

# Mid-Air Gestures for Virtual Modeling with Leap Motion

Jian Cui<sup>1,2(✉)</sup>, Dieter W. Fellner<sup>1,3</sup>, Arjan Kuijper<sup>1,3</sup>, and Alexei Sourin<sup>2</sup>

<sup>1</sup> Technische Universität Darmstadt, Darmstadt, Germany  
cui.jian@gris.informatik.tu-darmstadt.de

<sup>2</sup> School of Computer Engineering, Nanyang Technological University,  
Singapore, Singapore  
CUIJ0006@e.ntu.edu.sg

<sup>3</sup> Fraunhofer IGD, Darmstadt, Germany  
arjan.kuijper@igd.fraunhofer.de

**Abstract.** We study to which extent Leap Motion can be used for mid-air interaction while working on various virtual assembling and shape modeling tasks. First, we outline the conceptual design phase, which is done by studying and classification of how human hands are used for various creative tasks in real life. Then, during the phase of the functional design, we propose our hypothesis how to efficiently implement and use natural gestures with Leap Motion and introduce the ideas of the algorithms. Next we describe the implementation phase of the gestures in virtual environment. It is followed by the user study proving our concept.

## 1 Introduction and Motivation

Mid-air interaction is an emerging spatial input mode which has been used in many areas of interaction, e.g., mid-air keyboard typing [13], interaction with large displays [15], virtual and augmented reality [17], and touchless interaction [2, 9, 10]. Recent progress in hand tracking using affordable controllers such as Leap Motion ([www.leapmotion.com](http://www.leapmotion.com)), Nimble VR ([niblevr.com](http://niblevr.com)), and MS Kinect ([www.microsoft.com](http://www.microsoft.com)) boosted research and development on precise hand tracking, especially in the area of computer games [14]. Breakthroughs were also made in predicting self-occluded hand, e.g., [20], which, until recently, was a serious obstacle for using optical tracking devices. However, application of mid-air gestures for precise 3D object manipulation in virtual environments, such as virtual prototyping, assembling and various shape modeling operations, still remains a challenging research problem. Indeed, with 27 degrees of freedom for the hand, only one grasping gesture can be classified into 33 variants [6]. With the motor skills acquired with age and experience, we take and manipulate objects of different size, shape and weight in a way that we feel is most natural and productive. For various simulations and training, professional motor skills in virtual environments, such natural gestures, should be recognized and implemented by the interactive modeling system.

In this paper, we perform a feasibility study on using Leap Motion controller for virtual assembling and shape modeling operations mimicking real life gestures rather than using artificial, however possibly more efficient for capturing, gestures [12,29]. We first analyze the existing progress with hand tracking in these areas, as well as what has been achieved in hand tracking with Leap Motion controller. Next, we analyze and classify the hand gestures which are used in real life hand-based desktop constructions, assembling, and modeling operations. We then come up with just a few algorithms that allow for recognition of many possible hand gestures. Next, we briefly outline the implementation and describe the user tests which we conducted to verify our hypothesis and the devised algorithms. The paper ends with a conclusion.

## 2 Use of Hand Gestures for Virtual Assembling and Modeling

Hand tracking can be performed using different platforms, such as virtual gloves with bending sensors [4], mechanical tracking devices (e.g., exo-skeletons), and optical and depth tracking devices, like Leap Motion, Nimble VR, and Kinect.

When performing various assembling operations, the user should be able to take the object, relocate it while also changing its orientation, and release it. Shape modeling often requires hand-made deformations, such as elastic deformations and various cuttings. There are two different approaches to perform the respective hand interaction in virtual environments (VE): collision based and gesture-based. The first one assumes that a virtual hand, controlled by the users hand, collides with the objects in the VE thus implementing the real life collision between the hand (fingers and palm) and the objects. This process, when implemented physically-based, may realistically simulate the real hand operations with a high degree of precision. Based on this approach, many research works have been done to implement how the virtual objects can be grasped, moved and deformed. For example, Garbaya et al. [8] proposed a spring-damper model for hand interaction with mechanical components in their virtual assembly system. A similar model was also proposed in [1] for the whole hand virtual grasping where linear and torque forces were calculated to be exerted on objects to simulate their dynamics. Besides physically-based methods, heuristic approaches can also be used to manipulate objects with virtual hands. For example, in [28] a method for manipulating a virtual wrench is proposed, while in [11] a realistic kinematics model of a virtual hand (skeleton [26,27], muscle and skin) is discussed. In some works, the virtual hand is used for creating surfaces [18] and point clouds [7], as well as for various deformations of elastic objects [25] according to the amount of force exerted by it.

An alternative to virtual hand simulation approach assumes that the objects in VE are manipulated by various mid-air gestures. Some gestures mimic the way we interact with objects in the real life, others are rather abstract. Thus, one typical metaphor is to use pinching gesture to select objects, and then to relocate them by moving and rotating hands [19]. A pointing gesture [16] is

often used to specify a direction for the object selection. In [24], a sheet-of-paper metaphor is proposed for visual assembling to rotate the view point direction like pinching a sheet of paper. Another example is [21] where a handle bar metaphor is proposed to mimic manipulation of objects that are skewered with a bimanual handle bar. For shape modeling, an interaction system named shape-it-up was proposed in [23] with three basic gestures for different manipulation and deformation operations.

Using devices like Leap Motion controller opens new prospects for the precision with which fingers can be tracked. Not only bending but even slight displacements of the fingers sideways can be precisely captured by it. However, not many works have been done with Leap Motion in the studied area. The Playground app<sup>1</sup> uses a ghost hands for assembling based on a rigid collision model provided by Unity Game Engine. Examples of applications with gestural interfaces for 3D object manipulation and camera adjustment can be found in on various websites<sup>2</sup>.

Generally, though the methods based on collision detection are quite promising, their efficiency is limited by the used hand tracking systems, which may not be able to capture properly all the 27 degrees of freedom of the hand. Also, the inability to deliver a realistic tactile feedback from the virtual hand causes the users to replace it with various visual feedbacks (e.g., colored finger tips to reflect the amount of forces exerted, etc.). As for the mid-air gestures, though efficiently captured by the tracking system, they are neither intended for benefiting from real-life hand motor skills nor for training these skills. In this paper, we perform a feasibility study of using Leap Motion controller for tracking mid-air gestures with application to real life gestures used in various desktop hand-made operations.

### 3 Proposed Mid-Air Gestures

First, we outline the conceptual design phase, which is done by studying and classification of how human hands are used for various creative hand-made tasks in real life. Then, working on the functional design, we propose our hypothesis how to efficiently implement and use natural gestures with Leap Motion and describe the ideas of the main algorithms. Finally, we briefly outline the implementation of the gestures in virtual environment.

#### 3.1 Hand Gestures Study

We studied and classified the plethora of real life hand gestures used when different desktop-based assembling and modeling operations are performed, which include unimanual and bimanual grasping, motion (including relocation and rotations), and deformation gestures. For the various hand-made operations on

<sup>1</sup> <http://blog.leapmotion.com/inside-leap-motion-designing-playground/>.

<sup>2</sup> See e.g. <https://apps.leapmotion.com/apps/sculpting/windows> and <https://apps.leapmotion.com/apps/cyber-science-motion-zoology/windows>.

real objects, the object is usually first taken (possessed) by hand. This gesture of taking the object may have different names: grasping, pinching, grabbing, gripping, etc. There are a few classifications of all these gestures controlling the objects position and orientation (e.g., [6]) however for the gestures used for desktop hand-made operations we will consider only the following three groups:

1. When the fingers move towards the opposable thumb (e.g., as in grabbing and pinching).
2. When the fingers bend towards the palm (e.g., as in cylindrical grip).
3. When one or a few fingers of the whole palm are touching the object while exerting some force thus establishing control over the object (e.g., when touching and picking).

Releasing of the object is a converse process so that the hand and/or fingers lose the contact with the object. All possible motions (relocations) of objects, which can be done with one hand after the object is taken, are then performed by moving the hand from one position to another as a sequence of taking-moving-releasing gestures. All the unimanual object rotations, after the object is taken, can be eventually classified into three groups:

1. Incremental rotation by taking-rotation-releasing sequences of gestures when the object is firmly held by the fingers and the thumb while the wrist rotates.
2. Incremental rotation by only moving the fingers and the thumb with a fixed position and orientation of the wrist. This gesture may be performed together with the previous wrist rotation as well.
3. Rotation by one or a few fingers or the whole palm performed as a circular motion while the fingers/hand touch the surface of the object which is being rotated.

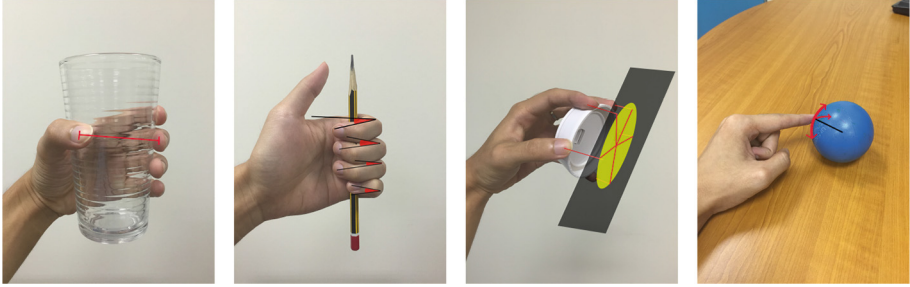
Finally, unimanual deformations, which can become useful for potential virtual modeling, are rather limited to only two groups:

1. Squeezing and twisting with the fingers moving towards the thumb.
2. Deformations by pressing the object with the thumb or one or a few fingers, as it is done in clay modeling.

Bimanual gestures add additional varieties to the considered groups of gestures. Thus, grasping can be done with two hands performing the grabbing, pinching or gripping gestures. While moving two hands holding the virtual object, the operations of relocation, rotation (like steering wheel rotation or a handle bar) and various deformations can also be performed.

### 3.2 Functional Design of the Mid-Air Gestures

Our research hypothesis is that we may achieve higher efficiency of natural mid-air gestures if we avoid displaying virtual hands since observation of the motion of the virtual objects/instruments controlled by the hand is more essential than an ability to see the simulated hand itself. As an advantage of this approach, we



**Fig. 1.** Mid-air gestures design. Form left to right: minimum distance from the thumb to fingertips; finger bending; circular rotation with fingers; rotation with one finger.

will not base most of our hand tracking algorithms on collision detection between the virtual hand and the objects but rather on recognition of the gesture itself. Therefore, the gesture algorithms will be not constrained by the number of polygons involved in construction of the virtual objects as soon as they can be rendered in real-time. This approach assumes that the virtual object or instrument is somehow selected or predefined, and once made visible, the user will manipulate it with the gestures which are commonly used with this object in real life. We also hypothesize that just a few basic gestures recognition algorithms can be devised to be able to still apply many varieties of the real life gestures.

We devised algorithms of unimanual grasping and pinching gestures based on computing the minimum distance from fingers to the thumb to trigger the event of grasping and pinching (First image of Fig. 1). The gripping gestures is based on computing the bending angle of the four fingers which, when exceeds a certain threshold value, will trigger the grasping event (Second image of Fig. 1). Releasing is a converse process to detect that the threshold value of the finger distance or bending angle is no longer exceeded. However, picking and touching have to be based on tracking the finger or palm positions and, depending on the content, may require performing a collision detection of the hand/fingers with the object or its bounding box.

Unimanual relocation algorithms (translation) are based on tracking the hand position. Unimanual rotation algorithms for taking-rotation-releasing sequences is based on tracking the wrist orientation. While for incremental rotation performed by moving fingers, we proposed an algorithm where the 3D positions of the finger and the thumb tips are projected onto a plane thus reducing the task of calculation of the angle of rotation to a 2D case (Third image of Fig. 1). Rotation with one finger is based on tracking the finger position as a particular case of the rotation by fingers (Last image of Fig. 1).

Unimanual deformation is a continuation of the respective grasping or gripping gesture so that the elastic objects can be deformed following the fingers motions after the grasping event is detected. This gesture can be performed as an incremental sequence grasping-squeezing-releasing. Deformation done by applying to the object a few fingers or the whole palm is based on tracking the

finger/thumb/palm positions and it may require, in contrast to other gestures, to compute the contact point with the virtual object surface.

Bimanual gestures are based on the unimanual gestures algorithms and assume that the virtual object is first grasped by one hand. Then, grasping the same object with the other hand triggers the event of bimanual gestures which are applied in a content sensitive way depending of the virtual object constraints i.e. whether and how the object can be moved, rotated or deformed.

### 3.3 Implementation

To implement the devised algorithms, we used Virtual Reality Modeling Language (VRML) and its successor Extensible 3D (X3D, cf. [5,22]) for defining interactive virtual worlds to be published in the networked environments as well as on local client computers. We used Bitmanagement BS Contact VRML/X3D viewer, which is a commonly used plugin to several Internet browsers. BS Contact provides an interface to support input devices by extending VRML/X3D through a DeviceSensor Node for receiving the input data from the device [3].

## 4 User Tests and Analysis of the Results

We have performed user studies on the efficiency of the proposed gestures in the virtual environment. We investigated:

1. whether each gesture can be seamlessly recognized when performed by different users,
2. whether the gestures will actually mirror natural gestures, and
3. whether the gestures can be used as fast and as precise as in real life.

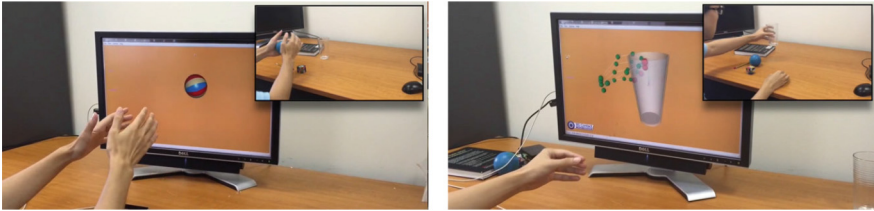
### 4.1 System Setup

We used a DELL PRECISION T7500 workstation with Intel(R) Xeon (R) CPU 2.40 GHz 2.39 GHz and 12 GB memory, running Windows 7 professional. The Leap Motion controller (software version is 2.3.1 +31549) was placed on the desktop facing up. The interaction height is approximately 20 cm above the desktop with a tracking rate of around 300 frames per second. The codes are written in VRML + java script, and visualized using MS Internet Explorer with BS Contact plugin. The hand data is transferred to the viewing platform by our plugin [3].

### 4.2 Testing Procedure

Two experiments were conducted with 10 participants (7 males and 3 females) with an age between 21 to 32 ( $M = 25$ ,  $SD = 3.8$ ) for the experiment. 4 of them had experience with using Leap Motion gestural interface. During the first experiment, the participants were required to sit in front of the computer monitor and

interact with their hands above the desktop while looking at the monitor (Fig. 2). Different objects were shown on the computer monitor for 5 s each, specifically, Rubiks cube, ball, glass, book, and pencil, while virtual hands were not displayed at all. Within 5 s, the participants had to take each object with any hand or with both hands and in a way how they would do in real life for the same object (grabbing, pinching, taking with two hands). When the object is taken (the gesture is recognized), the objects slightly changed their visual appearance to give a visual feedback that the object is possessed by the participant. Then, the participant had to move the object to a new place, while also changing its orientation either by rotating the wrist or by moving fingers, and, finally, to release it. The object could be also squeezed to deform it. After completion, the users were invited to perform the same operations with real objects and also within 5 s for each of the object. The two tests were then compared in terms of timing and precision.



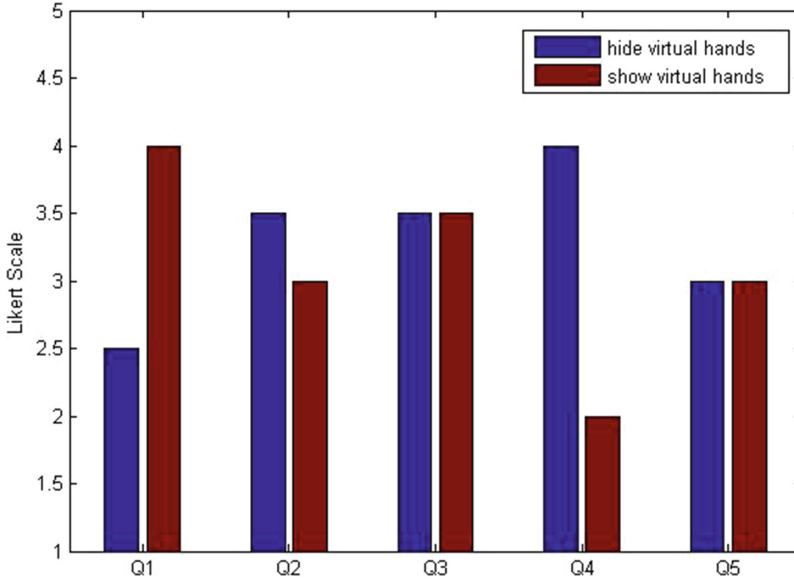
**Fig. 2.** Performing bare-hand operations with mid-air gestures. Left: without displaying virtual hands. Right: while displaying the virtual hands.

During the second experiment, the participants were required to perform the same 5 s per object tests while the virtual hands were displayed. After the experiments, the participants were asked to answer a questionnaire. The questionnaire was based on a 5 point Likert scale containing the following questions:

- Q1: I feel it is easy and not stressful to accomplish the task.
- Q2: I can remember and use the interaction techniques.
- Q3: I feel natural to interact with the objects.
- Q4: I feel no confusion during interaction.
- Q5: I like this interface.

### 4.3 Results and Discussion

The collected replies on the five questions are accumulated in Fig. 3. We found that interaction time without showing virtual hands ( $t = 3.34$ ,  $SD = 0.65$ ) is shorter than that when the virtual hands were shown ( $t = 4.13$ ,  $SD = 0.68$ ). Showing virtual hands can be less stressful (Q1), however it may lead to confusion (Q4) and excessive but unnecessary concentration on the virtual hands (Q2). The user preferences are equally split between the two approaches (Q5). Using motor



**Fig. 3.** Participants replies after the two experiments with mid-air gestures with and without displaying the virtual hands.

skills was also equally split (Q3). Experienced participants preferred hiding the virtual hands.

Moreover, when the virtual hands were not displayed, the participants tended to select more comfortable position for the hands as well as the best capturing position for the Leap Motion controller. This led to improving the quality of gesture recognition. We have also concluded that to achieve a better performance with the gesture recognitions without showing the hands, it may become useful to tune the interactive modeling system for some users, in a way how it is done for common mouse and touch-pad interactions.

When virtual hands are displayed, the users tended to move them towards the object, which reduced the efficiency of gesture recognition when the hands left the best capturing area of the Leap Motion controller.

## 5 Conclusion

We proved that Leap Motion can be efficiently used for mid-air interaction while working on various virtual assembling and shape modeling desktop tasks. We have proved with the user studies our research hypothesis that higher efficiency of natural mid-air gestures can be achieved without displaying the virtual hand hence shifting attention of the user to the hand-controlled virtual instruments and employing the existing hand motor skills of the users. The proposed approach allows the users to begin bare-hand virtual manipulations without learning any



special instructions by simply doing things in the same natural and logical way as in real life.

## References

1. Borst, C.W., Indugula, A.P.: A spring model for whole-hand virtual grasping. *Presence: Teleoper. Virtual Environ.* **15**(1), 47–61 (2006)
2. Braun, A., Wichert, R., Kuijper, A., Fellner, D.W.: Capacitive proximity sensing in smart environments. *JAISE* **7**(4), 483–510 (2015)
3. Cui, J., Sourin, A.: Feasibility study on free hand geometric modelling using leap motion in VRML/X3D. In: *Proceedings of the 2014 International Conference on Cyberworlds, CW 2014*, pp. 389–392 (2014)
4. Dipietro, L., Sabatini, A., Dario, P.: A survey of glove-based systems and their applications. *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.* **38**(4), 461–482 (2008)
5. Eicke, T.N., Jung, Y., Kuijper, A.: Stable dynamic webshadows in the X3DOM framework. *Expert Syst. Appl.* **42**(7), 3585–3609 (2015)
6. Feix, T., Pawlik, R., Schmiedmayer, H., Romero, J., Kragic, D.: A comprehensive grasp taxonomy. In: *Robotics, Science and Systems: Workshop on Understanding the Human Hand for Advancing Robotic Manipulation* (2009)
7. Fuge, M., Yumer, M.E., Orbay, G., Kara, L.B.: Conceptual design and modification of freeform surfaces using dual shape representations in augmented reality environments. *Comput. Aided Des.* **44**(10), 1020–1032 (2012)
8. Garbaya, S., Zaldivar-Colado, U.: The affect of contact force sensations on user performance in virtual assembly tasks. *Virtual Reality* **11**(4), 287–299 (2007)
9. Grosse-Puppenthal, T., Herber, S., Wimmer, R., Englert, F., Beck, S., von Wilmsdorff, J., Wichert, R., Kuijper, A.: Capacitive near-field communication for ubiquitous interaction and perception. In: *The 2014 ACM Conference on Ubiquitous Computing, UbiComp 2014, Seattle, WA, USA, 13–17 September 2014*, pp. 231–242 (2014)
10. Große-Puppenthal, T.A., Braun, A., Kamieth, F., Kuijper, A.: Swiss-cheese extended: an object recognition method for ubiquitous interfaces based on capacitive proximity sensing. In: *2013 ACM SIGCHI Conference on Human Factors in Computing Systems, CHI 2013, Paris, France, 27 April–2 May 2013*, pp. 1401–1410 (2013)
11. Huagen, W., Shuming, G., Qunsheng, P.: Virtual grasping for virtual assembly tasks. In: *Third International Conference on Image and Graphics (ICIG 2004)*, pp. 448–451 (2004)
12. Lai, D., Sourin, A.: Interactive free-form shape modeling in cyberworlds. *Vis. Comput.* **29**(10), 1027–1037 (2013)
13. Markussen, A., Jakobsen, M.R., Hornbæk, K.: Vulture: a mid-air word-gesture keyboard. In: *Proceedings of the 32nd Annual ACM Conference on Human Factors in Computing Systems, CHI 2014*, pp. 1073–1082 (2014)
14. Moser, C., Tscheligi, M.: Physics-based gaming: exploring touch vs. mid-air gesture input. In: *Proceedings of the 14th International Conference on Interaction Design and Children, IDC 2015*, pp. 291–294 (2015)
15. Nancel, M., Wagner, J., Pietriga, E., Chapuis, O., Mackay, W.: Mid-air pan-and-zoom on wall-sized displays. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI 2011*, pp. 177–186 (2011)

16. Nickel, K., Stiefelhagen, R.: Pointing gesture recognition based on 3D-tracking of face, hands and head orientation. In: Proceedings of the 5th International Conference on Multimodal Interfaces, ICMI 2003, pp. 140–146 (2003)
17. Piumsomboon, T., Clark, A., Billinghamurst, M., Cockburn, A.: User-defined gestures for augmented reality. In: CHI 2013 Extended Abstracts on Human Factors in Computing Systems, CHI EA 2013, pp. 955–960 (2013)
18. Schkolne, S., Pruett, M., Schröder, P.: Surface drawing: creating organic 3D shapes with the hand and tangible tools. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI 2001, pp. 261–268 (2001)
19. Segen, J., Kumar, S.: Gesture VR: Vision-based 3D hand interace for spatial interaction. In: Proceedings of the Sixth ACM International Conference on Multimedia, MULTIMEDIA 1998, pp. 455–464 (1998)
20. Sharp, T., Keskin, C., Robertson, D., Taylor, J., Shotton, J., Kim, D., Rhemann, C., Leichter, I., Vinnikov, A., Wei, Y., Freedman, D., Kohli, P., Krupka, E., Fitzgibbon, A., Izadi, S.: Accurate, robust, and flexible real-time hand tracking. In: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, CHI 2015, pp. 3633–3642 (2015)
21. Song, P., Goh, W.B., Hutama, W., Fu, C.W., Liu, X.: A handle bar metaphor for virtual object manipulation with mid-air interaction. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI 2012, pp. 1297–1306 (2012)
22. Stein, C., Limper, M., Kuijper, A.: Spatial data structures to accelerate the visibility determination for large model visualization on the web. In: Web3D14, pp. 53–61 (2014)
23. Murugappan, S., Liu, H., Ramani, K.: Shape-it-up: hand gesture based creative expression of 3D shapes using intelligent generalized cylinders. *Comput. Aided Des.* **45**(2), 277–287 (2013)
24. Wang, R., Paris, S., Popović, J.: 6D hands: markerless hand-tracking for computer aided design. In: Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology, UIST 2011, pp. 549–558 (2011)
25. Yang, C.C., Lin, P.J., Sun, C.C.: Product form design using virtual hand and deformable models. *Int. J. Digit. Content Technol. Appl.* **6**(11), 8–17 (2012)
26. Yoon, S.M., Kuijper, A.: Human action recognition using segmented skeletal features. In: 20th International Conference on Pattern Recognition, ICPR 2010, Istanbul, Turkey, 23–26 August 2010, pp. 3740–3743 (2010)
27. Yoon, S.M., Kuijper, A.: Human action recognition based on skeleton splitting. *Expert Syst. Appl.* **40**(17), 6848–6855 (2013)
28. Zachmann, G., Rettig, A.: Natural and robust interaction in virtual assembly simulation. In: Eighth ISPE International Conference on Concurrent Engineering: Research and Applications ISPE/CE2001, July 2001
29. Zhang, X., Sourin, A.: Image-inspired haptic interaction. *J. Visual. Comput. Anim.* **26**(3–4), 311–319 (2015)