

Available online at www.sciencedirect.com**SciVerse ScienceDirect**

Procedia Technology 7 (2013) 265 – 272

Procedia
Technology

The 2013 Iberoamerican Conference on Electronics Engineering and Computer Science

Development of a haptic virtual reality system for assembly planning and evaluation

Germanico Gonzalez-Badillo^{a*}, Hugo I. Medellín-Castillo^a, Theodore Lim^b^aUniversidad Autónoma de San Luis Potosí, Manuel Nava #8, San Luis Potosí S.L.P., 78290, México.^bHeriot-Watt University, Riccarton, EH14 4AS, Edinburgh, Scotland, UK.

Abstract

Virtual reality systems can be used to simulate, analyze and optimize manufacturing processes including assembly. Haptic technologies enable the user to feel the force feedback from the virtual environment, leading to a more intuitive and natural way to simulate the assembly process during the design phase of new components even before any physical prototype is created. This paper presents the development of a haptic virtual reality platform to perform, plan and evaluate virtual assemblies of components. The system allows real-time manipulation and interaction of virtual components. Physics simulation engines are used to enable physic based behavior and collision detection of virtual objects in the virtual environment. One of the outstanding characteristics of the proposed platform is that the user can modify various simulation parameters during run-time, such as the weight of virtual objects, model representation algorithm and the physics simulation engine being used, which is very important in order to evaluate the influence of each parameter on the performance of virtual assembly tasks.

© 2013 The Authors. Published by Elsevier Ltd. Open access under [CC BY-NC-ND license](https://creativecommons.org/licenses/by-nc-nd/4.0/).

Selection and peer-review under responsibility of CIIIECC 2013

Keywords: Virtual assembly; haptic assembly; physic based modelling; collision detection; model representation.

1. Introduction

Haptic technologies have been studied and used in many fields such as engineering (design, manufacture, and training) and medicine (surgical planning and training). Assembly planning and evaluation is one of the engineering applications that can be widely explored by using haptic devices in virtual environments (VE). Haptic manipulation of virtual objects reduces the gap between virtual reality and real world since it allows a more natural manipulation of objects during the virtual assembly process. Several authors have proposed different virtual assembly platforms using numerous assembly techniques such as feature matching recognition [1], [2], constrained motion [3]-[6], physics based modelling [7]-[11], import of CAD assembly constraints [12]-[15], and more recently the use of haptics [16]-[21]; a description of the main characteristics of some of these platforms is presented, Table 1. In general, it has been observed that the time required to complete a virtual assembly task is always larger than the completion time of the real world assembly task. Some of the factors that may contribute to this difference are: the manipulation interface used (mouse, glove, 3D mouse, haptic device, etc.), the physics simulation performance, the manipulability of virtual objects, the shape and weight of virtual objects, the camera manipulation, the

* Corresponding author. Tel.: +52-444-8262491 Ext. 2906.

E-mail addresses: germanicoglz@gmail.com (Germanico Gonzalez-Badillo), hugoivanmc@uaslp.mx (Hugo Medellín-Castillo), t.lim@hw.ac.uk (Theodore Lim).

rendering type (stereo-view, 2D screen, head mounted displays), the force feedback, etc.

According to Howard and Vance [22], a successful virtual assembly environment requires virtual parts to emulate real parts in real world. Two basic methods for simulating physical part behaviour have been used: physically based modelling (PBM) and constraint based modelling (CBM). PBM uses Newtonian physics laws to describe the motion and forces of virtual objects and to model the physical behaviour in the virtual environment. CBM uses geometric constraints to locate the parts in the assembly position by reducing the degrees of freedom of the manipulated part, such as in a CAD system. Since simulation engines are not created for haptic rendering, the integration of haptics into a PBM platform must consider several issues such as the synchronization, non-effective collision detection, high computational cost and a negative impact on the performance of the application [23]. This is mainly because the typical frequency of haptics simulations is over 1 kHz, while in the physics simulations it is around 100 Hz [24]. Glondou [25] introduced the possibilities of implementing a modular system that relies on physical simulation and haptic rendering. Four physical simulation libraries were evaluated: Havok, PhysX, Bullet and OpenTissue, and Havok resulted in the best average computation time, stability and friction accuracy. The performance criterion was based on computation time, stability and accuracy.

The literature review makes clear that there are several challenges that VE must overcome to create a more realistic assembly process and to reduce the gap between the virtual task completion time (TCT) and TCT of the real process. Thus, this paper presents a flexible haptic assembly platform that allows the adjustment of simulation parameters during running time. This characteristic of the system allows the evaluation of the effect of the different simulation parameters on the virtual assembly performance and TCT.

Table 1. Key features of some virtual assembly platforms.

System	Year	Assembly method	Key Features	Haptic Device
HIDRA[16]	2001	Collision detection	Integrates a haptic feedback into a (dis) assembly simulation environment	Phantom desktop
			Manipulate parts using two fingers	
MIVAS [10]	2004	Physics based	Optimization techniques for complex models and assembly operations	CyberGrasp one hand
			Tracking of user movements and voice commands	
			Realistic virtual hand interaction for grasping of virtual parts	
			Documentation of assembly planning results	
VADE [12]	2004	Physics and constraints	Users can perform the assembly using hands and tools such as screw drivers	CyberGlove two handed
			During the assembly process VADE maintains a link with the CAD system	
			Let the user to make decision and design changes	
			Swept volume generation and trajectory editing	
SHARP [11]	2006	Physics based	Capability of create subassemblies	Phantom omni dual handed
			Swept volumes for maintainability	
			Network module for communication with different VR systems	
			Portable, runs on different VR systems such as HMD, CAVE, projection walls and monitors	
VEDAP-II [7]	2009	Physics based	Oriented to assembly planning and evaluation	CyberGrasp one hand
			Focuses on modeling the dynamic behavior of parts during virtual assembly operation	
MRA [21]	2009	Physics based	Use of low cost technologies for two hands assembly	6D35-45 / Wii-mote
			Real scale projection and tracking system to change point of view	
			The system demonstrates the assembly procedures and the user must repeat it	
VCG [3]	2010	Constraint based	Oriented to assembly planning and training	Virtuose 6D35-45 one hand
			Method of constraint guidance to perform the assembly	
			Use of virtual fixtures, use of mechanical constraints, intuitive assembly, on-line activation of kinematic constraints	
IMA-VR [5]	2010	Constraint based	Virtual training system for the cognitive and motor skills transfer combining haptic, gestures and visual feedback	Phantom/ LHiFAM/ GRAB two hands
			Uses the concept of spring-damper model to avoid parts interpenetration	
			Visual dynamic behavior of parts to represent manipulation of real parts	
HITsphere system [6]	2011	Physics and constraints	Immersive virtual environment with walking capability to simulate ground walking	Phantom Premium one hand
			Free manipulation of virtual objects	
			Automatic data integration interface	
			Constraint-based data model is rebuilt to construct the virtual assembly environment	

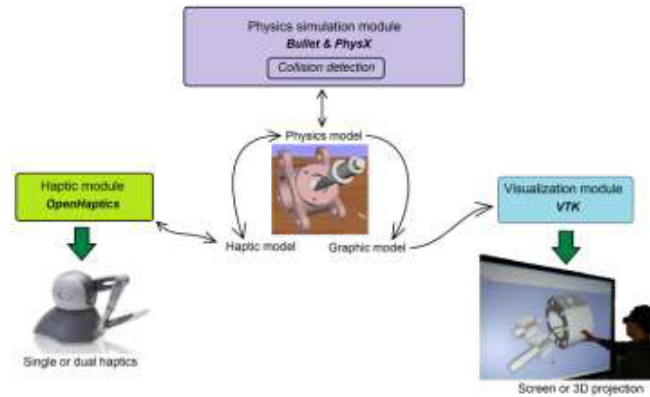


Fig.1. HAMMS architecture with three modules: Physics simulation, visualization and haptic rendering.

2. System description

The proposed haptic virtual assembly system, named as HAMMS (Haptic Assembly, Manufacturing and Machining System), has been developed using virtual reality (VR) techniques for graphic, haptic and physic rendering. Figure 1 shows the architecture of the HAMMS system, which comprises three main modules:

- Visualization module, responsible of the graphics rendering of virtual objects and virtual environments.
- Physics simulation module, responsible of simulating physical part behaviour.
- Haptic rendering module, to provide force feedback, 3D free-form manipulation of objects, collision detection and the sense of touch to the user.

The integration and implementation of the system was carried out using Microsoft Foundation Classes (MFC) and Visual C++ included in Visual Studio 2010 [26]. The graphic rendering is based on the Visualization Toolkit libraries (VTK 5.10) [27]. Open Haptics Toolkit v3.0 [28] is used for haptic rendering. In the HAMMS system single and dual haptic interaction is provided via two Omni haptic devices from Sensable. In order to enable the virtual objects with physical based behaviour and collision detection two physics simulation engines (PSEs) have been implemented: PhysX™ SDK v2.8.4 [29] and Bullet physics v2.80 [30]; the user can select either of them during running time. The GUI of the HAMMS system is shown in Figure 2.

2.1. System functionality

One of the outstanding characteristic of the system is that it allows the user to modify in real time the simulation parameters in order to evaluate and optimize the assembly simulation of the system. This modification can be made by the control panel, Figure 3, which is a dockable dialog box that can be activated or hidden during the assembly simulation. Some of the parameters that can be modified through the control panel are: haptic properties, such as stiffness, damping, cursor size and friction; and physical properties, such as mass, restitution, tolerance, time-step and gravity.



Fig.2. HAMMS interface with dual haptics and 3D stereo-view projection.

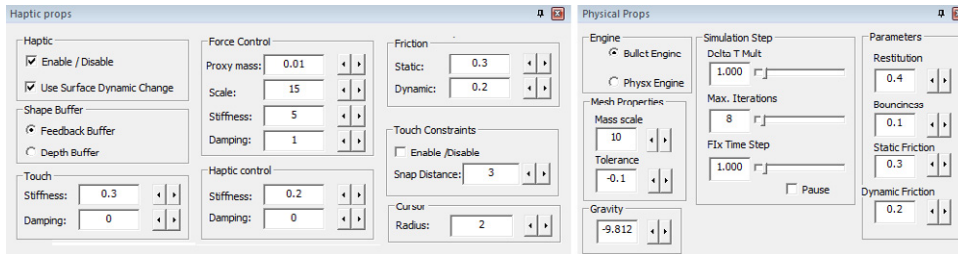


Fig.3. Control panels to modify haptic and physics simulation parameters.

Other functionalities of the HAMMS system are: use of single or dual haptics configurations, free-form manipulation of virtual objects, physics behaviour, haptic rendering, 3D graphic rendering (on projection walls or monitor), full camera manipulation with adequate haptic workspace, and different rendering algorithms (points, wireframe, solid, translucent, invisible and application of textures) for virtual objects.

2.2. Virtual model creation

Virtual models can be loaded into the HAMMS as STL, OBJ or VTK file formats. When loading a model the user must select the type of material and the collision shape. Four different materials can be selected: steel, aluminium, lead and plastic. The weight of the virtual object will depend on the material selected. The collision shape corresponds to the shape that will be used to create the physics representation of the model. Virtual models in PSEs can be represented by different types of collision shapes: primitives (box, capsule, sphere or cylinder), convex hulls and, in some PSEs, non-convex objects that can be created directly from the triangular mesh. Bullet supports collision detection between primitive shapes, convex shapes and triangular meshes by using the GIMPACT collision detection libraries [31]. It is also possible to represent non-convex shapes by using a convex decomposition algorithm such as the Hierarchical Approximate Decomposition Algorithm (HACD), which creates a compound object comprised of various convex hulls [32]. On the other hand, PhysX supports collision detection between primitives, convex hulls and compound objects. However, contact determination for triangular meshes has been discontinued since version 2.4, leaving only one possible solution: to use a convex decomposition algorithm; PhysX developers and users suggest Ratcliff's approximate convex decomposition algorithm (ACD) [33]. Another convex decomposition method called ConvexFT has been presented in [34], this method enable collision detection for triangular meshes in PhysX and has been implemented in HAMMS.

2.3. Model manipulation

Once a model is loaded into the HAMMS, three objects are generated: (i) a graphic model, used for graphic rendering on the screen; (ii) a haptic model, used for touch sensation and manipulation of virtual objects through the haptic device; and (iii) a physics model, which provides physics-based behaviour and collision detection by means of PSEs (Bullet and PhysX). When an object is being controlled by the user, the physics model is attached to the haptic model by a spring-damper system and the graphic model is updated according to the physics model data. A linear spring is used to transmit forces and translations and a torsional spring is used to transmit torques and rotations. The values of the spring constant and damping coefficient are empirically determined and adjusted to provide a smooth and stable manipulation of the virtual parts. The use of the spring-damper model invokes force feedback calculations that allow the user to feel collisions when virtual objects are being

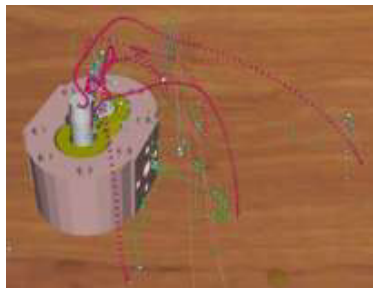


Fig.4. Chronocyclegraphs used to represent user movements when performing an assembly.

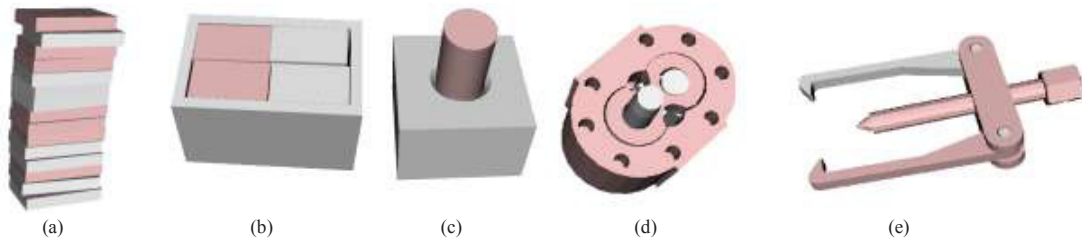


Fig.5. Assembly models: (a) pile of boxes, (b) packing box, (c) peg and hole, (d) gear oil pump, (e) bearing puller.

manipulated (e.g. weight/gravity perception). When a part-to-part collision occurs, the PSE computes the reaction forces applied to both objects. The reaction forces and torques are governed by Newton's laws or Lagrange method [22].

2.4. Data registration

The user movements and positions are logged by HAMMS during each time step. The use of Gilbreth's chronocyclegraphs [24] is also implemented in the HAMMS platform to analyse the assembly trajectories once a task has been completed, Figure 4. Gilbreth's chronocyclegraphs are representations of all the movements performed by the user in the scene; these movements are classified into three types:

1. The "wandering" movements (green spheres) are those to recognize the workspace without touching or controlling any object.
2. The "touching" movements (blue spheres) are those performed by the user when touching and recognizing the shape of an object.
3. The "controlling" movements (red spheres) are the user movements to control or manipulate an object.

The logged data can be graphically rendered or be exported as a text file for further processing.

3. System Evaluation

Three evaluations were performed in order to assess the influence of various simulation parameters on the virtual assembly:

1. Model representation evaluation.
2. Physics simulation engine evaluation.
3. Weight perception evaluation.

These evaluations were carried out using the assembly models shown in Figure 5.

3.1. Model representation evaluation

The objective of this evaluation was to analyse the geometric precision of each model representation algorithm. Thus, each assembly model (Figure 5) was represented using the different collision shape methods available in the PSEs. For non-convex objects the collision shape can be created using any of the previously described object representation algorithms (i.e. ACD, HACD, GIMPACT or ConvexFT). The geometric precision of each collision shape method was visually evaluated considering that all the geometric features, such as holes, corners, edges and concavities, of the original model, were properly reproduced by the algorithm. PhysX Visual Debugger (PVD) and MeshLab were used to visualize the model representations.

In general, the results from the evaluation of all the assembly models suggested that the GIMPACT and HACD algorithms offer the best geometric precision when representing concave objects in Bullet v2.80. On the other hand, the ConvexFT algorithm offers the best geometric precision when representing concave objects in PhysX. As an example, Figure 6 shows the results corresponding to the housing model of the gear oil pump assembly. HACD produced good results when using high resolution; however the PhysX convex elements cooking process is not fully compatible with the convex elements created by

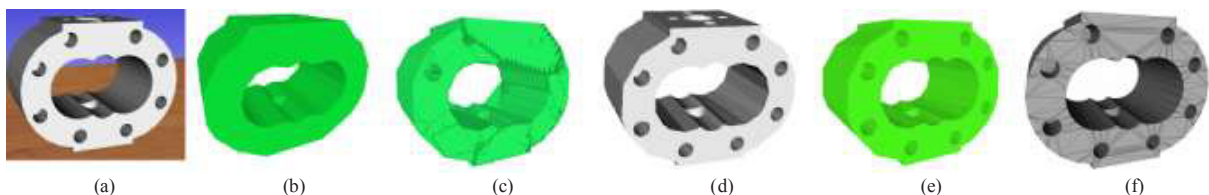


Fig.6. Model representation for collision shape of the housing: (a) graphic model (original), (b) collision shape using ACD algorithm, (c) collision shape using HACD low resolution, (d) collision shape using HACD high resolution, (e) collision shape using ConvexFT algorithm, (f) collision shape using GIMPACT.

Table 2. Minimum task completion time (TCT) for virtual assemblies performed in HAMMS.

Assembly Task	Bullet			PhysX		
	TCT (min)	Std. Dev.	Rep.	TCT (min)	Std. Dev.	Rep.
Pile of cubes	02:41.8	00:09.8	GIMPACT	03:23.7	00:06.9	HACD
Packing box	03:19.2	00:19.6	GIMPACT	02:09.7	00:03.8	HACD
Peg and hole	00:05.4	00:00.4	GIMPACT	00:06.5	00:02.4	ConvexFT
Gear oil pump	00:58.3	00:09.5	GIMPACT	01:21.0	00:18.7	ConvexFT
Bearing puller	01:33.9	00:14.7	GIMPACT	No feasible	---	---

HACD, producing a wrong representation of the model.

3.2. Physics simulation engine evaluation

The objective of this evaluation was to analyse the performance of each PSE in order to identify the usefulness, capability, pros and cons in virtual assembly tasks. Two PSEs were considered, Bullet and PhysX. Each of the five assembly tasks (Figure 5) was performed five times by an experienced user using a single haptic device. The TCT began when the user touched the first object and finished when the last part was released at its final assembly position.

The results are summarized in Table 2, where it can be observed the minimum TCT, the standard deviation (Std. Dev.) and the model representation algorithm (Rep.) that led to the minimum TCT. The results of this study are preliminary and a complete evaluation using more users is considered as future work to confirm these results and provide relevant findings.

Also, real and virtual assembly tasks of the pump assembly were carried out. An average TCT of 37 seconds was obtained in the real assembly task whilst in the virtual task the TCT was 58.3 seconds using Bullet (56% more than the real assembly) and 1.21 minutes using PhysX. The bearing puller assembly task could not be performed using PhysX due to its tight tolerances.

3.3. Weight perception evaluation

To evaluate the influence of the virtual weight on the assembly performance, particularly on the TCT, the gear oil pump assembly (Figure 5) was selected. Eight levels of weight for the pump components were defined. In the first level, L1, the weight of all the components is near to zero, and in the next levels the weight increase gradually. In the last level, L8, the weight of all the components is the maximum. The weight of the heaviest pump component is limited by the maximum feedback force of the haptic device, and the weight of the other components is scaled. In this evaluation the virtual assembly is also compared with the real assembly. Figure 7a shows the TCT results obtained for one hand assembly whilst Figure 7b shows the results for dual handed assembly.

It is observed that the virtual TCT for both, one and dual hand assembly, was closer to the real TCT when the weight of virtual components is minimal (0.02 N), level L1. The assembly time was smaller in Bullet (49 seconds for one hand and 52 for dual handed assembly) than in PhysX (64 seconds for one hand and 75 seconds for dual handed assembly). On the other hand, the real assembly TCT for one hand was 37 seconds and for dual handed assembly was 27 seconds. It can be concluded that the weight of virtual objects, rendered to the user through the haptic device, affects the performance of the assembly process; as the weight of virtual objects increases, the TCT will also increase.

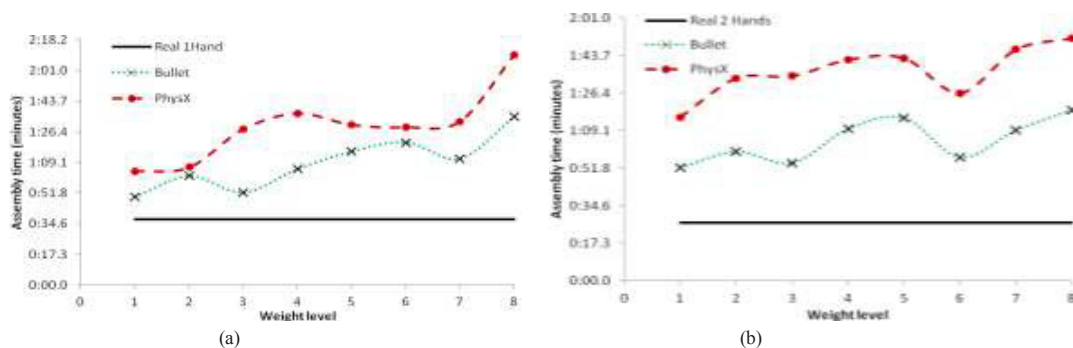


Fig.7. Results of weight perception evaluation, a) one hand task completion time, b) dual handed TCT .

4. Conclusions and future work

A flexible platform for haptic virtual assemblies, named as HAMMS, has been presented. The system has different functionalities that allow the user to modify simulation parameters and components during running time. This characteristic enables the evaluation of the effect of different parameters on the virtual assembly performance. The results have shown that GIMPACT and HACD algorithms for create the collision shape within the physics simulation engine provide the best geometric precision of the represented model in Bullet v2.80, whilst the ConvexFT algorithm provides the best geometric precision when loading a model in PhysX. The results also suggest that for assembly tasks that involve non-complex geometries like boxes and cylinders (primitives), the use of PhysX offers a better performance than Bullet; however when the assembly comprises more complex shape components, Bullet has superior performance than PhysX. It has been also observed that the TCT is directly affected by the weight of virtual objects; i.e. as the weight of virtual objects increases the TCT also increases.

A more comprehensive study must be carried out including the effect of other simulation parameters such as restitution coefficient, control spring stiffness, friction, etc. The use of assembly constraints to assist the virtual assembly process and the improvement of the algorithms to create the collision shape are also considered as future work.

Acknowledgements

The authors acknowledge the financial support from CONACYT (National Science and Technology Council), Research grant CB-2010-01-154430.

References

- [1] Jacob R, Mitrouchev P, Léon J, “Assembly simulation incorporating component mobility modelling” *Int. Journal on Interactive Design and Manufacturing (JIJDeM)*, vol. 5, no. 2, pp. 119-132, 2011.
- [2] Sato K, Date H, Onosato M, “Fast matching, combinations extraction and configuration of mesh models using graph-based feature representation”, *Int J Interact Des Manuf. (JIJDeM)*, vol. 5, pp. 133–136, 2011.
- [3] Tching L, Dumont G, Perret J, “Interactive simulation of CAD models assemblies using virtual constraint guidance”. *Int. J. on Interactive Design and Manufacturing*, vol. 4, no. 2, pp. 95-102, 2010.
- [4] Zaldivar-Colado U, Garbaya S, “Virtual Assembly Environment Modelling”, *Proceedings of the ASME/AFM 2009 World Conference on Innovative Virtual Reality WINVR2009*, February 25-26, 2009, Chalon-sur-Saône, France, 2009.
- [5] Gutiérrez T, Rodríguez J, Vélaz Y, Casado S, Suescun A, Sánchez EJ, “IMA-VR: A Multimodal Virtual Training System for Skills Transfer in Industrial Maintenance and Assembly Tasks”, *19th IEEE International Symposium on Robot and Human Interactive Communication*, Principe di Piemonte - Viareggio, Italy, Sept. 12-15, 2010.
- [6] Xia P, Lopes AM, Restivo MT, Yao Y, “A new type haptics-based virtual environment system for assembly training of complex products”, *The International Journal of Advanced Manufacturing Technology*, vol. 58, no. 1-4, pp. 379-396, 2012.
- [7] Garbaya S, Zaldivar-Colado U, “Modeling dynamic behavior of parts in virtual assembly environment”, *Proceedings of the ASME/AFM 2009 World Conference on Innovative Virtual Reality WINVR2009*, Chalon-sur-Saône, France.
- [8] Gupta R, Whitney D, Zeltzer D, “Prototyping and Design for Assembly analysis using Multimodal virtual Environments”, *Computer-Aided Design*, vol 29. no 8, pp. 585-597, 1997.
- [9] Aleotti J, Caselli S, “Physics based virtual reality for task learning and intelligent disassembly planning”, *Virtual Reality*, vol. 15, pp. 41–54, 2011.
- [10] Wan H, Gao S, Peng Q, Dai G, Zhang F, “MIVAS: a multi-modal immersive virtual assembly system”, *Proceedings of the ASME Design Engineering Technical Conference*, Salt Lake City, UT, pp.113-122, 2004.
- [11] Seth, A., Su H. J., Vance J. M. (2006). “SHARP: a system for haptic assembly & realistic prototyping”, ASME Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Philadelphia, PA, USA, 2006.
- [12] Jayaram S, Jayaram U, Wang Y, Tirumali H, Lyons K, Hart P, “VADE: a Virtual Assembly Design Environment”, *Computer Graphics and Applications. IEEE*, vol. 19, Issue 6, pp. 44 – 50, 1999.
- [13] Chamaret D, Ullah S, Naud PMR, “Integration and evaluation of haptic feedbacks: from CAD models to virtual prototyping” , *Int J Interact Des Manuf*. Vol. 4, pp. 87–94, 2010.
- [14] Chen CJ, Ong SK, Nee AYC, Zhou YQ, “Haptic-based interactive path planning for a virtual robot arm”, *Int J Interact Des Manuf*, vol. 4, pp. 113–123, 2010.
- [15] Cheng-jun C, Yun-feng W, Niu L, “Research on Interaction for Virtual Assembly System with Force Feedback”, *ICIC '10 Proceedings of the 2010 Third International Conference on Information and Computing*, vol 02, pp. 147-150, 2010.
- [16] Coutee AS, McDermott SD, Bras B, “A haptic assembly and disassembly simulation environment and associated computational load optimization techniques”, *Journal of Computing and Information Science in Engineering*, vol.1, pp.113-122, 2001.
- [17] Ji J, Lee K.M, Zhang S, “Cantilever Snap-Fit Performance Analysis for Haptic Evaluation”, *Journal of Mechanical Design, ASME*, vol. 133, no. 12, pp. 1-8, 2011.
- [18] Christiand, Yoon J, “Assembly simulations in virtual environments with optimized haptic path and sequence”, *Robotics and Computer-Integrated Manufacturing*, vol. 27, pp. 306–317, 2011.
- [19] Ladeveze N, Fourquet J, Puel B, “Interactive path planning for haptic assistance in assembly tasks”, *Computers & Graphics* vol. 34, pp. 17–25, 2010.

- [20] Liu M, Wang D, Zhang Y, “A novel haptic rendering algorithm for stable and precise 6-dof virtual assembly”, *Proceedings of the ASME 2010 World Conference on Innovative Virtual Reality*, May 12-14, Ames, Iowa, USA WINVR2010-3724, 2010.
- [21] Bordegoni M, Cugini U, Belluco P, Aliverti M, “Evaluation of a Haptic-Based Interaction System for Virtual Manual Assembly”, *Virtual and Mixed Reality*, LNCS 5622, pp. 303–312, 2009.
- [22] Howard BM, Vance JM, “Desktop haptic virtual assembly using physically based modeling”. *Virtual Reality*, vol. 11, pp. 207-215, 2007.
- [23] Seugling A, Rölin M, “Evaluation of physics engines and implementation of a physics module in a 3D authoring tool”. *Master's thesis in computing science Umea University*, 2006.
- [24] Ritchie JM, Lim T, Medellin H, Sung RS, “A haptic based virtual assembly system for the generation of assembly process plans”. *XV congreso internacional SOMIM*, 23- 25 September 2009, Cd. Obregon, Sonora. México.
- [25] Glondu L, Marchal M, Dumont G, “Evaluation of physical simulation libraries for haptic rendering of contacts between rigid bodies”. *Proceedings of ASME , WINVR 2010*, may 12-14, Ames, Iowa, USA, WINVR201-3726, pp. 41-49.
- [26] Microsoft Foundation Classes, Updated information available at <http://msdn.microsoft.com/en-us/library/>. Consulted on 9/10/12.
- [27] Visualization Toolkit, Updated information available at <http://www.vtk.org/VTK/resources/software.html>. Consulted on 9/10/12.
- [28] OpenHaptics Toolkit, Updated information available at <http://www.sensable.com/products-openhaptics-toolkit.htm>. Consulted on 9/10/12.
- [29] PhysX developer zone, Updated information available at <http://developer.nvidia.com/physx>. Consulted on 9/10/12.
- [30] Bullet Physics Library, Updated information available at <http://bulletphysics.org/wordpress/>. Consulted on 9/10/12.
- [31] Gimpact, GIMPACT, geometric tools for VR, Updated information available at <http://gimpact.sourceforge.net>. Consulted on 24/04/12.
- [32] Mamou K and Ghorbel F, “A Simple and Efficient Approach for 3D Mesh Approximate Convex Decomposition,” *16th IEEE International Conference on Image Processing (ICIP '09)*, pp. 3501-3504, Cairo, Nov 7-10, 2009.
- [33] John Ratcliff's code suppository, Updated information available at <http://codesuppository.blogspot.com/2006/04/approximate-convex-decomposition.html>. Consulted on 5/02/12.
- [34] Gonzalez G , Medellin HI, Lim T, Ritchie JM, Sung RC, “3D object representation for physics simulation engines and its effect on virtual assembly tasks” *Proceedings of the ASME 2012 International Design Engineering Technical Conferences & Computer and Information in Engineering Conference IDETC/CIE 2012*, Chicago, Il, USA. 2012.