



Contents lists available at ScienceDirect

Engineering Science and Technology, an International Journal

journal homepage: www.elsevier.com/locate/jestch



Full Length Article

A voxelize structured refinement method for registration of point clouds from Kinect sensors



Erdal Özbay ^{*}, Ahmet Çinar

Computer Engineering Dep., Firat University, 23119 Elazig, Turkey

ARTICLE INFO

Article history:

Received 29 June 2018

Revised 25 September 2018

Accepted 26 September 2018

Available online 6 October 2018

Keywords:

Alignment

Microsoft Kinect

Point cloud

Refinement

3D scanning

ABSTRACT

3D scanning of objects has been widely used for many years in computer graphics and computer vision. There are a variety of solutions in this area, such as the motion or multiple sensors for scanning. In this study, we propose an approach that generates a scan with a natural motion of the user, through a fixed Kinect sensor whose usage is more practical and cost-effective than conventional 3D scanners. Local voxelized structure based on (LVS) is proposed for efficient 3D point cloud, captured by Kinect as low-quality. The approach allows the generation of full point cloud data in a wide range of indoor and short-range 3D objects. The developed system for object scanning is easy to set up, generating simple and impressive results. The 3D object standing on the turntable facing a single fixed Kinect sensor is rotated at specific angles (e.g. 90°) to obtain multiple point cloud scan data. Afterward, the center of gravity of each scanned point cloud data is shifted into (0,0,0) origin position for merging and aligning operations. So subsequent scans are obtained. The point cloud data obtained from the second and subsequent scans are transformed in the y-axis direction with respect to the center point (0,0,0), respectively. In some case, the axis-x and axis-z can be used for rotating too. The transformed point cloud data obtained from the different angles are aligned with respect to each other, shifted according to the determined merging key points. An algorithm that runs on the sections of point cloud for refinement operation is performed on a complete 3D point cloud data. Thus, the resulting scan has a 3D, clean and orderly structure free from the data crowd. Our approach has verified over a large number of users and different 3D objects and compared with a reference scan according to metric specifications.

© 2018 Karabuk University. Publishing services by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

1. Introduction

Realistic 3D objects and human body modeling are widely used in many areas such as computer games, computer vision, animation, computer graphics, and human-computer interaction. However, it is not an easy task to obtain the suitable model. Today, highly accurate measurements can be obtained very close to reality using 3D structural light scanning and rapid laser scanning techniques [1]. For example, 3D modeling and structural analysis operations carry out on structures belonging to cultural heritage using high-precision terrestrial laser scanners are among valuable works [2]. Another worthwhile work using MapFuse method for completion and optimization of 3D models [3]. The difficulties in implementing and the high cost of the systems are obstacles to the widespread use of these techniques. Furthermore, an expert knowledge is often consulted to manage the devices used in these

systems. The main problem of object scanning, while creating a full 3D model, one of the camera or object, and sometimes both of them had to move. Similar distresses may occur during the extraction of the human 3D model, depending on the human not to be stably standing. Different approaches have been put forward in order to overcome such negativities. In both 3D object modeling and human body modeling, image-based methods are frequently used. The state of the art multi-view method can provide effective and accurate results [4]. In the shape from silhouettes (SFS) method, synchronous use of more than one sensor can produce effective results [5,6]. The most important advantage of this method is that it is sufficient to look at the silhouette information on the textures only. But, these methods have high computational costs, and occlusions in sparse or complex textures from different views cause problems depending on the number of synchronized cameras [7].

The Microsoft Kinect range-sensor cameras, has attracted the attention of researchers working in the area of computer graphics on distance measurements. Compared to known 3D scanners, it can capture image and depth at video rate. Besides, they may also

* Corresponding author.

E-mail addresses: erdalozbay@firat.edu.tr (E.Özbay), acinar@firat.edu.tr (A.Çinar).

Peer review under responsibility of Karabuk University.



Fig. 1. Scanning system setup.

be sensitive to light and texture conditions. Easy to carry, simple, and low cost are the reasons why Kinect is often preferred. Researchers have used the depth information of physical scenes to reveal many studies. In recent years, many of them have used the Kinect camera as a scanner in their similar works. In some studies, Kinect is used to build dense 3D maps of indoor environments [8]. Furthermore, Kinect sensors have a remarkably low x , y , z resolution and depth accuracy for 3D scanning [9]. In addition, after 360° scanning, the individual captures must be collected and the pattern must be aggregated up to the correct alignment. A super-resolution method has described in one of the studies to improve the resolution values of the scan data [10,11]. There are also studies using external graphics hardware with Kinect in this field [12]. In another work, Kinect has used to compute a high-quality personalized avatar by means of a single (RGB) image and the corresponding depth map [13]. Real-time captured facial expressions are mapped into specific characters.

In recent other works, a SCAPE model has been introduced to predicate the human body using a single Kinect using silhouette images and depth data [14]. But this method considers pose and shape deformations separately. On the other hand, tensor-based model approach deals with pose and shape deformations together [15].

In this paper, as shown in Fig. 1, we introduce a system that performs a full scan of 3D objects in an indoor area using a single fixed Kinect sensor. A refinement process has been applied to the point clouds in geometric alignment used for 360-degree scanning. Thus, a closed 3D shape scan data is generated with reasonable quality and a much smaller number of regular data than a normal scanned data. First, the object in opposite a single fixed Kinect is scanned at different angles with a turntable. The number of scans is depended on the state of the occlusion in object scanning. The scans are obtained from 360° completed angles in total. The overlapping regions are then subjected to a refinement process. Each scan is shifted to center data (0,0,0) for implementation of geometric transformation. This process will provide the correct point conversion. A sequence of key poses obtained from the different angles is aligned with each other, paired with the selected alignment key points. Following alignment, a series of refinement operation is applied to the resulting full 3D scan, rendering the data more natural.

2. Materials and methods

The technologies used in 3D object scanning methods generally use either active systems such as laser or structural light stripes scanners or passive systems that operate directly such as image-

based systems [11]. While acquiring of depth images at the video rate, the capabilities of 3D or depth sensors stand out in new scanning approaches. Kinect sensors offer more clean data and relatively less noise scanning than Time of Flight (ToF) cameras. Robust alignment techniques are used to register the scans of this structured data such as Iterative Closest Points (ICP) and its variants or global rigid alignment [16,17]. The merging method can be applied to create a single 3D structure mesh following rigid alignment [18,19]. In some works, a procedure has been implemented in which hand-held cameras rotate freely around a stationary 3D object. In other cases, on the contrary, the 3D objects rotated with e.g. a turntable, against fixed 3D sensors are scanned from different angles [20].

3D object scanning and modeling activities are roughly categorized into four different scenarios according to the number of cameras and the motion of the 3D object. These different approaches lead to the development of different fields of study. These scenarios can be categorized as follows; (1) moving object with a single fixed sensor (MOSFS), (2) static object with a single moving sensor (SOSMS), (3) static object with multiple fixed sensors (SOMFS), and (4) moving object with multiple fixed sensors (MOMFS).

2.1. (MOSFS) moving object with a single fixed sensor

MOSFS based approaches are generally applied to depth based statistical learning methods. In these methods, the shape of the object is estimated by matching it with the fitting image silhouettes and depth data to form a SCAPE model [14]. A scheme is learned in which the best model matches are predicted. The final model is derived from a candidate subspace mapping. The implementation process for this approach can be relatively slow.

In a work that suggests a simple 3D scanner, the shape is measured like the shadow of a moving pencil over the object. This approach is only suitable for static setup. There is no any movement around the object [21]. As an alternative to active object scanning studies, there are image-based passive approaches such as SFS reconstruction and stereo [20,22].

In our work, a scanning system has been realized with a turntable that can be controlled by the object movement instead of performing a troublesome learning process such as model matching.

2.2. (SOSMS) static object with a single moving sensor

In the scanning approaches of the SFS based methods, the sensor is mounted on a robot arm so that the arm moves circularly around the object. The sequential frames that are tracked during this circular motion are restricted by a determined fundamental

matrix of degrees of freedom (DoF) [23]. In order to estimate all of the fundamental matrix parameters, the re-projection errors of the corresponding epipolar tangents can be minimized [24]. Thus, 3D object models can be reconstructed in motion computed from these fundamental matrices.

In recent times, the KinectFusion scanning approach, which is one of the depth based methods, has been quite successful [25]. In this approach, operations such as camera pose estimation, volumetric shape representation, and integration scan alignment to the iterative closest point are carried out [12]. It is possible to perform dense tracking based on volumetric representation and reconstruction of a static 3D scene in real time [26]. Although it works on a static scene, the failure cases can occur during the registering of points in a moving scan. In order to achieve performance in real-time work, applications must be supported by the GPU. It is possible to align faster with the Kinect sensor compared to the ToF camera. Because it is more tolerant to systematic distortions. In this respect, it is not directly possible to apply the KinectFusion approach to ToF cameras. Some approaches such as super-resolution have been put forward to avoid denoising [17].

2.3. (SOMFS) static object with multiple fixed sensors

In SFS-based approaches, the results can be made impressive at the end of an energy minimization and regularization process using a surface-based representation [27]. There are also proposed approaches for matching stereoscopic 3D scene reconstruction and SLAM-based camera tracking to dense depth maps [28,29]. But the approach has a limit in which the disorders can be resolved and recovered. Similarly, the problems experienced failure in capturing concavities of 3D objects are tried to be solved by the combination of SFS and photo consistency based refinement in SFS approaches [30]. Researchers can generate the shape of the 3D objects directly in 3D modeling with the introduction of range sensors widely on the market. 3D objects and human body modeling are performed using commercially produced multiple calibrated depth range sensors. The point clouds of 3D objects need to be aligned obtained from different cameras. Although high accuracy has achieved up to now in both methods, factors such as synchronization of multiple cameras and elimination of interference between cameras are removed from the fact that this system can be applied in an indoor domestic application [31].

2.4. (MOMFS) moving object with multiple fixed sensors

The number of cameras used in this methods is inversely proportional to the motion of the object. Increasing the angle of object motion allows the number of fixed cameras to be reduced. If a good registration provides to combine multiple scans of 3D objects, it is suitable to increase the number of cameras. In some methods of SFS approaches, Colored Surface Points (CSP) of the object is achieved and registration by located them [32]. The 6-DoF rigid motion incorporates information from the CSP between two consecutive frames.

A good registration operation can be implemented using the articulated version of the ICP algorithm in depth based approaches [33]. Nonetheless, pairwise non-rigid geometric registration or global registration can be performed iteratively [34]. As a result, all these methods require to use at least 3 or 4 synchronized sensor cameras. In addition, during the scanning, the object facing the camera needs to remain rigid or moved on a turntable.

Acquiring geometric content of real physical 3D objects is a basic concept when working on computer vision and computer graphics fields. Today, unfortunately, there is no ready-made system at a low cost that provides high quality and resolution distance

information of 3D scenes in real time [35]. Devices such as laser and structural light, which can produce high-quality scanning, are expensive and required expert knowledge use, such as the average price for Cyberware Scanner is \$ 220,000. Data acquisition can also be done using depth cameras with low-cost. These sensors work with two different approaches for depth measurement. The first one is called the time-of-flight (ToF), which resolves the distance over the flight time of the light signal between the object and the camera for each point of view. The SwissRanger is an industrial ToF camera, which provides high-resolution 3D image data in real time. The price of SR4000/SR4500 3D is about \$ 10,000. The second one uses an infrared pattern. The depth of the scene is calculated by coding through an infrared pattern deformation reflected the scene. These cameras are less costly than ToF sensors using infrared imager. Whereas the Kinect camera is about \$ 100.

In this work, we introduce a domestic application system that performs a full 3D scan using a single fixed Kinect. The domestic application can be easily used by daily users. Only Kinect is used for reliable shape captures. Although it is simple to use and cost-effective. For this reason, a refinement algorithm has been applied to obtain a regular 3D structure and naturalize the alignments.

3. Scanning system

The goal of the study is to generate a 3D shape scanner which produces a 360° 3D point cloud of real objects based on a ToF camera using a manually operated hand-held turntable [36]. The system consists of an active Kinect camera working with an infrared system and a 3D object standing on the turntable in front of it [37]. A series of scans are performed according to different angles of the same object with the help of the turntable. During this operation, while the Kinect remains stationary, the 3D object on the turntable is rotated manually. The number of scans and the angle of rotation are dependent on the visibility of the 3D object. When there is no visible occlusion, the scan degree is completed to 360. Scan angle has been selected in 90°. In this case, an excellent full scan cannot be expected. However, the incomplete or missing points are corrected by the refinement process after the 3D transformation and alignment processes [38]. The proposed method does not require a high-level data entry. So the system setup is simple, easy and low cost [39].

[Fig. 2](#) shows raw data from different angles of the human body. The 3D object facing the camera is rotated by 90° for each scan using a turntable. As a result, scan data is completed to 360 with little occlusion. When the four generated scans are displayed on the same coordinate plane, the resulting overlap is shown in [Fig. 2b](#).

3.1. Definition of cubic volume

LVS uses a local cubic volume to divide neighbor areas regularly and uniformly. Cubic volume occupations are widely applied in computer vision fields. The cubic volume structure has been orientated to attain rotation invariance. The details of the cubic volume definition are introduced in [Fig. 3](#).

In order to clarify the structure of the object, the local cubic-intersecting surface is calculated and partitioned in the space. Cubic volume provides advantages for fast and uniform space division. In this regard, a series of voxels are created, also it is called voxelization. The basic principle of LVS is based on the determination of whether one or more points exist in a voxel. Accordingly, the intersections of the voxel indices corresponding to the points of a specific voxel are examined first. The voxel index q_i on the

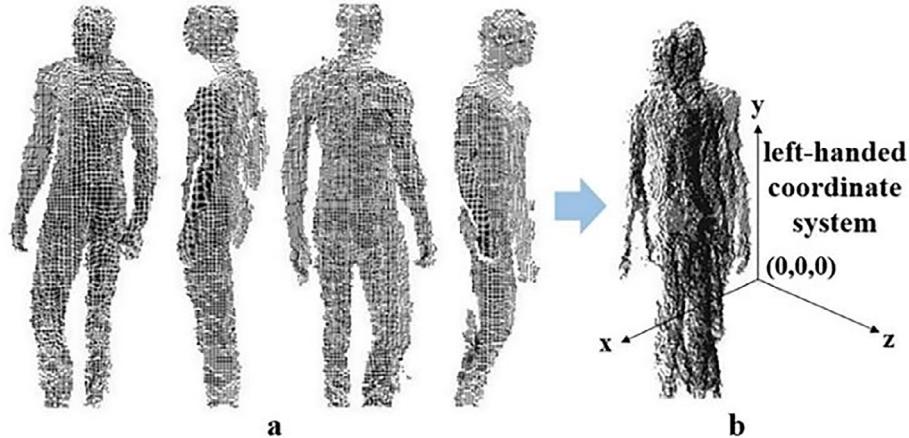


Fig. 2. (a) The system captures raw data of human body with a single Kinect from different angles and (b) display in the same coordinate system.

Voxel Grid

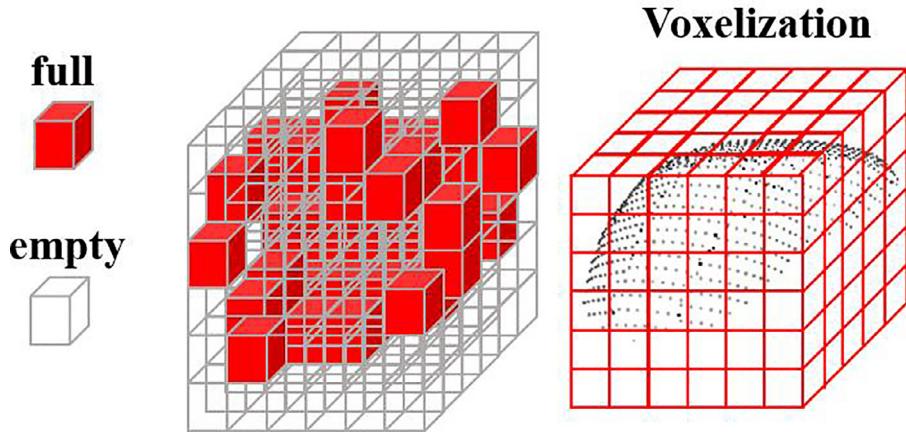
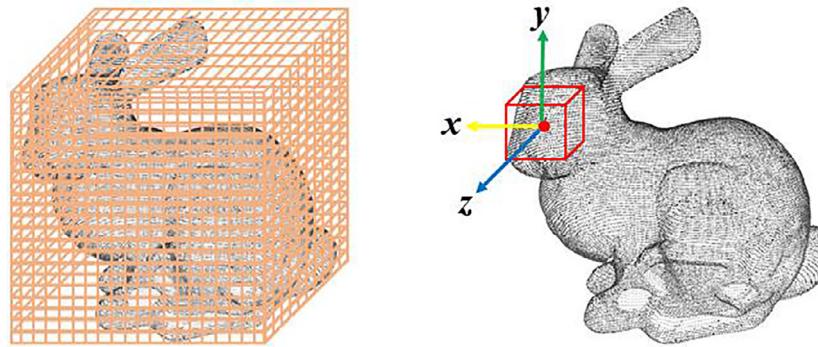


Fig. 3. Drawing of the LVS descriptor with cubic volume definition, voxelization and voxel label integration.

cubic-intersected surface Q_c of the default $N_v = m^3$ is obtained after voxelization calculation.

$$v_{id(q_i)} = \left\lfloor \frac{q_i.z + r}{l_{step}} \right\rfloor xm^2 + \left\lfloor \frac{q_i.y + r}{l_{step}} \right\rfloor xm + \left\lfloor \frac{q_i.x + r}{l_{step}} \right\rfloor \quad (1)$$

where m is the number of sections along each edge of the cubic volume, l is the partitioning step during the voxelization, i.e. $2r/m$. $\lfloor \cdot \rfloor$ demonstrate the round-down operation. After all points in Q_c have been indexed, a voxel v_i contains a point subset Q_{Vi} . Moreover, a label $l(v_i)$ is defined for voxel v_i with the help of Q_{Vi} as

$$l(v_i) = \begin{cases} 1 & \text{if } |Q_{Vi}| > 0 \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

in order to achieve binary structure characterization of a voxel, we have labeled each voxel in the cubic volume with 0 or 1. Then we integrate these labels bit by bit.

$$f_{LVS} = \{l(v_1), l(v_2), \dots, l(v_{N_v})\}, \quad (3)$$

This section shows that LVS has the potential to represent local shape geometry with significantly fewer bits and voxel points. This LVS procedure is expected to be invariant to rigid transformation and various challenges such as noise, point density variation, and point irregularity. This is summarized schematically in Fig. 4.

Firstly, LVS makes the geometric feature invariant to rigid transformations. Secondly, unlike the previous descriptors [40] taking a point as the feature encoding unit, the proposed LVS takes into

account the entire voxel as the coding unit and has good stability when faced with the point level perturbations such as noise, point density variation, and point irregularity. With the help of this technique, we also emphasize that the phenomenon of point irregularity is rarely explored in the literature, although it is quite common in point clouds captured by low-cost sensors like Kinect.

Algorithm 1.

Input: Get reference point cloud for each frame,
 $P_i^{\text{ref}}, \{i = 1 \dots n\}$.

Output: 3D transformed multiple scans.
 $RP_i^{\text{out}}, \{i = 1 \dots n\}$.

Calculate the center of gravity for each frame,
Initiate $x_{\max}, y_{\max}, z_{\max}, x_{\text{avg}}, y_{\text{avg}}, z_{\text{avg}}$ and count_{\max} .

while $i \leq n$ **do**

- $\text{count} \rightarrow \text{count} + 1$
- $x_{\max} \rightarrow x_{\max} + x_i, y_{\max} \rightarrow y_{\max} + y_i,$
- $z_{\max} \rightarrow z_{\max} + z_i.$

end while

- $x_{\text{avg}} \rightarrow x_{\max}/\text{count}_{\max}, y_{\text{avg}} \rightarrow y_{\max}/$
- $\text{count}_{\max}, z_{\text{avg}} \rightarrow z_{\max}/\text{count}_{\max}$

Center of gravity and all points are shifted to origin,
for $i = 1 \rightarrow n$ **do**

- $\sum P_i^{\text{ref}} x_i = x_i - x_{\text{avg}}, y_i = y_i - y_{\text{avg}}, z_i = z_i - z_{\text{avg}}$

end for

Compute 3D rotation for multiple scans.
 $P_i^{\text{in}}, \{i = 1 \dots n\}$. (3).

Return $RP_i^{\text{out}}, \{i = 1 \dots n\}$.

Rigid registration for all pairs frame.

3.2. Rigid registration

Rigid transformations of the 3D object can include translations and rotation processes without scaling and shearing. For example, a coordinate (y_1, y_2, y_3) defined in different space is determined for each point (x_1, x_2, x_3) of the data with Affine3D mapping. That is why these operations are usually used as subsets in Affine3D transformations.

In multiple scans for the same object, there is no drawback to the overlap of point clouds. However, gaps between overlaps cause problems. Because the overlapping redundancies are regularized by a series of reduction operations. However, if there is a gap between the overlaps, the correction could not be possible. While a full scan can perform with the Kinect camera with three different scans on the average of 120° , the scans are usually performed in a larger scan number and lower rotate angles.

A set of point cloud data is generated with a large number of voxels in each scan performed with Kinect. High-level interpolation solutions are slow because many voxels are calculated at the convergence of these data clusters. Rigid transformations of point cloud provide faster interpolation. Because only voxels are translated at the specified rates for translations operation [41].

In our algorithm, all point cloud data obtained using Kinect are turned according to the left-hand coordinate system. The weight centers of each scan captured for the same 3D object are calculated before performing the conversion operations. This center of gravity is shifted in the center of the coordinate system $(0,0,0)$ which is the origin. As shown in Fig. 5, all other data are shifted together in the same direction by the amount shifted. The pipeline for aligning 3D point cloud frames is given in Algorithm 1 as pseudo code.

A 3D geometric transformation operation is performed according to the origin for the point cloud data which is shifted to the

origin. A geometric transformation operation is a mapping from a coordinate system to itself. Generally, the transformed geometric models are changed to its Model Coordinate System (MCS). Transformation operations are applied to the raw point sets scanned for a 3D object. As a result of the transformation processes, object particles and their relative distances are preserved. Homogeneous coordinates in 3D give are raised to 4-dimensional position vector. A 3D point, P is represented in homogeneous coordinates by a 4-dimensional vector as (4);

$$P = \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \equiv \begin{bmatrix} \alpha x \\ \alpha y \\ \alpha z \\ \alpha \end{bmatrix} \quad (4)$$

Transformed point set $X^* = f(P, t)$, t is the transformation parameter. In homogeneous coordinates, 3D transformations are represented by 4×4 matrices and a point transformation is performed as (5);

$$\begin{bmatrix} a & b & c & t_x \\ d & e & f & t_y \\ g & h & i & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} a & b & c & t_x \\ d & e & f & t_y \\ g & h & i & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (5)$$

3D rotation of point cloud about the arbitrary axis in space is applied following steps;

- Arbitrary axis coincident with one of the coordinate axes.
- Consider an arbitrary axis passing through a point (x_0, y_0, z_0) with direction cosines (c_x, c_y, c_z) .
- Transforming (x_0, y_0, z_0) to the coordinate centre (origin).
- Rotate appropriately to make the axis coincide with the y -axis.
- Rotate the object about the y -axis.

There are three orthogonal planes in 3D transformations which object to rotate. These are called rotation planes around the axes. The first step of rotating the radians around an axis is formulated in (6);

$$[R_y] = \begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos \phi & 0 & -\sin \phi & 0 \\ 0 & 1 & 0 & 0 \\ \sin \phi & 0 & \cos \phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (6)$$

Clockwise rotation around the y -axis adjusted to the left-hand coordinate system is shown in (7);

$$p' = R_y(\phi)p \quad (7)$$

These matrices are multiplied in the appropriate order and the operations are performed. The order of these operations is important. The ϕ angle is the angle of rotation of the 3D object turned on the turntable during the point cloud scan. The rotation angles ϕ are introduced to the program in terms of π coefficient. Only the first scan of the 3D object does not involve a rotation operation. As seen in Fig. 6, a subsequent 3D rotation is performed at the rate of each turn for the scans. The first scan is stable. The angle values used in the program for the 3D rotation angle for each subsequent scan are $\pi/2$, π , and $3\pi/2$.

3.3. Alignment

Global alignment is one of the methods that can be applied to loop closure problems for rigid scanning [42]. In another brute-force solution, it is the iteration of all scans with the ICP algorithm [43]. Methods of such systems generally refer to large equation solutions. In another solution, with a greedy approach,

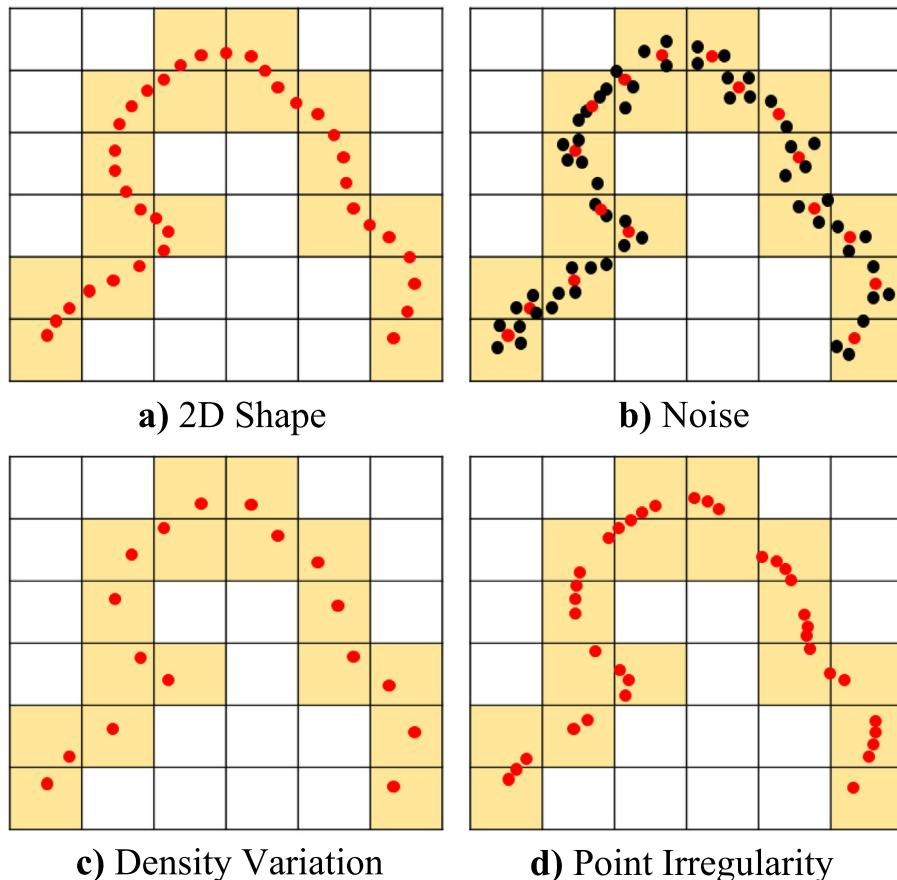


Fig. 4. (a) Demonstrate of LVS robustness in 2D and illustrate, (b) noise, (c) point density variation, and (d) point irregularity.

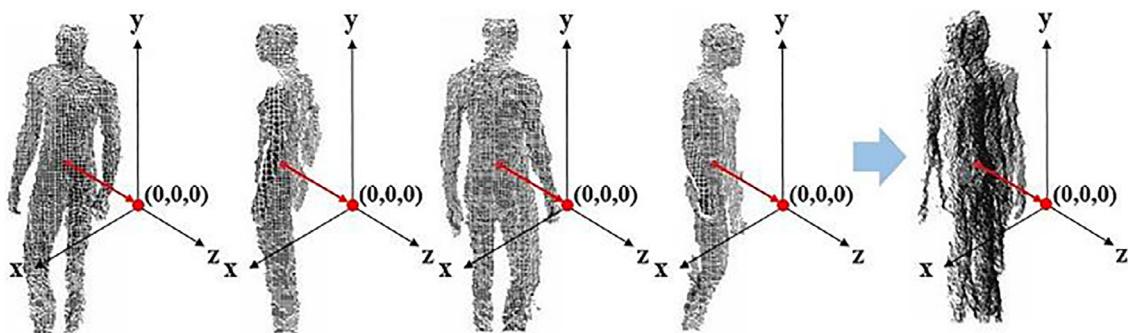


Fig. 5. The centers of gravity of different scans are shifted to the origin.

each new scan is aligned to the previous scan. However, in these methods, errors from previous scans may not be reducible and it is not guaranteed to produce a consistent loop. For this reason, the idea of a pairwise alignment emerges between the scans.

First, transformation operations are performed for the frames to rigid align in geometry level. Then, the alignment key points are determined for each consecutive two frames. These key points are determined by the user from the contrary features of the object, such as a corner, edge, endpoint and so on [44]. These points are selected from points that are thought to be common to both frames. Depth information is processed directly for specified points. Alignment is performed so that the key points of the frame that have implemented the transformation process are matched with the key points of the previous frame. Aligning key

points are merged in such a way that they are in a common position in the first frame. The entire point cloud data in the transformed frame is moved into the first frame, shifted by the same amount of deviation, along with the key points. Thus, a point cloud is obtained with the alignment process for the first two frames. The same alignment process is repeated for each subsequent frame. As shown in Fig. 7, multiple scanning operations are merged with the alignment operation implemented after the 3D transformation. The red dots selected in the previous scan for alignment are merged by aligning them with the blue dots selected in the next scan respectively. The data overflow in the intersections area is no problem for the alignment process. This is because the data is regularized by the refinement process after the alignment process. This method is practical for multi-scanning object modeling and produces effective results within a short time.

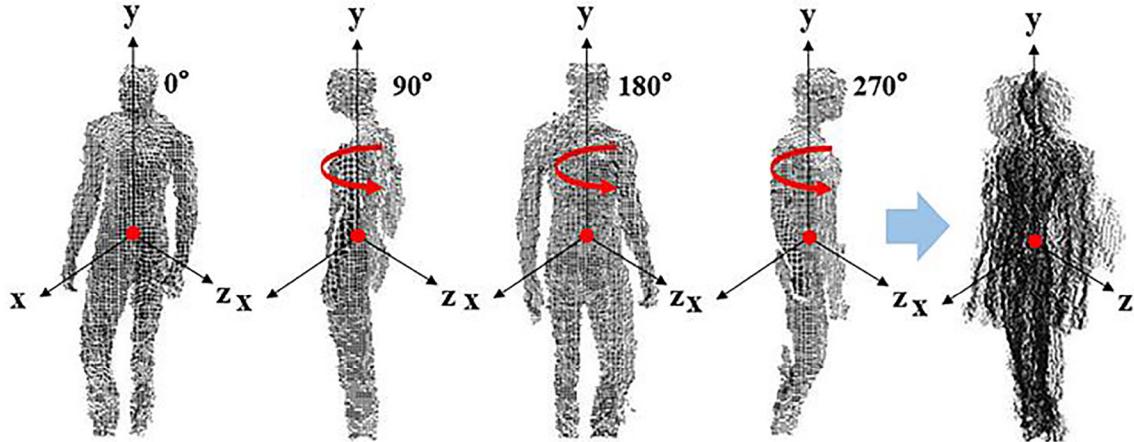


Fig. 6. 3D geometric transformation process on the origin.

There are three orthogonal planes in 3D transformations to rotate the object. These are called rotation planes around *any axes*. The first step of rotating the radians around an axis is formulated in (6);

Algorithm 2.

```

Input: Get, $P_i^{\text{ref}}, \{i = 1 \dots n\}$ .
Output: Improved registered, $P_i^{\text{out}}, \{i = 1 \dots n\}$ .
Sections calculation,
Sagittal ( $v_{\max}, v_{\min}, x_{\max}, x_{\min}$ )/(x-dimension)
while  $i \leq n$  do
  if  $x_i > v_{\max}$  then
     $v_{\max} \rightarrow x_i$ .
  if  $x_i < v_{\min}$  then
     $v_{\min} \rightarrow x_i$ .
end while
Transverse ( $v_{\max}, v_{\min}, y_{\max}, y_{\min}$ )/(y-dimension)
while  $i \leq n$  do
  if  $y_i > v_{\max}$  then
     $v_{\max} \rightarrow y_i$ .
  if  $y_i < v_{\min}$  then
     $v_{\min} \rightarrow y_i$ .
end while
Frontal ( $v_{\max}, v_{\min}, z_{\max}, z_{\min}$ )/(z-dimension)
while  $i \leq n$  do
  if  $z_i > v_{\max}$  then
     $v_{\max} \rightarrow z_i$ .
  if  $z_i < v_{\min}$  then
     $v_{\min} \rightarrow z_i$ .
end while
All points are improved  $P_j^{\text{sec}}, \{j = 1 \dots m\}$ .
initiate  $x_t, y_t, z_t$ 
for  $j = v_{\min} \rightarrow v_{\max}, +1$  do
  while  $j \leq m$  do
    if  $y_t \geq v_{\min}$  and  $y_t \leq v_{\min} + 1$  then
       $y_t \rightarrow y_t + P_{yj}$ .
    if  $z_t \geq v_{\min}$  and  $z_t \leq v_{\min} + 1$  then
       $z_t \rightarrow z_t + P_{zj}$ .
  end while
   $x_{\text{avg}} \rightarrow (\nu_{\min} + 1)/2, y_{\text{avg}} \rightarrow y_t/m, z_{\text{avg}} \rightarrow z_t/m$ .
end for
Refinement algorithm applied to  $P_i^{\text{out}}, \{i = 1 \dots n\}$ .
return  $\{x_{\text{avg}}, y_{\text{avg}}, z_{\text{avg}}\}$ .
```

3.4. Refinement and reduction

After 3D object scans have been successfully registered, a point cloud refinement process is applied. Obviously, the obtained registers are not perfect results. Because even though accurate measurements are taken, the turntable is rotated manually. This causes the point cloud density in the alignment region to be higher than in the other regions. Our aim is to obtain a framework in which points will settle into a voxel. For this purpose, firstly a voxel frame is created which encompasses the entire point cloud that has been registered. Then, the grid is divided into three dimensions sections in such a way that each section includes one unit of voxel area. In Fig. 8, the registered point cloud data is divided into sections according to the left-hand coordinate system. The point clouds are divided into 69 units sections according to x-dimension, 171 units sections towards y-dimension, and 42 units sections according to z-dimension. The total number of registered point cloud for this model is 34181. With the algorithm implemented on the extracted sections, both the number of the point cloud forming the model is reduced and the points have a more regular structure.

The algorithm is applied directly to the point clouds on all sections extracted for each dimension. The section extraction process is divided into three different dimensions, sagittal, frontal, and transverse. The improvement process is implemented separately for these three different dimensions. First, an imaginary voxel is created which will be improved points by moving over each section. This voxel circulates throughout one tour the entire cross section. The algorithm improves at the points it encounters in its each unit voxel movement. The coordinate value for that dimension is assigned as the average value of the cross-section if the improvement is handled for which dimension. Other coordinate values are assigned by calculating the average of each dimension of the coordinate values of all the points in the voxel. For example, for an improvement made to cross section 43 on x-dimension is as in Fig. 8, the cross-sectional average at x coordinate is assigned as 43.5. The y and z values are assigned as averages of y and z values of all points in the voxel. Similarly, in the improvement of section 82 in y-dimension, y value is assigned 82.5, the x and z values are assigned as averages of x and z values of all points in the voxel. The same operations are valid for improvement for z-dimension. The pipeline for the refinement process on registered 3D point cloud is given in Algorithm 2.

Fig. 9 shows the improvement process on each section taken for three different dimensions. Accordingly, the imaginary voxel

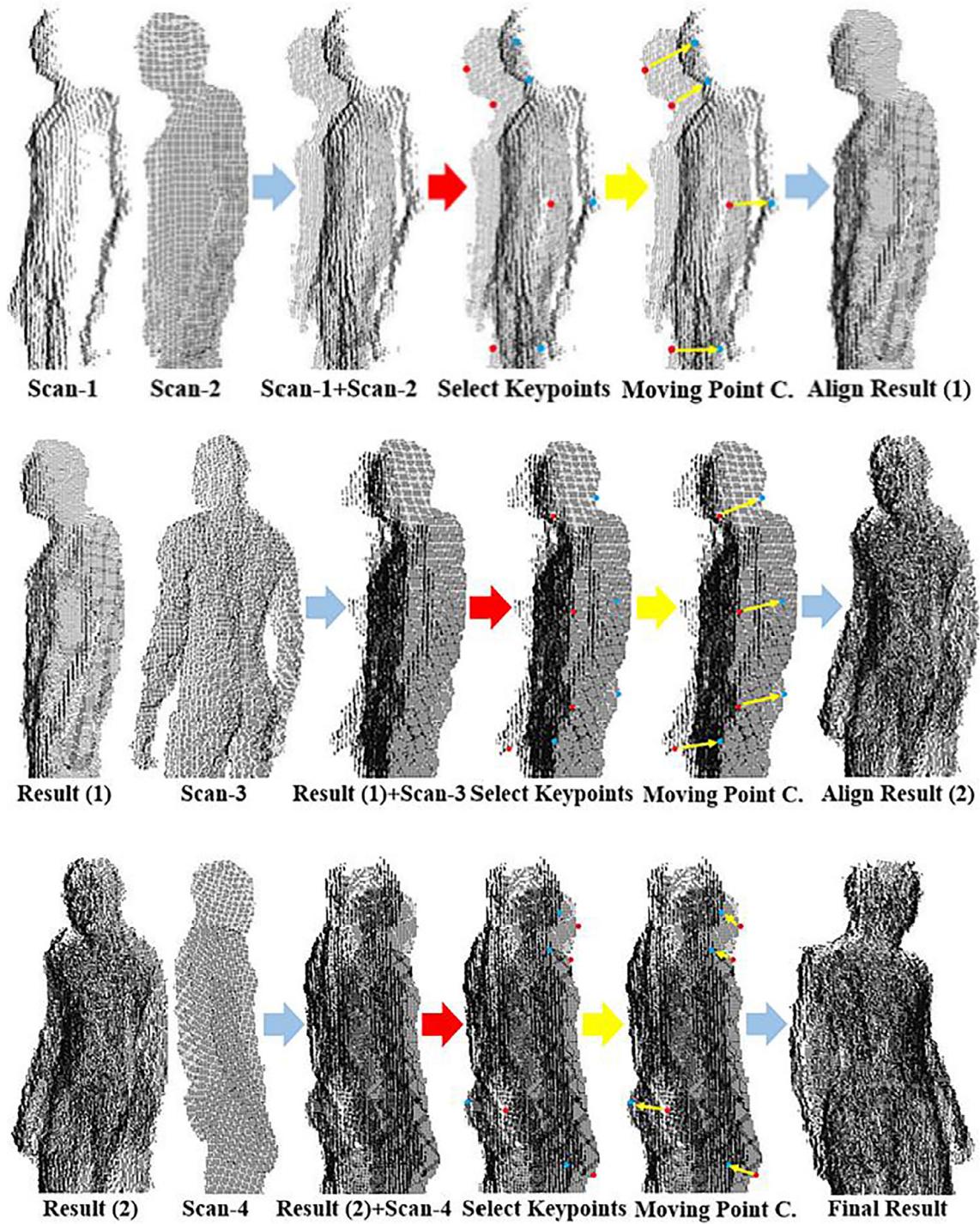


Fig. 7. Alignment pipeline of multiple scans.

passes through all the points and records an average value of the points in the voxel for each section. During the refinement process, the average constant value of each cross-section is assigned to the x coordinate value of the cross-section of the x -dimension. The remaining y and z values are assigned by calculating with the imaginary voxel. Likewise, y coordinate value of the cross-section of the y -dimension is assigned the average constant value of the cross-section. The remaining x and z values are assigned by calculating with the imaginary voxel. This operation is applied to z -dimension. Z coordinate value of the cross-section of the z -dimension is assigned the average constant value of the

cross-section. The remaining x and y values are assigned by calculating with the imaginary voxel.

Fig. 10 shows the improvement algorithm results implemented to the registered point clouds. According to this, firstly, the sections extracted on the transverse have been improved. As a result of the improvement step, the number of raw point clouds has decreased considerably. This number is reduced to 10,892 in the number of data from 34,181 points, and the value is reduced to approximately 1 in 3. The same process is also implemented for sagittal and frontal, which is reduced to 5383 and 2591 respectively. Alignments between the points have been made more consistent and

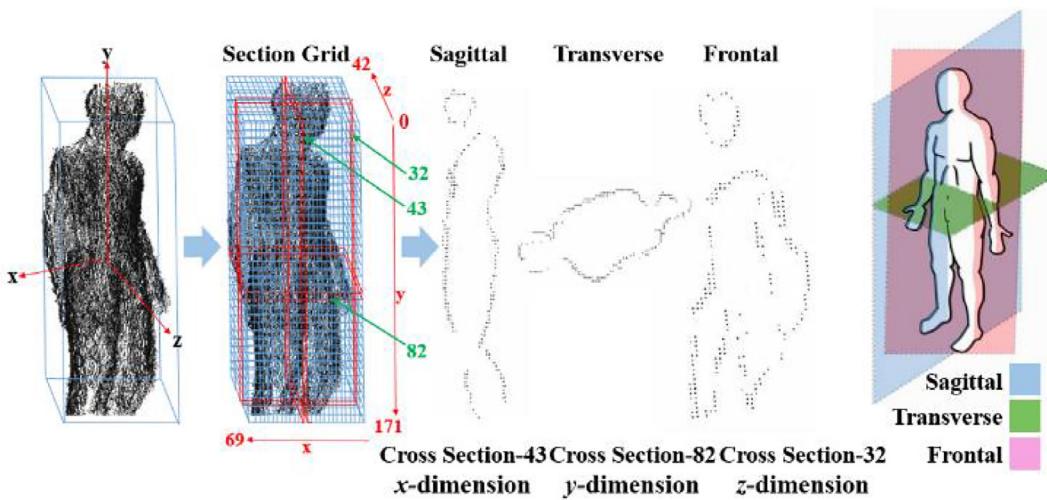


Fig. 8. Section extraction procedure and three-dimensional extracted sections.

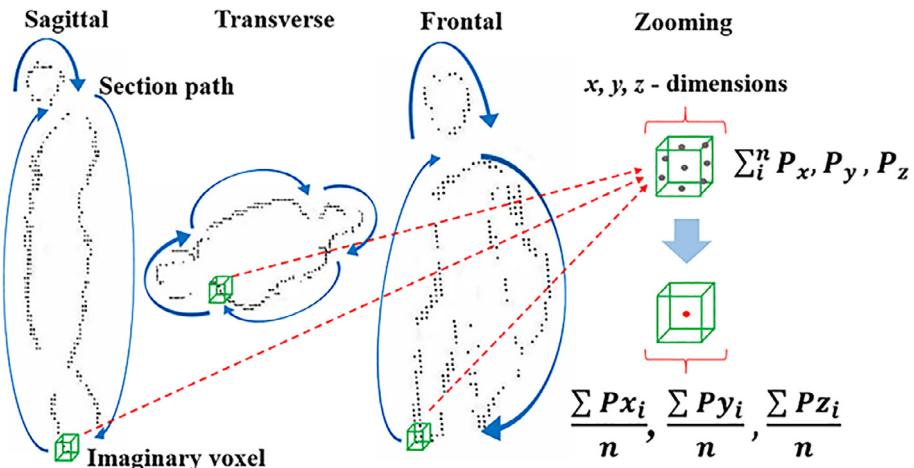


Fig. 9. The procedure of improves sections with the imaginary voxel.

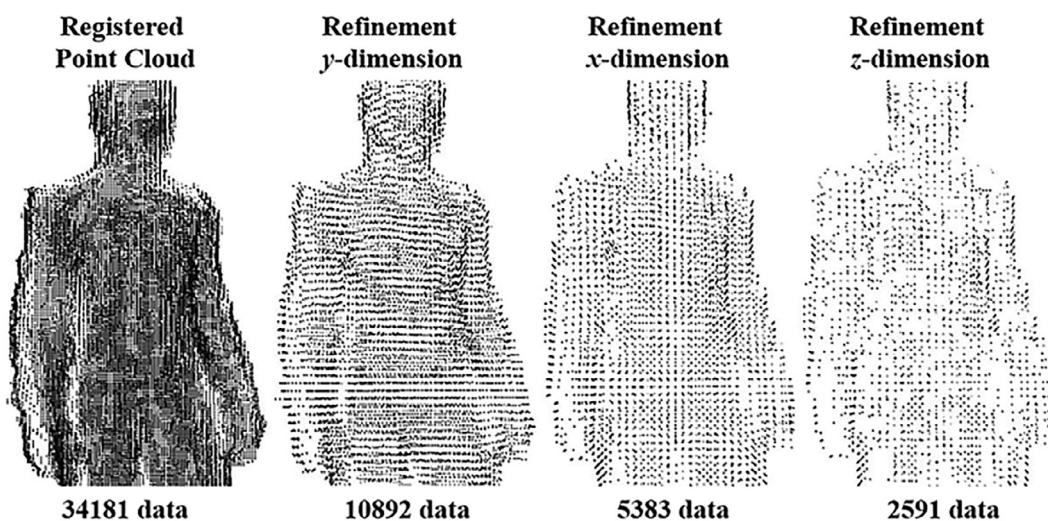


Fig. 10. Registered point cloud and refined point cloud results in different dimensions.

organized, and a significant reduction in the number of points has also been achieved. This eliminates the overlapping occurrences of the alignment result and the point cloud complexity.

4. Results and discussion

In many studies up to now, many criteria have studied in order for surface meshes to be evaluated as general purpose. The criteria used in the known methods are specially derived from the geometric features [45]. Geometric interpretations are studied by defining the theoretical aspects of these features [46]. In order to determine the regularity of the point cloud on the surface mesh, the distance between the points and the average of the angles between the triangular meshes are considered [47]. Fig. 11 shows the triangles angle changes and the average edge between two sections of the registered raw point cloud and improved point cloud.

Accordingly, the changes between the angles of the triangles are shown as a result of the improvement applied to the sections. As a result of these two sections, the smallest angle of the triangle increased from 18° to 22°. In addition, the sum of the average edges lengths of the triangles has risen from 1.42 to 2.36.

Regular results obtained with this scanning system provide a ready environment for use in extraction operations to be performed with a 3D full scan [48].

In this study, all algorithms developed for applications such as data acquisition with Kinect, 3D rotation, alignment, and refinement have been used in the C# programming language. The computer hardware properties that used for programming are 8 GB RAM 2.40 GHz Intel Core i7-3610QM CPU, GeForce 650 M graphic card. In addition, comparative evaluations were made using several real-valued methods. These are ROPS [40], SHOT [49], FPFH [50], Snapshots [51], and Spin image [52].

4.1. Evaluation of point cloud registration

In this section, we present an evaluation of the LVS-based registration algorithm. We also tested our algorithm on the Bologna Mesh Registration (BMR) dataset [40], in addition to the human model obtained using a Microsoft Kinect device. Both qualitative and quantitative results were compared together with the state-of-the-art local voxelize-based registration algorithms.

The dataset used for this evaluation is the BMR benchmark as shown in Fig. 12. The main reason for choosing this data set is the point clouds in BMR were scanned using a Kinect device, which presents a significant challenge for registration because of severe noise and point irregularity. The datasets are created by capturing the partial views of the Duck, Peter Rabbit, Doll, Frog, Mario, and Squirrel models respectively with six different looks. These models are composed of an average of 15–20 scans.

Theoretically, $n(n-1)/2$ pair of junction data is available for the registration of a model with n views, but not each pair of views contain overlap regions [53]. So, we have an overlap rate which is larger than 10% in the BMR benchmark over view pairs. Here, the overlap rate is calculated as follows,

$$\text{Overlap} = \frac{\#\text{of corresponding points (in } P_s \text{ and } P_t\text{)}}{\min(\#\text{ of points in } P_s, \#\text{ of points in } P_t)} \quad (8)$$

where P_s and P_t represent the source and target two point cloud views to be matched. We have obtained a total of 117 valid registration pairs for the experiment. In order to measure the accuracy of a recording algorithm, we used the recording accuracy [54] for quantitative evaluation.

$$\text{Reg.accuracy} = \frac{\#\text{ of correct registrations}}{\#\text{ of total registrations}} \quad (9)$$

In addition, the root mean squared error (RMSE) has been examined to judge whether a registration is true [55].

$$\epsilon_{rmse} = \sqrt{\frac{\sum_{i=1}^{N_c} \|R_{GT}.c_s^i + t_{GT}.c_t^i\|^2}{N_c}} \quad (10)$$

where c_s^i and c_t^i are two points to correspond each other and N_c is the number of correspondence. Just if $\epsilon_{rmse} < \tau_{rmse}$, the condition is considered correct. Thus, the judging threshold τ_{rmse} can be changed according to the desired recording accuracy [56].

We emphasize that the proposed algorithm is fully automated and that the aligned shape is produced as a result of the rigid alignment and transformation between the two separate point clouds taken as input during this process. In addition, the threshold τ_{rmse} value judgment is very important to determine the overall accuracy of a registration algorithm. In order to meet the different concerns regarding registration sensitivity, we set τ_{rmse} in our evaluation to 1 mr and 3 mr respectively. All the results of the evaluations are presented in Table 1.

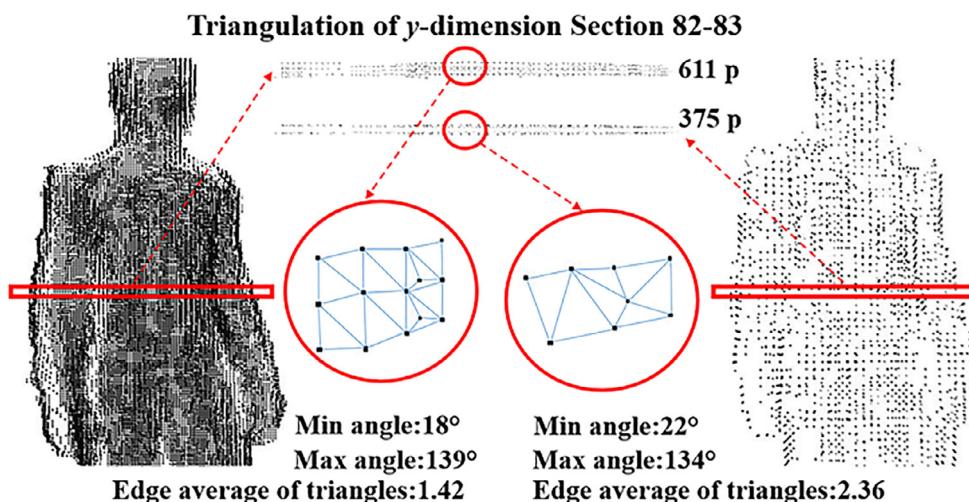


Fig. 11. Registered and refined point cloud surface mesh features.

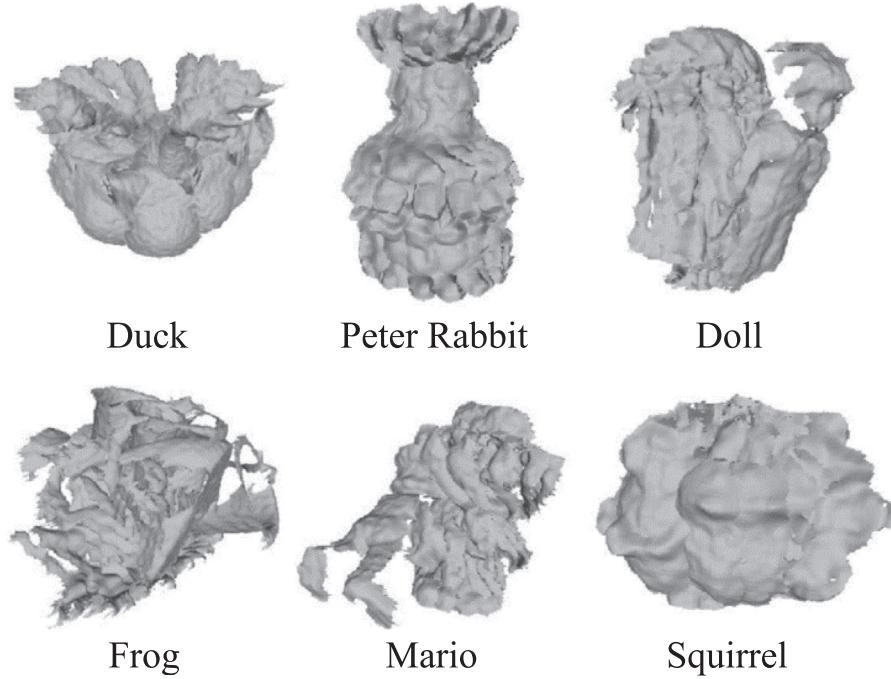


Fig. 12. Registered point cloud and refined point cloud results in different dimensions.

As shown in [Table 1](#), the proposed LVS-based registration algorithm provides the highest registration accuracy for all data-sets at all threshold (rmse) levels. The margin is especially large for low levels of threshold. For example, the average accuracy of our algorithm is 29.9% higher than that of the second order FPFH based algorithm when rmse = 3mr. It is also 61.3% higher than FPFH based algorithm when rmse = 1mr. In general, FPFH, SHOT, and RoPS-based algorithms implement better results than others.

The runtime of our registration method for samples in the BMR benchmark is shown in [Table 2](#). Codes that are executed in the program have been implemented in Microsoft Visual C# 2010 and we would like to note that parallel/GPU calculation techniques are not used during this process. In summary, with the developed algorithm, a few point clouds scanned with Kinect can be aligned within a few seconds. The complexity of the algorithm is $O(n^2 \log n)$ as given n correspondences. The

Table 2
Registration runtime statistics (Tsum is entire registration process).

Data (view ₁ - view _{last})	Data size	T _{sum} (s)
Duck	18,215	13.105
PeterRabbit	11,606	6.361
Doll	10,652	5.587
Frog	22,996	19.576
Mario	12,918	7.533
Squirrel	9330	4.418

quantity of point in the point cloud within all the data sets used is in the range of about 8000–35000. In order to demonstrate the effectiveness of the proposed registration algorithm, the results of 30 records are presented in [Fig. 13](#). This means that our algorithm successfully aligns them with a unified coordinate system.

Table 1

Comparison of our LVS based registration algorithm and the five different state-of-the-art algorithms. The optimum results are shown in bold.

τ_{rmse} (mr)	Datasets	Registration accuracy (%)					
		RoPS	SHOT	Fpfh	S.shots	Spin I.	LVS
1.0	Duck	11.2	3.7	10.2	2.8	0.0	40.1
	PeterRabbit	8.8	11.2	13.6	9.5	4.8	24.9
	Doll	11.8	16.7	14.7	4.9	8.8	42.9
	Frog	5.6	4.5	8.5	0.5	1.1	32.3
	Mario	11.4	9.9	17.0	11.4	8.5	19.3
	Squirrel	10.5	9.5	11.3	4.3	9.5	36.3
	Average	9.9	9.3	12.6	5.6	5.5	32.6
3.0	Duck	16.0	5.33	14.6	4.0	0.1	49.6
	PeterRabbit	12.6	16.0	19.5	13.7	6.9	37.5
	Doll	16.9	23.9	21.1	7.0	12.6	44.3
	Frog	8.1	6.5	12.2	0.8	1.6	42.1
	Mario	16.3	14.2	24.4	16.3	12.2	29.6
	Squirrel	15.0	13.7	16.2	6.2	13.7	37.5
	Average	14.1	13.2	18.0	8.0	7.8	40.1

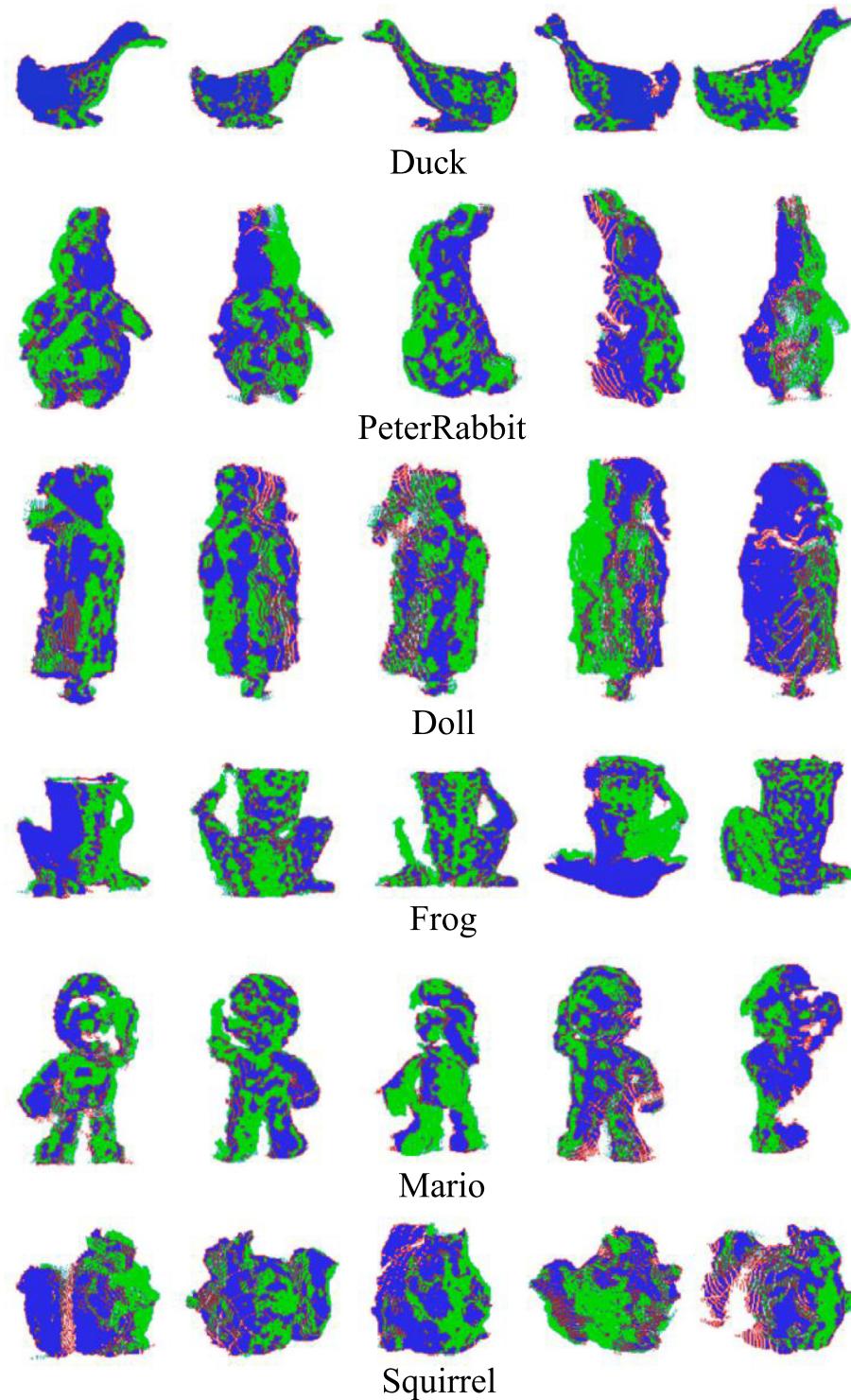


Fig. 13. Registration results of the LVS-based algorithm from the BMR benchmark.

5. Conclusion

This paper presented a 3D local voxelize structure, LVS-based registration algorithm for point cloud registration. The principal idea of LVS is that the 3D local structure is approximated using a set of voxels. We have designed a practical system that can be managed by naive users for 3D object scanning using a single fixed Kinect camera at indoor environments. An evaluation has been realized for the methods used in 3D object scanning. The presented method works on directly scanned point clouds. Moreover, the

developed method is suitable for 3D rotating in any direction. The implemented algorithm has tested via large-scale benchmark consist of Kinect-scanned point cloud views. By means of the implemented system, the complex algorithms used in the existing methods have been avoided due to the fact that high-cost systems. The results in the scan are evaluated according to the point-to-point angle changes after the refinement. The average angle changes between the registration point cloud and the point clouds which applied improvement give us an idea of regularity such as angle and edge changes. It is thought that natural results will be

obtained in future skeletal extraction algorithms over improved full 3D data obtained by this study.

Acknowledgment

This work is supported by Research Fund of the Firat University, Project Number: MF.16.62. The authors thank Scientific Research Project Support Center (FUBAP) of Firat University for financial support of data collection and analysis.

References

- [1] B. Allen, B. Curless, Z. Popovic, The space of human body shapes: reconstruction and parameterization from range scans, *ACM Trans. Graph. (TOG)* 22 (3) (2003) 587–594.
- [2] A. Guarneri, F. Pirotti, M. Pontin, A. Vettore, 3D Surveying for structural analysis applications, in: Proc. of; 2006, p. 22–4.
- [3] M. Aerntsen, B. Bellekens, M. Weyn, MapFuse: complete and realistic 3D modelling, *J. Robot.* (2018).
- [4] E. De Aguiar, C. Stoll, C. Theobalt, N. Ahmed, H.-P. Seidel, S. Thrun, Performance capture from sparse multi-view video, *ACM Trans. Graph. (TOG)* 27 (3) (2008) 98.
- [5] A. Tabb, Shape from silhouette probability maps: reconstruction of thin objects in the presence of silhouette extraction and calibration error, in: Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on, 2013, pp. 161–168.
- [6] A. Laurentini, The visual hull concept for silhouette-based image understanding, *IEEE Trans. Pattern Anal. Mach. Intell.* 16 (2) (1994) 150–162.
- [7] Y.M. Kim, C. Theobalt, J. Diebel, J. Kosecka, B. Masicusik, S. Thrun, Multi-view image and tof sensor fusion for dense 3d reconstruction, in: Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on, 2009, pp. 1542–1549.
- [8] P. Henry, M. Krainin, E. Herbst, X. Ren, D. Fox, RGB-D mapping: using kinect-style depth cameras for dense 3D modeling of indoor environments, *Int. J. Robot. Res.* 31 (5) (2012) 647–663.
- [9] K. Khoshelham, S.O. Elberink, Accuracy and resolution of Kinect depth data for indoor mapping applications, *Sensors* 12 (2) (2012) 1437–1454.
- [10] M. Kiechle, S. Hawe, M. Kleinsteuber, A joint intensity and depth co-sparse analysis model for depth map super-resolution, in: Computer Vision (ICCV), 2013 IEEE International Conference on, 2013, pp. 1545–1552.
- [11] Y. Cui, D. Stricker, 3D body scanning with one kinect, in: 2nd International Conference on 3D Body Scanning Technologies, 2011.
- [12] S. Izadi, D. Kim, O. Hilliges, D. Molnyneux, R. Newcombe, P. Kohli, J. Shotton, S. Hodges, D. Freeman, A. Davison, KinectFusion: real-time 3D reconstruction and interaction using a moving depth camera, in: Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology, 2011, pp. 559–568.
- [13] M. Zollhöfer, M. Martinek, G. Greiner, M. Stamminger, J. Süßmuth, Automatic reconstruction of personalized avatars from 3D face scans, *Comput. Anim. Virtual Worlds* 22 (2) (2011) 195–202.
- [14] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, J. Davis, SCAPE: shape completion and animation of people, *ACM Trans. Graph. (TOG)* (2005) 408–416.
- [15] Y. Chen, Z. Liu, Z. Zhang, Tensor-based human body modeling, in: Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on, 2013, pp. 105–112.
- [16] N. Senin, B. Colosimo, M. Pacella, Point set augmentation through fitting for enhanced ICP registration of point clouds in multisensor coordinate metrology, *Rob. Comput. Integr. Manuf.* 29 (1) (2013) 39–52.
- [17] Y. Cui, S. Schuon, S. Thrun, D. Stricker, C. Theobalt, Algorithms for 3d shape scanning with a depth camera, *IEEE Trans. Pattern Anal. Mach. Intell.* 35 (5) (2013) 1039–1050.
- [18] Q.-X. Huang, S. Flöry, N. Gelfand, M. Hofer, H. Pottmann, Reassembling fractured objects by geometric matching, *ACM Trans. Graph. (TOG)* 25 (3) (2006) 569–578.
- [19] B. Curless, M. Levoy, A volumetric method for building complex models from range images, in: Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques, 1996, pp. 303–312.
- [20] S. Rusinkiewicz, O. Hall-Holt, M. Levoy, Real-time 3D model acquisition, *ACM Trans. Graph. (TOG)* 21 (3) (2002) 438–446.
- [21] J.-Y. Bouguet, P. Perona, 3D photography on your desk, in: Computer Vision, 1998. Sixth International Conference on, 1998, pp. 43–50.
- [22] D. Scharstein, R. Szeliski, A taxonomy and evaluation of dense two-frame stereo correspondence algorithms, *Int. J. Comput. Vision* 47 (1) (2002) 7–42.
- [23] K.-Y. Wong, R. Cipolla, Structure and motion from silhouettes, in: Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on, 2001, pp. 217–222.
- [24] K.-Y.K. Wong, P.R. Mendonça, R. Cipolla, Head model acquisition from silhouettes, in: Visual Form 2001, Ed., 2001, pp. 787–796.
- [25] S. Izadi, R.A. Newcombe, D. Kim, O. Hilliges, D. Molnyneux, S. Hodges, P. Kohli, J. Shotton, A.J. Davison, A. Fitzgibbon, Kinectfusion: real-time dynamic 3d surface reconstruction and interaction, in: ACM SIGGRAPH 2011 Talks, 2011, pp. 23.
- [26] D. Jeong, Y. Li, H.J. Lee, S.M. Lee, J. Yang, S. Park, J. Kim, Efficient 3d volume reconstruction from a point cloud using a phase-field method, *Math. Probl. Eng.* (2018).
- [27] K. Kolev, M. Klodt, T. Brox, D. Cremers, Continuous global optimization in multiview 3d reconstruction, *Int. J. Comput. Vision* 84 (1) (2009) 80–96.
- [28] R.A. Newcombe, A.J. Davison, Live dense reconstruction with a single moving camera, in: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, 2010, pp. 1498–1505.
- [29] R.A. Newcombe, S.J. Lovegrove, A.J. Davison, DTAM: Dense tracking and mapping in real-time, in: Computer Vision (ICCV), 2011 IEEE International Conference on, 2011, pp. 2320–2327.
- [30] K.N. Kutulakos, S.M. Seitz, A theory of shape by space carving, *Int. J. Comput. Vision* 38 (3) (2000) 199–218.
- [31] A. Maimone, H. Fuchs, Reducing interference between multiple structured light depth sensors using motion, in: Virtual Reality Short Papers and Posters (VRW), IEEE, 2012, pp. 51–54.
- [32] K. German, 3D reconstruction algorithm combining shape-from-silhouette with stereo, *IEEE Trans. PAMI* 11 (2003) 1265–1278.
- [33] Y. Chen, G. Medioni, Object modelling by registration of multiple range images, *Image Vis. Comput.* 10 (3) (1992) 145–155.
- [34] J. Tong, J. Zhou, L. Liu, Z. Pan, H. Yan, Scanning 3d full human bodies using kinects, *IEEE Trans. Visual Comput. Graphics* 18 (4) (2012) 643–650.
- [35] A. Kolb, E. Barth, R. Koch, R. Larsen, Time-of-flight sensors in computer graphics, in: Eurographics (STARs), 2009, pp. 119–134.
- [36] Y. Cui, S. Schuon, D. Chan, S. Thrun, C. Theobalt, 3D shape scanning with a time-of-flight camera, in: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, 2010, pp. 1173–1180.
- [37] Y. Cui, D. Stricker, 3d shape scanning with a kinect, in: ACM SIGGRAPH Posters, 2011, pp. 57.
- [38] L.-C. Chen, D.-C. Hoang, H.-I. Lin, T.-H. Nguyen, Innovative methodology for multi-view point cloud registration in robotic 3D object scanning and reconstruction, *Appl. Sci.* 6 (5) (2016) 132.
- [39] E. Özbay, A. Çınar, 3D reconstruction technique with kinect and point cloud computing, *Global J. Technol.* 3 (2013) 1748–1754.
- [40] Y. Guo, F. Sohel, M. Bennamoun, M. Lu, J. Wan, Rotational projection statistics for 3D local surface description and object recognition, *Int. J. Comput. Vision* 105 (1) (2013) 63–86.
- [41] J. Ashburner, K.J. Friston, Rigid body registration, Statistical parametric mapping: The analysis of functional brain images, 2007, pp. 49–62.
- [42] E. Battle, C. Matabosch, J. Salvi, Overview of 3d registration techniques including loop minimization for the complete acquisition of large manufactured parts and complex environments, in: Eighth International Conference on Quality Control by Artificial Vision, International Society for Optics and Photonics, 2007, p. 635605.
- [43] A. Makadia, A. Patterson, K. Daniilidis, Fully automatic registration of 3D point clouds, in: Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on, 2006, pp. 1297–1304.
- [44] Y. An, Z. Li, C. Shao, Feature extraction from 3D point cloud data based on discrete curves, *Math. Probl. Eng.* (2013).
- [45] P.J. Frey, H. Borouchaki, Geometric evaluation of finite element surface meshes, *Finite Elem. Anal. Des.* 31 (1) (1998) 33–53.
- [46] E. Özbay, A. Çınar, A metric based novel classification approach to kinect point cloud data, *Technol. Appl. Sci.* 11 (4) (2016) 153–170.
- [47] E. Özbay, A. Çınar, A novel approach to smoothing on 3d structured adaptive mesh of the kinect-based models, in: Proc. 2th International Conference on Advanced Information Technologies and Applications, 2013, pp. 13–22.
- [48] E. Özbay, A. Çınar, Z. Güler, A hybrid method for skeleton extraction on Kinect sensor data: combination of L_1 -Median and Laplacian shrinking algorithms, *Measurement* 125 (2018) 535–544.
- [49] F. Tombari, S. Salti, L. Di Stefano, Unique signatures of histograms for local surface description, in: European conference on computer vision, 2010, pp. 356–369.
- [50] R.B. Rusu, N. Blodow, M. Beetz, Fast point feature histograms (FPFH) for 3D registration, in: Robotics and Automation, ICRA'09. IEEE International Conference on IEEE, 2009, pp. 3212–3217.
- [51] S. Malassiotis, M.G. Strintzis, Snapshots: A novel local surface descriptor and matching algorithm for robust 3D surface alignment, *IEEE Trans. Pattern Anal. Mach. Intell.* 29 (7) (2007) 1285–1290.
- [52] A.E. Johnson, M. Hebert, Using spin images for efficient object recognition in cluttered 3D scenes, *IEEE Trans. Pattern Anal. Mach. Intell.* 5 (1999) 433–449.
- [53] A.S. Mian, M. Bennamoun, R.A. Owens, A novel representation and feature matching algorithm for automatic pairwise registration of range images, *Int. J. Comput. Vision* 66 (1) (2006) 19–40.
- [54] Y. Guo, M. Bennamoun, F. Sohel, M. Lu, J. Wan, 3D object recognition in cluttered scenes with local surface features: a survey, *IEEE Trans. Pattern Anal. Mach. Intell.* 36 (11) (2014) 2270–2287.
- [55] S. Rusinkiewicz, M. Levoy, Efficient variants of the ICP algorithm, in: 3-D Digital Imaging and Modeling, IEEE Proceedings. Third International Conference on, 2001, pp. 145–152.
- [56] S. Quan, J. Ma, F. Hu, B. Fang, T. Ma, Local voxelized structure for 3D binary feature representation and robust registration of point clouds from low-cost sensors, *Inf. Sci.* 444 (2018) 153–171.



Erdal Özbay is a Research Assistant in the Department of Computer Engineering at the University of Firat where he has been a faculty member since 2012. He has completed his MSc and Ph.D. at Firat University. His research interests lie in the area of programming languages, computer graphics, image processing, human-computer interaction, mesh generation, ranging from theory to design to implementation, with a focus on improving software quality.



Ahmet Çınar He received the Ph.D. degree in Electric-Electronics Engineering in 2003 from Firat University. He has been working on Firat Univ. Department of Computer Engineering, (Assistant Professor). His research is interested in development and improvement of mesh generation methods, and applications of virtual reality, augmented reality, artificial intelligence, and game programming.