspired dynamic vision sensors [32] asynchronously measure per-pixel intensity changes and have multiple advantages over conventional cameras, including high temporal resolution, high dynamic range, low power consumption and low data bandwidth. These properties potentially allow capturing very fast motions with significantly higher data efficiency in general lighting conditions. Nevertheless, using the event camera for motion capture is challenging. First, the high temporal resolution leads to very sparse measurements (events) in each frame interval, since the inter-frame intensity changes are subtle. The resulting low signal-to-noise ratio (SNR) makes it difficult to track the motion robustly. Second, since the event stream only encodes temporal intensity changes, it is difficult to initialize the tracking and prevent drifting. A naïve solution is to reconstruct images at a high frame rate by accumulating the events and apply existing methods on the reconstructed images. Such a policy makes the data dense again, and the temporal information encoded in the events is lost.

To tackle these challenges, we propose *EventCap* – the first monocular event-based 3D human motion capture approach (see Fig. 1 for an overview). More specifically, we design a hybrid and asynchronous motion capture algorithm that leverages the event stream and the low frame rate intensity image stream from the event camera in a joint optimization framework. Our method consists of three stages: First, we track the events in 2D space in an asynchronous manner and reconstruct the continuous spatio-temporal event trajectories between each adjacent intensity image frames. By evenly slicing the continuous event trajectories, we achieve 2D event tracking at the desired high frame rate. Second, we estimate the 3D motion of the human actor using a batch-based optimization algorithm. To tackle drifting due to the accumulation of tracking errors and depth ambiguities inherent to the monocular setting, our batch-based optimization leverages not only the tracked event trajectories but also the CNN-based 2D and 3D pose estimation from the intensity images. Finally, we refine the captured high-speed motion based on the boundary information obtained from the asynchronous event stream. To summarise, the **main contributions** of this paper include:

- We propose the first monocular approach for event camera-based 3D human motion capture.
- To tackle the challenges of low signal-to-noise ratio (SNR), drifting and the difficulty in initialization, we propose a novel hybrid asynchronous batch-based optimization algorithm.
- We propose an evaluation dataset for event camera-based fast human motion capture and provide high-quality motion capture results at 1000 *fps*. The dataset will be publicly available.

2. Related Work

3D Human Motion Capture. Marker-based multi-view motion capture studios are widely used in both industry and academia [65, 62, 43], which can capture fast motions at high frame rate (*e.g.,* 960 *fps* [43]). Those systems are usually costly, and it is quite intrusive for the users to wear the marker suites. Markerless multi-camera motion capture algorithms overcome these problems [5, 57, 67, 66, 22, 16, 50, 51, 53, 25, 68]. Recent work [2, 6, 14, 46, 47, 41, 52] even demonstrates robust out-of-studio motion capture. Although the cost is drastically reduced, synchronizing and calibrating multi-camera systems is still cumbersome. Furthermore, when capturing fast motion at high frame rate [30], a large amount of data from multiple cameras becomes a bottleneck not only for the computation but also for data processing and storage.

The availability of commodity depth cameras enabled low-cost motion capture without complicated multi-view setups [49, 3, 64, 71, 19]. To capture fast motions, Yuan *et al.* [72] combine a high frame rate action camera with a commodity 30 *fps* RGB-D camera, resulting in a synthetic depth camera of 240 *fps*. However, the active IR-based cameras are unsuitable for outdoor capture, and their high power consumption limits the mobile application.

Recently, purely RGB-based monocular 3D human pose estimation methods have been proposed with the advent of deep neural networks [23, 48, 11, 60, 29]. These methods either regress the root-relative 3D positions of body joints from single images [31, 55, 73, 34, 56, 40, 35], or lift 2D detection to 3D [4, 74, 10, 70, 24]. The 3D positional representation used in those works is not suitable for animating 3D virtual characters. To solve this problem, recent works regress joint angles directly from the images [26, 28, 38, 42, 54]. In theory, these methods can be applied directly on high frame rate video for fast motion capture. However, in practice, the tracking error is typically larger than the inter-frame movements, which leads to the loss of fine-level motion details. Methods combining data-driven 3D pose estimation and image-guided registration alleviate this problem and can achieve higher accuracy [69, 20]. However, data redundancy is still an issue. Furthermore, when capturing a high frame rate RGB video, the scene has to be well-lit, since the exposure time cannot be longer than the frame interval. Following [69], we combine data-driven method with batch optimization. Differently, instead of using high frame rate RGB video, we leverage the event stream and the low frame rate intensity image stream from an event camera. Compared to RGB-based methods, our approach is more data-efficient and works well in a broader range of lighting conditions.

Tracking with Event Cameras. Event cameras are causing a paradigm shift in computer vision, due to their high dynamic range, absence of motion blur and low power consumption. For a detailed survey of the event-based vision applications, we refer to [17]. Event-based object tracking is the most closely related to our approach.

The specific characteristics of the event camera make it very suitable for tracking fast moving objects. Most of the related works focus on tracking 2D objects like known 2D

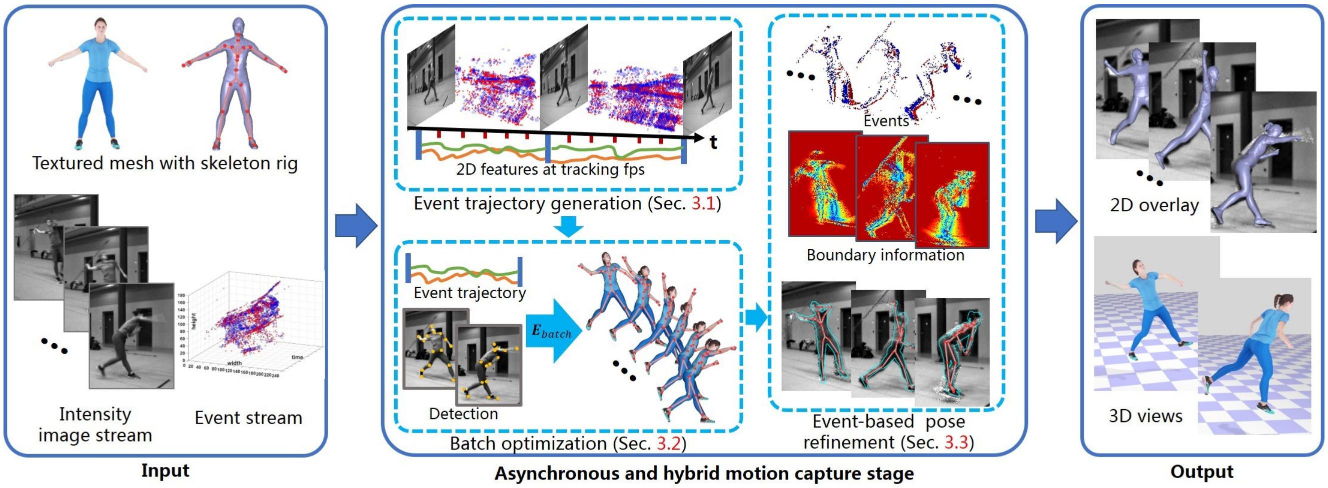


Figure 2: The pipeline of EventCap for accurate 3D human motion capture at a high frame rate. Assuming the hybrid input from a single event camera and a personalized actor rig, we first generate asynchronous event trajectories (Sec. 3.1). Then, the temporally coherent per-batch motion is recovered based on both the event trajectories and human pose detections (Sec. 3.2). Finally, we perform event-based pose refinement (Sec. 3.3).

templates [37, 36], corners [61] and lines [15]. Piatkowska *et al.* [44] propose a technique for multi-person bounding box tracking from a stereo event camera. Valeiras *et al.* [59] track complex objects like human faces with a set of Gaussian trackers connected with simulated springs.

The first 3D tracking method was proposed in [45], which estimates 3D pose estimation of rigid objects. Starting from a known object shape in a known pose, their method incrementally updates the pose by relating events to the closest visible object edges. Recently, Calabrese *et al.* [7] provide the first event-based 3D human motion capture method based on multiple event cameras. A neural network is trained to detect 2D human body joints using the event stream from each view. Then, the 3D body pose is estimated through triangulation. In their method, the events are accumulated over time, forming image frames as input to the network. Therefore, the asynchronous and high temporal resolution natures of the event camera are undermined, which prevents the method from being used for high frame rate motion capture.

3. EventCap Method

Our goal in this paper is to capture high-speed human motion in 3D using a single event camera. In order to faithfully capture the fine-level details in the fast motion, a high temporal resolution is necessary. Here, we aim at a tracking frame rate of 1000 *fps*.

Fig. 2 provides an overview of EventCap. Our method relies on a pre-processing step to reconstruct a template mesh of the actor. During tracking, we optimize the skeleton parameters of the template to match the observation of a single event camera, including the event stream and the low frame rate intensity image stream. Our tracking algorithm consists of three stages: First, we generate sparse event trajectories between two adjacent intensity images, which extract the asynchronous spatio-temporal information from the event stream (Sec. 3.1). Then, a batch optimization scheme is performed to optimize the skeletal motion at 1000 *fps* using the event trajectories and the CNN-based body joint detection from the intensity image stream (Sec. 3.2). Finally, we refine the captured skeletal motion based on the boundary information obtained from the asynchronous event stream (Sec. 3.3).

Template Mesh Acquisition. We use a 3D body scanners [58] to generate the template mesh of the actor. To rig the template mesh with a parametric skeleton, we fit the Skinned Multi-Person Linear Model (SMPL)[33] to the template mesh by optimizing the body shape and pose parameters, and then transfer the SMPL skinning weights to our scanned mesh. One can also use image-based human shape estimation algorithms, e.g. [26], to obtain a SMPL mesh as the template mesh, if the 3D scanner is not available. A comparison of these two methods is provided in Sec. 4.1. To resemble the anatomic constraints of body joints, we reduce the degrees of freedom of the SMPL skeleton. Our skeleton parameter set $\mathbf{S} = [\boldsymbol{\theta}, \mathbf{R}, \mathbf{t}]$ includes the joint angles $\boldsymbol{\theta} \in \mathbb{R}^{27}$ of the N_J joints of the skeleton, the global rotation $\mathbf{R} \in \mathbb{R}^3$ and translation $\mathbf{t} \in \mathbb{R}^3$ of the root.

Event Camera Model. Event cameras are bio-inspired sensors that measure the changes of logarithmic brightness $\mathcal{L}(u, t)$ independently at each pixel and provide an asynchronous event stream at microsecond resolution. An event $e_i = (u_i, t_i, \rho_i)$ is triggered at pixel u_i at time t_i when the logarithmic brightness change reaches a threshold: $\mathcal{L}(u_i, t_i) - \mathcal{L}(u_i, t_p) = p_i C$, where t_p is the timestamp of the last event occurred at u_i, $p_i \in \{-1, 1\}$ is the event polarity corresponding to the threshold $\pm C$. Besides the event stream, the camera also produces an intensity image stream at a lower frame rate, which can be expressed as an average of the latent images during the exposure time:

$$\mathcal{I}(k) = \frac{1}{T}\int_{t_k - T/2}^{t_k + T/2} \exp(\mathcal{L}(t))dt, \quad (1)$$

where t_k is the central timestamp of the k-th intensity image and T is the exposure time. Note that $\mathcal{I}(k)$ can suffer from severe motion blur due to high-speed motions.

3.1. Asynchronous Event Trajectory Generation

A single event does not carry any structural information and therefore tracking based on isolated events is not robust. To extract the spatio-temporal information from the event stream, in the time interval $[t_k, t_{k+1}]$ (denoted as the k-th batch) between adjacent intensity images $\mathcal{I}(k)$ and $\mathcal{I}(k+1)$, we use [18] to track the photometric 2D features in an asynchronous manner, resulting in the sparse event trajectories $\{\mathcal{T}(h)\}$. Here, $h \in [1, H]$ denotes the temporal 2D pixel locations of all the H photometric features in the current batch, which are further utilized to obtain correspondences to recover high-frequency motion details.

Intensity Image Sharpening. Note that [18] relies on sharp intensity images for gradient calculation. However, the intensity images suffer from severe motion blur due to the fast motion. Thus, we first adopt the event-based double integral (EDI) model [39] to sharpen the images $\mathcal{I}(k)$ and $\mathcal{I}(k+1)$. A logarithmic latent image $\mathcal{L}(t)$ can be formulated as $\mathcal{L}(t) = \mathcal{L}(t_k) + \mathcal{E}(t)$, where $\mathcal{E}(t) = \int_{t_k}^{t} p_i(s)C\delta(s)ds$ denotes continuous event accumulation. By aggregating the latent image $\mathcal{I}(k)$ (see Eq. (1)) and the logarithmic intensity changes, we follow [39] obtain the sharpened image:

$$\mathcal{L}(t_k) = \log\left(\mathcal{I}(k)\right) - \log\left(\frac{1}{T}\int_{t_k - T/2}^{t_k + T/2} \exp\left(\mathcal{E}(t)\right)dt\right). \quad (2)$$

We extract 2D features from the sharpened images $\mathcal{L}(t_k)$ and $\mathcal{L}(t_{k+1})$ instead of the original blurry images.

Forward and Backward Alignment. The feature tracking can drift over time. To reduce the tracking drifting, we apply the feature tracking method both forward from $\mathcal{L}(t_k)$ and backward from $\mathcal{L}(t_{k+1})$. As illustrated in Fig. 3, the bidirectional tracking results are stitched by associating the closest backward feature position to each forward feature position at the central timestamp $(t_k + t_{k+1})/2$. The stitching is not applied if the 2D distance between the two associated locations is farther than a pre-defined threshold (four pixels). For the h-th stitched trajectory, we fit a B-spline curve to its discretely tracked 2D pixel locations in a batch and calculate a continuous event feature trajectory $\mathcal{T}(h)$.

Trajectory Slicing. In order to achieve motion capture at the desired tracking frame rate, e.g. 1000 *fps*, we evenly slice the continuous event trajectory $\mathcal{T}(h)$ at each millisecond time stamp (see Fig. 3). Since we perform tracking on each batch independently, for simplification we omit the subscript k and let $0, 1, ..., N$ denote the indexes of all the tracking frames for the current batch, where N equals to the desired tracking frame rate divided by the frame rate of the intensity image stream. Thus, the intensity images $\mathcal{I}(k)$ and $\mathcal{I}(k+1)$ are denoted as $\mathcal{I}_0$ and $\mathcal{I}_N$ for short, and the corresponding latent images as $\mathcal{L}_0$ and $\mathcal{L}_N$.

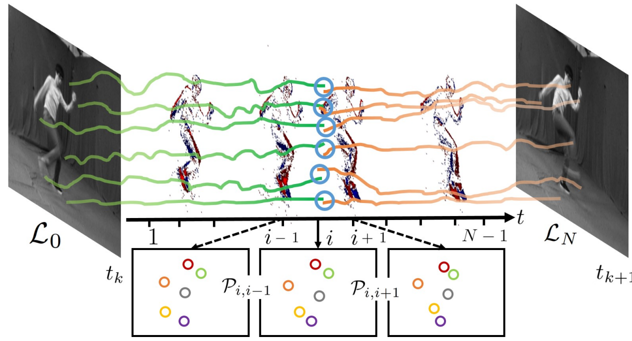


Figure 3: Illustration of asynchronous event trajectories between two adjacent intensity images. The green and orange curves represent the forward and backward event trajectories of exemplary photometric features. The blue circles denote alignment operation. The color-coded circles below indicate the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

3.2. Hybrid Pose Batch Optimization

Next, we jointly optimize all the skeleton poses $\mathcal{S} = \{\mathbf{S}_f\}, f \in [0, N]$ for all the tracking frames in a batch. Our optimization leverages the hybrid input modality from the event camera. That is, we leverage not only the event feature correspondences obtained in Sec. 3.1, but also the CNN-based 2D and 3D pose estimates to tackle the drifting due to the accumulation of tracking errors and the inherent depth ambiguities of the monocular setting. We phrase the pose estimation across a batch as a constrained optimization problem:

$$\begin{aligned} &\mathcal{S}^* = \arg\min_{\mathcal{S}} \boldsymbol{E}_{\text{batch}}(\mathcal{S}) \\ &\text{s.t.} \quad \boldsymbol{\theta}_{min} \le \boldsymbol{\theta}_f \le \boldsymbol{\theta}_{max}, \quad \forall f \in [0, N], \end{aligned} \quad (3)$$

where $\boldsymbol{\theta}_{min}$ and $\boldsymbol{\theta}_{max}$ are the pre-defined lower and upper bounds of physically plausible joint angles to prevent unnatural poses. Our per-batch objective energy functional consists of four terms:

$$\begin{aligned} \boldsymbol{E}_{\text{batch}}(\mathcal{S}) =& \lambda_{\text{cor}}\boldsymbol{E}_{\text{cor}} + \lambda_{\text{2D}}\boldsymbol{E}_{\text{2D}} + \\ & \lambda_{\text{3D}}\boldsymbol{E}_{\text{3D}} + \lambda_{\text{temp}}\boldsymbol{E}_{\text{temp}}. \end{aligned} \quad (4)$$

Event Correspondence Term. The event correspondence term exploits the asynchronous spatio-temporal motion information encoded in the event stream. To this end, for the i-th tracking frame in a batch, we first extract the event correspondences from the sliced trajectories on two adjacent frames $i-1$ and $i+1$, as shown in Fig. 3. This forms two sets of event correspondences $\mathcal{P}_{i,i-1}$ and $\mathcal{P}_{i,i+1}$, where

$\mathcal{P}_{i,*} = \{(p_{i,h}, p_{*,h})\}, h \in [1, H]$. The term encourages the 2D projection of the template meshes to match the two sets of correspondences:

$$\boldsymbol{E}_{\text{cor}}(\mathcal{S}) = \sum_{i=1}^{N-1} \sum_{j \in \{i-1, i+1\}} \sum_{h=1}^{H} \tau(p_{i,h}) \| \pi(v_{i,h}(\mathbf{S}_j)) - p_{j,h} \|_2^2, \tag{5}$$

where $\tau(p_{i,h})$ is the indicator which equals to 1 only if the 2D pixel $p_{i,h}$ corresponds to a valid vertex of the mesh at the i-th tracking frame; $v_{i,h}(\mathbf{S}_j)$ is the corresponding vertex on the mesh in the skeletal pose $\mathbf{S}_j$ and $\pi \colon \mathbb{R}^3 \to \mathbb{R}^2$ is the perspective projection operator from 3D space to the 2D image plane.

2D and 3D Detection Terms. These terms encourage the posed skeleton to match the 2D and 3D body joint detection obtained by CNN from the intensity images. To this end, we apply VNect [35] and OpenPose [8] on the intensity images to estimate the 3D and 2D joint positions, denoted as $\mathbf{P}_{f,l}^{3D}$ and $\mathbf{P}_{f,l}^{2D}$, respectively, where $f \in \{0, N\}$ is the frame index, and l is the joint index. Beside the body joints, We also use the four facial landmarks from the OpenPose [8] detection to recover the face orientation. The 2D term penalizes the differences between the projection of the landmarks of our model and the 2D detection:

$$\boldsymbol{E}_{\text{2D}}(\mathcal{S}) = \sum_{f \in \{0,N\}} \sum_{l=1}^{N_J+4} \| \pi(J_l(\mathbf{S}_f)) - \mathbf{P}_{f,l}^{2D} \|_2^2, \tag{6}$$

where $J_l(\cdot)$ returns the 3D position of the l-th joint or face marker using the kinematic skeleton. Our 3D term aligns the model joints and 3D detection:

$$\boldsymbol{E}_{\text{3D}}(\mathcal{S}) = \sum_{f \in \{0,N\}} \sum_{l=1}^{N_J} \| J_l(\mathbf{S}_f) - (\mathbf{P}_{f,l}^{3D} + \mathbf{t}') \|_2^2, \tag{7}$$

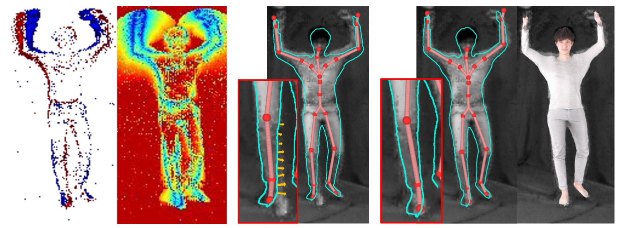
where $\mathbf{t}' \in \mathbb{R}^3$ is an auxiliary variable that transforms $\mathbf{P}_{f,l}^{3D}$ from the root-centred to the global coordinate system [69].

Temporal Stabilization Term. Since only the moving body parts can trigger events, so far, the non-moving body parts are not constrained by our energy function. Therefore, we introduce a temporal stabilization constraint for the non-moving body parts. This term penalizes the changes in joint positions between the current and previous tracking frames:

$$\boldsymbol{E}_{\text{temp}}(\mathcal{S}) = \sum_{i=0}^{N-1} \sum_{l=1}^{N_J} \phi(l) \| J_l(\mathbf{S}_i) - J_l(\mathbf{S}_{i+1}) \|_2^2, \tag{8}$$

where the indicator $\phi(\cdot)$ equals to 1 if the corresponding body part is not associated with any event correspondence, and equals 0 otherwise.

Optimization. We solve the constrained optimization problem (3) using the Levenberg-Marquardt (LM) algorithm of *ceres* [1]. For initialization, we minimize the 2D and 3D joint detection terms $\boldsymbol{E}_{\text{2D}} + \boldsymbol{E}_{\text{3D}}$ to obtain the initial values of $\mathbf{S}_0$ and $\mathbf{S}_N$, and then linearly interpolate $\mathbf{S}_0$ and $\mathbf{S}_N$ to obtain the initial values of all the tracking frames $\{\mathbf{S}_f\}$ in the current batch proportional to their timestamps.



(a) (b) (c)

Figure 4: Event-based pose refinement. (a) Polarities and color-coded normalized distance map ranging from 0 (blue) to 1 (red). (b, c) The skeleton overlapped with the latent image before and after the refinement. Yellow arrows indicate the refined boundaries and exemplary 2D correspondences.

3.3. Event-Based Pose Refinement

Most of the events are triggered by the moving edges in the image plane, which have a strong correlation with the actor's silhouette. Based on this finding, we refine our skeleton pose estimation in an Iterative Closest Point (ICP) [12] manner. In each ICP iteration, we first search for the closest event for each boundary pixel of the projected mesh. Then, we refine the pose $\mathbf{S}_f$ by solving the non-linear least squares optimization problem:

$$\boldsymbol{E}_{\text{refine}}(\mathbf{S}_f) = \lambda_{\text{sil}} \boldsymbol{E}_{\text{sil}}(\mathbf{S}_f) + \lambda_{\text{stab}} \boldsymbol{E}_{\text{stab}}(\mathbf{S}_f). \tag{9}$$

Here, we enforce the refined pose to stay close to its initial position using the following stability term:

$$\boldsymbol{E}_{\text{stab}}(\mathbf{S}_f) = \sum_{l=1}^{N_J} \| J_l(\mathbf{S}_f) - J_i(\hat{\mathbf{S}}_f) \|_2^2, \tag{10}$$

where $\hat{\mathbf{S}}_f$ is the skeleton pose after batch optimization (Sec. 3.2). The data term $\boldsymbol{E}_{\text{sil}}$ relies on the closest event search, which we will describe later. Let s_b and v_b denote the b-th boundary pixel and its corresponding 3D position on the mesh based on barycentric coordinates. For each s_b, let u_b denote the corresponding target 2D position of the closest event. Then $\boldsymbol{E}_{\text{sil}}$ measures the 2D point-to-plane misalignment of the correspondences:

$$\boldsymbol{E}_{\text{sil}}(\mathbf{S}_f) = \sum_{b \in \mathcal{B}} \| \mathbf{n}_b^{\mathbf{T}} \big(\pi(v_b(\mathbf{S}_f) - u_b) \big) \|_2^2, \tag{11}$$

where $\mathcal{B}$ is the boundary set of the projected mesh and $\mathbf{n}_b \in \mathbb{R}^2$ is the 2D normal vector corresponding to s_b.

Closest Event Search. Now we describe how to obtain the closest event for each boundary pixel s_b. The criterion for

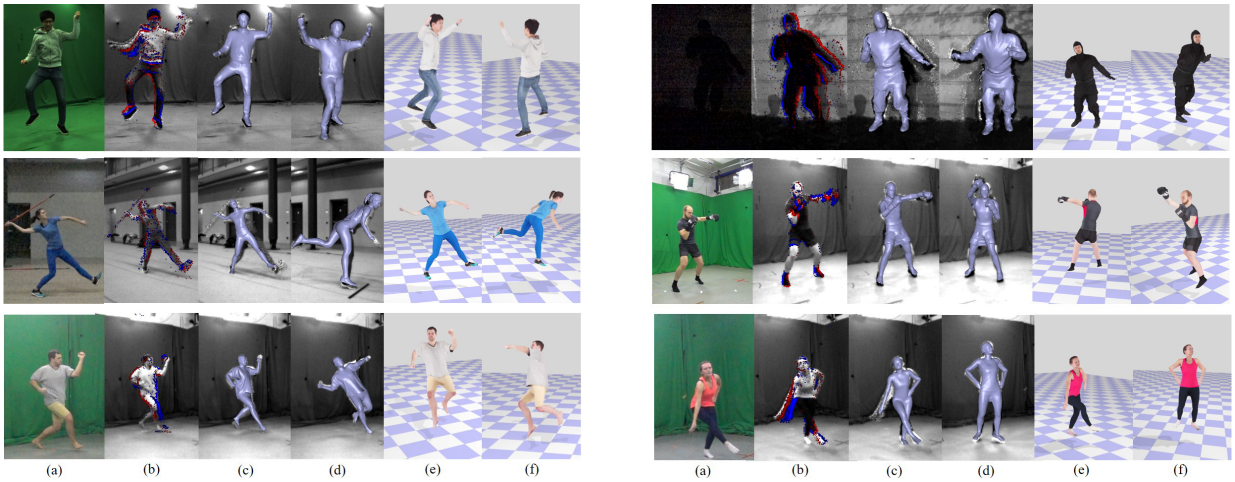


Figure 5: Qualitative results of EventCap on some sequences from our benchmark dataset, including "wave", "ninja", "javelin", "boxing", "karate" and "dancing" from the upper left to lower right. (a) The reference RGB image (not used for tracking); (b) Intensity images and the accumulated events; (c,d) Motion capture results overlaid on the reconstructed latent images; (e,f) Results rendered in 3D views.

the closest event searching is based on the temporal and spatial distance between s_b and each recent event $e = (u, t, \rho)$:

$$\mathcal{D}(s_b, e) = \lambda_{dist} \| \frac{t_f - t}{t_N - t_0} \|_2^2 + \| s_b - u \|_2^2, \tag{12}$$

where t_f is the timestamp of the current tracking frame, λ_{dist} balances the weights of temporal and spatial distances, and $t_N - t_0$ equals to the time duration of a batch. We then solve the following local searching problem to obtain the closest event for each boundary pixel s_b:

$$e_b = \arg\min_{e \in \mathcal{P}} \mathcal{D}(s_b, e). \tag{13}$$

Here, $\mathcal{P}$ is the collection of events, which happen within a local 8×8 spatial patch centred at s_b and within the batch-duration-sized temporal window centered at t_f. The position u_b of the closest event e_b is further utilized in Eq. (11).

Optimization. During the event-based refinement, we initialize $\mathbf{S}_f$ with the batch-based estimates and typically perform four ICP iterations. In each iteration, the energy in Eq. (9) is solved using the LM method provided by *ceres* [1]. As shown in Figs. 4(b) and 4(c), our iterative refinement based on the event stream improves the pose estimates.

4. Experimental Results

In this section, we evaluate our EventCap method on a variety of challenging scenarios. 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. In all experiments, we use the following empirically determined parameters: $\lambda_{3D} = 1$, $\lambda_{2D} = 200$, $\lambda_{adj} = 50$, $\lambda_{temp} = 80$, $\lambda_{sil} = 1.0$, $\lambda_{stab} = 5.0$, and $\lambda_{dist} = 4.0$.

EventCap Dataset. To evaluate our method, we propose a new benchmark dataset for monocular event-based 3D motion capture, consisting of 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*.

Fig. 5 shows several example frames of our EventCap results on the proposed dataset. For qualitative evaluation, we reconstruct the latent images at 1000 *fps* from the event stream using the method of [39]. We can see in Fig. 5 that our results can be precisely overlaid on the latent images (c-d), and that our reconstructed poses are plausible in 3D (e-f). The complete motion capture results are provided in our supplementary video. From the 1000 *fps* motion capture results, we can see that our method can accurately capture the high-frequency temporal motion details, which cannot be achieved by using standard low *fps* videos. Benefiting from the high dynamic range of the event camera, our method can handle various lighting conditions, even many extreme cases, such as the actor in black ninja suite captured outdoor in the night (see Fig. 5 top right). While it is already difficult for human eyes to spot the actor in the reference images, our method still yields plausible results.

4.1. Ablation Study

In this section, we evaluate the individual components of EventCap. Let *w/o_batch* and *w/o_refine* denote the

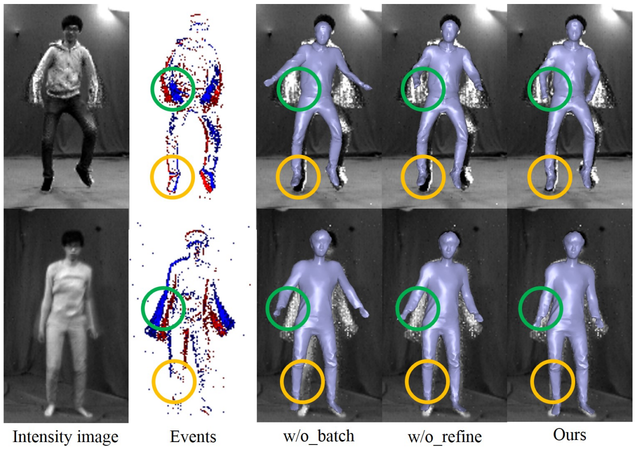


Figure 6: Ablation study for the EventCap components. In the second column, polarity events are accumulated between the time duration from the previous to the current tracking frames. Results of the full pipeline overlay more accurately with the latent images.

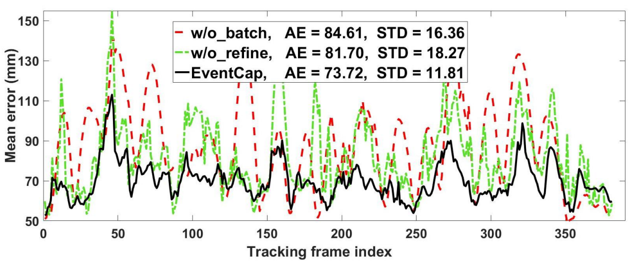


Figure 7: Ablation study: the average per-joint 3D error demonstrates the effectiveness of each algorithmic component of EventCap. Our full pipeline consistently achieves the lowest error.

variations of our method without the batch optimization (Sec. 3.2) and the pose refinement (Sec. 3.3), respectively. For *w/o_batch*, we optimize the pose for each tracking frame $t \in [0, N]$ independently. The skeleton poses $\mathcal{S}_t$ are initialized with linear interpolation of the poses obtained from the two adjacent intensity images $\mathcal{I}_0$ and $\mathcal{I}_N$. As shown in Fig. 6, the results of our full pipeline are overlaid on the reconstructed latent images more accurately than those of *w/o_batch* and *w/o_refine* (the full sequence can be found in our supplementary video). We can see that, benefiting from the integration of CNN-based 2D and 3D pose estimation and the event trajectories, our batch optimization significantly improves the accuracy and alleviated the drifting problem. Our pose refinement further corrects the remaining misalignment, resulting in a better overlay on the reconstructed latent images.

This is further evidenced by our quantitative evaluation in Fig. 7, where we obtain ground truth 3D joint positions using a multi-view markerless motion capture software [9]. Then, we compute the average per-joint error (AE) and the standard deviation (STD) of AE on every 10th tracking frame, because our tracking frame rate is 1000 *fps* while the maximum capture frame rate of [9] is 100 *fps*. Following [69], to factor out the global pose, we perform Procrustes analysis to rigidly align our results to the ground truth. Fig. 7 shows our full pipeline consistently outperforms the baselines on all frames, yielding both the lowest AE and the lowest STD. This not only highlights the contribution of each algorithmic component but also illustrates that our approach captures more high-frequency motion details in fast motions and achieves temporally more coherent results.

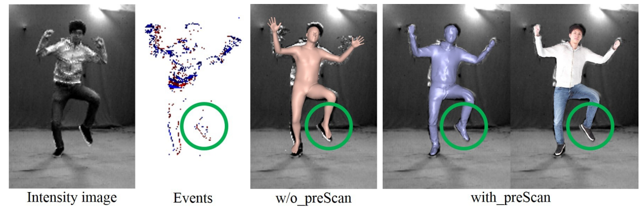


Figure 8: Influence of the template mesh accuracy. Our results using a prescanned template and using SMPL mesh are comparable, while the more accurate 3D scanned template improves the overlay on the latent images.

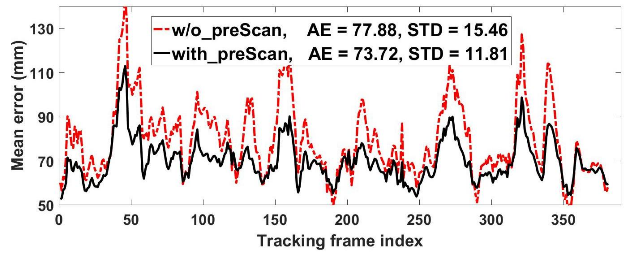


Figure 9: Quantitative analysis of the template mesh. The more accurate template improves the tracking accuracy in terms of average per-joint error.

We further evaluate the influence of the template mesh accuracy. To this end, we compare the result using SMPL mesh from image-based body shape estimation [26] (denoted as *w/o_preScan*) against that using more accurate 3D scanned mesh (denoted as *with_preScan*). As shown in Fig. 8, the two methods yield comparable pose estimation results, while the 3D scanned mesh helps in terms of an image overlay since the SMPL mesh cannot model the clothes. Quantitatively, the method using 3D scanned mesh achieves a lower AE (73.72mm vs 77.88mm) as shown in Fig. 9.

4.2. Comparison to Baselines

To the best of our knowledge, our approach is the first monocular event-based 3D motion capture method. Therefore, we compare to existing monocular RGB-based approaches, HMR [26] and MonoPerfCap [69], which are most closely related to our approach. For a fair comparison, we first reconstruct the latent intensity images at 1000 *fps* using [39]. Then, we apply HMR [27] and MonoPerfCap[1] [69] on all latent images, denoted as *HMR_all* and *Mono_all*, respectively. We further apply MonoPerfCap [69] and HMR [27] only on the raw intensity images of low frame rate and linearly upsample

[1]Only the pose optimization stage of MonoPerfCap is used, as their segmentation does not work well on the reconstructed latent images.

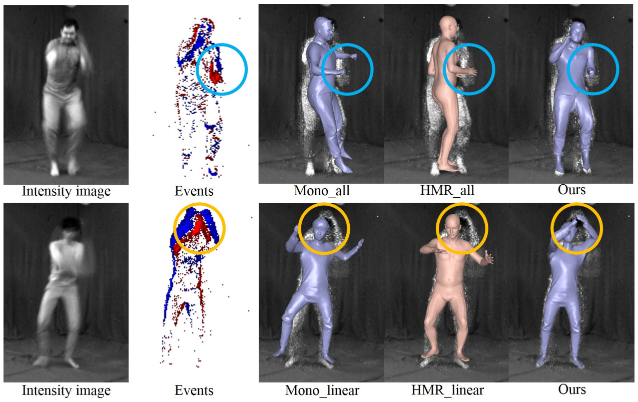


Figure 10: Qualitative comparison. Note that the polarity events are accumulated between the time duration from the previous to the current tracking frames. Our results overlay better with the latent images than the results of other methods.

	AE_all (mm)	AE_raw (mm)	AE_nonRaw (mm)	Size_sec (MB)
Mono_linear	88.6±17.3	89.2±19.7	88.5±16.8	**1.83**
Mono_all	98.4±22.8	90.2±21.4	99.8±23.0	58.59
HMR_linear	105.3±19.2	104.3±20.6	105.4±19.1	**1.83**
HMR_all	110.3±20.4	105.5±19.5	105.4±20.4	58.59
Ours	**73.7±11.8**	**75.2±13.3**	**73.5±11.3**	2.02

Table 1: Quantitative comparison of several methods in terms of tracking accuracy and data throughput.

the skeleton poses to 1000 *fps*, denoted as *Mono_linear* and *HMR_linear*, respectively. As shown in Fig. 10, both *HMR_all* and *Mono_all* suffer from inferior tracking results due to the accumulated error of the reconstructed latent images, while *Mono_linear* and *HMR_linear* fail to track the high-frequency motions. In contrast, our method achieves significantly better tracking results and more accurate overlay with the latent images. For quantitative comparison, we make use of the sequence with available ground truth poses (see Sec. 4.1). In Table 1, we report the mean AE of 1) all tracking frames (**AE_all**), 2) only the raw intensity frames (**AE_raw**), and 3) only the reconstructed latent image frames (**AE_nonRaw**). We also report the data throughput as the size of processed raw data per-second (**Size_sec**) for different methods. These quantitative results illustrate that our method achieves the highest tracking accuracy in our high frame rate setting. Furthermore, our method uses only 3.4% of the data bandwidth required in the high frame rate images setting (*HMR_all* and *Mono_all*), or only 10% higher compared to the low frame rate upsampling setting (*Mono_linear* and *HMR_linear*).

Furthermore, we apply MonoPerfCap [69] and HMR [27] to the high frame rate reference RGB images directly, denoted as *HMR_refer* and *Mono_refer*, respectively. Due to the difference of image resolution between the reference and the event cameras, for a fair comparison, we downsample the reference images into the same resolution of the intensity image from the event camera. As shown in Fig. 11,

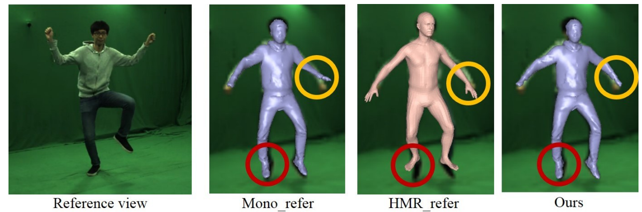


Figure 11: Qualitative comparison. Our results yield similar and even better overlay with the reference image, compared to results of *Mono_refer* and *HMR_refer*, respectively.

	AE_all (mm)	STD (mm)	Size_sec (MB)
Mono_refer	76.5	13.4	58.59
HMR_refer	83.5	17.8	58.59
Ours	**73.7**	**11.8**	**2.02**

Table 2: Quantitative comparison against *Mono_refer* and *HMR_refer* in terms of tracking accuracy and data throughput.

our method achieves similar overlap to the reference image without using the high frame rate reference images. The corresponding AE and STD for all the tracking frames, as well as the *Size_sec* are reported in Table 2. Note that our method relies upon only 3.4% of the data bandwidth of the reference image-based methods, and even achieves better tracking accuracy compared to *Mono_refer* and *HMR_refer*.

5. Discussion and Conclusion

Limitations. Our approach shares a few common limitations with other monocular motion capture methods, such as being not able to handle topology change and severe (self-)occlusion. Besides, our approach requires a stable capture background and cannot handle the challenging scenarios like sudden lighting changes or moving the camera, which will lead to a large amount of noise events. In future work, we intend to investigate handling large occlusions and topological changes and improve the runtime performance.

Conclusion. We have presented the first approach for markerless 3D human motion capture using a single event camera and a new dataset with high-speed human motions. Our batch optimization makes full usage of the hybrid image and event streams, while the captured motion is further refined with a new event-based pose refinement approach. Our experimental results demonstrate the effectiveness and robustness of EventCap in capturing fast human motions in various scenarios. We believe that it is a significant step to enable markerless capturing of high-speed human motions, with many potential applications in AR and VR, gaming, entertainment and performance evaluation for gymnastics, sports and dancing.

Acknowledgments. We thank Ole Burghardt, Franziska Müller and Janine Vieweg for recording the sequences. This work is supported by Natural Science Foundation of China (NSFC) under contract No. 61722209 and 6181001011 and the ERC Consolidator Grant 4DRepLy (770784).

References

[1] Sameer Agarwal, Keir Mierle, and Others. Ceres solver. http://ceres-solver.org. 5, 6

[2] Sikander Amin, Mykhaylo Andriluka, Marcus Rohrbach, and Bernt Schiele. Multi-view pictorial structures for 3D human pose estimation. In *British Machine Vision Conference (BMVC)*, 2009. 2

[3] Andreas Baak, Meinard Müller, Gaurav Bharaj, Hans-Peter Seidel, and Christian Theobalt. A data-driven approach for real-time full body pose reconstruction from a depth camera. In *International Conference on Computer Vision (ICCV)*, 2011. 2

[4] Federica "Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J." Black. Keep It SMPL: Automatic Estimation of 3D Human Pose and Shape from a Single Image. In *European Conference on Computer Vision (ECCV)*, 2016. 2

[5] Chris Bregler and Jitendra Malik. Tracking people with twists and exponential maps. In *Computer Vision and Pattern Recognition (CVPR)*, 1998. 2

[6] Magnus Burenius, Josephine Sullivan, and Stefan Carlsson. 3D pictorial structures for multiple view articulated pose estimation. In *Computer Vision and Pattern Recognition (CVPR)*, 2013. 2

[7] Enrico Calabrese, Gemma Taverni, Christopher Awai Easthope, Sophie Skriabine, Federico Corradi, Luca Longinotti, Kynan Eng, and Tobi Delbruck. DHP19: Dynamic vision sensor 3d human pose dataset. In *Computer Vision and Pattern Recognition (CVPR) Workshops*, 2019. 3

[8] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *Computer Vision and Pattern Recognition (CVPR)*, 2017. 5

[9] The Captury. http://www.thecaptury.com/. 6, 7

[10] Ching-Hang Chen and Deva Ramanan. 3d human pose estimation = 2d pose estimation + matching. In *Computer Vision and Pattern Recognition (CVPR)*, 2016. 2

[11] Wenzheng Chen, Huan Wang, Yangyan Li, Hao Su, Changhe Tu, Dani Lischinski, Daniel Cohen-Or, and Baoquan Chen. Synthesizing training images for boosting human 3D pose estimation. In *International Conference on 3D Vision (3DV)*, 2016. 2

[12] Yang Chen and Gérard Medioni. Object modelling by registration of multiple range images. *Image and Vision Computing (IVC)*, 10(3):145–155, 1992. 5

[13] Andrew J. Davison, Jonathan Deutscher, and Ian D. Reid. Markerless motion capture of complex full-body movement for character animation. In *Eurographics Workshop on Computer Animation and Simulation*, 2001. 1

[14] Ahmed Elhayek, Edilson de Aguiar, Arjun Jain, Jonathan Tompson, Leonid Pishchulin, Mykhaylo Andriluka, Chris Bregler, Bernt Schiele, and Christian Theobalt. Efficient ConvNet-based marker-less motion capture in general scenes with a low number of cameras. In *Computer Vision and Pattern Recognition (CVPR)*, 2015. 2

[15] Lukas Everding and Jrg Conradt. Low-latency line tracking using event-based dynamic vision sensors. *Frontiers in Neurorobotics*, 12:4, 2018. 3

[16] Juergen Gall, Bodo Rosenhahn, Thomas Brox, and Hans-Peter Seidel. Optimization and filtering for human motion capture. *International Journal of Computer Vision (IJCV)*, 87(1–2):75–92, 2010. 2

[17] Guillermo Gallego, Tobi Delbruck, Garrick Orchard, Chiara Bartolozzi, Brian Taba, Andrea Censi, Stefan Leutenegger, Andrew Davison, Joerg Conradt, Kostas Daniilidis, and Davide Scaramuzza. Event-based vision: A survey. *arXiv e-prints*, 2019. 2

[18] Daniel Gehrig, Henri Rebecq, Guillermo Gallego, and Davide Scaramuzza. Asynchronous, photometric feature tracking using events and frames. In *European Conference on Computer Vision (ECCV)*, 2018. 4

[19] Kaiwen Guo, Jonathan Taylor, Sean Fanello, Andrea Tagliasacchi, Mingsong Dou, Philip Davidson, Adarsh Kowdle, and Shahram Izadi. Twinfusion: High framerate non-rigid fusion through fast correspondence tracking. In *International Conference on 3D Vision (3DV)*, pages 596–605, 2018. 2

[20] Marc Habermann, Weipeng Xu, Michael Zollhöfer, Gerard Pons-Moll, and Christian Theobalt. Livecap: Real-time human performance capture from monocular video. *ACM Transactions on Graphics (TOG)*, 38(2):14:1–14:17, 2019. 2

[21] Nils Hasler, Bodo Rosenhahn, Thorsten Thormahlen, Michael Wand, Juergen Gall, and Hans-Peter Seidel. Markerless motion capture with unsynchronized moving cameras. In *Computer Vision and Pattern Recognition (CVPR)*, pages 224–231, 2009. 1

[22] Michael B. Holte, Cuong Tran, Mohan M. Trivedi, and Thomas B. Moeslund. Human pose estimation and activity recognition from multi-view videos: Comparative explorations of recent developments. *Journal of Selected Topics in Signal Processing*, 6(5):538–552, 2012. 2

[23] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6M: Large Scale Datasets and Predictive Methods for 3D Human Sensing in Natural Environments. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2014. 2

[24] Ehsan Jahangiri and Alan L Yuille. Generating multiple hypotheses for human 3d pose consistent with 2d joint detections. In *International Conference on Computer Vision (ICCV)*, 2017. 2

[25] Hanbyul Joo, Hao Liu, Lei Tan, Lin Gui, Bart Nabbe, Iain Matthews, Takeo Kanade, Shohei Nobuhara, and Yaser Sheikh. Panoptic studio: A massively multiview system for social motion capture. In *International Conference on Computer Vision (ICCV)*, 2015. 2

[26] Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Computer Vision and Pattern Regognition (CVPR)*, 2018. 2, 3, 7

[27] Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. End-to-end recovery of human shape and

pose. In *Computer Vision and Pattern Regognition (CVPR)*, 2018. 7, 8

[28] Nikos Kolotouros, Georgios Pavlakos, and Kostas Daniilidis. Convolutional mesh regression for single-image human shape reconstruction. In *Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[29] Onorina Kovalenko, Vladislav Golyanik, Jameel Malik, Ahmed Elhayek, and Didier Stricker. Structure from Articulated Motion: Accurate and Stable Monocular 3D Reconstruction without Training Data. *Sensors*, 19(20), 2019. 2

[30] Adarsh Kowdle, Christoph Rhemann, Sean Fanello, Andrea Tagliasacchi, Jonathan Taylor, Philip Davidson, Mingsong Dou, Kaiwen Guo, Cem Keskin, Sameh Khamis, David Kim, Danhang Tang, Vladimir Tankovich, Julien Valentin, and Shahram Izadi. The need 4 speed in real-time dense visual tracking. In *SIGGRAPH Asia*, pages 220:1–220:14, 2018. 1, 2

[31] Sijin Li and Antoni B Chan. 3D Human Pose Estimation from Monocular Images with Deep Convolutional Neural Network. In *Asian Conference on Computer Vision (ACCV)*, 2014. 2

[32] Patrick Lichtsteiner, Christoph Posch, and Tobi Delbruck. A 128×128 120 db $15\mu s$ latency asynchronous temporal contrast vision sensor. *IEEE Journal of Solid-State Circuits*, 43(2):566–576, 2008. 1

[33] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. In *SIGGRAPH Asia*, volume 34, pages 248:1–248:16, 2015. 3

[34] Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *International Conference on 3D Vision (3DV)*, 2017. 2

[35] Dushyant Mehta, Srinath Sridhar, Oleksandr Sotnychenko, Helge Rhodin, Mohammad Shafiei, Hans-Peter Seidel, Weipeng Xu, Dan Casas, and Christian Theobalt. Vnect: Real-time 3d human pose estimation with a single rgb camera. *ACM Transactions on Graphics (TOG)*, 36(4), 2017. 2, 5

[36] Abhishek Mishra, Rohan Ghosh, Ashish Goyal, Nitish V Thakor, and Sunil L Kukreja. Real-time robot tracking and following with neuromorphic vision sensor. In *International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, 2016. 3

[37] Zhenjiang Ni, Sio-Hoi Ieng, Christoph Posch, Stphane Rgnier, and Ryad Benosman. Visual tracking using neuromorphic asynchronous event-based cameras. *Neural computation*, 27:1–29, 02 2015. 3

[38] Mohamed Omran, Christoph Lassner, Gerard Pons-Moll, Peter V. Gehler, and Bernt Schiele. Neural body fitting: Unifying deep learning and model-based human pose and shape estimation. In *International Conference on 3D Vision (3DV)*, 2018. 2

[39] Liyuan Pan, Cedric Scheerlinck, Xin Yu, Richard Hartley, Miaomiao Liu, and Yuchao Dai. Bringing a blurry frame alive at high frame-rate with an event camera. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 4, 6, 7

[40] Georgios Pavlakos, Xiaowei Zhou, Konstantinos G Derpanis, and Kostas Daniilidis. Coarse-to-fine volumetric prediction for single-image 3D human pose. In *Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[41] Georgios Pavlakos, Xiaowei Zhou, Konstantinos G Derpanis, and Kostas Daniilidis. Harvesting multiple views for marker-less 3d human pose annotations. In *Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[42] Georgios Pavlakos, Luyang Zhu, Xiaowei Zhou, and Kostas Daniilidis. Learning to estimate 3D human pose and shape from a single color image. In *Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[43] Phasespace impulse x2e. `http://phasespace.com/x2e-motion-capture/`. Accessed: 2019-07-05. 1, 2

[44] Ewa Piatkowska, Ahmed Nabil Belbachir, Stephan Schraml, and Margrit Gelautz. Spatiotemporal multiple persons tracking using dynamic vision sensor. In *Computer Vision and Pattern Recognition (CWPR) Workshops*, pages 35–40, 2012. 3

[45] David Reverter Valeiras, Garrick Orchard, Sio-Hoi Ieng, and Ryad B. Benosman. Neuromorphic event-based 3d pose estimation. *Frontiers in Neuroscience*, 9:522, 2016. 3

[46] Helge Rhodin, Nadia Robertini, Christian Richardt, Hans-Peter Seidel, and Christian Theobalt. A versatile scene model with differentiable visibility applied to generative pose estimation. In *International Conference on Computer Vision (ICCV)*, 2015. 2

[47] Nadia Robertini, Dan Casas, Helge Rhodin, Hans-Peter Seidel, and Christian Theobalt. Model-based outdoor performance capture. In *International Conference on 3D Vision (3DV)*, 2016. 2

[48] Grégory Rogez and Cordelia Schmid. Mocap Guided Data Augmentation for 3D Pose Estimation in the Wild. In *Neural Information Processing Systems (NIPS)*, 2016. 2

[49] Jamie Shotton, Andrew Fitzgibbon, Mat Cook, Toby Sharp, Mark Finocchio, Richard Moore, Alex Kipman, and Andrew Blake. Real-time human pose recognition in parts from single depth images. In *Computer Vision and Pattern Recognition (CVPR)*, 2011. 2

[50] Leonid Sigal, Alexandru O. Bălan, and Michael J. Black. HumanEva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion. *International Journal of Computer Vision (IJCV)*, 2010. 2

[51] Leonid Sigal, Michael Isard, Horst Haussecker, and Michael J. Black. Loose-limbed people: Estimating 3D human pose and motion using non-parametric belief propagation. *International Journal of Computer Vision (IJCV)*, 98(1):15–48, 2012. 2

[52] Tomas Simon, Hanbyul Joo, Iain Matthews, and Yaser Sheikh. Hand keypoint detection in single images using multiview bootstrapping. In *Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[53] Carsten Stoll, Nils Hasler, Juergen Gall, Hans-Peter Seidel, and Christian Theobalt. Fast articulated motion tracking us-

ing a sums of Gaussians body model. In *International Conference on Computer Vision (ICCV)*, 2011. 1, 2

[54] Vince Tan, Ignas Budvytis, and Roberto Cipolla. Indirect deep structured learning for 3d human body shape and pose prediction. In *British Machine Vision Conference (BMVC)*, 2018. 2

[55] Bugra Tekin, Isinsu Katircioglu, Mathieu Salzmann, Vincent Lepetit, and Pascal Fua. Structured Prediction of 3D Human Pose with Deep Neural Networks. In *British Machine Vision Conference (BMVC)*, 2016. 2

[56] Bugra Tekin, Pablo Márquez-Neila, Mathieu Salzmann, and Pascal Fua. Fusing 2D Uncertainty and 3D Cues for Monocular Body Pose Estimation. In *International Conference on Computer Vision (ICCV)*, 2017. 2

[57] Christian Theobalt, Edilson de Aguiar, Carsten Stoll, Hans-Peter Seidel, and Sebastian Thrun. Performance capture from multi-view video. In *Image and Geometry Processing for 3-D Cinematography*, pages 127–149. Springer, 2010. 2

[58] Treedy's. https://www.treedys.com/. Accessed: 2019-07-25. 3

[59] David Reverter Valeiras, Xavier Lagorce, Xavier Clady, Chiara Bartolozzi, Sio-Hoi Ieng, and Ryad Benosman. An asynchronous neuromorphic event-driven visual part-based shape tracking. *Transactions on Neural Networks and Learning Systems (TNNLS)*, 26(12):3045–3059, 2015. 3

[60] Gül Varol, Javier Romero, Xavier Martin, Naureen Mahmood, Michael Black, Ivan Laptev, and Cordelia Schmid. Learning from synthetic humans. In *Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[61] Valentina Vasco, Arren Glover, Elias Mueggler, Davide Scaramuzza, Lorenzo Natale, and Chiara Bartolozzi. Independent motion detection with event-driven cameras. In *International Conference on Advanced Robotics (ICAR)*, pages 530–536, 2017. 3

[62] Vicon Motion Systems. https://www.vicon.com/, 2019. 1, 2

[63] Yangang Wang, Yebin Liu, Xin Tong, Qionghai Dai, and Ping Tan. Outdoor markerless motion capture with sparse handheld video cameras. *Transactions on Visualization and Computer Graphics (TVCG)*, 2017. 1

[64] Xiaolin Wei, Peizhao Zhang, and Jinxiang Chai. Accurate realtime full-body motion capture using a single depth camera. *SIGGRAPH Asia*, 31(6):188:1–12, 2012. 2

[65] Xsens Technologies B.V. https://www.xsens.com/, 2019. 1, 2

[66] Lan Xu, Wei Cheng, Kaiwen Guo, Lei Han, Yebin Liu, and Lu Fang. Flyfusion: Realtime dynamic scene reconstruction using a flying depth camera. *IEEE Transactions on Visualization and Computer Graphics*, pages 1–1, 2019. 2

[67] Lan Xu, Yebin Liu, Wei Cheng, Kaiwen Guo, Guyue Zhou, Qionghai Dai, and Lu Fang. Flycap: Markerless motion capture using multiple autonomous flying cameras. *IEEE Transactions on Visualization and Computer Graphics*, 24(8):2284–2297, Aug 2018. 2

[68] Lan Xu, Zhuo Su, Lei Han, Tao Yu, Yebin Liu, and Lu Fang. Unstructuredfusion: Realtime 4d geometry and texture reconstruction using commercialrgbd cameras. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2019. 2

[69] Weipeng Xu, Avishek Chatterjee, Michael Zollhöfer, Helge Rhodin, Dushyant Mehta, Hans-Peter Seidel, and Christian Theobalt. Monoperfcap: Human performance capture from monocular video. *ACM Transactions on Graphics (TOG)*, 37(2):27:1–27:15, 2018. 1, 2, 5, 7, 8

[70] Hashim Yasin, Umar Iqbal, Bjorn Kruger, Andreas Weber, and Juergen Gall. A Dual-Source Approach for 3D Pose Estimation from a Single Image. In *Computer Vision and Pattern Recognition (CVPR)*, 2016. 2

[71] Tao Yu, Zerong Zheng, Kaiwen Guo, Jianhui Zhao, Qionghai Dai, Hao Li, Gerard Pons-Moll, and Yebin Liu. Doublefusion: Real-time capture of human performances with inner body shapes from a single depth sensor. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2019. 2

[72] Ming-Ze Yuan, Lin Gao, Hongbo Fu, and Shihong Xia. Temporal upsampling of depth maps using a hybrid camera. *Transactions on Visualization and Computer Graphics (TVCG)*, 25(3):1591–1602, 2019. 1, 2

[73] Xingyi Zhou, Xiao Sun, Wei Zhang, Shuang Liang, and Yichen Wei. Deep Kinematic Pose Regression. In *European Conference on Computer Vision (ECCV) Workshops*, 2016. 2

[74] Xiaowei Zhou, Menglong Zhu, Spyridon Leonardos, Konstantinos G Derpanis, and Kostas Daniilidis. Sparseness Meets Deepness: 3D Human Pose Estimation from Monocular Video. In *Computer Vision and Pattern Recognition (CVPR)*, 2016. 2