iHuman3D: Intelligent Human Body 3D Reconstruction using a Single Flying Camera

Wei Cheng*
TBSI, Tsinghua University
Hong Kong Univ. of Sci. and Tech.
wchengad@connect.ust.hk

Lan Xu*
TBSI, Tsinghua University
Hong Kong Univ. of Sci. and Tech.
lxuan@connect.ust.hk

Lei Han
TBSI, Tsinghua University
Hong Kong Univ. of Sci. and Tech.
lhanaf@connect.ust.hk

Yuanfang Guo†
TBSI, Tsinghua University
eeandyguo@connect.ust.hk

Lu Fang†
TBSI, Tsinghua University
fanglu@sz.tsinghua.edu.cn

ABSTRACT

Aiming at autonomous, adaptive and real-time human body reconstruction technique, this paper presents *iHuman3D*: an intelligent human body 3D reconstruction system using a single aerial robot integrated with an RGB-D camera. Specifically, we propose a real-time and active view planning strategy based on a highly efficient ray casting algorithm in GPU and a novel information gain formulation directly in TSDF. We also propose the human body reconstruction module by revising the traditional volumetric fusion pipeline with a compactly-designed non-rigid deformation for slight motion of the human target. We unify both the active view planning and human body reconstruction in the same TSDF volume-based representation. Quantitative and qualitative experiments are conducted to validate that the proposed *iHuman3D* system effectively removes the constraint of extra manual labor, enabling real-time and autonomous reconstruction of human body.

CCS CONCEPTS

• **Human-centered computing** → *Virtual reality*;

KEYWORDS

Flying Camera, Human 3D Reconstruction, TSDF, Next Best View

ACM Reference format:
Wei Cheng, Lan Xu, Lei Han, Yuanfang Guo, and Lu Fang. 2018. iHuman3D: Intelligent Human Body 3D Reconstruction using a Single Flying Camera. In *Proceedings of 2018 ACM Multimedia Conference, Seoul, Republic of Korea, October 22–26, 2018 (MM '18),* 9 pages.
https://doi.org/10.1145/3240508.3240600

*Equal contribution
†The corresponding authors are Yuanfang Guo (eeandyguo@connect.ust.hk) and Lu Fang (fanglu@sz.tsinghua.edu.cn).
Acknowledgement: This work is supported in part by Natrual Science Foundation of China (NSFC) under contract No.61722209 and 61331015.
For more information, please refer to our web page http://www.luvision.net/FlyFusion

MM '18, October 22–26, 2018, Seoul, Republic of Korea

ACM ISBN 978-1-4503-5665-7/18/10…$15.00
https://doi.org/10.1145/3240508.3240600

1 INTRODUCTION

Robust perception and understanding of humans present can enable numerous applications, such as human robot interaction, human motion analysis and recognition, computer games, and virtual reality. One of the key problems is to reconstruct a realistic 3D models of human bodies. Benefit from the emergence of depth sensor that can capture depth and image data at video frame rate, human body reconstruction based on a depth sensor has received great attention in the fields of multimedia, computer vision, graphics and robotics etc. Among them, consumer-level depth camera like Kinect is widely used for 3D reconstruction due to the advantages of being low-price, compact and portable [19, 26, 34].

To scan a full human body, a consumer-grade depth camera needs to work around 2 meters away from the target given the small FOV, leading to the little valid geometry information that can be captured in the depth map. While the information of multiple frames can be fused to enhance the final resolution [8], the reconstruction result is still over-smooth. Tong *et al.* [29] presented a system to scan 3D full human body shapes using multiple Kinects. However, all these methods are significantly constrained with a static camera array or human handheld cameras to follow the target actors. In this work, we propose ***iHuman3D*** – a real-time human body 3D reconstruction scheme with a single flying camera, which is an autonomous aerial robot equipped with a low weight and easily power-supplied depth camera Asus Xtion Pro to capture RGB-D video stream with acceptable quality. Specifically, *iHuman3D* consists of three modules: (i) human body reconstruction, (ii) active view planning, and (iii) robot positioning mechanism, similar as [12, 18].

For the human body reconstruction module, we maintain a human-centered Truncated Signed Distance Function (TSDF) [9] volume similar as KinectFusion [26], which aligns the temporal information of the dense 3D input data from the depth sensor and fuse a human body model in real-time. Note that consecutive work after KinectFusion [26] in the computer vision and computer graphics communities tends to focus on embedding new information for the final reconstructed model, but rarely analyzes the influence of the shooting angles for the dynamic reconstruction. We, however, take the next best view (NBV) evaluation [7, 28] into account in the active view planning module, so as to achieve autonomous and adaptive reconstruction. To be consistent with our volumetric fusion pipeline for human body reconstruction, we adopt the

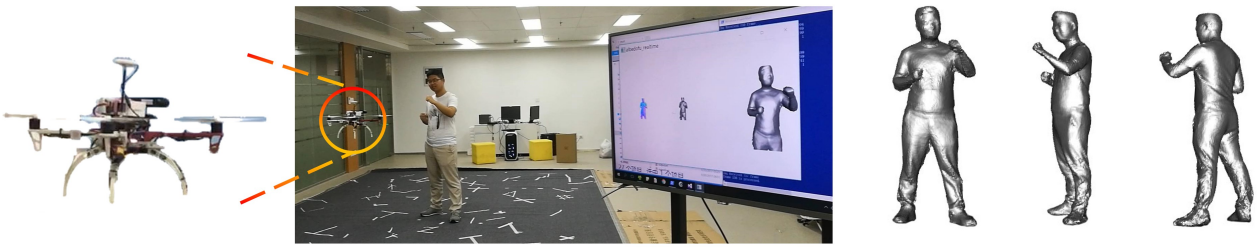


Figure 1: Realtime 3D human body reconstruction by a flying camera. Left: aerial robot mounted with a depth camera. Middle: live demo. Right: high quality realistic live mesh.

volumetric NBV framework, which accumulates the information gain (IG) along all the rays casted to the volumetric representation [4, 10, 12, 18]. In particular, we formulate the information gain of the volumetric NBV directly in the TSDF volume, and a highly parallel and efficient algorithm to calculate the IG based on modern GPU hardware is introduced to enable the real-time performance of *iHuman3D*. In summary, the spotlights of the proposed *iHuman3D* include:

- Adaptive: our scheme is the first work, to our knowledge, to employ an autonomous flying robot for human body reconstruction in an adaptive way, which removes the constraints and labor of previous available systems in terms of the performers, the capturers and the environments. Moreover, the active sensing strategy enables the adaptive reconstruction of human body.
- Real-time: both the geometric reconstruction module and the active view planning module are tightly combined in a uniform TSDF volumetric representation, all of the human model reconstruction, the light-weight non-rigid motion regularizer and the highly efficient IG calculating algorithm (achieving 200x speed up as shown in Sec. 4.1) run in real-time.

Given above distinctiveness, we believe our work bridges the gap among efficiency, accuracy and adaptability for human 3D reconstruction, and will further promote both multimedia and robotics communities in areas of human robot interaction, human motion analysis and recognition etc.

2 RELATED WORK

This section presents an overview of research works related to our *iHuman3D* system. We first provide an overview of the human model reconstruction technologies, followed by an overview of representative active view planning algorithms.

2.1 Human model Reconstruction

Acquiring 3D geometric content from real world is an essential task for many applications in robotics domain, computer vision and computer graphics communities. Detailed human models can be created using 3D scanning devices, such as structured light or laser scan [1]. However, such devices are too expensive and often require expert knowledge for the operation. The multi-view method [11, 20] can get very impressive results. But this kind of methods is computationally expensive. Some researchers have tried to use comsumer-grade depth sensors as 3D scanners for accurate real-time mapping of complex scenes [19, 26, 32, 34]. These methods utilized the TSDF volume [9] for both representing the the geometry information and analyze the camera localization information.

To capture the full human body, Tong *et al.* [29] used multiple Kinects to scan 3D full human body shapes. However, such methods suffer from the fixed capture volume constraint. Some researchers [15, 37] utilized human handheld cameras to follow and reconstruct the human body, which have to rely on extra manual labor. [33] proposed to reconstruct accurate human pose from a linear system and [3] introduce a data-driven hybrid strategy to tackle the challenges of occlusions and input noises.

To eliminate any manual pre-processing like skeleton embedding while achieving appealing human body reconstruction results in real-time, we adopt the volumetric fusion pipeline, where a compactly-designed non-rigid deformation is further adopted to model the unavoidable slight human motion during reconstruction. Both the data capture and reconstruction modules run fully automatically, without any user interference.

2.2 Active View Planning

NBV based active view planning problem determines new viewpoints for taking sensor measurements to maximize information collection from the current environment, which can date back several decades [2, 7]. Scott *et al.* [28] provided an overview of early approaches while Chen *et al.* [5] provided a survey of more recent work. Scott *et al.* [28] categorized the NBV algorithms into model-based and non-model-based methods.

Model-based methods assume at least an approximation of the scene is known as priori, which may not be available in many real world scenarios. Non-model based methods use relaxed assumptions about the scene, but require that the NBV must be estimated online based on the gathered data. Scott *et al.* [28] classified existing non-model based methods as volumetric or surface-based. In a surface-based approach, the boundaries of the surface are examined for evaluating the NBV [6, 27]. However, it is computationally expensive for more complex operations to the surface representation.

Volumetric non-model based methods, on the other hand, have become popular because they implicitly model the spatial information and facilitate simple visibility operations. NBVs are evaluated by casting rays into the model from the view candidates and examining the traversed voxels, which can be efficiently implemented on parallel computing devices such as GPUs. Vasquez-Gomez *et al.* [31] and Yamauchi *et al.* [36] counted the frontier voxels, defined as the voxels bordering free and unknown space.

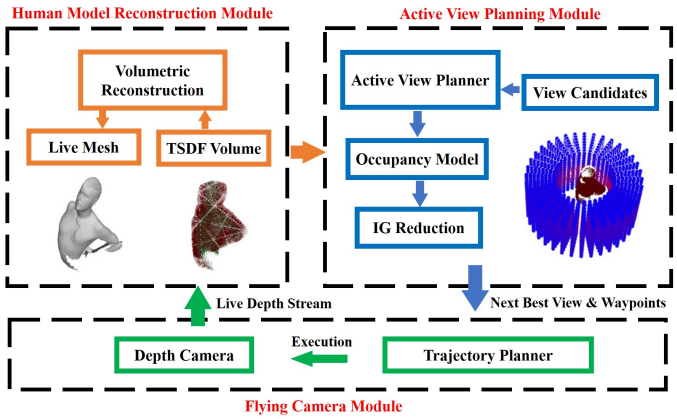


Figure 2: The architecture of *iHuman3D*. Three modules consist with *iHuman3D*, flying camera module captures human's depth stream and executes human scanning; human model reconstruction module absorbs depth stream, estimates the TSDF volume and reconstructs human mesh; active view planning model calculate the TSDF based information gain from view candidates and generates next-best-view and waypoints guiding flying camera's motion. Blue dots surrounding the TSDF are view candidates.

Recent methods either use learning-based evaluation of 3D meshes directly [13] or use probabilistic approaches to define the information gain along the casted ray to the volume. Kriegel *et al.* [21] used the information theoretic entropy to estimate expected information. Recently, Stefan *et al.* [18] and Jeffrey *et al.* [12] proposed a set of information gain formulations and provided a comprehensive comparison among the volumetric information gain metrics for active 3D reconstruction. Jonathan *et al.* [10] proposed an adaptable and probabilistic NBV method without making any assumptions on the reconstructed object. Most recent work [24] proposed a contour based method for NBV selection in exploiting KinectFusion, TSDF is used to classify three kinds of voxels in volume.

For the active view planning module in our system, we choose a volumetric representation for its compactness and efficiency with respect to visibility. We formulate the information gain directly in the TSDF volume. Moreover, a highly parallel and efficient algorithm is proposed to enable real-time performance of our adaptive human body reconstruction system.

3 IHUMAN3D: INTELLIGENT HUMAN BODY 3D RECONSTRUCTION USING A SINGLE FLYING CAMERA

3.1 System Overview

Recall that *iHuman3D* aims for adaptive human body reconstruction using a single aerial robot. As shown in Fig. 1, we adopt a compactly designed aerial robot, equipped with: *NUC* – a mini PC that acts as the brain with computation and control units, *Guidance* [38] – armed with an ultrasonic sensor and stereo cameras working as a navigation system, providing the pose estimation using its internal VO algorithm by fusing the IMU data, and *Xtion* – serving as the 3D sensor device to acquire the RGB-D data (VGA resolution) of the scene. In particular, the aerial robot works around 2 meters away from the captured dynamic target. Such setting is the compromise between the field of view (FOV) and depth accuracy of the RGB-D sensor.

Fig. 1 gives a sketch of the working pipeline of *iHuman3D*. The aerial robot works as a flying camera to capture the depth information in real-time. The captured depth data is streamed via a wireless network connection to a desktop machine that runs our human body reconstruction and active view planning modules. To reduce the required bandwidth for real-time performance, we use data compression based on *zlib* for the depth stream. Then, the view planning result is streamed back to the aerial robot interface via the same network. Both the human body reconstruction and the view planning are performed in a highly parallel way on the modern GPU hardware to enable real-time performance.

The system architecture of *iHuman3D* is illustrated in Fig. 2, which relates to: (i) human body reconstruction module, (ii) active view planning module, and (iii) flying camera module. The reconstruction module fuses the live depth input into the TSDF volumes, and meanwhile provides a real-time mesh visualization result. Based on the real-time live TSDF volume, the active view planning module examines the NBV from all the view candidates in parallel on the modern GPU hardware. For the stability of the whole system, the NBV results are transmitted back to the aerial robot in another fixed frame rate (10fps). In the flying camera module, we use the same robot interface to the hardware platform of the aerial robot as [35], which provides a depth stream with the corresponding camera location information in 30fps. Note that the whole system is synchronized with a common NTP server.

3.2 Active View Planning Module

3.2.1 Occupancy Probability in TSDF Volume. Aiming at real-time next-best-view selection, we follow the pioneer work [24] on information gain calculation in TSDF volume. As defined in [26], two components are stored in TSDF which represents a fusion of the registered depth measurements from frames $1, \cdots, k$ for each voxel $\mathbf{p} \in \mathbb{R}^3$,

$$\mathrm{S}_k(\mathbf{p}) \mapsto [\mathrm{F}_k(\mathbf{p}), \mathbf{W}_k(\mathbf{p}), \mathrm{L}_k(\mathbf{p})], \tag{1}$$

where $\mathrm{F}_k(\mathbf{p})$ is the truncated distance value and $\mathbf{W}_k(\mathbf{p})$ indicates the measurement weight. For each voxel with a distance r from camera center along depth map ray, the distance from depth value is truncated with a range $\pm\mu$ centered at the measurement.

As illustrated in Fig. 3, in TSDF representation, a voxel with high positive value indicates it locates outside from the object surface with high probability to be free, whereas the negative volexs is on the opposite side. Similar as [24], we classify voxels into three sets: unknown set $\mathbb{U}_{1:k}$, occupied set $\mathbb{O}_{1:k}$ and empty set $\mathbb{E}_{1:k}$, according to $\mathrm{F}_k(\mathbf{p})$ and $\mathbf{W}_k(\mathbf{p})$ as follows:

$$\begin{cases} \mathbf{W}_k(\mathbf{p}) = 0 & \rightarrow \text{unknow voxel}, \mathbf{p} \in \mathbb{U}_k \\ \mathbf{W}_k(\mathbf{p}) > 0, \mathrm{F}_k(\mathbf{p}) = 1 & \rightarrow \text{empty voxel}, \mathbf{p} \in \mathbb{E}_k \\ \mathbf{W}_k(\mathbf{p}) > 0, -1 \leq \mathrm{F}_k(\mathbf{p}) < 1 & \rightarrow \text{occupied voxel}, \mathbf{p} \in \mathbb{O}_k. \end{cases} \tag{2}$$

To model occupancy uncertainty for view information gain calculation, we adopt an occupancy probability model, where the occupancy grid mapping integration [25] is used instead of the

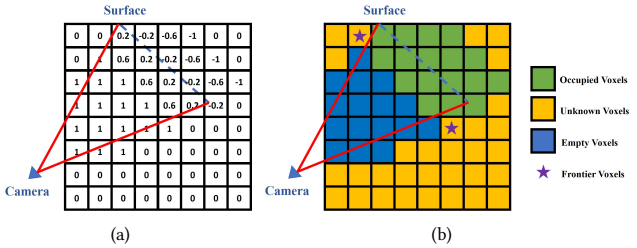


Figure 3: TSDF representation. (a) Voxels are assigned with a truncated distance value along the camera casting ray. (b) Three basic voxel categories based on TSDF values and weights, occupied voxels (green), unknown voxels (yellow) and empty voxels (blue). Frontier voxels are unknown voxels whos neighbour contains both occupied voxels and empty voxels.

TSDF update scheme in [26], i.e.,

$$P(\mathbf{p}|D_{1:k}) = [1 + \frac{1-P(\mathbf{p}|D_k)}{P(\mathbf{p}|D_k)} \frac{1-P(\mathbf{p}|D_{1:k-1})}{P(\mathbf{p}|D_{1:k-1})} \frac{1-P(\mathbf{p})}{P(\mathbf{p})}]^{-1}. \quad (3)$$

Here $P(\mathbf{p}|D_k)$ is the probability given current calibrated depth measurement D_k, $P(\mathbf{p}|D_{1:k})$ and $P(\mathbf{p}|D_{1:k-1})$ are integrated probability via all previous measurements in k and $k-1$ frame, $P(\mathbf{p})$ is a prior probability. We assume that the occupancy of $\mathbf{p} \in \mathbb{O}_k$ in current measurement D_k is a normal distribution according to the new TSDF value $\mathbf{F}_{D_k}(\mathbf{p})$.

$$P(\mathbf{p}|D_k) = exp(-\frac{\mathbf{F}_{D_k}(\mathbf{p})^2}{2\sigma_1^2}). \quad (4)$$

Here we set $\sigma_1 = \mu/3$ to force occupancies distribute inside the truncate band mostly.

Similar to [16], under the assumption of an uniform prior $P(\mathbf{p})$ and the usage of log-odds probability notation, Eqn. 3 can be simplified as:

$$\mathbf{L}(\mathbf{p}|D_{1:k}) = \mathbf{L}(\mathbf{p}|D_{1:k-1}) + \mathbf{L}(\mathbf{p}|D_k). \quad (5)$$

3.2.2 View Information Gain. Given the basic voxel category in Fig. 3, similar to [24][10], the frontier voxels denoted as $\mathbf{f}_i \in \mathbb{F}_{1:k}$ are considered as the unknown voxels which border both empty voxels and occupied voxels. Note that these frontier voxels are near the boundary of the estimated human model, thus we assume the unknown voxels $\mathbf{p} \in \mathbb{U}_{1:k}$ near the frontier voxels may have a high probability to belong to the estimated human model. We then formulate the frontier information as:

$$Q(\mathbf{p}) = \max_{\mathbf{f}_i \in \mathbb{F}_{1:k}} exp(\frac{\|\mathbf{p}-\mathbf{f}_i\|_2^2}{-2\sigma_2^2}), \quad (6)$$

where σ_2 is set to be the same as the truncated band μ empirically.

The volumetric information from virtual view D_{k+1} is defined

$$\mathbf{I}_\mathbf{p}(\mathbf{p}, \mathbf{r}) = Entropy(\mathbf{p}) \prod_j^{m-1} [1 - P(\mathbf{p}_j|D_{1:k})], \quad (7)$$

where $\{\mathbf{p}_j, j = 0, ..., m-1\}$ are all voxels traversed along a ray $\mathbf{r}$ before hitting the voxel $\mathbf{p}$, and $\prod_j^{m-1} [1 - P(\mathbf{p}_j|D_{1:k})]$ indicates the visibility of $\mathbf{p}$. $Entropy(\mathbf{p})$ is the entropy of $\mathbf{p}$ related to Q as follows:

$$Entropy(\mathbf{p}) = -Q(\mathbf{p}) \ln Q(\mathbf{p}) - [1 - Q(\mathbf{p})] \ln [1 - Q(\mathbf{p})]. \quad (8)$$

Finally, the total view information of the virtual view $\mathbf{v}$ can be formulated as:

$$\mathbf{I}_\mathbf{v} = \sum_l^L \sum_i^I \mathbf{I}_\mathbf{p}(\mathbf{p}_{l,i}, \mathbf{r}_l), \quad (9)$$

where $\mathbf{r}_l$ is all possible casting ray of current view candidate $\mathbf{v}$ and $\mathbf{p}_{l,i}$ is all voxels casted through by a ray $\mathbf{r}_l$ before hitting on surface or volume boundaries. Focusing on the object-centric reconstruction tasks, we model the candidate view search space as a series of cylinder around the maintaining TSDF volume center, parameterized by $\mathbf{v} = (r, \theta, l)$ with all candidate views $\mathbb{V}$ pointing to object center.

For the evaluation of the information gain (IG) of all the candidate viewpoints through ray casting operation, we make use of the modern GPU hardware to achieve real-time implementation. The observation here is that all the candidate viewpoints and all the casted rays are independent to each other. So that different candidate viewpoints can be attached to different blocks in the GPU, while a thread in the block is related to a small batch of rays. In our setting, the casted rays for each view point candidates are measured on $h \times w$ resolution, and we further split the rays into k threads and each thread casts m rays to the volume. After calculating the IG for such each m rays per-thread, the evaluation about all the candidate viewpoints is to perform intra-block sum reduction operation, which can be done efficiently on the GPU using the share memory and the warp reduction operation, as shown in the Algorithm 1. Note that the parameters $\{n, h, w, t, m, warp, lane\}$ above are set to be $\{4608, 64, 64, 1024, 4, 32, 6\}$ respectively in our implementation.

Benefited from the efficient parallel computation, the proposed method brings a considerable lift of speed on NBV calculation. Moreover, it provides a general framework for real-time IG reduction, as long as the computational complexity of information-based function in step. 9 of Algorithm 1 equals to or is less than $O(n)$.

3.2.3 Next Best View Scanning. We select the next-best-view by optimizing the following energy function:

$$\mathbf{v}^\star = \arg\max_\mathbf{v} \lambda_I \mathbf{I}_\mathbf{v} + \lambda_C \mathbf{C}_\mathbf{v} + \lambda_S \mathbf{S}_\mathbf{v}, \quad (10)$$

where $\mathbf{I}_\mathbf{v}$ is the view information calculated by Eqn. 9, $\mathbf{C}_\mathbf{v}$ is the movement cost term which penalties views that need large robot movement from current position, $\mathbf{S}_\mathbf{v}$ is the trajectory smoothness term which encourages the view points lying on current moving direction, and $\lambda_I, \lambda_C, \lambda_S$ are the corresponding coefficients.

The main challenge in human-centric scanning is that the human body may suffer from slightly non-rigid movement during scanning. As studied in [35], the non-rigid deformation of human body and the rigid motion of the aerial robot are coupled together. To turn the selected NBV in virtual view space in Eqn. 9 to the physical world space for aerial robot control, we initialize the camera pose in the volume space, denoted as $\mathbf{T}_{d2v}$, by performing the traditional Rigid-ICP algorithm with the TSDF volume and the live depth image. On the other hand, the camera pose in the world space, denoted as $\mathbf{T}_{d2w}$, can be directly retrieved from the onboard Guidance module of the aerial robot. And then we can simply get

Algorithm 1 Algorithm for IG reduction

Input: TSDF volume, pre-computed view candidates $\mathbf{T}_{view}[n]$
Output: $\mathbf{IG}_{view}[n]$

Initialisation : prepare $m \times t$ ray directions, $\mathbf{Ray}[m \times t]$.
Block-wise LOOP Process in parallel
for $i = 0$ to $n-1$ **do**
  $\mathbf{T}_{curr} = \mathbf{T}_{view}[i]$.
  Thread-wise LOOP Process in parallel
  for $j = 0$ to $t-1$ **do**
    allocate share memory $\mathbf{SEM}[warp]$.
    $\mathbf{IG}_{mray} = 0$.
    LOOP Process of the ray-batch
    for $rayIdx = 0$ to $m-1$ **do**
      $\mathbf{Ray}_{curr} = \mathbf{Ray}[j + t \times rayIdx]$.
      Find $voxelIdx$ by using 3D Bresenham to rasterize $\mathbf{Ray}_{curr}$ and $\mathbf{T}_{curr}$
      calcute **ig** in the $voxelIdx$.
      $\mathbf{IG}_{mray} += \mathbf{I}_{\mathbf{p}}(\mathbf{p}, \mathbf{r})$.
    end for
    $warpid = tid >> (warp - 1)$
    $laneid = tid\&(lane - 1)$
    warp reduction for $\mathbf{IG}_{mray}$
    $reducedValue = \mathbf{IG}_{mray}$
    $\mathbf{SEM}[warpid] = warpReduct(reducedValue)$.
    share memory reduction for $\mathbf{IG}_{view}[i]$
    $reducedValue = \mathbf{SEM}[laneid]$
    $\mathbf{IG}_{view}[i] = warpReduct(reducedValue)$.
  end for
end for

the rigid transformation from the world space to the volume space, denoted as $\mathbf{T}_{w2v}$ as follows:

$$\mathbf{T}_{w2v} = \mathbf{T}_{d2v}(\mathbf{T}_{d2w})^{-1}. \tag{11}$$

To guide the movement of flying camera from current position to next-best-view spot, we utilize a quality-driven method to adaptively insert waypoints before reaching predicted spot. Here we consider different robot orientations on smooth trajectory generated via [14]. The angle formed between the camera ray's orientation and the surface normal is expected to be small, so as to guarantee sensing quality of depth camera. We define the quality of virtual depth image D generated by a yaw angle $\phi \in (-\pi/2, \pi/2)$ as:

$$N(D) = \sum_{l}^{L} < \bar{\mathbf{n}}_l, \bar{\mathbf{r}}_l >, \tag{12}$$

where $\bar{\mathbf{r}}_l$ is the unit vector with the opposite direction with pixel casting ray from camera principle point and $\bar{\mathbf{n}}_l$ is the unit surface normal. Note that we ignore the casting ray and normal pairs that have negative inner product. To obtain the optimal ϕ, we use the same reduction scheme in 3.2.2 and find the maximum quality view in 18 uniformly sampled candidates. The reconstruction ends when the highest information gain of all NBV candidates is smaller than a user-defined threshold.

3.3 Human Body Reconstruction Module

Our human body reconstruction module follows the conventional volumetric fusion pipeline [26], where the TSDF volume aligns the temporal information of the dense 3D data from depth camera and fuses a human body in real-time. On one hand, the TSDF volume is utilized by the active view planning module as described before. On the other hand, we use the Marching Cube algorithm to generate a mesh for visualization from the TSDF volume.

Moreover, with the observation that the human target always has slight motion during the reconstruction process, a light-weight non-rigid deformation method is adopted when integrating the new depth image into the TSDF volume. Similar to recent work [22, 35], we use the embedded deformation (ED) model to parameterize the non-rigid motion field. Given a reference mesh, the sparse ED nodes are uniformly sampled to cover the overall surface. Let $\mathbf{x}_i$ be the i-th ED node location, which is also associated with a set of parameters to represent the deformation around the ED node. Furthermore, neighboring ED nodes are connected together to form a digraph called ED graph, which is collectively represented by all the deformation parameters and ED node locations on it. Since each mesh vertex is "skinned" with its K neighboring ED nodes (with $K = 4$ in our system), the mesh can be deformed according the given parameters of an ED graph.

Aiming at compactly representing the static human model with slight non-rigid deformation, we use the rigid transformation (6-DOF) in the ED node and apply the Linear Blending Skinning (LBS) method for skinning. Thus, the full parameter set for the deformation is $G = \{\mathbf{T}_i\}$. For a particular mesh vertex $\mathbf{v}_j$, its new position is formulated as

$$\mathbf{v}'_j = ED(\mathbf{v}_j; G) = \sum_{\mathbf{x}_i} w(\mathbf{v}_j, \mathbf{x}_i)\mathbf{T}_i\mathbf{v}_j, \tag{13}$$

where $w(\mathbf{v}_j, \mathbf{x}_i)$ measures the influence of the node $\mathbf{x}_i$ to the vertex $\mathbf{v}_j$. Please refer to [35] for details about calculating w for all mesh vertices. Note that Eqn. (13) omits the conversion between the 3-vectors and their corresponding homogeneous 4-vectors (as needed for multiplications with $\mathbf{T}_i$) for simplicity of notation.

The data term is then designed to force vertices on the model to move to the corresponding depth point of the input depth data, especially along the norm direction, which can be considered as the first order approximation of the real surface geometry. As in [35], we find the dense depth correspondences between the model and the depth images via a projective lookup method, and discard those pairs with a highly distinct depth value (larger than 20 mm) or normal direction (larger than 20 degrees):

$$E_{\text{data}}(G) = \sum_{j=1}^{\mathbf{C}} \|\mathbf{n}_{\mathbf{v}_j}^T(ED(\mathbf{v}_j; G) - \mathbf{c}_j)\|_2^2, \tag{14}$$

where $\mathbf{C}$ denotes all correspondent pairs between mesh vertices (denoted as $\mathbf{v}_j$) and depth points (denoted as $\mathbf{c}_j$) in the depth image captured by the aerial robot. Regarding the regular term that prevents unreasonable local deformation of the model, as we utilize 6-DOF rigid transformation instead of 12-DOF affine transformation, it is formulated as

$$E_{\text{reg}}(G) = \sum_{\mathbf{x}_j\mathbf{x}_i} \sum_{\in N(\mathbf{x}_j)} w(\mathbf{x}_j, \mathbf{x}_i)\|\mathbf{T}_i\mathbf{x}_j - \mathbf{T}_j\mathbf{x}_j\|_2^2, \tag{15}$$

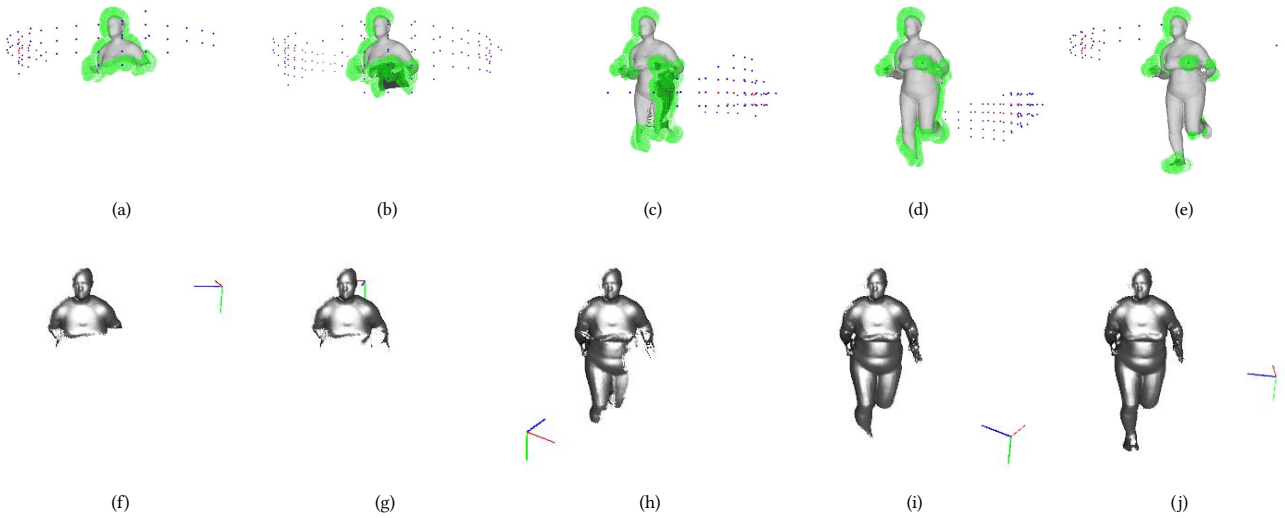


Figure 4: Based on partially reconstructed model (gray mesh), frontier information volume (green dots) is predicted. Top view candidates are colorized according to score. Red represents high scoring, while blue is relative low score view points. (a)-(e) represent $\{2, 3, 4, 6, 9\}th$ NBV iteration respectively. (f)-(g) represent canonical model and camera pose in each NBV iteration.

where $w(\mathbf{x}_j, \mathbf{x}_i)$ defines the weight associated with the edge in the ED node graph. The overall energy is formulated as

$$E_{\text{total}}(G) = \phi_{\text{data}} E_{\text{data}}(G) + \phi_{\text{reg}} E_{\text{reg}}(G) \tag{16}$$

where ϕ_{data} and ϕ_{reg} are coefficients of both terms.

Given the energy terms related to $\{\mathbf{T}_i\}$, we minimize them in an iterative closest point (ICP) framework, where dense pairs are updated by the projective lookup method. In each ICP iteration, the energy above can be rewritten as a sum of squares. In this form, the minimization problem can be seen as a standard sparse non-linear least-squares problem, which can be solved efficiently using the Gauss-Newton method. When performing Gauss-Newton optimization, we adopt the Taylor expansion of the exponential map around current estimated camera poses by introducing small Lie algebra parameters. For compacting the non-rigid deformation to fit the slight motion assumption of our human body reconstruction module and to achieve real-time performance, the number of the ED nodes is restricted so that the ED graph is roughly covered the entire mesh. In our system, the number of all the ED nodes is around 100.

4 EXPERIMENTAL RESULTS

In this section, we first illustrate the computational efficiency of the proposed NBV method, and then experiments on the *iHuman3D* system using both the synthetic data and real-time human scanning data are conducted respectively. We highly recommend readers to refer project page http://www.luvision.net/FlyFusion for more details and comprehensive results.

4.1 Computational Efficiency

For human-centric 3D reconstruction, we maintain a $2m \times 2m \times 2m$ TSDF volume with a 256×256×256 voxel resolution. Empirically, the searching space of $\mathbf{v}$ is restricted by $\{\mathbf{v}|r \in [1, 1.5], \theta \in [-\pi, \pi), l \in [-1, 1]\}$. Jointly considering the depth measurement range and robot maneuverability, 4608 view candidates are uniformly sampled surrounding the volume center to sufficiently represent all possible views.

The efficiency of proposed next-best-view method is assessed. We implemented Algorithm 1 on NVIDIA GeForce GTX1080 using CUDA, and it takes about 30 ms to calculate the IGs from all the n = 4608 candidate viewpoints. As shown in Table 1, compared to 8s for 88 candidate viewpoints using Isler's methods [18], denoted as *Isler*, our method evaluates viewpoints on the order of 1.0×10^5 faster. The speed of evaluating viewpoints in our scheme is comparable with the *APORA*[1] proposed in [10]. Note that the *APORA* still takes 12s to exhaustively evaluate all the candidate viewpoints during a NBV iteration, while the proposed Algorithm 1 enables real-time active view planning. Specifically, for each NBV iteration, our method achieves 200x speed up compared with *APORA* and *Isler*.

Table 1: Computational speed evaluation on candidate viewpoints

	Isler [18]	*APORA* [10]	*iHuman3D*
Average Number of Viewpoints	88	1.5×10^6	4608
Average Time	8 s	12 s	30 ms
Average Views/Second	11	1.3×10^5	1.5×10^5

4.2 Simulation Platform and Evaluation on Synthetic Data

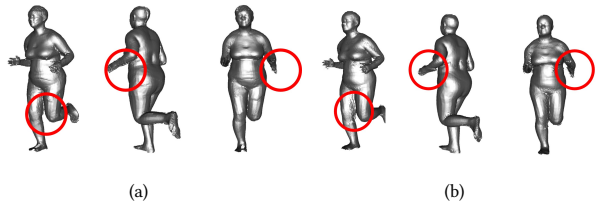
To better assess *iHuman3D* system, we built a simulation platform to evaluate and visualize the next-best-view selection to improve

[1] *APORA*'s 'Average View/Second' number 1.3×10^6 reported in [10] is typo which should be 1.3×10^5 as confirmed by [10]'s author.

the efficiency of our evaluations on the proposed active view planning and human body reconstruction algorithms. Based on our system architecture in Fig.2, we replaced the flying camera module with a simulation platform, which simulates the maneuverability of the physical aerial robot [14] and generates synthetic depth streams. We utilized the recent work SURREAL [30] which embedded Human3.6M [17] pose skeletons into various human SMPL [23] models and the render engine *Blender* to render synthetic depth streams with the ASUS Xtion camera intrinsic parameters.

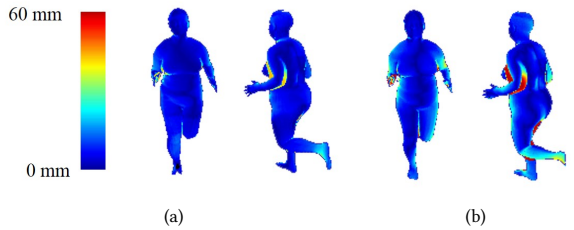
To evaluate the proposed NBV guided human model reconstruction method, in Fig.4, we visualized the reconstructed mesh, camera position, frontier information volume and top view candidates together in the selected NBV iterations. Guided by the frontier information, *iHuman3D* automatically picks out the views which quickly fill the unobserved area, as shown in Fig.4 (c). Note that the robots motion regularization term and trajectory smoothness term ensure the picked view candidates to be close to current camera position and consistent to the direction of the moving robot, as shown in Fig.4 (b, c).

We further compare our method with the most related work *FlyCap* [35] qualitatively and quantitatively. *FlyCap* scans a human body for 3 circles with a fixed spiral-down trajectory, with an off-line human reconstruction algorithm. Focusing on evaluating different view planning method, for fair comparision, we use the same reconstruction module by feeding *FlyCap* a synthetic depth stream with the pre-defined spiral-down trajectory. The qualitative comparison of the reconstruction results is provided in Fig.5, which indicates the proposed method has a better overall quality especially in the tough cases emphasized by the red circles. Fig.6 provides the quantitative per-vertex reconstruction error compared to the ground-truth synthetic model. Our method achieves 14.86 mm average per-vetex error, compared to 20.25 mm of *FlyCap*. These results illustrate the superiority of our NBV guided method in terms of reconstruction.



(a) (b)

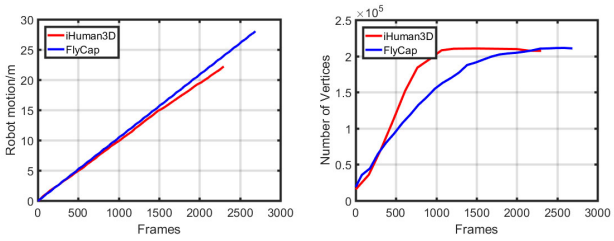
Figure 5: Quality comparison between (a) *iHuman3D* and (b) *FlyCap*.



(a) (b)

Figure 6: Error map compared to groundtruth. (a) *iHuman3D* (b) *FlyCap*

To evaluate the reconstruction efficiency, we compared the online generated trajectory of *iHuman3D* with the one of *FlyCap*. As shown in Fig.7 (a), our method can finish the scanning task more quickly, leading to slower robot motion increase and superior robot efficiency. To evaluate reconstruction efficiency, the vertices convergence property is considered as shown in Fig.7 (b). It indicates that the proposed method can effectively guide human reconstruction. We argue that in conventional scanning methods, it is hard to model the observation overlaps between scan fragments, due to robot localization error and rigid ICP error caused by slight non-rigid movement of the human model. Whereas, in our *iHuman3D* with frontier information guided NBV selection, the robot adaptively moves to spots which can complete the model more efficiently.



(a) (b)

Figure 7: (a) Robot motion according to reconstruction frames. (b) Mesh vertices increase according to frames.

4.3 Real World Experiments

In this subsection, we evaluate the *iHuman3D* system in the practical scenarios as shown in Fig.8. As explained in Sec 3, a flying camera is used to scan a target human, while a desktop machine executes online reconstruction and active view planning. Given a manually selected initial pose, the flying camera will automatically scan the human model until the task is completed. The Flying camera and the desktop machine exchange data via wireless network, while the live output mesh and canonical model are displayed on a screen in real-time.

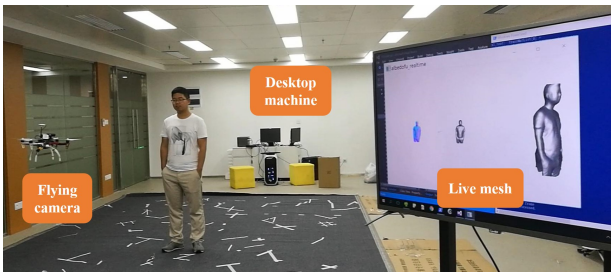


Figure 8: System setup in realtime human reconstruction.

Two reconstructed human models of the practical scenarios is provided in Fig.9 rendered from different views, which obtain considerably high quality results, even with noisy depth input and limited volume resolution for real-time purpose. Note that the experiment is hard to be conducted without any external sensor like Vicon, since the aerial robot cannot receives GPS signals indoor and only the on-board visual odometry module helps for the robot

localization. The initial location observations observed from the flying camera is poor and a naive 3D reconstruction might fail. Thanks to the proposed NBV guided human model reconstruction strategy, our method achieves relatively high accuracy and is robust to the noise of the robot localization module.

(a) (b)

Figure 9: Two different human model reconstructed in real-time experiments, (a) Standing still and (b) Punching.

5 LIMITATIONS AND DISCUSSION

Restricted by current capability of aerial robot, there exists a certain gap before commercial usage of *iHuman3D*, which may further inspires us/communities to solve the related issues in the future work. 1) The flying camera has to maintain in low-speed for the safety issue, and no obstacle avoidance is used in currently system which may limit its application. A more compactly designed drone integrated with different kinds of sensors can definitely assure higher speed for 3D scanning. 2) iHuman3D works for the human body 3D reconstruction with slight motions. As TSDF inherently can work with 3D optical flow to handle larger deformation, we will consider utilizing flow estimation for dynamic human body 3D reconstruction. 3) While current NBV estimation relies on the information gain formulation merely, utilizing human model prior to improve the NBV estimation will be a possible improvement.

6 CONCLUSIONS

We have presented a novel adaptive human body 3D reconstruction system using a single flying camera, which removes the extra manual labor constraint. Besides the autonomous aerial robot, our system adopts a real-time active view planning strategy, based on a novel IG formulation and a highly efficient ray casting algorithm in GPU. For the human reconstruction, a compactly-designed non-rigid deformation method is proposed with the traditional volumetric fusion pipeline. Note that both the active view planning and the human body reconstruction are unified into TSDF volume-based representation. Extensive experiments are conducted to validate our approach. We believe our system steps towards enabling more general human model reconstruction for wider robotic and visual applications.

REFERENCES

[1] Brett Allen, Brian Curless, and Zoran Popović. 2003. The Space of Human Body Shapes: Reconstruction and Parameterization from Range Scans. *ACM Trans. Graph.* 22, 3 (July 2003), 587–594. https://doi.org/10.1145/882262.882311

[2] John Aloimonos, Isaac Weiss, and Amit Bandyopadhyay. 1988. Active vision. *International Journal of Computer Vision* 1, 4 (01 Jan 1988), 333–356. https://doi.org/10.1007/BF00133571

[3] Andreas Baak, Meinard Müller, Gaurav Bharaj, Hans-Peter Seidel, and Christian Theobalt. 2011. A data-driven approach for real-time full body pose reconstruction from a depth camera. In *Proc. of ICCV*.

[4] F. Bissmarck, M. Svensson, and G. Tolt. 2015. Efficient algorithms for Next Best View evaluation. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 5876–5883. https://doi.org/10.1109/IROS.2015.7354212

[5] Shengyong Chen, Youfu Li, and Ngai Ming Kwok. 2011. Active Vision in Robotic Systems: A Survey of Recent Developments. *Int. J. Rob. Res.* 30, 11 (Sept. 2011), 1343–1377. https://doi.org/10.1177/0278364911410755

[6] S. Y. Chen and Y. F. Li. 2005. Vision sensor planning for 3-D model acquisition. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 35, 5 (Oct 2005), 894–904. https://doi.org/10.1109/TSMCB.2005.846907

[7] C. Connolly. 1985. The determination of next best views. In *Proceedings. 1985 IEEE International Conference on Robotics and Automation*, Vol. 2. 432–435. https://doi.org/10.1109/ROBOT.1985.1087372

[8] Yan Cui and Didier Stricker. 2011. 3D Shape Scanning with a Kinect. In *ACM SIGGRAPH 2011 Posters (SIGGRAPH '11)*. ACM, New York, NY, USA, Article 57, 1 pages. https://doi.org/10.1145/2037715.2037780

[9] Brian Curless and Marc Levoy. 1996. A Volumetric Method for Building Complex Models from Range Images. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '96)*. ACM, New York, NY, USA, 303–312. https://doi.org/10.1145/237170.237269

[10] J. Daudelin and M. Campbell. 2017. An Adaptable, Probabilistic, Next-Best View Algorithm for Reconstruction of Unknown 3-D Objects. *IEEE Robotics and Automation Letters* 2, 3 (July 2017), 1540–1547. https://doi.org/10.1109/LRA.2017.2660769

[11] Edilson de Aguiar, Carsten Stoll, Christian Theobalt, Naveed Ahmed, Hans-Peter Seidel, and Sebastian Thrun. 2008. Performance Capture from Sparse Multi-view Video. *ACM Trans. Graph.* 27, 3, Article 98 (Aug. 2008), 10 pages. https://doi.org/10.1145/1360612.1360697

[12] Jeffrey Delmerico, Stefan Isler, Reza Sabzevari, and Davide Scaramuzza. 2017. A comparison of volumetric information gain metrics for active 3D object reconstruction. *Autonomous Robots* (22 Apr 2017). https://doi.org/10.1007/s10514-017-9634-0

[13] K. Desai, K. Bahirat, and B. Prabhakaran. 2017. Learning-based objective evaluation of 3D human open meshes. In *2017 IEEE International Conference on Multimedia and Expo (ICME)*. 733–738. https://doi.org/10.1109/ICME.2017.8019470

[14] Fei Gao and Shaojie Shen. 2016. Online quadrotor trajectory generation and autonomous navigation on point clouds. In *Safety, Security, and Rescue Robotics (SSRR), 2016 IEEE International Symposium on*. IEEE, 139–146.

[15] Nils Hasler, Bodo Rosenhahn, Thorsten Thormahlen, Michael Wand, Juergen Gall, and Hans-Peter Seidel. 2009. Markerless Motion Capture with Unsynchronized Moving Cameras. In *Proc. of CVPR*.

[16] Armin Hornung, Kai M Wurm, Maren Bennewitz, Cyrill Stachniss, and Wolfram Burgard. 2013. OctoMap: An efficient probabilistic 3D mapping framework based on octrees. *Autonomous Robots* 34, 3 (2013), 189–206.

[17] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. 2014. Human3.6M: Large Scale Datasets and Predictive Methods for 3D Human Sensing in Natural Environments. *IEEE Trans. Pattern Anal. Mach. Intell.* 36, 7 (jul 2014), 1325–1339.

[18] S. Isler, R. Sabzevari, J. Delmerico, and D. Scaramuzza. 2016. An information gain formulation for active volumetric 3D reconstruction. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*. 3477–3484. https://doi.org/10.1109/ICRA.2016.7487527

[19] C. Kerl, J. Sturm, and D. Cremers. 2013. Dense Visual SLAM for RGB-D Cameras. In *Proc. of the Int. Conf. on Intelligent Robot Systems (IROS)*.

[20] I. Khan. 2018. Robust Sparse and Dense Nonrigid Structure From Motion. *IEEE Transactions on Multimedia* 20, 4 (April 2018), 841–850. https://doi.org/10.1109/TMM.2017.2758740

[21] Simon Kriegel, Christian Rink, Tim Bodenmüller, and Michael Suppa. 2015. Efficient next-best-scan planning for autonomous 3D surface reconstruction of unknown objects. *Journal of Real-Time Image Processing* 10, 4 (01 Dec 2015), 611–631. https://doi.org/10.1007/s11554-013-0386-6

[22] Hao Li, Bart Adams, Leonidas J Guibas, and Mark Pauly. 2009. Robust single-view geometry and motion reconstruction. *ACM Trans. Graph. (Proc. of SIGGRAPH Asia)* 28, 5 (2009), 175.

[23] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. 2015. SMPL: A skinned multi-person linear model. *ACM Trans. Graph. (Proc. of SIGGRAPH)* 34, 6 (2015), 248.

[24] Riccardo Monica and Jacopo Aleotti. 2018. Contour-based next-best view planning from point cloud segmentation of unknown objects. *Autonomous Robots* 42, 2 (01 Feb 2018), 443–458. https://doi.org/10.1007/s10514-017-9618-0

[25] Hans Moravec and Alberto Elfes. 1985. High resolution maps from wide angle sonar. In *Robotics and Automation. Proceedings. 1985 IEEE International Conference on*, Vol. 2. IEEE, 116–121.

[26] Richard A. Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J. Davison, Pushmeet Kohli, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. 2011. KinectFusion: Real-Time Dense Surface Mapping and Tracking. In *Proc. of ISMAR*. 127–136.

[27] Richard Pito. 1999. A Solution to the Next Best View Problem for Automated Surface Acquisition. *IEEE Trans. Pattern Anal. Mach. Intell.* 21, 10 (Oct. 1999), 1016–1030. https://doi.org/10.1109/34.799908

[28] William R. Scott, Gerhard Roth, and Jean-François Rivest. 2003. View Planning for Automated Three-dimensional Object Reconstruction and Inspection. *ACM Comput. Surv.* 35, 1 (March 2003), 64–96. https://doi.org/10.1145/641865.641868

[29] J. Tong, J. Zhou, L. Liu, Z. Pan, and H. Yan. 2012. Scanning 3D Full Human Bodies Using Kinects. *IEEE Transactions on Visualization and Computer Graphics* 18, 4 (April 2012), 643–650. https://doi.org/10.1109/TVCG.2012.56

[30] Gül Varol, Javier Romero, Xavier Martin, Naureen Mahmood, Michael J. Black, Ivan Laptev, and Cordelia Schmid. 2017. Learning from Synthetic Humans. In *Proc. of CVPR*.

[31] J. Irving Vasquez-Gomez, L. Enrique Sucar, Rafael Murrieta-Cid, and Efrain Lopez-Damian. 2014. Volumetric Next-best-view Planning for 3D Object Reconstruction with Positioning Error. *International Journal of Advanced Robotic Systems* 11, 10 (2014), 159. https://doi.org/10.5772/58759 arXiv:https://doi.org/10.5772/58759

[32] K. Wang, G. Zhang, and S. Xia. 2017. Templateless Non-Rigid Reconstruction and Motion Tracking With a Single RGB-D Camera. *IEEE Transactions on Image Processing* 26, 12 (Dec 2017), 5966–5979. https://doi.org/10.1109/TIP.2017.2740624

[33] Xiaolin Wei, Peizhao Zhang, and Jinxiang Chai. 2012. Accurate realtime full-body motion capture using a single depth camera. *ACM Trans. Graph. (Proc. of SIGGRAPH Asia)* 31, 6 (2012), 188.

[34] T. Whelan, R. F. Salas-Moreno, B. Glocker, A. J. Davison, and S. Leutenegger. 2016. ElasticFusion: Real-Time Dense SLAM and Light Source Estimation. *Intl. J. of Robotics Research, IJRR* (2016).

[35] L. Xu, Y. Liu, W. Cheng, K. Guo, G. Zhou, Q. Dai, and L. Fang. 2017. FlyCap: Markerless Motion Capture Using Multiple Autonomous Flying Cameras. *IEEE Transactions on Visualization and Computer Graphics* PP, 99 (2017), 1–1. https://doi.org/10.1109/TVCG.2017.2728660

[36] B. Yamauchi. 1997. A frontier-based approach for autonomous exploration. In *Computational Intelligence in Robotics and Automation, 1997. CIRA'97., Proceedings., 1997 IEEE International Symposium on*. 146–151. https://doi.org/10.1109/CIRA.1997.613851

[37] Genzhi Ye, Yebin Liu, Nils Hasler, Xiangyang Ji, Qionghai Dai, and Christian Theobalt. 2012. Performance Capture of Interacting Characters with Handheld Kinects. In *Proc. of ECCV*.

[38] Guyue Zhou, Lu Fang, Ketan Tang, Honghui Zhang, Kai Wang, and Kang Yang. 2015. Guidance: A Visual Sensing Platform For Robotic Applications. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.