

# Point cloud augmented virtual reality environment with haptic constraints for teleoperation

**Dejing Ni<sup>1</sup>, AYC Nee<sup>2</sup>, SK Ong<sup>2</sup>, Huijun Li<sup>1</sup>, Chengcheng Zhu<sup>1</sup>  
and Aiguo Song<sup>1</sup>**

Transactions of the Institute of  
Measurement and Control  
2018, Vol. 40(15) 4091–4104  
© The Author(s) 2018  
Article reuse guidelines:  
[sagepub.com/journals-permissions](http://sagepub.com/journals-permissions)  
DOI: [10.1177/0142331217739953](https://doi.org/10.1177/0142331217739953)  
[journals.sagepub.com/home/tim](http://journals.sagepub.com/home/tim)



## Abstract

Remote manipulation of a robot without assistance in an unstructured environment is a challenging task for operators. In this paper, a novel methodology for haptic constraints in a point cloud augmented virtual reality environment is proposed to address this human operation limitation. The proposed method generates haptic constraints in real time for an unstructured environment, including regional constraints and guidance constraints. A modified implicit surface method is applied for regional constraint generation for the entire point cloud. Additionally, the isosurface derived from the implicit surface is proposed for real-time three-dimensional artificial force field estimation. For guidance constraint generation, a new incremental prediction and local artificial force field generation method based on the modified sigmoid model is proposed in an unstructured point cloud virtual reality environment. With the generated haptic constraints, the operator can control the robot to realize obstacle avoidance and easily reach the target tasks. System evaluation is conducted, and the result demonstrates the effectiveness of the proposed method. In addition, a 10-participant study with users who control the robot to three specific targets shows that the system can enhance human operation efficiency and reduce time costs by at least 59% compared with no-haptic-constraint operations. Additionally, the designed questionnaire also demonstrates that the proposed methodology can reduce the workload during human operations.

## Keywords

Haptic constraints, local artificial force field, teleoperation, point cloud, virtual reality

## Introduction

Teleoperation application has been a topic of research for decades. Teleoperation has been proposed mainly to assist humans in accomplishing unreachable and risky tasks at a remote distance and is widely deployed in outer space, deep ocean exploration, and telesurgical operations, among other applications. Owing to the poor robustness and non-steady performance of the state-of-the-art autonomous robot during critical tasks in an unforeseen unstructured environment (Bohren et al., 2011), the human-in-the-loop strategy is a more feasible and safe solution for teleoperations.

In recent years, virtual reality (VR)-based teleoperation systems have been widely applied to address the long time delay and poor virtual transparency problems that are typical challenges in teleoperation research (Passenberg et al., 2010; Sun et al., 2014). In a VR-based teleoperation, a human controls the virtual robot motion using human-computer-interaction (HCI) devices, which are mapped to the real robot system. During these operations, trauma, fatigue and stress are common human limitations that could result in poor accuracy, unintended collisions and an increased human workload (Rosenberg, 1993). Therefore, it is essential to realize human safety regulation with guidance assistance and hazard avoidance in the VR environment during teleoperation. Virtual fixtures have been introduced to assist operators.

The prime theory of the virtual fixture (Rosenberg, 1993) is an overlay of abstract sensory information on a workspace to regulate human operation in manipulated tasks. Currently, virtual fixtures based on haptics are much more widely applied to greatly reinforce or improve user understanding in teleoperation (Nam et al., 2014).

Haptic constraints are categorized into regional constraints and guidance constraints. It is clear that regional constraints are safety-oriented features that prevent the user from entering forbidden areas of the workspace, and guidance constraints are task-oriented regulations that assist the user in controlling the robot along the desired path or towards the target in the workspace.

Most early research focused on the construction of haptic constraints based on structured primitives. Rosenberg initially implemented several types of haptic constraints using specific

<sup>1</sup>School of Instrument Science and Engineering, Southeast University, Nanjing, China

<sup>2</sup>Department of Mechanical Engineering, National University of Singapore, Singapore

## Corresponding author:

Aiguo Song, School of Instrument Science and Engineering, Southeast University, Sipailou 2#, Nanjing 210096, China.  
Email: a.g.song@seu.edu.cn

surfaces to assist with peg-in-hole tasks and increased operator performance by up to 70% (Rosenberg, 1993). Later, other researchers applied haptic constraints in different applications. Regional constraints for assembly tasks were implemented by Peshkin et al. (2001) by building a virtual surface to assist the operator in completing assembly tasks with less effort. A basis of preferred directions (Marayong et al., 2003) was used to generate guidance constraints to guide a user to a curve, surface and orientation in the VR environment and improve the accuracy and speed of the user in manipulating a cooperative or tele-robotic system. A type of haptic constraints (Prada and Payandeh, 2009) was implemented towards point positioning and curve following as well as extensions of these efforts to tubes, cones, etc. A predefined path and permitted region were set for haptic constraint construction in these systems, and certain similar research studies were presented (Abbott and Okamura, 2006; Li et al., 2007; Maddahi et al., 2015).

Early research was mostly aimed towards a pre-set environment, which was not widely used in other systems. Recently, haptic constraints based on computer vision have been proposed and can be applied to an unknown environment. Yamamoto et al. (2012) implemented regional constraints for tissue protection based on surface reconstruction (Yamamoto et al., 2012). A guidance constraint using augmented reality was implemented for a blanket-tearing task on the master side (Xia et al., 2012). Raiola et al. (2015) realized a guidance constraint to control the motion of a robot to task-relevant trajectories. Other haptic constraint construction methods can be found in a recent survey (Bowyer, 2014).

In the teleoperation area, most environments are traditionally unstructured. Thus, it is considerably more difficult to acquire a precise geometric environment model for accurate haptic constraint definition and construction. Recently, point clouds with Red Green Blue(RGB) and depth information have been used to represent the environment easily, and certain regional constraint generation methods for dynamic environment have been implemented (Nia Kosari et al., 2014; Xia et al., 2012).

Although selected studies have been performed, no guidance constraints are suitable for arbitrary target acquisition. At the same time, little automatic generated assistance is available with both guidance constraints and regional constraints for the operator during manipulation in an unstructured environment. In our previous work (Ni et al., 2016), an initial concept of a real-time haptic constraints construction method was proposed. However, the methods are rough and not complete for a whole system.

Considering the current research background, a novel real-time haptic constraint construction approach is proposed with both regional constraints and guidance constraints based on unstructured point cloud data in a VR environment. This method can avoid the hard collisions caused by mis-operation and supply the operator with real-time guidance constraints in an unstructured environment. In this manner, an operator without any training could implement the task safely and efficiently.

The main contributions of this paper are listed as follows:

- (1) Implementation of a methodology for construction of a point cloud virtual environment with both regional haptic constraints and dynamic guidance constraints.

- (2) Proposal of a novel guidance constraint construction method from the point cloud based on the local artificial force field estimation (LAFF) method.

The remains of the paper are organized as follows. Section 2 introduces the overall design of the point cloud augmented virtual reality (PCAVR) teleoperation system. Section 3 describes the haptic constraint generation method, which includes the regional constraint method and guidance constraint construction approach in detail. Section 4 presents the experiments and results. Finally, conclusions are drawn in Section 5.

## System architecture

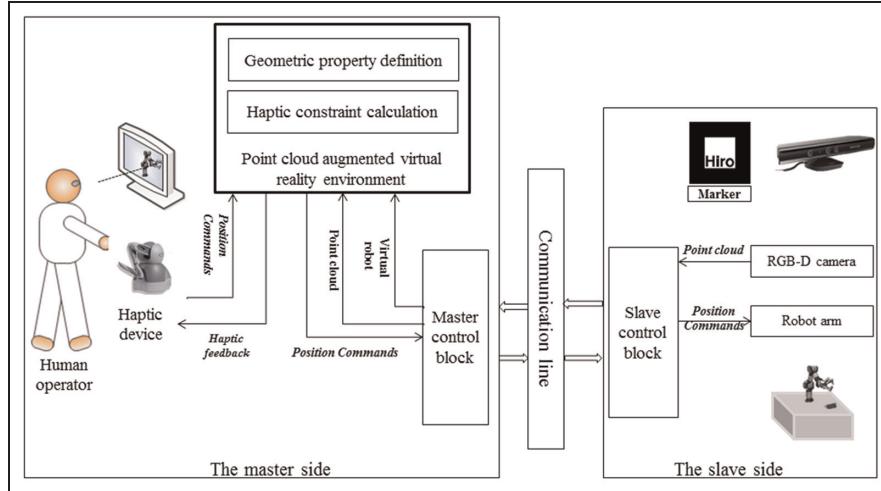
Figure 1 describes the framework of the PCAVR teleoperation system. The VR environment is composed of a virtual robot model, the point cloud from the slave side, and real-time defined haptic constraints. The operator controls the virtual robot using the haptic device, which uses six degree-of-freedom positional sensing as a motion input device. A PHANTOM haptic device is adopted in this system. During the virtual robot interaction with the transmitted point cloud, the haptic constraints are established in real-time and fed back to the operator for motion regulation in the following step. The commands are simultaneously forwarded to the real robot for real-time control. At the slave side, the point cloud is acquired using the RGB-Depth (RGB-D) camera. To reduce the bandwidth of transmission, the point cloud pre-filter process is applied prior to transmission. Because the coordinates among the transmitted point cloud, the virtual robot, and the virtual device are not uniform, coordinate registration is implemented for PCAVR environment construction. In this paper, the authors focus on haptic constraint construction in the PCAVR environment and consider the motion of the robot end effector (REE).

During teleoperation, it is commonly required that the robot must avoid obstacles along the path to achieve the target position. Traditionally, all objects in the real environment are set as rigid objects, which means that even a slight collision between the robot and the objects can be quite dangerous. Therefore, two aspects are considered in the haptic constraint definition. First, the regional constraints are constructed for all point cloud data because all points represent real environments. Second, for manipulation efficiency, the guidance constraints are defined in the VR environment. In this paper, the authors propose LAFF method based guidance constraints for real-time motion orientation indication.

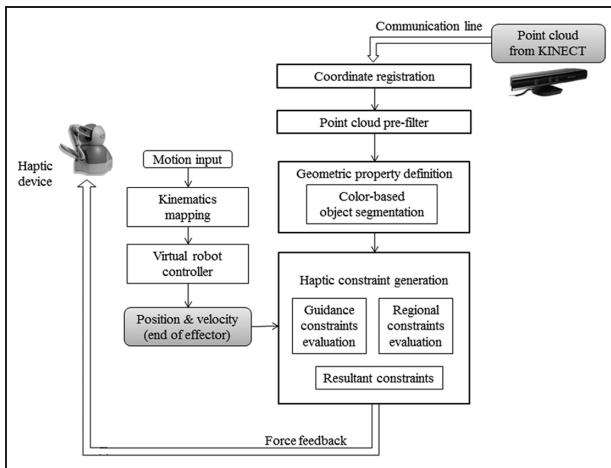
## Haptic constraint generation

The detailed flowchart for construction of the haptic constraints in the PCAVR environment is shown in Figure 2. The procedure primarily consists of the pre-filter module, coordinate registration module, geometric property definition module and haptic constraint generation module.

Two haptic constraints are defined in the PCAVR environment. While the operator moves the virtual robot in the PCAVR environment, the guidance constraints are put into



**Figure 1.** Framework of the PCAVR teleoperation system.



**Figure 2.** Detailed flowchart for generation of haptic constraints.

effect for motion assistance. An LAFF method is proposed for current haptic guidance estimation and is dependent on analysis of the repulsive potential field and attractive potential field in the current limited region. In addition, the regional constraints are evaluated using the relationship between the REE position and the nearby neighbourhood points in the point cloud. The method proposed by Nia Kosari et al. (2014) is similar to the haptic rendering theory. However, the method by Nia Kosari only defines the regional constraints and adopts the haptic rendering method, which is not steadily extensible (Leeper et al., 2012). To realize haptic constraints for the robot manipulation, several challenges must be addressed as follows:

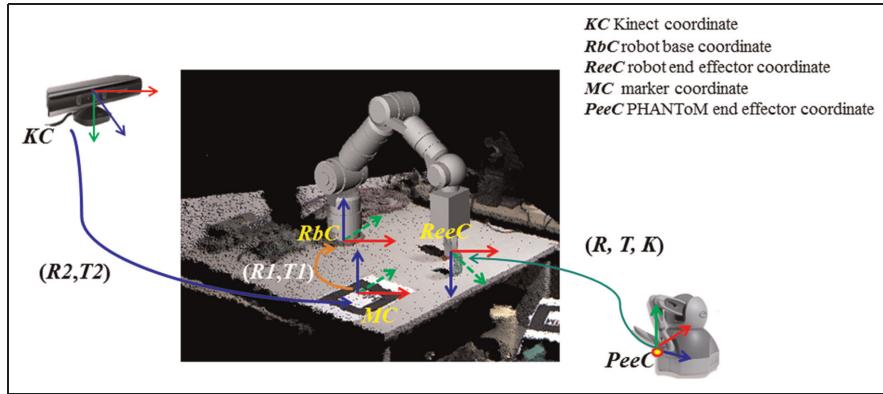
- (a) Coordinate registration. Several coordinates exist in the proposed system in this paper, such as the Kinect coordinate, the robot coordinate, and the haptic device coordinate. Coordinate uniformity is fundamental for accurate manipulation, and thus the

marker coordinate is applied as an intermediate step for the coordinate registration of the PCAVR environment.

- (b) Data pre-filtering. The raw point cloud data from the RGB-D sensors are quite noisy and cluttered. For high-accuracy geometric modelling, a de-noising method is required. In addition, because the virtual robot task space can be estimated, the outlier points can be eliminated accordingly.
- (c) Geometric property definition. For LAFF application, it is necessary to identify the portion of the point cloud that contains the obstacles and the portion of the point cloud that contains the target. The colour homogeneity based object segmentation method is adopted for clustering the obstacles and the target point cloud.
- (d) Guidance constraint generation. According to the geometric property, artificial force field based guidance constraint generation is difficult to establish for an unknown and unstructured environment. An incremental prediction based LAFF estimation method is proposed for real-time guidance constraint generation and is dependent on the REE position, REE velocity, and the surrounding points.

### Coordinate registration

In the PCAVR environment, the robot arm base is set as the origin of the world coordinate system, and the robot arm coordinate system is set as the world coordinate system. Several coordinate systems exist in the teleoperation system, including the robot base coordinate (RbC), the robot end effector coordinate (ReeC), the Kinect coordinate (KC), the marker coordinate (MC), and the PHANToM end effector coordinate (PeeC). Assuming that a point in the point cloud captured by Kinect is described as the vector  $(x, y, z)^T$  in the KC, the transformation from the MC to RbC is  $(R_I, T_I)$ , which is measured by moving the REE to the marker centre,



**Figure 3.** Point cloud augmented virtual reality environment.

and the transformation from KC to MC is  $(R_2, T_2)$ , which is acquired based on a specific marker detection method using ARToolKit. All of the points in the point cloud are transformed into the real world coordinate, as shown in equation (1). The scene of the point cloud with the robot model registration is shown in Figure 3.

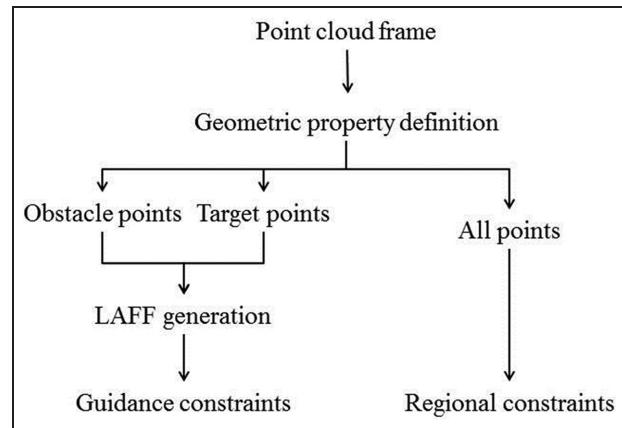
$$(x_w, y_w, z_w)^T = R \bullet (x, y, z)^T + T = R_I(R_2(x, y, z)^T + T_2) + T_I \quad (1)$$

For the isomers of the structures between the robot and the PHANToM device, it is fundamental to realize a continuous, high-resolution mapping method for virtual robot control. Because the motion range of the PHANToM end effector is smaller than the robot motion range, the incremental position calculation method is applied to achieve high-resolution mapping, which adds the deviation value from the initial PHANToM position to the initial REE position. Assuming that the mapped end effector position  $(x_R, y_R, z_R)$ , PHANToM end effector position  $(x_P, y_P, z_P)$ , and the initial positions of the robot and the PHANToM end effector are  $(x_{RI}, y_{RI}, z_{RI})$ ,  $(x_{PI}, y_{PI}, z_{PI})$  respectively, and using the gain factor  $K$ , the relative transformation is  $(R, T)$ . The mapping method is shown in equation (2). The orientation is mapped in the same way.

$$\begin{aligned} (x_R, y_R, z_R)^T &= (x_{RI}, y_{RI}, z_{RI})^T \\ &+ K \bullet [R \bullet (x_P - x_{PI}, y_P - y_{PI}, z_P - z_{PI})^T + T] \end{aligned} \quad (2)$$

### Point cloud pre-filter

The Kinect sensor captures depth data mainly in 4 metres (m) using the Prime Sense technology, which is easily affected by ambient occlusion. Thus, raw point cloud data are noisy and unsuitable for haptic constraints generation. Besides, after registration of the coordinate systems, the spatial range of the interaction between the robot and the worktable can be acquired by moving the robot to the worktable boundaries. The points outside the interaction range are set robot-not-reachable and could cause high computation cost during point cloud processing.



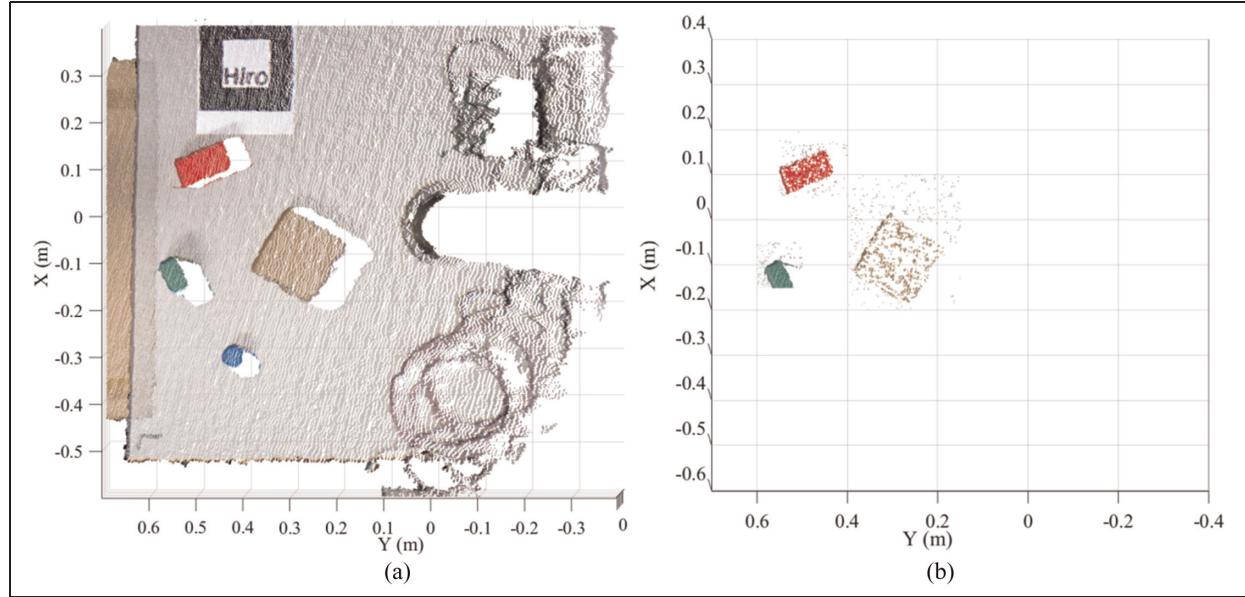
**Figure 4.** Main haptic constraints in the point cloud.

Therefore, two filters are applied for reducing the influence of noise and outliers in the point cloud. Firstly, a pass-through filter is adopted for eliminating the point cloud data outside the workspace limitation mainly on the X and Y axis. The amount of point cloud data can be reduced. Secondly, a  $5 \times 5 \times 5$  median filter (a typical nonlinear filter method) is applied to remove the noise in the point cloud data.

### Haptic constraint definition

Two haptic constraints including regional constraints and guidance constraints are defined in the PCAVR environment. Because the point cloud represents objects that are rigid and cannot be penetrated, all points of the point cloud are treated as regional spaces. Given this assumption, the regional constraints are generated for all points. The guidance constraints in the PCAVR environment are generated according to the LAFF definition, which is dependent on the obstacle and target points. The main haptic constraint generation procedure is shown in Figure 4.

**Geometric property definition.** With the development of computer vision, it has become much easier to segment objects in



**Figure 5.** Geometric property definition: (a) regional point cloud scene, (b) labelled objects after segmentation.

a cluttered environment. Object segmentation is typically used to locate objects in the entire scene. In the proposed system, object segmentation is applied in the point cloud for obstacle and target definitions. Because object segmentation is the most fundamental work for guidance haptic constraint generation, the authors adopt the robust colour-based object segmentation method. The hue character in the Hue-Saturation-Lightness (HSL) colour space is a continuous representation of colour and is invariant to certain types of highlights, shading and shadows (Cheng et al., 2001). Thus, the hue character is adopted for coloured object segmentation. As shown in Figure 5, the objects are segmented and labelled in different colours in the real workspace.

**Guidance haptic constraints based on local AFF.** The original artificial force field (AFF) concept was first proposed for real-time obstacle avoidance by Khatib (1986), and later, it was used in construction of haptic constraints for teleoperation (; Lam et al., 2009; Turro et al., 2001). The main idea is to define a field of force in the task space. The position to be reached is an attractive pole for the REE, and the obstacles are the repulsive surfaces for the REE. With the AFF definition in the workspace, the user can be assisted by the resultant force and obtain the motion tendency when manipulating a robot. Although it is attractive for application in obstacle avoidance and facilitation of target approach, the AFF is restricted owing to difficulty in handling arbitrarily shaped and unknown objects. Different methods have been proposed to define an AFF model. In previous research, most AFFs have been based on simple geometries (cone, curve and cylinder; pre-known), which are not suitable for an unstructured environment. In addition, most methods are focused on 2D navigation. Although certain researchers have attempted to optimize the performance of AFF modelling with arbitrarily

shaped objects (Ni et al., 2013), motion estimation is still challenging for real-time AFF modelling of unstructured obstacles.

It is commonly known that the AFF is similar to an electromagnetic field in the robot motion space (Khatib, 1986). However, in most practical applications, the repulsive field is effective for a limited distance. In other words, the actual force field can be simulated, as shown in Figure 6. In this paper, the authors propose an LAFF estimation method for real-time haptic constraint estimation. This concept is illustrated in Figure 6.

Supposing an object  $Q$  with a negative electric charge is located in an unstructured environment with three pieces of negative plates  $Np1$ ,  $Np2$  and  $Np3$  and one positive charged particle  $Pp$ . The repulsive force is effective only when  $Q$  is located less than radius  $R_f$  from the plates. At the same time, the attractive force is always available for  $Q$ . Therefore, as shown in Figure 6, the guidance of object  $Q$  is generated based on the resultant force of the current effective repulsive force and the attractive force.

Because it is difficult to model the complete unstructured obstacles, the LAFF method based on the regional point cloud is a novel way to consider the modelling of the AFF. First, a local implicit surface definition method is described for LAFF isosurface calculation. Second, the incremental LAFF estimation method based on the modified sigmoid function model is presented. Third, based on the LAFF modelling, the guidance constraints are applied for the haptic feedback to assist human manipulation.

**Local implicit surface based on point clouds.** The implicit surface based on the point cloud structure has been researched for several years. Leeper et al. (2012) proposed an algorithm for rendering point cloud data as an implicit surface defined using the weighted average of nearby points. Because this

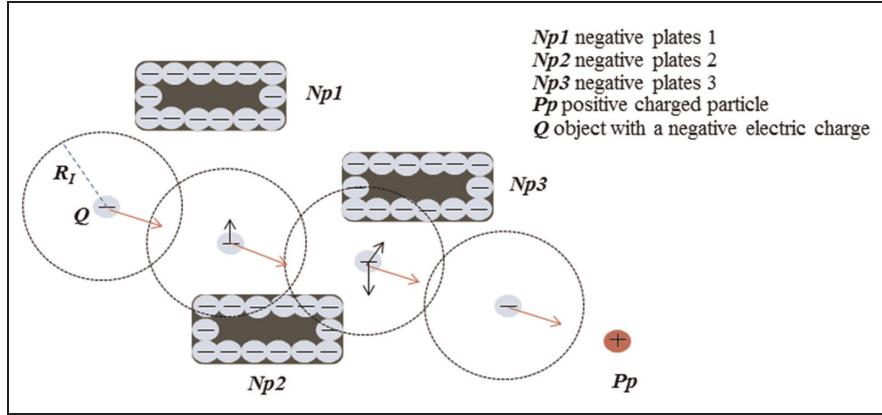


Figure 6. Concept of the LAFF estimation method.

algorithm has been demonstrated to be resistant to sensor noise, it is suitable for arbitrary point clouds and can achieve high efficiency and robustness. This method is slightly modified in the subsequent LAFF definition.

- (1) Point weighting based on the Wendland function  $W(p)$ . This weight function can supply a continuous and smooth surface and suppress undesirable oscillations (Wendland, 1995).

$$r(p) = \|REE - p\| \quad (3)$$

$$W(p) = \begin{cases} \left(1 - \frac{r(p)}{R}\right)^6 \left(35\left(\frac{r(p)}{R}\right)^2 + 18\left(\frac{r(p)}{R}\right) + 3\right) & r(p) \leq R_I \\ 0 & r(p) > R_I \end{cases} \quad (4)$$

where  $r(p)$  represents the distance between the REE and its nearby points from the point cloud acquired using the Kinect sensor, and  $R_I$  is the pre-set influence radius of the local area.

- (2) Point-set implicit surface definition. As depicted by Leeper et al. (2012), the points within the radius  $R_I$  around the REE position can be selected for implicit surface definition. For REE location, the weighted average centre of the point series is defined as  $ep$ , and the weighted average normal is defined as  $en$ .

$$ep = (x, y, z)^T = \left( \frac{\sum_i W(p_i)p_i(x, y, z)^T}{\sum_i W(p_i)} \right) \quad (5)$$

$$en = (x, y, z)^T = \left( \frac{\sum_i W_i(p_i)(REE(x, y, z)^T - p_i(x, y, z)^T)}{\left\| \sum_i W_i(p_i)(REE(x, y, z)^T - p_i(x, y, z)^T) \right\|} \right) \quad (6)$$

Therefore, the relative implicit surface is described as follows

$$f(p) = en^T(p - ep) \quad \{p_x \text{ in } X_{lim}, p_y \text{ in } Y_{lim}, p_z \text{ in } Z_{lim}\} \quad (7)$$

Where  $f(p) = 0$  is the implicit surface. For LAFF modelling, the boundary of the implicit surface is necessarily clarified for safety concerns. Hence,  $X_{lim}$ ,  $Y_{lim}$  and  $Z_{lim}$  are confirmed based on the points in the local area.

**AFF generation based on a modified sigmoid model.** Traditionally, the AFF generation process involves modelling a target as an attractive pole and modelling obstacles as repulsive surfaces. Although the local minima challenge is difficult in global path planning, this problem is less critical in this system because the proposed system is under human operation. The main challenge lies in obstacle modelling. Because the AFF is used in generating haptic constraints in the proposed system, the performance of AFF is critical only for online estimation of the guidance constraints and haptic rendering updates greater than 1 KHz. Therefore, in this paper, an optimized sigmoid model is proposed for obstacle LAFF generation arising from the theory in (Ren et al., 2007). This method is proven to be an accurate analytic description of objects in three dimensions and is rather modest in computation at run time.

AFF generation is based on a modified sigmoid function, with the position of the REE described using a vector  $q = (x, y, z)$  in the workspace. The AFF from the obstacles generated from the modified sigmoid function is shown in equation (8).

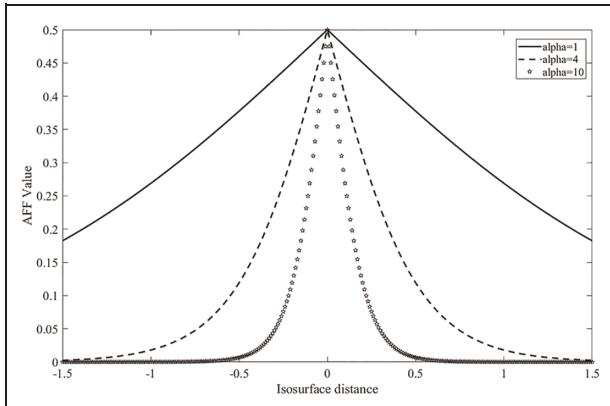
$$U_{rep}(q) = U_{sig}(iso(q)) \quad (8)$$

where

$$iso(q) = f(q) \quad (9)$$

$$U_{sig}(iso) = \frac{I}{1 + e^{\gamma^*|iso|}} \quad (10)$$

Equation (9) is introduced as the isosurface function of the local implicit surface that describes the distance from the local implicit surface. The character of the  $U_{sig}$  model is shown in



**Figure 7.** Influence of the parameter  $\gamma$  on the AFF value.

Figure 7. When the  $iso$  value is nearly zero, the AFF magnitude is the largest, and its gradient is relatively steep compared with others. Additionally, the parameter  $\gamma$  relates to the gradient that influences the generation of the repulsive force.

Supposing the local implicit surface function based on point clouds is  $a(x-2) + b(y-1) + c(z+2) = 0$ ,  $x \in \{0, 1\}$ ,  $y \in \{0, 1\}$ ,  $z \in \{1, 3\}$ . The simulated isosurface data of the implicit surface are shown in Figure 8(a), and the AFF value generated from this first-order plane is shown in Figure 8(b).

In practice, the distance is always positive because the robot moves outside the objects. Therefore, the local force based on AFF modelling can be calculated using equation (11)

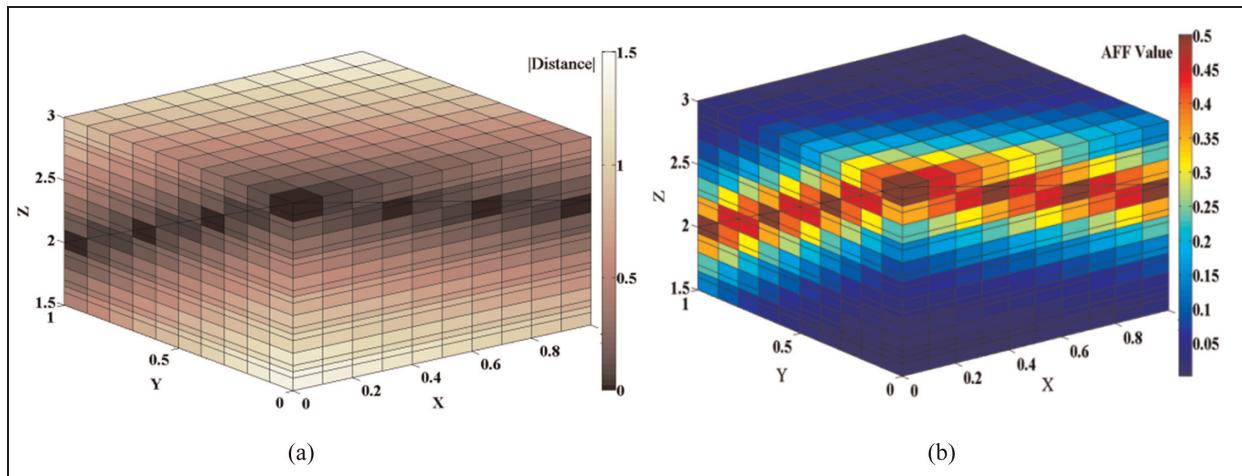
$$U'_{sig}(q) = -\frac{\gamma e^{\gamma^* iso(q)}}{(1 + e^{\gamma^* iso(q)})^2} \frac{\partial(iso(q))}{\partial q} \quad (11)$$

where  $\frac{\partial(iso(q))}{\partial q} = \left[ \frac{\partial(iso(q))}{\partial x}, \frac{\partial(iso(q))}{\partial y}, \frac{\partial(iso(q))}{\partial z} \right]$  indicates the direction of the repulsive force, and  $iso(q)$  is the estimated isosurface at the REE position  $q$ . Using this equation, the repulsive force value and direction for haptic rendering can be calculated.

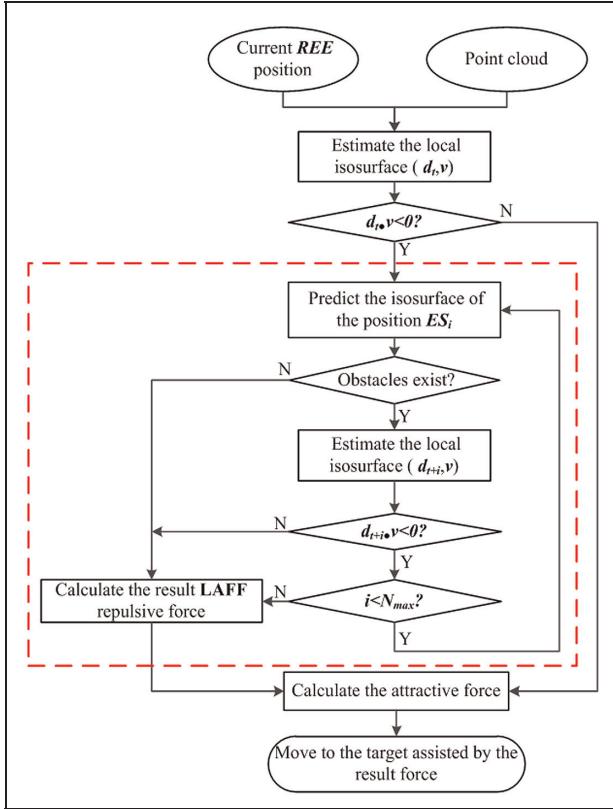
As shown in Figure 8(b), the function isosurface  $iso=0$  when it is located on the surface. At the same time, the gradient of the AFF value is infinitely large owing to the property of the modified sigmoid function (Ren et al., 2007). Although the modified sigmoid function is an effective and low time-consumption method used to construct the artificial field, it is not perfectly suitable for the proposed system. First, because the implicit surface is estimated from the local point cloud, it does not consider the points surrounding the local implicit surface, which might be located on the boundary of the estimated local isosurfaces. Second, the gradient force generated by the modified sigmoid might not be directly suitable for practical use because of the haptic device limitations and restrictions on feedback of the guidance force to the operators. In this work, a magnitude adjustment parameter  $A_F$  for force enforcement is imported. Additionally, the modified sigmoid method based AFF modelling does not consider the velocity. In fact, in selected circumstances, such as moving the robot away from obstacles and remaining stationary, applying a repulsive force on the operator leads to a heavy workload for the operators. It is essential to consider the motion velocity before applying the haptic constraints. With consideration of these conditions, LAFF generation based on the incremental prediction in surrounding point cloud is proposed.

**Incremental prediction based LAFF generation.** Because it is possible to insert the local isosurface with the isosurface of another local area, which could cause oscillations, it is futile to consider only the local estimated surface. When this condition occurs, the repulsive force direction and magnitude are suddenly damped. To address this situation, incremental prediction method based LAFF generation is proposed, as shown in Figure 9.

Incremental prediction method based LAFF generation consists of repulsive force generation and attractive force generation. As shown in Figure 9, the module in the dashed line box is the repulsive force generation procedure. The definition of the parameters in the flowchart and the specific steps are described as follows. For ease of understanding, Figure 10



**Figure 8.** Isosurface and AFF value simulation: (a) isosurface data of the implicit surface, (b) AFF value generated from the isosurface.



**Figure 9.** Flowchart for incremental prediction based LAFF generation.

presents a diagram of repulsive force generation. Assuming an obstacle surface exists in the range of radius  $R_I$  around the REE position, the repulsive force generation method consists of three main steps described as follows:

**Step 1:** Estimate the current isosurface of the REE position, assuming the gradient of the  $U_{sig}$  at the REE position is  $d_t$ , and the velocity of REE is  $v$ , as shown in the blue and yellow lines, respectively.

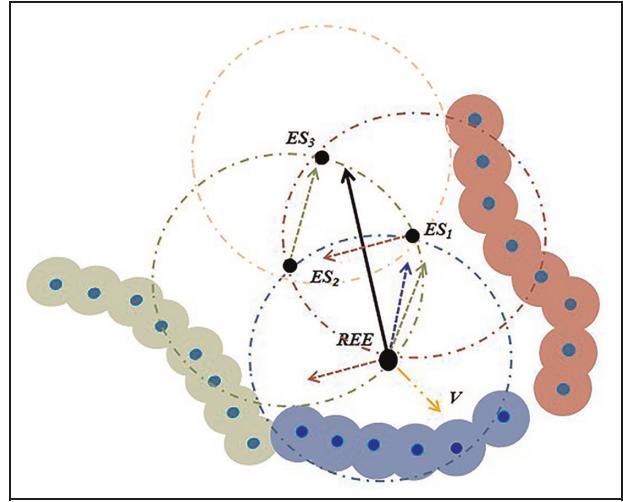
If  $d_t \cdot v < 0$ , go to Step 2.

Else, move to the target assisted by the attractive force.

In this work,  $d_t \cdot v < 0$  means that the motion of REE is towards the local obstacle, and  $d_t \cdot v \geq 0$  means that the REE remains stationary or moves away from the current obstacle.

**Step 2:** Along the  $d_t$  direction, predict one step of movement with  $R_I$  magnitude to the position  $ES_I$ . Estimate the nearby obstacle surface.

If a local obstacle surface exists, acquire the gradient of the  $U_{sig}$  at the  $ES_I$  position  $d_{t+1}$ . Incrementally go to Step 2 and predict the gradient of  $U_{sig}$  at the  $ES_2$  position  $d_{t+2}$  until there is no surrounding obstacle surface. If the number of iteration times is greater than  $N_{max}$ , go to Step 3. The parameter  $N_{max}$  is set to avoid finite computation and is adjusted in the actual application.



**Figure 10** Incremental prediction based LAFF generation.

Else go to Step 3.

**Step 3:** Calculate the resulting LAFF repulsive force shown by the solid line arrow. Although the incremental prediction method considers the obstacles, the repulsive force contributions are distinct as the relative distance changes. Therefore, a weighted function (12) is imported to balance the contribution based on the distance between REE and  $ES$ . Finally, the resulting repulsive force is the sum of the repulsive forces from all obstacles.

$$w(p_{ES_i}) = e^{(-2\|p_{REE}-p_{ES_i}\|)} \quad (12)$$

$$U'_{rep}(p_{REE}) = \begin{cases} \sum_{n=1}^{N_{Obstacles}} \left( U'_{sig\_n}(p_{REE}) + \sum w(p_{ES_i}) U'_{sig\_n}(p_{ES_i}) \right), & \vec{v} \cdot \vec{d}_{t,n} < 0 \\ 0, & \vec{v} \cdot \vec{d}_{t,n} \geq 0 \end{cases} \quad (13)$$

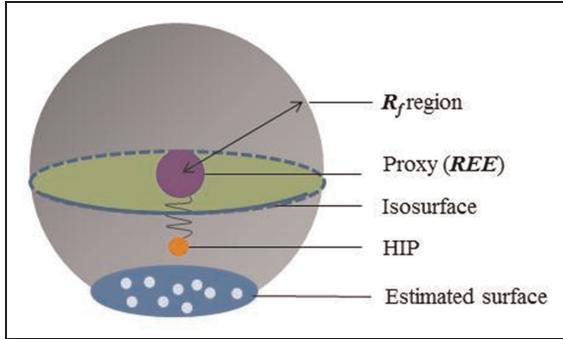
The attractive force field can be treated in a simpler manner. The centre of the target point cloud is used as the attractive pole. Assuming that the target point cloud centre is  $P_{pole}$ , which is acquired by averaging all of the target point cloud data, the attractive force field is defined as in equation (14).

$$U_{att} = I/2\lambda(P_{pole} - p_{REE})^2 \quad (14)$$

where  $\lambda$  is the gain constant. At the same time, the attractive force value is obtained from equation (15)

$$U'_{att} = \lambda(P_{pole} - p_{REE}) \quad (15)$$

Thus, the resultant force from the AFF is the sum of the repulsive force and the attractive force. However, because the resultant force is fed back to the haptic device, which might overtake the hardware output, a magnitude adjustment



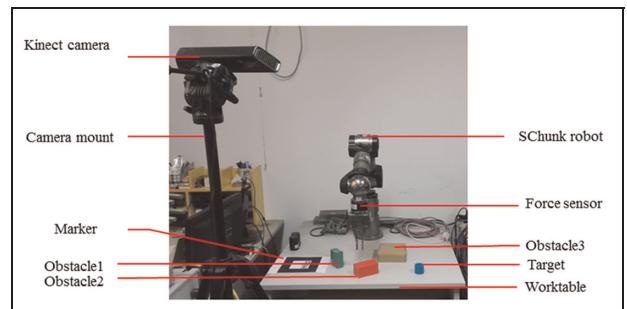
**Figure 11.** Description of regional constraints.

parameter  $A_F$  is adopted for practical applications. Therefore, the guidance constraint is defined by equation (16).

$$\text{Guidance\_Constraint} = A_F \left( U'_{rep} + U'_{att} \right) \quad (16)$$

**Regional constraints.** Several methods exist for three-dimensional haptic rendering in the point clouds. A three-level-radius sphere structured proxy is defined for contact detection (Ryden and Howard, 2013), and a point-set surface estimation method similar to Leeper et al.'s (2012) is used to calculate the force feedback. A related method (Iravani et al., 2016) has also been researched in a detailed experiment that evaluates the haptic rendering method. Xu et al. (2014) adopted this method for model-mediated tele-operation systems. In addition, another method maps the haptic interface point (HIP) into the pixel coordinates and calculates the haptic rendering using the depth data (Rasool and Alexei, 2016). This method is proven with fast collision detection. In the proposed system, the regional constraint generation method extends the theory of haptic rendering and uses the implicit surface method in Section 3.3.2. The regional constraints can retain the proxy on a protection surface in the point cloud by constructing an isosurface, which is the boundary of the regional constraints, as shown in Figure 11. Distinct from LAFF generation, the regional constraints are set for all points in the point cloud because any collision is risky for robot manipulation. Additionally, to reduce the computation, the surface detection area is limited by the radius  $R_f$  of the local area, which is much smaller than  $R_l$  in the LAFF generation module.

As shown in Figure 11, the HIP represents the motion of the PHANTOM end effector. To achieve collision-free operation in the proposed system, the proxy (shown in Figure 11) in the PCAVR environment represents the REE position of the virtual robot. When there are no points in the  $R_f$  region, the position of the proxy is the same as the HIP position, and the virtual robot is only influenced by the LAFF. Once there are points in the  $R_f$  region, as shown by the blue points, the implicit surface is calculated as shown by the blue plane, and the isosurface is shown by the green plane. Additionally, the regional constraints prevent collision. The force value generated by the regional constraints is linear to the distance between the proxy and the HIP and is calculated from equation (17), where  $K$  is the scaling factor. To obtain a more realistic force constraint, the velocity is added for damping spring force modelling. In this condition, the position of the proxy is



**Figure 12.** System setup.

estimated in real time using equation (18). In this equation,  $B$  is the damping factor, and  $RR$  is the pre-set value of  $R_f$ .

$$F_{RC} = K(P_{proxy} - P_{HIP}) + B \bullet V_{HIP} \quad (17)$$

$$P_{proxy} = ep + en \bullet RR \quad (18)$$

## Experiment and discussion

### System setup

The PCAVR environment is constructed based on the real slave environment, which consists of a seven degree-of-freedom (DOF) SChunk robot arm, the Kinect sensor, and the workstation. At the master side, the Geomagic Touch haptic, which can supply 6-DOF position measurements with force feedback, is used to realize the haptic interaction. The Kinect sensor is installed, and the position is fixed beside the worktable. The software is based on VS2010, the PCL library, and the OpenCV library at the master side to complete the haptic constraint-based PCAVR construction, and the OpenGL is used to render the 3D graphics at the master side. The computers work at 3.2 GHz. The actual system setup is shown in Figure 12. The manipulation workspace with the origin at the robot base, as captured by the Kinect, is limited by [-0.6, 0.4] [-0.7, 0.4] [-0.4, 0.2] [m] in the X, Y and Z directions, respectively. Three obstacles are placed on the worktable, and the marker is fixed on the worktable corner.

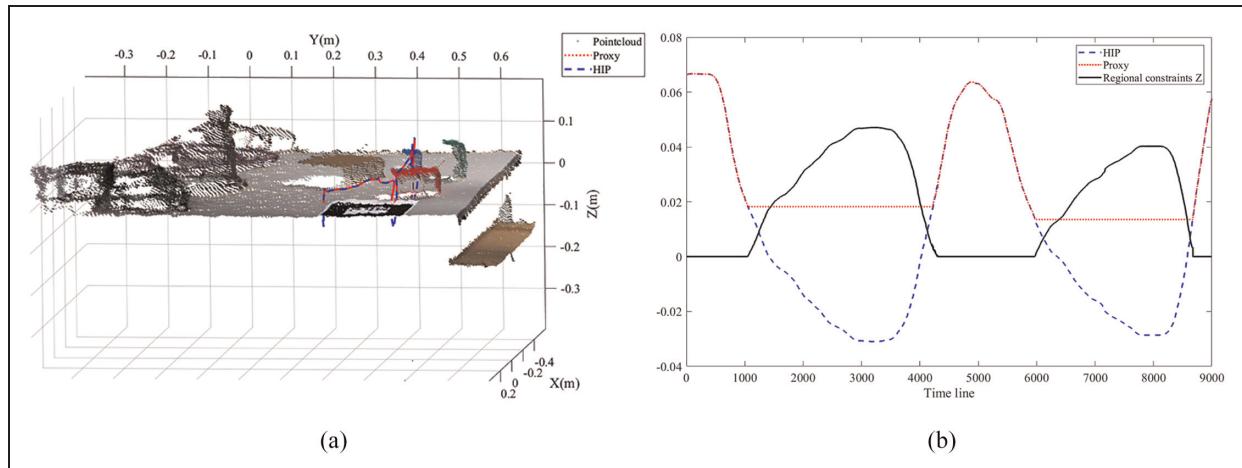
### Constraint generation evaluation

After the PCAVR environment reconstruction, two specific tasks are designed for the operator to evaluate the constraint generation method. These tasks are described as follows.

**Task I:** Control the virtual robot using the haptic device in the PCAVR environment arbitrarily without specifically avoiding the objects.

**Task II:** Control the virtual robot and move towards three different target positions.

In addition, the operator is required to complete each task in the PCAVR environment without the haptic constraints (Environment I) and in the PCAVR environment with the haptic constraints proposed in this paper (Environment II).



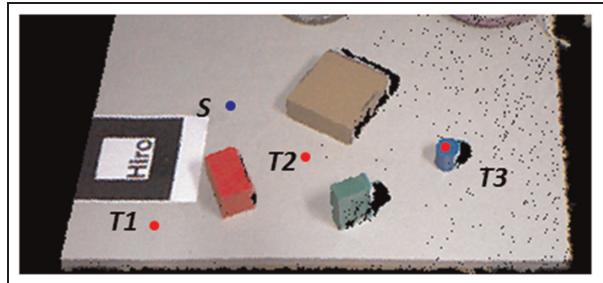
**Figure 13.** Regional constraint evaluation: (a) moving routes of HIP and the REE, (b) regional constraints during motion in the Z axis.

**Regional constraint evaluation.** To test the effectiveness and efficiency of the regional constraint generation method, the operator is asked to complete Task I in Environment I and Environment II separately. In the application, the surface detection area is limited by the radius  $R_f$ , and  $R_f$  is set to 0.001 m. The scaling factor  $K$  is set to 10, and the damping factor  $B$  is set to 0.1.

Figure 13 depicts the procedure in which the operator moves the virtual robot in Environment II. The positions of REE (the proxy), the HIP, and the regional constraints are noted and shown in Figure 13(a) in the dot line, dashed line, and solid vector, respectively.

As observed from Figure 13, the dot line describes the REE (the proxy) position, and the dashed line shows the HIP position motion. The solid line shows the force of the regional constraints. In addition, the REE and HIP positions and regional constraints in the Z-axis in the real-world coordinates are shown in Figure 13(b). When the operator moves the virtual robot in a free space, the HIP maintains the same velocity and position as the motion of the proxy in the VR environment. However, when the REE enters the regional constraints, the HIP penetrates the regional constraints and the proxy remains on the boundary of the regional constraints. The haptic constraints are instantly generated when penetration occurs. With the regional constraints, no collision occurs during the operation in Environment II. At the same time, the motion procedure and the number of collisions in Environment I can be observed. In Environment I, no regional constraints exist to regulate the motion of the human operators. The number of contacts between the REE and the point cloud is quite high during the operator's arbitrary manipulation. Therefore, the regional constraint generation method is demonstrated effectively during real-time manipulation.

**Guidance constraint evaluation.** After the obstacles and target have been labelled automatically by the geometric property definition module, the guidance constraint is generated in real time during the motion of the virtual robot. The operator operates the haptic device and manipulates the virtual robot with the guidance force. The start area is labelled as point S,

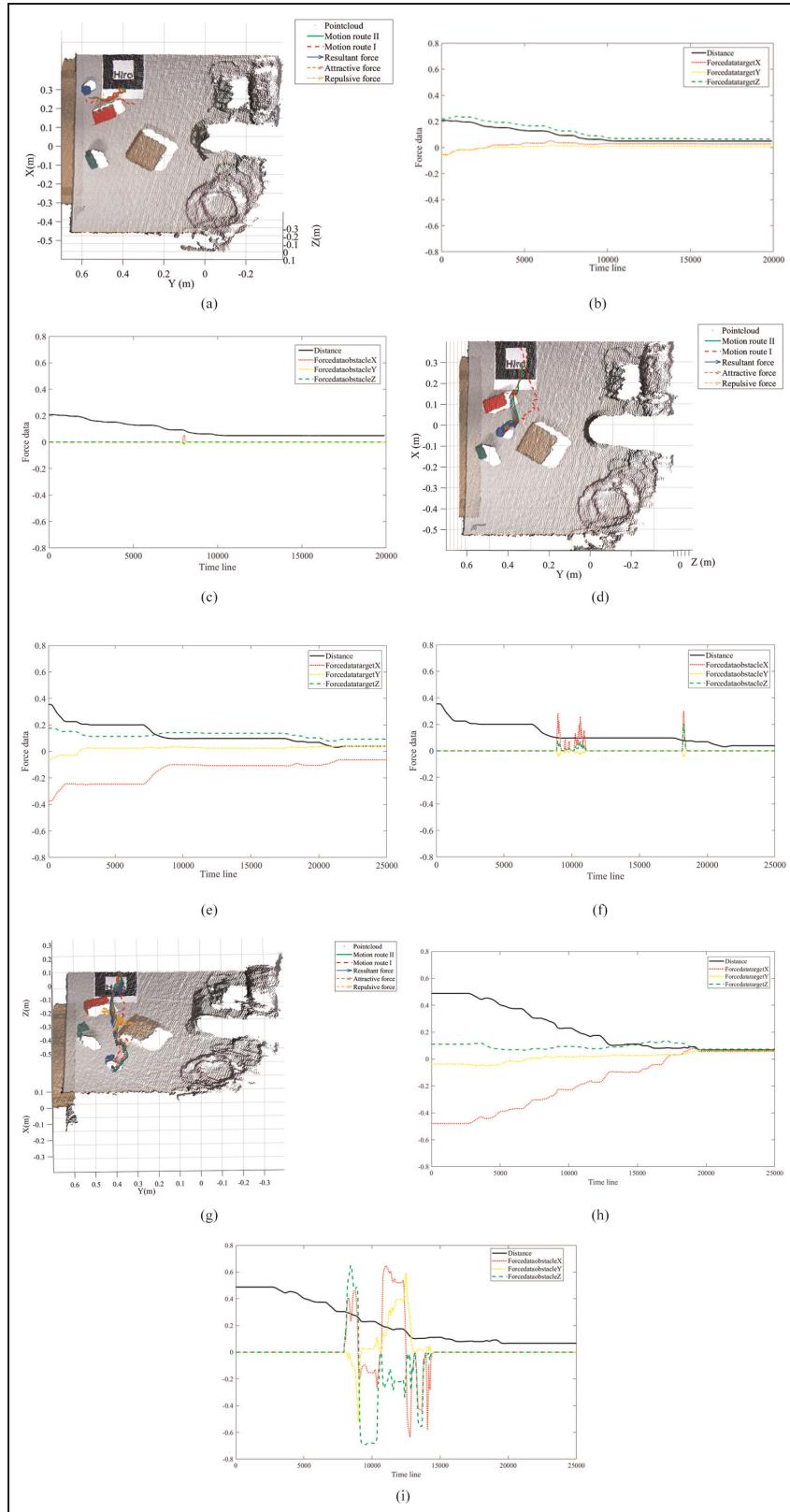


**Figure 14.** Labelled areas in Task II.

while three different targets are labelled as T1, T2 and T3 respectively. The operator is required to move the robot to the target area. During this task, there is no assistance but only visualization in Environment I, whereas in Environment II, there is real-time guidance constraint assistance. In the application, the local area is limited by the radius  $R_I$ , and  $R_I$  is set to 0.05 m. The number of iterations  $N_{max}$  is set to 5 for high-speed computation.

Six routes are noted and drawn in Figure 15. The dashed lines in Figures 15(a), (d) and (g) show the motions in Environment I towards T1, T2 and T3, respectively, and the solid lines show the motions in Environment II. The solid arrows show the real-time resultant forces applied during motion, the dash-and-dot arrows yellow line represents the repulsive forces, and the dashed arrows show the attractive forces. In Figures 15(b), (c), (e), (f), (h) and (i), the solid line represents the distance between the REE and the target areas T1, T2 and T3. The generated attractive forces and repulsive forces are respectively illustrated.

According to Figure 15(a), (b) and (c), few repulsive forces are generated when moving towards T1. Because T1 is located relatively far away from any obstacles, the repulsive pulse is triggered when HIP is near the obstacles. The resultant guidance forces are mainly composed of the attractive force, and the operator can easily control the robot towards the target position. As shown in Figure 15(d), (e) and (f), when the motion occurs towards the obstacle, a relatively large



**Figure 15.** Guidance constraint evaluation: (a), (d), (g) moving routes in both environments towards T1, T2 and T3; (b), (e), (h) values of the attractive force applied in Environment II in the X, Y, and Z axis, respectively; (c), (f), (i) values of the repulsive force applied in Environment II in the X, Y, Z axis, respectively.

**Table I.** Computational speed evaluation.

Function	Thread 1 Haptic rendering on haptic devices	Thread 2 Regional constraint generation	Thread 3 Guidance constraint generation
Communication	1 ms	-----	-----
Implicit surface estimation	-----	16 ms	95 ms
Force calculation	-----	1 ms	3 ms

repulsive force is generated, and the resultant force is affected and modified. The repulsive force leads to motion away from the obstacles, and the resultant force leads to motion towards the target. Similarly, Figure 15(g), (h) and (i) shows the motion towards T3. Additionally, it can be observed that the repulsive force is reduced sharply to zero when the motion is away from the obstacle. Therefore, a certain amount of jittering occurs, as shown in the resultant force data in Figure 15(f) and (h). From Figure 15(a) (d) and (g), it is obvious that motion without any assistance can be affected by human jitter and vision deviation.

In addition, comparing the motion in Environment I with that in Environment II, it is much easier for an operator to control the robot in the PCAVR with the haptic constraints. With the guidance constraint assistance, the motion is much smoother and simpler.

**Computational speed evaluation.** The speed of the costly components of the constraint generation processing pipeline is profiled first. In the proposed methodology, three main threads are designed for haptic rendering implementation. First, haptic rendering for communication with the haptic device is implemented in one thread (Thread 1). For a realistic haptic impression for human operation, this thread works at 1 kHz. The haptic feedback value is refreshed through the haptic constraint generation threads.

Second, two different threads are distributed for regional constraint generation (Thread 2) and guidance constraint generation (Thread 3). Both constraints are based on the local implicit surface estimation method. The computational cost of the implicit surface method must be observed. In addition,

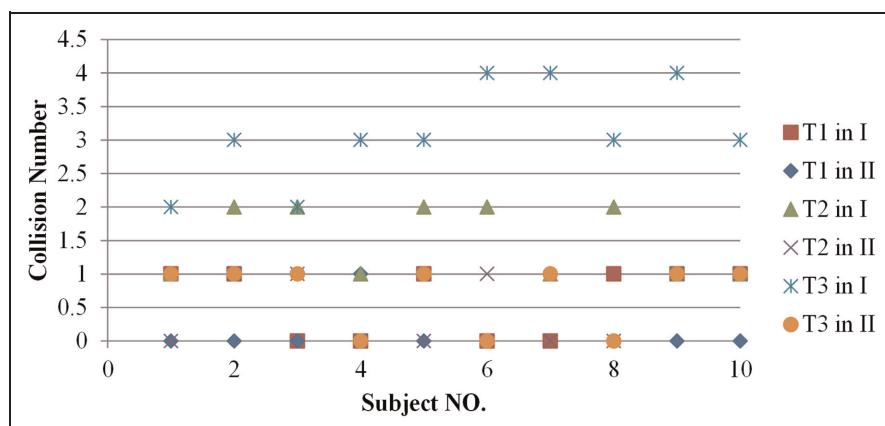
during the guidance constraint generation, LAFF generation accounts for most of the computational cost. Thus, these two aspects are evaluated. In this system, the point cloud is assumed to be static, which could sharply reduce the computational cost.

After evaluation, the time spent on the implicit surface estimation is approximately 16 ms, whereas the LAFF generation requires approximately 100 ms. A detailed description of the computational speed evaluation is shown in Table 1.

Furthermore, it is observed that the LAFF computation is deeply influenced by the number of iterations  $N_{max}$ . Although the computation can guarantee smooth operation, the haptic constraint generation method still must be optimized to reduce the time cost to create more transparent haptic perception for human operators.

**User study.** The system was tested by 10 subjects, five with teleoperation experiences and five without. Before the test, the subjects were asked to learn the manipulation procedure and be familiar with haptic constraint perception in several trials. In this experiment, the subjects were asked to repeat the steps 10 times. The subjects performed this experiment to accomplish Task I and Task II in Environment I and in Environment II separately. During Task I, the average number of collisions of each subject was recorded, and the average time spent on reaching the target of each subject is also noted during Task II.

When the subject implements Task I in Environment I, the number of collision times is large. Thus, we mainly compared the average number of collisions for each subject in Environment I and II during Task II as shown in Figure 16.

**Figure 16.** Number of collisions for Task II.

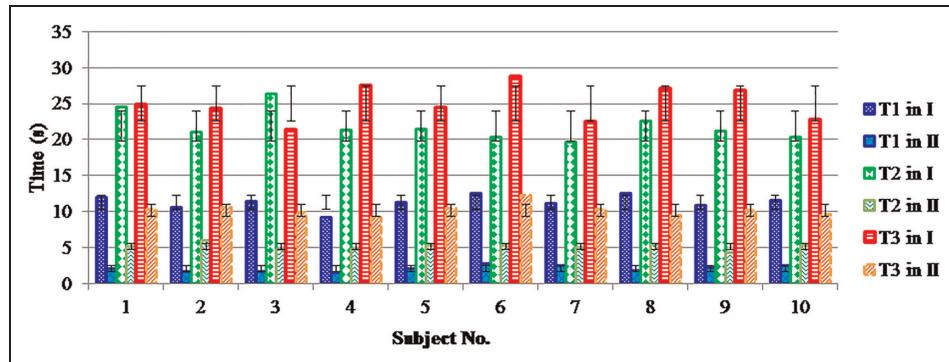


Figure 17. Time required for Task II.

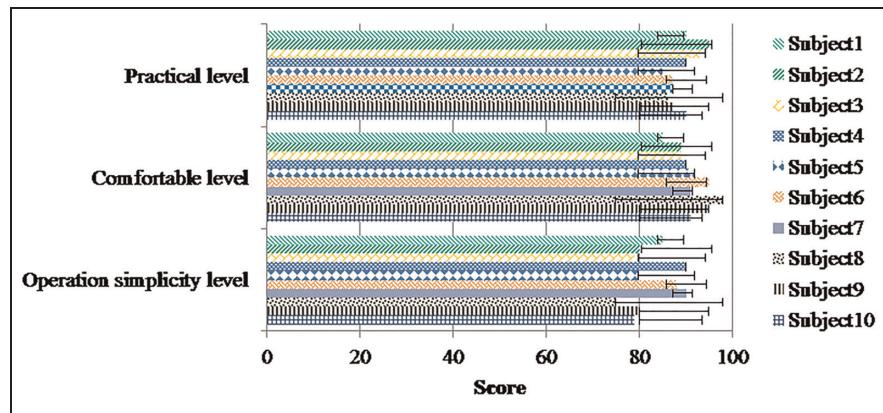


Figure 18. Average scores with haptic constraints.

Because the point cloud might display low resolution and holes, this can affect the simulation of the surface edge. Thus, even with the assistance of regional constraints and guidance constraints, collisions might still occur. However, as shown in Figure 16, collisions occur much more frequently in Environment I than in Environment II owing to dependence on vision information only. It is obvious that the haptic constraints can effectively reduce the possibility of collisions.

Furthermore, the average time of each subject is noted, as shown in Figure 17. The first two columns show the time required to reach T1 in Environments I and II, respectively, and the average times required are 11.29 and 2.03 s respectively, with a reduction of 82%. The middle two columns represent the time required to reach T2 in Environment I and II, and the average time required are 21.87 and 5.09 s, respectively, with a reduction of 77%. The last two columns describe the time required to move towards T3 in Environments I and II. The average time required to accomplish the task are 25.1 and 10.2 s, with a 59% reduction. It is obvious from the experimental data that the haptic constraint assistance can reduce the task implementation time significantly.

After this experiment, a questionnaire was designed to investigate the subjects' feelings during implementation of the tele-operation tasks with and without haptic constraints. The questionnaire is mainly composed of three aspects (practical level, comfortable level and operation simplicity level), and each aspect is assumed to have a score of 100. When

operating with haptic constraints, the average scores for the practical, comfortable, and operation simplicity levels are 89.0, 92.9 and 82.6 respectively, whereas the average scores without haptic constraints are 54.1, 64.0 and 60.5, respectively. As shown in Figure 18, subjects feel helpful about the task implementation and comfortable in perceiving the haptic feedback. Furthermore, it is easy for subjects to operate with the haptic constraints in the PCAVR environment.

## Conclusion

In this paper, a novel workflow in the PCAVR environment is proposed with haptic constraints for teleoperation. The regional constraint and the guidance constraint generation methods were evaluated effectively using experiments. The point cloud that represents the real environment is protected by the regional constraints in real time. The guidance constraints greatly assist the operator in accomplishing the tasks. As seen from the experimental result, the task accomplishment efficiency can be significantly improved and the human workload can be greatly reduced with the aid of haptic constraints.

However, certain limitations exist in the experimental portion because collisions can occur at the boundary of the holes if selected point data are absent in the point cloud. In the view of these limitations, selected optimization objectives (mainly related to the point cloud hole inpainting and repulsive force smoothing) are research topics for future work. Furthermore,

specific visual cues will be designed and added into the PCAVR environment for a multimodal feedback option. Meanwhile, future research will also look into the performance of the multimodal feedbacks on complicated environments by evaluating the ergonomics during the human robot interaction.

### Declaration of conflicting interest

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

### Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was funded by National Key Research and Development Program of China (grant number No.2016YFB1001301), National Science Foundation for Distinguished Young Scholars of China (grant number 61325018), the China Scholarship Council, the National University of Singapore, Natural Science Foundation of China (grant number 61403080), Jiangsu Provincial Science Technology Support Program (grant number BE2014132).

### References

- Abbott JJ and Okamura AM (2006) Stable forbidden-region virtual fixtures for bilateral telemanipulation. *Journal of dynamic systems, measurement, and control* 128(1): 53–64.
- Bohren J, Guerin K, Xia T, et al. (2011) Toward practical semi-autonomous teleoperation: Do what I intend, not what I do. In: *2011 IEEE Workshop on Advanced Robotics and its Social Impacts*, California, USA, 2–4 October 2011, pp. 20–23. New York: IEEE.
- Bowyer SA, Davies BL and Baena FR (2014) Active constraints/virtual fixtures: A survey. *IEEE Transactions on Robotics* 30(1): 138–157.
- Cheng HD, Jiang XH, Sun Y and Wang J (2001) Color image segmentation: advances and prospects. *Pattern recognition* 34(12): 2259–2281.
- Iravani E, Talebi HA, Zareinejad M and Dehghan MR (2016) Real-time point-cloud-based haptic rendering with fast contact detection. In: *2016 4th International Conference on Robotics and Mechatronics (ICROM)*, Tehran, Iran, 26–28 October 2016, pp. 578–583. New York: IEEE.
- Khatib O (1986) Real-time obstacle avoidance for manipulators and mobile robots. *The international journal of robotics research* 5(1): 90–98.
- Lam TM, Boschloo HW, Mulder M and Van Paassen MM (2009) Artificial force field for haptic feedback in UAV teleoperation. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans* 39(6): 1316–1330.
- Leeper A, Chan S and Salisbur K (2012) Point clouds can be represented as implicit surfaces for constraint-based haptic rendering. In: *IEEE International Conference on Robotics and Automation*, St Paul, USA, 14–18 May 2012, pp. 5000–5005. New York: IEEE.
- Li M, Ishii M and Taylor RH (2007) Spatial motion constraints using virtual fixtures generated by anatomy. *IEEE Transactions on Robotics* 23(1): 4–19.
- Maddahi Y, Zareinia K and Sepehri N (2015) An augmented virtual fixture to improve task performance in robot-assisted live-line maintenance. *Computers & Electrical Engineering* 43(C): 292–305.
- Marayong P, Li M, Okamura AM and Hager GD (2003) Spatial motion constraints: Theory and demonstrations for robot guidance using virtual fixtures. In: *2003 IEEE International Conference on Robotics and Automation*, Taiwan, China, 14–19 September 2003, pp. 1954–1959. New York: IEEE.
- Nam CS, Richard P, Yamaguchi T and Bahn S (2014) Does touch matter? The effects of haptic visualization on human performance, behavior and perception. *International Journal of Human-Computer Interaction* 30(11): 839–841.
- Ni DJ, Nee AYC, Ong SK and Song AG (2016) Local implicit surface based virtual fixtures in point cloud virtual environment for teleoperation. In: *The 10th International Conference on Sensing Technology*, Nanjing, China, 11–13 November 2016, pp. 588–593.
- Ni T, Zhang H, Xu P and Yamada H (2013) Vision-based virtual force guidance for tele-robotic system. *Computers and Electrical Engineering* 39(7): 2135–2144.
- Nia Kosari S, Rydén F, Lendvay TS, et al. (2014) Forbidden region virtual fixtures from streaming point clouds. *Advanced Robotics* 28(22): 1507–1518.
- Passenberg C, Peer A and Buss M (2010) A survey of environment-, operator-, and task-adapted controllers for teleoperation systems. *Mechatronics* 20(7): 787–801.
- Peshkin MA, Colgate JE, Wannasuphprasit W, et al. (2001) Cobot architecture. *IEEE Transactions on Robotics and Automation* 17(4): 377–390.
- Prada R and Payandeh S (2009) On study of design and implementation of virtual fixtures. *Virtual Reality* 13(2): 117–129.
- Raiola G, Rodriguez-Ayerbe P, Lamy X, et al. (2015) Parallel guiding virtual fixtures: Control and stability. In: *IEEE International Symposium on Intelligent Control*, Sydney, Australia, 21–23 September 2015, pp. 53–58. New York: IEEE.
- Rasool S and Alexei S (2016) Real-time haptic interaction with RGBD video streams. *The Visual Computer* 32(10): 1311–1321.
- Ren J, McIsaac KA, Patel RV and Peters TM (2007) A potential field model using generalized sigmoid functions. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 37(2): 477–484.
- Rosenberg LB (1993) Virtual fixtures as tools to enhance operator performance in telepresence environments. In: *International Society for Optics and Photonics*, Boston, USA, 21 December 1993, pp. 10–21. SPIE Proceedings.
- Ryden F and Howard JC (2013) A proxy method for real-time 3-DOF haptic rendering of streaming point cloud data. *IEEE transactions on Haptics* 6(3): 257–267.
- Sun D, Naghdy F and Du H (2014) Application of wave-variable control to bilateral teleoperation systems: A survey. *Annual Reviews in Control* 38(1): 12–31.
- Turro N, Khatib O and Coste-Maniere E (2001) Haptically augmented teleoperation. In: *IEEE International Conference on Robotics and Automation*, Seoul, Korea, 21–26 May 2001, pp. 386–392. New York: IEEE.
- Wendland H (1995) Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree. *Advances in computational Mathematics* 4(1): 389–396.
- Xia T, Léonard S, Deguet A, et al. (2012) Augmented reality environment with virtual fixtures for robotic telemanipulation in space. In: *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vilamoura-Algarve, Portugal, 7–12 October 2012, pp. 5059–5064. New York: IEEE.
- Xu X, Cizmeci B, Al-Nuaimi A and Steinbach E (2014) Point cloud-based model-mediated teleoperation with dynamic and perception-based model updating. *IEEE Transactions on Instrumentation & Measurement* 63(11): 2558–2569.
- Yamamoto T, Abolhassani N, Jung S, et al. (2012) Augmented reality and haptic interfaces for robot-assisted surgery. *International Journal of Medical Robotics + Computer Assisted Surgery Mrcas* 8(1): 45–56.