View Birdification in the Crowd: Ground-Plane Localization from Perceived Movements

Mai Nishimura[1][2]
mai.nishimura@sinicx.com
Shohei Nobuhara[1]
nob@i.kyoto-u.ac.jp
Ko Nishino[1]
kon@i.kyoto-u.ac.jp

[1] Kyoto University
Kyoto, Japan
[2] OMRON SINIC X
Tokyo, Japan

Abstract

We introduce *view birdification*, the problem of recovering ground-plane movements of people in a crowd from an ego-centric video captured from an observer (*e.g.*, a person or a vehicle) also moving in the crowd. Recovered ground-plane movements would provide a sound basis for situational understanding and benefit downstream applications in computer vision and robotics. In this paper, we formulate view birdification as a geometric trajectory reconstruction problem and derive a cascaded optimization method from a Bayesian perspective. The method first estimates the observer's movement and then localizes surrounding pedestrians for each frame while taking into account the local interactions between them. We introduce three datasets by leveraging synthetic and real trajectories of people in crowds and evaluate the effectiveness of our method. The results demonstrate the accuracy of our method and set the ground for further studies of view birdification as an important but challenging visual understanding problem.

1 Introduction

We, human beings, are capable of mentally visualizing our surroundings in a third-person view. Imagine walking down a street alongside other pedestrians. Your mental model of the movements of surrounding people is not a purely two-dimensional one, but rather in 3D, albeit imperfect. It lets you guess your present location and how the geometric layout of your surroundings changes as you navigate even in a dense crowd where everything around you is dynamic. Endowing such 3D spatial perception with computers remains elusive. Despite the significant progress in computational 3D and motion perception, structure from motion, and SLAM, reconstructing the 3D geometry and motion in an "everywhere-dynamic" scene is still challenging. Past works fundamentally rely on the visibility of textured background to extract static keypoints at all times, so that the ego-motion can be estimated regardless of surrounding movements.

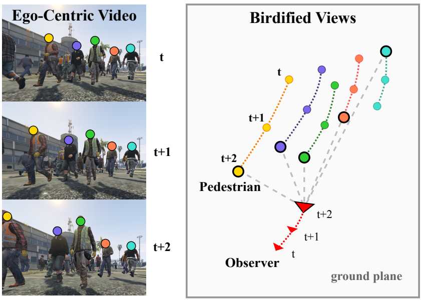

Figure 1: View birdification aims to recover the ground-plane trajectory of people in a crowd from an ego-centric video captured by a dynamic observer without static references.

In this paper, we ask a fundamental question of 3D computational perception. Can we recover our own and surrounding movements on the ground plane from their perceived movements in the image plane of our view, when we can't easily discern our ego-motion? That is, given 2D ego-centric views from an agent moving in a dynamic environment consisting of other moving agents, can we localize all agents on the ground plane without requiring that static background is visible in the images? We refer to this problem as view birdification in a crowd, the problem of computing a bird's-eye view of the movements of surrounding people from a single dynamic ego-centric view (see Fig. 1). Note that our focus is on the movements, not the appearance for which recent work has introduced various approaches. The need for view birdification frequently arises in a wide range of vision tasks when, for instance, a person is walking in a dense dynamic crowd where ego-centric views of the surrounding are limited, making static reference requirements unrealistic. A robust method to this key question would bring us a large step forward towards robust robot navigation and situational awareness in the wild, and also expand the horizon of surveillance.

We introduce a purely geometric approach to view birdification. The method only requires 2D bounding boxes of people in the ego-view and would generalize to different appearances. Our method is based on two key insights. First, the movement of the pedestrians are not arbitrary, but exhibit coordinated motion that can be expressed with crowd flow models [14, 32]. That is, the interaction of pedestrians' movements in a crowd can be locally described with analytic or data-driven models. Second, the scale and difference of human heights are proportional to estimated geometric depth [24]. In other words, the positions of pedestrians on the ground plane can be constrained along the lines that pass through a center of projection. These insights lend us a natural formulation of view birdification as a geometric reconstruction problem. We formulate view birdification as a cascaded optimization problem, consisting of camera ego-motion estimation constrained by predicted pedestrian motion, and pedestrian localization given the ego-motion estimate. We solve this with a cascaded optimization consisting of gradient descent and combinational optimization under the projection constraints and the assumed interaction model.

We experimentally validate our method on synthetic ego-centric views of people walking on trajectories extracted from publicly available crowd datasets. Since our method is appearance agnostic, these datasets exactly correspond to reality except for possible errors in bounding box extraction (*i.e.*, multi-object tracking). To evaluate the end-to-end accuracy including tracking errors, we create a photorealistic crowd dataset that simulates real camera projection with limited field of views and occluded pedestrian observations while moving in the crowd. These datasets allow us to quantitatively evaluate our method systematically and set the stage for further studies on view birdification. Experimental results demonstrate the effectiveness of our approach for view birdification in crowds of various densities.

Our contributions are threefold: (i) the introduction of view birdification, the simultaneous recovery of ground-plane trajectories of surrounding pedestrians and that of the observation camera just from an ego-centric view, as a novel research problem, (ii) the derivation of a cascaded optimization framework with a Bayesian formulation to solve the view birdification problem, and (iii) the construction of view birdification datasets consisting of paired real human trajectories and synthetic ego-views. We believe view birdification finds a wide range of applications and these contributions have strong implications in computer vision and robotics as they establish view birdification as a foundation for downstream visual understanding applications including crowd behavior analysis [1, 2, 11], self-guidance [18, 19], and robot navigation [31, 38].

2 Related Work

To our knowledge, our work is the first to formulate and tackle view birdification which finds relevance to several fundamental computer vision and robotics problems.

Bird's Eye View Transformation Conceptually, view birdification may appear similar to bird's eye view (BEV) synthesis. These two are fundamentally different in three critical ways. First, view birdification concerns the movements not the appearance, in contrast to BEV synthesis [33, 40, 41, 50, 51] or cross-view association [3, 4, 37]. Second, unlike most BEV methods [23, 30, 39], view birdification cannot rely on ground plane keypoints, multi-view images, or paired images between the views as they are usually not available in crowded scenes. Also note that, in crowded scenes, the ground plane and footsteps cannot be clearly extracted, which makes simple homography-based approaches impossible. Third, view birdification aims to localize all agents in a single coordinate frame across time, unlike BEV which is relative to the observer's location at each time instance [6, 27, 49]. As such, BEV synthesis methods are not directly applicable to view birdification.

Dynamic SLAM. View birdification can be considered as a dynamic SLAM problem in which all points, not just the observer but also the scene itself, are dynamic. Typical approaches to dynamic SLAM explicitly track and filter dynamic objects [7, 48] or implicitly minimize outliers caused by the dynamic objects [12, 13, 25]. In contrast to these approaches that sift out static keypoints from dynamic ones, methods that leverage both static and dynamic keypoints by, for instance, constructing a Bayesian factor graph [15, 16, 22] have also been introduced. The success of most of these approaches, however, depends on static keypoints which are hard to find and track in cluttered dynamic scenes such as in a dense crowd. In view birdification, we require no static keypoints and can reconstruct both ego-motion and surrounding dynamics only from the observed motions in the ego-view.

Crowd Modeling Modeling human behavior in crowds is essential for a wide range of applications including crowd simulation [20], trajectory forecasting [1, 11, 17], and robotic navigation [2, 31, 38]. Popular approaches include multi-agent interactions based on social force models [2, 14, 28], reciprocal force models [42], and imitation learning [38]. Recently, data-driven approaches have achieved significant performance gains on public crowd datasets [1, 11, 17]. All these approaches, however, are only applicable to near top-down views. Forecasting future location of people from first-person viewpoints has also been explored [26, 47], but they are limited to localization in the image plane. View birdification may provide a useful foundation for these crowd modeling tasks.

3 Geometric View Birdification

A typical scenario for view birdification is when a person with a body-worn camera is immersed in a crowd consisting of people heading towards their destinations while implicitly interacting with each other. Our goal is to deduce the global movements of people from the local observations in the ego-centric video captured by a single person.

3.1 Problem Setting

As a general setup, we assume that K people are walking on a fixed ground plane and an observation camera is mounted on one of them. We set the z-axis of the world coordinate system to the normal of the ground plane and denote the on-ground location of the k^{th} pedestrian as $\boldsymbol{x}_k = [x_k, y_k]^\top$. Let us denote the location of 0^{th} person in the crowd $\boldsymbol{x}_0$ as the observer capturing the ego-centric video of pedestrians $k \in \{1, 2, \ldots, K\}$ who are visible to the observer. The observation camera is located at $[x_0, y_0, h_0]^\top$, where the mounted height h_0 is constant across the frames. We assume that the viewing direction is parallel to the ground plane, *e.g.*, the person has a camera mounted on the shoulder. The same assumption applies when the observer is a vehicle or a mobile robot. At each timestep t, the pedestrians are observed by a camera with pose $[R|\boldsymbol{x}_0]^t$, where we assume 2D rotation and translation on the ground plane *i.e.*,$R \in SO(2)$ and $\boldsymbol{x}_0 \in \mathbb{R}^2$, respectively.

We assume that bounding boxes of the people captured in the ego-video are already extracted. For this, we can use an off-the-shelf multi-object tracker [45, 46] which provides the state of each pedestrian on the image plane $\boldsymbol{s}_k^t = \left[u_k^t, v_k^t, l_k^t\right]^\top$ which consists of the projections of center location and height, $\boldsymbol{p}_k^t = \left[u_k^t, v_k^t\right]^\top$ and l_k^t, respectively. Note that our method is agnostic to the actual tracking algorithm. Pedestrian IDs $k \in \{1, 2, \ldots, K\}$ can also be assigned by the tracker. Given a sequence of pedestrian states $\mathcal{S}_k$ from the first visible frame τ_1 to the last visible frame τ_2, *i.e.* , $\mathcal{S}_k^{\tau_1:\tau_2} = \{\boldsymbol{s}_k^{\tau_1}, \boldsymbol{s}_k^{\tau_1+1}, \ldots, \boldsymbol{s}_k^{\tau_2}\}$, our goal is to simultaneously reconstruct the K trajectories of the surrounding pedestrians $\mathcal{X}_k^{\tau_1:\tau_2} = \{\boldsymbol{x}_k^{\tau_1}, \boldsymbol{x}_k^{\tau_1+1}, \ldots, \boldsymbol{x}_k^{\tau_2}\}$ and that of the observation camera $\mathcal{X}_0^{\tau_1:\tau_2} = \{\boldsymbol{x}_0^{\tau_1}, \boldsymbol{x}_0^{\tau_1+1}, \ldots, \boldsymbol{x}_0^{\tau_2}\}$ with its viewing direction $\mathcal{R}^{\tau_1:\tau_2} = \{R^{\tau_1}, R^{\tau_1+1}, \ldots, R^{\tau_2}\}$ on the ground plane.

3.2 Observation Model

In the following, we set the z-axis of the world coordinate system to the normal of the ground plane (x-y plane). Let us denote rotation angles about the x-, y-, and z-axis with θ_x, θ_y, and θ_z, respectively. Assuming that the viewing direction of the camera is stabilized and parallel to the ground plane, we can approximate the rotation angles about the x- and y-axis to be $\Delta\theta_x = 0$ and $\Delta\theta_y = 0$ across the frames. That is, the camera pose to be estimated is represented by its rotation $R_z(\Delta\theta_z) \in SO(2)$ and translation $\Delta\boldsymbol{x}_0 \in \mathbb{R}^2$ on the ground plane.

We assume a regular perspective ego-centric view, but the following derivation also applies to other projection models including generic quasi-central cameras for fish-eye lens [8]. In the case of perspective projection with focal length f and intrinsic matrix $A \in \mathbb{R}^{3\times 3}$, the distance of the pedestrian from the observer is proportional to the ratio of the pedestrian height h_k and its projection l_k, *i.e.*,h_k/l_k. Given the footpoint of the pedestrian in the image plane $\boldsymbol{s}_k = [u_k, 0, l_k]$, the on-ground location estimate of the pedestrian relative to the camera

$\boldsymbol{z}_k = [\tilde{x}_k, \tilde{y}_k, 0]^\top$ can be computed by inverse projection of the observed image coordinates,

$$\begin{bmatrix}\tilde{x}_k & 0 & \tilde{y}_k\end{bmatrix}^\top = \frac{fh_k}{l_k} A^{-1} \begin{bmatrix}u_k & 0 & 1\end{bmatrix}^\top, \tag{1}$$

where the intrinsics A and focal length f are known since the observation camera can be calibrated a priori. The relative coordinates $\boldsymbol{z}_k$ are thus scaled by the unknown pedestrian height parameter h_k. The absolute position of the pedestrian $\boldsymbol{x}_k = [x_k, y_k]^\top$ can be computed by the relative coordinates $\boldsymbol{z}_k = [\tilde{x}_k, \tilde{y}_k]^\top$, the camera position $\boldsymbol{x}_0 = [x_0, y_0]^\top$, and the viewing direction θ_z,

$$\begin{bmatrix}x_k \\ y_k\end{bmatrix} = R_z(\theta_z)^\top \begin{bmatrix}\tilde{x}_k \\ \tilde{y}_k\end{bmatrix} + \begin{bmatrix}x_0 \\ y_0\end{bmatrix}. \tag{2}$$

Given a sequence of states $\mathcal{S}_k^{\tau_1:\tau_2} = \{\boldsymbol{s}_k^{\tau_1}, \boldsymbol{s}_k^{\tau_1+1}, \dots, \boldsymbol{s}_k^{\tau_2}\}$, we obtain corresponding on-ground location estimates relative to the camera $\mathcal{Z}_k^{\tau_1:\tau_2} = \{\boldsymbol{z}_k^{\tau_1}, \boldsymbol{z}_k^{\tau_1+1}, \dots, \boldsymbol{z}_k^{\tau_2}\}$ by inverse projection with unknown scale parameters using Eq. (1). The trajectories of pedestrians on the ground plane $\mathcal{X}_k^{\tau_1:\tau_2} = \{\boldsymbol{x}_k^{\tau_1}, \boldsymbol{x}_k^{\tau_1+1}, \dots, \boldsymbol{x}_k^{\tau_2}\}$ can be decomposed into the camera motion $\mathcal{X}_0^{\tau_1:\tau_2}$, $\mathcal{R}^{\tau_1:\tau_2}$ and the relative positions $\mathcal{Z}_k^{\tau_1:\tau_2}$ of pedestrians centered around the camera position. Our goal is to recover the camera ego-motion $\mathcal{X}_0^{\tau_1:\tau_2}$, $\mathcal{R}^{\tau_1:\tau_2}$ and the pedestrian trajectories $\{\mathcal{X}_1^{\tau_1:\tau_2}, \mathcal{X}_2^{\tau_1:\tau_2}, \dots, \mathcal{X}_K^{\tau_1:\tau_2}\} \in \mathbb{R}^{2\times K\times(\tau_2-\tau_1)}$ on the ground plane from the observations $\{\mathcal{S}_1^{\tau_1:\tau_2}, \mathcal{S}_2^{\tau_1:\tau_2}, \dots, \mathcal{S}_k^{\tau_1:\tau_2}\} \in \mathbb{R}^{3\times K\times(\tau_2-\tau_1)}$ captured by an ego-centric viewer.

4 A Cascaded Optimization for View Birdification

We derive a cascaded optimization approach to the geometric view birdification problem based on a Bayesian perspective.

4.1 A Bayesian Formulation

When a frame is pre-processed to a set of states $\mathcal{S}_{1:K}^t = \{\boldsymbol{s}_1^t, \boldsymbol{s}_2^t, \dots, \boldsymbol{s}_K^t\} \in \mathbb{R}^{2\times K}$ at time t, we obtain a set of on-ground position estimates relative to a camera $\mathcal{Z}_{1:K}^t = \{\boldsymbol{z}_1^\tau, \boldsymbol{z}_2^\tau, \dots, \boldsymbol{z}_K^t\} \in \mathbb{R}^{2\times K}$ corresponding to the states $\mathcal{S}_{1:K}^t$. Assuming that we have sequentially estimated on-ground positions up to time $t-1$, $\mathcal{X}_{0:K}^{t-\tau:t-1} = \{\mathcal{X}_0^{t-\tau:t-1}, \mathcal{X}_1^{t-\tau:t-1}, \dots, \mathcal{X}_K^{t-\tau:t-1}\} \in \mathbb{R}^{2\times(K+1)\times\tau}$ with a temporal time window τ, the posterior probability of the on-ground positions $\mathcal{X}_{0:K}^t = \{\boldsymbol{x}_0^t, \boldsymbol{x}_1^t, \dots, \boldsymbol{x}_K^t\} \in \mathbb{R}^{2\times(K+1)}$ at time t can be factorized as

$$p(\mathcal{X}_{0:K}^t | \mathcal{Z}_{1:K}^t, \mathcal{X}_{0:K}^{t-\tau:t-1}) \propto p(\mathcal{X}_{0:K}^t | \mathcal{X}_{0:K}^{t-\tau:t-1}) p(\mathcal{Z}_{1:K}^t | \mathcal{X}_{0:K}^t, \mathcal{X}_{0:K}^{t-\tau:t-1}). \tag{3}$$

Let $\Delta\boldsymbol{x}_0^t = [\Delta x_0^t, \Delta y_0^t, \Delta\theta^t] \in \mathbb{R}^3$ be the camera ego-motion from timestep $t-1$ to t consisting of a 2D translation $[\Delta x_0, \Delta y_0]$ and a change in viewing direction $\Delta\theta$ on the ground plane. The optimal motion of the camera $\Delta\hat{\boldsymbol{x}}_0^t$ and those of the pedestrians $\hat{\mathcal{X}}_{1:K}^t = \{\boldsymbol{x}_1^t, \boldsymbol{x}_2^t, \dots, \boldsymbol{x}_K^t\} \in \mathbb{R}^{2\times K}$ can be estimated as those that maximize the posterior distribution (Eq. (3)). The motion of observed pedestrians $\mathcal{X}_{1:K}^{t-1:t}$ are strictly constrained by the observing camera position $\boldsymbol{x}_0^t$ and its viewing direction θ^t. With recovered pedestrian parameters $\hat{\mathcal{X}}_{1:K}^t$, the optimal estimate of the camera ego-motion $\Delta\hat{\boldsymbol{x}}_0^t$ becomes

$$\Delta\hat{\boldsymbol{x}}_0^t = \underset{\Delta\boldsymbol{x}_0^t \in \mathbb{R}^3}{\operatorname{argmax}}\, p(\boldsymbol{x}_0^t | \mathcal{X}_0^{t-\tau:t-1}) \prod_k p(\boldsymbol{x}_k^t | \hat{\mathcal{X}}_k^{t-\tau:t-1}, \Delta\boldsymbol{x}_0^t) p(\boldsymbol{z}_k^t | \boldsymbol{x}_k^t, \Delta\boldsymbol{x}_0^t), \tag{4}$$

where $p(\boldsymbol{x}_0^t|\mathcal{X}_0^{t-\tau:t-1})$ and $p(\boldsymbol{x}_k^t|\mathcal{X}_k^{t-\tau:t-1}, \Delta\boldsymbol{x}_0^t)$ are motion priors of the camera and pedestrians conditioned on the camera motion, respectively. If the observer camera is mounted on a pedestrian following the crowd flow, $p(\boldsymbol{x}_0^t|\mathcal{X}_0^{t-\tau:t-1})$ obeys the same motion model as $p(\boldsymbol{x}_k^t|\mathcal{X}_k^{t-\tau:t-1})$.

As in previous work for pedestrian detection [24], we assume that the heights of pedestrians h_k follow a Gaussian distribution. This lets us define the likelihood of observed pedestrian positions $\boldsymbol{z}_k^t$ relative to the camera $\boldsymbol{x}_0^t$ as

$$\|\boldsymbol{z}_k^t\| \sim p(\boldsymbol{z}_k^t|\boldsymbol{x}_k^t; h_k) = \mathcal{N}(\mu_h, \sigma_h^2)\,, \tag{5}$$

where $\mathcal{N}(\mu_h, \sigma_h^2)$ is a Gaussian distribution with mean μ_h and variance σ_h^2. Once the ego-motion of the observing camera is estimated as $\Delta\hat{\boldsymbol{x}}_0^t$, the pedestrian positions $\hat{\mathcal{X}}_{1:K}^t$ that maximize the posterior $p(\mathcal{X}_{1:K}^t|\mathcal{Z}_{1:K}^t, \Delta\boldsymbol{x}_0^t)$ can be obtained as

$$\hat{\mathcal{X}}_{1:K}^t = \underset{\boldsymbol{x}_k^t \in \mathcal{X}_{1:K}^t}{\operatorname{argmax}} \prod_k p(\boldsymbol{x}_k^t|\mathcal{X}_k^{t-\tau:t-1}, \Delta\hat{\boldsymbol{x}}_0^t)\, p(\boldsymbol{z}_k^t|\boldsymbol{x}_k^t, \Delta\hat{\boldsymbol{x}}_0^t)\,. \tag{6}$$

That is, we can estimate the ego-motion of the observer constrained by the perceived pedestrian movements which conform to the crowd motion prior and the observation model.

4.2 Energy Minimization

Once the camera ego-motion is estimated, we can update the individual locations of pedestrians given the ego-motion in an iterative refinement process. View birdification can thus be solved with a cascaded optimization which first estimates the camera ego-motion and then recovers the relative locations between the camera and the pedestrians given the ego-motion estimate while taking into account the local interactions between pedestrians. Minimization of the negative log probabilities, Eqs. (4) and (6), can be expressed as

$$\underset{\Delta\boldsymbol{x}_0^t \in \mathbb{R}^3}{\text{minimize}} \quad \mathcal{E}_c \quad (\Delta\boldsymbol{x}_0^t; \hat{\mathcal{X}}_{1:K}^t, \mathcal{Z}_{1:K}^t, \mathcal{X}_{0:K}^{t-\tau:t-1})\,, \tag{7}$$

$$\text{subject to } \hat{\mathcal{X}}_{1:K}^t = \underset{\mathcal{X}_{1:K}^t}{\operatorname{argmin}}\, \mathcal{E}_p(\mathcal{X}_{1:K}^t; \Delta\hat{\boldsymbol{x}}_0^t, \mathcal{Z}_{1:K}^t, \mathcal{X}_{0:K}^{t-\tau:t-1})\,, \tag{8}$$

where we define the energy functions for positions of camera $\mathcal{E}_c$ and pedestrians $\mathcal{E}_c$ as

$$\mathcal{E}_c(\Delta\boldsymbol{x}_0^t; \hat{\mathcal{X}}_{1:K}^t, \mathcal{Z}_{1:K}^t, \mathcal{X}_{1:K}^{t-\tau:t-1}) = -\ln p(\boldsymbol{x}_0^t|\mathcal{X}_0^{t-\tau:t-1}) + \mathcal{E}_p\,, \tag{9}$$

$$\mathcal{E}_p(\mathcal{X}_{1:K}^t; \Delta\hat{\boldsymbol{x}}_0^t, \mathcal{Z}_{1:K}^t, \mathcal{X}_{0:K}^{t-\tau:t-1}) = \sum_{k=1}^{K} -\ln p(\boldsymbol{x}_k^t|\mathcal{X}_k^{t-\tau:t-1}, \Delta\boldsymbol{x}_0^t) + \sum_{k=1}^{K} -\ln p(\boldsymbol{z}_k^t|\boldsymbol{x}_k^t, \Delta\boldsymbol{x}_0^t)\,. \tag{10}$$

We minimize the energy in Eq. (7) by first computing an optimal camera position $\hat{\boldsymbol{x}}_0^t$ from Eq. (7) with gradient descent and initial state $\boldsymbol{x}_0^t = \boldsymbol{x}_0^{t-1}$. Given the estimate of the observer location $\hat{\boldsymbol{x}}_0^t$, we then estimate the pedestrian locations by solving the combinatorial optimization problem in Eq. (8) for $\mathcal{X}_k^t$ while considering all possible combinations of $\{\boldsymbol{x}_1^t, \ldots, \boldsymbol{x}_K^t\}$ that satisfy the projection constraint in Eq. (1) and the assumed pedestrian interaction model. This can be interpreted as a fully connected graph consisting of K pedestrian nodes with unary potential and interaction edges with pairwise potential. Similar to prior works on low-level vision problems [5, 21], Eq. (10) can be optimized by iterative message passing [10] on the graph. The possible states $\boldsymbol{x}_i$ are uniformly sampled on the projection line around μ_h with interval $[\mu_h - \delta S/2, \mu_h + \delta S/2]$, where S is the number of samples and $\delta = 0.01$. Considering only pairwise interactions and Gaussian potential, the complexity of the optimization is $\mathcal{O}(KS^2T)$, where T is the number of iterations required for convergence.

In this paper, we use two types of analytical interaction models, ConstVel [35] and Social Force [14]. We provide a detailed derivation of energy functions for these in the supplementary material.

5 Experiments

We validate the effectiveness of the proposed geometric view birdification method through an extensive set of experiments. Unfortunately, the COVID-19 pandemic has made real data collection impossible as it would inevitably involve many people. Instead, we fully leverage existing real pedestrian trajectories combined with synthetic camera views to thoroughly evaluate the accuracy of our method. Since our method only requires bounding boxes of people in the ego-centric view, we can fully evaluate the effectiveness of our method in real scenes by using real trajectories.

5.1 View Birdification Datasets

To the best of our knowledge, no public dataset is available for evaluating view birdification (*i.e.*, ego-video in crowds). We construct the following three datasets, which we will publicly disseminate, for evaluating our method and also to serve as a platform for further studies on view birdification. Please see supplementary material for detailed statistics of them.

Synthetic Pedestrian Trajectories The first dataset consists of synthetic trajectories paired with their synthetic projections to an observation camera. This data allows us to evaluate the effectiveness of view birdification when the crowd interaction model is known. The trajectories are generated by the social force model [14] with a varying number of pedestrians $K \in \{10, 20, 30, 40, 50\}$, and a perspective observation camera mounted on one of them. To evaluate the validity of our geometric formulation and optimization solution with this dataset, we assume ideal observation of pedestrians, *i.e.*, pedestrians do not occlude each other and their projected heights can be accurately deduced from the observed images. We also assume that the pedestrians are extracted from the ego-centric video perfectly but their heights h_k are sampled from a Gaussian distribution $h_k \sim \mathcal{N}(\mu_h, \sigma_h^2)$ with mean $\mu_h = 1.70$ [m] and a standard deviation $\sigma_h \in [0.00, 0.07]$ [m] based on the statistics of European adults [43].

Real Pedestrian Trajectories The second dataset consists of real pedestrian trajectories paired with their synthetic projections to an observation camera. The trajectories are extracted from publicly available crowd datasets: three sets of sequences from ETH [32] and UCY [20]. As in the synthetic pedestrian trajectories dataset, we render corresponding ego-centric videos from a randomly selected pedestrian's vantage point. With this, we obtain test sequences which we refer to as **Univ**, **Hotel** from ETH, and **Students** from UCY. Hotel, Univ, and Students datasets correspond to sparsely, moderately, and densely crowded scenarios, respectively. This dataset allows us to evaluate the effectiveness of our method on real data (movements).

Photorealistic Crowd Simulation The last dataset consists of synthetic trajectories paired with their photo-realistic projection captured with limited field of views and frequent occlusions between pedestrians. Evaluation on this dataset lets us examine the end-to-end

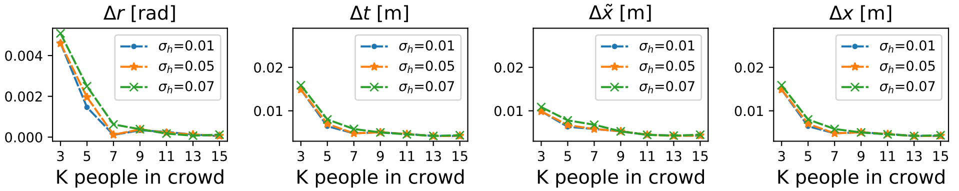

Figure 2: **Results on synthetic pedestrian trajectories.** Circle, star, and squared markers denote errors of estimated camera rotations Δr, translations $\Delta \boldsymbol{t}$, relative $\Delta \tilde{\boldsymbol{x}}$ and absolute localization errors $\Delta \boldsymbol{x}$, respectively, with standard deviations of pedestrian heights, $\sigma = 0.01, 0.05, 0.07$ [m], respectively.

effectiveness of our method including robustness to tracking errors. Inspired by previous works on crowd analysis and trajectory prediction [9, 44], we use the video game engine of *Grand Theft Auto V* (GTAV) developed by *Rockstar North* [34] with crowd flows automatically generated from programmed destinations with collision avoidance. We collected pairs of ego-centric videos with 90° field-of-view and corresponding ground truth trajectories on the ground plane using Script Hook V API [36]. We randomly picked 50 different person models with different skin colors, body shapes, and clothes. We prepare two versions of this data, one with manually annotated centerline and heights of the pedestrians in the observed video frames and the other with those automatically extracted with a pedestrian detector [45] pretrained on MOT-16 [29] which includes data captured from a moving platform.

5.2 View Birdification Results

Evaluation Metric We quantify the accuracy of our method by measuring the differences between the estimated positions of the pedestrians $\boldsymbol{x}_k^t$ and the observer $R^t, \boldsymbol{x}_0^t$ on the ground plane from their ground truth values $\dot{\boldsymbol{x}}_k^t, \dot{R}^t$, and $\dot{\boldsymbol{x}}_0^t$, respectively. The translation error for the observer is $\Delta \boldsymbol{t} = \frac{1}{T}\sum^T \|{\boldsymbol{x}_0}^t - \dot{\boldsymbol{x}}_0^t\|$, where T is a timestep duration of the sequence. The rotation error of the observer is $\Delta \boldsymbol{r} = \frac{1}{T}\sum_t \arccos(\frac{1}{2}\ \mathrm{trace}(R^t(\dot{R}^t)^\top - 1)$. We also evaluate the absolute and relative reconstruction errors of surrounding pedestrians which are defined by $\Delta \boldsymbol{x} = \frac{1}{K}\frac{1}{T}\sum_k\sum_t \|\boldsymbol{x}_k^t - \dot{\boldsymbol{x}}_k^t\|$ and $\Delta \tilde{\boldsymbol{x}} = \frac{1}{K}\frac{1}{T}\sum_k\sum_t \|(\boldsymbol{x}_k^t - \boldsymbol{x}_0^t) - (\dot{\boldsymbol{x}}_k^t - \dot{\boldsymbol{x}}_0^t)\|$, respectively.

Results on known interaction model Fig. 2 shows the view birdification results on the synthetic trajectories dataset. Although both rotation and translation errors slightly increase as the height standard deviation σ_h becomes larger, the error rate becomes lower as the number of people K increases. This suggests that the more crowded, the more certain the camera position and thus the more accurate the birdification of surrounding pedestrians.

Results on unknown real interaction models The real trajectories data allow us to evaluate the accuracy of our method when the interactions between pedestrians are not known. We employ two pedestrian interaction models, Social Force (SF) [14] and ConstVel (CV) [35]. We first evaluate the accuracy of our view birdification (VB) using these models, referred to as *VB-SF* and *VB-CV*, and compare them with baseline prediction models. In these baseline models, referred to as *ConstVel (CV)* and *Social Force (SF)*, we extrapolate a pedestrian position $\mathcal{X}_k^t$ from its past locations $\mathcal{X}_k^{t-2:t-1}$ based on the corresponding interaction model without using the observer's ego-centric view. That is, the baseline model is not view birdification but extrapolation according to pre-defined motion models on the ground plane.

Table 1: **Birdification results on real trajectories.** Relative and absolute localization errors of pedestrians, $\Delta\tilde{\boldsymbol{x}}, \Delta\boldsymbol{x}$ (top), and camera ego-motion errors, Δr and $\Delta\boldsymbol{t}$ (bottom), were computed for each frame for three different video sequences. Baseline methods only extrapolate movements on the ground plane resulting in missing entries (–). The results demonstrate the effectiveness of our view birdification.

Dataset		**Hotel / sparse**		**Univ / mid**		**Students / dense**	
	σ_h	$\Delta\tilde{\boldsymbol{x}}$ [m]	$\Delta\boldsymbol{x}$ [m]	$\Delta\tilde{\boldsymbol{x}}$ [m]	$\Delta\boldsymbol{x}$ [m]	$\Delta\tilde{\boldsymbol{x}}$ [m]	$\Delta\boldsymbol{x}$ [m]
CV [35]	–	–	0.294 ± 0.186	–	0.275 ± 0.195	–	0.223 ± 0.169
SF [14]	–	–	0.289 ± 0.207	–	0.261 ± 0.174	–	0.222 ± 0.163
VB-CV	0.00	0.051 ± 0.029	0.070 ± 0.030	0.089 ± 0.045	0.115 ± 0.049	0.022 ± 0.008	0.023 ± 0.008
	0.07	0.051 ± 0.029	0.070 ± 0.030	0.090 ± 0.045	0.116 ± 0.050	0.021 ± 0.007	0.022 ± 0.008
VB-SF	0.00	**0.048 ± 0.027**	**0.052 ± 0.033**	**0.070 ± 0.040**	**0.079 ± 0.047**	**0.009 ± 0.003**	**0.010 ± 0.006**
	0.07	**0.049 ± 0.027**	**0.052 ± 0.032**	**0.071 ± 0.040**	**0.080 ± 0.047**	**0.009 ± 0.004**	**0.010 ± 0.006**

	σ_h	$\Delta\boldsymbol{r}$ [rad]	$\Delta\boldsymbol{t}$ [m]	$\Delta\boldsymbol{r}$ [rad]	$\Delta\boldsymbol{t}$ [m]	$\Delta\boldsymbol{r}$ [rad]	$\Delta\boldsymbol{t}$[m]
VB-CV	0.00	**0.015 ± 0.030**	0.066 ± 0.089	0.016 ± 0.027	0.095 ± 0.125	**0.001 ± 0.001**	0.010 ± 0.007
	0.07	0.017 ± 0.039	0.069 ± 0.100	0.019 ± 0.034	0.110 ± 0.148	**0.001 ± 0.001**	0.010 ± 0.007
VB-SF	0.00	0.015 ± 0.036	**0.062 ± 0.104**	**0.015 ± 0.031**	**0.089 ± 0.135**	**0.001 ± 0.001**	**0.009 ± 0.006**
	0.07	**0.016 ± 0.042**	**0.062 ± 0.103**	**0.016 ± 0.035**	**0.091 ± 0.153**	**0.001 ± 0.001**	**0.009 ± 0.006**

Table 1 shows the errors of our method and baseline models. These results clearly show that our method, both *VB-CV* and *VB-SF*, can estimate the camera ego-motion and localize surrounding people more accurately, which demonstrates the effectiveness of birdifying the view and exploiting the geometric constraints on the pedestrians through it. *VB-SF* performs better than *VB-CV* especially in scenes with rich interactions such as Univ and Students, while they show similar performance on the Hotel dataset that includes less interactions. Both *VB-SF* and *VB-CV* show accurate camera ego-motion results in the Students dataset, which demonstrates the robustness of ego-centric view localization regardless of the assumed pedestrian interaction models. Our method achieves high accuracy on all three datasets across different standard deviations of heights $\sigma_h \in [0.00, 0.07]$. This also shows that the method is robust to variation in human heights.

Photorealistic Crowds. Fig. 3 shows qualitative results on the photorealistic crowd dataset. As shown in the top two rows, our method accurately estimates camera ego-motion and on-ground positions of automatically detected pedestrians with an off-the-shelf tracker [45]. People tracked in more than three frames are birdified. Even with occlusions in the image and noisy height estimates computed from detected bounding boxes, our approach robustly estimates the camera ego-motion and surrounding pedestrian positions. Due to perspective projection, localization error caused by erroneous detection in the image plane is proportional to the ground-plane distance between the camera and the detected pedestrian. We further compared these results with manually annotated pedestrian heights as shown in the bottom two rows Fig. 3 to highlight the effect of automatically detecting the pedestrians for view birdification (*i.e.*, to see how the results change if the pedestrian heights were accurate). The resulting accuracies are comparable, which demonstrates the end-to-end effectiveness. To further ameliorate the errors caused by detection noises, our method can also be extended, for instance, by replacing the noise model in Eq. (5) with a 2D Gaussian distribution. Please also see the supplemental material and video.

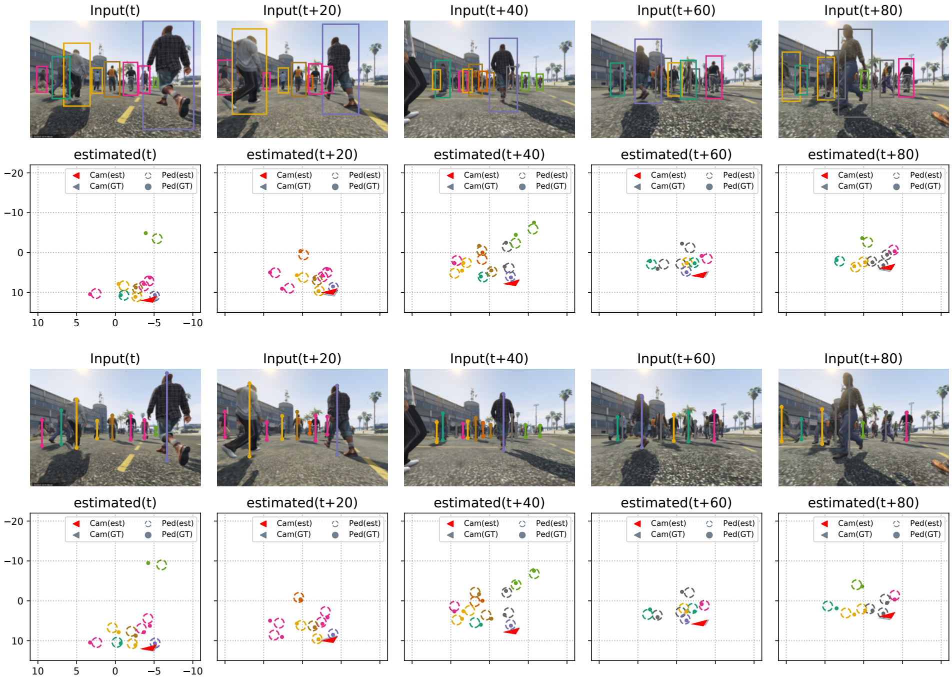

Figure 3: **Results on photorealistic crowd dataset.** The top row shows detected pedestrians with a multi-object tracker in bounding boxes and the third row shows manually annotated human heights (center lines). The figures in the second and fourth rows depict view birdification results for them. Colors correspond to Pedestrian IDs. Red triangles denote camera position estimates $\boldsymbol{x}_0^t$ and dashed circles denote estimated pedestrian positions $\boldsymbol{x}_k^t$ at time t. Grey triangles and circles denote ground-truth camera and pedestrian positions, respectively. View birdification results for both automatic and manually detected people show consistently high accuracy. These results demonstrate the end-to-end accuracy of view birdification.

6 Conclusion

In this paper, we introduced view birdification, the problem of recovering the movement of surrounding people on the ground plane from a single ego-centric video captured in a dynamic cluttered scene. We formulated view birdification as a geometric reconstruction problem and derived a cascaded optimization approach that consists of camera ego-motion estimation and pedestrian localization while fully modeling the local pedestrian interactions. Our extensive evaluation demonstrates the effectiveness of our proposed view birdification method for crowds of varying densities. Currently, the occlusion handling is carried out by an external multi-object tracker. We envision a feedback loop from our birdification framework that can inform the multi-object tracker to reason better about the occluded targets, which will likely enhance the accuracy as a whole even in heavily occluded scenes. We believe our work has implications for both computer vision and robotics, including crowd behavior analysis, self-localization, and situational awareness, and opens new avenues of applications including dynamic surveillance.

References

[1] Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. Social lstm: Human trajectory prediction in crowded spaces. In *Proc. CVPR*, pages 961–971, 2016.

[2] Bani Anvari and Helge A. Wurdemann. Modelling social interaction between humans and service robots in large public spaces. In *Proc. IROS*, pages 11189–11196, 2020. doi: 10.1109/IROS45743.2020.9341133.

[3] Shervin Ardeshir and Ali Borji. Ego2top: Matching viewers in egocentric and top-view videos. In *Proc. ECCV*, pages 253–268. Springer, 2016.

[4] Shervin Ardeshir, Krishna Regmi, and Ali Borji. Egotransfer: Transferring motion across egocentric and exocentric domains using deep neural networks. *CoRR*, 2016.

[5] Vijay Badrinarayanan, Ignas Budvytis, and Roberto Cipolla. Mixture of trees probabilistic graphical model for video segmentation. *IJCV*, 110(1):14–29, 2014.

[6] Lorenzo Bertoni, Sven Kreiss, and Alexandre Alahi. Monoloco: Monocular 3d pedestrian localization and uncertainty estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6861–6871, 2019.

[7] Berta Bescos, José M. Fácil, Javier Civera, and José Neira. DynaSLAM: Tracking, Mapping and Inpainting in Dynamic Scenes. In *Proc. IROS*, 2018.

[8] Pierre-Andre Brousseau and Sebastien Roy. Calibration of axial fisheye cameras through generic virtual central models. In *Proc. ICCV*, 2019.

[9] Zhe Cao, Hang Gao, Karttikeya Mangalam, Qizhi Cai, Minh Vo, and Jitendra Malik. Long-term human motion prediction with scene context. In *Proc. ECCV*. 2020.

[10] Pedro F Felzenszwalb and Daniel P Huttenlocher. Efficient belief propagation for early vision. *IJCV*, 70(1):41–54, 2006.

[11] Agrim Gupta, Justin Johnson, Li Fei-Fei, Silvio Savarese, and Alexandre Alahi. Social gan: Socially acceptable trajectories with generative adversarial networks. In *Proc. CVPR*, pages 2255–2264, 2018.

[12] Dirk Hähnel, Dirk Schulz, and Wolfram Burgard. Map building with mobile robots in populated environments. In *Proc. IROS*, pages 496–501, 2002.

[13] Dirk Hahnel, Rudolph Triebel, Wolfram Burgard, and Sebastian Thrun. Map building with mobile robots in dynamic environments. In *2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422)*, volume 2, pages 1557–1563. IEEE, 2003.

[14] Dirk Helbing and Peter Molnar. Social force model for pedestrian dynamics. *Physical review E*, 51(5):4282, 1995.

[15] Mina Henein, Jun Zhang, Robert Mahony, and Viorela Ila. Dynamic slam: The need for speed. In *Proc. ICRA*, pages 2123–2129. IEEE, 2020.

[16] Jiahui Huang, Sheng Yang, Tai-Jiang Mu, and Shi-Min Hu. Clustervo: Clustering moving instances and estimating visual odometry for self and surroundings. In *Proc. CVPR*, pages 2168–2177, 2020.

[17] Boris Ivanovic and Marco Pavone. The trajectron: Probabilistic multi-agent trajectory modeling with dynamic spatiotemporal graphs. In *Proc. ICCV*, pages 2375–2384, 2019.

[18] Seita Kayukawa, Keita Higuchi, João Guerreiro, Shigeo Morishima, Yoichi Sato, Kris Kitani, and Chieko Asakawa. Bbeep: A sonic collision avoidance system for blind travellers and nearby pedestrians. In *Proc. CHI*, CHI '19, pages 52:1–52:12, New York, NY, USA, May 2019. ACM. ISBN 978-1-4503-5970-2/19/05. doi: 10.1145/3290605.3300282. URL https://doi.org/10.1145/3290605.3300282.

[19] Seita Kayukawa, Ishihara Tatsuya, Hironobu Takagi, Shigeo Morishima, and Chieko Asakawa. Blindpilot: A robotic local navigation system that leads blind people to a landmark object. In *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems (CHI EA '20)*, CHI EA '20, New York, NY, USA, Apr 2020. ACM. ISBN 978-1-4503-6819-3/20/04. doi: 10.1145/3334480.3382925. URL https://doi.org/10.1145/3334480.3382925.

[20] Alon Lerner, Yiorgos Chrysanthou, and Dani Lischinski. Crowds by example. *Computer graphics forum*, 26(3):655–664, 2007.

[21] Jose Lezama, Karteek Alahari, Josef Sivic, and Ivan Laptev. Track to the future: Spatio-temporal video segmentation with long-range motion cues. In *Proc. CVPR*, pages 3369–3376, 2011. doi: 10.1109/CVPR.2011.6044588.

[22] Peiliang Li, Tong Qin, et al. Stereo vision-based semantic 3d object and ego-motion tracking for autonomous driving. In *Proc. ECCV*, pages 646–661, 2018.

[23] Chien-Chuan Lin and Ming-Shi Wang. A vision based top-view transformation model for a vehicle parking assistant. *Sensors*, 12(4):4431–4446, 2012.

[24] Yan Luo, Chongyang Zhang, Muming Zhao, Hao Zhou, and Jun Sun. Where, what, whether: Multi-modal learning meets pedestrian detection. In *Proc. CVPR*, pages 14065–14073, 2020.

[25] Zhaoyang Lv, Frank Dellaert, James M Rehg, and Andreas Geiger. Taking a deeper look at the inverse compositional algorithm. In *Proc. CVPR*, pages 4581–4590, 2019.

[26] Osama Makansi, Özgün Çiçek, Kevin Buchicchio, and Thomas Brox. Multimodal future localization and emergence prediction for objects in egocentric view with a reachability prior. In *Proc. CVPR*, pages 4354–4363, 2020. URL http://lmb.informatik.uni-freiburg.de/Publications/2020/MCBB20.

[27] Kaustubh Mani, Swapnil Daga, Shubhika Garg, Sai Shankar Narasimhan, Madhava Krishna, and Krishna Murthy Jatavallabhula. Monolayout: Amodal scene layout from a single image. In *Proc. WACV*, pages 1689–1697, 2020.

[28] Ramin Mehran, Alexis Oyama, and Mubarak Shah. Abnormal crowd behavior detection using social force model. In *Proc. CVPR*, pages 935–942. IEEE, 2009.

[29] Anton Milan, Laura Leal-Taixé, Ian Reid, Stefan Roth, and Konrad Schindler. Mot16: A benchmark for multi-object tracking. *arXiv preprint arXiv:1603.00831*, 2016.

[30] Armin Mustafa, Hansung Kim, Jean-Yves Guillemaut, and Adrian Hilton. General dynamic scene reconstruction from multiple view video. In *Proc. ICCV*, December 2015.

[31] Mai Nishimura and Ryo Yonetani. L2b: Learning to balance the safety-efficiency tradeoff in interactive crowd-aware robot navigation. In *Proc. IROS*, pages 11004–11010, 2020. doi: 10.1109/IROS45743.2020.9341519.

[32] Stefano Pellegrini, Andreas Ess, Konrad Schindler, and Luc Van Gool. You'll never walk alone: Modeling social behavior for multi-target tracking. In *Proc. ICCV*, pages 261–268, 2009.

[33] Krishna Regmi and Ali Borji. Cross-view image synthesis using conditional gans. In *Proc. CVPR*, pages 3501–3510, 2018.

[34] Rockstar Games. Rockstar Games. https://www.rockstargames.com.

[35] Christoph Schöller, Vincent Aravantinos, Florian Lay, and Alois Knoll. What the constant velocity model can teach us about pedestrian motion prediction. *IEEE Robotics and Automation Letters*, 5(2):1696–1703, 2020.

[36] Script Hook V. Script Hook V. http://www.dev-c.com/gtav/.

[37] Bilge Soran, Ali Farhadi, and Linda Shapiro. Action recognition in the presence of one egocentric and multiple static cameras. In *Proc. ACCV*, pages 178–193. Springer, 2014.

[38] Lei Tai, Jingwei Zhang, Ming Liu, and Wolfram Burgard. Socially compliant navigation through raw depth inputs with generative adversarial imitation learning. In *Proc. ICRA*, pages 1111–1117. IEEE, 2018.

[39] Aparna Taneja, Luca Ballan, and Marc Pollefeys. Modeling dynamic scenes recorded with freely moving cameras. In *Proc. ACCV*, pages 613–626, 2010.

[40] Hao Tang, Dan Xu, Nicu Sebe, Yanzhi Wang, Jason J. Corso, and Yan Yan. Multichannel attention selection gan with cascaded semantic guidance for cross-view image translation. In *Proc. CVPR*, 2019.

[41] Hao Tang, Dan Xu, Yan Yan, Philip HS Torr, and Nicu Sebe. Local class-specific and global image-level generative adversarial networks for semantic-guided scene generation. In *Proc. CVPR*, 2020.

[42] Jur Van Den Berg, Stephen J Guy, Ming Lin, and Dinesh Manocha. Reciprocal n-body collision avoidance. In *Robotics research*, pages 3–19. Springer, 2011.

[43] Peter M. Visscher. Sizing up human height variation. *Nature Genetics*, 40:489–490, 2008.

[44] Qi Wang, Junyu Gao, Wei Lin, and Yuan Yuan. Learning from synthetic data for crowd counting in the wild. In *Proc. CVPR*, pages 8198–8207, 2019.

[45] Zhongdao Wang, Liang Zheng, Yixuan Liu, and Shengjin Wang. Towards real-time multi-object tracking. In *Proc. ECCV*, 2020.

[46] Yuliang Xiu, Jiefeng Li, Haoyu Wang, Yinghong Fang, and Cewu Lu. Pose Flow: Efficient online pose tracking. In *Proc. BMVC*, 2018.

[47] Takuma Yagi, Karttikeya Mangalam, Ryo Yonetani, and Yoichi Sato. Future person localization in first-person videos. In *Proc. CVPR*, pages 7593–7602, 2018.

[48] Chao Yu, Zuxin Liu, Xin-Jun Liu, Fugui Xie, Yi Yang, Qi Wei, and Qiao Fei. Ds-slam: A semantic visual slam towards dynamic environments. In *Proc. IROS*, pages 1168–1174. IEEE, 2018.

[49] Jianfeng Zhang, Dongdong Yu, Jun Hao Liew, Xuecheng Nie, and Jiashi Feng. Body meshes as points. *arXiv preprint arXiv:2105.02467*, 2021.

[50] Tinghui Zhou, Shubham Tulsiani, Weilun Sun, Jitendra Malik, and Alexei A Efros. View synthesis by appearance flow. In *Proc. ECCV*, 2016.

[51] Xinge Zhu, Zhichao Yin, Jianping Shi, Hongsheng Li, and Dahua Lin. Generative adversarial frontal view to bird view synthesis. In *Proc. 3DV*, pages 454–463. IEEE, 2018.