

# Efficient Bimanual Symmetric 3D Manipulation for Markerless Hand-Tracking

Markus Schlattmann, Reinhard Klein

University of Bonn, Institute of Computer Science II – Computer Graphics  
D-53117 Bonn, Germany  
{markus, rk}@cs.uni-bonn.de

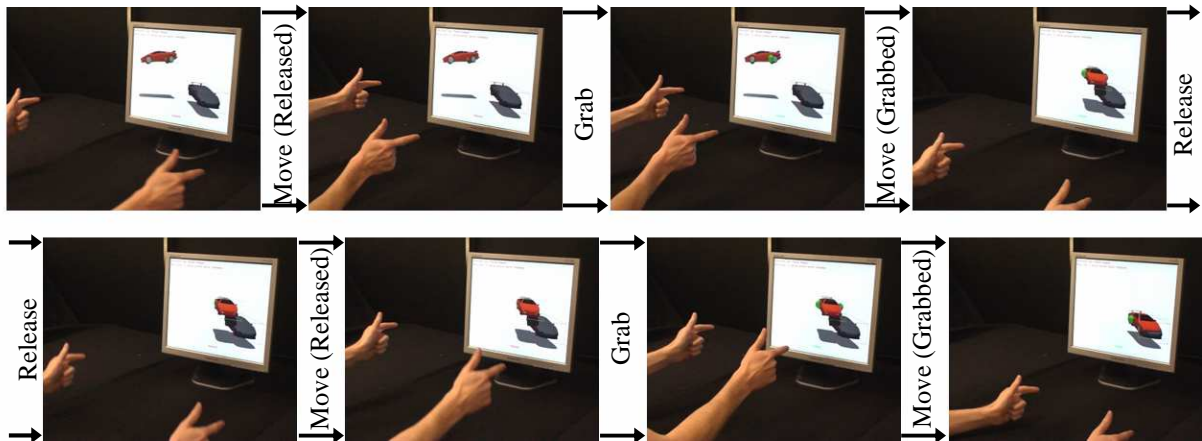


Figure 1: Image sequence of our demo system. Based on our bimanual symmetric interaction technique a virtual object (red car) can be grabbed (move hands together), manipulated (as if gripped between hands) and released (move hands apart).

## Abstract

Recently, stable markerless 6 DOF video based hand-tracking devices became available. These devices simultaneously track the positions and orientations of the user's both hands in different postures with at least 25 frames per second. Such hand-tracking allows for using the human hands as natural input devices. However, the absence of physical buttons for performing click actions and state changes poses severe challenges in designing an efficient and easy to use 3D interface on top of such a device. In particular, for coupling and decoupling the object's movements to the user's hand (i.e. grabbing and releasing) a solution has to be found. In this paper, we introduce a novel technique for efficient two-handed grabbing and releasing objects and intuitively manipulating them in the virtual space. This technique is integrated in a novel 3D interface for virtual manipulations. A user experiment shows the superior applicability of this new technique.

## Keywords

Virtual reality, interaction techniques, bimanual interaction

## 1. Introduction

For interaction with a virtual environment, hand-tracking is one of the favorite approaches, because it directly exploits the ease and perfection with which humans employ their hands in everyday life. In order to support immersive user experience, markerless real-time hand-tracking without the need of special initialization procedures gained a lot of interest in recent years. Presently, methods fulfilling these properties are capable of tracking up to 6 continuous degrees of freedom (DOF) of both hand poses (global positions and orientations) simultaneously and recognizing several stiff postures for each hand.

As suggested in [5] we strictly distinguish between the notions pose, posture and gesture. The

pose is meant to be the combination of a rigid body's 3D position and orientation (e.g. the hand's global pose). A hand posture is defined as a specified configuration of finger limb positions and orientations relative to the hand pose (e.g. the fist posture). A hand gesture (or free-hand gesture) describes a predefined movement of the hand pose(s) (e.g. writing a letter in the air).

Given the pose(s) of one or both hands simultaneously, the movements of a virtual object can be coupled to the movements of the user's hand(s). While the pose of one single hand is in principle enough for sufficiently controlling an object, the use of both hands can significantly improve certain manipulation tasks (see e.g. [14]). For example the symmetric bimanual technique named grab-and-twirl proposed by Cutler et al. [6] (a virtual object is moved as if gripped between

the hands) gives more control over an object; the center of rotation can intuitively be chosen and a considerably higher rotational accuracy is achieved.

In this paper we focus on interaction techniques for two-handed manipulation of an object's pose as an extension for a one-handed interface. We assume fundamental mechanisms such as selection of objects and movement of the virtual camera as well as single-handed interaction to be already solved and available. Nevertheless, two major problems have to be solved in order to build a 3D interface on top of the basic hand-tracking technology.

First, working with both hands simultaneously induces additional strain to the user, so permanently working with two hands is not desired and therefore an interface should additionally enable easily transiting between single and two-handed interaction.

Second, an interface for two-handed interaction has to provide mechanisms for grabbing and releasing the virtual object, which are inherent tasks during 3D interaction sessions due to the following reasons: The virtual world is typically significantly larger than the working volume of the user (the 3D region the hand is tracked in). Therefore, to enable users to move a virtual object to every position in the virtual space it has to be possible to grab it and release it in order to move it step by step. Scaling the working volume to the whole virtual world is not an option, because the accuracy would decrease too heavily. Similar, the range of angular movements of the human wrist is limited. In order to fully rotate and inspect an object from all viewing directions grabbing and releasing are indispensable. In interfaces that employ a standard 2D mouse, the problem of grabbing and releasing is solved either by lifting the mouse (while it is lifted a mouse movement induces no object movement) or by exploiting button states (usually the object is only moved if a button is held down). But in contrast to a standard 2D mouse no direct equivalent exists for markerless hand-tracking. Simple solutions for the realization of a grab and release cycle in the absence of physical buttons are disposing one degree of freedom (DOF) of one of the hand poses or applying different postures for different button states. Unfortunately, these approaches also have drawbacks. Exploiting a DOF of one hand pose reduces the available degrees of freedom (e.g. if z-coordinate value is used for triggering grabbing). The use of different postures, one for grabbing and another for releasing, is significantly more demanding for the user than simply pressing a mouse button, because the physical effort as well as the complexity of coordination of changing postures is considerably higher, especially if various specified postures are concurrently needed. Moreover, a posture change always induces an unintended pose change mainly in rotation in current markerless hand-tracking

systems. This is due to the problem that the tracking state is temporarily undefined during a posture change. Therefore, it would nearly be impossible to instantly stop an object's movement by switching to another posture.

An easy to use solution of these problems is crucial for usability and efficiency of the interface. Such approaches have to be adapted to the users' capabilities and incapacities, the applications' specifications as well as the requirements and drawbacks of the used hand-tracking method. This diversity of demands poses several challenges in the design of easy to use interfaces.

The main contributions of this paper are summarized in the following. We introduce a novel technique for symmetric object manipulations, which allows for efficient and intuitive two-handed grabbing, moving and releasing of objects without the need of posture changes. This technique is well suited as an extension for one-handed object manipulation in order to enable a superior object control in complex and high precision tasks. We show that performing transitions between single and our two-handed interaction can be established easily and effectively. We further introduce several simple solutions for compensating some general drawbacks that appear if vision based markerless two-hand-tracking is employed. Last but not least a short evaluation and discussion of this technique is as well presented, which is based on a user study, user questioning and our observations.

## 2. Related Work

The large amount of literature on interaction techniques makes it practically impossible to give a full review of the previously reported methods here; elaborate analysis can be found in [5], or [9] for multi modal interaction. We will only discuss the most related methods that are designed for or can be applied to 3D interaction interfaces based on markerless hand-tracking as an input device.

### 2.1 Object Grabbing and Releasing Techniques

According to Zachmann [21] grabbing an object (i.e. attaching the object to the hand(s)) can be realized in (at least) three different ways: single-step, two-step or naturally. Single-step grabbing attaches the object at a certain event (e.g. a spoken command like "grab thing"). Two-step grabbing can be further divided into the following interaction steps:

1. Some event (e.g. a posture or spoken command) switches the grabbing mode on; only in this mode, objects can be grabbed.
2. The object is attached to the hand(s) at another event.

To release the object usually the same event as in the first step is used. In the grabbing mode natural grabbing is typically realized by conditioning

collisions of virtual hand representations with the object (e.g. two virtual hand models must collide with the object). The object's movements will be coupled directly to the hands', when the object is touched this way. Note that if the virtual scene is rendered into the hands' working area/volume the virtual hand representations are normally not visualized, because the real hands can represent themselves. Otherwise, the virtual representations of the user's hands have to be rendered. However, in both cases precise two-handed manipulation of a small object is nearly impossible, because the distance between the hands has to be very small for grabbing and holding the object.

The easiest solution for triggering an attach or release action is extending the hand-tracking interaction interface with additional physical buttons as for example floor pedals. However, this kind of interaction turned out to be awkward and slow (according to [7]).

Another approach is using a dwell time threshold for triggering a grab or release event as for example used in [20] and [19]. Although this is simple, it introduces a constant lag in the interaction.

A further approach is to use speech to signal grabbing or releasing [2]. But this is especially excessive if several grab and release cycles are needed.

Therefore, most approaches adopt grabbing postures to determine whether an object is attached to the hand or not (e.g. [12] or [3]). Unfortunately, it turned out to be quite difficult to release an object at a precise position [13]. The reasons for this are: first, it is demanding for people to fix their hands precisely in midair without having physical support. Second, judging the release point without tactile feedback can be difficult. Third, the finger movements of a grabbing action often induce involuntary changes of the hand's global pose. To solve the third problem of using grabbing postures Osawa [13] proposed an approach to automatically adjust the release pose of a virtual object based on the relative speed of the two grabbing fingers (usually the thumb and one forefinger).

Furthermore, in current markerless hand-tracking systems most of the different postures humans exploit for grabbing (e.g. the 3-point pinch grab exploiting the thumb and two other fingers or the power grab exploiting the whole hand [21]) are not available; typically only one grabbing posture is supported per hand. Additionally, in order to ensure a stable tracking, this posture must be formed very exactly and clearly. We observed this to be cumbersome for most users.

## 2.2 Object Manipulation Techniques

In a hand-tracking based 3D interface commonly virtual representations of the user's hands are shown in the 3D scene. When one or both (depends

on the employed technique) virtual representations intersect with an object, the object can be grabbed. Once grabbed, the movements of the virtual hand(s) is(are) directly applied to the object in order to move, rotate or deform it. This is called the virtual hand metaphor, which is the most common direct manipulation technique for manipulating objects [1]. When the coupling between the physical world (hand or device) and the virtual representations work well, this interaction technique turns out to be very intuitive, since it is similar to every-day manipulation of objects. The main drawback of the virtual hand metaphor is the scaling problem; the limited workspace of the user's limbs or the input device, which makes distant objects unreachable.

To solve this problem various approaches were reported. For example the Go-Go technique [16] simulates an interactively non-linear growing of the user's arm(s). When a user's hand is close to her, the mapping of real hand pose to virtual object pose is one to one. As she extends her hand and arm beyond a certain range, the mapping becomes nonlinear and the virtual arm "grows". Thus, she is able to reach objects out of her range. Several other solutions to the scaling problem were introduced as for instance the World-in-Miniature technique [19], HOMER (hand-centered object manipulation extending ray-casting) [4], Scaled-World Grab [11] or Voodoo Dolls [15]. However, none of these techniques can be identified as the "best" solution; their performance depends on the task and environment.

Two-handed 3D interaction techniques are commonly distinguished (according to [8]) into bimanual asymmetric, where the hands perform different actions (e.g. the non-dominant hand holds something while the other manipulates it), or bimanual symmetric, where the hands perform identical actions (e.g. pulling a rope, typing on the keyboard). As in this paper a technique for bimanual symmetric interaction is introduced, we will not discuss asymmetric techniques here, an overview of asymmetric interaction can be found in [5]. In the following the symmetric 3D interaction techniques most relevant to our work are outlined.

Two-handed scaling of objects is a popular example for symmetric bimanual interaction. This can be solved as follows: the user picks up two sides of an object and can scale the object by moving her hands apart or together (e.g. [22]).

As well traveling in the virtual environment can be solved bimanually symmetric. In the Polyshop system [10] a user can travel by performing a rope gesture. By pulling on an invisible rope with both hands the user can pull her through the environment.

Using both hands for moving an object is most related to our work. In this context Cutler et al. [6] introduced several bimanual symmetric techniques: the grab-and-twirl technique enables the user to

pick up two sides of an object and then to carry and turn it around with both hands. By fixing one or more DOFs of the applied transformation several other techniques were derived such as the grab-and-carry technique (no roll around the line connecting the two hands is allowed) or the turntable technique (only turning around a fixed axis of rotation is allowed). These techniques turned out to be both intuitive and efficient. However, these techniques implicitly assume the employment of the virtual hand metaphor for determining the two pivot points (i.e. the two points where the object is grabbed).

### 3. The Hand-Tracking Device

To markerless track the 6 continuous DOFs of the user's both global hand poses in several different stiff postures we implemented the method of Schlattmann et al. [17]. We decided to use this approach due to the real-time capability and automatic initialization. In contrast to [17], we connected all three cameras to one and the same computer where as well the hand-tracking is computed. The intersection of the viewing volumes of the three cameras defines the working volume of the hand-tracking. It describes the physical space the user's hand poses and postures are determined in (approximately  $80cm \times 50cm \times 50cm$ ).

In this method for each hand the fingertip positions of two stretched fingers are identified and the hand center (we define the center to be the hand position) is computed as a third point to sufficiently determine the hand pose in each frame. Note that due to the dependency on the fingertip positions movements of the stretched fingers slightly influence the derived hand pose. The different hand postures are distinguished by evaluating some heuristics based on the local surrounding of the fingertip points. In our interface we restrict ourselves to the employment of the posture with the stretched thumb and index-finger as it can be tracked with the highest accuracy.

### 4. Bimanual Symmetric Grab

For our grabbing and releasing technique we assume an object has been selected and now the user's intention is manipulating the objects pose by exploiting both hands in order to gain a high precision and control. In this case an additional suitable mechanism is needed for determining when the objects movements shall be coupled to the hands' movements or not (grabbed or released). This mechanism should enable precise releasing of objects and should be manageable efficiently and intuitively. To this end, we introduce an approach based on different velocities of the grabbing/releasing action. The idea is to trigger a grabbing action if the hands get closer while a release action is performed if the hands move apart as illustrated in Figure 2.



Figure 2: Illustration of real two-handed grabbing, manipulating and releasing. The red arrows indicate either a grabbing gesture (Left) or a releasing gesture (Right).

Thereby, a reasonable formulation of the exact conditions is crucial in order to avoid involuntary grabbing or releasing actions.

Grabbing and releasing of an object can be expressed by a state machine comprising the two states 'Grabbed' and 'Released' and the transitions in between. The conditions for transiting between these states are based on the grabbing velocity  $v_G^i$  and the manipulation velocity  $v_M^i$ .  $v_G^i$  denotes the signed velocity with which the hands moved together (positive) or apart (negative) between the previous frame  $i-1$  and the current frame  $i$  and is defined as:

$$v_G^i := \frac{\|p_L^{i-1} - p_R^{i-1}\| - \|p_L^i - p_R^i\|}{t^i - t^{i-1}},$$

where  $p_L^i$ ,  $p_L^{i-1}$ ,  $p_R^i$  and  $p_R^{i-1}$  are the tracked positions of the left and right hand in frame  $i$  and  $i-1$ , respectively.  $t^i$  and  $t^{i-1}$  denote the times of frame  $i$  and  $i-1$ . The manipulation velocity  $v_M^i$  denotes the sum of the hands' translational velocities that will potentially be used for object manipulation. In other words  $v_M^i$  is the sum of the translational velocities of both hands minus the grabbing velocity and is defined as:

$$v_M^i := \frac{\|p_L^i - p_L^{i-1}\|}{t^i - t^{i-1}} + \frac{\|p_R^i - p_R^{i-1}\|}{t^i - t^{i-1}} - |v_G^i|.$$

Now, in order to discover whether the user wants to perform a grab/release action or simply an object manipulation, the grabbing and manipulation velocities are analyzed. To this end, we introduce the modified signed grabbing velocity  $\tilde{v}_G^i$  which is defined as:

$$\tilde{v}_G^i := \begin{cases} \text{sgn}(v_G^i)(|v_G^i| - v_M^i) & , \text{if } |v_G^i| > v_M^i \\ 0 & , \text{else} \end{cases}$$

Thereby,  $\text{sgn}$  denotes the sign function. Note that  $\tilde{v}_G^i$  is zero while the manipulation velocity is either dominant or equal to the grabbing velocity. If the grabbing velocity is dominant  $\tilde{v}_G^i$  is either positive (the hands perform a grab movement) or negative (the hands perform a release movement). Now, our grabbing and releasing technique can be expressed by the state machine depicted in Figure 3.

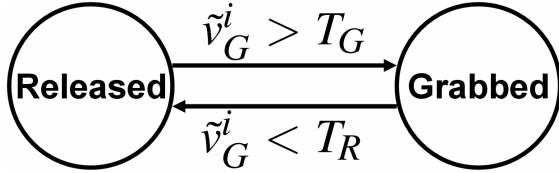


Figure 3: State machine illustrating how an object can be grabbed and released. Thresholds  $T_G$  and  $T_R$  are applied to the modified signed grabbing velocity  $\tilde{v}_G^i$ .

$T_G$  and  $T_R$  are thresholds for triggering a grab or release action, whereby  $T_G$  has to be a positive and  $T_R$  a negative real value. These parameters are used to adjust the velocity that has to be performed by the user to grab or release an object. In our current setting we use symmetric thresholds with  $T_G = 10 \frac{cm}{s}$  and  $T_R = -10 \frac{cm}{s}$ . These values were determined in a short test scenario, where several users should perform this kind of grabbing and releasing gestures and manipulation movements while we recorded the employed velocities. Additionally, we categorized the velocities manually to belong to the ‘Grabbed’ or ‘Released’ state. By analyzing several of these sequences we identified the threshold values that would lead to the fewest errors. An error can either be a falsely triggered grab/release action or an intentional grab/release action that was not triggered.

Using this formulation the grabbing and releasing of a virtual object can be performed with only a slight movement; no spacious and therefore uncomfortable gestures are needed. Moreover, employing the manipulation velocity to inhibit grab/release actions avoids involuntary grabbing/releasing during a manipulation task. For the same reason the velocity thresholds  $T_G$  and  $T_R$  can be chosen with such a low value which enables grabbing/releasing with slow movements. Note that these thresholds have to be greater than zero because otherwise unintentional grab/release actions could occur due to the natural tremor of human hands or slight inaccuracies of the hand-tracking device.

Furthermore, this formulation allows for grabbing and holding an object with arbitrary distance between the hands. This supports the adaptation to different demands of the current task, e.g. if a very precise rotation shall be performed a greater distance between the hands is superior while a smaller distance is more convenient for large translational manipulations due to the limited work space. Note that a simple solution as for example using a specified distance between the hands to trigger grabbing and releasing would not have these features.

Using this criterion for releasing operations enables fairly precise positioning of objects. However, humans in general do not perform release operations with perfect symmetric movements of

both hands and only in the grabbing direction. Hence, sometimes slight unintentional movements occur while the user wants to release the object but the velocity threshold  $T_R$  is still not exceeded. To overcome this problem, we inhibit object movements while  $\tilde{v}_G^i < 0$  causing the object to stand still while the grabbing velocity is dominant and the hands perform very slow releasing movements. This way, the object’s movements stop immediately when the user starts to perform a release action. Note that this condition generally imposes no restriction on intended object manipulations, because in these cases the manipulation velocity is dominant and  $\tilde{v}_G^i$  is equal to zero. Applying this simple modification to our two-handed grabbing/releasing technique the precision for positioning virtual objects can further be improved.

Additionally, our technique enables performing grab and release actions in a comfortable and effective way. Furthermore, due to the close relation to natural two-handed grabbing it is intuitive and no long practicing is needed.

## 5. Object Movement Modification

In order to bimanually move an object we developed a manipulation technique that is inspired by the grab-and-twirl technique introduced by Cutler et al. [6]. The grab-and-twirl technique can be formalized as follows. The translational object movement is determined by the average translational movements of both hands. The rotational object movement is determined by two different perpendicular rotations: the rotation of the line connecting the two hand positions and the average hand rotations around this line. Both rotations are applied to the object. Note that Cutler et al. [6] proposed to use not the average hand rotations but instead the rotation of only one hand. However, because we observed the combination of both enabling more precise rotations we preferred to use the average.

In contrast to the grab-and-twirl technique we do not trigger grabbing and releasing of an object by moving the hands’ virtual representations until they both intersect the object and then performing a grabbing posture. Instead, using the technique introduced in Section 4 grabbing and releasing can take place independent of the size or position of the virtual object or the distance between the hands. Therefore, grabbing and releasing is significantly easier and more comfortable, but in order to preserve the intuitiveness of object movements the hands’ physical movements must not be mapped to the virtual world in the same way as done in the grab-and-twirl technique. In the grab-and-twirl technique an object is translated by a certain mapping of the physical center between the two hands to the virtual center of the object. This is

reasonable, because the virtual hand positions coincide with the two pivot points where the object is grabbed (see Figure 4(Top left)). In our technique, the user also anticipates two pivot points being located close to the object's surface where it is grabbed, but as in our case the distance between the hands' positions is not fixed to the object size, the same mapping as in Figure 4(Top left) of physical to virtual movements would lead to unnatural results (see Figure 4(Top right)). To obtain reasonable results the mapping has to be modified as illustrated in Figure 4(Bottom left).

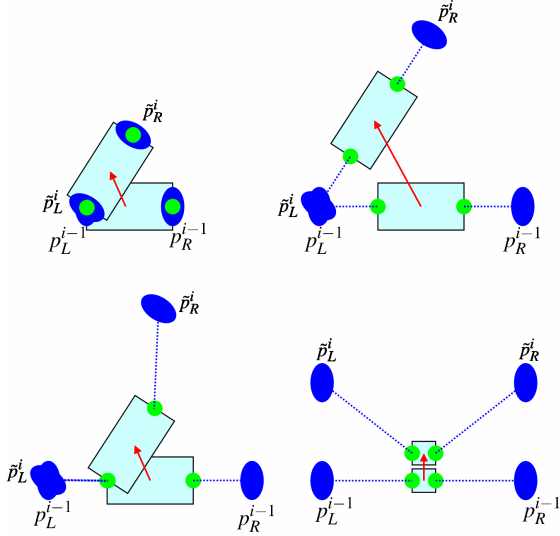


Figure 4: Illustration of the problem of mapping real to virtual coordinates with fixed distance between the hands. An object (rectangular box) is moved by exploiting the positions of both hands (blue ellipses). The pivot points are illustrated in green and the applied translation vector in red. Top left: distance between hands is equal to distance between pivot points. Top right: hand distance is greater than distance between pivot points but the same mapping is used as in (Top left). Bottom left: hand distance is greater than distance between pivot points and a suitable mapping is used. Bottom right: hand distance is much greater than distance between pivot points and a scaling by the ratio of pivot point distance to hand position distance is used.

Unfortunately, if a solution which simply scales the translation by the ratio of the distance between the two pivot points (determined by the two intersection points of the object's convex hull with the line through the object center having the direction of the line connecting the two hand positions) to the distance between the hands would be applied, small objects could hardly be translated any more as illustrated in Figure 4(Bottom right). Our solution consists in two steps: first, we adjust the hands' positions such that their distance is normalized and second, we split the translational movement up into an asymmetric part and a symmetric part which are scaled differently.

First, the left and right hand positions  $p_L^i$  and  $p_R^i$  in the current frame  $i$  are adjusted to have the

same distance as in the previous frame  $i-1$ . The adjusted position for  $p_L^i$  is obtained by moving  $p_L^i$  in direction of  $p_R^i$  with a certain signed distance. This signed distance is equal to the difference between the hand distances  $d^i$  and  $d^{i-1}$  in frame  $i$  and  $i-1$ , respectively, multiplied by a suitable weighting factor  $\lambda_L^i$ . The scheme for obtaining the adjusted left hand position  $\tilde{p}_L^i$  is defined as:

$$\tilde{p}_L^i := \lambda_L^i (d^i - d^{i-1}) \frac{p_R^i - p_L^i}{\|p_R^i - p_L^i\|} + p_L^i.$$

The weighting factor  $\lambda_L^i$  is defined to be the ratio of the translational amount of the right hand to the sum of translational amounts of both hands and can be formalized as follows:

$$\lambda_L^i := \frac{\|p_R^i - p_R^{i-1}\|}{\|p_R^i - p_R^{i-1}\| + \|p_L^i - p_L^{i-1}\|}.$$

$\tilde{p}_R^i$  is determined analogously by swapping all  $L$ 's and  $R$ 's in both equations. This way the hand that moved slower becomes stickier for the object (i.e. the object tends to stay in touch with this hand). If for example one hand remains at the same position, the object can not be translated toward the grabbing direction by movements of the other hand. Furthermore, no more translational object movements are induced by decreasing or increasing the distance between the hands. Note that  $\lambda_L^i$  is equal to  $(1 - \lambda_R^i)$  and  $\lambda_L^i \in [0,1]$ , therefore the distance between the adjusted hand positions is guaranteed to be equal to the distance between the previous hand positions.

Second, the translational movement is split up into an asymmetric part (difference of the hand translation amounts) and a symmetric part (similarity of the hand translation amounts). The asymmetric part describes the amount of translation that influences the position of the center of rotation. This part has to be applied in a one-to-one mapping (i.e. it has to be mapped as in Figure 4(Bottom left)). The symmetric part describes the amount of translation that has no influence on the displacement of the rotation center. This part can be applied to the object translation with an arbitrary scaling (independent of the object size or the hand distance) without affecting the intuitiveness. The two parts can be computed as follows: if  $t_L^i$  and  $t_R^i$  are the left and right hand translations with  $t_{L/R}^i := \tilde{p}_{L/R}^i - \tilde{p}_{L/R}^{i-1}$  in frame  $i$ , then  $t^i := \frac{1}{2}(t_L^i + t_R^i)$  describes the total amount of translation. We then define  $\mu^i$  to be the percentage of symmetric translation with

$$\mu^i := \frac{\min(\|t_L^i\|, \|t_R^i\|)}{\max(\|t_L^i\|, \|t_R^i\|)}.$$

Now, the resulting modified total translation  $\tilde{t}^i$  can be written as:

$$\tilde{t}^i := s\mu^i t^i + r(1 - \mu^i)t^i,$$

where  $r$  is defined to be the ratio of object extend to the current distance between the hands and  $s$  can be chosen arbitrarily to alter the mapping from real to virtual coordinates. This way, if an asymmetric rotation is performed (e.g. one hand is moved fast, the other stands still) the object is rotated intuitively around the pivot point touching the object and corresponding to the hand that was not moved. On the contrary, if a symmetric movement of both hands in equal directions is performed the object translation does not depend on the object size or distance between the hands.

In order to provide a visual clue about where the object is grabbed and how it can be turned around the resulting two pivot points are visualized in the 3D interface. Two spheres are drawn at the corresponding positions as long as the object is grabbed. Additionally, this gives a direct visual feedback about whether the object is grabbed or released.

With this technique both human hands are exploited for simultaneous manipulation of the whole 6 DOFs of an object's pose in order to gain a significantly improved rotational precision and a generally higher control over the object. This is particularly due to the following reasons: first, the rotation depends on the distance between the hands leading to an increased precision if the distance is increased. Second, the center of rotation can be chosen more intuitively than in one-handed interaction, because object rotations are determined by hand translations (e.g. fixing one hand to a certain position leads to rotating an object around this point when the other hand is moved).

## 6. Integration with a One-Handed Interface

We argue that symmetric two-handed object manipulation is primarily suitable as an extension of one-handed manipulation. Only if high precision and/or control of the object is needed the application of both hands can significantly improve the current task. A reasonable scenario could be as follows: a user selects an object and moves it to the approximate pose by employing only one hand. Only then she uses both hands for fine adjustment of the object pose. In order to allow for such scenarios in an effective way transitioning from single-handed to two-handed interaction and backwards must be manageably fast and uncomplicated.

The first case of switching from single to two-handed interaction can simply be solved by

automatically enabling the two-handed interaction mode when the second hand enters the working volume.

The second case of changing from two to single-handed interaction is more complicated. This is due to the problem that a hand could have left the working volume unintentionally. This occurs if the user currently performs an object manipulation at the edge of the working volume and moves one hand outside by mistake. To this end, we distinguish between two cases: first, if the object was grabbed when the hand left the working volume we assume leaving was unintended and prompt a message on the screen to move the hand back inside. Second, if the object was already released we assume the user wants to switch back to the one-handed manipulation mode which is therefore automatically enabled in this case. This distinction turned out to be intuitive and fast to handle.

## 7. Stabilizing the Object Movement

Exploiting the human hands as direct input devices by using vision based markerless hand-tracking has several great advantages (e.g. intuitive, non-obtrusive). However, as well some problems arise when such devices are exploited. In the following we explain some major drawbacks and present our solutions.

One drawback of vision based markerless hand-tracking is the dependency on the hand segmentation in the camera images. Performing a good segmentation becomes even harder when two hands are tracked simultaneously due to reciprocal occlusions. Such occlusions occur more often if the distance between the hands decreases. We observed the tracking stability of the hands' orientations to be primarily sensitive to such hand configurations. Note that in the employed two-handed object manipulation technique the hand orientations are only exploited for rotating the object around the line connecting the two hand positions (see Section 5). In order to avoid distracting rotational jumps during object manipulation we propose the following simple stabilization procedure: if the angular velocity of one hand exceeds a certain threshold (currently we use  $2\pi \frac{rad}{s}$ ) its rotation is set to the identity. This threshold value was determined by analyzing the typical velocities of rotational jumps induced by an incorrect tracking. This way, the jumps of the hands' orientations are not applied in the current step leading to a stable object manipulation. In return, very fast rotations around this axis can not be performed any more. This is a clear limitation, but as it counts only for velocities higher than  $2\pi \frac{rad}{s}$  the interaction is marginally affected. In practice, we discovered such high angular velocities to be exploited for less than 1% of object rotations in typical manipulation tasks.

Another problem induced by segmentation issues is that tracking two hands is impossible when they can not be separated in each camera image. In this case, a message is prompted requesting the user to increase the distance between her hands.

In particular relevant for two-handed object manipulations is the problem of limited work space; the hands can only be tracked if they can be seen by enough cameras. If an object shall be translated a long distance the hands have to be moved a proportional way in the working volume. If in this case the user grabs, holds and moves an object with a large space between her hands, the range of available translational movement becomes even smaller and one hand often leaves the working volume unintended. Therefore, we prompt a hint suggesting the user to decrease the distance between the hands if one hand leaves the working volume and the distance between the hands was higher than 30cm in the preceding frame. This threshold value was used, because we observed that larger distances do not significantly improve the accuracy of object manipulations.

## 8. Experiments

We conducted an experiment to evaluate the performance of the proposed technique for two-handed grabbing and releasing virtual objects in an immersive environment. Our hypothesis was that our method would be superior to other two-handed techniques commonly applied for hand-tracking devices. Moreover, we expected that our technique would generally be superior to one-handed techniques in high precision tasks.

In order to investigate these hypotheses, we compared our two-handed grabbing and releasing technique to three other approaches: the exploitation of a grabbing posture for triggering grabbing/releasing (i.e. only if the subject forms the grabbing posture with one or both hand(s) the virtual object moves according to her hands), the use of a standard 6 DOF controller (3D mouse) and the application of a recently developed one-handed technique; the Jerky Release technique (see [18]). The idea of this one-handed technique is to grab the object by default and only release it while the user performs a fast and jerky hand movement. In a short user study this technique outperformed a grabbing posture based technique and showed similar results as a standard 3D mouse in 6 DOF manipulation tasks.

### 8.1 Experimental Setting

Two connected PC-based systems (Intel E6600, Geforce 8800) were used in the experiment, one coupled to the cameras for tracking the hand (see Section 3) and one for running the virtual environment application. The application was visualized on a standard 19" TFT-Display.

Additionally, a 3D connexion SpaceNavigator was connected to the second PC.

An illustration of how this system works together with our bimanual technique is given in Figure 1.

#### 8.1.1 Experimental Task

In the experimental task a virtual object had to be moved to a specified position and orientation (see Figure 5) by using the different 6 DOF techniques/controllers. The time that is needed for completing one task was split up into the approach time and the precision time. The approach time is measured when the object is approximately approached (see Figure 5(Middle)); when it is moved roughly to the target pose (i.e. less than 6 degrees rotational error and 4 units translational error).

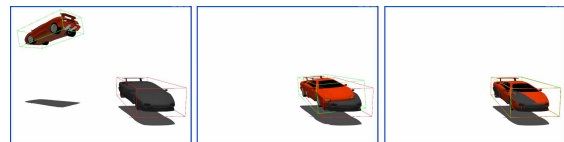


Figure 5: Illustration of the experimental task that had to be solved. (Left) Initial situation. (Middle) Roughly approached target pose (the approach time is measured). (Right) Solved task (the precision time is measured).

The period from this moment on until the user releases the object almost precisely (i.e. less than 1,5 degrees rotational error and 0,5 units translational error) at the target pose (see Figure 5(Right)) is defined to be the precision time. This way, both the capabilities of performing rough/spacious as well as precise/fine manipulations are measured for each technique/controller. To reach the target pose several grab and release cycles had to be performed for the hand-tracking based techniques. No snapping algorithm (the object snaps to the desired pose, when it is near by) was applied.

#### 8.1.2 Participants

Ten participants (all males, all university students) took part in the experiment. They had little or no virtual reality experience.

#### 8.1.3 Procedure

Each participant had to solve the task four times by employing each of the four controllers. Thereby, the sequence of the employed controllers was permuted almost evenly and all tasks had to be finished until the next controller was adopted. Before starting the test for each controller, its mode of action was explained and the subjects could familiarize with it in a short preparation time (two minutes).



## 8.2 Results and Discussion

The average approach times and precision times including standard deviations are depicted in Figure 6. Roughly approaching an object was significantly superior if only one hand or the 3D mouse was employed instead of a two-handed technique. This is mainly because more grab and release cycles were used for two-handed translational movements due to the higher need of workspace (both hands have to be located in the working volume) and the slightly reduced anatomical range (the shoulder has to remain straight if both arms are stretched). No significant difference was found between the 3D mouse and the single-handed object approaching. However, our two-handed technique was significantly superior to the two-handed posture based technique.

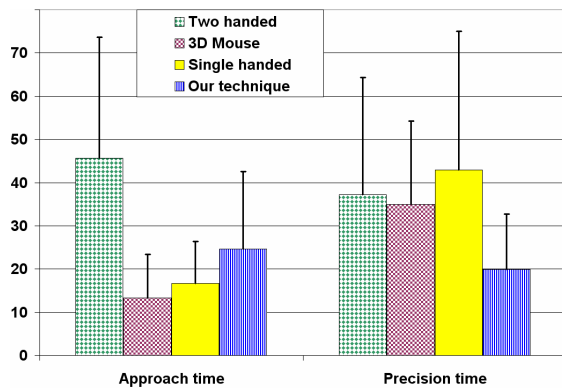


Figure 6: Completion times (in seconds) with standard deviations (error bars).

We observed the subjects have some problems using different postures for grabbing and releasing primarily because a higher concentration was needed for switching between the standard and the grabbing posture.

Precisely positioning an object was significantly superior if our two-handed technique was applied. Surprisingly, the precision times of the two-handed posture based technique were similar as of the 3D mouse and the one-handed technique. This was caused by overhasty hand movements during a release operation while the hands were still holding the virtual object. Note that in this case the involuntary object movement leads to losing the target pose so the subject must repeat the high precision positioning.

The average total times of the different controllers were 83 (two handed), 48.3 (3D Mouse), 59.6 (single handed) and 44.5 (our technique) seconds. Note that our technique also performed best for the whole task.

The high standard deviations in Figure 6 are caused by two major reasons. First, the timings were strongly dependent on the manual skills of each individual; a subject that performed *slowly/fast* with a certain controller mostly performed *slowly/fast* with the other controllers, too. Thereby,

*slow/fast* means having high/low average per controller timings with respect to the other participants.

Second, the timings were dependent on the individual controller preferences. While several users could easily handle one controller some others were nearly incapable to work with it. In particular precisely positioning by using the 3D mouse had very inconsistent results. For this reason we think that in an end user 3D interface redundancies between different techniques (if possible) and/or selectable alternatives should be available.

After the user experiments, we questioned the subjects for their subjective impression concerning the performance of the different controllers. For precisely positioning eight participants would prefer our two-handed technique; one would prefer the two-handed posture based technique and another one the 3D mouse. For approaching the target they rated both the single-handed technique and the 3D mouse best with four votes for each; two participants voted for our two-handed technique. Note that these results correspond exactly to the measured times.

## 9. Conclusions and Future Work

We presented a novel technique for two-handed grabbing and releasing virtual objects which can be used for efficient symmetric object manipulations. We further introduced some modifications needed for preserving the intuitiveness of these symmetric manipulations. We show how our two-handed technique can reasonably be integrated with a single-handed interface and introduced some stabilization procedures needed for compensating some drawbacks of markerless two hand-tracking. Last but not least, we performed a user experiment to compare our two-handed technique to another two-handed technique, to a single-handed technique as well as to a standard 6 DOF controller.

The results of our user study confirmed our hypotheses that single-handed manipulation performs better for coarse tasks and two-handed for high precision tasks. In order to benefit from both advantages a possible solution would be combining a single-handed technique with our two-handed technique. To discover which single-handed technique best fits to our two-handed technique we plan to investigate the performance of different combinations.

Another future direction might be adapting different bimanual asymmetric techniques for the special problems and requirements of markerless hand-tracking.

## References

- [1] J. D. Boeck, C. Raymaekers, and K. Coninx. Are existing metaphors in virtual environments

- suitable for haptic interaction. In *Proceedings of Virtual Reality International Conference*, pages 261–268, 2005.
- [2] R. A. Bolt. “put-that-there”: Voice and gesture at the graphics interface. *SIGGRAPH Comput. Graph.*, 14(3):262–270, 1980.
- [3] C. W. Borst and A. P. Indugula. Realistic virtual grasping. In *VR '05: Proceedings of the 2005 IEEE Conference 2005 on Virtual Reality*, pages 91–98, Washington, DC, USA, 2005. IEEE Computer Society.
- [4] D. A. Bowman and L. F. Hodges. An evaluation of techniques for grabbing and manipulating remote objects in immersive virtual environments. In *SI3D '97: Proceedings of the 1997 symposium on Interactive 3D graphics*, pages 35–38, New York, NY, USA, 1997. ACM.
- [5] D. A. Bowman, E. Kruijff, J. J. LaViola, and I. Poupyrev. *3D User Interfaces: Theory and Practice*. Addison Wesley Longman Publishing Co., Inc., Redwood City, CA, USA, 2004.
- [6] L. D. Cutler, B. Fröhlich, and P. Hanrahan. Two-handed direct manipulation on the responsive workbench. In *SI3D '97: Proceedings of the 1997 symposium on Interactive 3D graphics*, pages 107–114, New York, NY, USA, 1997. ACM.
- [7] C. Grätzel, T. Fong, T. Fong, S. Grange, and C. Baur. A noncontact mouse for surgeon-computer interaction. *Technol. Health Care*, 12(3):245–257, 2004.
- [8] Y. Guiard. Asymmetric division of labor in human skilled bimanual action: The kinematic chain as a model, 1987.
- [9] A. Jaimes and N. Sebe. Multimodal human-computer interaction: A survey. *Comput. Vis. Image Underst.*, 108(1-2):116–134, 2007.
- [10] D. Mapes and J. M. Moshell. A two-handed interface for object manipulation in virtual environments. *Presence: Teleoperators and Virtual Environments*, 4(4):403–416, 1995.
- [11] M. R. Mine, J. Frederick P. Brooks, and C. H. Sequin. Moving objects in space: exploiting proprioception in virtual-environment interaction. In *SIGGRAPH '97: Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 19–26, New York, NY, USA, 1997. ACM Press/Addison-Wesley Publishing Co.
- [12] N. Murray and T. Fernando. An immersive assembly and maintenance simulation environment. In *DS-RT '04: Proceedings of the 8th IEEE International Symposium on Distributed Simulation and Real-Time Applications*, pages 159–166, Washington, DC, USA, 2004. IEEE Computer Society.
- [13] N. Osawa. Automatic adjustments for efficient and precise positioning and release of virtual objects. In *VRCA '06: Proceedings of the 2006 ACM international conference on Virtual reality continuum and its applications*, pages 121–128, New York, NY, USA, 2006. ACM.
- [14] R. Owen, G. Kurtenbach, G. Fitzmaurice, T. Baudel, and W. Buxton. When it Gets More Difficult, Use Both Hands - Exploring Bimanual Curve Manipulation. *Proceedings of Graphics Interface*, 17-24, 2005.
- [15] J. S. Pierce, B. C. Stearns, and R. Pausch. Voodoo dolls: seamless interaction at multiple scales in virtual environments. In *I3D '99: Proceedings of the 1999 symposium on Interactive 3D graphics*, pages 141–145, New York, NY, USA, 1999. ACM.
- [16] I. Poupyrev, M. Billinghurst, S. Weghorst, and T. Ichikawa. The go-go interaction technique: non-linear mapping for direct manipulation in vr. In *UIST '96: Proceedings of the 9th annual ACM symposium on User interface software and technology*, pages 79–80, New York, NY, USA, 1996. ACM.
- [17] M. Schlattmann and R. Klein. Simultaneous 4 gestures 6 dof real-time two-hand tracking without any markers. In *ACM Symposium on Virtual Reality Software and Technology (VRST '07)*, November 2007.
- [18] M. Schlattmann, T. Na Nakorn, and R. Klein. 3D Interaction Techniques for 6 DOF Markerless Hand-Tracking. To appear in *Journal or Proceedings of WSCG*, 2009.
- [19] R. Stoakley, M. J. Conway, and R. Pausch. Virtual reality on a wim: interactive worlds in miniature. In *CHI '95: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 265–272, New York, NY, USA, 1995. ACM Press/Addison-Wesley Publishing Co.
- [20] A. Wilson and H. Pham. Pointing in intelligent environments with the world cursor. In *INTERACT*, 2003.
- [21] G. Zachmann. *Virtual Reality in Assembly Simulation – Collision Detection, Simulation Algorithms, and Interaction Techniques*. PhD thesis, Darmstadt University of Technology, Germany, Department of Computer Science, may 2000.
- [22] R. C. Zeleznik, A. S. Forsberg, and P. S. Strauss. Two pointer input for 3d interaction. In *SI3D '97: Proceedings of the 1997 symposium on Interactive 3D graphics*, pages 115–ff., New York, NY, USA, 1997. ACM.