ChinaRxiv

AI translation · View original & related papers at
chinarxiv.org/items/chinaxiv-201805.00067

# Postprint: Construction of Auto-Grasping Virtual Hand Interaction Behavior for Virtual Maintenance Training

**Authors:** Dong Jiankang, Wang Qifeng, Geng Hong

**Date:** 2018-05-02T00:00:00+00:00

## Abstract

To address the issue that current Maintenance Training Devices (MTD) cannot effectively demonstrate specific maintenance actions during aircraft maintenance processes to trainees, this work constructs a self-grasping virtual hand and corresponding interaction behaviors to improve the quality of maintenance personnel training. By analyzing the fundamental action elements of maintenance, a skinned-skeleton based virtual hand motion library is defined and constructed. During the scene roaming stage, the viewpoint position is acquired in real time to drive virtual hand motion, thereby reducing the control load of the virtual hand during scene roaming. In the specific maintenance operation stage, the proposed interaction behavior algorithm combines the virtual hand' s position self-adjustment with skeletal animation osgCal, and completes specific maintenance tasks by retrieving corresponding actions from the motion library. Experimental results demonstrate that the proposed method can effectively demonstrate specific maintenance operations and exhibits good usability for virtual maintenance training.

## Full Text

## Preamble

### Self-fetching Virtual Hand Interaction Behavior Building for Virtual Maintenance Training

*Dong Jiankang, Wang Qifeng, Geng Hong*
(College of Electronic Information & Automation, Civil Aviation University of China, Tianjin 300300, China)

**Abstract:** Current maintenance training devices (MTDs) cannot effectively demonstrate specific maintenance actions during aircraft maintenance procedures to trainees. To address this limitation and improve training quality, this paper establishes a self-fetching virtual hand and corresponding interaction behaviors. By analyzing the fundamental elements of maintenance operations, we define and construct a skinned mesh-based virtual hand action library. During the scene roaming stage, the virtual hand movement is driven by real-time viewpoint acquisition to reduce control load. During specific maintenance operations, the proposed interaction behavior algorithm combines virtual hand position self-adjustment with skeleton animation osgCal, enabling completion of specific maintenance tasks by retrieving appropriate actions from the action library. Experimental results demonstrate that the proposed method effectively exhibits specific maintenance operations and offers good usability for virtual maintenance training.

**Keywords:** virtual maintenance; virtual hands; skeleton animation osgCal; human-machine interaction

**Classification:** TP391
**DOI:** 10.3969/j.issn.1001-3695.2017.10.1021

## 0 Introduction

Maintenance training systems should enable trainees to not only familiarize themselves with maintenance procedures but also comprehend specific maintenance operation actions. Current virtual hand interaction modes in MTDs fall into two categories: external device-driven virtual hands and self-fetching virtual hands. External device-driven approaches, such as data glove mode [1] and Kinect mode [2], capture real-time hand posture information and map human movements to virtual hands in the maintenance environment. While effective for mastering actual operation movements, these methods incur high costs, require virtual reality headsets or stereoscopic glasses, cause fatigue, and impose heavy operation loads [3,4], making them unsuitable for two-dimensional MTDs that must support multi-user training.

Self-fetching virtual hands represent an alternative approach. One method proposed in literature [5] builds upon Kamper et al.' s research on fingertip trajectories during grasping [6], utilizing trajectory lines for self-fetching. However, this approach requires pre-collected real fingertip data from external devices to construct grasp trajectory lines, hindering extension and refinement of maintenance actions. Another approach in literature [7] employs 3D mouse-driven virtual hands with bounding box perception zones around objects, but this interaction method proves unsuitable for complex, fine-operation scenarios due to low interaction hit rates and cumbersome control processes.

Addressing these challenges, and considering the requirements of large-scene roaming, multi-scene switching, complex equipment, limited inter-device spacing, and precise operation needs for troubleshooting wiring in virtual aircraft (VA) environments, this paper adopts a skinned skeleton-based virtual hand modeling approach. The proposed method uses the mouse as the object selection tool, enabling demonstration of specific maintenance operation actions to trainees without increasing operation load or requiring peripheral devices.

## 1 Self-fetching Virtual Hand Interaction Flow

Figure 1 [Figure 1: see original paper] illustrates the self-fetching virtual hand interaction flow. During the scene roaming stage, users control navigation via mouse and keyboard to approach target virtual objects. The virtual hand continuously acquires the viewpoint matrix and adjusts its position accordingly, achieving adaptive position regulation during roaming. During the interaction stage, when a target object enters the appropriate viewpoint range, the user selects it with the mouse and issues a command. The object' s position, orientation, and grasp gesture attributes are then transmitted to the virtual hand, which performs position adjustment and retrieves skeleton animation data from the library to transition between different interaction states, completing grasp and release tasks. Users can adjust the virtual hand' s interaction state based on real-time visual feedback to proceed or return to previous steps. This workflow demonstrates specific maintenance operations without increasing operation load or reducing interaction hit rates.

## 2.1 Gesture Action Library Definition

The gesture action library forms the foundation for self-fetching virtual hands. Its definition depends on specific maintenance objects and tasks: too few gestures fail to authentically represent operations, while too many create redundancy and waste system resources. Based on Gilbreth' s therblig classification and using our independently developed Airbus A320 maintenance training device (HDMTD-2) as an example, the basic operation gestures involved in developed 3D field maintenance simulation training projects can be categorized into seven types, as shown in Figure 2 [Figure 2: see original paper].

**Basic Gestures:** - **Release:** Five fingers extended and spread, used for releasing objects and tools. - **Normal grasp:** Four fingers together and bent, used for gripping handles, wrenches, protective covers, and tools. - **Riveting grasp:** Thumb and index finger open in grasping posture, remaining three fingers bent and clenched, used for riveting different objects, wires, and tubes. - **Press:** Index finger extended, thumb bent near index finger, remaining three fingers bent and clenched, used for pressing circuit breakers and buttons. - **Pinch:** Index finger and thumb extended in pinching posture, used for holding small objects like screws and nuts. - **Grab:** Five fingers spread and bent, used for grabbing larger components like circular protective plugs. - **Twist:** Thumb slightly open, remaining fingers clenched with space between thumb and index finger, used for twisting switches and knobs.

Building upon these basic gestures, final stable grasp gestures $G_{ij}$ are defined for specific maintenance objects (where $i = 1, 2, ..., 6$ represents basic gesture numbers and $j = 1, 2, 3, ..., n$ represents final stable gesture numbers corresponding to basic gesture $i$). Each basic gesture shares the same hand shape with its corresponding series of final stable gestures, differing only in grasp depth. The basic gesture $G_i^{base}$ and final stable gesture $G_{ij}$ together constitute the gesture action library and animation keyframes.

## 2.2 Skeleton Animation Gesture Action Library Construction

Realistic virtual hands enhance training effectiveness [8]. Different maintenance tasks require different gestures, and even for the same task, the gesture library may need extension or updates. Considering geometric realism, implementation feasibility, and dynamic extensibility, we employ 3DMAX for virtual hand modeling and skeleton animation construction, following the workflow shown in Figure 3 [Figure 3: see original paper].

The construction process comprises two main phases: 3D skeleton animation creation and model data conversion/export. First, the virtual hand mesh model is created and smoothed in 3DMAX, then textured to enhance geometric realism. Next, bones are created and skinned to the mesh, with influence weights adjusted. Gesture skeleton animations are produced based on the defined gestures. Finally, Cal3DExp converts the skeleton animation data and exports bone files (.csf), mesh files (.cmf), texture files (.xrf), and animation files (.caf). After exporting these data files, a configuration text file (.cfg) records the skeleton animation information for OSG environment loading.

Figure 4 [Figure 4: see original paper] shows, from left to right, the original virtual hand mesh model in 3DMAX, the smoothed mesh model, and the final textured and skinned model. The constructed virtual hand model exhibits strong geometric realism.

## 3.1 Position Matching Process Implementation

The position matching process adjusts the virtual hand’ s position and orientation to align its attitude axes with the virtual object’ s attitude axes, placing the virtual hand in the correct grasping position. This stage also completes the $G_{base} \rightarrow G_i$ transition. Using the virtual hand grasping a screw as an example (Figure 5 [Figure 5: see original paper]), we discuss the position matching implementation.

Due to matrix constraint relationships between fingers and phalanges, a fixed gesture $G_i$’ s hand shape information can be represented by the orientation information of one finger’ s distal phalanx. Similarly, the entire virtual hand’ s position and orientation can be represented by one finger’ s distal phalanx position and orientation. In Figure 5, the thumb distal phalanx $P_B$’ s position/orientation and the entire virtual hand’ s position/orientation $P_D$ can be obtained from the index finger distal phalanx $P_C$’ s position/orientation through transformation matrices, satisfying:

$$\begin{cases} C_{MAT} = P'_B = M_{AT} \cdot P_B \\ D_{MAT} = P'_C = M_{AT} \cdot P_C \end{cases}$$

where $C_{MAT}$ and $D_{MAT}$ are $4 \times 4$ transformation matrices calculated using position and orientation information provided by 3DMAX during skeleton animation production. Since the index finger participates in all defined maintenance operations, we use the index finger distal phalanx as the attitude axis matching reference.

In the grasping process shown in Figure 5 [Figure 5: see original paper], the screw is an operable hot object with type 3 (visual feedback: small hand), pick ID 803, maintenance motion attribute: rotation, initial position $(393.891, 323.872, 339.293)$, basic grasp gesture attribute: pinch. The matching feature between virtual hand and screw requires the virtual hand' s index finger distal phalanx $B$ to be collinear with screw $A$ along the $z$-axis. After mouse selection of screw $A$ issues a grasp command, the virtual hand acquires screw $A$' s position and orientation. Based on $A$' s information and matching features, the position/orientation that index finger distal phalanx $B$ should achieve in state $G_i^P$ can be calculated. Using Equation (1), the virtual hand' s required position/orientation $M'_{AT}$ is computed. Based on this matrix information, OSG performs translation and rotation operations to transform the virtual hand from state $G_{base}$ to $G_i^P$, completing position matching.

## 3.2 Virtual Hand Interaction Behavior State Machine Model

Figure 6 [Figure 6: see original paper] shows the virtual hand interaction state machine model. The initial state is $G_{base}$. Based on interaction states, gesture states divide into three categories: $G_{base}$ represents the pre-grasping state (the release gesture defined in the gesture library), which serves as the initial frame for skeleton animation transformation in 3DMAX and follows the viewpoint matrix during keyboard roaming; $G_i^P$ represents the preliminary grasping state (where $i = 1, 2, ..., 6$ represents the six gestures excluding release), which acts as both the end frame for $G_{base} \to G_i^P$ transition and the start frame for $G_i^P \to G_i^D$ transition; $G_i^D$ represents the final stable grasp state corresponding to $G_i^P$ (where $i = 1, 2, ..., 6,\ j = 1, 2, 3, ...$), with the $j$ range extensible as needed, serving as the end frame for $G_i^P \to G_i^D$ transition in 3DMAX.

**Grasping Process:** Users roam the scene via mouse/keyboard to approach virtual objects. When the mouse contacts an operable object, it changes to a small hand to provide visual feedback. Right-clicking the object displays a context menu. Selecting the "pick" command while in $G_{base}$ state triggers acquisition of the object' s attitude matrix and basic grasp gesture attribute. After position matching and osgCal animation retrieval, the gesture transitions from $G_{base}$ to $G_i^P$, achieving preliminary grasp and marking the virtual hand state as $G_i^P$. Subsequent right-click and "grasp" command while in $G_i^P$ state retrieves osgCal animation to transition from $G_i^P$ to $G_i^D$, completing stable grasping through state machine conversion.

**Release Process:** When in $G_i^D$ state, pressing keyboard B or b triggers osgCal animation to transition from $G_i^D$ to $G_i^P$, achieving preliminary release. Continuing to press keyboard R or r while in $G_i^P$ state triggers osgCal animation to transition from $G_i^P$ to $G_{base}$ through inverse transformation, completing full object release and returning to roaming state.

## 4 Experiments

Experiments were conducted in the 3D virtual aircraft maintenance environment of our independently developed Airbus A320 maintenance training device (HDMTD-2). The software configuration included: 3DMAX 2013 X64, OSG (OpenSceneGraph-3.0) 3D engine, PROE 5.0, Cal3D 0.11.0, osgCal 0.3.1, and VS2012 on Windows 7 X64. Figure 7 [Figure 7: see original paper] shows the virtual hand performing partial tasks of replenishing brake reservoir (2624GM) oil quantity.

In Figure 7(a), the virtual hand follows the camera viewpoint while roaming in the first A320 field maintenance simulation scene. Figure 7(b) shows the virtual hand entering the second scene and approaching the FILLING EQUIPMENT-BRAKE RESERVOIR. When the mouse contacts the operable hot object (hand pump), it changes to a small hand for visual feedback (Figure 7(b)). Right-clicking the hand pump displays a context menu; selecting the "Pick" command (Figure 7(c)) triggers position matching and osgCal animation to transition from $G_{base}$ to $G_i^P$, completing preliminary grasp (Figure 7(d)). Subsequent right-click and "grasp" command transitions the state from $G_i^P$ to $G_i^D$, achieving stable grasp of the hand pump (Figure 7(e)). In Figure 7(f), the virtual hand executes the maintenance operation by pressing down the hand pump based on acquired action_stage information.

To evaluate effectiveness, we conducted comparative experiments with traditional MTD maintenance training systems. Participants were senior students undergoing CCAR-147 (Civil Aircraft Maintenance Training Organization Qualification Regulations) skills training, all with practical maintenance knowledge and prior MTD theoretical/practical courses. Forty students were divided into groups A and B (20 each), alternating between traditional VA and our improved VA to complete identical virtual maintenance training projects on identical computer configurations.

To enhance credibility, evaluation comprised objective metrics (Table 1 ) and subjective assessment (Table 2 ). For subjective evaluation, we provided system operation training without revealing experimental purposes, then administered experience questionnaires (metrics in Table 2, 10-point scale).

**Table 1** shows objective metrics: the self-fetching virtual hand VA required no additional interaction devices while maintaining identical mean frame rates and training completion times as traditional VA. **Table 2** shows subjective results: the improved VA system, without significantly increasing operation complexity, enabled trainees to more intuitively understand specific maintenance gestures, improved similarity to actual operations, and substantially enhanced overall training effectiveness compared to traditional VA.

## 5 Conclusion

Enabling trainees to comprehend specific maintenance actions is essential for improving practical skills. Current MTD maintenance training systems lack this capability. This paper developed a solution without adding peripheral devices or operation load. Experimental results demonstrate that the proposed method effectively demonstrates specific maintenance operations to trainees, offering better usability than traditional VA systems. However, since the virtual hand's palm and phalanges remain rigid models, tight contact between virtual hand skin and objects cannot yet be achieved, requiring further research for true stable grasping. Additionally, the limited number of skeleton animations in the current gesture library results in somewhat rigid demonstration; future work will extend the gesture library and refine maintenance actions to improve naturalness.

## References

[1] Liao J, Zhao G, Tan D, et al. Real-time virtual hand controlling using OpenSceneGraph [C]// Proc of IEEE International Conference on System Science, Engineering Design and Manufacturing Informatization. 2012: 29-32.

[2] Tang Z B, Zhang Z L, Liang F, et al. Gesture recognition in virtual maintenance training [J]. Computer Engineering and Design, 2014, 35(12): 4278-4283.

[3] Zhou Z, Zhou Y, Xiao J J. Survey on virtual reality enhancement technology [J]. Scientia Sinica (Informationis), 2015, 45(2): 157-180.

[4] Gao Y, Liu Y, Cheng D W, et al. Survey on head-mounted display development [J]. Journal of Computer-Aided Design & Computer Graphics, 2016, 28(6): 896-904.

[5] Chen W F. Research on dexterous virtual hand natural interaction [D]. Hangzhou: Zhejiang University, 2014.

[6] Kamper D G, Cruz E G, Siegel M P. Stereotypical fingertip trajectories during grasp [J]. Journal of Neurophysiology, 2003, 90(6): 3702-3710.

[7] Cai X J, Cheng C, Marwah A, et al. Virtual hand behavior construction in virtual manufacturing environment [J]. Journal of Computer-Aided Design & Computer Graphics, 2015, 27(3): 499-507.

[8] Aslandere T, Dreyer D, Pankratz F. Virtual hand-button interaction in a generic virtual reality flight simulator [C]// Proc of IEEE Aerospace Conference.

2015: 1-8.

[9] Xia P J, Lopes A M, Restivo M T, et al. A new type haptics-based virtual environment system for assembly training of complex products [J]. International Journal of Advanced Manufacturing Technology, 2012, 58(1-4): 379-396.

[10] Liu X, Cui X, Song G, et al. Development of a virtual maintenance system with virtual hand [J]. International Journal of Advanced Manufacturing Technology, 2014, 70(9-12): 2241-2247.

[11] Lu G, Shark L K, Hall G, et al. Dynamic hand gesture tracking and recognition for real-time immersive virtual object manipulation [C]// Proc of International Conference on Cyberworlds. [S.l.]: IEEE Computer Society, 2009: 29-35.

[12] Li J R, Xu Y H, Ni J L, et al. Glove-based virtual hand grasping for virtual mechanical assembly [J]. Assembly Automation, 2016, 36(4): 349-361.

[13] Gao W, Guo J. Implementation method of computer sign language action based on dual model [J]. Application Research of Computers, 2007, 24(11): 315-317, 320.

[14] Kumar V, Todorov E. MuJoCo HAPTIX: a virtual reality system for hand manipulation [C]// Proc of IEEE-RAS, International Conference on Humanoid Robots. 2015: 657-663.

[15] Kim J S, Park J M. Physics-based hand interaction with virtual objects [C]// Proc of IEEE International Conference on Robotics and Automation. 2015: 3312-3317.

[16] Hu C, Zhang X D, Ma H M. Virtual hand grasping adapted to object shape [J]. Journal of Image and Graphics, 2017, 22(5): 663-670.

[17] Yang X W, Zhang Z C, Kuang L Q, et al. Key technologies of human-computer interaction based on virtual hand [J]. Journal of Computer Applications, 2015, 35(10): 2945-2949.

[18] Li Y, Luo J X, Chen W D, et al. Virtual hand interaction collision simulation method based on skinned skeleton [J]. Journal of Zhejiang University (Engineering Science), 2014, 48(1): 105-112.

[19] Hu H, Chao J G, Lin W H, et al. Virtual assembly training simulation method for astronauts based on virtual hand interaction [J]. Journal of Computer Applications, 2015, 35(s2): 200-203.

[20] Gu Y, He Q C, Fan X M, et al. Assembly motion planning and intelligent optimization of virtual human arm in narrow space [J]. Computer Integrated Manufacturing Systems, 2016, 22(6): 1447-1455.

[21] Liu R Q, Huang H M, Liu J G. Design and implementation of an applicable role control system [J]. Application Research of Computers, 2009, 26(12): 4631-4634.

[22] Sagayam K M, Hemanth D J. Hand posture and gesture recognition techniques for virtual reality applications: a survey [J]. Virtual Reality, 2016, 21(2): 1-17.

[23] Chen N, Li W W, Nie L X, et al. Research on the simulation technology of virtual hand interventional manipulation in the virtual environment [J]. Applied Mechanics & Materials, 2014, 665: 698-705.

[24] Tang Y, Yang B, Gu H. Creating a realistic virtual hand for head-mounted displays [J]. Electronics World, 2013, 119(1925): 36-39.

[25] Chen W F, Zhu S J, Wan H G, et al. Dual quaternion based virtual hand interaction modeling [J]. Science China Information Sciences, 2013, 56(3): 1-14.

[26] Wang K R, Xiao Bingjia, Xia J Y, et al. A real-time vision-based hand gesture interaction system for virtual EAST [J]. Fusion Engineering & Design, 2016, 112: 829-834.

[27] Xiong W, Wang Q H, Li J R. Hierarchical interaction gesture technology for virtual assembly [J]. Journal of South China University of Technology (Natural Science Edition), 2016, 44(1): 78-84.

[28] Lin W, Du L, Harris-Adamson C, et al. Design of hand gestures for manipulating objects in virtual reality [C]// Proc of International Conference on Human-Computer Interaction. Cham: Springer, 2017: 584-592.

[29] Hirota K, Tagawa K. Interaction with virtual object using deformable hand [C]// Proc of IEEE Virtual Reality. 2016: 49-56.

[30] Huang J, Jing H. Research on gesture recognition based on Leap Motion in virtual interaction [J]. Application Research of Computers, 2017, 34(4): 1231-1234.

*Note: Figure translations are in progress. See original paper for figures.*

*Source: ChinaXiv —Machine translation. Verify with original.*