Free-Viewpoint Video in the Wild Using a Flying Camera

Zhengdong Hong and Wenhao Shen

Zhejiang University
12321092@zju.edu.cn
https://github.com/hongzhengdong

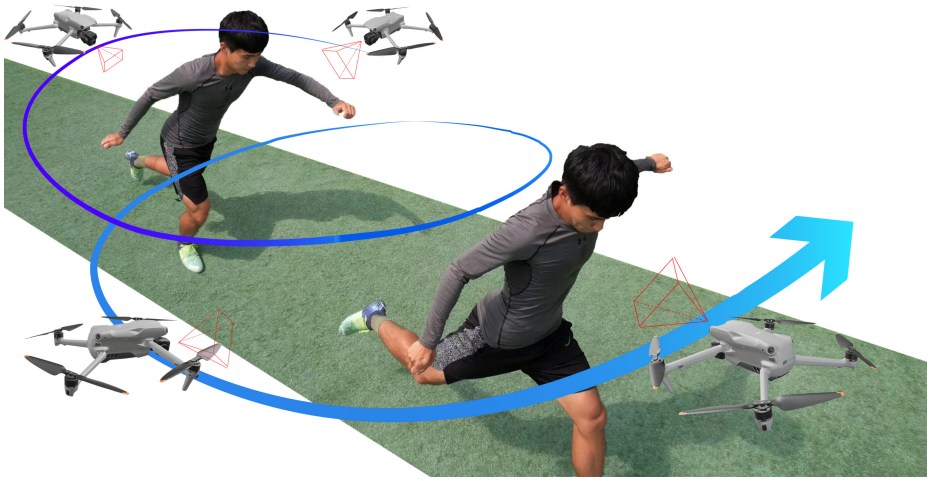

Fig. 1: Our system's input is a monocular video of outdoor sports, shot by a single drone autonomously orbiting around the athlete. Color gradients represent time changes. Video is available at:https://www.youtube.com/watch?v=JF-cQjc_sv4&t=3s

Abstract. We propose a novel drone application under real-world scenarios – free-viewpoint rendering of outdoor sports scenes, including the dynamic athlete and the 360° background. Outdoor sports have long-range human motions and large-scale scene structures which make the task rather challenging. Existing methods either rely on dense camera arrays which costs much, or a handheld moving camera which struggles to handle real sports scenes. We build a novel drone-based system using an RGB camera to reconstruct the 4D dynamic human along with the 3D unbounded scene, rendering free-viewpoint videos at any time. We also propose submodules for calibration and human motion capture, as a system-level design for improved robustness and efficiency. We collect a dataset AerialRecon and conduct extensive experiments on real-world scenarios. Compared with existing SOTA systems, our system demonstrates superior performance and applicability to real-world.

Keywords: Novel View Synthesis · Sports · Drone

1 Introduction

Neural implicit reconstruction and rendering has been extensively investigated these years. NeRF [29] and its follow-ups [28, 34, 48] have demonstrated promising performances in rendering both static scene and dynamic human. Some work target on rendering dynamic scenes with a moving human subject in it. Some of them take RGB images as input. However, most methods either rely on multi-view fixed cameras [34, 35] or require the human to stay in a limited space in front of cameras during data collection [9, 16, 25, 42]. The others adopt RGBD images as inputs [10, 31, 46], but depth sensors normally work poorly in outdoor settings, which restrains their real-world applications. For large-scale outdoor sports scenes, human performs large-range motion in a very wide space with unpredictable paths. It's impractical to set up fixed camera arrays closely surrounding the moving athlete, and is labor-intensive to use multiple handheld cameras, especially when the target human moves very fast. Therefore, it is rather challenging for dynamic reconstruction under large-scale outdoor settings, and existing fixed or handheld camera systems struggle to handle it.

A flying drone inherently possesses exceptional spatial freedom and rapid mobility, offering a plausible solution. With the vision-tracking algorithm, a drone camera can move quickly to track people in the wild without terrain constraints. It is very beneficial for real-world outdoor sports.

Another challenge for outdoor sports is the limited views of observation. On the one hand, for human reconstruction, existing backbones [9,13,16,25,33,34,42] highly rely on full observations of humans (including the person's back and side) to render a human in high-fidelity and intricate details. Therefore, the human subjects are asked to self-rotate in front of the camera to obtain full observations of human bodies during data capture. However, it's impossible to require a moving athlete to actively cooperate with the camera while performing real outdoor sports. And for most real-world sports videos, limited viewing angles and incomplete observations could result in artifacts and degradation of human rendering quality. Although [15, 35] set up multi-view fixed camera array surrounding the scene to gather more viewing angles. However, all the athletes are confined to act in a small area and a sized up scene necessitates much more cameras. On the other hand, as we aim to synthesize 360° free viewpoint videos of the whole scene, we also need to reconstruct the 3D large-scale background, which requires dense views as input.

A fastly orbiting drone provides a plausible solution. Based on the built-in navigation diagram, a drone can orbit around the target human, quickly switching the pose to capture multiple sides of the human body to get all-sides observation. Besides, while orbiting the human, a drone simultaneously captures 360° images of the surrounding background with dense views. The scene reconstruction part of our pipeline will take advantage of these dense views to create 360° realistic rendering of large-scale outdoor sports scenes.

Thanks to the virtues of an orbiting drone, we achieve promising improvements over the traditional handheld systems in multiple challenging outdoor sports scenes. Our contributions are as follows:

- We propose a novel application of novel-view-rendering of the dynamic athlete and a large-scale outdoor sports scene using only monocular video as input, which existing systems like a fixed camera or a handheld camera struggle to handle.
- We come up with a monocular RGB-only drone system under the challenging setting. We proposed some crucial system-level design based on drone orbiting diagram, including Drone Motion Constraint (DMC) and Drone-based Sequential Mocap (DSM), for improved system accuracy and robustness.
- We collect a new dataset which consists of various real-world outdoor sports scenes, where we perform extensive real-world experiments. The dataset will be released upon acceptance to benefit future researches of the community.

2 Related Work

2.1 Existing Systems

Multi-view systems like [15, 23, 35, 43] have showcased outstanding results on free-viewpoint rendering of dynamic scenes. Nevertheless, the requirement for multi-camera setup makes those system hard to apply in monocular scenes. Although there are some works under monocular settings [6, 21, 22, 32, 44, 49], most of them don't target on render the 360° views but only limited views around the input views. Moreover, they represent the dynamic human and static background with a single model which makes it hard to model the fast human motion appropriately. [16,25] use two different models to represent the human and scene, as well as incorporating human motion priors. However, they require the human to self-rotate in front of the camera to ensure the rendering quality, which is not always practical and largely limits their utility in real-world outdoor scenarios. We are the first monocular system that targets on the reconstruction and 360° rendering of challenging outdoor scenes like real-world sports.

2.2 Monocular RGB Human Reconstruction and Rendering

The rendering of human body with color and appearance using monocular RGB inputs has been extensively investigated. Shading-based human body shape refinement is applied in [2] but a complicated procedure of first segmenting then assembling is needed. PiFu-related algorithms [11, 37, 38] realize generalizable human reconstruction among human with different appearances but the rendering quality is not ideal. Different from them, per-scene optimization methods like [9, 13, 14, 33, 34, 42, 45] use the 3D human poses as input priors and enable novel-viewpoint rendering with higher fidelity. All these methods inevitably suffer from quality degradation under the monocular setting, particularly when certain parts of the human body, such as the back or sides, are not visible in the input data. Therefore, most human rendering backbones have a strong restriction during data capture: they ask the human subject to actively turn around in a limited space in front of the camera, so as to ensure the input observations

covers the whole human body. Therefore, we can see [16,25] use a data collection method like that. However, in real-world outdoor sports, human athlete moves unpredictably and fast where systems like could fail. Contrarily, we build a system based on a fast moving camera that can actively orbit around the people, during which various viewpoints are provided to facilitate human reconstruction.

2.3 Human Motion Capture and Drone-based Systems

Recent mainstream human rendering methods take precise 3D human poses as input priors represented by SMPL [26], a parameterized human model. Recovering 3D human poses from a monocular video remains challenging [40, 50, 52]. Firstly, monocular images have inherent depth ambiguity, and this could cause inconsistency and inaccuracy of the poses estimation in 3D world. Pipelines leverage depth sensors are able to tackle this problem but they do not work well in outdoor settings [30, 51]. Following [50], we utilize temporal information of video sequences during optimization to alleviate this problem. Secondly, in real-world sport videos, human motions could contain severe self-occlusion, where the 2D keypoints detection and SMPL-fitting could fail. GLAMR [52] uses generative models in helping recovering poses from occluded human parts. Similar to [8], we adopt an occlusion-adaptive detector based on 2D transformer [20] to alleviate this. For outdoor applications, drone-based systems have been proposed. Zhou [54] utilize the drone's fast moving characteristics to facilitate a NRSFM algorithm and improve the accuracy of 3D human skeleton. ActiveMoCap [18] further solves the Next-Best-View problem for optimal estimation. As they proved, the fast-varying viewpoints of a drone camera could largely benefit monocular human pose estimation. However, they target at 3D human skeleton recovery, while we aim to estimate human pose with shape, represented in SMPL. In contrast to other drone-based systems [1, 12, 36, 41, 47] focusing on recovering SMPL as the main goal, our aerial system pioneers the utilization of estimated SMPL for rendering the human with appearance and color.

2.4 3D Scene Reconstruction and Rendering

Reconstruction and Rendering a static scene from multi-view posed images have been widely explored. While conventional methods like structure-from-motion [39] and Dense-MVS [7] struggle to achieve realistic rendering results, NeRF [29] leverages neural representation for scene modeling. NeRF++ [53] employs inverted sphere parameterization to enable NeRF to model unbounded scenes. Mip-NeRF360 [4] enhances rendering results on unbounded 360-degree scenes through optimized sampling strategies. [17] is a emerging technology of high-quality rendering that necessitates a 3D scene point cloud as input. These approaches rely on dense-view images for reconstructing large-scale outdoor scenes. Fortunately, our system benefits from drone mobility to capture dense views of larger-scale, intricate structures in real-world sports scenes.

3 Methods

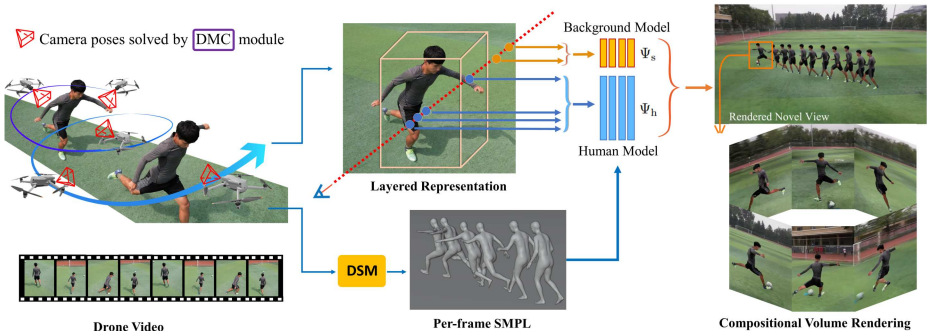

Fig. 2: Provided with a drone video, we reconstruct the human and background, enabling 360° rendering of the whole scene at any time. We generate novel viewpoint rendering (top right), and 360° renderings of the kicking moment (bottom right).

3.1 Calibration

Our drone is equipped with a single uncalibrated onboard camera. Given a captured video $\{\boldsymbol{I}_i\}, i \in \{1, \ldots, n\}$, Our first step involves estimating camera parameters $\{\mathbf{P}_i\}, i \in \{1, \ldots, n\}$. The camera pose for each frame is defined as:

$$\mathbf{P}_i = \mathbf{K}_i \left[\mathbf{R}_i \mid \mathbf{T}_i\right].$$

We modify COLMAP [39], a structure-from-motion method, into a sequential version with drone motion constraints. Firstly, as the consecutive ordering of the image sequences is known, we adopt sequential feature matching rather than random matching to identify feature correspondences during the triangulation phase of SfM. Subsequently, during bundle adjustment, we introduce the **DMC (Drone Motion Constraint)** as a constraint to ensure smooth and continuous camera trajectory. Specifically, as our setup involves an autonomously navigating drone capturing video sequences, following predefined flying control diagrams, it orbits around the human in an arc trajectory at a consistent linear speed scalar and height. With recording at a fixed frame rate of 30fps, we impose constraints on the camera trajectory between consecutive frames.

$$\left| ||C_{i+1} - C_i|| - ||C_i - C_{i-1}|| \right| < \varepsilon \tag{1}$$

where $||\cdot||$ is the euclidean distance of three-dimensional coordinates. C_i denotes the 3D position of the camera center of the frame i under the world coordinate:

$$C_i = -\mathbf{R}_i^{-1}\mathbf{T}_i \tag{2}$$

Substituting Eq. 2 into Eq. 1, we can get

$$\left| || - \mathbf{R}_{i+1}^{-1}\mathbf{T}_{i+1} + \mathbf{R}_i^{-1}\mathbf{T}_i|| - || - \mathbf{R}_i^{-1}\mathbf{T}_i + \mathbf{R}_{i-1}^{-1}\mathbf{T}_{i-1}|| \right| < \varepsilon \tag{3}$$

we set ε a very small value approximately equal to zero.

During the bundle adjustment, we optimize camera parameters by minimizing the sum-of-squared reprojection errors:

$$E = \min_{\mathbf{R}_i, \mathbf{T}_i, \mathbf{X}_j} \sum_{i=1}^{n} \sum_{j=1}^{n_{3D}} \left\| \mathbf{x}_{ji} - g\left(\mathbf{X}_j, \mathbf{R}_i, \mathbf{T}_i, \mathbf{K}_i\right) \right\|^2 \tag{4}$$

$$g\left(\mathbf{X}_j, \mathbf{R}_i, \mathbf{T}_i, \mathbf{K}_i\right) \sim \mathbf{P}_i \mathbf{X}_{ji} \tag{5}$$

where n_{3D} is the number of 3D points in the scene. $\mathbf{X}_j$ is the j-th 3D point in the scene and observation x_{ji} is the 2D image coordinates of point $\mathbf{X}_j$ in camera i. The mapping $g(\cdot)$ is a transformation that projects a 3D point $\mathbf{X}_j$ onto the image plane of camera i.

We take Eq. 3 as an additional constraint for the optimization problem in Eq. 4. We utilize the camera characteristics during the drone's automatic orbiting process and improve the robustness of camera pose estimation.

Note that in the feature detection process [27], we mask out the dynamic region in the scene (e.g. human) using [19] to avoid incorrect matches. We also scale the camera parameters by γ to match the real-world scale, by placing a calibration board with 0.1m grid size in the scene during data capture, with the distance of two 3D points restored in COLMAP ι_1:

$$\gamma = \frac{\iota_1}{0.1} \tag{6}$$

3.2 Human Motion Capture (MoCap)

Recovering human appearance directly from drone videos is challenging due to sparse observations and highly dynamic human movements. Therefore, we use human motion capture (MoCap) as a prerequisite for training our human model in 3.3. Here, we use a parameterized human model, SMPL, from [26]. This model encodes the dynamic motions prior of humans in videos.

The SMPL model is a differentiable function $\boldsymbol{M}(\boldsymbol{\theta}, \boldsymbol{\beta}) \in \mathbb{R}^{3 \times N_v}$ where the pose parameters $\boldsymbol{\theta} \in \mathbb{R}^{72}$ and the shape parameters $\boldsymbol{\beta} \in \mathbb{R}^{10}$ are mapped to a triangulated mesh with $N_v = 6890$ vertices. $\boldsymbol{R^h} \in SO(3)$ and $\boldsymbol{T^h} \in \mathbb{R}^3$ denote the global rotation and translation of human under the world coordinate, respectively. The 3D body joints $\boldsymbol{J}(\boldsymbol{\theta}, \boldsymbol{\beta})$ of the model can be defined as a linear combination of the mesh vertices. Therefore, for N_j joints, we defined the body joints $\boldsymbol{J}(\boldsymbol{\theta}, \boldsymbol{\beta}) \in \mathbb{R}^{3 \times N_j} = \mathcal{J}(\boldsymbol{M}(\boldsymbol{\theta}, \boldsymbol{\beta}))$, where $\mathcal{J}$ is a pre-trained linear regressor to convert SMPL vertices to OpenPose [5] keypoints.

Given a drone video $\{\boldsymbol{I}_i\}, i \in \{1, \ldots, n\}$, we aim to recover $(\boldsymbol{\theta}_i, \boldsymbol{\beta}_i, \boldsymbol{R}_i^h, \boldsymbol{T}_i^h), i \in \{1, \ldots, n\}$. For each frame of the video, we first estimate the 25-point 2D human keypoints $\{\boldsymbol{W}_i\}, i \in \{1, \ldots, n\}$ defined in [5]using a 2D transformer-based network in [20] which is more accurate compared to CNN-based model in sports scenes where severe self-occlusion occurs, claimed in [8, 50]. Then we optimize the per-frame SMPL model by utilizing temporal information by drone orbiting diagrams, which we call the **Drone-based Sequential MoCap (DSM)**.

In monocular human pose estimation, it has been verified that the viewpoints of the camera will largely affect the MoCap result [18, 54]. Here we utilize the drone camera poses $\mathbf{P}_i$ derived from our **DMC** module to formulate a optimization procedure by leveraging the fast varying viewpoints under our setting:

The optimization objective consists of two terms, the reprojection error term L_{2d} and the smoothness term L_{temp}, ω is a weight factor:

$$\min_{\boldsymbol{\theta}_i, \boldsymbol{\beta}_i, \boldsymbol{R}_i^h, \boldsymbol{T}_i^h} f(\boldsymbol{\theta}_i, \boldsymbol{\beta}_i, \boldsymbol{R}_i^h, \boldsymbol{T}_i^h) = L_{2d} + \omega L_{temp} \tag{7}$$

The projection term penalizes the 2D distance between the estimated 2D keypoints $\boldsymbol{W_i}$ and the corresponding projected SMPL joints:

$$L_{2d} = \sum_{i=1}^{n} \|\boldsymbol{W}_i - \Pi[\boldsymbol{R_i}(\boldsymbol{R_i^h}\boldsymbol{J}(\boldsymbol{\theta}, \boldsymbol{\beta})_i + \boldsymbol{T_i^h}) + \boldsymbol{T_i}]\|^2 \tag{8}$$

where Π is the projection from 3D to 2D using camera intrinsics $\mathbf{K}_i$. Unlike [16, 25] which estimates 3D human pose in the camera coordinate and requires a post-processing stage to solve the scene-human alignment using an assumed ground plane, we take advantage of decoupling the camera poses and human poses to get 3D-consistent human registrations in the world coordinate directly.

As our drone camera shoots at a constant 30fps, we add a temporal smoothness constraint to ensure that joint positions and human body shapes do not vary too much between consecutive frames. λ_β is a weight.

$$L_{temp} = \sum_{i=1}^{n-1} \|\boldsymbol{\theta}_i - \boldsymbol{\theta}_{i+1}\|_F^2 + \lambda_\beta \sum_{i=1}^{n-1} \left\|\boldsymbol{\beta}_i - \boldsymbol{\beta}_{i+1}\right\|^2 \tag{9}$$

Since we control our drone in a constant rotation manner, which is one of the optimal camera positioning diagrams evaluated by ActiveMoCap [18], we are able to take advantages of the fast-varying viewpoints from drone mobility under our settings.

3.3 Human Model $\Psi_\mathbf{h}$

Neural Radiance Field Revisited A Neural Radiance Field (NeRF) is a mapping from 3D location $\mathbf{x} = (x, y, z)$ and 2D viewing direction $\mathbf{d}$ to density σ and RGB color value $\mathbf{c} = (r, g, b)$. The mapping function can be written as

$$F_\Theta : (\mathbf{x}, \mathbf{d}) \rightarrow (\mathbf{c}, \sigma), \tag{10}$$

A camera ray can be denoted as $\boldsymbol{r}(t) = \boldsymbol{o} + t\boldsymbol{d}$, where the ray origin is represented by $\boldsymbol{o}$. Then classical volume rendering is applied. The pixel color for ray $\boldsymbol{r}$ is computed by:

$$\hat{C}(\mathbf{r}) = \mathcal{R}(\mathbf{r}, \mathbf{c}, \sigma) = \sum_{i=1}^{N} T_i \left(1 - \exp\left(-\sigma_i \delta_i\right)\right) \mathbf{c}_i; T_i = \exp\left(-\sum_{j=1}^{i-1} \sigma_j \delta_j\right), \tag{11}$$

where $\delta_i = t_{i+1} - t_i$ is the distance between two neighboring sampling points, and $T_i = \exp\left(-\sum_{j=1}^{i-1} \sigma_j \delta_j\right)$ is the transmittance for each point along the ray.

Human Deformation Inspired by [34,42], We learn to unwrap the performer's observed poses in the observation space to a canonical T-pose represented by SMPL [26]. Therefore, sampling points can be queried in canonical space and warpped back to the observation space as a deforming strategy guided by poses.

Assuming there is a 3D sampling point $\mathbf{x}$ in frame i, Initially, we identify the SMPL mesh vertex closest to the target point by employing the Nearest Neighbor algorithm. Subsequently, we transform this vertex into the canonical space by applying the explicit rigid inverse transformation derived from the SMPL mesh:

$$\mathcal{T}(\boldsymbol{M}_i, \mathbf{x}) = \left(\sum_{k=1}^{K} w_{k,i} G_k(\boldsymbol{M}_i, \mathbf{J}_i) \right)^{-1}, \tag{12}$$

where $w_{k,i}$ serves as a blending weight for $k-th$ joint in the frame i's blending weight volume obtained from SMPL. The transformation of the $k-th$ joint is represented by $G_k(\boldsymbol{M}_i, \mathbf{J}_i)$, and the 3D location of K is represented by joints $\mathbf{J} \in \mathcal{R}^{3K}$. Then we are able to calculate $\mathbf{x}$ in canonical coordinate

$$\mathbf{x}_c = \mathcal{T}(\boldsymbol{\theta}_i, \mathbf{x}) \cdot \mathbf{x}, \tag{13}$$

Canonical Human Model Once the parameters are obtained, the remaining task is to retrieve the corresponding points within a canonical volumetric representation. The volume F_c is realized by a single MLP which can output color $\mathbf{c}$ and density σ of a 3D point $\mathbf{x}$ in the canonical space

$$F_c(\mathbf{x}) = \mathrm{MLP}_c(\mathbf{x}_c, L_i). \tag{14}$$

A set of latent code $L_i = \{z_1, z_2, \ldots, z_{6890}\}$ is also embedded on vertices of the SMPL model for each frame (z has a dimenson of 16) that implicitly ensures temporal consistency in the reconstructed human model. Eventually, we query the color for a 3D point in observation space: Ψ_{h}:

$$\mathbf{c}_i(\mathbf{x}) = \mathrm{MLP}_c(\mathcal{T}(\boldsymbol{M}_i, \mathbf{x}) \cdot \mathbf{x}, L_i). \tag{15}$$

Training the Human Model Leveraging our dynamic human model (Eq. 15), we can incorporate it with volume rendering (Eq. 11). For each frame i, we can synthesize images from specific viewpoints, while constraining the sampling point range within the 3D bounding boxes of the corresponding SMPL meshes. Over the training procedure, MLP_c and L_i are jointly optimize by minimizing difference between the true pixel color $\mathbf{C}_i(\mathbf{r})$ and the rendered pixel color $\hat{\mathbf{C}}_i(\mathbf{r})$,

$$L_{\mathrm{rgb}} = \sum_{r \in \mathcal{S}} \left\| \hat{\mathbf{C}}_i(\mathbf{r}) - \mathbf{C}_i(\mathbf{r}) \right\|^2 \tag{16}$$

where $\mathcal{S}$ is the represented as a set of sampling rays which intersecting with human's 3D bounding box.

3.4 Scene model $\Psi_{\mathbf{s}}$

Mip-NeRF360 [4] is performed as the background scene backbone Ψ_{s}: Improved from Mip-NeRF [3], each ray is split into intervals $T_i = [t_i, t_{i+1})$, and for each conical frustum, the mean $\boldsymbol{\mu}$, as well as the covariance $\boldsymbol{\Sigma}$ are calculated. Following [53], here is a contraction function.

$$\xi(\mathbf{x}) = \begin{cases} \mathbf{x} & \|\mathbf{x}\| \leq 1 \\ \left(2 - \frac{1}{\|\mathbf{x}\|}\right)\left(\frac{\mathbf{x}}{\|\mathbf{x}\|}\right) & \|\mathbf{x}\| > 1 \end{cases} \tag{17}$$

Afterwards, we can calculate contracted Gaussian parameters $(\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}})$:

$$(\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}}) = \left(\xi(\boldsymbol{\mu}), \mathbf{J}_\xi(\boldsymbol{\mu})\boldsymbol{\Sigma}\mathbf{J}_\xi(\boldsymbol{\mu})^{\mathrm{T}}\right) \tag{18}$$

here the item $\mathbf{J}_\xi(\boldsymbol{\mu})$ is a Jacobian of ξ of $\boldsymbol{\mu}$. Hence, the whole model is:

$$\Psi_{\mathrm{s}}(\hat{\eta}(\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}})) \longmapsto (\mathbf{c}, \sigma) \tag{19}$$

Note that $\eta(\cdot)$ is integrated positional encoding introduced in [4].

3.5 Layered Representation in Training and Rendering

We adopted a neural layered representation similar to [15, 35] for joint training and compositional volume rendering, so as to train the human model Ψ_{h} and the background model Ψ_{s} jointly without 2D segmentation masks. Compared to [16], we no longer need foreground human masks for training, and avoid artifacts due to α-composition during rendering in [16].

Layered Ray Sampling Strategy During the human motion capture, we obtain the per-frame SMPL meshes under the 3D world coordinate, along with the corresponding 3D bounding boxes. We utilize this 3D bounding box as a spatial clue for parsing the 3D scene. Then we adopt a layered ray sampling strategy. For a camera ray that intersect with the human, we sample points inside the bounding box for training our human model Ψ_{h}, and sample points outside the bounding box for training our background model Ψ_{s}.

Compositional Volume Rendering. With the help of the layered Ray Sampling Strategy, we query the Ψ_{h} and Ψ_{s} respectively and get the composited rendering result. We annotate $C^h(\mathbf{r})$ as the accumulated color of samples inside the 3D bounding box, and $C^s(\mathbf{r})$ as the accumulated color of samples outside the 3D bounding box. The rendered color of a pixel $\hat{C}(\mathrm{r})$ can be computed by:

$$\hat{C}(\mathbf{r}) = C^h(\mathbf{r}) + \left(1 - \alpha^h(\mathbf{r})\right) C^s(\mathbf{r}) \tag{20}$$

Here the α^h is the alpha values of the human layer

$$\alpha_i^h = 1 - \exp\left(-\sigma_i^h \delta_i^h\right) \tag{21}$$

Joint Training Without Segmentation Masks We uses L2-norm as a loss function for supervised training:

$$\mathcal{L} = \sum_{\mathrm{r} \in \mathcal{R}} \left(\left\| C(\mathrm{r}) - \hat{C}(\mathrm{r}) \right\|_2^2 \right) \tag{22}$$

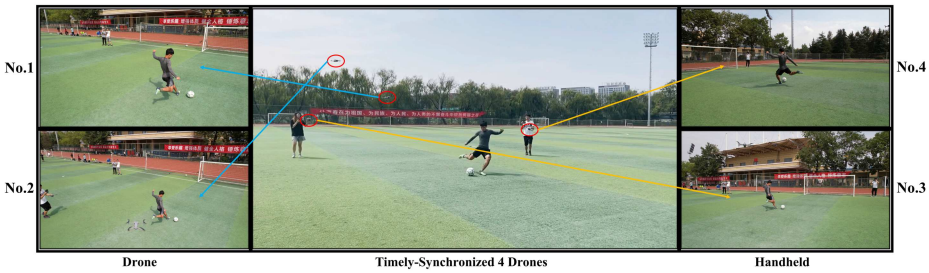

Fig. 3: Illustration of 4-drone dataset collection in AerialRecon.

4 Real World Experiment

AerialRecon Dataset: We conduct extensive real-world experiments on our AerialRecon Dataset, which covers diverse types of real sports scenes, captured by 2 or 4 synchronized drone cameras. It comprises 400 drone video clips and more than 120K images. Each scene has multi-view images registered to the same point cloud under the world coordinate system. Every clip from the 2 or 4 drone cameras is manually synchronized using a movie clapperboard. With videos captured at 30FPS, the time synchronization error is less than 33 milliseconds.
Data Collection: As illustrated in Fig. 3, during the capture process, two collection methods are applied, which is for ablation studies. We ensure all drones are configured with the same photography parameters to maintain consistency.
1. Drone Drones No. 1 and No. 2 automatically orbit the athlete in an arc trajectory under the same constant linear speed but different phases and heights. We only utilize a single video from drone No. 1 for training purposes, while drone No. 2 provides test views for quantitative evaluation. Consequently, we obtain image pairs from two distinct viewpoints at any given time frame.
2. Handheld drones No. 3 and No. 4 are operated by two photographers moving along different sides of the athlete. Drone No. 3 is utilized for training the handheld system, while drone No. 4 is exclusively used for testing.

4.1 Camera Trajectory

We select over 300 trajectories (each has 600 consecutive frames) in 25 real sports scenes on AerialRecon for the ablation study of our module design in calibration. We define two failure cases: **Registration failure:** After calibration, the number of registered cameras is less than the number of total frames (600). **Trajectory Failure:** Obvious breaks in the visualization of camera trajectories. Fig.4 demonstrates the case. Table.1 reports the number of wrong trajectories removing different components, which demonstrates the improvement of robustness using sequential matching and DMC constraints.

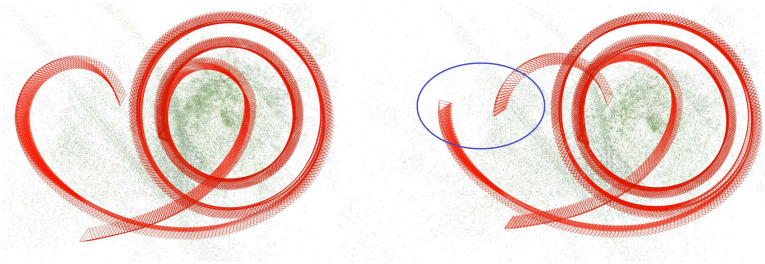

Fig. 4: Visualization of obvious break in trajectory failures

Table 1: Ablation in Calibration

Method	#Total Failure	#Regis.Failure	#Traj.Failure
COLMAP	52	27	25
ours	**0**	**0**	**0**
w/o DMC	37	14	23
w/o sequential matching	6	6	0

4.2 3D Human Pose Estimation

We use the mean average precision (AP) under different threshold (AP^{50}, AP^{75}) defined by the COCO challenge [24] for quantitative evaluation. Specifically, we use drone No.1 for pose estimation, and we select consecutive 100 frames from the test view camera (drone No.2) from each scene in the 25 scenes and carefully annotate the 17 2D keypoints (referring to COCO 2017 dataset [24]) per frame as the corresponding GT keypoints. We use our MoCap system to estimate 3D pose skeletons only using the input 25×100 frames from drone No.1 and project the estimated 3D skeletons to their corresponding test view (drone No.2) to calculate their keypoint similarity with GT keypoints and measure the metrics in Table 2. Here we use different methods for comparison including: 1) OpenPose [5] 2) using 2D CNN rather than 2D transformer as 2D pose detector 3) removing sequential smoothness in optimization 4) MoCap system in NeuMan [16].

Table 2: Comparison in Human Motion Capture Methods

Method	AP	AP^{50}	AP^{75}
OpenPose [5]	62.7	70.2	66.9
ours	**67.9**	**89.2**	**78.3**
ours w/o 2D transfomer	64.5	76.0	70.1
ours w/o sequential smoothness	66.3	83.4	74.2
NeuMan [16]	66.7	83.2	76.1

4.3 Reconstruction and Rendering

We compare with monocular SOTA systems designed for outdoor settings [16, 21,25], and test them for novel view rendering on 3 datasets. On AerialRecon, all the methods use the same training data from drone No.1 and the same test view from drone No.2 for calculating metrics. Our method outperforms other methods in qualitative (See Fig.5) and quantitative (See Table. 3) results. NeRF-T [21] does not use human pose priors and is difficult to reconstruct a fast-moving human body. NeuMan [16] Uses Vanillar-NeRF to reconstruct the background, displaying obvious blurs in the scene. Besides, NeuMan [16] and HOSNeRF [25] have misaligned human body positions compared with GT, and some problems on the scale (e.g. penetrated human feet in climbing). We also compare our method with [16] and [25] on their datasets in Table. 4 and Table. 5. Results show we are superior than or on-par with SOTA methods on their datasets.

Table 3: Quantitative Comparison of SOTA Methods on AerialRecon Dataset

	Climb			Jogging			Kungfu			Football		
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
HOSNeRF [25]	21.80	0.743	0.271	21.28	0.724	0.314	21.07	0.698	0.277	20.92	0.745	0.291
NeRF-T [21]	16.37	0.563	0.389	14.82	0.497	0.617	13.62	0.378	0.633	17.38	0.591	0.379
NeuMan [16]	19.36	0.658	0.363	19.76	0.623	0.352	19.31	0.657	0.579	20.01	0.680	0.315
Ours	**24.24**	**0.891**	**0.203**	**22.94**	**0.841**	**0.243**	**23.29**	**0.874**	**0.237**	**22.41**	**0.793**	**0.268**

Fig. 5: Qualitative Comparison of SOTA Methods on AerialRecon Dataset

Table 4: Quantitative Comparison of SOTA Methods on NeuMan Dataset [16]

	SEATTLE			PARKING			BIKE		
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
HOSNeRF [25]	26.35	0.893	0.135	**26.98**	0.879	0.158	26.03	0.901	0.177
NeuMan [16]	24.01	0.792	0.254	26.07	0.803	0.277	25.90	0.823	0.236
Ours	**27.32**	**0.925**	**0.112**	26.80	**0.912**	**0.131**	**27.32**	**0.939**	**0.159**
	JOGGING			LABS			CITRON		
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
HOSNeRF [25]	24.97	0.878	0.182	24.89	0.892	0.175	24.09	0.893	0.184
NeuMan [16]	23.69	0.725	0.306	**25.61**	0.862	0.254	25.31	0.811	0.263
Ours	**26.03**	**0.907**	**0.158**	25.53	**0.903**	**0.173**	**26.18**	**0.928**	**0.157**

Table 5: Quantitative Comparison of SOTA Methods on HOSNeRF Dataset [25]

	BACKPACK			TENNIS			SUITCASE		
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
HOSNeRF	22.53	0.775	0.241	**24.50**	**0.910**	0.323	**21.79**	**0.839**	**0.389**
NeuMan	21.68	0.613	0.458	23.93	0.802	0.377	21.33	0.635	0.433
Ours	**23.58**	**0.806**	**0.229**	23.95	0.827	**0.308**	21.67	0.821	0.412
	PLAYGROUND			DANCE			LOUNGE		
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
HOSNeRF	22.59	0.793	0.346	22.67	0.802	**0.250**	27.84	0.967	0.218
NeuMan	21.93	0.672	0.427	22.03	0.603	0.389	27.79	0.923	0.253
Ours	**24.56**	**0.852**	**0.279**	**23.54**	**0.833**	0.297	**28.91**	**0.975**	**0.157**

4.4 Ablation Study Between Drone and Handheld

As we illustrated, we have 4 drone videos at the same time for certain scenes. Specifically, we use drone No.3 for training Handheld system, and drone No.4 for providing test view GT. We use drone No.1 for training Drone system, and drone No.2 for providing test view GT. This is to clarify the advantages of using an orbiting drone in our system, rather than a handheld camera used in [16, 25].

3D Human Pose Estimation For each system, we project the estimated 3D human SMPL mesh to the corresponding render view GT image. In some handheld videos, the human body could fail to align. For example, in Fig. 6, the climber's arms and hands have bad results, as extreme view directions by ground cameras increase the difficulties of 2D keypoints detection. Also, the video shot by a handheld camera mostly contains only one-sided observations of the athlete, which aggravates self-occlusions, hindering accurate estimation.

Novel View Rendering Qualitative comparisons are shown in Fig. 7. Compared with an orbiting drone, a handheld camera cannot guarantee to cover all surrounding 360° viewing angles, causing artifacts in human and distorted scene details in backgrounds. We also report the metrics in Table. 6. Consequently, the shortcoming of existing SOTA systems using a handheld camera as input like [16, 25] constrain their application in real-world sports.

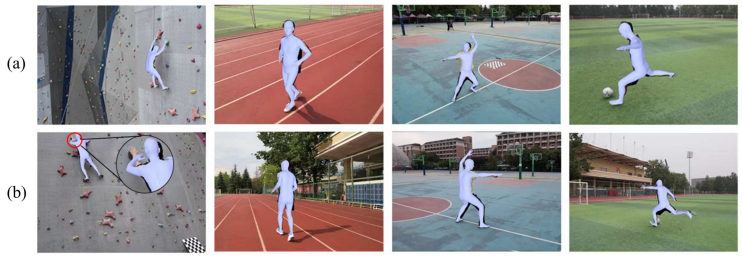

Fig. 6: Qualitative MoCap comparison between a)drone videos and b) handheld videos.

Table 6: Ablation Study Between Drone and Handheld System on AerialRecon

	Climb			Jogging			Kungfu			Football		
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
Drone	**22.18**	**0.715**	**0.289**	**21.54**	**0.723**	**0.324**	**20.84**	**0.698**	**0.357**	**21.03**	**0.695**	**0.309**
Handheld	20.34	0.677	0.337	17.28	0.587	0.412	18.97	0.626	0.633	15.34	0.492	0.625

5 Dataset and Future Work

Our new dataset AerialRecon paves the way for future work. It fills the gap in multi-view dataset of real-world outdoor sports scenes. Extension work could further support tracking and reconstruction of moving objects.

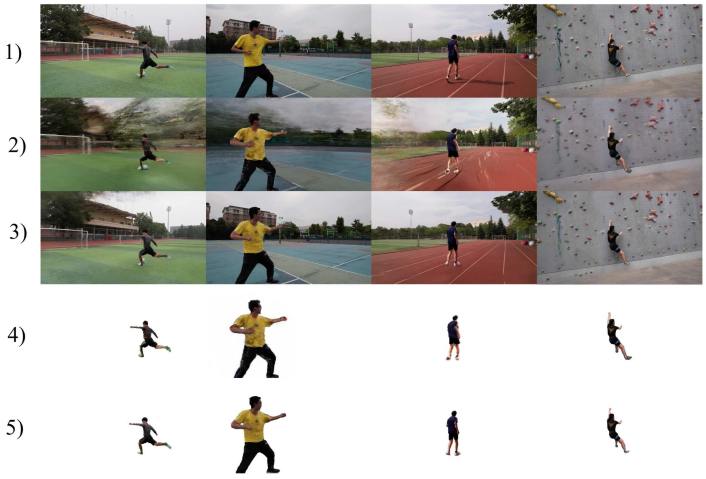

Fig. 7: Novel view rendering comparison between Handheld and Drone. Here: 1)GT 2)Handheld 3)Drone 4)Handheld human only 5)Drone human only

Acknowledgements

I would like to express my sincere gratitude to everyone who was willing to help collect data and discuss with me in depth.

References

1. Ahmad, A., Price, E., Tallamraju, R., Saini, N., Lawless, G., Ludwig, R., Martinovic, I., Bülthoff, H.H., Black, M.J.: Aircap – aerial outdoor motion capture. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2019), Workshop on Aerial Swarms (Nov 2019)
2. Alldieck, T., Magnor, M., XuWeipeng, Theobalt, C., Pons-Moll, G.: Detailed human avatars from monocular video. In: 2018 International Conference on 3D Vision (3DV). pp. 98–109. IEEE (2018)
3. Barron, J.T., Mildenhall, B., Tancik, M., Hedman, P., Martin-Brualla, R., Srinivasan, P.P.: Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. ICCV (2021)
4. Barron, J.T., Mildenhall, B., Verbin, D., Srinivasan, P.P., Hedman, P.: Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5470–5479 (2022)
5. Cao, Z., Hidalgo Martinez, G., Simon, T., Wei, S., Sheikh, Y.A.: Openpose: Realtime multi-person 2d pose estimation using part affinity fields. IEEE Transactions on Pattern Analysis and Machine Intelligence (2019)
6. Fridovich-Keil, S., Meanti, G., Warburg, F.R., Recht, B., Kanazawa, A.: K-planes: Explicit radiance fields in space, time, and appearance. In: CVPR (2023)
7. Furukawa, Y., Ponce, J.: Accurate, dense, and robust multiview stereopsis. IEEE transactions on pattern analysis and machine intelligence **32**(8), 1362–1376 (2009)
8. Goel, S., Pavlakos, G., Rajasegaran, J., Kanazawa*, A., Malik*, J.: Humans in 4D: Reconstructing and tracking humans with transformers. In: International Conference on Computer Vision (ICCV) (2023)
9. Guo, C., Jiang, T., Chen, X., Song, J., Hilliges, O.: Vid2avatar: 3d avatar reconstruction from videos in the wild via self-supervised scene decomposition. In: Computer Vision and Pattern Recognition (CVPR) (2023)
10. Guo, K., Xu, F., Yu, T., Liu, X., Dai, Q., Liu, Y.: Real-time geometry, albedo, and motion reconstruction using a single rgb-d camera. ACM Transactions on Graphics (ToG) **36**(4), 1 (2017)
11. He, T., Collomosse, J., Jin, H., Soatto, S.: Geo-pifu: Geometry and pixel aligned implicit functions for single-view human reconstruction. Advances in Neural Information Processing Systems **33**, 9276–9287 (2020)
12. Ho, C., Jong, A., Freeman, H., Rao, R., Bonatti, R., Scherer, S.: 3d human reconstruction in the wild with collaborative aerial cameras. In: 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 5263–5269. IEEE (2021)
13. Hu, S., Liu, Z.: Gauhuman: Articulated gaussian splatting from monocular human videos. arXiv preprint arXiv:2312.02973 (2023)
14. Jena, R., Iyer, G.S., Choudhary, S., Smith, B., Chaudhari, P., Gee, J.: Splatarmor: Articulated gaussian splatting for animatable humans from monocular rgb videos. arXiv preprint arXiv:2311.10812 (2023)

15. Jiakai, Z., Xinhang, L., Xinyi, Y., Fuqiang, Z., Yanshun, Z., Minye, W., Yingliang, Z., Lan, X., Jingyi, Y.: Editable free-viewpoint video using a layered neural representation. In: ACM SIGGRAPH (2021)
16. Jiang, W., Yi, K.M., Samei, G., Tuzel, O., Ranjan, A.: Neuman: Neural human radiance field from a single video. In: Proceedings of the European conference on computer vision (ECCV) (2022)
17. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering. ACM Transactions on Graphics **42**(4) (2023)
18. Kiciroglu, S., Rhodin, H., Sinha, S.N., Salzmann, M., Fua, P.: Activemocap: Optimized viewpoint selection for active human motion capture. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2020)
19. Li, P., Xu, Y., Wei, Y., Yang, Y.: Self-correction for human parsing. IEEE Transactions on Pattern Analysis and Machine Intelligence (2020)
20. Li, Y., Mao, H., Girshick, R., He, K.: Exploring plain vision transformer backbones for object detection. In: European Conference on Computer Vision. pp. 280–296. Springer (2022)
21. Li, Z., Niklaus, S., Snavely, N., Wang, O.: Neural scene flow fields for space-time view synthesis of dynamic scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2021)
22. Li, Z., Wang, Q., Cole, F., Tucker, R., Snavely, N.: Dynibar: Neural dynamic image-based rendering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2023)
23. Lin, H., Peng, S., Xu, Z., Yan, Y., Shuai, Q., Bao, H., Zhou, X.: Efficient neural radiance fields for interactive free-viewpoint video. In: SIGGRAPH Asia Conference Proceedings (2022)
24. Lin, T.Y., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick, C.L., Dollár, P.: Microsoft coco: Common objects in context (2014), `http://arxiv.org/abs/1405.0312`, cite arxiv:1405.0312Comment: 1) updated annotation pipeline description and figures; 2) added new section describing datasets splits; 3) updated author list
25. Liu, J.W., Cao, Y.P., Yang, T., Xu, E.Z., Keppo, J., Shan, Y., Qie, X., Shou, M.Z.: Hosnerf: Dynamic human-object-scene neural radiance fields from a single video. arXiv preprint arXiv:2304.12281 (2023)
26. Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J.: Smpl: A skinned multi-person linear model. ACM transactions on graphics (TOG) **34**(6), 1–16 (2015)
27. Lowe, D.G.: Object recognition from local scale-invariant features. In: Proceedings of the seventh IEEE international conference on computer vision. vol. 2, pp. 1150–1157. Ieee (1999)
28. Martin-Brualla, R., Radwan, N., Sajjadi, M.S., Barron, J.T., Dosovitskiy, A., Duckworth, D.: Nerf in the wild: Neural radiance fields for unconstrained photo collections. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7210–7219 (2021)
29. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. Communications of the ACM **65**(1), 99–106 (2021)
30. Newcombe, R.A., Fox, D., Seitz, S.M.: Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In: CVPR. pp. 343–352 (2015)

31. Pandey, R., Tkach, A., Yang, S., Pidlypenskyi, P., Taylor, J., Martin-Brualla, R., Tagliasacchi, A., Papandreou, G., Davidson, P., Keskin, C., et al.: Volumetric capture of humans with a single rgbd camera via semi-parametric learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9709–9718 (2019)
32. Park, K., Sinha, U., Hedman, P., Barron, J.T., Bouaziz, S., Goldman, D.B., Martin-Brualla, R., Seitz, S.M.: Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields. ACM Trans. Graph. **40**(6) (dec 2021)
33. Peng, S., Zhang, S., Xu, Z., Geng, C., Jiang, B., Bao, H., Zhou, X.: Animatable neural implicit surfaces for creating avatars from videos. arXiv preprint arXiv:2203.08133 (2022)
34. Peng, S., Zhang, Y., Xu, Y., Wang, Q., Shuai, Q., Bao, H., Zhou, X.: Neural body: Implicit neural representations with structured latent codes for novel view synthesis of dynamic humans. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9054–9063 (2021)
35. Qing, S., Chen, G., Qi, F., Sida, P., Wenhao, S., Xiaowei, Z., Hujun, B.: Novel view synthesis of human interactions from sparse multi-view videos. In: SIGGRAPH Conference Proceedings (2022)
36. Saini, N., Bonetto, E., Price, E., Ahmad, A., Black, M.J.: AirPose: Multi-view fusion network for aerial 3D human pose and shape estimation. IEEE Robotics and Automation Letters **7**(2), 4805 – 4812 (Apr 2022). https://doi.org/10.1109/LRA.2022.3145494, also accepted and presented in the 2022 IEEE International Conference on Robotics and Automation (ICRA)
37. Saito, S., Huang, Z., Natsume, R., Morishima, S., Kanazawa, A., Li, H.: Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2304–2314 (2019)
38. Saito, S., Simon, T., Saragih, J., Joo, H.: Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 84–93 (2020)
39. Schonberger, J.L., Frahm, J.M.: Structure-from-motion revisited. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4104–4113 (2016)
40. Shen, W., Yin, W., Wang, H., Wei, C., Cai, Z., Yang, L., Lin, G.: Hmr-adapter: A lightweight adapter with dual-path cross augmentation for expressive human mesh recovery. In: ACM Multimedia 2024
41. Tallamraju, R., Saini, N., Bonetto, E., Pabst, M., Liu, Y.T., Black, M., Ahmad, A.: Aircaprl: Autonomous aerial human motion capture using deep reinforcement learning. IEEE Robotics and Automation Letters **5**(4), 6678 – 6685 (Oct 2020). https://doi.org/10.1109/LRA.2020.3013906, https://ieeexplore.ieee.org/document/9158379, also accepted and presented in the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).
42. Weng, C.Y., Curless, B., Srinivasan, P.P., Barron, J.T., Kemelmacher-Shlizerman, I.: Humannerf: Free-viewpoint rendering of moving people from monocular video. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16210–16220 (2022)
43. Wu, G., Yi, T., Fang, J., Xie, L., Zhang, X., Wei, W., Liu, W., Tian, Q., Xinggang, W.: 4d gaussian splatting for real-time dynamic scene rendering. arXiv preprint arXiv:2310.08528 (2023)

44. Wu, T., Zhong, F., Tagliasacchi, A., Cole, F., Oztireli, C.: D^ 2nerf: Self-supervised decoupling of dynamic and static objects from a monocular video. Advances in Neural Information Processing Systems **35**, 32653–32666 (2022)
45. Xu, H., Alldieck, T., Sminchisescu, C.: H-nerf: Neural radiance fields for rendering and temporal reconstruction of humans in motion. Advances in Neural Information Processing Systems **34**, 14955–14966 (2021)
46. Xu, L., Cheng, W., Guo, K., Han, L., Liu, Y., Fang, L.: Flyfusion: Realtime dynamic scene reconstruction using a flying depth camera. IEEE transactions on visualization and computer graphics **27**(1), 68–82 (2019)
47. Xu, L., Liu, Y., Cheng, W., Guo, K., Zhou, G., Dai, Q., Fang, L.: Flycap: Markerless motion capture using multiple autonomous flying cameras. IEEE transactions on visualization and computer graphics **24**(8), 2284–2297 (2017)
48. Yang, B., Zhang, Y., Xu, Y., Li, Y., Zhou, H., Bao, H., Zhang, G., Cui, Z.: Learning object-compositional neural radiance field for editable scene rendering. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 13779–13788 (2021)
49. Yang, Z., Gao, X., Zhou, W., Jiao, S., Zhang, Y., Jin, X.: Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction. arXiv preprint arXiv:2309.13101 (2023)
50. Ye, V., Pavlakos, G., Malik, J., Kanazawa, A.: Decoupling human and camera motion from videos in the wild. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2023)
51. Yu, T., Zheng, Z., Guo, K., Zhao, J., Dai, Q., Li, H., Pons-Moll, G., Liu, Y.: Doublefusion: Real-time capture of human performances with inner body shapes from a single depth sensor. In: CVPR. pp. 7287–7296 (2018)
52. Yuan, Y., Iqbal, U., Molchanov, P., Kitani, K., Kautz, J.: Glamr: Global occlusion-aware human mesh recovery with dynamic cameras. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11038–11049 (2022)
53. Zhang, K., Riegler, G., Snavely, N., Koltun, V.: Nerf++: Analyzing and improving neural radiance fields. arXiv preprint arXiv:2010.07492 (2020)
54. Zhou, X., Liu, S., Pavlakos, G., Kumar, V., Daniilidis, K.: Human motion capture using a drone. In: 2018 IEEE international conference on robotics and automation (ICRA). pp. 2027–2033. IEEE (2018)