

**IDETC2016-59202**

## **A UNITY SKETCH BASED MODELLING ENVIRONMENT FOR VIRTUAL ASSEMBLY AND MACHINING TO EVALUATE DFMA METRICS**

**Avery Read**

School of Engineering & Physical Sciences  
Heriot Watt University, Edinburgh, UK

**James Ritchie**

School of Engineering & Physical Sciences  
Heriot Watt University, Edinburgh, UK

**Theodore Lim**

School of Engineering & Physical Sciences  
Heriot Watt University, Edinburgh, UK

### **ABSTRACT**

Current tools used to carry out design for manufacture and assembly (DFMA) evaluations are time consuming to use. This increases the cost of bringing a product to market by extending the length of the design process. The steps that comprise the design, analysis and redesign process are typically carried out in series over long periods of time using different tools. Introducing concurrent engineering practices could significantly reduce the time taken to complete this process and improve workplace DFMA learning. An opportunity exists to create and test an integrated real-time DFMA tool using the UNITY game engine, which could potentially address these problems. If this approach can be achieved it has the potential to decrease the time taken for the design process and enable a greater number of solutions to be considered potentially leading to a more optimal design solution. Having a more optimal design could lead to major cost reduction in later stages of product development by reducing the work needed to plan and carry out the manufacturing process and creating a product that is easier and less costly to maintain.

This paper reports on a pilot haptic sketch-based system to investigate its feasibility to conduct concurrent DFMA. .

### **1. INTRODUCTION**

One of the principal ways to deliver innovation and to advance production is through Design for Manufacture and Assembly (DFMA). An increasing challenge for manufactures is how to evaluate different product design options and how to compare one set of solutions with another. Currently available software implementations of DFMA methods have utilized a WIMP (windows, icons, menus, and pointing) based approach to user interaction.

A widely investigated alternative approach to user interaction is that of a sketch based interface combined with gestural commands. This type of approach has been seen to provide a faster and more intuitive way of translating concepts into designed geometry. [1]

The aim of this research is to determine if an integrated haptic enabled sketch based modelling environment can be combined with virtual assembly and machining to evaluate DFMA metrics as products are being designed in real-time. The initial focus is on design for Assembly (DFA) prior to embedding in a full fledged haptic sketch-based DFMA system.

### **2. BACKGROUND**

DFA facilitates the design of discrete products by analysing key assembly attributes to determine how the number of parts and associated assembly operations can be reduced with respect to cost and lead time. Two well known approaches to DFA analysis are the Lucas [2] and the Boothroyd and Dewhurst's [3] methods. Both methods have been extended into a number of areas with a variety of technological support, which are briefly covered in this section.

The first stage in software solutions for DFA analysis was the computerized version of Boothroyd and Dewhurst handbook [4]. This approach while removing the user from manual calculations still requires a lengthy analysis process consisting of forty-nine questions per part [5].

Improving on these new DFA software solutions saw the implementation of automatic feature extraction directly from CAD models. Early examples include IDEARS [6] which used a feature based approach to extract some basic geometrical features. A similar approach was taken by Li and Hwang [7]. Despite being semi-automated both of these were still heavily

reliant on user input for most DFA criteria. Advances in research began to see systems such as DFAES [8], an approach based on a series of expert systems, begin to appear. Fuzzy-based approaches such as FuzzyDFA [9] [10] allowed DFA analysis to be performed earlier in the design process as it is less reliant on details of part design.

Dalglish et. al. [11] proposed an approach to DFA analysis to be used in the early stages of design. Unique here was feedback being provided to users as they are defining the product structure and designing the part geometry within an embedded CAD modeller. The system incorporated an expert system support for DFA [12], interactive evaluation of the assembly sequence [13] and automatic symmetry detection [14]. Further developments using this approach focus on analysing complexity [15] and the incorporation of manufacturability assessment [16].

The background findings suggest that more enhanced intuitive interaction capability can benefit DFA. In light of this, a research question is raised: *Can haptic virtual reality with sketch-based interactivity provide new methods for evaluating and automating DFA criteria while capturing associated DFA knowledge in a unique fashion?* This paper now describes a pilot platform which will be investigated to determine the benefits, if any, of such systems.

### 3. SKETCH BASED INTERFACES

Sketch based interfaces for modelling can be roughly split into two main types of systems; those that use offline recognition and those that use online recognition. Offline recognition systems use sketches prepared in advance, e.g. those made during a design meeting, as an input to an automated or human-assisted modelling process [17]. The latter has the advantage where the user interacts with the system as they would sketch with a pen and paper. Online recognition in contrast is where the user sketches the model incrementally while the computer interprets this input as or soon after each part is drawn. There is greater interactivity, which opens up the possibility of rapidly creating a model, carrying out a simulation of the properties of the model (e.g. finite element analysis[18]) and rapidly iterating on the model to produce better conceptual designs. Most online recognition systems use simple operations, and having users resolve ambiguities enables more complex models to be produced. While the modelling methods between offline and online system do not vary greatly they require the user to learn a set of operations before being able to create a model. This could increase the cognitive load as users focus more on how they are modelling rather than what they are modelling.

Surveys on modelling techniques undertaken by Cook[19] and Olsen [20] highlight a significant gap in sketching systems which combine drawing function with structured relational geometries and forms to support downstream evaluation and analysis. However, Olsen et al. emphasizes the need for a hybrid approach combining drawing and human like understanding and interpretation. Their categorization of model creation methods and system attributes are relevant to the study involving DFMA reported here.

Despite the volume of research in the area of sketch based interfaces for modeling (SBIM) no method has been devised that address the functionality needed for DFMA.

SBIM applications generally use either reconstruction or extrusion approaches. Extrusion approaches involve the user sketching profiles and extruding or sweeping those profiles along a path, e.g. [21]–[24]. Reconstruction approaches involve the user sketching a 2.5D representation, which the system then interprets the intended 3D shape, e.g. [25]–[27]. A variation on this approach allows the user to utilize a more interactive method of creation where the model is sketched in sections and the reconstruction algorithm is applied allowing the user to see how the system is interpreting their input sketches [28]. In GIDeS++ [29] the dimensions of sketched features in the model can be altered allowing the user precise control. This feature facilitates the creation of more detailed models suitable for further design and analysis activities.

Interfaces typically found in sketch based modelling systems have moved away from a WIMP style interface, where the user primarily interacts with the program via menus and buttons, instead favouring gestural and suggestive approaches.

Table 1 lists sketch based modelling systems in 3D space, excluding systems exclusively dealing with pure sketching in 3D space. Of these reviewed 5 out of 8 include passive haptic feedback. This was accomplished in two different ways; the first being the use of a desk / table to sketch [1], [30], and the second being the use of a see through portable panel to sketch on (or select menu items on) as if it were a sketchpad being moved in the virtual space [31][32][33]. None of the systems include any active haptic feedback. However, Wesche and Seidel [34] stated that the addition of active haptic feedback would be a useful addition to their systems.

		Interaction space		Input devices					Model creation method				Visual feedback			Other				
		Desktop	Workbench	Full scale	Stylus	Tablet	Hand	Other artefacts	Extrusion	Blobby inflation	Wireframe fill	Iconic	Menu	Monitor	HMD	Stereoscopic	Large display	CAVE	Passive haptic	Multimodal
1998	ErgoDesk [30]		•																	
2000	Bimber et al [31]			•	•	•													•	•
2001	FreeDrawer [34]		•		•															
2004	Fleish et al [32]		•		•	•	•													•
2006	Construct3D [33]				•	•	•					•		•						•
2010	Immersive Fibermesh [35]				•	•				•										
2013	MockupBuilder [1]		•					•								•				•
2015	SmartPen [36]	•			•						•			•						

**Table 1:** 3D sketch based interfaces for modelling.

The use of a SBIM can provide several advantages over more traditional WIMP based CAD modelling interfaces. Research shows they are more intuitive to use, take less time to learn, allow for faster modelling times[1]. However, the findings indicate that SBIM has not yet been used to conduct DFMA analysis. This is the aim of this reported work.

#### 4. SYSTEM OVERVIEW

A proposed concept for a haptic SBIM is illustrated in Figure 1.

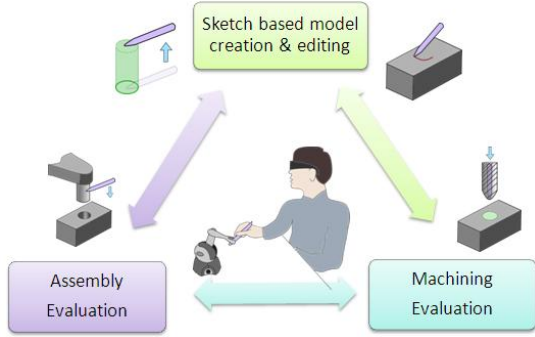


Figure 1: Overview of haptic SBIM DFMA Framework.

The sketch based modelling module allows the user to create and edit imported geometry. A combination of reconstruction and extrusion methods then translates sketches into 3D shapes. Once completed DFMA criteria (Table 1) are automatically generated for use by the assembly and manufacturing evaluation modules. The user can continue to edit the model and successive analysis performed. This approach is analogous to the FEA (finite element analysis) informed design tool mentioned in [18] where FEA details are available to the designer as they alter their model using a sketch-based interface. The final output of the modelling module is the geometry of the parts that comprise the model assembly.

Table 1: The proposed DFMA Criteria [37] and output generated by each module.

Module	DFMA Criteria	Output
<b>Modeller</b>	Volume Bounding box Mass Axis and types of symmetry Potential for obstruction & interference Ease of insertion	Models of parts Redesign suggestions (Handling)
<b>Assembly</b>	Order of assembly Assembly reorientations Ease of assembly Time for each operation Cost of assembly	Assembly plan Redesign Suggestions (Fixing)
<b>Machining</b>	Number of operations Time to machine Ease of machining Cost of machining	Process plan Redesign suggestions (Machining)

The assembly module presents the user with all of the parts in the model and allows the user to fit them together to form the final assembly. Logging techniques [38][39] capture interaction information needed to create the assembly plan in addition to factors necessary to perform DFA (design for assembly) analysis such as the number of times the assembly needs to be re-orientated and the potential for certain parts to be combined. The DFA criteria used follows Boothroyd and Dewhurst [37, pp. 85–146].

The machining module presents the user with a billet of raw material with the final model embedded in it along with a range of work holding options and tools for machining. The user can test and compare different work holding setups to find an optimum setup. They can also test and compare different machining sequences and processes. As before the system automatically captures information needed for the DFM (design for machining) analysis. The DFM criteria follows Boothroyd and Dewhurst [37, pp. 267–337].

#### 5. IMPLEMENTATION

A prototype platform was developed using the Unity3D game engine, which includes support for physics interactions and graphical rendering (Figure 2). It also provides multi platform targeting and support for the use of different input devices.

The haptic device chosen for the system is the Phantom Omni. This device allows 6DOF input for the user and provides 3DOF haptic feedback, providing the user with a sense of weight, surface texture stiffness and 3D shape. This has been shown to be useful for the design of products in virtual space [40].

Open Haptics library [41] is used alongside a plug-in to allow successful UNITY integration [42] as UNITY has no native support of haptic devices.

To provide import and export of model files in STEP format a wrapper was developed for the Open Cascade libraries [43]. This is also used for modelling operations and the generation of mesh models as none of these features are natively supported by UNITY.

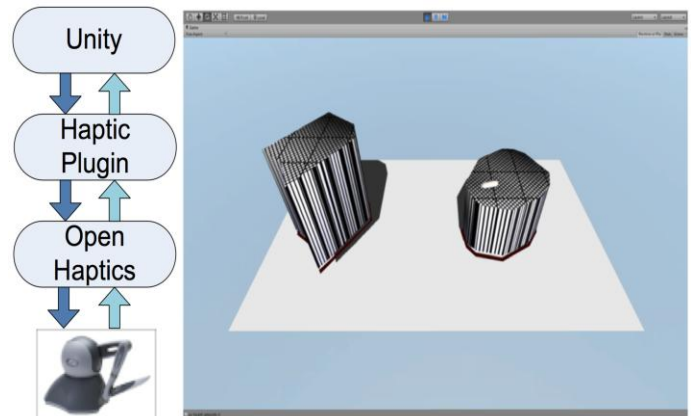
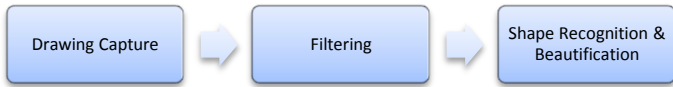


Figure 2: Haptic SBIM implemented with UNITY engine.

#### 6. MODELLING WITH THE HAPTIC SBIM

An overview of the steps involved in the conversion of the captured movements of the haptic pen into sketched lines useable in the modelling process is shown in Figure 3.

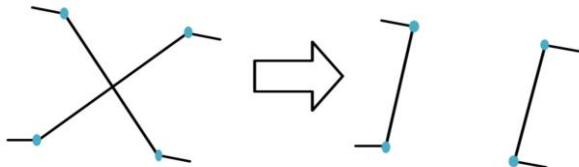


**Figure 3:** Outline of Sketching with haptic device.

*Drawing capture* - With the haptic pen the user draws on a 2d plane within the scene. The process of the user sketching is broken down into strokes. A stroke is considered as the path taken by the tip of the pen from the time when the pen tip makes contact with the drawing plane until the pen tip ceases to make contact with the drawing plane. Tracking the pen tip at regular time intervals during drawing captures the motion of the pen tip. After the stroke is drawn these points are resampled, filtered and then beautified and stored as either open or closed strokes in the scene.

*Filtering* - A point refinement algorithm [44] is used to filter the sketched profile. This removes extra points from hand jitter during drawing and significantly reduces the number of points required to represent the shape of the line drawn without significant loss of the user's intended shape profile. The removal of these extra points also serves to make later stages of the shape recognition process significantly faster.

*Shape recognition & beautification* - Sketch strokes are first classified as either closed (the stroke is connected at both ends) or open (both ends do not connect). It is then checked for self-intersection. If the stroke does self intersect it is split at the point of intersection creating two sub line segments which will be treated as separate strokes from this point onwards (Figure 4).



**Figure 4:** Splitting and beautifying an intersecting stroke.

After all intersections have been removed each of the new individual strokes can be beautified in a process similar to the one demonstrated in [1]. The beautification process splits each of the strokes into linear and curved segments after which all of the linear segments are checked for similar lengths, parallel and perpendicular segments, and adjusted accordingly. The beautified shapes are then compared against a set of rules that define a set of commonly drawn shapes using a fuzzy logic approach similar to the one used in [45]

This classification can be used to further refine the stroke, e.g. if a circle is found it can be redrawn using the drawn centre and diameter as a guide. It can also be used in subsequent modelling operations to form the base of more complex features, e.g. if the stroke is classified as circular and later in the editing process this stroke is extruded, this new feature would be classified as a cylindrical feature.

The user can also change the classification of the feature if it is interpreted incorrectly.

Extrusion is performed when the user selects a closed stroke with the haptic pen. Using a push or pull motion while the stroke is selected will displace the stroke profile along a direction parallel to the plane on which the stroke lies. Using this technique the user can create new extruded features, protrusions from existing parts and holes in existing parts. The same basic principle can be extended to the creation of featured extruded along a non-linear path with the additional selection of another stroke that defines the extrusion path.

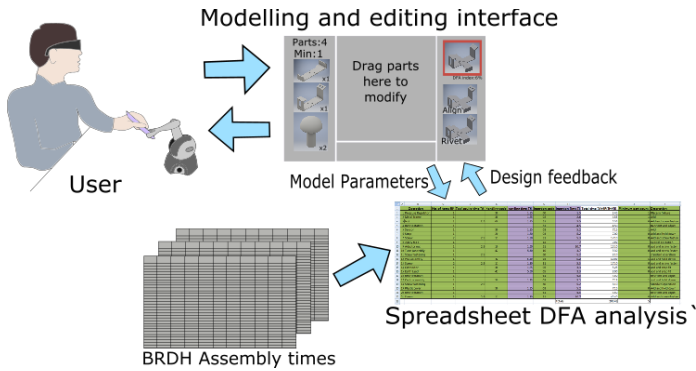
## 7. ASSEMBLY AND DFA ANALYSIS

To allow the assembly of the model parts in real time it is essential to have an appropriately detailed collision mesh with concave and convex features. However, the underlying physics engine in UNITY only supports convex hulls. Within the unity environment this problem is usually addressed by either an approximation made from primitive collision shapes (created manually by the user) or with the use of automatically generated convex hull collision meshes. The convexFT [46] technique is implemented here.

Assembly with haptic feedback is logged within the system to create an assembly plan.

Concurrently, DFA principles rank the assembly and highlight areas in which the design of the assembly could be improved. This process begins on importing the model into the UNITY sketch-based modeling platform. Key geometric and other parameters related to the Boothroyd and Dewhurst criteria are set within a DFA matrix, e.g. symmetry/asymmetry, scale, relative position, parts count, fasteners (stock part types), mass, etc. However, there are still some criteria needed for the DFA analysis that cannot be evaluated automatically, e.g. these features need to be manually set by the user. This is achieved by annotating the parts and sub assembly models with attributes and to generate build sequences/ plans the latter in a manner similar to [47] These are subsequently used by the system to derive the missing DFA criteria e.g. tangling, thermal/electrical isolation, power tool use/manual assembly, need for special handling, dynamic/stationary, function, etc. All of these inputs are fed into an Excel DFA matrix spreadsheet to produce the necessary results. This is illustrated in Figure 5 with detail of a typical results matrix shown later in Figure 9 and Figure 10.





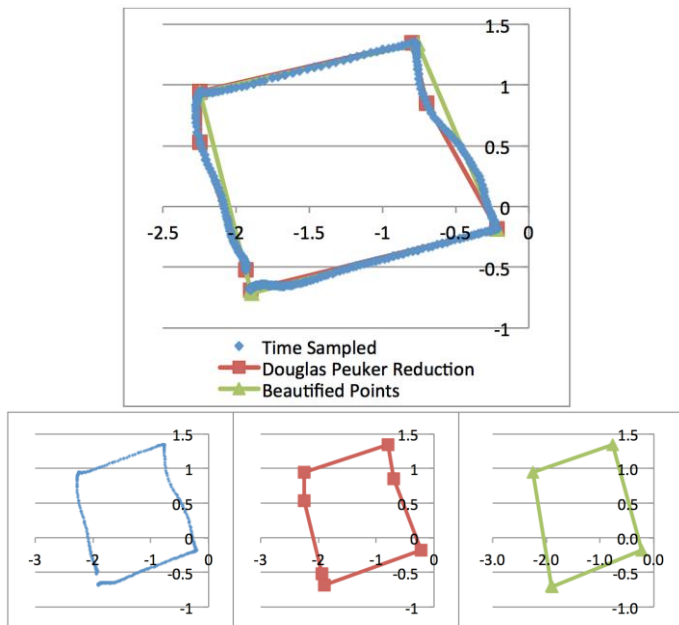
**Figure 5:** Linking DFA criteria to External Spreadsheets

## 8. MACHINING ANALYSIS

Machining will be implemented at a later stage of this work and is the final module of the haptic SBIM DFMA system. The only current relevant work being that by Fletcher et. al [47] on haptic computer aided process planning is adapted here for manufacturing analysis.

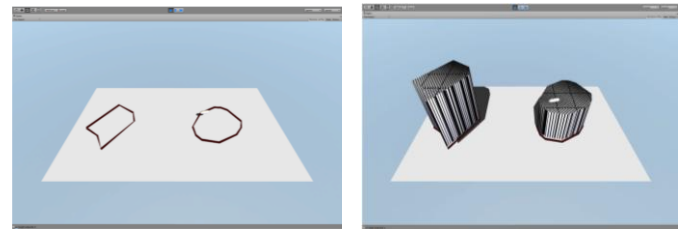
## 9. RESULTS & DISCUSSION

The results presented here are the preliminary trials of the proposed haptic based integrated DFMA system combining features of sketch based modelling, virtual assembly, and virtual machining.



**Figure 6:** Refinement of the time-sampled points of a sketch. (All units in mm)

Figure 6 shows the refinement of points associated to a sketch of a rectangular shape. Point reduction and beautification is achieved via the Ramer–Douglas–Peucker algorithm on sketched input (Figure 7). The process is fast and takes between 16ms and 100ms depending on the speed of sketching and the complexity of the shape drawn.



**Figure 7:** Line sketching and sketch profile extrusion.

STEP imports allow premade assemblies or part models to be redesigned using the haptic SBIM system. The assembly and machining sequences are logged while the user is interacting and manipulating the part/assembly models. The system automatically gathers and updates key features of the parts, assembly operations and machining processes ready for performing DFMA analysis.

Once the user has completed this initial setup process they are presented with the results of the initial DFMA analysis. Figure 8 shows the graphical user interface after the initial DFMA.



**Figure 8:** Interface after initial DFMA analysis.

All of the parts contained within the assembly can be viewed alongside the number of each part required and whether this part has been identified as a candidate for merging / deletion. Any feedback on potential design improvements to the parts will also be displayed here; e.g. areas where the part could be modified to improve handling time.

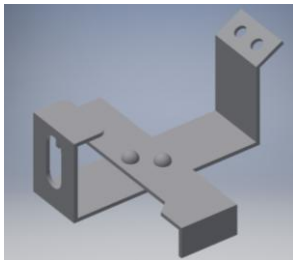
On the right the assembly operations can be viewed. A brief description of the operation is displayed, for instance an insertion operation would display “Insertion”, a reorientation operation “reorientation”. Alongside the calculated operation time is presented to the user, colour coded to indicate which operations are taking up the most time. Any feedback regarding potential design improvements to improve assemblability is displayed here; e.g. “this operation is lengthy due to mismatched tolerance”.

The centre panel contains the modelling environment where the user can create, alter and merge parts using sketch based modelling techniques. As the user receives feedback

they are free to make changes to the design and as they do the parameters will be updated automatically and feedback given. This allows the user to immediately assess the impact of changes to the design. Figure 9 and Figure 10 illustrate the ‘before’ and ‘after’ DFA case study example with corresponding spreadsheet analysis.

Suggestions for design improvement

- Lower Bracket doesn’t meet criteria for minimum parts
  - Consider combining it with Upper Bracket
- Rivet doesn’t meet criteria for minimum parts as its primary function is to connect other parts
  - Consider merging other parts to eliminate need for Rivet
- Rivet is difficult to fix to the assembly
  - Consider adding self locating features
  - Consider redesign to eliminate the need for tools



	Type	Repeats	Handling Code	Handling Time	Insertion Code
1 Upper Bracket	Add	1	30	1.95	00
2 Lower Bracket Half	Add	1	30	1.95	00
3 Rivet	Add	2	10	1.50	31
<b>Insertion Time</b>		<b>Total Time</b>	<b>Minimum Part Count</b>		
1.50		3.45	1		
1.50		3.45	0		
5.30		16.50	0		
<b>Totals</b>		<b>23.40</b>	<b>1</b>		
Not min Part	Fastener	Connector	Seperate Operation	Difficult Handling	Difficult Insertion
✓					✓

Figure 9: Original part and initial DFMA analysis results. (Times in seconds)

Figure 10 depicts the merging sequence of the two parts. Once the user has completed this process the parameters associated with the merged parts and deleted parts are updated and the DFMA metrics are reevaluated providing feedback on the design change.

	Type	Repeats	Handling Code	Handling Time	Insertion Code
1 Merged Bracket	Add	1	30	1.95	00
<b>Insertion Time</b>		<b>Total Time</b>	<b>Minimum Part Count</b>		
1.50		3.45	1		
<b>Totals</b>		<b>3.45</b>	<b>1</b>		
Not min Part	Fastener	Connector	Seperate Operation	Difficult Handling	Difficult Insertion

Figure 10: Part modification and optimisation via DFMA analysis. Merging process to erase rivets. (Times in seconds)

The timings for each operation are reported back to the user and the parts with potential to be merged are highlighted. Additional feedback on the design is given, e.g. “this part is asymmetrical. For easier handling consider either making symmetrical or making it pronounced asymmetrical”.

As the user adds or deletes parts, the system automatically gathers the information needed to analyse the model, which is derived from both the geometry of the individual parts and the way in which the user assembles the parts. Using this information the parts and operations are classified using methods from Boothroyd and Dewhurst and then the time taken for each operation is calculated using this data. Additionally parts that will be combined are identified and the theoretical minimum part count is calculated.

Figure 11 shows the system being used to assemble the bracket assembly. The real-time visualisation and feedback during the DFMA evaluation provides workplace problem-solving learning. The process of design engineering and requirements for optimizing design is more easily understood and appreciated.

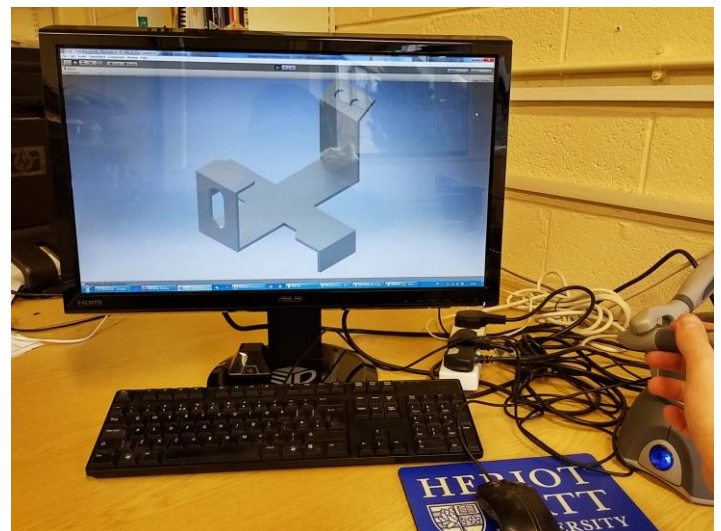


Figure 11: Bracket after merging (physical setup).

## 10. CONCLUSION

This paper has demonstrated the potential to use haptic virtual reality as a means of evaluating product designs by carrying out DFA analysis using currently available techniques, in this case the classic Boothroyd and Dewhurst analysis. Using the flexibility of the UNITY engine it will be possible to automatically and semi-automatically define the DFA criteria associated with a specific assembly design, populate a DFA array and then analyse this to output English syntax recommendations regarding how a product can better designed to be assembled. This system architecture and associated methodology will provide an ideal experimental platform from which to expand and test haptic DFMA. Indeed, embedding haptic sketching and annotation into the process also points to the capability to provide effective, intuitive support and instructions for the redesign of a product, although a more complex form of sketching will be required to facilitate full non-CAD package modification to any design concept.

The integration of DFA combined with the ability to actually virtually assemble the product and taking advantage of the already proven benefits of haptic interaction during such an assembly process points to significant potential benefits for more intuitive product design tools in the future.

Future work will focus on incorporating design for manufacture [47] (DFM) within the system based on previously successful virtual process planning work and then rigorously testing the system both from functionality and user perspectives via a series of proven DFMA case studies.

## ACKNOWLEDGMENTS

The work presented herein is partially funded under the European H2020 Programme BEACONING project, Grant Agreement nr. 687676.

- [1] B. R. De Araújo, G. Casiez, J. a. Jorge, and M. Hachet, "Mockup Builder: 3D modeling on and above the surface," *Comput. Graph.*, vol. 37, no. 3, pp. 165–178, 2013.
- [2] Lucas Engineering Systems Ltd, *Design For Manufacture and Assembly Practitioners Manual. Version 10*. University of Hull, 1993.
- [3] G. Boothroyd and L. Alting, "Design for Assembly and Disassembly," *CIRP Ann. - Manuf. Technol.*, vol. 41, no. 2, pp. 625–636, 1992.
- [4] "DFMA." [Online]. Available: <http://www.dfma.com/>. [Accessed: 24-May-2016].
- [5] E. Owensby, A. Shanthakumar, V. Rayate, E. Namouz, and J. D. Summers, "Evaluation and Comparison of Two Design for Assembly Methods: Subjectivity of Information Inputs," *Int. Des. Eng. Tech. Conf. Comput. Inf. Eng. Conf.*, vol. 9, pp. 721–731, 2011.
- [6] R. Sturges and M. Kilani, "Towards an integrated design for an assembly evaluation and reasoning system," *Comput. Des.*, vol. 24, no. 2, pp. 67–79, 1992.
- [7] R.-K. Li and C.-L. Hwang, "A framework for automatic DFA system development," *Comput. Ind. Eng.*, vol. 22, no. 4, pp. 403–413, 1992.
- [8] X. F. Zha, S. Y. E. Lim, and S. C. Fok, "Integrated knowledge-based approach and system for product design for assembly," *Int. J. Comput. Integr. Manuf.*, vol. 12, no. 3, pp. 211–237, 1999.
- [9] O. Coma, C. Mascle, and P. Veron, "Geometric and form feature recognition tools applied to a design for assembly methodology," *CAD Comput. Aided Des.*, vol. 35, no. 13, pp. 1193–1210, 2003.
- [10] O. Coma, C. Mascle, and M. Balazinski, "Application of a fuzzy decision support system in a Design for Assembly methodology," *Int. J. Comput. Integr. Manuf.*, vol. 17, no. 1, pp. 83–94, Jan. 2004.
- [11] G. F. Dalgleish, G. E. M. Jared, and K. G. Swift, "Design for assembly: Influencing the design process," *J. Eng. Des.*, vol. 11, no. 1, pp. 17–29, 2000.
- [12] H. Mei and P. A. Robison, "Adding expert support to assembly-oriented computer aided design tools," *Proc. Inst. Mech. Eng. Part B J. Eng. Manuf.*, vol. 214, no. 1, pp. 81–88, 2000.
- [13] C. J. Barnes, G. E. M. Jared, and K. G. Swift, "A pragmatic approach to interactive assembly sequence evaluation," *Proc. Inst. Mech. Eng. Part B J. Eng. Manuf.*, vol. 217, no. 4, pp. 541–550, 2003.
- [14] S. J. Tate and G. E. M. Jared, "Detection of Symmetry and Primary Axes in Support of Proactive Design for Assembly," *System*, pp. 151–158.
- [15] C. . Rodriguez-Toro, G. Jared, and K. Swift, "Product - Development Complexity Metrics: a Framework for Proactive-Dfa Implementation," *Proc. Des. 2004, 8th Int. Des. Conf.*, pp. 483 – 490, 2004.
- [16] P. S. Collection, "Manufacturability in the Designers ' Sandpit," 2007.
- [17] a. Shtof, a. Agathos, Y. Gingold, a. Shamir, and D. Cohen-Or, "Geosemantic Snapping for Sketch-Based Modeling," *Comput. Graph. Forum*, vol. 32, no. 2pt2, pp. 245–253, May 2013.
- [18] M. Masry and H. Lipson, "A sketch-based interface for iterative design and analysis of 3D objects," *ACM SIGGRAPH 2007 courses - SIGGRAPH '07*, p. 31, 2007.
- [19] M. T. Cook and A. Agah, "A survey of sketch-based 3-D modeling techniques," *Interact. Comput.*, vol. 21, no. 3, pp. 201–211, Jul. 2009.
- [20] L. Olsen, F. F. Samavati, M. C. Sousa, and J. a. Jorge, "Sketch-based modeling: A survey," *Comput. Graph.*, vol. 33, no. 1, pp. 85–103, Feb. 2009.
- [21] R. Schmidt, B. Wyvill, M. C. Sousa, and J. a. Jorge, "ShapeShop: sketch-based solid modeling with BlobTrees," *ACM SIGGRAPH 2006 Courses - SIGGRAPH '06*, p. 14, 2006.
- [22] L. Eggli, C. Hsu, B. Bruederlin, G. Elber, B. D. Brüderlin, and G. Elber, "Inferring 3D models from freehand sketches and constraints," *Comput. Des.*, vol. 29, no. 2, pp. 101–112, 1997.
- [23] A. Shesh and B. Chen, "SMARTPAPER: An interactive

- and user friendly sketching system,” *Comput. Graph. Forum*, vol. 23, no. 3 SPEC. ISS., pp. 301–310, 2004.
- [24] H. Wang and L. Markosian, “Free-form Sketch,” vol. 1, 2007.
- [25] E. Schweikardt and M. D. Gross, “Digital clay: Deriving digital models from freehand sketches,” *Autom. Constr.*, vol. 9, no. 1, pp. 107–115, 2000.
- [26] A. Turner, D. Chapman, and A. Penn, “Sketching space,” *Comput. Graph.*, vol. 24, no. 6, pp. 869–879, 2000.
- [27] H. L. Shpitalni, “Optimization-based reconstruction of a 3D object from a single freehand line drawing,” *Computer-Aided Design*, vol. 28, no. 8. pp. 651–663, 1996.
- [28] M. Contero, F. Naya, J. Jorge, and J. Conesa, “CIGRO : A Minimal Instruction Set Calligraphic Interface for Sketch-Based Modeling,” *Work*, pp. 549–558.
- [29] J. Pereira, J. Jorge, V. Branco, and F. Ferreira, “Towards Calligraphic Interfaces: Sketching 3D Scenes with Gestures and Context Icons,” *Wscg*, 2000.
- [30] a. Forsberg, a. Forsberg, J. LaViola, J. LaViola, R. Zeleznik, and R. Zeleznik, “ErgoDesk: A framework for two-and three-dimensional interaction at the ActiveDesk,” *Proc. Second Int. Immersive Proj. Technol. Work.*, pp. 11–12, 1998.
- [31] O. Bimber, L. M. Encarnação, and A. Stork, “Multi-layered architecture for sketch-based interaction within virtual environments,” *Comput. Graph.*, vol. 24, no. 6, pp. 851–867, 2000.
- [32] T. Fleisch, G. Brunetti, P. Santos, and A. Stork, “Stroke-input methods for immersive styling environments,” *Proc. - Shape Model. Int. SMI 2004*, no. JULY, pp. 275–283, 2004.
- [33] H. Kaufmann and D. Schmalstieg, “Designing immersive virtual reality for geometry education,” *Proc. - IEEE Virtual Real.*, vol. 2006, p. 7, 2006.
- [34] G. Wesche, H.-P. Seidel, “FreeDrawer - A free-form sketching system on the responsive workbench,” *Proc. ACM Symp. Virtual Real. Softw. Technol. - VRST '01*, p. 167, 2001.
- [35] H. Perkunder, J. Israel, and M. Alexa, “Shape Modeling with Sketched Feature Lines in Immersive 3D Environments,” *Proc. Seventh Sketch-Based Interfaces Model. Symp.*, pp. 127–134, 2010.
- [36] B. Milosevic, F. Bertini, E. Farella, and S. Morigi, “A SmartPen for 3D interaction and sketch-based surface modeling,” *Int. J. Adv. Manuf. Technol.*, 2015.
- [37] G. Boothroyd, P. Dewhurst, W. Knight, *Product Design for Manufacture and Assembly*, Second edi. New York, New York, USA: Marcel Dekker, 2002.
- [38] T. Lim, M. Calis, J. Ritchie, J. Corney, R. Dewar, and Desmulliez, “A Haptic Assembly, Machining and Manufacturing System (HAMMS) Approach,” in *1st International Virtual Manufacturing Workshop*, 2006, p. 27.
- [39] R. Sung, J. Ritchie, T. Lim, and H. Medellin, “Assembly planning and motion study using virtual reality,” in *ASME/AFM World Conference on Innovative Virtual Reality 2009*, 2009, pp. 31–38.
- [40] H. G. Teklemariam and a. K. Das, “A case study of phantom omni force feedback device for virtual product design,” *Int. J. Interact. Des. Manuf.*, 2015.
- [41] Geomagic.com, “OpenHaptic Toolkit Overview.” [Online]. Available: <http://www.geomagic.com/en/products/open-haptics/overview/>. [Accessed: 11-May-2015].
- [42] V. Poyade, M, Kargas, M, Portela, “Haptic Plug-In for Unity.” Digital Design Studio (DDS), Glasgow School of Art, Glasgow, United Kingdom, 2014.
- [43] “Open Cascade.” [Online]. Available: <http://www.opencascade.com/>. [Accessed: 09-May-2016].
- [44] D. H. Douglas and T. K. Peucker, “Algorithms for the Reduction of the Number of Points Required to Represent a Digitized Line or its Caricature,” in *Classics in Cartography: Reflections on Influential Articles from Cartographica*, 2011, pp. 15–28.
- [45] M. J. Fonseca and J. a. Jorge, “Experimental evaluation of an on-line scribble recognizer,” *Pattern Recognit. Lett.*, vol. 22, no. 12, pp. 1311–1319, 2001.
- [46] G. Gonzalez, H. I. Medellin, T. Lim, J. M. Ritchie, and R. C. W. Sung, “3D object representation for physics simulation engines and its effect on virtual assembly tasks,” in *Proceedings of the ASME Design Engineering Technical Conference*, 2012, vol. 2, pp. 1449–1459.
- [47] C. Fletcher, J. Ritchie, T. Lim, and R. Sung, “The development of an integrated haptic VR machining environment for the automatic generation of process plans,” in *Computers in Industry*, 2013, vol. 64, pp. 1045–1060.