# Pseudo-Physical Interaction with a Virtual Car Interior in Immersive Environments

Mathias Moehring

VOLKSWAGEN AG, Wolfsburg, Germany

mathias.moehring@volkswagen.de

Bernd Froehlich

Bauhaus University Weimar, Germany

bernd.froehlich@medien.uni-weimar.de

**Abstract**

*The use of Virtual Reality technology for the product engineering process in the automotive industry has a lot of potential – in particular in the area of usability and reachability studies. We analyzed the manipulation of knobs, controls, flaps, mirrors and other components in a real car and classified them with respect to their constraints. Based on this classification, we developed a set of pseudo-physical direct interaction techniques, which closely approximate the real world behavior without employing a force-feedback device. Our approach uses a hierarchical grasp heuristic to decouple the interaction from the collision of the fingers with the virtual car components. This approach makes the interaction more robust while no haptic feedback is available. A pilot study of our implementation revealed that our direct manipulation techniques are a good step towards more realistic interaction with virtual car interiors.*

## 1. Introduction

The automotive industry performs a variety of analyses during the product engineering process (PEP) covering crash behavior, ergonomics, design, and several other fields. For these assays a lot of hardware models have to be built in different stages of the process, which is time-consuming and cost-intensive. Some of these assays can be performed in virtual environments, which reduces the number of physical mock-ups and allows virtual analyses to be performed earlier in the process, when only CAD-data is available. Thus errors and impossibilities can be detected before building hardware models. In addition, virtual assays allow the exploration of different variants and alternatives as well as studies that can not be done with physical mock-ups.

We developed a system that supports direct interaction of the user's hands with a virtual car interior in immersive environments (*cf.* Figure 1). The focus of our work is on the realistic interaction with movable parts, such as the steering wheel, flaps, openers, mirrors, and various knobs and buttons. All of these parts are constrained in their motion to less than six degrees of freedom and in addition there are range limits to consider. For example, the interior mirror is constrained to three rotational degrees of freedom and the motion is additionally limited by the windshield. Due to the lack of haptic feedback, these constraints and limitations require a sophisticated mapping of the user's hand motions to achieve a realistic behavior of the virtual parts. Our approach computes the collision between finger segments, and virtual objects, detects common grasps for the various parts based on a hierarchical grasping heuristic, and moves the constrained parts appropriately.

An interactive virtual car interior can be used in a number of assays during the PEP. In particular ergonomic studies may highly benefit from such an application. Virtual seating bucks are used to perform reachability studies,

**Figure 1:** *A user interacting with a virtual car interior in a projection-based immersive environment*

entry and exit simulations, and visibility studies. First implementations for virtual assays using seating bucks are already available, but they are often developed for head mounted displays and the functionality is mostly limited to basic interactions with the steering wheel and doors. We extend this work to projection-based environments and to a variety of parts in the car interior, which can be manipulated by the user's hands. Most virtual assays can not be performed with input devices such as wands or joysticks, since these devices allow only indirect interaction.

In this paper we have carefully analyzed the typical interactions with components of a car interior. The observed grasps were classified and a hierarchical grasp heuristic was derived. The various movable parts in a car interior were grouped according to their constraints and appropriate techniques for pseudo-physical interactions with these components were developed. We implemented our approach for a data glove and a new optical finger tracking system. A pilot study for a complex car interior confirmed the potential of our approach.

## 2. Related work

At first sight direct interaction with the user's hand seems to be the most intuitive metaphor for manipulating objects in virtual environments. Moreover, proprioception [MBS97] enables users to estimate the positions of their hands without looking at them. Direct interaction with virtual objects and the user's hands has been studied extensively, e.g. [MBS97], [Mul98], but mostly without considering complex constraints for virtual parts such as those found in a car interior.

However, it has become clear that the virtual hand metaphor suffers from a variety of limitations in projection-based immersive environments and it suffers from a lack of haptic feedback (*cf.* Figure 2). In projection-based immersive environments, the user's real hand is located between the user and the display. Thus the real hand occludes virtual objects, but the virtual world is not able to occlude the real hand. In addition, the user's eyes focus in some cases the real hand and not the display. Both problems affect the stereoscopic perception, but they are not the focus of this work.

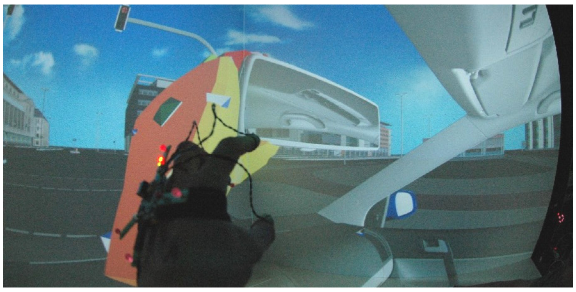

**Figure 2:** *Example for limitations of direct interaction in projection-based immersive environments. The user's hand may be focused, which leads to an unfocussed perception of the display and a reduced stereoscopic perception of the virtual scene.*

The lack of haptic feedback leads to further problems. The user's hand can not be prevented from penetrating the object and the user can not rest the arm or hand on the objects. All these haptic effects can not be simulated without an appropriate force feedback device. Existing force feedback approaches are difficult to use in projection-based environments because they disturb the user's view, they have limited accuracy and the necessary calculations for a complex car interior may impact real-time performance.

Many abstract interaction techniques for virtual environments have been developed to overcome various problems with direct interaction. [BKLP04] describes and evaluates the most important approaches. Often these abstract metaphors provide a better way to select and manipulate virtual objects. However, for our application direct manipulation with the user's hand is essential and the manipulation of objects out of reach is not needed.

The reaction of the objects in the real car follows physical rules and laws. There are approaches to simulate such processes for virtual objects. Rigid body simulations simulate effects that occur between objects, e.g. collision and motion constraints. [Bar97] presents an introduction to this topic. More complex physical simulations allow deformations of objects and manipulations with force feedback input devices. Rigid body simulations do not solve the problem of direct hand interaction with constrained parts, since they can not stop the real hand from penetrating virtual objects. In addition, preparing CAD data for physical simulations is a challenging task due to inconsistent normals, complex surfaces, and large models.

Direct interaction based on virtual grasping simulations have always been very popular despites their limitations. The reason for this is the powerful character of this metaphor and that the used display systems avoid problems in some cases or a virtual hand representation with offset to the real hand could be used. [MT94] classifies two groups of grasping approaches "analytical" and "empirical" ones. Analytical solutions calculate a physically correct position of the phalanxes to form a valid grasp. This can be done when the virtual hand is the only visual feedback for the user, e.g. in character animation or in robotics [BH02]. Other approaches use physical simulation systems to calculate a realistic virtual grasping [BI05]. In the context of this work the real hand of the user can not be controlled by the virtual environment and it is always in view of the user. Furthermore the scene is much more complex than the environments shown with analytical approaches.

Empirical approaches observe human grasping scenarios and derive simple rules for grasping decisions from them. These grasping heuristics decide if the user has grabbed a virtual object and they are generally easy to compute. Often grasping heuristics are used as a basis for analytical approaches. [Ull99] defines five rules for valid grasps with two hands, which are further refined in this work (*cf.* chapter 4.2).

Observing human grasping scenarios leads to just a few distinguishable grasps. A number of taxonomies that classify human grasps can be found in the literature, e.g. in [KI91], [Ull99], [ZR01]. Some of them take both hands into account, some do not. They have in common that they mainly divide the grasps into two groups. Some separate them into power and precision grasps, some into grasps, using the palm, and grasps not using the palm. Most consider this segmentation to be equivalent, although [KI91] discovered a difference. However merging these taxonomies and checking the grasps for usability in the car interior only five different grasps with one hand can be encountered. These grasps are the basis for our grasping heuristics (*cf.* chapter 4.2).

For direct interaction with the human hand our VR system supports the Immersion Corporation CyberGlove. This system tracks the finger joints, which leads to the largest errors occurring at the finger tips. This is due to the forward kinematic approach, tracking the user's wrist and using bend sensors to evaluate the bend of the finger joints down to the fingertips. Unfortunately for interaction in the car interior the position and orientation of the finger tips is most relevant. Further disadvantages of this system like poor results for the thumb's joint [KHW95] and its lack of comfort accrue.

During the implementation of our system a new finger tracking system was developed by A.R.T. GmbH [Hil05]. This finger-tracking system uses optical tracking to determine the position and orientation of the palm and the position of the tips of thumb, index and middle finger. These are the dominant and relevant fingers for interaction, ring finger and pinky can be omitted due to their minor contribution.

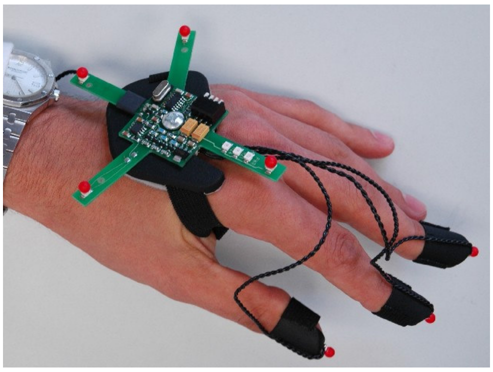

**Figure 3:** *Left handed prototype of the finger-tracking system of A.R.T.*

Using sequentially flashing active markers for the finger tips, the tracking system is able to distinguish between the three fingers. A possible hand configuration matching the found marker arrangement is calculated with inverse kinematics. Errors for the position of the finger tips may emerge only from the optical tracking system which is rather precise. We were able to use an early prototype of this system (*cf.* Figure 3), which was only available for the left hand and still uses some wires.

## 3. Interaction classes in a real car interior

Our first step was to observe some users interacting with a real car interior. The focus of this study was to find out how the users grab objects, how they move them and how the assemblies react on the user's input.

We found that the objects can be classified depending on their constraints. First of all assemblies are mounted with joints which restrict the degrees of freedom (dof) of their motion. Furthermore, the motion allowed by the joints can be restrained in two ways. Firstly, the moving range can be limited. This can be caused by explicit stoppers or the assemblies collide with other parts of the interior. For example, the steering wheel can not be turned more than a number of degrees in each direction or the interior mirror collides with the wind shield and the roof liner. Secondly, there are lock positions that let the assembly snap in mechanically. We identify groups of assemblies with similar behavior based on similar types of constraints and use them as a basis for the implementation of our interactions.

**One-dof-rotational** objects (*cf.* Figure 4a) are mounted such that they can be rotated around one axis. This group contains a number of objects. Limited to a certain range of rotation, *flaps* are usually pushed with one or more fingers or grabbed at the edge with two fingers. The *steering-wheel*, also limited to a range and relocatable with one or more fingers, can be grabbed with the whole hand wrapping the rim. A special interaction that is very important for ergonomics is the adjustment of the wheel's position. *Openers,* relocatable and limited to a range, can be grabbed with two fingers. They resile to their design position when they are released. Turned to their maximum angle, they open a door. *Doors,* limited to a range, can be pushed to open. For closing they have to be pushed or grabbed at the handle with two or three fingers or the whole hand wrapping the handle. They can only be interacted with when they are open. *Controls* are limited to a range and can only be moved when grabbed with two fingers. They can have lock positions that switch a car function.

**Two-dof-rotational** objects (*cf.* Figure 4c) are mounted with two saddle-joints in a row. They can be rotated around two axes, where one axis is fixed and the other depends on the rotation around the first. *Levers* are usually pushed with one or two fingers or grabbed with two. They are limited to a range in each dimension and have lock positions for switching car functions. They can tend to resile to a lock position. The car function can be switched at the ends of each range without a snap-in of the lever. The manually controlled *side mirror* can be moved by a knob which is pushed or grabbed with two fingers. The knob's motion is mapped onto the motion of the mirror glass, which rotates around two different axes. The electrical side mirror control uses rate control instead of position control.

There is only one **three-dof-rotational** object (*cf.* Figure 4b) – the *interior mirror*. It is limited to a rotation range in each dimension and can be moved in many different ways. Most commonly used is a grasp with two or three fingers right of the rotation center. With this grasp the mirror is clamped between the thumb at the lower edge and the other fingers at the upper edge of the mirror.

All objects in our car interior that only allow translation are limited to a motion along a single direction. There are two groups which are similar to **one-dof-translational** objects (*cf.* Figure 4d). Both can be pushed whereas usually only one finger is employed. The motion is limited to a range. *Buttons* resile to their design position when they are released. Pushing it to the maximum depth toggles a car function. The only difference to *Switches* is that they stay at a lock position when the switch state is on and resile to their design position when the switch state is off.

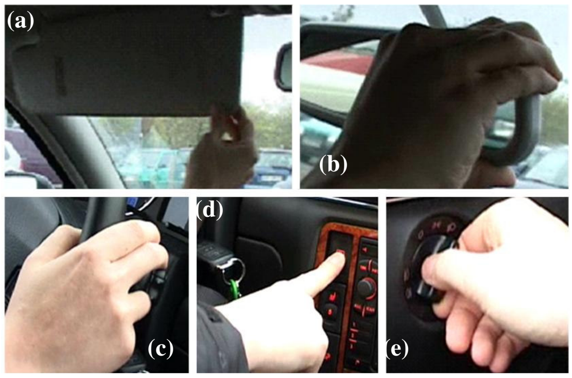


**Figure 4:** *Examples for interaction classes: flap (a), interior mirror (b), lever (c), button (d) and switch control (e).*

Besides the steering wheel adjustment, which will be handled separately (*cf.* chapter 4.3), the only object allowing rotation and translation, is a **one-dof-translational-one-dof-rotational** object (*cf.* Figure 4e) – the *light switch control*. This is a special control that can be interacted with like the other rotational controls with three lock positions, but allows a translation along one direction when it stays in two of the three lock positions. The constraint direction is defined by the rotation axis, which makes the translation independent of the rotation. The range of the translation is limited and has lock positions, too. Various car functions are switched in these lock positions. The switch control resiles to its design position if it is pulled out and rotated into the lock position, where no translation is possible.

## 4. Direct manipulation of constrained objects

Whenever the user interacts with real car components the collision of parts of the hand with the objects is a pre-condition. However, during interaction with the virtual car interior a simplified mapping can be used when the user has grabbed an object. User input leads to a reaction of the car components that is strongly influenced by constraints (*cf.* Figure 5). The implementation of these three aspects of interaction is described here.

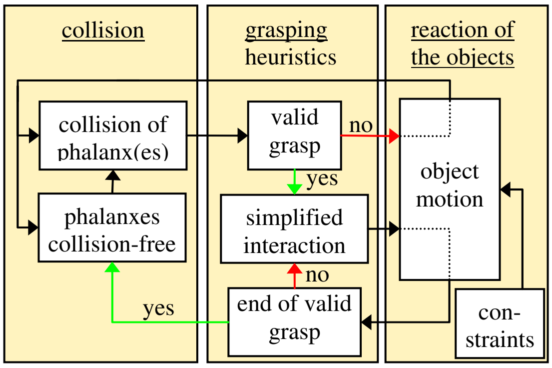


**Figure 5:** *Schematic view of the application flow.*

### 4.1. Collision-based interaction

In reality the hand collides with the objects and applies forces to them, causing a motion of the assembly. Relevant for this issue are the points where the collision occurs at first and the motion of the hand afterwards.

The collision detection of a virtual representation of the user's hand and the virtual objects is performed by a collision module of the VR system. When a collision of one part of the hand with an assembly in one frame is encountered, it has already penetrated the object since the detection works on a per-frame basis. For computing the exact collision point of the hand and the assembly, it would be necessary to interpolate between position (and orientation) of the hand phalanx and the object in the previous and current frame. This is rather complex and computationally expensive. We decided to use a simplified solution. The position of the colliding phalanx in the frame before collision acts as the starting point. The vector of motion is the distance between current and previous position (*cf.* Figure 6).

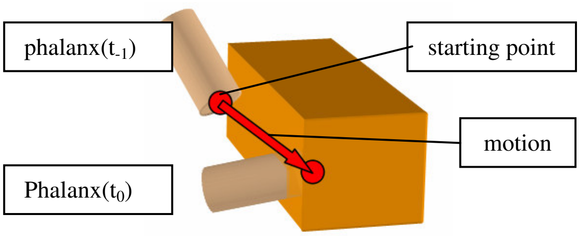


**Figure 6:** *Simplified starting point and motion vector determination.*

With this simplification an error is introduced that causes in general a slightly larger motion than necessary and therefore the object moves slightly away from the hand, which is not noticeable in most cases (*cf.* chapter 7.1).

### 4.2. Grasping heuristics

Because of the missing haptic feedback we can not be sure that the users are able to hold up collision between their hand and the virtual object during interaction. When turning the steering wheel for example, it is likely that users can not follow the motion of the wheel rim all the time. To avoid these problems, we detach interaction from collision by introducing a grasping heuristic. When the user has grabbed an assembly, interaction can be simplified because the collision does not have to be valid during the whole interaction process and it can be assumed that the user is behaving in a typical way for this grasp. Only the phalanxes involved in the detected grasp have to be considered.

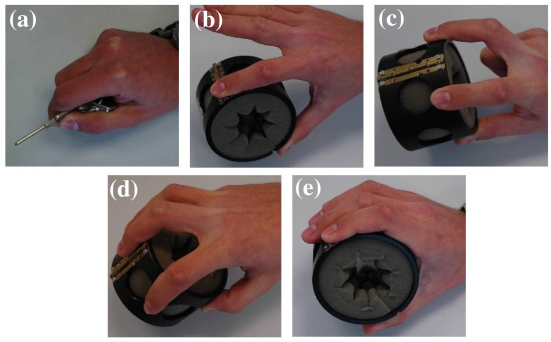


**Figure 7:** *Typical grasps in a car interior: Key Grasp (a), Tip Pinch (b), Three-Point Tip Pinch (c), Three-Point Pinch (d), Cylinder Wrap (e).*

As mentioned before only a few grasps are used in the car interior:

- With the **Key Grasp** the user is clamping an assembly between the tip of the thumb and the middle phalanx of the index finger similar to holding a key before moving it into a lock (*cf.* Figure 7a).
- For the **Tip Pinch** the tips of thumb and index finger are used to clamp an object (*cf.* Figure 7b). A Tip Pinch of thumb and middle finger is also possible but ignored here, due to its infrequent use.
- The **Three-Point Tip Pinch** is quite similar, but thumb, index and middle finger are used (*cf.* Figure 7c).
- For the **Three-Point Pinch** the middle phalanxes of the same fingers are employed (*cf.* Figure 7d).
- The **Cylinder Wrap** is the only grasp using the palm. It is called wrap because the whole hand is wrapping the object. The shape of the object, which also defines the shape of the hand, is a cylinder (*cf.* Figure 7e).

We are trying to detect these five grasps during user interaction with the help of grasping heuristics. Therefore the grasping conditions are checked in each frame. The heuristically determined conditions are:

- simultaneous collision of all involved phalanxes with the object (palm, tip of thumb, index and middle finger for Cylinder Wrap)
- assembly located between the involved phalanxes
- the detected grasp is allowed for this assembly

To check whether the object is present between the phalanxes a ray test is used. This test checks if a ray pointing from one phalanx to another is intersecting the object. Because of these simple conditions different grasps can be valid at the same time. For this case we introduce a grasping hierarchy, which prioritizes grasps with higher stability. The Cylinder Wrap ranks highest followed by Three-Point Tip Pinch and its middle phalanx derivative. The Tip Pinch is slightly more stable than a Key Grasp, which ranks the lowest.

Once a valid grasp is detected the interaction is detached from the collision and performed until one of the follwoing two break conditions is fulfilled:

- significant increase of the distance between the involved phalanxes (*cf.* Figure 8a)
- significant increase of the distance of the hand to the barycenter / the rotation point of the object (*cf.* Figure 8b)

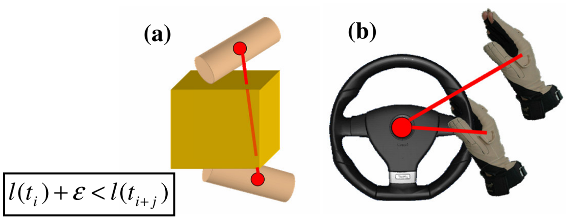


**Figure 8:** *Break conditions for grasping: current distances are checked for significant increase*

The first break condition represents the fact that the user is opening his hand for releasing a grasp. During interaction the hand is usually moved around the rotation center of the object. When users move their hand away they usually want to abandon the grasp as well, which is covered by the second rule. At grab time both distances are stored, increased by a tolerance for robustness and they are compared with the current distances in each frame.

In summary, we base our interactions on the collisions of the hand geometry with the virtual assemblies. We defined a starting point of the collision and the motion the colliding phalanx performed afterwards to determine the corresponding motion of the object. Because of the lack of haptic feedback, we detach the interaction from permanent collision by using grasping heuristics. If a grasp is established the phalanxes involved in the grasp are handled as if they would collide.

### 4.3. Reaction of the virtual objects

In the real car interior the assemblies' reaction on user input is based on physical rules and laws, which we try to approximate in this work. In the real world users apply forces to objects with their fingers. Constraints cause counter forces that are produced by mechanical structures, such as joints or stoppers. All the forces acting on the object are superpositioned. If an object is manipulated with several phalanxes a complex set of forces acts on the object. We observed these basic principles in the real world and derived a simplified model based on geometric calculations and the measured motion of the phalanxes. The rotation and translation of the object is calculated such that it shies away from the colliding phalanxes in the same direction as it would in the real car.

All the assemblies in our car interior are constrained in their motion, which simplifies interaction. As mentioned before the objects can be classified with respect to their constraints. The collision-dissolving approaches are described for each of these groups separately.

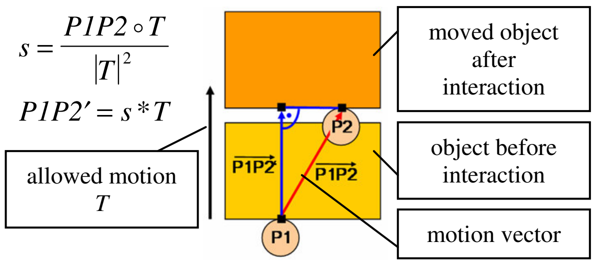


**Figure 9:** *Calculation of the object reaction on single phalanx collision*

**One-dof-translational** objects are moved along a single direction until the collision with the phalanx is dissolved. This can be achieved by projecting the vector describing the motion of the phalanx after collision onto the constraint unit vector. The translation described by this vector (*cf.* Figure 9) might have to be limited by stops.

For multiple finger interactions there are two different cases to consider: two or more fingers push the object into the same direction or they push it into different directions (*cf.* Figure 10). In the first case we simply use largest motion, since it also dissolves all other collisions. For the second case no motion can be found that eliminates collision. Here a compromise has to be found. We let all colliding phalanxes penetrate the object with the same depth. Thus for each colliding finger the caused motion of the object is calculated. The resulting motion vectors are grouped into those that push against the allowed constraint vector and those that point into the same direction. From both groups the largest vector is selected and the resulting motion is the average between the two vectors.

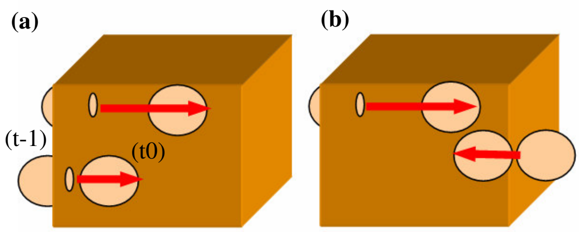


**Figure 10:** *Dissolving of multi-finger collision: fingers that push in the same direction (a) or in different directions (b)*

Because the motion is constrained to a vector the position of the object can be traced by adding up the lengths of each motion applied to the object. With the help of this offset to the design position the other constraints, lock positions and motion range, can be defined and simulated.

**One-dof-rotational** objects can rotate around a defined rotation axis. In this case the motions of a phalanx along the rotation axis have no effect on the reaction of the object. The starting point and motion vector are projected into the interaction plane orthogonal to the rotation axis. Only motions of the phalanx in this plane have to be taken into account for object reaction calculation (*cf.* Figure 11).

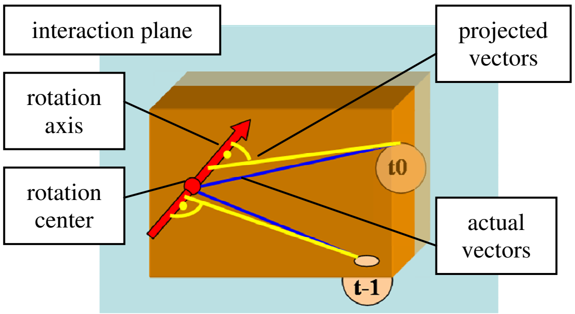


**Figure 11:** *Relevant vectors for object reaction from rotation center to starting point (t-1) and current position (t0) of the phalanx that are coplanar to the interaction plane*

Since the rotation axis is fixed only the desired angle has to be calculated. At first sight it seems that the angle between the vectors projected into interaction plane should be the angle the object has to be rotated with. This approach leads to correct interactions only in the following two cases:

- rotation center and starting point are located in a plane, which is part of the object's surface
- starting point and current position of the phalanx define a circular path around the rotation center

Both cases cannot be assumed in general. The correct angle can be calculated when an auxiliary starting point can be found which satisfies the following conditions (*cf.* Figure 12):

- located on the surface of the object
- same distance to the rotation center as the current position of the phalanx
- a minimal distance to the actual starting point

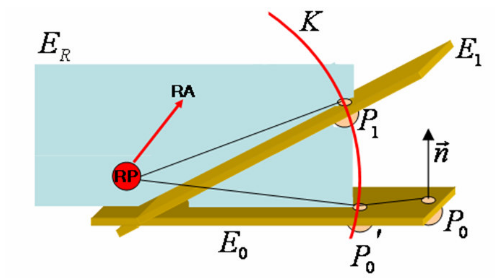


**Figure 12:** *Determination of the virtual starting point necessary for correct angle calculation*

Because of the complex geometries in the virtual car interior, it is non-trivial to find such an auxiliary starting point. We use simply the angle between the projected vectors connecting the rotation center and starting point and current position of the phalanx respectively. The error introduced by the simplification is small in most observed cases and positive and negative errors cancel out to a certain extent.

Further constraints, lock positions and rotation range, can be defined and simulated in the same way as for 1-dof-translational objects. Therefore, the angular offset to the design position is added up during interaction. If the object is grasped by the user and the grasp heuristics accepts the grasp, the previously described error is no longer relevant since collisions are no longer considered.

**Two-dof-rotational** objects are mounted with two saddle joints in a row. Each saddle joint causes a 1-dof-rotational constraint, where the second rotation axis depends on the rotation around the first (*cf.* Figure 13). For each one-dof-constraint the rotation according to the motion of the phalanxes is calculated as described. The motion of the rotation axis defining the constraint for the second saddle joint has to be considered as well. The scene graph of the virtual object is organized such that the dependency of the second rotation is directly represented (*cf.* Figure 13). Each rotation is handled in a separate coordinate system. This approach is a simplification of the real processes, since friction plays a strong role for such configurations.

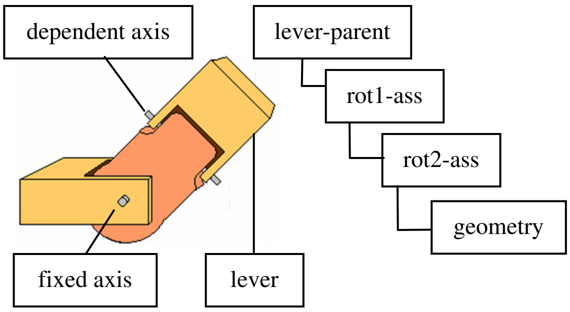


**Figure 13:** *Schematic structure of two-dof joints and their representation in the scene graph*

Interaction with **three-dof-rotational** objects is very complex. As mentioned before not only rotations of the object caused by the individual phalanx motions occur but also rotations that are caused by interactions of the fingers and the object with each other. For this interaction in the real car users dynamically adapt the pressure of their fingers on the object to the reaction of the object. Without haptic feedback it is not possible to simulate this process correctly. From observations in the real car (*cf.* chapter 3) we know that most users grab the interior mirror, which is the only three-dof-rotational object, with one of the pinch grasps on the right hand side of the rotation center (steering wheel assumed to be on the left side of the car). Typically one of the three following rotations are applied to the object (*cf.* Figure 14):

- a rotation performed by all phalanxes together around the rotation center
- a rotation caused by the phalanxes at the upper edge of the mirror around the axis connecting the rotation center and the phalanx at the lower edge
- a rotation caused by the lower phalanx around the axis from rotation center to the upper phalanxes

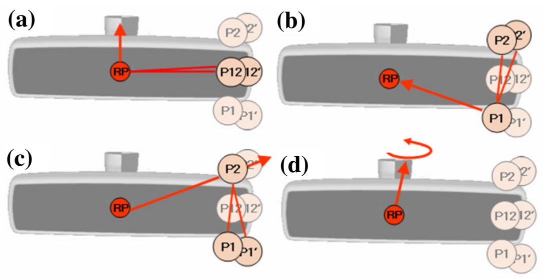


**Figure 14:** *Three partial rotations, one around the rotation center (a) and two caused by the interaction of the phalanxes with each other (b, c), are averaged to one (d)*

In the case of a three finger grasp the index and middle finger are handled as if they were a single finger. This means that both starting point and current position of these fingers are averaged.

Each potential rotation is determined in the same way as the one-dof-rotations. The axis for the second and third rotation is fixed for the mirror. The axis for the first rotation is orthogonal to the plane, which contains the center of rotation, and the previous and current position of the involved phalanx. Instead of trying to detect which of the potential rotations is currently used, we simply average all of them together, which somewhat smoothes the object motion.

The light switch control is the only rotatable and translatable object. This **one-dof-rotational-one-dof-translational** object can be rotated around a single axis, which also defines the translation constraint. Translation is independent of rotation due to this construction. Rotation and translation can be determined in the same way as their single-constraint relatives. Similarly, further constraints can be handled by tracing the angle and motion offset to the design position. The calculated transformations are applied in two different coordinate systems: the rotation node is the parent to the translation node in the scene graph.

The steering wheel – generally a one-dof-rotational object – provides a further interaction possibility that is of particular interest for ergonomics. It can be adjusted, usually both vertically and with respect to the proximity to the driver. Thus the steering wheel adjustment is a one-dof-rotational-one-dof-translational process. The adjustment in the real car is unlocked with a lever that is located at the lower side of the steering column. Thus it is not visible for the user. Moreover, during adjustment the user moves the wheel and column grabbing at the wheel. Usually this is done with both hands in a real car, such that the wheel can be prevented from rotation. In our virtual car interior the rotation of the wheel has to be blocked since two handed interaction is not implemented so far. Because of these two problems we implemented the steering wheel adjustment as a separate interaction mode, which is reached through a menue. In this mode the steering column can be adjusted without rotation of the wheel itself. In interaction mode the adjustment is fixed and the wheel can be rotated.

The steering wheel adjustment interaction is realized similar to the interaction with the light control switch, which is the only other one-dof-rotational-one-dof-translational object. Therefore the same scene graph structure is used. The steering wheel rotation node is a child to the nodes for the steering wheel adjustment. Simulating the range constraints is very important for the ergonomics. We use a non-axis aligned bounding box, which defines the possible range for a reference point on the steering wheel. This is only an approximation of the real range for the steering wheel adjustment, which moves more in a spherical segment.

Some of the car components, such as controls, act as direct switches for car functionality, e.g. the light control turns on the headlights of the car and the lights for the instruments in the dashboard. Others cause more indirect dependent actions, such as opening the door causes the interior light to switch on. These dependent actions are important when simulating the interactions with the car interior.

## 5. Problems and strategies

Due to the challenges of direct interaction in virtual environments a number of problems occur while interacting with the virtual car components. The missing haptic feedback is one of the main sources for interaction failures, since the real hand of the user can not be influenced by the virtual scene. The most important problems are presented here as well as our approach to deal with them.

One failure situation can occur at the stop positions of components. In the real car the user would push the object until it stops. Due to the physical presence of the object the user's hand would stop as well. In the virtual world there is no physical presence of the object. Thus the hand of the user might keep on moving and some phalanxes would become collision-free at the back of the object (*cf.* Figure 15). When the user pulls the hand back new collisions with the object would occur, which would result in an incorrect movement of the object in most cases.

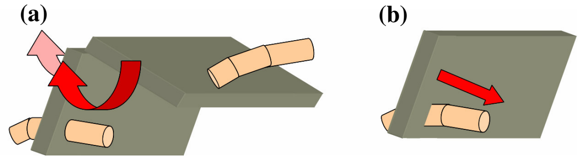


**Figure 15:** *The object stops and some of the phalanxes become collision-free (a). When moving the hand back new collision and motion occur (b).*

We deal with this issue using a test that is performed when a phalanx becomes collision-free. During the initial collision of the phalanx the last collision-free position is stored. Our test checks if the geometry of the collision object is located between this initial collision-free position and the current position of the phalanx. If this is the case, the phalanx has left the object on the back and collisions are ignored until there is a free line of sight between current and last collision-free position.

Even if the phalanx did not become collision-free an interaction failure may occur. If a phalanx is penetrating an object and is pulled back, this motion is applied to the object as well, which is unexpected by the user. This problem can not be solved by the just introduced strategy because the collision remains valid. To overcome these types of problems the direction describing the motion that caused the first collision is stored and transformed with the object (*cf.* Figure 16). During interaction only motions of the phalanx in approximately the same direction are taken into account. Here a threshold of 45 degrees has shown good results.

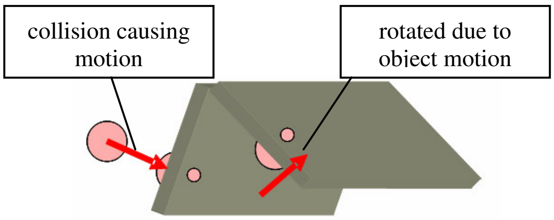


**Figure 16:** *The motion that causes collision is stored and transformed due to object motion*

Another problem can emerge during grasping. Usually the user grabs an object and adjusts it until it is in the desired orientation. When releasing the grasp the user expects the object to keep this state and stop moving. We deal with the grasp release situation by ignoring any motion of the phalanxes until the complete hand becomes collision-free.

## 6. Implementation in Virtual Design 2

We implemented this work as a module of the VR system VD2 by VRCom GmbH. The module consists of four dynamic shared objects (DSOs) which are loaded at application start up time. Each DSO performs several dedicated tasks:

- The **input module** manages the hand geometry, handles collisions and generates collision feedback.
- The **interaction module** handles all interactive objects and their reaction.
- The **menu module** manages the interaction mode selection.
- The **glove module** receives the values defining the hand configuration from the finger-tracking system via the network and maps them on the hand geometry.

These modules are controlled by the VR system using events and callback actions. The classification of the car interior components presented in chapter 3 is implemented as a class hierarchy defining groups of interactive objects (*cf.* Figure 17). In this hierarchy constrained motion of the objects is calculated in common base classes.

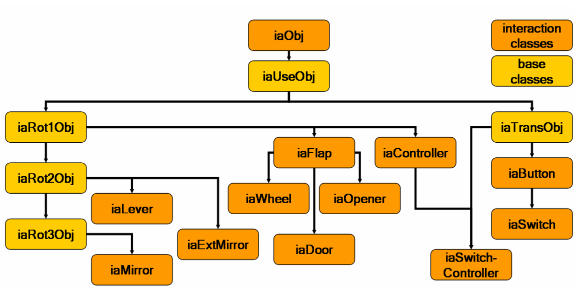


**Figure 17:** *Hierarchy of interactive object classes.*

We used a three-sided CAVE (front, left and lower side) operated with active stereo and CRT projectors for experimenting with our system. The user is tracked with an optical tracking system by A.R.T. GmbH, who also developed the optical finger-tracking system. Our display is driven by a PC cluster consisting of six Evans & Sutherland nodes running RedHat Linux 8.0. In this configuration the whole application runs typically at around ten frames per second.

## 7. Results

We generally observed that our approximation of the physical manipulation of virtual car components works well for simple objects, such as flaps or switches. They can be handled intuitively and realistically without any training. Even more complex objects like doors, the steering wheel and the controls react in an expected way and can be handled with some training. Even the most complex interaction with the interior mirror works well for most users, probably because of the smoothing caused by averaging different partial rotations. In contrast interaction with very small objects like the control knob for the exterior mirrors is error-prone. These small objects are often completely occluded by the user's hand which makes it difficult to judge if a proper grasp has been applied. Problems can arise with objects that are very narrow e.g. the door openers. Here it can happen that the users lose contact with the object if they move too fast. This is mostly avoided if a valid grasp can be detected.

### 7.1. Pilot study

The results of this approach were evaluated in a pilot study, which provides a first idea of the usability of the pseudo-physical interaction. For this initial study six test persons from our lab and related groups were observed while using the application. A short questionnaire was filled out after the tests, which was used to get an impression of the quality of the different interaction classes. The CyberGlove and the new finger-tracking prototype from A.R.T. were used as input devices. This direct interaction was compared with a wand (Flystick) interaction. The wand-based interaction is more indirect, since it uses a virtual cursor. It allows only coarse reachability studies. Usability and ergonomic studies are not possible with the wand. The CyberGlove is used with a virtual representation of the hand, which is slightly offset from the real hand, since the the virtual hand does not match the real hand perfectly well. The A.R.T.-finger tracking system is very precise and would occlude a virtual hand in almost all cases. Thus we switched the virtual hand representation off for the A.R.T finger tracking system.

The users were asked to perform seven tasks with each input devices:

1. Turning the wheel to the left until it stops and to the right until it stops.
2. Adjust the temperature to the warmest and the fan to the highest setting.
3. Adjust the interior mirror.
4. Switch on the emergency flashers.
5. Open the glove box and close it.
6. Open the left door and close it.
7. Flap the sun shield down until it stops and flap it back to initial position.

The observation of the users during interaction showed – as expected – that the main problem is the missing haptic feedback. The users had problems to correctly estimate the depth of their real phalanxes in the virtual scene, due to occlusion and focus and convergence problems. Moreover, some users were not able to recognize if they grabbed an object properly. This is especially true for the A.R.T.-glove where we did not provide a visual feedback of the virtual hand. With the Flystick interactions are explicitly initiated and ended with a button. The Flystick interaction was rated best, since it provides good control, but the users agreed that it is not the best device for usability and reachability assays. They criticized the lack of realism and the offset of the virtual cursor. The comments and suggestions for the A.R.T.-glove mainly aimed at a better feedback for collisions and grasping. For this input device, it has to be taken into account that we just used a left handed early prototype of the system and many interactions are typically performed with the right hand.

We also observed that after a short time of familiarization our users were able to interact with most virtual parts. The errors introduced by the simplifications in our approach were not noticed or reported. However, it is still difficult for the users to estimate the depth relationship between the real hand and the objects of the virtual scene on a purely visual basis.

## 8. Summary and Future work

We developed a set of pseudo-physical interaction metaphors, which are based on observations of interactions in a real car interior. The different parts in the car interior are classified with respect to their constraints, which led to the development of appropriate direct manipulations techniques considering geometric constraints and additionally stop and lock positions. Our approach uses a hierarchical grasp heuristic to decouple the interaction from the collision of the fingers with the virtual car components. This approach makes the interaction more robust while no haptic feedback is available. A pilot study of our implementation revealed that our direct manipulation techniques are a good approximation of the real world behavior of car components.

The lack of haptic feedback is one of the major difficulties for implementing pseudo-physical interactions. We are planning to attach tactile actors to the finger tips to provide some feedback for the collision of the hand with the virtual world and for the proper grasps of virtual parts.

It also became clear that the implementation of true two-handed interactions is necessary for a realistic simulation of the car interior. As soon as the optical finger tracking will become available for the right hand, we will integrate it with our system and perform an extended user study to fully assess the applicability of our approach for projection-based virtual environments.

The integration of a rigid body simulation and a constraint engine will simplify the handling of some interior parts at the expense of increased computational costs. However, our pseudo-physical simulation is already a promising step towards virtual assays in the area of reachability and usability studies.

## 9. References


[Bar97] BARAFF D.: Rigid Body Simulation. *Course Notes of ACM Siggraph '97,* Los Angeles (CA), 1997

[BH02] BORST CH., HIRZINGER G.: Calculating Hand Configurations for Precision and Pinch Grasps. *Proceedings of IEEE / RSJ Int'l Conference on Intelligent Robots and Systems,* Lausanne (CH), 2002

[BI05] BORST CH., INDUGULA A.P.: Realisitic Virtual Grasping. *Proceedings of IEEE Virtual Reality (VR2005),* Bonn (GER), 2005

[BKLP04] BOWMAN D. A., KRUIJFF E., LAVIOLA JR. J. J., PUOPYREV I.: 3D User Interfaces. Theory and Practice. *Addison-Wesley,* Boston (MA), 2004

[DKL98] DAM E.B., KOCH M., LILLHOLM M.: Quaternions, Interpolation and Animation. *Technical Report (DIKU-TR-98/5), Department of Computer Science, University of Copenhagen*, Copenhagen (DK), 1998

[Hil05] HILLEBRAND G.: Invers-kinematisches infrarotbasiertes Hand- und Fingertracking (German). *Diploma Thesis at University of Munic,* Munic (GER), 2005

[KI91] KANG S. B., IKEUCHI K.: A Framework for Recognizing Grasps. *Techn. Rep. CMU–RI–TR–91-24 Carnegie Mellon University,* Pittsburgh (PA), 1991

[MBS97] MINE M. R., BROOKS JR. F. P., SEQUIN C. H.: Moving Objects in Space: Exploiting Proprioception in Virtual-Environment Interaction. *Proceedings of ACM Siggraph '97,* 1997

[MT94] MAS SANSO R., THALMANN D.: A Hand Control and Automatic Grasping System for Synthetic Actors. *Computer Graphics Forum 13, 3, 1994,* pp. 167-177, 1994

[Mul98] MULDER J. D.: Remote Object Translation Methods for Immersive Virtual Environments. *Presented at the Virtual Environments Conference '98 and 4th Eurographics Workshop,* 1998

[Ull99] ULLMANN T.: Ein heuristisches Verfahren fuer virtuelles Greifen mit dem Datenhandschuh (German). *Diploma Thesis at University of Ilmenau,* Ilmenau (GER), 1999

[ZR01] ZACHMANN G., RETTIG A.: Natural and Robust Interaction in Virtual Assembly Simulation. *Proceedings of 8th ISPE Int'l Conference on Concurrent Engineering: Research and Applications (ISPE/CE2001),* 2001