



Article A Guiding and Positioning Motion Strategy Based on a New Conical Virtual Fixture for Robot-Assisted Oral Surgery

Yan Wang¹, Wei Wang¹, Yueri Cai^{1,*}, Qiming Zhao¹, Yuyang Wang¹, Yaoqing Hu² and Shaoan Wang²

¹ School of Mechanical Engineering and Automation, Beihang University, Beijing 100191, China

² College of Engineering, Peking University, Beijing 100871, China

* Correspondence: caiyueri@buaa.edu.cn

Abstract: In robot-assisted oral surgery, the surgical tool needs to be fed into the target position to perform surgery. However, unmodeled extraoral and complex intraoral environments bring difficulties to motion planning. Meanwhile, the motion is operated manually by the surgeon, causing relatively limited accuracy as well as the risk of misoperation. Moreover, the random movements of the patient's head bring additional disturbance to the task. To achieve the task, a motion strategy based on a new conical virtual fixture (VF) was proposed. First, by preoperatively specifying a conical guiding cone as the VF, virtual repulsive forces were applied on the out-of-range end effector. Then, based on the two-point adjustment model and velocity conversion, the effect of VF was established to prevent the end-effector from exceeding the constraint region. Finally, a vision system corrects the guiding cone to compensate for the random movement of the patient's head to feed to a dynamic target. As an auxiliary framework for surgical operation, the proposed strategy has the advantages of safety, accuracy, and dynamic adaptability. Both simulations and experiments are conducted, verifying the feasibility of the proposed strategy.

Keywords: oral surgery; motion planning; virtual fixture; adjustment model; vision system



Citation: Wang, Y.; Wang, W.; Cai, Y.; Zhao, Q.; Wang, Y.; Hu, Y.; Wang, S. A Guiding and Positioning Motion Strategy Based on a New Conical Virtual Fixture for Robot-Assisted Oral Surgery. *Machines* **2023**, *11*, 3. https://doi.org/10.3390/ machines11010003

Academic Editors: George Moustris and Costas Tzafestas

Received: 19 November 2022 Revised: 16 December 2022 Accepted: 18 December 2022 Published: 21 December 2022



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/).

1. Introduction

1.1. Background and Task

Oral diseases, such as oral tumors and tooth loss, are very common worldwide [1–3], and reduce the life quality of patients [4,5]. Oral surgery, such as tumor resection and dental implant, is indispensable in the treatment of oral diseases [6,7]. Traditional oral surgery relies entirely on the manual operation of the surgeon, so the accuracy, safety, and prognosis quality cannot be well guaranteed. The development of robotics provides new ways for oral surgery, which has great potential to assist surgeons to provide high-quality surgery and become the mainstream solution in the future [8–10].

This paper focuses on a new oral surgery robot system (OSRS) that is under development [11–13]. To perform the surgery, it is an important step to feed the end-effector from an arbitrary initial position outside the mouth into the target position inside the oral cavity, which is also the research objective of this paper.

This task is similar to the motion planning of industrial manipulators moving from a start position to the destination, and there are two typical methods: heuristic algorithm [14] and optimization-based algorithm [15]. However, as for the OSRS, unmodeled extraoral environment, complex oral geometry, and dynamic target are three features that bring additional difficulties. First, the extraoral environment is unstructured, which means obstacles like operating tables are not modeled in advance, making it difficult to deploy automatic planning due to the lack of collision detection. Second, the oral cavity is similar to the lofting surface with a complicated structure [16], which is adverse to the real-time performance of collision detection and automatic planning. Third, patients in oral surgery are usually not under general anesthesia, thus there are random disturbance movements

in the head, making the oral cavity dynamic rather than static, which requires real-time trajectory generation. In addition, the surgical tool is usually a rigid rod, forming a geometric configuration in which a rigid rod (end effector) is inserted into the oral cavity through the mouth. This configuration makes the end effector surrounded by the inner surfaces of the oral cavity, rather than a typical bounding box such as a cuboid, sphere, or cylinder [17]. This condition limits the adjustment space of the end-effector and reduces the fault tolerance. To sum up, the feeding task is more complex, and an effective strategy needs to be designed to achieve safe and accurate guiding and positioning.

1.2. Related Work

In recent years, oral surgical robots have been studied by scholars. On the one hand, computer vision is widely used. Ma et al. [18] proposed a markerless model to execute preoperative planning for the maxillofacial surgery robot. Hu et al. [19] used deep learning to detect the oral cavity and generated the trajectory for a transoral robot by using oral center constraint and optimization methods. Cheng et al. [20] obtained the implantation point through an L-shaped intraoral marker, and its transformation is derived to achieve accurate positioning. Toosi et al. [21] developed simulation software for a master-slave dental robot using collision detection. On the other hand, master-slave control and cooperative control are commonly adopted. Kasahara et al. [22] achieved better dental drilling through the hybrid force/position control strategy in the master-slave control for dental drilling robot, and Iijima et al. [23] also adopted a similar control strategy in their work. Li et al. [24] used motion mapping with position and pose separated and converted the pose increments by the rotation vector method. Kwon et al. [25] developed a master-slave transoral robot system. The above works have studied the navigation and motion control of oral surgical robots, where master-slave control and computer vision are mostly adopted, and meaningful contributions are made.

Besides, as a key means for guidance and safety, virtual fixture (VF) plays an important role in surgical robots and has attracted many scholars. Rosenberg proposed the concept of VF for teleoperation [26]. Abbott et al. [27,28] and Bettini et al. [29] proposed different types of VF, and admittance control was combined [30]. Kikuuwe et al. [31] proposed a plastic VF that can either guide the operation or force it to escape from the VF. These works have built the foundation of VF, which is widely used in medical robots [32,33]. Subsequently, some scholars have done more research on VF. Tang et al. [34] simplified the organs from a group of points, and then established the VF through collision detection, artificial potential fields, and force feedback, to achieve safe master-slave operation. Pruks et al. [35] used an RGB-D camera to generate the VF in unstructured environments. Similarly, Marinho et al. [36] constructed the VF using vector field inequalities through a set of points. Xu et al. [37] and Li et al. [38] designed virtual fixtures for orthopedic surgery and nasal surgery respectively using 3D models and geometric algorithms. Ren et al. [39] trained a neural network to fit the geometry of the heart to build a virtual fixture. He et al. [40] established VF for an ophthalmic robot by converting the contact force to the velocity of the probe. The above research enrich the theory and application of VF and are very constructive to the work of this paper.

However, prior studies rarely involved VF to better adapt to oral surgery. Additionally, most VF focus only on position control, while orientation is less involved, lowering the fault tolerance under the geometric configuration of oral surgery. In general, motion strategies for oral surgery robots are still limited, and a convenient and integrated solution is still needed. Thus, this paper proposes a guiding and positioning strategy for the OSRS. In the strategy, a flared guiding cone is designed as the VF to better fit the oral structure after the master–slave control is built. Subsequently, the two-point adjustment model and velocity conversion are derived as the effect of the VF to adjust the surgical tool. Next, the VF is integrated with the vision system to compensate for the head disturbance of the patient, becoming the full strategy proposed. In addition, the strategy also provides an

optional, automatic mode that differs from typical automatic planning methods. The main contributions of this paper can be summarized as follows:

- A new conical virtual fixture is proposed for surgical operation with the master–slave mapping built. In the virtual fixture, a flared guiding cone is designed in accordance with the geometric configuration of oral surgery.
- Two-point adjustment model and velocity conversion are proposed to be the effect of the VF, which can simultaneously adjust the position and orientation of the tool.
- A new mouth opener is used as a marker for the vision system to locate the oral cavity. The guiding cone is corrected in real-time to compensate for the random disturbance of the patient's head and realize the feeding to the dynamic target.
- The new VF and vision system are integrated into a full guiding and positioning strategy. This strategy serves as a novel active adjustment framework that not only assists safe and accurate feeding, but also provides an automatic mode to choose.

The rest of this paper is organized as follows. Section 2 presents the OSRS, where hardware constitution and master–slave mapping are given. Section 3 introduces the concept of the conical virtual fixture. Section 4 describes the effect of the guiding cone, including the two-point adjustment model and velocity conversion, and hence the new VF is fully designed. Section 5 introduces system integration, especially the combination of a binocular vision system. Simulations and experimental validations are given in Section 6. Finally, conclusions are drawn in Section 7.

2. Oral Surgery Robot System (OSRS)

2.1. Hardware Constitution

The hardware of OSRS includes a universal manipulator, a surgical instrument, a master controller, a mouth opener, and a vision system. The universal manipulator has 6 DOFs, with the surgical instrument mounted on it. Surgical instruments, also the end effector, have four types: wound suture, dental implant, laser drilling, and tumor resection. The master controller is a Phantom Omni product, by which the surgeon can drive the manipulator and take the surgical instrument into the oral cavity. The mouth opener can expand the patient's mouth while serving as a marker for vision localization. The vision system can locate the oral cavity when the mouth opener is detected. The OSRS is shown in Figure 1a, and the mouth opener is shown in Figure 1b.



Figure 1. The OSRS: (a) The overall system; (b) Local magnification of the mouth opener.

2.2. Master-Slave Mapping

The master–slave mapping of the OSRS includes position mapping and orientation mapping. The position and orientation of the master controller can be obtained from the phantom's API. Both position mapping and orientation mapping are incremental, which

means the master controller's increment is mapped to the slave manipulator by a scaling factor. For position, the mapping is shown in Equation (1).

$$\Delta \boldsymbol{p}_{\rm s} = k_{\rm pos} \Delta \boldsymbol{p}_{\rm m} = k_{\rm pos} \left(\boldsymbol{p}_{\rm m}^{(k)} - \boldsymbol{p}_{\rm m}^{(k-1)} \right) \tag{1}$$

where Δp_s represents the position increment executed by the slave manipulator, Δp_m is the position increment of the master controller, k_{pos} is a scaling factor for position mapping, and $p_m^{(k)}$ and $p_m^{(k-1)}$ are the positions of the last two frames of the master controller. Next, for orientation, the increment of the main controller is shown in Equation (2).

$${}^{k-1}_{k}\boldsymbol{R} = {\binom{\mathbf{m}}{k-1}\boldsymbol{R}}^{-1}_{k}\boldsymbol{R}$$
(2)

where ${}^{k-1}_{k}R$ is the rotation matrix between the last two frames of the master controller, ${}^{m}_{k}R$ and ${}^{m}_{k-1}R$ are the rotation matrices read from the master controller. Due to the redundancy of the rotation matrix and the singularity of the Euler angle, the rotation vector is adopted to describe the orientation increment. From Rodrigues's formula [41], ${}^{k-1}_{k}R$ can be converted to a rotation vector, as shown in Equation (3).

$$\begin{cases} \theta_{\Delta} = \arccos\left(0.5 \operatorname{trace} \begin{pmatrix} k-1 \\ k \end{pmatrix} - 0.5 \right) \\ n_{\Delta}^{\wedge} = \binom{k-1 \\ k }{k} R^{-k-1} R^{\mathrm{T}} \right) / (2 \sin \theta_{\Delta}) \\ r_{\Delta} = \theta_{\Delta} n_{\Delta} \end{cases}$$
(3)

where θ_{Δ} and n_{Δ} are the rotation angle and the normalized rotation axis vector respectively corresponding to ${}^{k-1}_{k}R$, r_{Δ} is the rotation vector corresponding to ${}^{k-1}_{k}R$, n_{Δ}^{\wedge} represents the skew symmetric matrix that is associated with the vector n_{Δ} .

With r_{Δ} obtained, another scaling factor k_{ori} is used to map the orientation increment of the master controller, and then the current rotation matrix ${}_{k}^{s}\mathbf{R}$ for the slave manipulator to execute is given in Equation (4) by using Rodrigues's formula, as the orientation command to the slave manipulator.

$$\begin{aligned} {}^{s}_{k} \boldsymbol{R} &= {}^{s}_{k-1} \boldsymbol{R} \cdot \Delta \boldsymbol{R}_{k} \\ &= {}^{s}_{k-1} \boldsymbol{R} \left(\cos(k_{\mathrm{ori}} \theta_{\Delta}) \boldsymbol{I} + (1 - \cos(k_{\mathrm{ori}} \theta_{\Delta})) \boldsymbol{n}_{\Delta} \boldsymbol{n}_{\Delta}^{\mathrm{T}} + \sin(k_{\mathrm{ori}} \theta_{\Delta}) \boldsymbol{n}_{\Delta}^{\wedge} \right) \end{aligned}$$
(4)

where ${}_{k-1}^{s}\mathbf{R}$ is the rotation matrix of the of the last frame and $\Delta \mathbf{R}_{k}$ is the rotation increment to be executed by the manipulator between the last two frames. Up to now, the master–slave mapping for the OSRS, also shown in Figure 2, is introduced.



Figure 2. Master-slave mapping of the OSRS.

3.1. Geometry of Guiding Cone

A scanned model of a volunteer's head (including oral cavity) is shown in Figure 3a. In Figure 3b, several surgical tools are simultaneously inserted into the oral cavity. It can be seen that under the geometric configuration of oral surgery, the obstacle avoidance workspace of the surgical tool is similar to two cones, which is the basis to design a VF.



Figure 3. Geometric configuration: (a) A scanned model; (b) Conical workspace of the tool.

As shown in Figure 4, the guiding cone consists of several conical segments connected together. When the tool is far from the target, the guiding cone is fat and has a larger free space. When the tool is closed to the mouth, the guiding cone becomes thinner, and the motion is gradually constrained. When the tool approaches or reaches the target, the guiding cone shrinks into a cylinder which totally limits the position and orientation of the surgical tool. This design provides a certain free space for manual operation and constrains the undesired motion, which is in accordance with the geometric conditions of oral surgery. Besides, the guiding cone can not only constrain the position of the surgical tool, but also can correct the orientation, which is essential for accurate positioning.



Figure 4. Design of the guiding cone.

The surgical tool is similar to a cylindrical rod, and different tools have different radii. In order to efficiently calculate the position relationship between the tool and the guiding cone, the tool (cylinder) should be simplified into a line segment. To do this, the guiding cone needs to shrink the radius of the tool inward, as shown in Figure 5a. It can be seen that the connection point of two adjacent cones moves backward by ΔL , and the radius of the connection of two adjacent cones shrinks from R_2 to r_2 . The geometric relationship of the shrinkage is shown in Figure 5b by which the parameters ΔL and r_2 can be obtained as shown in Equations (5) and (6). After radius compensation (shrinkage), the surgical instrument is simplified as a line segment, which brings convenience to the calculation of the position relationship described later. In the subsequent discussion on the virtual repulsive force and the effect of the guiding cone (Sections 3.2 and 4), it is assumed that the

end effector is simplified into a line segment and the guiding cone is shrunk inward by the radius of the surgical tool.

$$r_2 = R_2 - r_e \left(\cos \theta_1 + \tan \left(\frac{\theta_1 - \theta_2}{2} \right) \sin \theta_1 \right)$$
(5)

$$\Delta L = r_{\rm e} \sin\left(\frac{\theta_1 + \theta_2}{2}\right) / \cos\left(\frac{\theta_1 - \theta_2}{2}\right) \tag{6}$$

where R_2 and r_2 are the radius of the connection of two adjacent cones, R_2 represents the original radius, r_2 represents the shrunk radius, r_e is the radius of the surgical tool, ΔL is the movement of the connection point of two adjacent cones, and θ_1 and θ_2 are the cone angles of two adjacent cones.



Figure 5. Radius compensation: (a) The shrinkage; (b) The geometric relationship.

3.2. Distance and Repulsive Force

The distance from a point *P* to the guiding cone is key to generate repulsive force and exert adjustment effect. To simplify the definition, the distance is defined by the direction vertical to the cone axis, not the distance from the point to the surface. In other words, the distance is defined on a slice of the guiding cone. The definition is shown in Figure 6, and distance is given in Equations (7) and (8).

$$\lambda = \overrightarrow{O_1 O_2} \cdot \overrightarrow{O_1 P} / \left\| \overrightarrow{O_1 O_2} \right\|^2 \tag{7}$$

$$d_P = r_P - r_{\text{slice}} = \left\| \overrightarrow{O_1 P} - \lambda \overrightarrow{O_1 O_2} \right\| - (r_1 + \lambda (r_2 - r_1))$$
(8)

where λ is the ratio of the slice where point *P* is located to the height of the cone, d_P is the distance point *P* exceeds the guiding cone, r_P represents the length of OP which is the distance from *P* to the slice center *O*, r_{silce} is the radius of the slice, and r_1 and r_2 are the two radii of one segment of the guiding cone.

If the end-effector is beyond the guiding cone, it will be subject to repulsive forces, pushing it back into the guiding cone. The magnitude of the repulsive force is related to the distance (the direction will be discussed in Section 4). The repulsive force can be divided into the free phase, the transition phase, and the maximum repulsive phase according to the distance. For point P in Figure 6, in the free phase, the point is inside the guiding cone, the repulsive force remains 0, and the point P can move freely. In the transition phase, the point starts to exceed the guiding cone, and the repulsive force gradually increases from 0, trying to push the point P back to the guiding cone. In the maximum repulsive phase, the repulsive force reaches the maximum but does not continue to increase to avoid excessive response. The relationship between the magnitude of the repulsive force and the distance is shown in Equation (9).



Figure 6. Definition of the distance from point *P* to the guiding cone.

$$F_{\rm rep} = \begin{cases} 0 & (d_P < 0) \\ 1 - \cos(0.5\pi \, d_P / d_{\rm max}) & (0 \le d_P \le d_{\rm max}) \\ 1 & (d_P > 1) \end{cases}$$
(9)

where F_{rep} is the magnitude of the repulsive force, d_{max} is the distance at which the repulsive force will no longer increase. The repulsive force and its relationship with d_P is also shown in Figure 7.



Figure 7. Repulsive force: (a) Before adjustment; (b) After adjustment; (c) Force–distance profile.

4. Effect of Guiding Cone

4.1. Two-Point Adjustment Model

In the effect of the guiding cone, since the surgical tool is shrunk to a line segment, it can be simplified into two points: a top end *T* and a far end *F*. Repulsive forces will be inflicted on the two endpoints, and then the repulsive forces will be converted to their respective adjustment velocities $v_{\text{far}}^{(\text{adj})}$ and $v_{\text{far}}^{(\text{adj})}$, so as to push the tool back into the guiding cone. The above is the two-point adjustment model, as shown in Figure 8.



Figure 8. Two-Point Adjustment: (a) Two-point force; (b) Repulsive force convert to velocity.

For the repulsive force, after the magnitude is discussed in Section 3.2, the direction will be defined here, as shown in Figure 9. For the repulsive force of the top end *T*, in order not to interfere with the motion along the guiding cone axis O_1O_2 , the direction vector D_{top} is toward the slice center O_T (along vector $\overrightarrow{TO_T}$), and the component along the guiding cone axis O_1O_2 is 0. For the repulsive force at the far end *F* of the surgical tool, its direction vector D_{far} is defined in plane γ that is vertical to the surgical tool (line segment \overrightarrow{FT}), so that the component along the tool direction \overrightarrow{FT} is 0. Following that, the direction of the repulsive force at the far end *F* is defined as the projection vector that vector $\overrightarrow{FO_F}$ projected on plane γ , where vector $\overrightarrow{FO_F}$ is from the far end *F* to the corresponding slice center O_F .



Figure 9. Definition of the direction of the repulsive forces.

Based on the vector projection matrix, the direction of the repulsive force on the two endpoints can be obtained as Equations (10) and (11).

$$D_{\text{top}} = P_{\text{OT}} - P_{\text{T}}$$

= $P_{\text{O1}} + \frac{n \cdot n^{\text{T}}}{n^{\text{T}} \cdot n} \cdot (P_{\text{T}} - P_{\text{O1}}) - P_{\text{T}}$ (10)

where D_{top} is the 3D direction vector of the repulsive force on the top point, P_{OT} , P_{O1} , are the coordinate of points O_T , O_1 ; *n* represents the vector $\overrightarrow{O_1O_2}$.

$$D_{\text{far}} = \left(I - \frac{t \cdot t^{\text{T}}}{t^{\text{T}} \cdot t}\right) \cdot \left(P_{\text{OF}} - P_{\text{F}}\right)$$

= $\left(I - \frac{t \cdot t^{\text{T}}}{t^{\text{T}} \cdot t}\right) \cdot \left(P_{\text{O1}} + \frac{n \cdot n^{\text{T}}}{n^{\text{T}} \cdot n} \cdot \left(P_{\text{F}} - P_{\text{O1}}\right) - P_{\text{F}}\right)$ (11)

where D_{far} is the direction vector of the repulsive force on the far point, *t* represents the tool's vector \overrightarrow{FT} . P_{OF} is the coordinate of point O_{F} . Thus, the repulsive forces F_{top} and F_{far} can be expressed as:

$$F_{\text{top}} = F_{\text{rep}} \cdot \boldsymbol{D}_{\text{top}} / \|\boldsymbol{D}_{\text{top}}\|$$

$$F_{\text{far}} = F_{\text{rep}} \cdot \boldsymbol{D}_{\text{far}} / \|\boldsymbol{D}_{\text{far}}\|$$
(12)

Following that, the adjustment velocities $v_{top}^{(adj)}$ and $v_{far}^{(adj)}$ can be obtained by a gain α that converts the repulsive force to the adjustment velocity:

Furthermore, there is one special situation that needs to be briefly explained. In this situation, the tool crosses two or more guiding cone segments, in other words, the far end *F* is behind the slice where O_1 is located. Under the situation, the point *F* should be deactivated, and take the intersection point of the O_1 slice and the tool to be the new far

end *F*, so as to make the two endpoints subjected to one guiding cone segment and avoid the tool crossing the boundary at the connection of two guide cone segments.

The above is the content of the two-point adjustment model which has the following two advantages. First, since the two endpoints of the line are the most likely to go beyond a cone among the line segment, examining the two endpoints is sufficient and the most efficient way. Second, the orientation of the surgical tool can be effectively adjusted through the two endpoints.

4.2. Velocity Conversion

As the second part of the guiding cone's effect, the velocity conversion is a process that converts the adjustment velocity of the surgical tool obtained in Equation (13) into the joint space of the manipulator which is the plant to directly perform the motion. First, we used the Denavit–Hartenberg method to build the forward kinematics model [42]. The coordinate systems of the joints are shown in Figure 10, where a T-shape connector is used to mount the surgical tool on the mounting flange, and a coordinate system $\{End\}$ is fixed on the top end of the tool. The forward kinematics is expressed in Equation (14).

$${}_{e}^{b}T = {}_{1}^{b}T_{2}^{1}T \cdots {}_{6}^{5}T_{e}^{6}T = \begin{pmatrix} {}_{e}^{b}R & P_{end} \\ O & 1 \end{pmatrix}$$
(14)

where ${}_{e}^{b}T$ is the transformation matrix from the base coordinate system {*Base*} to the end coordinate system {*End*}, ${}_{e}^{b}R$ and P_{end} are the corresponding rotation matrix and position vector of top point of the tool, and ${}_{e}^{6}T$ is the transformation matrix from the the mounting flange to the end point of the tool.



Figure 10. Definition of D-H coordinate systems from joint1 to joint6.

The adjustment velocities $v_{top}^{(adj)}$ and $v_{far}^{(adj)}$ of the surgical tool are described in the base coordinate system {*Base*}. In order to describe the rigid body motion of the tool, transfer $v_{top}^{(adj)}$ and $v_{far}^{(adj)}$ to the end coordinate system {*End*} to get the adjustment velocities ${}^{e}v_{top}^{(adj)}$ and ${}^{e}v_{far}^{(adj)}$ that was described in {*End*}.

$$\begin{cases} {}^{e}\boldsymbol{v}_{far}^{(adj)} = {}^{e}_{b}\boldsymbol{R} \cdot \boldsymbol{v}_{far}^{(adj)} = \left({}^{b}_{e}\boldsymbol{R} \right)^{\mathrm{T}} \cdot \boldsymbol{v}_{far}^{(adj)} \\ {}^{e}\boldsymbol{v}_{top}^{(adj)} = {}^{e}_{b}\boldsymbol{R} \cdot \boldsymbol{v}_{far}^{(adj)} = \left({}^{b}_{e}\boldsymbol{R} \right)^{\mathrm{T}} \cdot \boldsymbol{v}_{top}^{(adj)} \end{cases}$$
(15)

Next, the adjustment angular velocity of the rigid body motion can be expressed as:

$$\begin{cases} {}^{e}\omega_{x}^{(adj)} = 0 \\ {}^{e}\omega_{y}^{(adj)} = \left({}^{e}v_{far,z}^{(adj)} - {}^{e}v_{top,z}^{(adj)} \right) / d \\ {}^{e}\omega_{z}^{(adj)} = - \left({}^{e}v_{far,y}^{(adj)} - {}^{e}v_{top,y}^{(adj)} \right) / d \end{cases}$$
(16)

where ${}^{e}\omega_{x}^{(adj)}$, ${}^{e}\omega_{y}^{(adj)}$, ${}^{e}\omega_{z}^{(adj)}$ are the *x*, *y*, and *z* components, respectively, of adjustment angular velocity described in {*End*}, and *d* is the tool length shown in Figure 10. Then, the 6-dimension velocity vector ${}^{e}V_{adj}$ of the tool described in {*End*} can be written as:

$${}^{e}V_{adj} = \left({}^{e}v_{top}^{(adj)}, {}^{e}\omega_{x}^{(adj)}, {}^{e}\omega_{y}^{(adj)}, {}^{e}\omega_{z}^{(adj)}\right)^{\mathrm{T}}$$

$$= \left({}^{e}v_{x}^{(adj)}, {}^{e}v_{y}^{(adj)}, {}^{e}v_{z}^{(adj)}, {}^{e}\omega_{x}^{(adj)}, {}^{e}\omega_{y}^{(adj)}, {}^{e}\omega_{z}^{(adj)}\right)^{\mathrm{T}}$$

$$(17)$$

where ${}^{e}v_{x}^{(adj)}$, ${}^{e}v_{y}^{(adj)}$, ${}^{e}v_{z}^{(adj)}$ are the *x*, *y*, and *z* components, respectively, of the adjustment velocity ${}^{e}v_{top}^{(adj)}$ of the top end on the tool described in {*End*}.

In order to convert the 6-dimension velocity vector ${}^{e}V_{adj}$ of the tool into the joint space of the manipulator, the Jacobian matrix of the manipulator should be involved, which can be obtained from the forward kinematics result in Equation (14):

$${}^{b}J = \begin{pmatrix} \frac{\partial p_{end}}{\partial \theta_{1}} & \frac{\partial p_{end}}{\partial \theta_{2}} & \cdots & \frac{\partial p_{end}}{\partial \theta_{6}} \\ \frac{\partial r_{3}}{\partial \theta_{1}} r_{2}^{T} & \frac{\partial r_{3}}{\partial \theta_{2}} r_{2}^{T} & \cdots & \frac{\partial r_{3}}{\partial \theta_{6}} r_{2}^{T} \\ \frac{\partial r_{1}}{\partial \theta_{1}} r_{3}^{T} & \frac{\partial r_{1}}{\partial \theta_{2}} r_{3}^{T} & \cdots & \frac{\partial r_{1}}{\partial \theta_{6}} r_{3}^{T} \\ \frac{\partial r_{2}}{\partial \theta_{1}} r_{1}^{T} & \frac{\partial r_{2}}{\partial \theta_{2}} r_{1}^{T} & \cdots & \frac{\partial r_{2}}{\partial \theta_{6}} r_{1}^{T} \end{pmatrix}$$
(18)

where ^b*J* represents the 6 × 6 Jacobian matrix of the manipulator described in {*Base*}; θ_1 , θ_2 , . . . , θ_6 are the six joint angles of the manipulator; r_1 , r_2 , r_3 are the 1 × 3 row vectors that respectively represents the line 1, 2, 3 of the rotation matrix ${}_e^b R$. Then, transfer the ^b*J* into the end coordinate system {*End*} as shown in Equation (19):

$${}^{e}J = \begin{pmatrix} {}^{b}R & O \\ O & {}^{b}R \end{pmatrix} \cdot {}^{b}J$$
(19)

where ^e*J* represents the 6 × 6 Jacobian matrix described in {*End*}. After that, the 6 × 1 adjustment velocity vector ${}^{e}V_{adj}$ can be converted to the joint adjustment velocity vector $\dot{\theta}_{adj}$ as shown in Equation (20).

$$\dot{\boldsymbol{\theta}}_{\mathrm{adj}} = {}^{\mathrm{e}}\boldsymbol{J}^{\mathrm{+}} \cdot {}^{\mathrm{e}}\boldsymbol{V}_{\mathrm{adj}} \tag{20}$$

where ${}^{e}J^{+}$ is the pseudo inverse matrix of ${}^{e}J$. Finally, the joint angles θ_{exe} , which should be executed by the manipulator, can be derived as follows:

$$\boldsymbol{\theta}_{\text{exe}} = \boldsymbol{\theta}_{\text{cmd}} + \int_0^t \dot{\boldsymbol{\theta}}_{\text{adj}} \, dt \tag{21}$$

where θ_{cmd} represents the joint angle command from the master–slave operation. Thus, the effect of the guiding cone is established, and the diagram of the adjustment effect is shown in Figure 11. From Figure 11, it can be seen that design of the guiding cone and the adjustment effect form the complete proposed virtual fixture framework.



Figure 11. Adjustment effect and the complete virtual fixture framework.

5. System Integration of Guiding Cone

5.1. Calibration and Correction

In OSRS, a binocular vision system and a mouth opener are involved to locate the oral cavity in real time, thereby the guiding cone can be corrected to suit a dynamic target. Since the position and orientation of the detected mouth opener are described in the camera's coordinate system, hand–eye calibration, as the approach for obtaining the transformation between the manipulator and camera is responsible for obtaining the target's description in the manipulator's coordinate system.

In hand–eye calibration, a checkerboard is fixed on the end-effector, shown in Figure 12. To calculate the transformation from {Base} to {Camera}, the manipulator should be driven to at least two different positions. Given that the transformation from end-effector to checkerboard remains unchanged regardless of how the manipulator moves, the relation between transformation matrices can be expressed as Equation (22).

$$\begin{pmatrix} {}^{\mathbf{b}}_{\mathbf{c}} \boldsymbol{T}^{(2)}{}^{\mathbf{b}}_{\mathbf{c}} \boldsymbol{T}^{(1)} \end{pmatrix} \cdot {}^{\mathbf{b}}_{\mathbf{c}} \boldsymbol{T} = {}^{\mathbf{b}}_{\mathbf{c}} \boldsymbol{T} \cdot \begin{pmatrix} {}^{\mathbf{c}}_{\mathbf{ch}} \boldsymbol{T}^{(2)}{}^{\mathbf{ch}}_{\mathbf{c}} \boldsymbol{T}^{(1)} \end{pmatrix}$$
(22)

where ${}_{e}^{b}T^{(1)}$ is the transformation from {*Base*} to {*End*1}. ${}_{c}^{b}T$ is the transformation from {*Base*} to {*Camera*} to be solved in hand–eye calibration. ${}_{c}^{c}T^{(1)}$ is the transformation from {*Checkerboard*1} to {*Camera*}. Then, ${}_{e}^{b}T^{(2)}{}_{b}^{e}T^{(1)}$ and ${}_{ch}^{c}T^{(2)}{}_{c}^{c}T^{(1)}$ can be written as T_{A} and T_{B} , respectively, with rotational partitions R_{A} , R_{B} and translational partitions t_{A} , t_{B} assembled in them. Thereby, the calibration result can be obtained as follows.

$$\begin{pmatrix}
\begin{pmatrix}
R_{A} & O \\
R_{A} \\
O & R_{A}
\end{pmatrix} - K \begin{pmatrix}
R_{B}^{1} & O \\
R_{B}^{T} \\
O & R_{B}^{T}
\end{pmatrix} K r_{X} = 0$$

$$t_{X} = (R_{A} - I)^{-1}(R_{X}t_{B} - t_{A})$$
(23)

where R_X and t_X are, respectively, rotational and translational partitions of ${}_c^b T$, and $r_X = (r_{11}, r_{21}, r_{31}, \dots, r_{23}, r_{33})^T$ is the vectorized expression of R_X and can be calculated by using SVD decomposition. *K* is a 9 × 9 constant matrix with elements in column 1, 4, 7, 2, 5, 8, 3, 6, 9 of line 1 to 9, respectively, are one, and all other elements are zero.

The guiding cone can be determined by "anchor-arrow" description: an anchor point and an arrow (3D vector) represent the position and orientation, respectively. The "anchorarrow" description is preoperatively defined under the mouth opener's local coordinate system whose transformation to the camera's coordinate system can be obtained from the vision system. To convert the "anchor-arrow" description from the mouth opener to the manipulator, the hand–eye calibration result is necessary. The correction of the guiding cone is shown in Figure 13, Equations (24) and (25).



Figure 12. Hand-eye calibration and its coordinate systems.

$$\begin{cases} {}^{b}P_{an} = {}^{b}_{c}T \cdot {}^{c}_{op}T \cdot {}^{op}P_{an} \\ {}^{b}D_{ar} = {}^{b}_{c}T \cdot {}^{op}_{op}T \cdot {}^{op}D_{ar} \end{cases}$$
(24)

$${}^{\mathbf{b}}\boldsymbol{P}_{\mathbf{o}} = {}^{\mathbf{b}}\boldsymbol{P}_{\mathbf{a}\mathbf{n}} + {}^{\mathbf{b}}\boldsymbol{D}_{\mathbf{a}\mathbf{r}} \cdot \boldsymbol{l}_{\mathbf{a}\mathbf{n}} / \left\| {}^{\mathbf{b}}\boldsymbol{D}_{\mathbf{a}\mathbf{r}} \right\|$$
(25)

where ${}^{b}P_{an}$ and ${}^{op}P_{an}$ are the positions of the anchor point described in {*Base*} and {*Opener*}, respectively, and ${}^{b}D_{an}$ and ${}^{op}D_{an}$ are the direction vectors of the arrow under {*Base*} and {*Opener*}, respectively. ${}^{b}_{c}T$ is the hand–eye calibration result. ${}^{c}_{op}T$ is the transformation from {*Camera*} to {*Opener*} obtained from the vision system. ${}^{b}P_{o}$ is an arbitrary slice's center in Figure 6 under {*Base*}, and l_{an} is the distance between a slice's center and the anchor point. With the "anchor-arrow" description, a guiding cone can be fully defined when the radii and distances of all the slices are given.



Figure 13. Correction of guiding cone can be determined by an "anchor-arrow" description.

5.2. Automatic Feeding Mode

If no obstacles are inside the guiding cone, an optional automatic mode may be preferable. In the automatic mode, the master–slave motion command is replaced by position and orientation interpolation, and then the guiding cone can adjust the interpolated motion command to ensure safety. The intermediate points can be expressed as follows, in which the position is directly interpolated, and the orientation is smoothly interpolated by using the rotation vector method:

$$\begin{cases} {}^{\mathbf{b}}\boldsymbol{P}_{t} = {}^{\mathbf{b}}\boldsymbol{P}_{\text{init}} + u(t) \left({}^{\mathbf{b}}\boldsymbol{P}_{\text{aim}} - {}^{\mathbf{b}}\boldsymbol{P}_{\text{init}} \right) \\ {}^{\mathbf{b}}\boldsymbol{R}_{t} = {}^{\mathbf{b}}\boldsymbol{R}_{\text{init}} \cdot mat \left(u(t) \cdot vec \left({}^{\mathbf{b}}\boldsymbol{R}_{\text{init}}^{-1} \cdot {}^{\mathbf{b}}\boldsymbol{R}_{\text{aim}} \right) \right) \end{cases}$$
(26)

where ${}^{b}P_{t}$, ${}^{b}P_{init}$, and ${}^{b}P_{aim}$ are the end-effector's positions of time *t*, time 0, and the destination, respectively, under {*Base*}. ${}^{b}R_{t}$, ${}^{b}R_{init}$, and ${}^{b}R_{aim}$ are corresponding destination's orientations. *mat* and *vec* are conversion functions between rotation matrix and rotation vector. u(t) is a normalized time factor as well as a monotonic function whose variable is time *t*, when t = 0, u(t) = 0, and when $t \ge end$, u(t) = 1.

Meanwhile, with the mouth opener, ${}^{b}P_{aim}$ and ${}^{b}R_{aim}$ are refreshed in real time through ${}^{b}_{c}T$ and ${}^{c}_{op}T$ obtained from the vision system, as shown in Equation (27).

$${}^{b}T_{aim} = {}^{b}_{c}T \cdot {}^{c}_{op}T \cdot {}^{op}T_{aim}$$
⁽²⁷⁾

where ${}^{b}T_{aim}$ is assembled by ${}^{b}P_{aim}$ and ${}^{b}R_{aim}$, and ${}^{op}T_{aim}$ is the end-effector's destination under {*Opener*}. In fact, ${}^{op}T_{aim}$ is similar to ${}^{op}P_{an}$ and ${}^{op}D_{an}$, and all of them coincide with each other. The difference is that ${}^{op}T_{aim}$ describes the end-effector, while ${}^{op}P_{an}$ and ${}^{op}D_{an}$ form the "anchor-arrow" to describe the guiding cone.

By now, the system is integrated, forming a complete motion strategy. To sum up, the new conical virtual fixture is established with geometry design and effect mode, and then combined with the vision system to build the overall strategy. Moreover, an optional automatic mode was provided for users. The full strategy is shown in Figure 14.



Figure 14. The overall motion strategy: (a) The mainline; (b) The detailed items.

6. Simulations, Experiments, and Discussion

6.1. Simulations

To verify the feasibility of the proposed virtual fixture, simulations are carried out. To begin with, the virtual fixture consists of four cone segments, whose parameter setups are listed in Table 1, where α is the gain in Equation (13) and d_{max} is the maximum repulsive force layer in Equation (9).

Cone	O_1	<i>O</i> ₂	r ₁	r ₂	α
1	(900, -550, 620)	(1100, -520, 570)	250 mm	50 mm	150
2	(1100, -520, 570)	(1200, -505, 545)	50 mm	10 mm	150
3	(1200, -505, 545)	(1300, -490, 520)	10 mm	1 mm	150
4	(1300, -490, 520)	(1400, -475, 495)	1 mm	1 mm	150

Table 1. Parameter setups of the virtual fixture.

The first simulation group is "static and manual" feeding, that is, the guiding cone is fixed, and the motion is generated by master–slave motion. This group is to verify the adjustment effect, in other words, to check whether the guiding cone can play its role. The results are shown in Figure 15, including the motion process (see Figure 15a,b) and the joint space's angle (see Figure 15c). The results of the virtual fixture are activated and deactivated (only master–slave command is allowed) are displayed, forming a comparison.

*d*_{max} 5 mm 5 mm 5 mm

5 mm

14 of 22



Figure 15. Simulation of "static and manual" feeding: (**a**) Motion process when VF is activated, and the target is reached; (**b**) Motion process when VF is deactivated, which is pure master–slave motion, and the target is missed; (**c**) Angle–time curves of joint space.

The second simulation group is "dynamic and manual" feeding. In this group, position and orientation of the guiding cone are changing gradually, representing the disturbance of the oral cavity. The original motion command is generated by master–slave operation. When time is 0 s, under the "anchor-arrow" description, the guiding cone's anchor point is $(1400, -550, 620)^{T}$ and the arrow vector is $(-1, 0, 0)^{T}$. When time is 7.5 s, the anchor point is $(1400, -475, 495)^{T}$ and the arrow vector is $(-0.96, -0.144, 0.24)^{T}$. The motion process of the tool subjected to a dynamic guiding cone is displayed in Figure 16, from which it can be seen that the surgical tool can still track the guiding cone well and hit the target, although the guiding cone is ever-moving.

In Figure 16, it takes approximately 18 s for the surgical tool to reach the target, and then keeps following the target in the remaining time. Next, the third simulation group is the "static and automatic" feeding, in which the path is automatically interpolated instead of the master–slave operation, and the guiding cone is fixed. The simulation result is shown in Figure 17. Because direct path interpolation (see Figure 17b) cannot ensure safety, the guiding cone was adopted, as shown in Figure 17a, and the path can be limited inside the guiding cone, eliminating almost all the out-of-range motion that may collide with obstacles. Moreover, from Figure 17c, it can be seen that the trajectory of automatic mode is more smooth than manual operation, making this mode more preferable to some extent.

In addition, the fourth simulation group is the "dynamic and automatic" feeding. However, all else being equal, the simulated process is just the same as Figure 16 that can sufficiently represent the "dynamic and automatic" feeding process. Therefore, the figure to show this group can be omitted.



Figure 16. Simulation of "dynamic and manual" feeding. Despite the guiding cone moving, the tool can still track and hit the dynamic target successfully. Additionally, the result of simulation on "dynamic and automatic" feeding is just the same.



Figure 17. Simulation of "static and automatic" feeding: (**a**) Motion process when path interpolation and VF adjustment are combined; (**b**) Motion process when VF is deactivated, which is pure path interpolation, and the VF is exceed; (**c**) Angle–time curves of joint space.

6.2. Experiments

Following the simulations, experiments on an oral model were conducted. Different from the simulations in which only the virtual fixture is concerned, the experiments involve the full strategy where the virtual fixture and vision system are combined. The setup of the experiments is shown in Figure 18. The user grasps the master controller, while another hand shakes the oral model to generate the disturbance motion, with a mouth opener worn on the oral model. The anchor-arrow description of the target is preoperatively defined. The camera calibration and hand–eye calibration is all done. The virtual fixture, as well as the full strategy, is programmed with C++ and runs on an industrial computer with Intel(R) Core(TM) i9-12900H at 2.5 GHz and 32 GB RAM in a 64bit win10 system. The parameters of the virtual fixture are the same as Table 1.



Figure 18. The experimental setup.

Experiments are conducted with the following three groups: static and manual feeding, dynamic and manual feeding, and dynamic and automatic feeding. The motion process of the three groups is shown in Figure 19a,b and Figure 20, respectively.

In Figure 19a, it takes about 22 s for the surgical tool to reach the target. In Figure 19b, it takes about 24 s to reach the target, and then follows the target in the remaining time. In Figure 20, it takes approximately 20 s to reach the target and then follow it. From the motion processes of the three groups, it can be seen that the motion strategy can successfully guide the surgical tool into the target position inside the oral cavity. Even if the oral cavity is ever-moving, the surgical tool can still hit the target and actively follow the disturbance motion of the oral model, illustrating the effectiveness of the virtual fixture and the motion strategy that is combined with the vision system.

6.3. Discussion

To further discuss the results, the detailed data of the experiments and simulations were extracted and visualized. The motion errors of some groups, which are the distance between the target under anchor-arrow description (also the anchor-arrow description of the guiding cone) and the end-effector, are shown in Figure 21. The disturbances of the target during dynamics groups (shaken by hand) are shown in Figure 22.



Figure 19. The motion process of manual operation: (a) Static and manual feeding. (b) Dynamic and manual feeding.

From Figure 21a,c, it can be seen that the motion process can be divided into two stages: the feeding stage and the following stage. During the feeding stage, the surgical tool continuously moves forward, and the error declines gradually. During the following stage, the surgical tool reaches the target and remains to keep up with it, making the error close to zero as shown in Figure 21b,d, which are the local magnified figure of the following stage. Then, the average positioning errors in the following stage, including position and orientation, can be obtained as shown in Table 2, reflecting the extent of accuracy of the motion strategy.



Figure 20. The motion process of dynamic and automatic feeding.



Figure 21. The errors between surgical tool and target during some groups of experiments or simulations: (a) Position errors of overall process. (b) Local enlargement of position errors during the following stage. (c) Orientation errors of the overall process. (d) Local enlargement of orientation errors during the following stage.



Figure 22. Visualization of the dynamic disturbance motion of the target, also how the oral model is shaken: (a) The group of experiment with manual operation and dynamic target. (b) The group of experiment with automatic operation and dynamic target. (c) The group of simulation with manual operation and dynamic target.

Table 2. Average positioning error of different groups.

Туре	Operation	Target	Average Positioning Error
Experiment	Manual	Static	0.366 mm and 0.227 deg
Experiment	Manual	Dynamic	0.977 mm and 1.017 deg
Experiment	Automatic	Dynamic	0.912 mm and 0.677 deg
Simulation	Manual	Static	0.202 mm and 0.082 deg
Simulation	Manual	Dynamic	0.439 mm and 0.289 deg

From Table 2, it can be seen that the errors of experiments are obviously larger than that of simulations where the virtual fixture is stand-alone without connecting to the vision system. The extra error between simulations and experiments is induced by the measurement noise from the vision system, which is generally larger when the oral model is dynamic and smaller when the oral model is static. In other words, the guiding cone (or the target) is ever-swinging by the measurement noise during the task, making errors all the time, despite a filter being applied to alleviate the noise.

Besides, there are additional errors between static feeding and dynamic feeding, which are caused by the random disturbances of the oral model. The disturbances are displayed in Figure 22. To follow disturbance motion, a lag is inevitable since it must take a small time interval for the guiding cone to exert its adjusting effect. Therefore, the error is triggered. Relatively more noise of vision measure also contributes to the additional errors between static feeding and dynamic feeding. Moreover, compared with manual operation, automatic mode will make relatively fewer orientation errors (angle error), in that automatic trajectory generation will bring more targeted origin commands than that of manual operation.

In addition, what must be declared here is that error is only defined by the distance between the target read from the vision measurement and the surgical tool. The error is just on behalf of the guiding cone, not the absolute error, which is influenced by hand–eye calibration and accuracy of the vision system. Therefore, the above accuracy data are only to evaluate the virtual fixture itself. If the absolute error is to be obtained, a third-party measurement equipment might be required.

Despite a few errors that still exist, the accuracy of the proposed virtual fixture is sufficient anyway, given that the average following error is less than 1.0 mm and 1.05 deg during a dynamic process in which the oral model is intentionally shaken, as shown in Figure 22a,b. Because the oral disturbance is usually less than the intentional shaking, the error will be less in a real task. The results show that the proposed virtual fixture, as well as the motion strategy, is an effective way to execute the feeding task and fulfill the

requirement of oral surgery. To sum up, the position and orientation accuracy can be well guaranteed with the help of the proposed strategy, no matter whether the oral cavity is static or moving. Therefore, the method can simplify a skill-demanding task into an easy task, eliminating the concern of feeding to a wrong position or misoperation. Moreover, the trait of dynamic compensation has the potential to reduce the extra damage to the oral tissue during the surgery.

7. Conclusions and Future Work

This paper introduces a guiding and positioning strategy for robot-assisted oral surgery. The work is conducted during geometry design, effect design, system integration, simulations, and experiments. Conclusions can be listed as follows:

- 1. A new conical virtual fixture, also called the guiding cone, is designed according to the structure of the oral cavity, making it suitable for robot-assisted oral surgery.
- 2. The effect mode of the virtual fixture is established through two-point adjustment and velocity conversion, which can actively adjust the position and orientation of the surgical tool simultaneously.
- 3. The virtual fixture is integrated with a vision system to compensate for the disturbance of the oral cavity. The system calibration is done, the master–slave mapping is combined, and an automatic mode is adopted, forming a complete motion strategy.
- 4. Simulations and experiments are carried out in the work, the targets are reached, and the errors are quantitatively estimated. In simulations, the estimated positioning errors are 0.202 mm and 0.082 deg for a static target, and 0.439 mm and 0.289 deg for a dynamic target, representing the theoretical ability of VF to adjust the surgical tool. In experiments, the estimated positioning errors are 0.366 mm and 0.227 deg for a static target and 0.977 mm and 1.017 deg for a dynamic target. Despite a few errors, the results suggest the effectiveness of the proposed VF and the motion strategy, which can help the operator keep a relatively high level of accuracy during the manipulating process.
- 5. Safety, accuracy, and dynamic adaptability are three features of the proposed method, making it a palatable auxiliary framework to assist oral surgery operations.

In the future, the motion error will be further studied with the help of precision measurement in order to archive better accuracy. Additionally, force sensing and force feedback methods will be introduced into the current framework to further improve the performance. In addition, the programming of this work will be integrated into a 3D-visualization software for the oral surgical robot system to make it more convenient.

Author Contributions: Conceptualization, W.W., Y.W. (Yan Wang) and Y.C.; software, Y.W. (Yan Wang) and Y.H.; simulation, Y.W. (Yan Wang), S.W. and Q.Z.; experiment, Y.W. (Yan Wang), Q.Z. and S.W.; investigation, W.W. and Y.C.; writing, Y.W. (Yan Wang) and Y.C.; review, W.W. and Y.W. (Yuyang Wang). All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Key R&D Program of China, grant number 2020YFB1312800.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

- 1. Shanti, R.M.; O'Malley, B.W. Surgical Management of Oral Cancer. Dent. Clin. N. Am. 2018, 62, 77–86. [CrossRef] [PubMed]
- Johnston, L.; Warrilow, L.; Fullwood, I.; Tanday, A. Fifteen-minute consultation: Oral ulceration in children. Arch. Dis. Chidhood-Educ. Pract. Ed. 2021, 107, 257–264. [CrossRef] [PubMed]
- Carvalho, R.; Botelho, J.; Machado, V.; Mascarenhas, P.; Alcoforado, G.; Mendes, J.; Chambrone, L. Predictors of tooth loss during long-term periodontal maintenance: An updated systematic review. J. Clin. Periodontol. 2021, 48, 1019–1036. [CrossRef] [PubMed]

- 4. Kudsi, Z.; Fenlon, M.; Johal, A.; Baysan, A. Assessment of Psychological Disturbance in Patients with Tooth Loss: A Systematic Review of Assessment Tools. *J. Prosthodont. Implant Esthet. Reconstr. Dent.* **2020**, *29*, 193–200. [CrossRef] [PubMed]
- Hosadurga, R.; Soe, H.; Lim, A.; Adl, A.; Mathew, M. Association between tooth loss and hypertension: A cross-sectional study. J. Fam. Med. Prim. Care 2020, 9, 925–932. [CrossRef]
- McMahon, J.; Steele, P.; Kyzas, P.; Pollard, C.; Jampana, R.; MacIver, C.; Subramaniam, S.; Devine, J.; Wales, C.; McCaul, J. Operative tactics in floor of mouth and tongue cancer resection—The importance of imaging and planning. *Br. J. Oral Maxillofac. Surg.* 2021, 59, 5–15. [CrossRef]
- 7. Flanagan, D. Rationale for Mini Dental Implant Treatment. J. Oral Implantol. 2021, 47, 437–444. [CrossRef]
- 8. Hupp, J. Robotics and Oral-Maxillofacial Surgery. J. Oral Maxillofac. Surg. 2020, 78, 493–495. [CrossRef]
- 9. Yee, S. Transoral Robotic Surgery. Aorn J. 2017, 105, 73-81. [CrossRef]
- 10. Ahmad, P.; Alam, M.K.; Aldajani, A.; Alahmari, A.; Alanazi, A.; Stoddart, M.; Sghaireen, M.G. Dental Robotics: A Disruptive Technology. *Sensors* 2021, *21*, 3308. [CrossRef]
- Zhang, J.; Wang, W.; Cai, Y.; Li, J.; Zeng, Y.; Chen, L.; Yuan, F.; Ji, Z.; Wang, Y.; Wyrwa, J. A Novel Single-Arm Stapling Robot for Oral and Maxillofacial Surgery-Design and Verification. *IEEE Robot. Autom. Lett.* 2022, 7, 1348–1355. [CrossRef]
- 12. Yuan, F.; Liang., S.; Lyu., P. A Novel Method for Adjusting the Taper and Adaption of Automatic Tooth Preparations with a High-Power Femtosecond Laser. J. Clin. Med. 2021, 10, 3389. [CrossRef]
- Gao, Z.; Zhu, M.; Yu, J. A Novel Camera Calibration Pattern Robust to Incomplete Pattern Projection. *IEEE Sens. J.* 2021, 21, 10051–10060. [CrossRef]
- 14. Veras, L.; Medeiros, F.; Guimaraes, L. Rapidly exploring Random Tree* with a sampling method based on Sukharev grids and convex vertices of safety hulls of obstacles. *Int. J. Adv. Robot. Syst.* **2019**, *16*, 1729881419825941. [CrossRef]
- 15. Machmudah, A.; Parman, S.; Zainuddin, A.; Chacko, S. Polynomial joint angle arm robot motion planning in complex geometrical obstacles. *Appl. Soft Comput.* **2013**, *13*, 1099–1109. [CrossRef]
- 16. Kim, J.H.; Kim, K.B.; Kim, S.H.; Kim, W.C.; Kim., H.Y.; Kim, J.H. Quantitative evaluation of common errors in digital impression obtained by using an LED blue light in-office CAD/CAM system. *Quintessence Int.* **2015**, *46*, 401–407. [CrossRef]
- 17. Chen, Z.; Su, W.; Li, B.; Deng, B.; Wu, H.; Liu, B. An intermediate point obstacle avoidance algorithm for serial robot. *Adv. Mech. Eng.* **2018**, *10*, 1687814018774627. [CrossRef]
- Ma, Q.; Kobayashi, E.; Suenaga, H.; Hara, K.; Wang, J.; Nakagawa, K.; Sakuma, I.; Masamune, K. Autonomous Surgical Robot With Camera-Based Markerless Navigation for Oral and Maxillofacial Surgery. *IEEE-ASME Trans. Mechatron.* 2020, 25, 1084–1094. [CrossRef]
- Hu, Y.; Li, J.; Chen, Y.; Wang, Q.; Chi, C.; Zhang, H.; Gao, Q.; Lan, Y.; Li, Z.; Mu, Z.; et al. Design and Control of a Highly Redundant Rigid-flexible Coupling Robot to Assist the COVID-19 Oropharyngeal-Swab Sampling. *IEEE Robot. Autom. Lett.* 2022, 7, 1856–1863. [CrossRef]
- Cheng, K.; Kan, T.; Liu, Y.; Zhu, W.; Zhu, F.; Wang, W.; Jiang, X.; Dong, X. Accuracy of dental implant surgery with robotic position feedback and registration algorithm: An in-vitro study. *Comput. Biol. Med.* 2021, 129, 104153. [CrossRef]
- Toosi, A.; Arbabtafti, M.; Richardson, B. Virtual Reality Haptic Simulation of Root Canal Therapy. *Appl. Mech. Mater.* 2014, 66, 388–392. [CrossRef]
- Kasahara, Y.; Kawana, H.; Usuda, S.; Ohnishi, K. Telerobotic-assisted bone-drilling system using bilateral control with feed operation scaling and cutting force scaling. *Int. J. Med. Robot. Comput. Assist. Surg.* 2012, *8*, 221–229. [CrossRef] [PubMed]
- Iijima, T.; Matsunaga, T.; Shimono, T.; Ohnishi, K.; Usuda, S.; Kawana, H. Development of a Multi DOF Haptic Robot for Dentistry and Oral Surgery. In Proceedings of the 2020 IEEE/SICE International Symposium on System Integration, Honolulu, HI, USA, 12–15 January 2020; pp. 52–57.
- Li, J.; Lam, W.; Hsung, R.; Pow, E.; Wu, C.; Wang, Z. Control and Motion Scaling of a Compact Cable-driven Dental Robotic Manipulator. In Proceedings of the 2019 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, Hong Kong, China, 8–12 July 2019; pp. 1002–1007.
- Kwon, Y.; Tae, K.; Yi, B. Suspension laryngoscopy using a curved-frame trans-oral robotic system. *Int. J. Comput. Assist. Radiol.* Surg. 2014, 9, 535–540. [CrossRef] [PubMed]
- Rosenberg, L. Virtual fixtures: Perceptual tools for telerobotic manipulation. In Proceedings of the IEEE Virtual Reality Annual International Symposium, Seattle, WA, USA, 18–22 September 1993; pp. 76–82.
- 27. Abbott, J.; Marayong, P.; Okamura, A. Haptic virtual fixtures for robot-assisted manipulation. In Proceedings of the 12th International Symposium on Robotics Research, San Francisco, CA, USA, 12–15 October 2005; pp. 49–64.
- Abbott, J.; Okamura, A. Stable forbidden-region virtual fixtures for bilateral telemanipulation. J. Dyn. Syst. Meas. Control-Trans. ASME 2006, 128, 53–64. [CrossRef]
- 29. Bettini, A.; Marayong, P.; Lang, S.; Okamura, A.; Hager, G. Vision-assisted control for manipulation using virtual fixtures. *IEEE Trans. Robot. Autom.* 2004, 20, 953–966. [CrossRef]
- Abbott, J.; Okamura, A. Pseudo-admittance bilateral telemanipulation with guidance virtual fixtures. Int. J. Robot. Res. 2007, 26, 865–884.
 [CrossRef]
- Kikuuwe, R.; Takesue, N.; Fujimoto, H. A control framework to generate nonenergy-storing virtual fixtures: Use of simulated plasticity. *IEEE Trans. Robot.* 2008, 24, 781–793. [CrossRef]

- 32. Lin, A.; Tang, Y.; Gan, M.; Huang, L.; Kuang, S.; Sun, L. A Virtual Fixtures Control Method of Surgical Robot Based on Human Arm Kinematics Model. *IEEE Access* 2019, 7, 135656–135664. [CrossRef]
- Bazzi, D.; Roveda, F.; Zanchettin, A.; Rocco, P. A Unified Approach for Virtual Fixtures and Goal-Driven Variable Admittance Control in Manual Guidance Applications. *IEEE Robot. Autom. Lett.* 2021, *6*, 6378–6385. [CrossRef]
- Tang, A.; Cao, Q.; Pan, T. Spatial motion constraints for a minimally invasive surgical robot using customizable virtual fixtures. *Int. J. Med. Robot. Comput. Assist. Surg.* 2014, 10, 447–460. [CrossRef]
- Pruks, V.; Farkhatdinov, I.; Ryu, J. Preliminary Study on Real-Time Interactive Virtual Fixture Generation Method for Shared Teleoperation in Unstructured Environments. In Proceedings of the 11th International Conference on Haptics—Science, Technology, and Applications, Pisa, Italy, 13–16 June 2018; pp. 648–659.
- 36. Marinho, M.; Adorno, B.; Harada, K.; Mitsuishi, M. Dynamic Active Constraints for Surgical Robots Using Vector-Field Inequalities. *IEEE Trans. Robot.* 2019, *35*, 1166–1185. [CrossRef]
- 37. Xu, C.; Lin, L.; Aung, Z.; Chai, G.; Xie, L. Research on spatial motion safety constraints and cooperative control of robot-assisted craniotomy: Beagle model experiment verification. *Int. J. Med. Robot. Comput. Assist. Surg.* **2021**, 17, e2231. [CrossRef]
- Li, M.; Ishii, M.; Taylor, R. Spatial Motion Constraints Using Virtual Fixtures Generated by Anatomy. *IEEE Trans. Robot.* 2007, 17, 4–19. [CrossRef]
- Ren, J.; Patel, R.; McIsaac, K.; Guiraudon, G.; Peters, T. Dynamic 3-D virtual fixtures for minimally invasive beating heart procedures. *IEEE Trans. Med. Imaging* 2008, 27, 1061–1070. [CrossRef]
- 40. He, C.; Yang, E.; McIsaac, K.; Patel, N.; Ebrahimi, A.; Shahbazi, M.; Gehlbach, P.; Iordachita, I. Automatic Light Pipe Actuating System for Bimanual Robot-Assisted Retinal Surgery. *IEEE Trans. Mechatron.* **2020**, *25*, 2846–2857. [CrossRef]
- 41. Valdenebro, A. Visualizing rotations and composition of rotations with the Rodrigues vector. *Eur. J. Phys.* **2016**, *37*, 065001. [CrossRef]
- 42. Shi, H.; Liu, Q.; Mei, X. Visualizing rotations and composition of rotations with the Rodrigues vector. *Machines* **2021**, *9*, 213. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.