Virtual Reality (2018) 22:339–351
https://doi.org/10.1007/s10055-018-0337-4

ORIGINAL ARTICLE

An evaluation of multimodal interaction techniques for 3D layout constraint solver in a desktop-based virtual environment

Marounene Kefi[1] · Thuong N. Hoang[2] · Paul Richard[1] · Eulalie Verhulst[1]

Received: 25 September 2014 / Accepted: 13 February 2018 / Published online: 24 February 2018

Abstract

We propose a new approach to the 3D layout problems based on the integration of constraint programming and virtual reality interaction techniques. Our method uses an open-source constraint solver integrated in a popular 3D game engine. We designed multimodal interaction techniques for the system, based on gesture and voice input. We conducted a user study with an interactive task of laying out room furniture to compare and evaluate the mono- and multimodal interaction techniques. Results showed that voice command provided the best performance and was most preferred by participants, based on the analysis of both objective and subjective data. Results also revealed that there was no significant difference between the voice and multimodal input (voice and gesture). Our original approach opens the way to multidisciplinary theoretical work and promotes the development of high-level applications for the VR applications.

Keywords Interaction techniques · Constraint solver · 3D layout · Virtual environments · Multimodality

1 Introduction

Constraint programming (CP) is employed to model constraints, propose possible solutions of complex problems, and assist the user in the layout task. CP is often applied to solve constraint satisfaction problems (CSPs) and provides resolution methods and techniques for problems defined with discrete and continuous variables. CP has been successfully used in several areas such as planning, scheduling, and resources allocation (Régin 2004; Rossi et al. 2006; Goel et al. 2015).

A layout problem generally involves the task of arranging a set of components, such as furniture or devices, while respecting geometrical and functional constraints. An example is the layout of electronic components with such functional constraints as heating, consumption, and electronic compatibility. The 3D layout problems are usually solved manually without any user's assistance and using tools that are limited in terms of 3D visualization and interaction. Therefore, expertise of the designers is required to assess and validate the final design.

Virtual reality (VR) is a powerful tool for interactive visualization and immersion of the user in 3D virtual spaces (Cal et al. 2016; Seth et al. 2011). In the context of a layout problem, VR can be efficiently used to visualize the 3D space and to allow users to interact with the objects. However, VR systems do generally not provide user with assistance for 3D layout tasks and the objects have to be selected, picked, and placed manually, which is generally tedious.

In order to provide more intuitive and natural interaction techniques for selection and manipulation of virtual objects, multimodal approaches using speech and gestures have been proposed (Bolt and Herranz 1992; Zhao and Madhavan 2005; Chun et al. 2015). However, multimodal interaction is underexplored in the context of user assistance systems such as constraint solver.

In this paper, we propose a new approach to solve 3D layout problems by integrating desktop VR techniques and constraint programming. We aim toward the development of a decision-making system that proposes feasible configurations of 3D spaces given a set of objects and constraints. Our approach contains an intelligent module to model and solve constraint-based problems. The contribution of our paper is the empirical evaluation of multimodal interaction

✉ Thuong N. Hoang
thuong.hoang@deakin.edu.au

[1] LARIS, University of Angers, Angers, France

[2] School of Information Technology, Deakin University, Melbourne, Australia

techniques in a virtual environment with digital user assistance compared to a single modal approach.

The remainder of the paper is organized as follows: The next section provides a survey of the related work concerning 3D layout problems and constraint solvers, interaction techniques, and multimodality. Section 3 presents an overview of the proposed system, including description of the supported constraints that can be used to layout 3D scenes. In Sect. 4, we describe our user experiment comparing two monomodal and multimodal interaction techniques. Section 5 concludes the paper and discusses directions for future work.

2 Background

2.1 3D layout problems and constraint solvers

Different approaches involving layout problems have been developed. For example, Xu et al. (2002) addressed the combination of physics, semantics, and placement constraints for scene layout The authors proposed a new modeling technique where users can create scenes by specifying the number and distribution of each class of object to be included in the scene. In a similar way, Sanchez et al. (2003) presented a general-purpose constraint-based system for non-isothetic 3D-object layout built on a genetic algorithm. This system is able to process a complex set of constraints, including geometric and pseudo-physics ones. The paper focuses mainly on the algorithmic contribution with textual constraints, without the ability for interactive user intervention.

Calderon et al. (2003) presented a novel framework for the use of VEs in interactive problem solving. This framework extends visualization to serve as a natural interface for the exploration of configuration space and enables the implementation of reactive VEs. The implementation is based on a fully interactive mechanism where both visualization and the generation of a new solution are under the control of the user. Tutenel et al. introduced a novel rule-based layout solving approach which is suited for use in conjunction with procedural generation methods (Tim et al. 2009). Authors showed how this solving approach can be used for procedural generation by providing the solver with a user-defined plan. In this plan, users can specify objects to be placed as instances of classes, which in turn contain rules about how instances should be placed. Being procedurally generated, user intervention in the system is minimal. The initial user-defined plan is two-dimensional, which does not provide visual feedback for the layout solutions. There is no mechanism for the user to adjust the layout afterward.

Medjdoub proposed an interactive system for ceiling mounted fan coil system in a building ceiling void (Medjdoub 2004). He used a hybrid approach combining case-based reasoning and CP techniques. The system is used with interactive modification of the 3D parametric model. The author has shown the potential to significantly reduce design costs by reducing design time by 50%, improve the quality of the solution, and produce additional benefits elsewhere in the supply chain.

The user interaction in the approaches mentioned above lacks an intuitive and suitable interaction interface. They all focused on solving the layout problem without proper investigation of the interactivity between the user and the developed system. The use of other input modalities, such as speech interaction, has not been considered in previous approaches (Sanchez et al. 2003; Calderon et al. 2003; Fages et al. 2004; Jacquenot 2009). Our research investigates the use of voice command for 3D layout assistance in a virtual environment.

2.2 Voice-based interaction

In VR applications, the interaction technique and the proposed protocol must be appropriate for the task to be performed (Jacob et al. 1994) and utilize multiple modalities that is considered natural interaction for the user.

Voice-based interaction techniques were introduced for the first time by Bolt and Herranz (1992) through the *Put-That-There* application. Voice-based interaction is mainly used in multimodal interfaces in combination with other interaction modalities (Turk 2014).

Kulyukin (2004) presents both practical and scientific arguments for human-machine voice communication. Pires (2006) outlines the usefulness of voice communication in industrial robotic cells, where humans and robots safely share the workspace.

Voice interface is one of the most natural and intuitive ways to facilitate interaction between human and machines (Billinghurst 1998). It has been embedded into virtual environments and CAD systems to support free-hand interaction and mobility, which are major advantages over classic interaction devices.

Research on voice-based interactive devices or interfaces has covered many application domains, including industrial applications, manufacturing, and education. Vacher et al. explored the usage of voice-based interaction for home automation (Vacher et al. 2013). The system uses speech recognition to target a range of user population, including seniors, visually impaired users, and users with no special needs. Rogalski and Wielgat (2010) detailed the realization of a speech recognition-based aircraft control system for general aviation aircraft. Through a set of voice commands, the proposed avionic system supports direct control of the flight of the aircraft. Ku et al. (2013) developed a voice-based interaction for in-vehicle interface. Their study suggested that speech interaction modality reduces mental workload for drivers behind the wheels.

Rogowski (2012a) combined automatic speech recognition and web-based remote control to develop an effective tool for remote voice control of industrial robotized cells. The author discussed the prerequisite conditions to provide an effective and faultless voice communication.

In manufacturing, Weyrich and Drews (1999) integrated speech recognition, synthesis, 6D pointing pens, and data gloves in their virtual manufacturing environment. The speech-based module was defined by a 50-word vocabulary. Garcia et al. (2010) introduced the “Voice Interactive Classroom,” a software solution that proposes a middleware approach to provide cross-platform multichannel access to internet-based learning. Their research showed that visual and auditory e-learning can be achieved by adapting visual-only learning into naturalistic voice dialogs. Using a service-oriented middleware, voice dialogs can easily be reused and integrated within a heterogeneous set of e-learning platforms.

Voice-based interfaces are also found in modeling applications. Gao et al combine voice-based command and direct 3D manipulations using 3D mouse and menu in a semi-immersive environment (Gao et al. 2000). A voice command module was used to activate different 3D actions, such as translation and scaling. Bolt and Herranz employ a combination of voice input, bimanual gesture, and eye tracking approaches to manipulate 3D solids (Bolt and Herranz 1992). Object manipulations are supported through simple expressions like “turn the block” to activate the rotation of the object that the eyes are looking at.

In many application domains, especially for industrial systems, a high level of accuracy from voice command is required. A poor speech recognition accuracy can lead to erroneous actions to be performed by robots and/or machines. Rogowski (2012b) defined a set of requirements to ensure reliability and effectiveness of speech communication.

There are some limitations to voice communication. Users are required to familiarize themselves with the voice commands that must be correctly uttered to activate a specific action. Therefore, voice-based interaction could impose a high cognitive load on users, limiting the freedom and ease of use of the system. Other practical issues include external noise and recognition latency (Laviola 1999) or recognition error (Azenkot and Lee 2013) that could cause further frustration for users. We aim to compare and evaluate voice command-based interactions in order to identify the more intuitive and effective control of our constraint solver-based system, especially with regard to cognitive load.

3 System overview

We propose a real-time 3D environment developed using Unity3D game engine (version 3.2.0f4), commonly used for the development of 3D games and VEs (Unity 2013). The system supports data exchange with the open-sourced Gecode constraints solver (Schulte et al. 2013) using C# scripts.

3.1 Problem formulation

The modeling of 3D layout problems requires an efficient and well-structured formalism. CP is particularly appropriate for the resolution of planning problems, as demonstrated by Honda and Mizoguchi (1995) and Pfefferkorn (1975). Moreover, CP techniques allow modeling with respect to physical constraints (see Sect. 3.2) for which the manual satisfaction without a solving engine is very difficult. We use CSP formalism (constraint satisfaction problem), which is a simple and formal framework for representing and solving a constraint satisfaction problem. In our problem domain, the unknowns are the 3D positions of objects (such as room furniture) and the layout constraints are relations or restrictions among the objects. The resolution of a such CSP consists in assigning values to the variables (unknowns) while satisfying all the constraints. Algorithms making it possible to solve a CSP are called constraints solvers. The process of solving a CSP is to specify the variables and constraints to the solver, which will provide the solutions to the user.

To formulate our problem in CSP form, we suppose that a virtual environment of certain dimensions (w, h, d) is composed of n objects related by m constraints. Let X be the set of unknowns of the problem (3D positions of objects), $\mathcal{D}$ be a function that associates a domain (authorized values) to each variable (x_i, y_i, z_i), and C be the set of layout constraints. Thus, the problem can be defined by the triplet $(X, \mathcal{D}, C)$:

- $X = \{x_1, y_1, z_1, \dots, x_n, y_n, z_n\}, (x_i, y_i, z_i / i \in [1, n])$ position (center) of $object_i$
- $\mathcal{D}(x_i) = [w_i/2, w - w_i/2], w_i/i \in [1, n]$ is the width of $object_i$
  $\mathcal{D}(y_i) = [h_i/2, h - h_i/2],\ h_i/i \in [1, n]$ is the height of $object_i$
  $\mathcal{D}(z_i) = [d_i/2, d - d_i/2], d_i/i \in [1, n]$ is the depth of $object_i$
- $C = \{c_1^{i,j}, c_2^{i,j}, \dots, c_m^{i,j}\}, c_k^{i,j}/(k \in [1, m]$ and $i, j \in [1, n])$ is a constraint between $object_i$ and $object_j$.

3.2 Supported constraints

In our approach, we developed specific constraints related to a layout problem proposed by an industrial partner.

Physical constraints have been developed to take into account for the possible effect of certain objects on some others.

- **Geometric constraints**
  -*No_overlapping constraint*: Objects should not overlap.
  -*Min_distance constraint*: This constraint can be applied to at least two selected objects. The solver ensures that the selected objects are separated by a minimum distance which can be set by the user. The same principle is used for the *Max_distance and Fixed_distance constraints*.
  -*Surround constraint*: The solver looks at positioning n objects around a central one (obj_0: the first one selected) through an association of the *Min_distance* and *Max_distance constraints* (a default distance is set for each constraint): $Max\_distance(obj_0, obj_i, \_dmax)$ and $Min\_distance(obj_i, obj_j, \_dmin)/i,j \in [1, n-1]$
  -*Right constraint*: It can be used to place the first selected object at the right of the second one. The same principle is used for *Left*, *Front* and *Back constraints*.
  -*Object_on_object constraint*: It can be applied to two objects; the second selected object being placed on top of the first one.
- **Physical constraints**
  They are based on the laws of physics and have been developed to minimize the effect of electric and magnetic fields on electro- or magneto-sensitive objects. After adding an object in the 3D scene, the user should specify to what extend the added object emits an electric or magnetic field by entering the value of the electric charge $\boldsymbol{q}$. For the magnetic field, the user should also specify the speed $\boldsymbol{v}$ of $\boldsymbol{q}$.

3.3 User interaction

The user interacts with the resolution system by selecting 3D objects (such as chairs, desks, or sofas) via a menu (see Fig. 1). Another menu allows them to select geometric and/or physical constraints to be applied to the selected objects. Each selection is a toggle button to indicate which constraints are currently being applied. At any time, the user can trigger the resolution of the layout problem. A CSP is created, in which the variables are the 3D positions of the centers of the objects and the constraints are those selected by the user. The solver calculates feasible configurations within a timeout period (5 s) to solve the CSP. The complexity of the problem is based on the number of objects and the nature of the constraints. Once the solving of the CSP is completed, the list of found solutions is transmitted to the VE, which updates the 3D object positions according to the new values of object centers. The user can step through the solutions on the list by pressing the left and right arrows on the interface. The system will reorganize the objects on the screen based on the current solution.

Contrary to some previous works (Sanchez et al. 2003; Xu et al. 2002; Fernando et al. 1999; Fages et al. 2004), our system enables the user to intervene and change the proposed 3D configuration according to his/her preferences. Two cases are possible:

- The user can increase the problem size by inserting new 3D objects and selecting new constraints. In this case, the system creates a new CSP by adding new variables (positions of the new objects) and constraints. The solver deals this new CSP to find a new solution;
- The user can manually modify the location of certain objects. In this case, the solver adds new constraints to the moved objects and updates the already existing CSP by setting the new 3D positions of the displaced object. If the

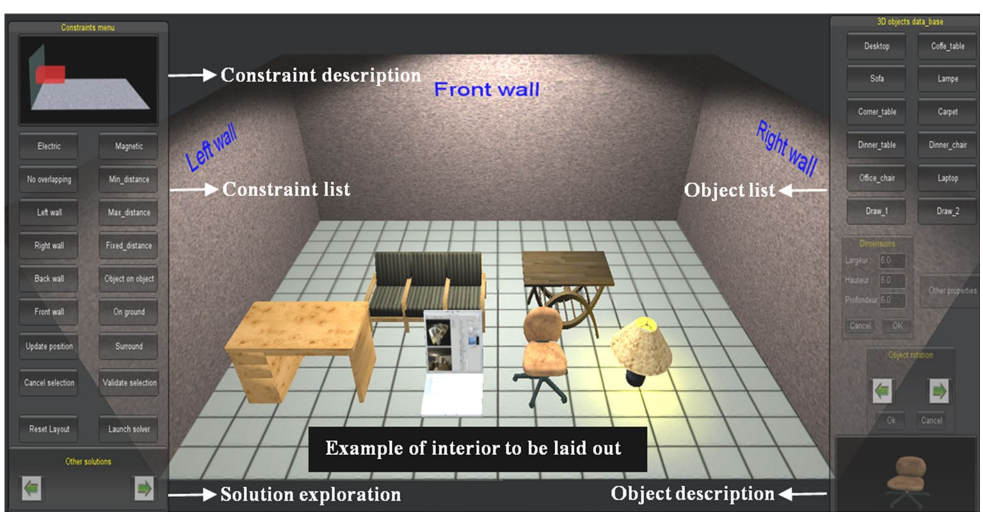

Fig. 1 User interface of the application: list of constraints on the left and the list of objects on the right. The virtual scene is in the center of the interface

CSP failed to find a solution, the system cancels the user action and replace the object in their last correct positions.

Our system is based on a set of algorithms, which are defined and illustrated in Kefi et al. (2012), to accomplish these functionalities. In this paper, we focus on the identification of a suitable interaction for the user to communicate with our system.

The communication between the user and the solver is completed through the VE. Each user action, such as objects manipulation and launch of resolution, generates an event in the form of input queries sent to the solver to update the current CSP. Calculations of the solver are transmitted to the VE to update or rearrange the 3D scene. The *Manager class* manages the creation, transmission, and execution of the events and queries.

Unity3D physics engine is used with collision detection of 3D objects to enhance realism. This application of a physics engine is to improve the user interaction and has no effect on the resolution mechanism.

3.4 User assistance

Although previous works proposed interesting approaches to solve layout problems using a constraint solving engine, most of them (Sanchez et al. 2003; Xu et al. 2002; Fernando et al. 1999; Jacquenot 2009; Fages et al. 2004) did not focus on user assistance during interaction with the 3D scene to be laid out. As described before, our system allows resolution of a 3D layout problem by proposing feasible configurations to the user. In addition, the system is able to cancel any manual object's displacement if at least one constraint is not satisfied. In most approaches, the provided assistance generally occurs once the objects placement is achieved. In order to have a more efficient system, we decided to provide a predictive real-time visual assistance (similar to the ***look ahead*** mechanism) which occurs before the manual placement. The user selects a desired object obj_i and enables the visualization of impossible 3D zones for this object. By splitting the 3D scene in many zones, the solver identifies and transmits the impossible ones to the VE for visualization in form of 3D red cubes. Impossible zones for obj_i are the areas in which obj_i cannot be placed because of constraints violation. More details of this user assistance are described in Kefi et al. (2012).

4 User study

4.1 Motivation and objective

The integration of 3D models in a VE allows simulating and assessing different layouts problems. A suitable hardware configuration consisting of interaction interfaces and devices is required to support user interaction. Through these interfaces and devices, the user is able to control the constraints solver and interact with virtual entities in a natural, fast, and efficient manner.

An interaction interface greatly influences the perception of the VE, which subsequently affects users performance in the layout task in terms of time and accuracy. Moreover, users' appreciation and satisfaction of the proposed system are highly dependent on the relevance of the proposed interface. Therefore, we are interested in evaluating the interaction interfaces across a 3D layout task that required selection, manipulation, and pointing, using our proposed system.

The study investigates the usability and the effectiveness of the three interaction devices currently proposed to interact and control the resolution system: a classic interaction device using a standard mouse, a voice command interface, and a combined input modality of voice and mouse. We define usability as a subset of quality in use consisting of effectiveness, efficiency, and satisfaction by the users of the application (ISO/IEC 2011).

The focus of our comparison is user performance in terms of task completion time, workload and their influence on the learning process, subjective users assessment, and user preference.

Our hypotheses are: H1. There is a measurable difference in task completion time among the three interaction mechanisms. H2. There is a measurable difference in work load among the three interaction mechanisms. H3. There is a measurable difference in user preference among the three interaction mechanisms.

The outcomes of this study will inform the design of the appropriate interaction technique that is intuitive and easy to use for 3D layout problems. We explored both monomodal and multimodal approaches. For monomodal interaction, we are interested in voice-based interaction for a natural and intuitive interface. The proposed experiment compares voice-based interaction with the mouse as the most common monomodal technique for desktop application. The mouse is also a representative of handheld input devices for virtual environment. One of our hypotheses is that the mouse can be tedious and time-consuming when dealing with a large number of objects and various constraints in the layout system, due to increased movements. On the other hand, there are several drawbacks affecting the efficiency of speech recognition, such as environmental noise. Users are also required to familiarize themselves with the voice commands that need to be correctly uttered to activate a specific action. For these reasons, we decided to combine both the mouse and the voice interface to interact with the VE, representing a multimodal approach.

Our study uses projection screen instead of a head-mounted display system. Previous research has proven that the usage of projection screen is effective as an alternative to head-mounted display for virtual environments (Patrick et al. 2000). Therefore, our study findings can be applicable in a VR context.

5 Methods

5.1 Population

Twelve participants (6 males and 6 females) aged between 24 and 49 years took part in this experiment. All of the participants were students at the University of Angers and were familiar with a classic interaction technique of mouse and keyboard on a daily basis, but had not previously used speech interface or other 3D manipulation techniques. They all had similar experience of using virtual reality applications, as part of their course work.

5.2 Task

The participants were required to complete a 3D layout task, consisting of laying out a set of five workstations and two cabinets in a 3D space by controlling the constraints solver in accordance with a set of predefined constraints.

For $i \in [1, 5]$:

- Put the computer$_i$ on the desktop$_i$ $\Longrightarrow$ *constraint Object_on_object*;
- Place the chair$_i$ behind the desk$_i$ at a fixed distance $\boldsymbol{d_f}$*ixed* $\Longrightarrow$ *constraint Distance_fixed* + *constraint Behind*;
- Place the cabinet$_i$ in the corner between the left and the front walls $\Longrightarrow$ *constraint Left* + *constraint Front*;
- Place the cabinet$_i$ in the corner between the right and the front walls$\Longrightarrow$ *constraint Right* + *constraint Front*;
- Place the office$_1$ and office$_2$ against the right wall $\Longrightarrow$ *constraint Right*;
- Place office$_3$ and office$_4$ against the left wall $\Longrightarrow$*constraint Left*;
- Place office$_5$ against the front wall $\Longrightarrow$ *constraint Front*;
- Separate all offices by a minimum distance $\boldsymbol{d_m}$*in* $\Longrightarrow$ *constraint Distance_minimum*.

Each workstation was composed of a desk, a chair, and a computer (see Fig. 2b). An number ID was given to each object to facilitate manipulation. For example, office$_i$ was associated with chair$_i$ and the computer. All the 3D objects were randomly placed in the 3D scene at the beginning of the task (see Fig. 2a). The goal was to layout the scene as shown in Fig. 2b, and the task was performed using three different interaction techniques (two monomodal and one multimodal techniques).

5.3 Questionnaires

To measure the workload during the experiment in VR, the NASA Task Load Index (NASA-TLX) (Hart and Staveland 1988) questionnaire was used. The assessment of workload level caused by the 3D layout task is measured through six indices without weighting: mental demands (MD), physical demands (PD), temporal demands (TD), own performance (OP), effort (EF), and frustration (FR).

The participants were also asked to rate the intuitiveness of the interaction techniques, based on a seven-point Likert scale, with 1 for "very easy" and 7 for "very complex" to assess the interaction technique usability.

The questionnaires aimed to evaluate the benefits of the interaction techniques and the user preferences.

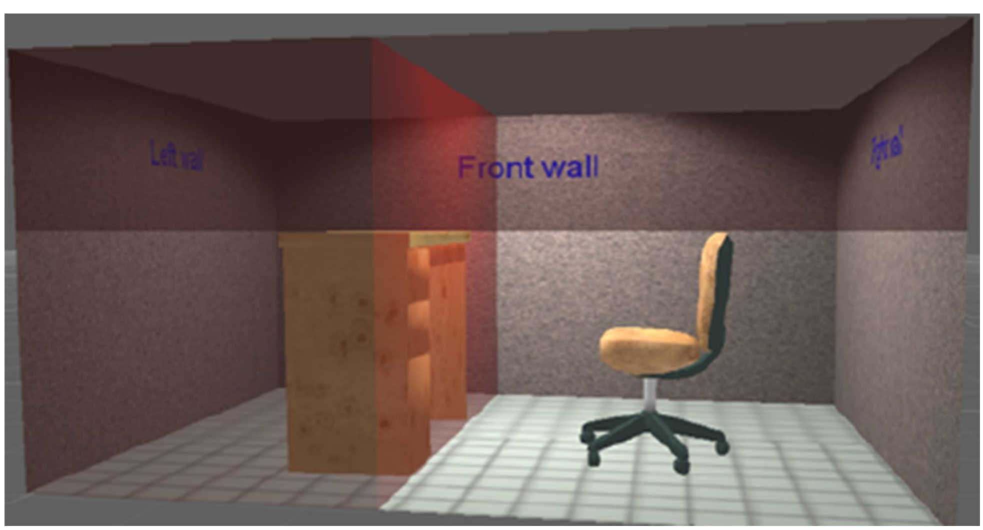

Fig. 2 Impossible zone highlighted as red (color figure online)

5.4 Procedure

The experiment was performed on a computer with *Intel*^{TM} i7-frequency 2.00 GHz processor. The refresh rate of the projector was 120 Hz. The VR interface is projected on a wall in front of the participants at 1.5 m to increase immersiveness.

After a training session, the participant performed the task three successive times with a rest period of 30 s between each experiment condition. The order of the conditions is randomized. After each experimental condition, the participants were asked to complete subjective questionnaires and were asked to indicate their preferred technique, based on their personal preferences, and to justify their choice.

5.4.1 3D layout using WIMP interaction technique: mouse condition

The participants used a standard desktop mouse to select the objects in the 3D scene and chose the constraints from the menu displayed on the left of the screen (see Fig. 3a). The mouse could also be used to change the view of the scene during the layout, which can be necessary to select objects that may be occluded in the scene. View changing was completed by pressing and holding down the right button to pan around the scene and the middle button to rotate the viewpoint. This was introduced to the participants during training. This process was repeated for each association of object(s)–constraint(s). The launch of the resolution (or the solver call) could be triggered by a button on the constraints menu. For this experiment condition, the participants were sitting on a chair in front of the projected wall (see Fig. 4). Other interaction techniques and/or input devices for virtual reality, such as pointer, joystick, wand, hand tracking system, use similar gestures, and modality as the mouse. Jaimes and Sebe (2007) define pointing device as a separate category to human senses in the classification of multimodal interaction, which justified our approach of adding voice as an additional modality. Therefore, we use this condition as a representative of pointer-based handheld input techniques/devices for virtual environment. The experiment focuses on the different modalities of input.

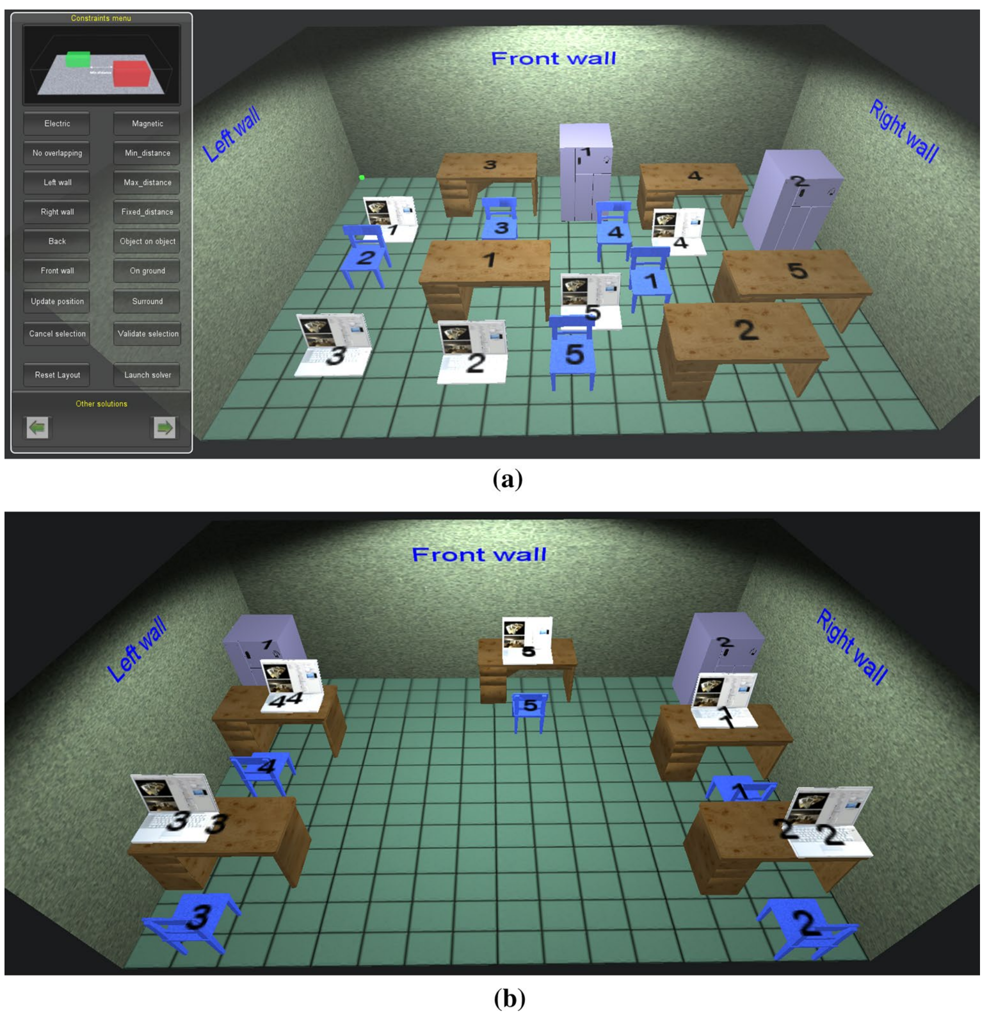

Fig. 3 Initial configuration of the scene (**a**) and the arrangement to achieve (**b**)

Fig. 4 A subject selecting objects and constraints using the mouse

5.4.2 3D layout using voice commands

For this condition, the participants used voice command to select objects and constraints and to call the solver (see Fig. 5). This condition represents the second monomodal interaction. While standing, the participants spoke the name and ID of an object to select and then spoke the name of the constraints to apply on the selected object.

To facilitate objects selection, participants could identify an object simply by speaking the first letter of its name followed by its ID. For example, "C1" for the chair$_1$ and "O$_2$" for the office$_2$. We implemented the voice input interface

Fig. 5 A subject selecting objects and constraints using voice commands

using Microsoft Speech Application Programming Interface (SAPI) 5.1. A small set of vocabulary was implemented to ensure an efficient recognition rate, since it is faster to recognize two letters than a word or phrase. Table 1 outlines the vocabulary used to interact with the VE.

5.4.3 3D layout using both mouse and voice command

For this condition, the participants were seated in front of the projected wall and selected objects by voice commands and chose the constraints using the mouse. The selection of objects was based on the vocabulary given in Table 1. This implementation is based on our observation that it seems easier to memorize the name of an object than that of a constraint. In this study, we focused on this assumption, but we plan to test the other assumptions in which objects are selected by the mouse and constraints by voice commands.

For both conditions, the system only provided feedback upon the calling of the constraint solver. The feedback was display textually at the bottom of the screen, to indicate whether the participants had satisfied all the constraints; if not, the system listed the constraints that were not satisfied or placed incorrectly. The participants performed corrections and called the solver again, until all constraints were met.

5.5 Data collection

We measured the task completion time as an indication of user performance. The task completion time is the time taken by the user to layout the 3D scene according to the configuration given in Fig. 2. The start of the task was triggered when the participants selected the first object and the task ended when the solver was launched.

Table 1 The vocabulary words used to select objects and constraints

Keywords	Associated action
"***Bi***"	Select/deselect office number $i (i \in [1,5])$
"***Ci***"	Select/deselect chair number $i (i \in [1,5])$
"***Oi***"	Select/deselect computer number $i (i \in [1,5])$
"***Ai***"	Select/deselect cabinet number $i (i \in [1,2])$
"***Front wall***"	Apply *Front* between selected object and front wall
"***Right wall***"	Apply *Right* between selected object and right wall
"***Left wall***"	Apply *Left* between selected object and left wall
"***Back***"	Apply *Back* between two selected objects
"***Top***"	Apply *Object_on_object* between two selected objects
"***Fixed Distance***"	Apply *Distance_fixed* between selected objects
"***Min***"	Apply *Distance_min* between selected objects
"***OK***"	Validate selected constraints
"***Layout***"	Launch resolution

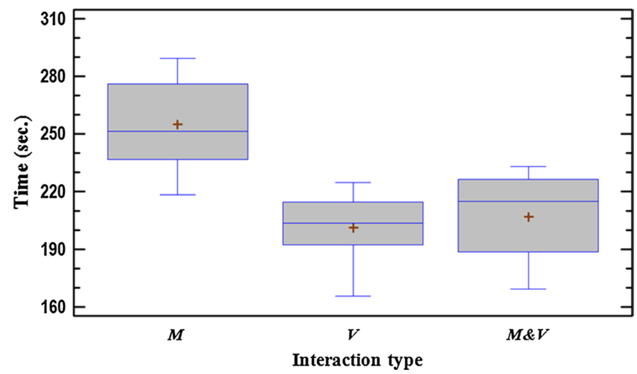

Fig. 6 Box plot of the task completion time for each type of interaction (mouse M, voice V, mouse and voice M&V)

5.6 Results and discussion

5.6.1 Objective analysis

We performed a repeated-measures ANOVA *within subjects* to investigate the effect of interaction techniques on the performance. The only independent variable was the type of interaction (***I***), while the task completion time was the only dependent variable. The ***I*** factor is defined by three conditions: interaction with mouse (***M***), interaction with voice (***V***), and multimodal interaction (***M&V***) with both mouse and voice commands. In addition, pair-wise *t* tests (see Table 2) were carried out to determine whether there is a significant difference between each pair of obtained data (***M*** with ***V***, ***M*** with ***M&V***, and ***V*** with ***M&V***).

1. *Task completion time*
   As shown in Fig. 6, there was a significant effect of ***I*** on the task completion time ($F_{2,11} = 21.88, p < 0.05$). It can be concluded that when the voice interface was used, performance was increased approximately 21.1% for the exclusive use (***V***), and about 18.8% for partial use (***M&V***), as compared to the mouse only condition (***M***) (see Table 2). *H1 was supported.*
   The difference of performance between the interaction ***M*** and the two others can be explained by the relatively long and tedious selection cycle of 3D objects. During

Table 2 Means and standard deviations of the task completion time for each type of interaction

	Interaction (*I*)	*N*	Means	SD
Time (s)	***M***	12	254.96	24.17
	V	12	201.30	18.30
	M&V	12	207.03	22.57

Mouse M, voice V, mouse and voice M&V

the experiment, the participants were required to: (1) identify the object to select in the 3D scene, (2) change the point of view of the camera for selection, if required, and (3) manually move the mouse cursor to the desired object. Changing viewpoints to select occluded objects can add more time to the task, as compared to the voice option where the user can select occluded objects with voice commands. Moreover, selecting the constraints menu took more time since the participants needed to visually identify the names of the correct constraints. The use of the voice interface (***V*** and ***M&V***) significantly reduced object selection time because the participants only needed to pronounce the objects' names without changing the point of view of the camera and without hand movements (Table 3).

Although the exclusive use of the voice command interface seemed the easiest and most intuitive way to layout the VE, a further ANOVA analysis between the ***V*** and ***M&V*** techniques showed no significant difference in performance ($F_{1,11} = 0658, p = 0.49$). The selection of constraints, which differentiated the techniques, took almost the same time. The results indicated that there was little difference between remembering the name of a constraint to be applied (***V***) and visually identifying constraints in the menu (***M&V***).

2. *Learning effect*
The concept of learning effect allows the identification of performance issues during repetitive testing of a given task. A lack of learning or a degradation in performance can also indicate the difficulties in the usage of the techniques.

We analyzed the evolution of the task completion time during the repetitive sessions of the layout task, to further investigate the learning effect. An important performance gain of 73, 63, and 67% was observed, respectively, for ***M***, ***V***, and ***M&V*** (see Fig. 7).

5.6.2 Subjective aspects

In this subsection, we evaluated the perceived level of workload based on the NASA-TLX questionnaire (Hart and Staveland 1988) and analyzed participants' feedbacks observed during the experiment, including task difficulty and user preference. As for the objective data, a one-way analysis of variance (ANOVA) was performed. Table 4 summarizes

Table 3 The p value of the t test for the three experimental conditions [mouse (M), voice (V), mouse&voice (M&V)]

Levels of the independent variable (I)	M–V	M–$M\&V$	V–$M\&V$
t test	$p = 0.00065$	$p = 0.00018$	$p = 0.36360$

the main effect of the independent variable on the workload indices. Except for OP, ***I*** significantly affected MD, PD, TD, EF, and FR indices. We are particularly interested in the effects of the multimodal interactions; therefore, we performed a post hoc t test between the two conditions ***M*** and ***V*** and found a significant difference in MD, PD, TD, EF, and FR indices (all tests $p < 0.001$).

1. *Workload*
Figure 8 shows the average score for each index involved in the workload evaluation. The results showed that the participants associated the use of mouse for VE interaction with higher mental, physical, and temporal demands ($\text{Mean}_{\{\text{MD}\}} = 4.41, \text{Mean}_{\{\text{PD}\}} = 5.08$ and $\text{Mean}_{\{\text{TD}\}} = 4.75$). This condition also led to more frustration ($\text{Mean}_{\{\text{FR}\}} = 5.66$) compared to the other techniques. Voice interaction also caused a high level of mental demand ($\text{Mean}_{\{\text{MD}\}} = 4$) as compared to the multimodal technique ($\text{Mean}_{\{\text{MD}\}} = 3.25$). This may be because the participants were asked to remember the objects and constraints names. For a large number of objects and constraints, we speculate that this technique can quickly reach a cognitive limit. For the combined technique (***M&V***), a higher physical demand (($\text{Mean}_{\{\text{PD}\}} = 3.16$) was caused compared with the voice-based interaction ($\text{Mean}_{\{\text{PD}\}} = 2.5$). Similar results were obtained for the TD, EF, and FR. This was possibly due to the use, even partial, of the mouse.

For the own performance (OP) index, a higher score is observed in the voice-based tasks and also in the mouse-based one (Fig. 8). This is due to the fact that the OP index was presented as a personal estimation of successful task accomplishment. Participants were aware of the existence of a solver assistance.

2. *Evaluation of interaction techniques*
As shown in Fig. 9, more than 66% of the participants found that the mouse interaction (***M***) was complex or

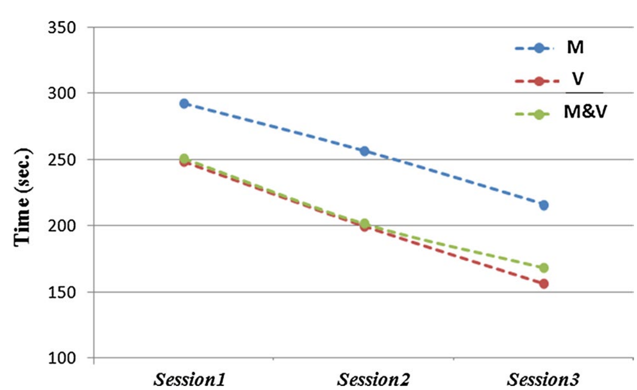

Fig. 7 The learning effect on the task completion time for each interaction technique. M (mouse), V (voice interface), and M&V

Table 4 The main effect of the interaction type (I) on the workload indices (mental demand (MD), physical demand (PD), temporal demands (TD), own performance (OP), effort (EF), frustration (FR)

Effect	MD	PD	TD	OP	EF	FR
I	$p = 0.0242$	$p < 0.0001$	$p = 0.0266$	$p = 0.3564$	$p < 0.0001$	$p < 0.0001$

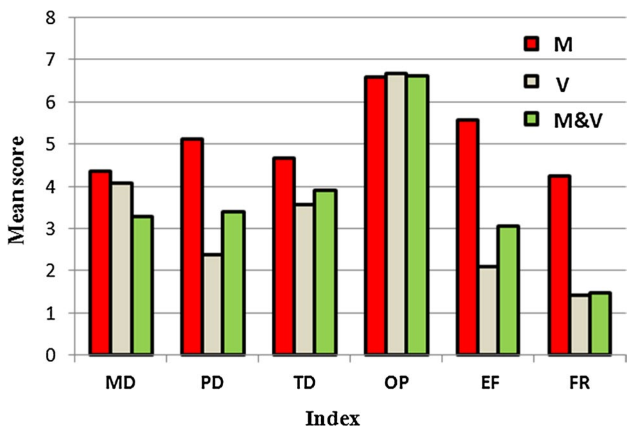

Fig. 8 Scores of workload index for each type of interaction (mouse M, voice V, mouse and voice M&V

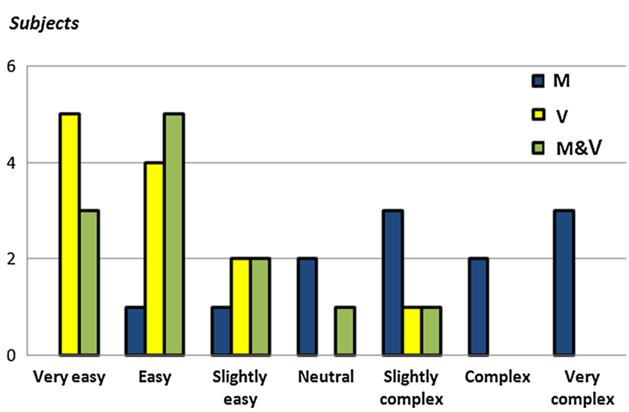

Fig. 9 Subjective evaluation of each type of interaction (mouse M, voice V, mouse and voice M&V

very complex to use. In contrast, nearly 92% of the participants found that the use of the voice-based interaction was easy or very easy. Similar opinions were expressed concerning the multimodal technique.

3. *Users preferences*
The descriptive statistics analysis of the user preference confirmed the results described in the previous paragraph. The participants' preferences were shared between the interactions ***V*** and ***M&V*** in which the voice interface was both used. Only one participant expressed a preference for the mouse interaction. Most of participants justified their choice by pointing out that the use of the voice interface for a 3D layout task was easier and more intuitive than the mouse interaction. They reported that, unlike the interaction using the mouse, they did not need getting accustomed to the system interface to interact with the VE.

The study is focused on evaluating different modalities of interaction with a solver-assisted 3D layout system, or any other similar virtual reality system with an intelligent or digital assistant or agent. The results of the study confirm that the use of the speech modality has significantly helped the participants to accomplish the layout task, as compared to a pointer-based handheld input device, represented by the mouse condition. Even the partial use of this technique (associated with the mouse) allowed the improvement of the performance and has proven itself in terms of performance and participants' satisfaction. However, we suspected that the exclusive use of this technique in a more complex 3D scene involving a large number of objects and constraints is not necessarily the most efficient and suitable one. The user will be required to memorize all the words identifying constraints, which leads us to believe that a multimodality technique would be more appropriate.

Our findings provide design guidelines for multimodal applications with focus on designing, building, and modifying the virtual environment, i.e., VR world editor. The consideration of the speech modality has a positive impact on task completion.

6 Conclusion

In this paper, we propose an interactive approach to solve 3D layout problems by using a constraint programming techniques. We specifically aim toward the development of a decision-making system that proposes feasible 3D configurations to the user, given a set of selected objects and constraints to be applied. We developed several interaction techniques (monomodal and multimodal) so that users can experiment naturally with the 3D prototype and the proposed system. We presented a usability study comparing the proposed interaction interfaces across a 3D layout tasks. Both objective and subjective data were collected and analyzed. The layout task consisted of an interactive arrangement of a room with furniture (armoires, desk, and computers).

The results confirm that the use of the voice interface has significantly helped the subjects to accomplish the layout task. One of the limitations of our study is the limited number of constraints and objects. For more complex 3D scene, there may be a consideration for the trade-off between remembering the names of the constraints for voice commands versus visually identifying the constraints for mouse selection, with regard to the usage of voice commands. The contribution of the paper is an empirical evaluation of multimodal interaction technique, leading to the conclusion that *multimodality does not improve the interaction if an appropriate single modality is found*. The choice of the most appropriate interaction strongly depends on the size of the problem of 3D arrangement. Due to the limitation in the number of objects in our study, further evaluation with multimodal interaction for a larger number of objects will be required to be conducted in the future.

Our approach contributes to the multidisciplinary work by combining constraint programming and virtual reality interaction to solve a layout task. Our work contributes to potential future work in many related disciplines that will benefit from immersive VR interactions, such as virtual assistance using artificial intelligence, industrial, and architectural design.

References

Azenkot S, Lee NB (2013) Exploring the use of speech input by blind people on mobile devices. In: Proceedings of the 15th international ACM SIGACCESS conference on computers and accessibility. ACM, pp 1–8

Billinghurst M (1998) Put that where? Voice and gesture at the graphics interface. SIGGRAPH Comput. Graph. 32(4):60–63

Bolt RA, Herranz E (1992) Two-handed gesture in multi-modal natural dialog. In: 5th Annual ACM symposium on user interface software and technology, pp 7–14

Cal C, Baghabra J, Boges DJ, Holst GR, Kreshuk A, Hamprecht FA, Magistretti PJ (2016) Three-dimensional immersive virtual reality for studying cellular compartments in 3D models from EM preparations of neural tissues. J Comp Neurol 524(1):23–38

Calderon C, Cavazza M, Diaz D (2003) A new approach to the interactive resolution of configuration problems in virtual environments. Lect Notes Comput Sci 2733:112–122

Chun LM, Arshad H, Piumsomboon T, Billinghurst M (2015) A combination of static and stroke gesture with speech for multimodal interaction in a virtual environment. In: 2015 International conference on electrical engineering and informatics (ICEEI). IEEE, pp 59–64

Fages F, Soliman S, Coolenand R (2004) CLPGUI: a generic graphical user interface for constraint logic programming. Constraints 9:241–262

Fernando T, Murray N, Tan K, Wimalaratne P (1999) Software architecture for a constraint-based virtual environment. In: Proceedings of the ACM symposium on virtual reality software and technology, pp 147–154

Gao S, Wan H, Peng Q (2000) An approach to solid modeling in a semi-immersive virtual environment. Comput Graph 24:191–202

Garcia VM, Ruíz MP, Pérez JR (2010) Voice interactive classroom, a service-oriented software architecture for speech-enabled learning. J Netw Comput Appl 33:603–610

Goel V, Slusky M, van Hoeve WJ, Furman KC, Shao Y (2015) Constraint programming for LNG ship scheduling and inventory management. Eur J Oper Res 241(3):662–673

Hart S, Staveland L (1988) Development of NASA-TLX (Task Load Index): results of empirical and theoretical research, pp 139–183

Honda K, Mizoguchi F (1995) Constraint-based approach for automatic spatial layout planning. Conference on artificial intelligence for applications. IEEE Press, Ne York

ISO/IEC (2011) 25010:2011 Systems and software engineering—systems and software quality requirements and evaluation (SQuaRE)—system and software quality models, ISO

Jacob R, Sibert L, MacFarlane D, Mullen P Jr (1994) Integrality and separability of input devices. Comput Hum Interact 1:3–26

Jacquenot G (2009) Méthode générique pour l'optimisation d'agencement géometrique et fonctionnel. Thése de Doctorat, Ecole Centrale de Nantes

Jaimes A, Sebe N (2007) Multimodal human–computer interaction: a survey. Comput Vis Image Underst 108(1):116–134

Kefi M, Barichard V, Richard P (2012) A constraint-solver based tool for user-assisted interactive 3D layout. In: ICTAI, pp 199–206

Ku P-Y et al (2013) Ergonomics design on expert convenience of voice-based interface for vehicles AV systems. In: Kurosu M (ed) Human–computer interaction. Applications and services: 15th international conference, HCI international 2013, Las Vegas, NV, USA, July 21–26, 2013, Proceedings, Part II. Springer, Berlin, pp 606–611

Kulyukin VA (2004) Human–robot interaction through gesture-free spoken dialogue

Laviola JJ Jr (1999) Whole-hand and speech input in virtual environments (Doctoral dissertation, Brown University)

Medjdoub B (2004) Constraint-based adaptation for complex space configuration in building services. J Inf Technol Constr 243:627–636

Patrick E et al (2000) Using a large projection screen as an alternative to head-mounted displays for virtual environments. In: Proceedings of the SIGCHI conference on human factors in computing systems. ACM

Pfefferkorn C (1975) A heuristic problem solving design system for equipment or furniture layouts. Commun ACM 18(5):286–297

Pires JN (2006) Industrial robots programming: building applications for the factories of the future. Springer, Berlin

Régin JC (2004) Modlisation et Contraintes Globales en Programmation par Contraintes. Habilitation diriger des recherches, Universit de Nice

Rogalski T, Wielgat R (2010) A concept of voice guided general aviation aircraft. Aerosp Sci Technol 14:321–328

Rogowski A (2012a) A industrially oriented voice control system. Robot Comput Integr Manuf 28:303–315

Rogowski A (2012b) Web-based remote voice control of robotized cells. Robot Comput Integr Manuf 4:77–89

Rossi F, Van Beek P, Walsh T (eds) (2006) Handbook of constraint programming. Elsevier, Amsterdam

Sanchez S, Le Roux O, Inglese F, Luga H, Gaildart V (2003) Constraint-based 3D-object layout using a genetic algorithm. 3IA

Schulte C, Tack G, Lagerkvist M (2013) Modeling with Gecode

Seth A, Vance JM, Oliver JH (2011) Virtual reality for assembly methods prototyping: a review. Virtual Real 15(1):5–20

Tim T, Rafael B, Ruben MS, Klaas JK (2009) Rule-based layout solving and its application to procedural interior generation. In: Proceedings of the CASA workshop on 3D advanced media in gaming and simulation (3AMIGAS), pp 212–227

Turk M (2014) Multimodal interaction: a review. Pattern Recogn Lett 36:189–195

Unity 3D Game Engine (2013) http://Unity3d.com
Vacher M, Lecouteux B, Istrate D, Joubert T, Portet F et al (2013) Experimental evaluation of speech recognition technologies for voice-based home automation control in a smart home. In: 4th Workshop on speech and language processing for assistive technologies, Grenoble, France, pp 99–105
Weyrich M, Drews P (1999) An interactive environment for virtual manufacturing: the virtual workbench. Comput Ind 38:5–15
Xu K, Stewart J, Fiume E (2002) Constraint-based automatic placement for scene composition. In: Graphics interface proceedings, University of Calgary, pp 25–34
Zhao W, Madhavan V (2005) Integration of voice commands into a virtual reality environment for assembly design. In: Proceedings of the 10th annual international conference on industrial engineering theory, applications & practice, Clearwater Beach, FL, USA