IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received April 6, 2020, accepted April 25, 2020, date of publication April 29, 2020, date of current version May 14, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2991346

Perspective Independent Ground Plane Estimation by 2D and 3D Data Analysis

CHENGSI ZHANG[1] AND STEPHEN CZARNUCH[2], (Member, IEEE)

[1]Department of Electrical and Computer Engineering, Faculty of Engineering and Applied Science, Memorial University of Newfoundland, St. John's, NL A1C 5S7, Canada

[2]Department of Electrical and Computer Engineering, Faculty of Engineering and Applied Science and the Discipline of Emergency Medicine, Faculty of Medicine, Memorial University of Newfoundland, St. John's, NL A1C 5S7, Canada

Corresponding author: Chengsi Zhang (cz2075@mun.ca)

This work was supported by the Natural Sciences and Engineering Research Council of Canada under Grant RGPIN-2016-04165.

ABSTRACT Identifying the orientation and location of a camera placed arbitrarily in a room is a challenging problem. Existing approaches impose common assumptions (e.g. the ground plane is the largest plane in the scene, the camera roll angle is zero). We present a method for estimating the ground plane and camera orientation in an unknown indoor environment given RGB-D data (colour and depth) from a camera with arbitrary orientation and location assuming that at least one person can be seem smoothly moving within the camera field of view with their body perpendicular to the ground plane. From a set of RGB-D data trials captured using a Kinect sensor, we develop an approach to identify potential ground planes, cluster objects in the scenes and find 2D Scale-Invariant Feature Transform (SIFT) keypoints for those objects, and then build a motion sequence for each object by evaluating the intersection of each object's histogram in three dimensions across frames. After finding the reliable homography for all objects, we identify the moving human object by checking the change in the histogram intersection, object dimensions and the trajectory vector of the homgraphy decomposition. We then estimate the ground plane from the potential planes using the normal vector of the homography decomposition, the trajectory vector, and the spatial relationship of the planes to the other objects in the scene. Our results show that the ground plane can be successfully detected, if visible, regardless of camera orientation, ground plane size, and movement speed of the human. We evaluated our approach on our own data and on three public datasets, robustly estimating the ground plane in all indoor scenarios. Our successful approach substantially reduces restrictions on a prior knowledge of the ground plane, and has broad application in conditions where environments are dynamic and cluttered, as well as fields such as automated robotics, localization and mapping.

INDEX TERMS Image motion analysis, image segmentation, sensor orientation detection, ground plane detection.

I. INTRODUCTION

With one additional dimension, 3D data provide a more intuitive and realistic environmental perspective in computer vision applications than traditional 2D data. By combining traditional 2D RGB data with depth information, 3D data create a more comprehensive digital representation of real world environments, providing considerable value in many applications such as training and simulation [1]–[3], construction [4]–[6] and gaming [7]–[10]. The benefits of 3D data over 2D data are particularly noticeable in cluttered or dynamic environments. In these complex environments, 3D data allow enhanced visual understandings, improved precision and accuracy, easier risk/issue identification and analysis, and intuitive model manipulation [11]–[15]. For example, operating rooms typically have many objects that frequently change depending on the nature of the emergency, including multiple humans who enter and exit the room and interact with the objects and each other. Constructing an accurate 3D model of an operating room and recording videos of various processes within the room could create a helpful and interactive tool for training and simulation, or be used in real time to observe and monitor the room. For applications like gaming, the room is often modified to accommodate placement of a sensor (i.e., clearing out a space), the sensor is intentionally located in an ideal position, and users are

The associate editor coordinating the review of this manuscript and approving it for publication was Guitao Cao.

willing to undergo a calibration process if necessary. However, the applications we consider, such as the operating room, are complex, dynamic and cluttered real-world environments, where the sensor must be located out of the way of the processes or occupants of the room, and systems using the sensor would need to auto-calibrate because occupants of the room are unlikely to be willing to perform calibrations. Accordingly, in applications in these complex environments, the sensor's location and orientation in the room will generally be unknown (e.g., the sensor's field of view cannot be assumed to be parallel to the ground). In this paper, we focus on addressing the difficulties of estimating the ground plane and finding the camera orientation in a indoor environment without any prior knowledge of the sensor or room.

In order to process image information from a unknown environment, knowledge of the ground plane, and hence the position and orientation of the camera, is fundamental [16]–[20]. Indeed, most computer vision algorithms implicitly assume knowledge of the ground plane (e.g., that the ground is at the ''bottom'' of the scene [17], [21], [22] or is the largest plane [23], [25], [26]). However, in complex environments with unknown sensor placement, the ground plane may not be the largest visible plane (e.g., many objects on the ground) or at the ''bottom'' of the scene (e.g., overhead perspectives). Still, identifying the ground plane, and accordingly the camera position and orientation, is critical for most computer vision applications; especially for indoor tracking, exploring, navigation and scene analysis. For instance, in Simultaneous Localization and Mapping (SLAM) applications, RGB-D data have been used to extract the plane feature in indoor environments for localizing robot positions, outperforming both accuracy and efficiency of the traditional point feature-based methods, even with low image quality devices [55], [56]. With the recognition of the ground plane and camera orientation, the robot performs better SLAM occlusion detection during mapping [57] and obstacle detection [58]. In addition, finding the ground plane and calculating the camera orientation also facilitates improved 3D registration and 3D reconstruction of data from multiple sensors viewing the same scene by converting a 3D problem into a 2D problem. Ultimately our goal is estimating the ground plane for each sensor in a multi-sensor system, such that the ground can be used as a reference for finding the positions and orientations of each sensor relative to each other, which will facilitate the reliable 3D reconstruction of a complex room.

To accomplish our goal, we aim to develop a system that estimates the ground plane, camera orientations and relative locations of multiple RGB-D sensors with unknown positions and orientations in an indoor environment. Our only assumptions are: that most of at least one person can be seen smoothly moving in the RGB-D camera field of view; the person's body is perpendicular to the ground plane while moving; and the RGB-D camera's position and orientation remain unchanged until the ground plane estimation is complete. In order to estimate the ground plane under this condition, we combine the robustness of 3D Random Sample Consensus (RANSAC) and 2D homography decomposition. While 3D RANSAC extracts useful spatial information from each 3D point cloud segment, 2D homography decomposition constructs homography planes from people walking on the ground. Our approach even accommodates scenarios where the ground plane is a small region (i.e., barely visible) or even not visible in the field of view (FOV) of the sensor by utilizing other visible planes that are parallel to the trajectory of movement and estimating the actual ground plane.

II. RELATED WORK

Existing ground plane detection can be broadly categorized into 2D or 3D approaches based on the sensor type. Within 2D approaches, the most popular approach for ground plane estimation is homography. For example, homography-based approaches have been used to first find the feature key points in the scene, followed by Kalman filtering [27] or Modified Expectation Maximization [28] to build confidence in the ground plane transformation matrix across successive frames. These two approaches assumed the roll angle of sensors are zero and the camera only see the ground plane with objects above the plane. Homography has also been successfully used as a first step, with the homography decomposition results combined with a Bayes filter [29] or contour searching [30] to estimate the ground plane with 2D images. However, again the ground plane is assumed to be the area in front of the camera [29], or the single colour ground plane is assumed to occupy the majority of the FOV [30]. Other 2D approaches have used depth-image data or V-disparity values (the histogram of the disparity map [31]) rather than traditional RGB image data [23], [24]. Zhi Jin *et al.* [32] proposed a depth-map driven ground plane detection algorithm by growing a plane starting from the the largest area having similar depth values in the depth map, assuming the largest plane was the ground plane. Kircali and Tek [33] estimated the ground plane based on comparing the depth map of each new frame with a pre-calibrated depth map in which the ground plane was pre-defined. Assuming the majority area in the scene comprises the ground plane, the gradient of the V-disparity pixel values has also been successfully used to identify the ground plane with an arbitrary camera roll angle [23]. Furthermore, Cherian *et al.* [35] applied multiple texture based filters with a Markov Random Field to reconstruct the depth map from a single RGB image and estimate the ground plane based on texture-based searching segmentation. Due to the intrinsic features of the algorithm, this approach assumes the camera is parallel to the ground plane, and that the ground plane has a unique texture. Dragon *et al.* [34], [36] proposed an approach where RGB frames captured from a moving sensor are iteratively split into regions until reliable homographies can be estimated from the feature points within these regions. The decomposition of the homography with the highest probability indicates the orientation and ego motion of the sensor's movement. Unfortunately, this approach is not suitable for indoor environments with a stationary sensor because moving objects will be a small proportion of the scene, making it hard to distinguish between a homography generated from mismatched key points and a homography from a moving object. Further, their solution requires the shape of moving objects to remain unchanged to ensure successful feature correspondence between frames; a condition that cannot be guaranteed in indoor environments with an arbitrary fixed perspective. More recently, a ground plane estimation approach using monocular images with a predefined region of interest [38] was developed, but requires a known pitch angle. Although the above 2D approaches can successfully identify the ground plane, none of them work in dynamic or cluttered environments where the location and orientation of the sensor is unknown.

Ground plane estimation approaches in 3D commonly utilize 3D Hough transform or 3D RANSAC with the raw data. For example, a 3D Hough transform with a ball-based accumulator, which collects the vote values [37], has been used to define the ground plane based on the highest vote among accumulators [41]. Due to the voting procedure, this approach can only find the ground plane if it is the largest plane in the scene. 3D RANSAC, a more direct and brute-force approach, has been used on raw 3D data to find the ground plane with the assumption that the ground plane is the closest or largest plane in the camera FOV [21]. Other 3D approaches have used an estimation of the 3D normal vector for each raw data point rather than the raw points directly (e.g., [42] and [43]), but assume that the camera roll and pitch angles are zero. More recently, machine learning and a depth mask has been used, but requires minimal orientation variations (i.e., $0 \sim 15°$) [39]. Ground plane estimation has also been integrated into bigger applications (e.g., [21], [40], [57]), but they also share the common constraints, such as zero roll rotation or the ground plane being the largest plane. Similar to promising 2D ground plane approaches, these 3D approaches will also not work in cluttered or dynamic environments because of their underlying assumptions.

Together, the most robust and reliable 2D and 3D methods of finding the ground plane have common assumptions or predicates, such as the known and unchanged orientation of the camera, the ground plane being the largest plane in the field of view, the shape of moving objects in the scene remaining unchanged, the ground plane having a single color or depth value, or the ground plane only appearing at a certain location within the camera's FOV. While these assumptions restrict the complication of the ground plane estimation problem based on the requirements of specific applications, they cannot be used in real-world scenarios where the camera location and orientation are unknown, and the environment is complex, cluttered or dynamic. To overcome the limitations of these assumptions for our application, we build on the approach of Dragon *et al.* [34], [36] because the assumptions of their approach are closest to our conditions. Notably, while their approach requires the sensor to be moving, we assume that the sensor is stationary and something in the scene is instead moving. In our case, we will restrict our interest to a human moving in the scene, though this does not necessarily need to be the case. We present our approach to accomplish this in section III followed by our experimental setup and results in section IV. We then present our discussion and future work in section V.

III. METHODOLOGY

Our ground plane estimation approach combines the robustness of 2D and 3D computer vision algorithms. The major components of our approach are: 1) Data pre-processing (section III-A) where we described the preparation of 2D and 3D data with corresponding features; 2) 2D homography decomposition (section III-B), where we decomposed the 2D homography according to 3D feature restrictions to estimate the trajectory of any moving humanoid objects in the scene; and 3) 3D ground plane estimation (section III-C) where we derived the most probable ground plane by refining 2D homography decomposition results into confidence estimates.

A. DATA PRE-PROCESSING

To obtain a more useful 3D data representation, we first generated a 3D point cloud from the RGB-D data using the intrinsic and extrinsic parameters of the sensor. We calibrated using Zhang's approach with the intrinsic parameter matrix defined as: [44]: $K_c = \begin{bmatrix} fm_x & \gamma & u_0 \\ 0 & fm_y & v_0 \\ 0 & 0 & 1 \end{bmatrix}$, where f is the focal length, m_x and m_y are the scale factor in the image x- and y-axes, γ is the skew coefficient between the x and y axes, and (u_0, v_0) is the principal point. The extrinsic parameter matrix is $[\begin{smallmatrix} R_{3\times3} & T_{3\times1} \\ 0_{1\times3} & 1 \end{smallmatrix}]$, composed of rotation and translation parameters R and T. Finally, using radial distortion k_1, k_2, k_3 and tangential distortion p_1, p_2 coefficients, we calculated the camera matrix C by multiplying the intrinsic and extrinsic matrices, such that the depth images were undistorted based on camera parameters and distortion coefficients [45] according to

$$\begin{aligned} x' &= p_2(3x^2 + y^2) + x(k_2(x^2 + y^2) \\ &\quad + k_1(x^2 + y^2) + 1) + 2P_1xy \qquad (1) \\ y' &= p_1(x^2 + 3y^2) + y(k_2(x^2 + y^2)^2 \\ &\quad + k_1(x^2 + y^2) + 1) + 2p_2xy \qquad (2) \\ z' &= z \qquad (3) \end{aligned}$$

From Eqs.(1), (2) and (3), the coordinates (x, y) and value of each pixel z in each depth image was transformed to an individual point (x', y', z') in the associated 3D point cloud.

In general, the point cloud of an indoor environment is composed of planes (e.g., walls, floor), objects (e.g, drawers, chairs), and humans, though in some cases substantial portions of objects are also planes (e.g., desks). In a cluttered environment with unknown camera location and orientation, the ground plane may not be visible (e.g., if the sensor is on the ground facing up), or may be any region varying from a small region that is highly occluded by objects to the largest visible plane. Therefore, after down-sampling the point cloud by applying a voxel grid filter, we segmented the point cloud into planes and non-planar objects. First, we iteratively extracted, stored and removed the largest plane from the remaining point cloud, which is generated from the previous iteration, until the number of remaining points is less than 20% of the total points in the original point cloud, using Random Sample Consensus(RANSAC) [46] (See Algorithm 1).

Algorithm 1 Plane Extraction

```
procedure extract_PC_Planes($pointCloud$)
    $planes \leftarrow []$
    $pc \leftarrow pointCloud$
    $originSize \leftarrow$ Size of $pc$
    while Size of $pc > 20\% originSize$ do
        $plane \leftarrow$ RANSAC($pc$)
        if Size of $plane < Threshold$ then
            break;
        $planes \leftarrow ps$
        $pc \leftarrow pc - plane$
    $returnplanes, pc$
```

After we stored and removed the planes in the scene, we segmented the remaining point cloud into non-planar objects using Euclidean clustering [47]. We first employed Euclidean clustering to find groups of points that were physically close to each other, and then we stored all clustered objects S_o and extracted planes S_p.

To identify which clustered objects are moving in the scene in preparation for homography estimation, we needed to find corresponding objects between successive frames. We utilized SIFT [48] as the feature extractor on the RGB images to derive 2D feature points. SIFT was able to generate a sufficient number of 2D features for each object in the scenes; particularly for any humans. Additionally, SIFT accommodates a wide range of performance control through variation of the octave layer number $nOct$, edge-like feature filter threshold $eThresh$, and the sigma of Gaussian filter σ [49], allowing excellent optimization for keypoint detection. For each RGB frame, the 2D feature points were stored as an output of the data preparation phase, along with the 3D points of the clustered objects and the extracted planes.

B. HOMOGRAPHY ESTIMATION

A homography matrix [50] can be computed by matching features in two RGB images of an object captured by two cameras at different locations [27]. Since we assume the camera is static and humans move on the ground plane, we calculate the homography matrix using SIFT keypoints in two RGB frames, which are captured at time t and $t + \Delta t$, from a single sensor, using the moving humans as motion reference points. We used the homography between moving objects across successive frames to construct a plane that is perpendicular to the ground plane. With a minimal sample set of four feature key point correspondences between frames at time t and time $t + \Delta t$, a nine-parameter homography matrix $H = \begin{bmatrix} h_1 1 & h_1 2 & h_1 3 \\ h_2 1 & h_2 2 & h_2 3 \\ h_3 1 & h_3 2 & h_3 3 \end{bmatrix}$ can be generated, which represents the transformation between 2D points in image coordinates and 3D points in the camera coordinate system.

To find which objects were moving between successive frames, we implemented the Blockwise Linearity Assumption (see [34]). Instead of generating a result from each pair of consecutive frames, the Blockwise Linearity Assumption estimates an average result from an N-length block of frames by processing the first frame of the block, which is used as reference frame, and the i^{th} frame in the block (where $1 < i \leq N$). Assuming the human moves reasonably smoothly over the ground plane, the changes between the $1^{st} \sim i^{th}$ frame pair and the $1^{st} \sim (i+1)^{th}$ frame pair within one block will grow linearly. We segmented the entire data set into blocks $B = \{F_1, F_2, \ldots F_x\}$ of frames F ranging from frame 1 to x. Let S_o^1 and S_o^2 denote all the object segments in the first and second point clouds representing a pair of successive frames. We calculated the 1-D histogram of three dimensions $Hist_x, Hist_y, Hist_z$ for each object segment $S_{o_i}^1$ and $S_{o_j}^2$. Then, we matched a pair of object segments in F_1 and F_x that represented the same object O_i by determining if the intersection ratio, which is the Jaccard index [54] of the pair of object segments

$$intersection_{o_i}^x = \frac{A(S_{o_i}^1) \cap A(S_{o_j}^x)}{A(S_{o_i}^1)} \tag{4}$$

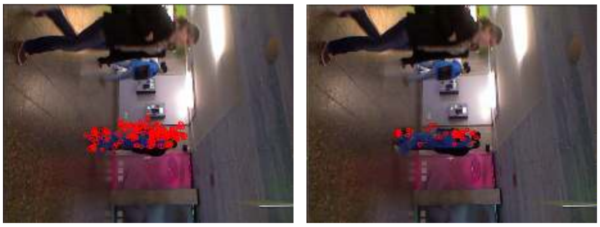
between the histogram areas of $S_{o_i}^1$ and $S_{o_j}^2$ was greater than zero, and decreased as x increased. To ensure the histogram intersection was larger than zero between the first frame F_1 and frame F_x, we chose a small block size similar to [34], [36]. The resulting list of matched pairs of 3D objects $S_{o_i}^1$ and $S_{o_j}^2$, including any moving humans, were projected to 2D pixel clusters $C_{o_i}^1$ and $C_{o_j}^2$ according to

$$x = x'(1 + k_1 r^2 + k_2 r^4) + 2p_1 x' y' \\ +p_2(r^2 + 2'^2) \tag{5}$$

$$y = y'(1 + k_1 r^2 + k_2 r^4) + 2p_2 x' y' \\ +p_1(r^2 + 2y'^2), \tag{6}$$

where (x', y') denotes the x and y values of a 3D point (x', y', z'), (x, y) denotes the corresponding distorted pixel coordinates and $r = \sqrt{x'^2 + y'^2}$. Consequently, each 2D pixel cluster $C_{o_i}^x$ is then converted to a 2D feature point cluster $R_{o_i}^x$ by using each 2D pixel (x_i, y_i) as the center point and searching for the closest feature points within the radius τ, shown in Figure 1(a).

We removed any feature keypoints that were outside of the regions, and applied Motion-Split-And-Merge (MSAM) [36] to each pair of corresponding regions $R_{o_i}^1$ and $R_{o_j}^x$ in F_1 and F_x respectively to find the most reliable keypoint clusters $C_{k_i}^x$ for generating homography matrices $H_{o_i}^x$ (Figure 1(b)).



(a) 2D feature cluster $R^{x}_{o_i}$ (b) MSAM result $C^{x}_{k_i}$

FIGURE 1. 2D human feature keypoint cluster example.

The homography matrix, which is directly generated from human feature points $R^{x}_{o_i}$, can be unreliable because of the different movement patterns of human heads, chest, arms and legs. MSAM accounts for these differing movement patterns by finding the most reliable keypoints (most likely keypoints that are within head or body region) out of the set $R^{x}_{o_i}$, allowing a reliable homography matrix to be generated that represents the human's stable movement through a block B(e.g, [60], [61]). Similarly, the MSAM result $H^{x}_{o_i}$ indicates the movement of a $R^{x}_{o_i}$ cluster without any prior knowledge or assumption. We then decomposed each homography $H^{x}_{o_i}$ into the four plane normal vector, trajectory vector, and rotation vector solutions $D^{x}_{o_i 1\sim4} = \{\vec{n}^{i}_{o_i 1\sim4}, \vec{t}^{i}_{o_i 1\sim4}, \vec{r}^{i}_{o_i 1\sim4}\}$ [51], and filtered out the invalid solutions to construct the most reliable decomposition solution $B_{R_o} = \{\vec{n}_{o_i}, \vec{t}_{o_i}, \vec{r}_{o_i}\}$ for each 2D object region R_{o_i} within a block. Here, invalid homography solutions were characterized by checking if a 2D key point (x_i, y_i) and a 3D point cloud point (x'_i, y'_i, z'_i) within region R_{o_j}, which yields $z'_i < 0$ $\left((x'_i, y'_i, z'_i) = H(x_i, y_i, 1) \;\; and \;\; \vec{n}^{T}_{o_i}(x'_i, y'_i, z'_i) = 1\right)$, exists [34]. Finally, we built the set of all the moving objects in the scene S_{mo_i} by extracting the object regions that had large and successively decreasing differences in intersection coefficient $intersection^{x}_{o_i}$ among all objects O in a block. Based on the decomposition result and the assumption that the person body is perpendicular to the ground plane while moving, we use three conditions, which includes the longest edge E_l of moving object bounding boxes larger than a length threshold $Thresh_l$; the ratios between the longest edge E_l and other two edges are larger than a ratio threshold $Thresh_r$; and trajectory vector $\vec{t}^{i}_{o_i}$ is perpendicular to the longest edge of object bounding box E_l, to determine the moving humanoid object among all moving objects [62]. The homography decomposition result of the moving objects in a block were the output of this phase, allowing us to estimate the ground plane out of the candidate planes extracted in section III-A.

C. GROUND PLANE ESTIMATION

According to the assumption that a person is moving on the ground, the ground plane is then the plane that best satisfies the following criteria:

- *c1:* its normal is parallel to the plane that is defined by the block homography decomposition's normal vector and trajectory vector for the moving object;
- *c2:* it is parallel to the trajectory vector of any moving object;
- *c3:* it does not dissect any objects in the scene;
- *c4:* it is close to the object segments S_o in the scene, and in particular to moving objects.

Based on these criteria we built a confidence estimate cascaded filter to score the likeliness that an extracted plane is the ground plane, ranging from 0 (very unlikely) to 10 (very likely), from the complex and noisy 3D environment. Conceptually, we found all horizontal planes (those parallel to the homography's normal and trajectory vector) from all known planes. We then increased or decreased our confidence in horizontal planes based on their proximity to the boundary of the 3D scene. Finally, we adjusted our confidence estimates based on each plane's relationship to objects in the scene, prioritizing their spatial relationship to moving objects. To distinguish between low-confidence valid planes and invalid planes, whose confidence estimates are reduced by our cascaded filter, we assigned an small initial confidence $conf_I = 1$ to each of the extracted planes S_p that were found in section III-A. We then evaluated the fit of each plane to our criteria to complete our confidence estimates. The overall confidence of each potential ground plane is found as:

$$conf_{S_{pp}} = conf_I + conf_{HD} + conf_{RP} + conf_{OD} \tag{7}$$

where $conf_{HD}, conf_{RP}, conf_{OD}$ represent Homography Decomposition Checking confidence, Relative Position Checking confidence, and Object Distance Checking confidence respectively.

1) HOMOGRAPHY DECOMPOSITION CHECKING FILTER

We scored each ground plane according to criteria *c1* and *c2*: how parallel each potential ground plane is to both the trajectory and the block homography decomposition of each human moving object. To identify the moving objects that were likely humans, we employed a heuristic. Since the camera orientation was arbitrary, we used the normal vector $\vec{n}_{pp_i}$ of each S_{p_i} as the camera's reference orientation. The complementary angle of the angle between $\vec{n}_{pp_i}$ and the x-axis $\theta_{\vec{n}x}$ indicates the roll angle of the camera, while the complementary angle of the angle between $\vec{n}_{pp_i}$ and the z-axis $\theta_{\vec{n}z}$ indicates the pitch angle of the camera. Hence, the roll rotation matrix and pitch rotation matrix were generated by: $R_{roll} = \begin{bmatrix} \cos(C_{\theta_{\vec{n}x}}) & -\sin(C_{\theta_{\vec{n}x}}) & 0 \\ \sin(C_{\theta_{\vec{n}x}}) & \cos(C_{\theta_{\vec{n}x}}) & 0 \\ 0 & 0 & 1 \end{bmatrix}$ and $R_{pitch} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(-C_{\theta_{\vec{n}z}}) & -\sin(-C_{\theta_{\vec{n}z}}) \\ 0 & \sin(-C_{\theta_{\vec{n}z}}) & \cos(-C_{\theta_{\vec{n}z}}) \end{bmatrix}$ based on right hand rule, where $C_{\theta_{\vec{n}x}}$ and $C_{\theta_{\vec{n}z}}$ represent the complementary angles of the roll and pitch angles respectively. After we transformed each moving object S_{mo_i} with its corresponding rotation matrices R_{roll} and R_{pitch} to ensure the bounding box of S_{mo_i} aligned with the x-, y- and z-axis, we determined whether the moving object was humanoid based on three conditions:

1) The longest dimension was at least 1.5 times larger than the other two dimensions [59];
2) The longest edge of the bounding box was longer than a learned threshold; and
3) The trajectory vector $\vec{t}_{mo_i}$ was perpendicular to the longest edge of the bounding box.

We first represented each human moving object S_{mo_i} as the 3D plane $P_{o_i}^{homo}$, constructed from the normal vector $\vec{n}_{mo_i}$ and the trajectory vector $\vec{t}_{mo_i}$. The contribution of the Homography Decomposition Checking confidence to the overall confidence estimate cascaded filter was:

$$conf_{HD} = 2\cos\theta \tag{8}$$

where θ is the angle between the normal $P_{o_i}^{homo}$ and the normal of S_{p_i}. The cosine of the angle is used to ensure that a small penalty is applied to planes that are nearly parallel (likely due to sensor noise), but a large penalty to planes that are not parallel. The constant scaling factor of two is the associated weight of this component relative to the other components of the confidence estimate cascaded filter. Since the confidence only represents the likelihood that a plane is horizontal, the associated weight factor is comparably small, while ensuring that the confidence score of planes that have a large θ angle are reduce to zero. Additionally, for each moving object, we generated a set of planes that were parallel to the movement of $P_{o_i}^{homo}$ as S_{pp_i}.

2) RELATIVE POSITION CHECKING FILTER

We scored each plane in S_{pp_i} according to criteria *c3*: how likely it is that potential ground planes do not dissect objects in the scene. In most cases, the ground plane will not have objects on both sides of it while other planes (e.g., tabletops) can have objects on both sides. In the exceptional scenario, where the floor contains planes with multiple height values (e.g. stairs or theater stages) and the person walks on the plane that has the higher height value, our confidence estimate directly relates to the size of each plane and the difference between the sensor and the two planes. We will discuss this rare scenario in the Section V. Furthermore, this filter was essential for remediating the effects of noise and sensor depth error in the data. We represented each plane S_{pp_i} by it's plane equation:

$$\rho = ax + by + cz + d \tag{9}$$

The value of ρ will be positive, zero, or negative, indicating which side of the plane the point is on, or whether the point is on the plane. We applied the 3D coordinates (x', y', z') of each point in each S_{pp_i} to Eq.(9), recording the number of positive ρ_+ and negative ρ_- results, the maximum distance $d_{max_i^+}$ from the points above the plane to plane S_{pp_i}, and the maximum distance $d_{max_i^-}$ from the points below the plane to plane S_{pp_i}. The contribution of the Relative Position Checking score to the overall confidence estimate function was represented by:

$$conf_{RP} = 2\cos\left(\frac{\rho_+}{\rho_+ + \rho_-}\right) \tag{10}$$

Again, the cosine of the proportion of points on one side of the plane was used to apply a smaller penalty from objects that are on one side of the plane and a larger penalty from objects that are on both sides of the plane. Additionally, similar to the Homography Decomposition Checking Filter factor, the constant scale factor of two again is the relative weight of this component to the overall confidence estimate.

3) OBJECT DISTANCES CHECKING FILTER

Finally, we scored each plane in S_{pp} according to criteria *c4*: how close all objects in the scene S_o are to the potential ground planes. Here, we utilized the knowledge that far more objects will be on the ground than any other plane, and in particular that people walk on the ground plane. Since some objects, such as decorations or lights can be on potential ground planes like the ceiling or walls, we assign higher weights to moving objects.

In order to calculate the object-to-plane distances, we align all the 3D object segments S_o and planes S_{pp_i} to the axes by applying roll and pitch rotation matrices R_{roll} and R_{pitch} found in section III-B. Since the ground plane is likely to be the highest or lowest plane in a 3D point cloud, our confidence estimate increased or decreased proportionally when the object-to-plane distance was smaller or larger than a learned value of one-fourth of the point cloud height. The contribution of the Object Distance Checking score to the overall confidence estimate function was represented by:

$$conf_{OD} = k_s \sum_{i=1}^{N_s} \frac{\left(\frac{h}{4} - D_{S_{o_i}^s}\right)}{\frac{h}{4}} + k_{mo} \sum_{i=1}^{N_{mo}} \frac{\left(\frac{h}{4} - D_{S_{o_i}^m}\right)}{\frac{h}{4}} \tag{11}$$

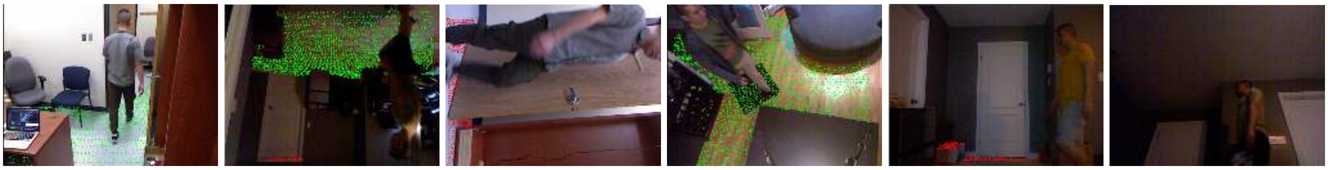
$$k_s = \frac{5}{N_s + W_{mo}N_{mo}} \tag{12}$$

$$k_{mo} = W_{mo}k_s \tag{13}$$

where k_s and k_{mo} are scaling factors for stationary and moving objects, $D_{S_{o_i}^s}$ denotes the absolute distance between a stationary object to plane S_{pp_i}, $D_{S_{o_i}^m}$ denotes the absolute distance between a moving human object to plane S_{pp_i}, N_s and N_{mo} denote the number of stationary object segments and the number of moving human objects, W_{mo} denotes the weight of the moving human object, and h denotes the height of the point cloud, as the confidence representation of each plane S_{pp_i}. Additionally, the constant scale factor of five in Eq.(12) is the relative weight of this component, maximizing the confidence of the real ground plane, and providing sufficient penalty to reduce the confidence of planes in the middle of the room, such as a table, to zero.

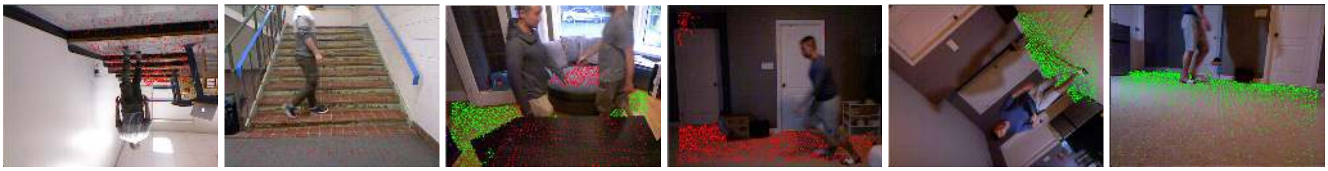
4) GROUND PLANE CONFIDENCE

A potential ground plane with a confidence estimate found with Eq.(7) that exceeded a learned confidence threshold ζ was then highly likely to be the true ground plane, suggesting that no further processing was required. However, we could



(a) 0°roll (b) 45°pitch 180°roll (c) 90°roll (d) -80°pitch (e) 0°pitch (f) 45°pitch

FIGURE 2. Ground plane estimation result examples with various camera orientations and locations.



(a) Multiple large planes (b) Multiple small planes (c) Multiple people (d) 2.6m/s moving speed (e) 3.0m/s moving speed (f) Half-body visible

FIGURE 3. Ground plane estimation result examples with special environmental scenarios.

only find the ground plane if the ground plane belonged to a plane in the set S_{pp}, and as such there may not be any planes such that $conf_{S_{pp}} > \zeta$. In many practical cases, the actual ground plane could be a small plane in the camera FOV, which would cause the ground plane to be segmented as an object or part of another object in S_o. Additionally, any surfaces that are close to the true ground plane, but having a larger area than the ground plane can lead to an incorrect identification of the true ground plane. Finally, the ground plane may not actually be visible in the scene. In these situations, we initiated a secondary ground plane estimation.

5) SECONDARY GROUND PLANE ESTIMATION

In the case where no plane from S_{pp} satisfied the condition $conf_{S_{pp}} > \zeta$, we applied RANSAC to each 3D object segment S_{o_i}, retrieving the largest plane within each object to generate set $S_{pp^{sd}}$ and iterating through the steps of sections III-A to III-C. If no plane had a confidence $conf_{S_{pp}} > \zeta$ after the secondary estimation, the ground plane did not exist in the camera FOV. In this scenario, the plane from S_{pp} that had the highest confidence was used to predict the ground plane. Using the distances $d_{max_i^+}$ and $d_{max_i^-}$ from section III-C.2, the ground plane formula was estimated as $Ax+By+Cz+D = d_{max_i^+}$ or $Ax + By + Cz + D = -d_{max_i^-}$, where (A, B, C, D) are the plane coefficients, based on which Object Distance Checking confidence was higher. However, if any plane in $S_{pp^{sd}}$ had a confidence $conf_{S_{pp^{sd}}} > \zeta$, the plane with the highest confidence was selected as the actual ground plane.

IV. EXPERIMENTS

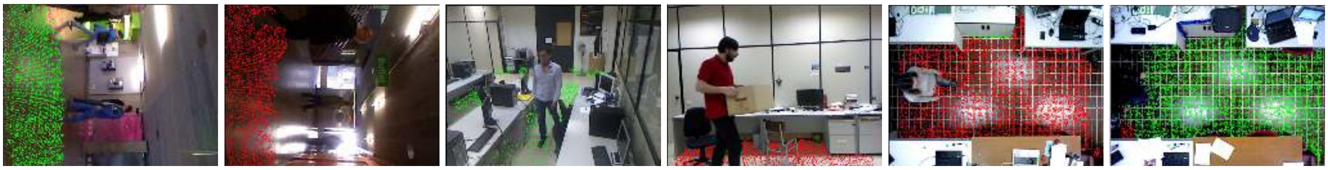
We evaluated our algorithm on our own dataset of generated video sequences, as well as on all relevant video sequences from three public datasets. In this way, we ensured our algorithm was generalizable, repeatable, and insensitive to artifacts that may be present in our own data collection. Specifically, we focused on representative scenarios with: a high variety of camera orientations; camera locations; ground plane size, shape and visibility; and room and occupant complexity.

A. GENERATED VIDEO SEQUENCE DATA

We collected video sequence data using the Kinect v1 which provides an RGB image and a depth image with a 27 frame per second rate (FPS) on average, image data we combine to form an RGB-D image, using a MacBook Pro (Retina, 13-inch, Mid 2014) with Dual core i5 CPU and 8G memory. We recorded video sequences by placing the camera in 24 unique scenarios, which included various combinations of different camera orientations and locations, multiple planes, multiple people, diverse moving speeds, and various body appearance ratios.

Our captured video sequences contained 40-140 data frames from the time the first person entered the camera's field of view or started moving to the time the last person left the camera FOV or stopped moving. Similar to the work of [36], we chose an MSAM block size of five frames. From experimentation, we determined that planes with a confidence score $\zeta > 8.5$ are highly likely to be the actual ground plane, while planes with a confidence score of $6.0 < \zeta < 8.5$ are planes that are parallel to the ground plane, and may be the ground plane. Based on our experimental results, the W_{mo} for the Object Distances Checking step in Section III-C is optimally set to 8.0 to ensuring the moving person becomes the decisive factor among all other stationary objects and noise. Figures 2 and 3 demonstrate some representative data sets and examples of our ground plane estimation results.

In the data preparation step, SIFT generated an average of approximately 4,000 keypoints in each full 2D image with 10 layers in each octave, 0.02 as contrast threshold, 20 as filter out edge-like features threshold, and 1.0 as sigma. The size of voxel grid down-sample filter for point cloud frames we selected was 2cm. The RANSAC distance threshold and



(a) RGBD People first_trial_2 (b) RGBD People second_trial_3 (c) SBM OutofRange_TopViewLab (d) SBM Shadow_genSeq2 (e) TVPR_g001 (f) TVPR_g003

FIGURE 4. Ground plane estimation result examples from public dataset sequences.

TABLE 1. Private dataset detailed results.

Trial Name	Top 3 confidences	Speed	Frames	# of Planes
speed_0.65	8.65, -, -	0.65	5	5
speed_1	9.17, -, -	1	5	5
speed_1.5	8.61, 3.87, 2.71	1.5	5	5
speed_2.6	9.55, 2.4, -	2.6	5	5
speed_3_with_rotation	9.285, -, -	3	35	4
2people_same_direction	8.65, 5.63, 5.78	1.2	15	12
2people_diff_direction	8.97, 6.54, 6.53	1.2	20	11
2people_-80_pitch_angle	8.65, 4.94, -	1.2	15	5
simple	9.39, -, -	1.2	10	3
bedroom_0_roll_angle	8.51, 6.78, 5.67	1.1	5	9
bedroom_30_roll_angle	9.03, 6.36, 3.93	1.1	5	7
bedroom_30_roll_angle_toward	9.54, 7.56, 5.54	1.2	10	6
bedroom_180_roll_angle	7.80, 7.10, 5.82	1.1	5	9
lab_90_roll_angle	8.21, -, -	1.2	15	1
lab_0_roll_angle_away	9.05, 4.48, 3.21	1.2	5	14
room_-80_pitch_angle	8.65, 4.78, 4.60	1.2	10	5
bedroom_-30_pitch_angle	9.10, -, -	1.2	5	3
bedroom_0_pitch_angle	8.88, 4.47, 4.30	1.2	10	5
bedroom_10_pitch_angle	7.48, 6.44, 4.99	1.2	20	4
bedroom_45_pitch_angle	-, -, -	1.1	-	1
bedroom_45_pitch_180_roll_angle	8.46, -, -	1.1	10	6
Multiple_small_planes	8.43, -, -	1	10	3
Multiple_large_planes	6.79, -, -	1	5	7
Small_body_appearance	9.17, 2.59, -	1.2	40	6

TABLE 2. Public dataset detailed results.

Dataset	Trial Name	Top 3 confidences	Frames	# of Planes
RGBD People [63] [64]	first_trial_1	8.14, -, -	5	2
	first_trial_2	9.03, 6.59, -	5	2
	first_trial_3	8.2, 7.05, -	5	3
	first_trial_4	8.43, 3.57, 1.24	5	3
	second_trial_1	8.27, 5.98, 3.50	5	4
	second_trial_2	8.10, 5.14, -	20	6
	second_trial_3	9.88, 4.64, 1.14	15	4
	second_trial_4	7.70, 2.03, -	10	5
	third_trial_1	8.64, 5.95, -	10	3
	third_trial_2	8.67, 7.91, -	15	3
	third_trial_3	7.64, 6.25, -	10	2
	third_trial_4	8.65, 4.13, -	10	2
SBM RGB-D [66]	IlluminationChanges_genSeq1	8.00, -, -	5	3
	OutofRange_MultiPeople1	8.03, 1.97, -	5	2
	OutofRange_MultiPeople2	8.12, 1.99, -	10	4
	OutofRange_TopViewLab	8.51, 3.19, -	20	12
	ColorCamouflage_Hallway	8.75, 4.01, 2.62	15	3
	DepthCamouflage_DCamSeq1	7.12, 3.98, 3.84	10	5
	DepthCamouflage - DCamSeq2	7.50, 4.53, 4.40	15	6
	Boostrapping_fall01cam	7.17, 1.55, -	5	4
	Shadow_genSeq2	7.57, 3.25, -	5	4
TVPR [65]	g001	8.70, 8.45, 5.26	20	5
	g002	8.02, 7.86, 3.82	25	7
	g003	8.93, 8.47, 2.70	15	5
	g004	8.92, 8.11, -	15	4
	g005	8.21, 5.54, 1.97	15	5

the cluster tolerance of Euclidean clustering were 2.5 and 2 times the voxel grid filter size respectively. Based on these parameter, we extracted anywhere from 4 to 10 planes from each scene, varying based on the indoor environment complexity and camera perspective. In the homography estimation step (section III-B), we set the block size to five to ensure we achieve sufficient histogram intersection between the reference frame F_1 and frame F_x. The number of SIFT feature keypoints on the human ranged from 150 to 380 out of the approximately 4,000 keypoints. In the experiments, the confidence of the results exceeded 7.5 even if the ground plane only occupied a small fraction in the FOVs in each scene where the ground plane was visible. Table 1 shows the confidence of the three planes that have the highest confidence, the human object's moving speed, the number of frames the algorithm took to estimate the ground plane, and the total number of ground plane candidates we had before the ground plane estimation checking steps. Notably, in Fig.2(f), the ground plane is not visible, so we do not identify any actual ground plane; rather we estimate the ground plane equation based on the ceiling plane function. For example, the estimated ground plane function for Fig.2(f) is $-0.046 \cdot x + (-0.699 \cdot y) + 0.713 \cdot z + (-0.0) = 0$, a scene in which only the ceiling is visible.

B. PUBLIC DATASETS

We evaluated our algorithm's accuracy on three public datasets: the RGB-D People Dataset [63], [64], the SBM-RGBD Dataset [66] and the TVPR Dataset [65]. The large ground plane that is directly visible in the RGB-D People and TVPR Datasets allow our algorithm produce high confidence estimates for the ground plane - even higher than those generated from our more challenging data trials, with results shown in Table 2. Figure 4 shows some sample results obtained from the dataset trials in these public datasets.

V. DISCUSSION AND CONCLUSIONS

In this paper, we proposed a novel ground plane estimation method using the combination of 2D and 3D data analyses.

Existing ground plane detection approaches require that significant assumptions are met (e.g., that the ground plane is the largest plane in the scene, the ground plane is at the bottom of the sensor field of view, that the ground plane is constant in colour or texture). These assumptions are not practical in dynamic or cluttered environments, or in situations where the sensor orientation or location are unknown, requiring more expensive and specialized equipment (e.g., to detect sensor orientation). Our approach robustly finds the indoor ground plane with unrestrictive assumptions: the sensors is an RGB-D camera; at least one person smoothly walks in the scene with most parts of the body visible within the camera field of view; and the human body is perpendicular to the ground plane while walking.

We first segment the point cloud that is generated from a pair of RGB and depth images into planes and object segments, while finding the SIFT 2D key points in the RGB image. This fundamental step requires the large planes and object segments corresponding to the real world objects and a sufficient amount of 2D feature key points. In general scenarios, our algorithm successfully segments all the planes and objects in the scene and provides a sufficient amount of SIFT feature points with the parameters we used in the experiments. Our algorithm can fail for one MSAM block if the majority of the human is not segmented as a single object segment, if no planes can be found in the FOV (i.e., RANSAC generates unreasonable planes), or if the 2D feature key points are too sparse to generate reliable homographies. However, these issues were resolved for all our trials by processing through the entire trial data set.

In the second step, we project 3D object segments to the 2D RGB image to find the regions that only contain the keypoints belonging to these objects, and apply MSAM to each region to find the decomposition of reliable homographies. MSAM splits the keypoints within each region in a tree structure taking 30-60 seconds to process with parallel threads, which makes real-time ground plane estimation unfeasible. Building object segment sequences within one block and identifying the human is achieved by calculating the histogram intersection ratio between two object segments. This approach is sensitive to movement in any direction; it provided 90% accuracy while matching corresponding object segments within a block, and only fails when Kinect sensor generates significant depth error. In addition, because of the depth error of our hardware sensor [52], [53], estimating the ground plane with only one block is not guaranteed for a video sequence because object translation could appear to occur in both directions for short sequences.

The final step builds the ground plane estimation confidence based on homography decomposition vectors, plane relative positions, and the distances between the planes and other objects. With only one iteration of the confidence estimation, our algorithm successfully estimated any ground plane that was large in the FOV. Only one additional iteration was required to retrieve the ground plane if it was smaller in the FOV. Our approach of identifying humans from all other objects is naive, mainly depending on the gross shape of the moving object segment and the correlation between the homography trajectory vector and moving object's bounding box. In some situations, such as if only the torso of the human (which has a similar dimension in both the $x-$ and $y-axes$) is segmented as a moving object, our algorithm will ignore this potentially valid segment. Similarly, sequences exemplified in Figures 2(e) and 2(f) take significantly more frames to estimate the ground plane because the movement of the human's arms and legs changed the bounding box's dimensions of the human. The current solution is processing through the full trial data set until the algorithm identifies the human body, while this issue could be potentially solved by synchronizing with other sensor in the system viewing the same scene from a different perspective. Furthermore, due to the limitations of our camera's depth sensor (specifically lens distortion), any wall characterized by the $x-$ and $y-axes$ often consisted of multiple layers of points. The RANSAC algorithm in the data preparation step yielded one slice of the wall as an object segment with approximately one third probability, which had an almost equal distance to both the ceiling and the ground plane. Conditions like this led to us increasing the confidence weight of the moving objects relative to non-moving objects, enlarging the difference between the ceiling's confidence and the ground plane's confidence. Specific to the distortion issues associated with segmenting the wall, we also increased the RANSAC distance threshold between models to reduce the number of slices generated from one wall; an issue that could easily be rectified by using a sensor with a higher depth resolution and accuracy. Accordingly, the correct ground plane estimation results heavily relied on finding the accurate human (moving object). We noted that increasing the RANSAC distance threshold between models also had drawbacks: multi-plane surfaces, such as stairs (fig.3(b)) and stages (fig.3(a)), are merged as one plane. Since the resulting single plane representing the stairs has a large angle value relative to the ground plane (fig.5; approximately 45°), it is found by our algorithm as a potential plane, but is ultimately given a low confidence as the actual ground plane. (fig.3(b)). Similarly, the plane corresponding to the stage floor barely exceeds our $\zeta > 6.0$ threshold (fig.3(a)) because this plane (comprised of points from the stage plane and lower ground plane) is not parallel to the ground plane leading the low confidence provided by the Object Distances Checking step.

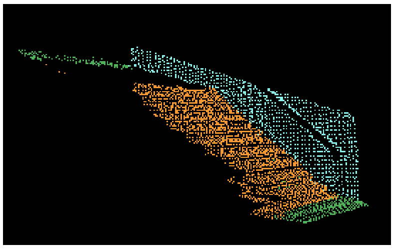


FIGURE 5. Point cloud plane segments for stairs.

We evaluated our approach on our own dataset, which included 24 unique scenarios (e.g., sensor perspectives and orientations, number of persons walking in the scene), as well as on three public datasets (see [63], [64], [66] and [65]), where we included 26 additional scenarios. Our approach robustly estimated the ground plane directly (when the plane was visible) or indirectly (when the plane was not visible) with a large variety of sensor orientations, different ground plane area sizes, room complexities, and multiple persons in the scene in 50 of 50 scenarios (100%). Our experimental results show that our algorithm is insensitive to the movement speed of walking humans and is tolerant to partial occlusion of the human body. In cases where the ground plane is not visible the scene, we successfully estimated the ground plane formula by translating the plane with the highest confidence in the scene, suggesting that other sensors that can see the ground plane can help to accurately find the ground plane. This is exemplified through two scenes (e.g., Figures 2(b) and 2(f)) where we successfully identify the ground plane directly in once case ($conf_{HP} = 2.99$, $conf_{RP} = 2.0$, $conf_{OD} = 3.99$, $conf_S = 8.99$), and indirectly in another ($conf_{HP} = 2.56, conf_{RP} = 2.0, conf_{OD} = 0.0, conf_S = 4.56$). In all cases, we were able to find the ground plane or a plane parallel to the ground plane using RGB-D sensors data without any pre-calibration or a prior knowledge of the sensor location or orientation.

In the future, we will focus on improving the performance of the algorithm; switching to a better RGB-D sensors which provides higher quality data; enhancing the robustness and accuracy of the human object detection algorithm; and achieving potential human recognition or identification within a RGB-D camera system. In addition, we will also test our algorithm on video sequences that have higher indoor complexity and more people visible in the scene.

REFERENCES

[1] V. Waran, V. Narayanan, R. Karuppiah, D. Pancharatnam, H. Chandran, R. Raman, Z. A. A. Rahman, S. L. F. Owen, and T. Z. Aziz, “Injecting realism in surgical training–initial simulation experience with custom 3D models,” *J. Surgical Edu.*, vol. 71, no. 2, pp. 193–197, Mar. 2014.

[2] A. G. Bruzzone and F. Longo, “3D simulation as training tool in container terminals: The TRAINPORTS simulator,” *J. Manuf. Syst.*, vol. 32, no. 1, pp. 85–98, Jan. 2013.

[3] Z. Duan, Z. Yuan, X. Liao, W. Si, and J. Zhao, “3D tracking and positioning of surgical instruments in virtual surgery simulation,” *J. Multimedia*, vol. 6, no. 6, pp 502–509, 2011.

[4] R. Sacks, C. M. Eastman, and G. Lee, “Parametric 3D modeling in building construction with examples from precast concrete,” *Autom. Construct.*, vol. 13, no. 3, pp. 291–312, May 2004.

[5] V. R. Kamat and J. C. Martinez, “Visualizing simulated construction operations in 3D,” *J. Comput. Civil Eng.*, vol. 15, no. 4, pp. 329–337, Oct. 2001.

[6] Y. W. D. Tay, B. Panda, S. C. Paul, N. A. Noor Mohamed, M. J. Tan, and K. F. Leong, “3D printing trends in building and construction industry: A review,” *Virtual Phys. Prototyping*, vol. 12, no. 3, pp. 261–276, Jul. 2017.

[7] K. Adams, “3D enhancements to gaming components in gaming systems with real-world physics,” U.S. Patent Appl. 10 115 261, Oct. 30, 2018.

[8] T. Argles, S. Minocha, and D. Burden, “Virtual field teaching has evolved: Benefits of a 3D gaming environment,” *Geol. Today*, vol. 31, no. 6, pp. 222–226, Nov. 2015.

[9] A. Kulshreshth, K. Pfeil, and J. J. LaViola, “Enhancing the gaming experience using 3D spatial user interface technologies,” *IEEE Comput. Graph. Appl.*, vol. 37.3, no. 2017, pp. 16–23.

[10] E. Kruijff and B. E. Riecke, “Navigation interfaces for virtual reality and gaming: Theory and practice,” in *Proc. Extended Abstr. CHI Conf. Hum. Factors Comput. Syst.*, New York, NY, USA: ACM, 2018, p. C11.

[11] D. M. Swanson, “Benefits of 3D breast tomosynthesis combined with 2D digital mammography in screening women for breast cancer,” Univ. North Dakota, Grand Forks, ND, USA, Physician Assistant Scholarly Project Posters 156, 2019.

[12] Y. Kudo and N. Ikeda, “Benefits of lung modeling by high-quality three-dimensional computed tomography for thoracoscopic surgery,” *Video-Assist. Thoracic Surg.*, vol. 4, p. 4, Feb. 2019.

[13] S. J. Trenfield, A. Awad, A. Goyanes, S. Gaisford, and A. W. Basit, “3D printing pharmaceuticals: Drug development to frontline care,” *Trends Pharmacolog. Sci.*, vol. 39, no. 5, pp. 440–451, May 2018.

[14] A. Hertault, B. Maurel, F. Pontana, T. Martin-Gonzalez, R. Spear, J. Sobocinski, I. Sediri, C. Gautier, R. Azzaoui, M. Rémy-Jardin, and S. Haulon, “Benefits of completion 3D angiography associated with contrast enhanced ultrasound to assess technical success after EVAR,” *Eur. J. Vascular Endovascular Surg.*, vol. 49, no. 5, pp. 541–548, May 2015.

[15] D. Mendes, F. Relvas, A. Ferreira, and J. Jorge, “The benefits of DOF separation in mid-air 3D object manipulation,” in *Proc. 22nd ACM Conf. Virtual Reality Softw. Technol. (VRST)*. New York, NY, USA: ACM, 2016, pp. 261–268.

[16] A. Buyval, I. Afanasyev, and E. Magid, “Comparative analysis of ROS-based monocular SLAM methods for indoor navigation,” in *Proc. 9th Int. Conf. Mach. Vis. (ICMV)*, vol. 10341, Mar. 2017, Art. no. 103411K.

[17] M. Vlaminck, Q. L. Hiep, V. N. Hoang, H. Vu, P. Veelaert, and W. Philips, “Indoor assistance for visually impaired people using a RGB-D camera,” in *Proc. IEEE Southwest Symp. Image Anal. Interpretation (SSIAI)*, Mar. 2016, pp. 161–164.

[18] S. Brahmbhatt, J. Gu, K. Kim, J. Hays, and J. Kautz, “Geometry-aware learning of maps for camera localization,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2616–2625.

[19] Y.-H. Su, K. Huang, and B. Hannaford, “Real-time vision-based surgical tool segmentation with robot kinematics prior,” in *Proc. Int. Symp. Med. Robot. (ISMR)*, Mar. 2018, pp. 1–6.

[20] H. Shin, H. Hwang, H. Yoon, and S. Lee, “Integration of deep learning-based object recognition and robot manipulator for grasping objects,” in *Proc. 16th Int. Conf. Ubiquitous Robots (UR)*, Jun. 2019, pp. 174–178.

[21] R. A. Zeineldin and N. A. El-Fishawy, “Fast and accurate ground plane detection for the visually impaired from 3D organized point clouds,” in *Proc. SAI Comput. Conf. (SAI)*, Jul. 2016, pp. 373–379.

[22] T.-J. Lee, D.-H. Yi, and D.-I. Cho, “A monocular vision sensor-based obstacle detection algorithm for autonomous robots,” *Sensors*, vol. 16, no. 3, p. 311, 2016.

[23] P. Skulimowski, M. Owczarek, and P. Strumiłło, “Ground plane detection in 3D scenes for an arbitrary camera roll rotation through ‘V-disparity’ representation,” in *Proc. Federated Conf. Comput. Sci. Inf. Syst.*, Sep. 2017, pp. 669–674.

[24] J. Zhao, J. Katupitiya, and J. Ward, “Global correlation based ground plane estimation using V-Disparity image,” in *Proc. IEEE Int. Conf. Robot. Autom.*, Apr. 2007, pp. 529–534.

[25] P. Herghelegiu, A. Burlacu, and S. Caraiman, “Robust ground plane detection and tracking in stereo sequences using camera orientation,” in *Proc. 20th Int. Conf. Syst. Theory, Control Comput. (ICSTCC)*, Oct. 2016, pp. 514–519.

[26] M. Vaz and R. Ventura, “Real-time ground-plane based mobile localization using depth camera in real scenarios,” *J. Intell. Robotic Syst.*, vol. 80, nos. 3–4, pp. 525–536, Dec. 2015.

[27] J. Arroóspide, L. Salgado, M. Nieto, and R. Mohedano, “Homography-based ground plane detection using a single on-board camera,” *IET Intell. Transp. Syst.*, vol. 4, no. 2, p. 149, 2010.

[28] D. Conrad and G. N. DeSouza, “Homography-based ground plane detection for mobile robot navigation using a modified EM algorithm,” in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2010, pp. 910–915.

[29] S. Kumar, A. Dewan, and K. M. Krishna, “A bayes filter based adaptive floor segmentation with homography and appearance cues,” in *Proc. 8th Indian Conf. Comput. Vis., Graph. Image Process. (ICVGIP)*. New York, NY, USA: ACM, 2012, p. 54.

[30] P. Ke, C. Meng, J. Li, and Y. Liu, "Homography-based ground area detection for indoor mobile robot using binocular cameras," in *Proc. IEEE 5th Int. Conf. Robot., Autom. Mechatronics (RAM)*, Sep. 2011, pp. 30–34.

[31] R. Labayrade, D. Aubert, and J.-P. Tarel, "Real time obstacle detection in stereovision on non flat road geometry through 'v-disparity' representation," in *Proc. IEEE Intell. Vehicle Symp.*, vol. 2, Jun. 2002, pp. 646–651.

[32] Z. Jin, T. Tillo, and F. Cheng, "Depth-map driven planar surfaces detection," in *Proc. IEEE Vis. Commun. Image Process. Conf.*, Dec. 2014, pp. 514–517.

[33] D. Kırcalı and F. B. Tek, "Ground plane detection using an RGB-D sensor," in *Information Sciences and Systems*. Cham, Switzerland: Springer, 2014, pp. 69–77.

[34] R. Dragon and L. V. Gool, "Ground plane estimation using a hidden Markov model," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 4026–4033.

[35] A. Cherian, V. Morellas, and N. Papanikolopoulos, "Accurate 3D ground plane estimation from a single image," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2009, pp. 2243–2249.

[36] R. Dragon, B. Rosenhahn, and J. Ostermann, "Multi-scale clustering of Frame-to-Frame correspondences for motion segmentation," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2012, pp. 445–458.

[37] R. O. Duda and P. E. Hart, *Use of the Hough Transformation to Detect Lines and Curves in Pictures*. Menlo Park, CA, USA: SRI International, 1971.

[38] Y. Man, X. Weng, X. Li, and K. Kitani, "GroundNet: Monocular ground plane normal estimation with geometric consistency," in *Proc. 27th ACM Int. Conf. Multimedia*, Oct. 2019, pp. 2170–2178.

[39] D. Zhou, Y. Dai, and H. Li, "Ground-plane-based absolute scale estimation for monocular visual odometry," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 2, pp. 791–802, Feb. 2020.

[40] Y. C. Lim and M. Kang, "Stereo-based pedestrian detection using the dynamic ground plane estimation method," in *Proc. 2nd Int. Conf. Commun. Inf. Process. (ICCIP)*, Nov. 2016, pp. 110–114.

[41] D. Borrmann, J. Elseberg, K. Lingemann, and A. Nüchter, "The 3D Hough transform for plane detection in point clouds: A review and a new accumulator design," *3D Res.*, vol. 2, no. 2, p. 3, Jun. 2011.

[42] D. Holz, S. Holzer, R. B. Rusu, and S. Behnke, "Real-time plane segmentation using RGB-D cameras," in *Robot Soccer World Cup*. Berlin, Germany: Springer, 2011, pp. 306–317.

[43] S. Choi, J. Park, J. Byun, and W. Yu, "Robust ground plane detection from 3D point clouds," in *Proc. 14th Int. Conf. Control, Autom. Syst. (ICCAS)*, Oct. 2014, pp. 1076–1081.

[44] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 11, pp. 1330–1334, Nov. 2000.

[45] J. Heikkila and O. Silven, "A four-step camera calibration procedure with implicit image correction," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 97, Jun. 1997, p. 1106.

[46] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[47] R. B. Rusu, "Semantic 3D object maps for everyday manipulation in human living environments," *KI-Künstliche Intelligenz*, vol. 24, no. 4, pp. 345–348, Nov. 2010.

[48] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proc. 7th IEEE Int. Conf. Comput. Vis.*, Sep. 1999, vol. 99, no. 2, pp. 1150–1157.

[49] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004.

[50] A. Criminisi, I. Reid, and A. Zisserman, "A plane measuring device," *Image Vis. Comput.*, vol. 17, no. 8, pp. 625–634, Jun. 1999.

[51] Y. Ma, *An Invitation to 3-D Vision: From Images to Geometric Models*, vol. 26. Springer, 2012.

[52] K. Khoshelham, "Accuracy analysis of kinect depth data," in *Proc. ISPRS Workshop Laser Scanning*, 2011, vol. 38, no. 1, pp. 133–138.

[53] K. Khoshelham and S. O. Elberink, "Accuracy and resolution of kinect depth data for indoor mapping applications," *Sensors*, vol. 12, no. 2, pp. 1437–1454, 2012.

[54] P. Jaccard, "The distribution of the flora in the alpine zone.1," *New Phytologist*, vol. 11, no. 2, pp. 37–50, Feb. 1912.

[55] T.-K. Lee, S. Lim, S. Lee, S. An, and S.-Y. Oh, "Indoor mapping using planes extracted from noisy RGB-D sensors," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2012, pp. 1727–1733.

[56] R. Guo, D. Zhou, K. Peng, and Y. Liu, "Plane based visual odometry for structural and low-texture environments using RGB-D sensors," in *Proc. IEEE Int. Conf. Big Data Smart Comput. (BigComp)*, Feb. 2019, pp. 1–4.

[57] T. Liu, Y. Liu, Z. Tang, and J.-N. Hwang, "Adaptive ground plane estimation for moving camera-based 3D object tracking," in *Proc. IEEE 19th Int. Workshop Multimedia Signal Process. (MMSP)*, Oct. 2017.

[58] A. Jamal, P. Mishra, S. Rakshit, A. K. Singh, and M. Kumar, "Real-time ground plane segmentation and obstacle detection for mobile robot navigation," in *Proc. INTERACT*, Dec. 2010, pp. 314–317.

[59] K. Lee, C. Yon Choo, H. Qing See, Z. Jiang Tan, and Y. Lee, "Human detection using histogram of oriented gradients and human body ratio estimation," in *Proc. 3rd Int. Conf. Comput. Sci. Inf. Technol.*, vol. 4, Jul. 2010, pp. 18–22.

[60] R. Eshel and Y. Moses, "Homography based multiple camera detection and tracking of people in a dense crowd," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.

[61] S. M. Khan and M. Shah, "A multiview approach to tracking people in crowded scenes using a planar homography constraint," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2006, pp. 133–146.

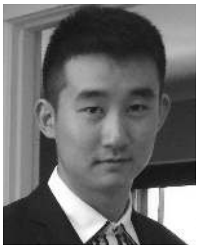
[62] C. Zhang and S. Czarnuch, "Perspective independent humanoid object detection by 2D and 3D data analysis," in *Proc. Newfoundland Elect. Comput. Eng. Conf.* St. John's, NL, Canada: IEEE Newfoundland Labrador Section, Jan. 2019.

[63] L. Spinello and K. O. Arras, "People detection in RGB-D data," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2011, pp. 3838–3843.

[64] M. Luber, L. Spinello, and K. O. Arras, "People tracking in RGB-D data with on-line boosted target models," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2011, pp. 3844–3849.

[65] D. Liciotti, M. Paolanti, E. Frontoni, A. Mancini, and P. Zingaretti, "Person re-identification dataset with RGB-D camera in a top-view configuration," in *Video Analytics. Face and Facial Expression Recognition and Audience Measuremen*. Cham, Switzerland: Springer, 2016, pp. 1–11.

[66] M. Camplani, L. Maddalena, G. Moyá Alcover, A. Petrosino, and L. Salgado, "A benchmarking framework for background subtraction in RGBD videos," in *Proc. Int. Conf. Image Anal. Process.* (Lecture Notes in Computer Science), S. Battiato, G. Gallo, G. M. Farinella, and M. Leo, Eds. Cham, Switzerland: Springer, 2017. [Online]. Available: http://rgbd2017.na.icar.cnr.it/SBM-RGBDdataset.html



CHENGSI ZHANG received the B.S. degree from the Guizhou University of China and the M.S. degree in computer engineering from Memorial University. He is currently pursuing the Ph.D. degree in computer engineering with the Faculty of Engineering and Applied Science, Memorial University of Newfoundland.

From 2016 to 2017, he was a Research Assistant with the Faculty of Computer Science at Memorial. In 2018, he started the position with Compusult Ltd as a Software Developer focusing on 2-D/3-D map management and analysis. His research field is focusing on 3-D data analysis, image recognition/reconstruction/registration, and machine learning.

STEPHEN CZARNUCH (Member, IEEE) is currently an Assistant Professor of biomedical engineering, joint-appointed to the Faculty of Engineering and Applied Science and the Faculty of Medicine, Memorial University. His research lies at the intersection of engineering, computer science, medicine, gerontology, rehabilitation, psychology, and sociology. Specifically, he seeks to develop relevant, accessible, acceptable and adoptable patient-oriented technological interventions for vulnerable populations. Further, he emphasizes the evaluation of device efficacy by connecting academic research with providers and patients in a real-world context. His technical focus is on machine learning, deep learning, and computer vision.

• • •