Proceedings of IDETC/CIE 2005
2005 ASME International Design Engineering Technical Conferences
& Computers and Information in Engineering Conference
Long Beach, California, USA, September 24-28, 2005

DETC2005-85462

INTEGRATING MULTIPLE ENGINEERING RESOURCES IN A VIRTUAL ENVIRONMENT FOR REVERSE ENGINEERING LEGACY MECHANICAL PARTS

Suraj R. Musuvathy, David E. Johnson, H. James de St. Germain, Elaine Cohen, Chimiao Xu, Richard F. Riesenfeld, Thomas C. Henderson
School of Computing
University of Utah
Salt Lake City, Utah 84112
Email: {srm, dejohnso, germain, cohen, xu, rfr, tch}@cs.utah.edu

ABSTRACT

Reverse engineering is a time-consuming and technically formidable process that is increasingly becoming an economic imperative due to replacement costs. The *Multiple Engineering Resources aGent Environment* (MERGE) system, introduced in this paper, is a new approach toward reverse engineering whose architecture and modules are driven specifically by the requirements of *legacy engineering*. Legacy engineering scenarios presume availability of multiple (possibly incomplete or inconsistent) sources of information, lack of digital descriptions of the parts, constrained time restrictions and need for significant domain knowledge expertise. The reverse engineering process must yield modern CAD models capable of driving state-of-the art CAM processes. The MERGE system aims at making the reverse engineering process more effective, using both intuitive interaction and visualization as key components, by enabling quick identification and resolution of inconsistencies among various resources in a unified environment. The MERGE system also aims at simplifying the reverse engineering process by integrating various *computational agents* to assist the reverse engineer in processing information and in creating the desired CAD models.

INTRODUCTION

Partly because of its complex, multifarious nature, reverse engineering is a time consuming, technically formidable process that is increasingly becoming an economic imperative because replacement costs of the entire original systems[1] are too high. Thus we are faced with a significant engineering challenge, since modern CAD/CAM systems are primarily conceived for *ab initio* design engineering, and give rather weak incidental support to reverse engineering. Our system specifically addresses the *legacy engineering* form of reverse engineering, i.e.,

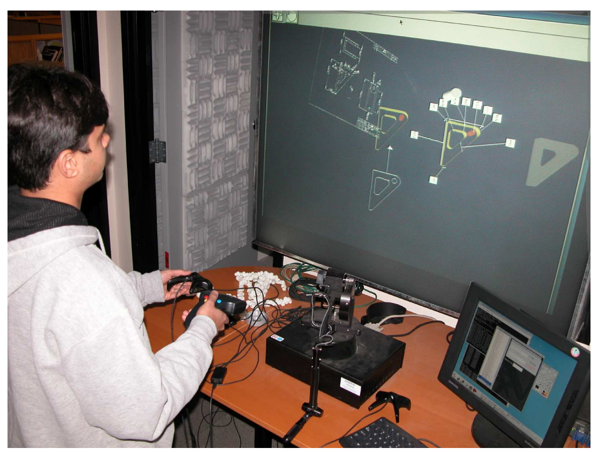

Figure 1. User interacting with the MERGE system.

[1]Boeing 707 aircrafts, developed in the early 1950s, are still in service to date even though production was stopped in 1978 [1].

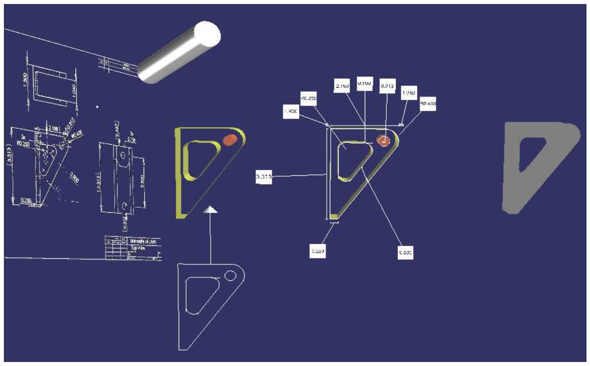

Figure 2. Multiple engineering resources in a unified environment. Starting from left, an engineering drawing, a CAD model structure visualization, a CAD model and a scan of a part.

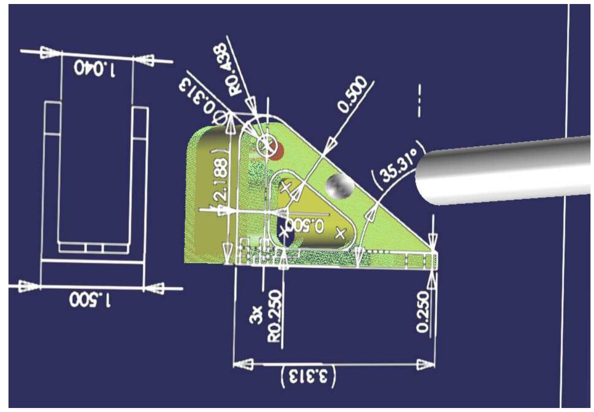

Figure 3. An engineering drawing directly compared with a reconstructed CAD model and a point cloud. The point cloud is registered with the CAD model. Also shown is the tracked wand represented by a gray cylinder.

the design of replacement parts or assemblies for systems that are no longer commercially supported nor properly documented because of their enduring service beyond any originally planned or projected scope specialized function.

Legacy engineering is influenced by the following set of challenges, namely, 1) proper documentation may not be available, 2) if available, the documentation is likely to be in a non-digital form, 3) available documentation typically has errors, inconsistencies, incompletenesses, and legibility problems, 4) a sample part may not reflect original dimensions due to undocumented modifications and wear, 5) the new target design system is likely to involve digital descriptions while the legacy part has none, 6) the original design may have failed or been inadequate for the current use, 7) the originally used materials may no longer be appropriate, or even available, 8) the original designers or vendors may not be available for information, 9) the original design rationale may be outdated and must be largely inferred from current accessible input sources, 10) multiple data sources acquired using a variety of sensing technologies may contain markedly different levels of error, and/or outright contradictions, 11) time is often critical in re-manufacturing, 12) significant domain knowledge and design engineering expertise is an essential ingredient, 13) much of the low-level processing can and should be automated, 14) the higher level engineering can be greatly facilitated with a conducive, modern computing and visualization environment, and 15) the legacy engineering process must yield modern CAD models capable of driving state-of-the-art CAM processes.

Traditionally, the process of reverse engineering has been performed via manual inspection of physical parts with or without the use of drawings. More recently, scanners have been used to speed up the process of inspection. The scanners generate point clouds that are used to reconstruct CAD models [2]. Feature based model reconstruction techniques have also recently emerged [3, 4]. However, a unified framework for the process of reverse engineering does not exist.

Goals

Our goal is to design a system that provides a unified environment in which a reverse engineer can visualize and process information simultaneously from multiple resources to obtain feature-based CAD models that more accurately capture the design intent of physical parts.

In particular, we aim at:

1. Providing intuitive inspection and visualization tools.
2. Facilitating intuitive transfer of data between resources.
3. Incorporating various *computational agents* in order to simplify the reverse engineering process.

The Multiple Engineering Resources aGent Environment (MERGE) System

We have developed a system called the Multiple Engineering Resources aGent Environment (MERGE) system that provides an interactive virtual environment to satisfy the requirements of legacy engineering. (See Fig. 1). MERGE provides a unified environment for comparing and manipulating original data sources, such as engineering drawings and laser-scanned physical artifacts, with derived engineering models, such as fitted surfaces and parametric model features. Agents in the system help analyze original data sources and present original data and derived models in a unified and accessible fashion. Figure 2 shows the various engineering resources present in the environment. It presents a snapshot of the system during the process of reverse engineering a legacy part, wherein a partial feature-based CAD model has been generated from an engineering drawing based on the front view of a part.

The MERGE system is built on top of the Alpha_1 advanced CAD research testbed that provides a feature-based modeling environment.[2] The virtual reality interface in the MERGE system includes 3D stereo display and 6DOF head/hand tracking. The Intersense IS900 system is used for tracking over a medium-room sized workspace. A rear-projected stereo-wall screen is used for 3D display. Alternatively, a head-mounted display can be used for 3D display and head tracking can be employed to achieve a more immersive effect. The user interacts with the system using a primary 6DOF tracked wand and a secondary 6DOF tracker. In Figure 1, the stereo-wall screen is shown with the user interacting with the system using a tracked wand and another tracker. The tracked wand is visually represented by a cylinder (Figures 2, 3).

Our contribution is a system that allows a user to readily compare, inspect, manipulate and modify heterogeneous sources of information of a legacy part. This is accomplished by transforming all the resources into higher-level forms that can be displayed, manipulated and queried for their attributes. This makes it more intuitive and less cumbersome than a traditional 2D menu-driven interface where one would associate names with objects and traverse through series of menus in order to modify the 3D objects. Figure 3 shows one such scenario from the MERGE environment where an engineering drawing visualization is placed over a fitted model for direct comparison of features.

In addition, we present a visualization of the data structure representing the derived feature-based CAD model and enable interaction with it to inspect and modify the features of the CAD model. We also present a simple algorithm that interactively maintains a comprehensible layout of labels (annotations) around a 3D model based on the relative orientation of the model with respect to the user.

BACKGROUND/RELATED WORK

Virtual environments have been explored for *ab initio* design [5], distributed design review [6] and analysis [7] of mechanical parts. The DVDS system proposed by Arangarasan and Gadh [5] discusses intuitive interaction techniques in an immersive environment including hand motions, gestures and voice commands for *ab initio* CAD design. The DDRIVE system of Daily et al. [6] utilizes a virtual environment for multi-party distributed design review. A virtual environment for interactive simulation and analysis has been developed by Yeh et al. [7]. Virtual environments have also been previously used for assembly simulation and planning as presented in the VADE [8] and VEGAS [9] systems.

There are a variety of commercial CAD-based reverse engineering software tools available such as Raindrop Geomagic®and Metrix Build!IT™. Although these tools provide significant functionality, they use 2D menu-driven interfaces and support automated reverse engineering from only point clouds. MERGE appears to be the first system that explores reverse engineering of mechanical parts using multiple heterogeneous resources in a virtual environment.

Previous work on feature-based CAD model structure visualization has been done by Bronsvoort et al. [10]. While they describe separate visualization of geometry and structure of feature-based models, our technique is an as-is visualization of the data structure underlying the geometry of the CAD model, thus showing the structure of the CAD model in addition to its geometry with correct spatial orientation relationships between features and entities. Our system also allows a user to interact directly with the visualization to modify the geometry of the target CAD model.

There has been extensive work on automated interpretation of engineering drawings [11–15]. Most of these algorithms require noise free conditions which is an unrealistic assumption in many cases. The proposed engineering drawing interpretation system in MERGE is designed to overcome such problems and is based on the system described by Henderson [16].

Prior literature on layout management of annotations exists for 2D [17–19] and 3D [20–23] information visualization environments. Of the algorithms for 3D visualization, the approach presented in Rose et al.'s work [20] is most directly related to ours. Their approach also aims at preventing the annotations from occluding the objects of interest, and uses connecting lines to attach the annotation with the corresponding objects. While our algorithm is not meant to replace these solutions, we provide a simple interactive solution and allow the user to select a subset of annotations to be displayed to avoid clutter. While most of the previous literature deals with annotations having static text, we allow users to interact with the annotations to modify the target CAD model.

REVERSE ENGINEERING A ROCKER MOUNT IN THE MERGE ENVIRONMENT

We present our system functionality through an example scenario of reverse engineering a rocker mount, a typical legacy part (See fig. 4). Resources available for reverse engineering the rocker mount include an exemplar physical part and documentation in the form of an engineering drawing. These resources are imported into the MERGE environment. Agents in the system, including *drawing analyzer agents, visualization manager agents and layout manager agents*, extract information from these resources and present them to the user in a coherent and comprehensible manner. An initial CAD model is fit from the drawing. The CAD model can be compared directly with

[2] Feature-based models are a natural fit in an interactive modeling environment as they allow a user to intuitively modify a model in order reflect design intent.

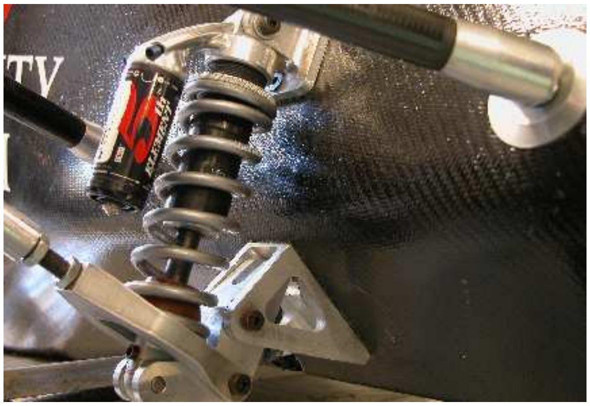

Figure 4. A rocker mount and its associated assembly.

available resources in order to interactively identify incorrect fits and fix them using information extracted from the resources.

Importing Multiple Resources into the MERGE Environment

The exemplar part is laser scanned to create a point cloud representation. The engineering drawing is scanned and imported as an image file. The image file is then rendered into the alpha channel of a transparent texture map, so that lines and annotations show as solid lines without occluding other objects in the MERGE environment. Figure 2 shows the drawing and the point cloud in the MERGE environment. Before being used in the virtual environment, drawing analyzer agents extract key data elements from the engineering drawing. In particular, drawing analyzer agents extract the individual 2D views (top, front, side) of the parts and also interpret the geometric dimensions of the part specified on the drawing.

Fitting an Initial Feature-based CAD Model

A feature-based CAD model of the part can be automatically generated from either the individual views extracted from the drawing or from the point cloud representation as described in the work of de St. Germain et al. [3, 4]. In our example, the features of the CAD model have been derived from the drawing. This model is imported into the virtual environment, which serves as a starting point from which the reverse engineer can create a more accurate CAD model of the exemplar part. In Figure 2, a partial feature-based CAD model derived from the front view of the part is shown.

Visualization of the Dependency Graph

The results of the process by which the features of the CAD model are derived are stored in a data structure called the "Dependency Graph" [24, 25]. The Dependency Graph is a directed-

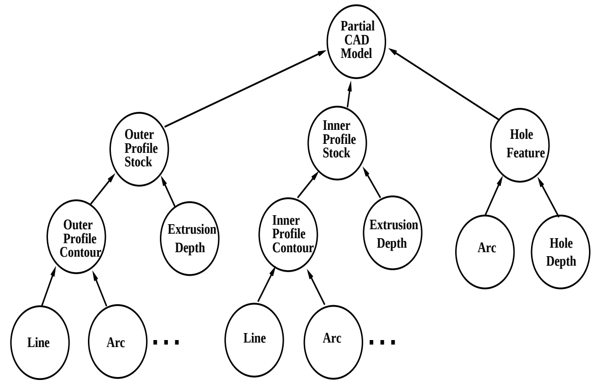

Figure 5. The Dependency Graph. For simplicity, only a single set of representative entities (lines, arcs) are shown for each feature.

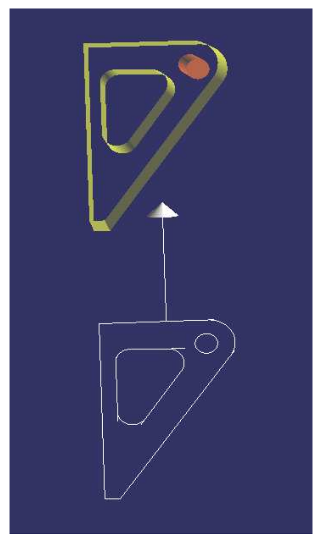

Figure 6. The Dependency Graph Visualization. The lower half of the figure shows the various line and arc entities comprising each feature. The upper half of the figure shows the extruded features comprising the partial CAD model.

acyclic graph whose nodes store information about the formative geometry of the part. Modification of any parameters of the features of a model is achieved by modifying the corresponding nodes of the dependency graph.

The dependency graph for the fitted partial CAD model is shown in Figure 5. As it can be seen, the dependency graph for a more complex part can easily get cluttered and difficult to understand. We present a more comprehensible visualization of the dependency graph to aid the reverse engineer in understanding how the final CAD model is generated and thus enabling him to decide how to modify given parameters in order to obtain the

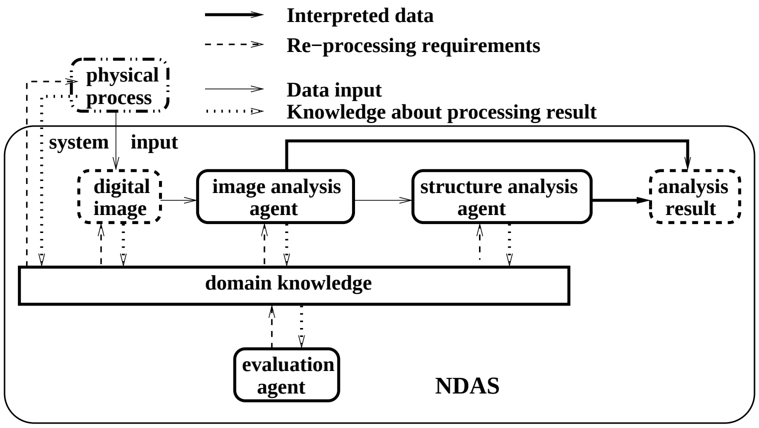

Figure 7. Overview of the NDAS system.

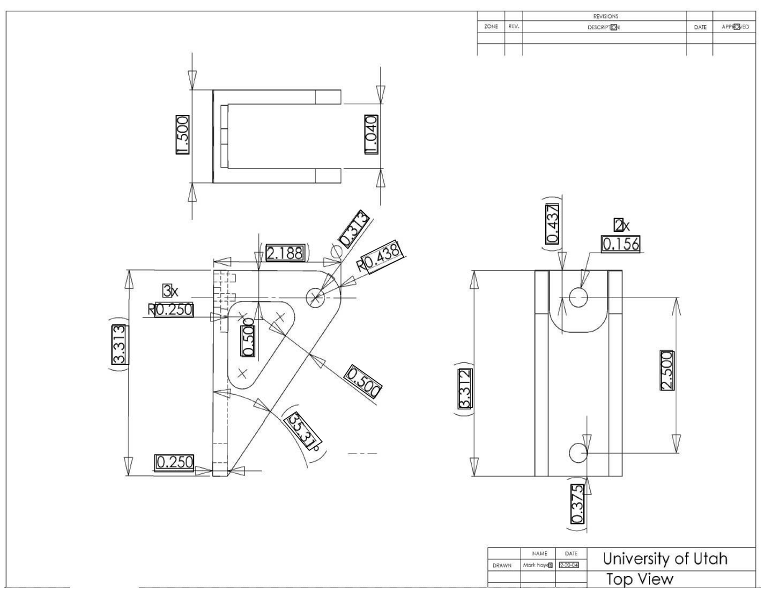

Figure 8. Results of dimension interpretation. The dimensions extracted are outlined with boxes.

desired model. We also allow the reverse engineer to interact directly with the visualization, thus enabling modification of the CAD model in an intuitive manner.

Nodes in the dependency graph corresponding to features and their entities (lines, arcs) are represented visually with their actual geometry and the actual spatial orientation relationships. Each level in the dependency graph conforms to a node in the dependency graph visualization. Figure 6 shows the dependency graph visualization at the point in the algorithm where a partial CAD model has been generated from the front view of the part. In this figure, the lower half(node) shows the entities defining the features of the partial CAD model. The upper half(node) shows the extruded hole feature and the extruded inner and outer profile stock features derived from the corresponding entities. All of the features and their entities are shown with the actual geometry and relative spatial orientations.

Visualization manager agents maintain comprehensible visualization of the dependency graph for each individual part and updates them as and when any of the features are modified by the reverse engineer. The MERGE environment allows a reverse engineer to work directly with the dependency graph data structure via its visualization. The reverse engineer can pick features or entities of interest using the wand to view information about their parameters or modify them. The parameters of selected entities are displayed around the 3D CAD model.

Interpretation of Dimensions on Drawings

The geometric dimensions are extracted from the drawing using a Non-Deterministic Agents System (NDAS) [11, 16]. As shown in Figure 7, NDAS is an automatic, domain-knowledge-guided system. It is an autonomous multi-agent system in itself. It consists of *image analysis agents*, *structure analysis agents and evaluation agents*. The image analysis agents interprets lines, arcs and characters (e.g., digits, letters, etc). The structure analysis agents identify dimensions that represent geometric information about the part. The evaluation agents calibrate, monitor and guide the analysis agents using explicit and persistent knowledge of the engineering drawing analysis process through stochastic optimization techniques. The physical process represents the process of digitizing the paper drawing. NDAS's evaluation agent can request the physical process to rescan the drawings with some requirements, such as higher resolution. In Figure 8, the dimensions extracted from the drawing of the rocker mount are outlined. These dimensions can be used to verify and correct the feature parameters derived from the model fitting process.

Manipulating Objects in 3D

In the virtual environment, the reverse engineer interacts with the resources using a 6DOF tracked wand in one hand and a 6DOF tracker in the other hand. The wand can be used to pick various objects such as the drawing, the CAD model or the point cloud which are then manipulated using the tracker in the other hand. The user can pick an object in the environment by pointing the wand at it at pressing one of the buttons on the wand.

Direct Comparison of Features

The reverse engineer can place the transparent drawing over the CAD model and/or the point cloud for direct comparison of features as shown in Figure 3. The reverse engineer can also compare the fitted features with the point cloud as shown in Figure 9. Currently, the CAD model and the point cloud are registered manually. Direct comparison enables quick identification of inconsistencies in geometry and relative orientation of features among different resources. In this example, the fitted hole is slightly undersized compared with point cloud representing the physical artifact.

Interactive View Folding

The individual views of the part from the drawing can be interactively folded over the fitted model and/or the point cloud to further aid the reverse engineer in associating and directly comparing features sketched in 2D and the features of the 3D model. The user can pick one of the views from the drawing using the wand and then request it to be registered with the 3D model using other buttons on the wand. The view is then animated by show-

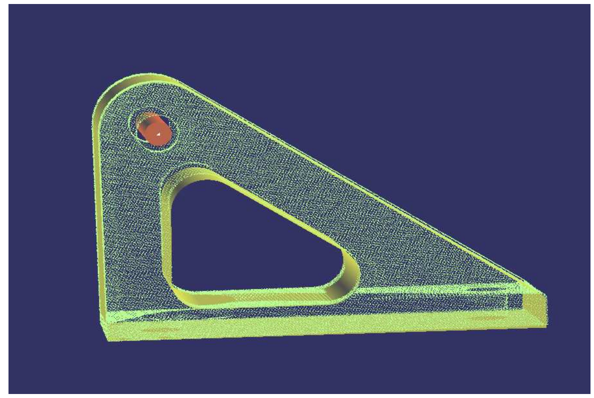

Figure 9. Close up of the CAD model registered with the point cloud for direct comparison of features.

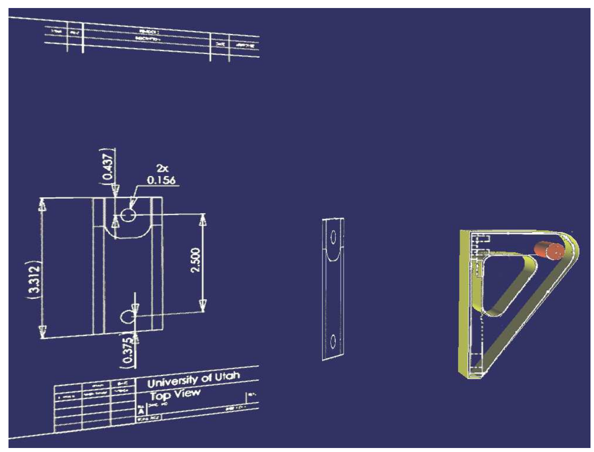

Figure 10. Side view of part from drawing being registered with 3D model.

ing it flying from the drawing and registering itself with the 3D model. Figure 10 shows the front view registered with the model and a snapshot of an animation of the side view registering itself with the model. Figure 11 shows the result of all the views registered with the model. In the current system, while the orientation of the views relative to each other are determined automatically, the orientations of the views relative to the model are determined manually.

Visual Representation of Feature Parameters via Labels

The current value of feature parameters are displayed using labels connected to the corresponding features in the 3D CAD model for quick verification and correction. Each label is connected to a symbolic line or a point on the 3D model that visu-

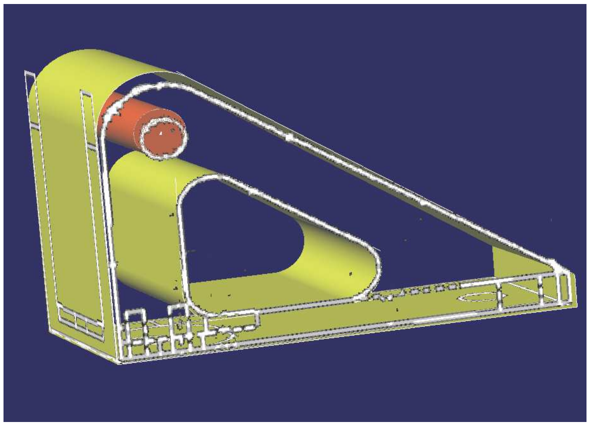

Figure 11. 2D views from the engineering drawing superimposed on the 3D model.

ally represents the feature parameters. These symbolic primitives help the reverse engineer in associating dimensions specified on the drawing with the features on the 3D model. Figure 12 shows a set of labels arranged around the CAD model. The symbolic lines are drawn in black.

In the virtual environment, the reverse engineer can manipulate the CAD model using a 6DOF tracker. Layout manager agents are responsible for maintaining a comprehensible layout for the labels for any orientation of the CAD model relative to the reverse engineer. The labels are rendered as planes on which the parameter value is texture-mapped and lines connecting the labels to their symbolic primitives are drawn. The layout manager interactively updates the position of the labels based on the orientation of the 3D CAD model relative to the reverse engineer.

In order to maintain a comprehensible view of all the labels, the layout of the labels must satisfy the following constraints:

1. Labels, ideally must be located as close to the feature or set of features they describe, while, at the same time, not occluding the 3D CAD model geometry.
2. Labels must not occlude each other.
3. Connecting lines should not cross each other.
4. Labels should not occlude any other label's connecting line.

In order to simplify the problem, we restrict the location of the labels to a plane whose normal is oriented along the line joining the eye-point of the camera and the center of the 3D model and is located between the user and the model. This plane is called the layout plane. Clearly, the layout plane changes with the user's orientation in the environment. In MERGE, an anchor is defined as the point on the symbolic primitives to which the label is connected. These anchors are projected onto the layout plane and sorted radially around the projection of the center of the model on the layout plane. The location of the label's centers are further restricted to a circle that has a radius greater than that of the bounding sphere of the model to ensure that the labels do not occlude the model. For each label in sorted order, an initial location is determined as the point on the circle that intersects the ray from the projected center to its anchor. Then, this location is checked to determine whether it lies close enough to any other previously placed label to occlude it. If it does, the label is moved anticlockwise on the circle so that it does not occlude the previously placed label. Figure 12 shows the results of the layout management algorithm for an arbitrary orientation of the part.

As a practical matter, for a large number of labels, there may not be enough room available to retain a comprehensible layout. The user can avoid this problem by requesting a small subset of labels to be displayed. This can be done by selecting certain features of interest via the dependency graph visualization or the CAD model using the wand. Accordingly, labels corresponding only to these selected features are displayed.

Modification of Feature Parameters by Direct Interaction with Drawings and CAD Models

Once the reverse engineer has identified an inconsistency in the parameter value of a feature (by direct comparison or by comparison of values on the label and the drawing), he can readily correct it using the dimension information extracted from the drawing. This can be achieved by first picking the corresponding dimension from the drawing and then transferring it to the appropriate feature on the 3D model, using the wand.

In our example, the diameter of the hole feature(as displayed on the corresponding label) does not match the dimension specified on the drawing. In order to fix this, the engineer first selects the corresponding dimension from the drawing as shown in Figure 13. This information is then transfered to the hole feature in the model by dropping the dimension value on the appropriate label as shown in Figure 14. The target feature to be modified can also be selected from the dependency graph visualization in a similar manner. By enabling direct interaction with drawings, fitted models and other derived resources, we have made the process of transferring data simpler and more intuitive.

SUMMARY AND CONCLUSIONS

Information about legacy parts is usually available from various sources, but, until now, the processing of each of these sources has been achieved via separate standalone solutions. The MERGE system simultaneously presents multiple heterogeneous resources to the reverse engineer. By placing this information in a unified environment, we assert that the problem of reverse engineering of legacy parts can be simplified. This paper gives an overview of the MERGE system and discusses its current capabilities by means of an example scenario, that of reverse engineering a typical legacy part.

The MERGE system provides tools for the visualization and augmentation of the following data sources: point clouds, engi-

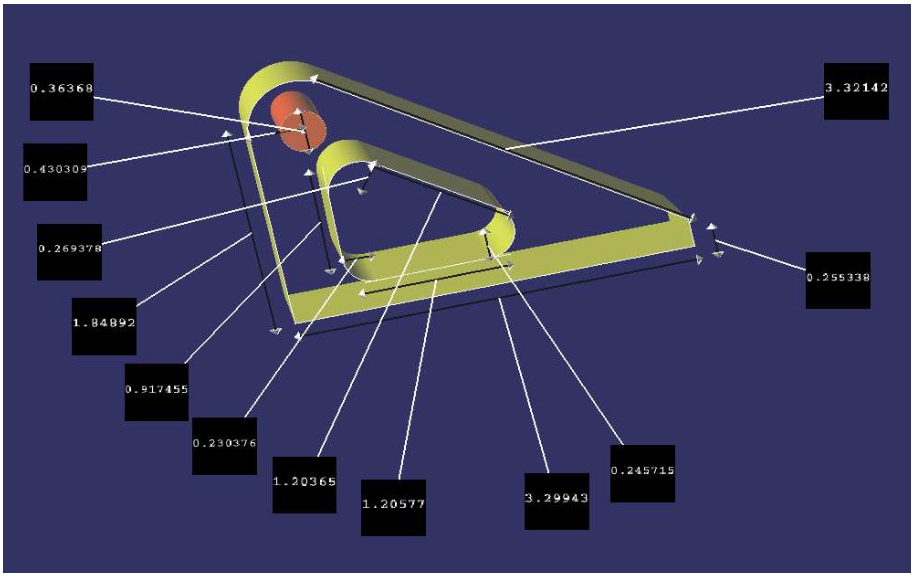

Figure 12. Layout of labels around the model in an arbitrary orientation. The labels are attached to symbolic lines that are drawn in black.

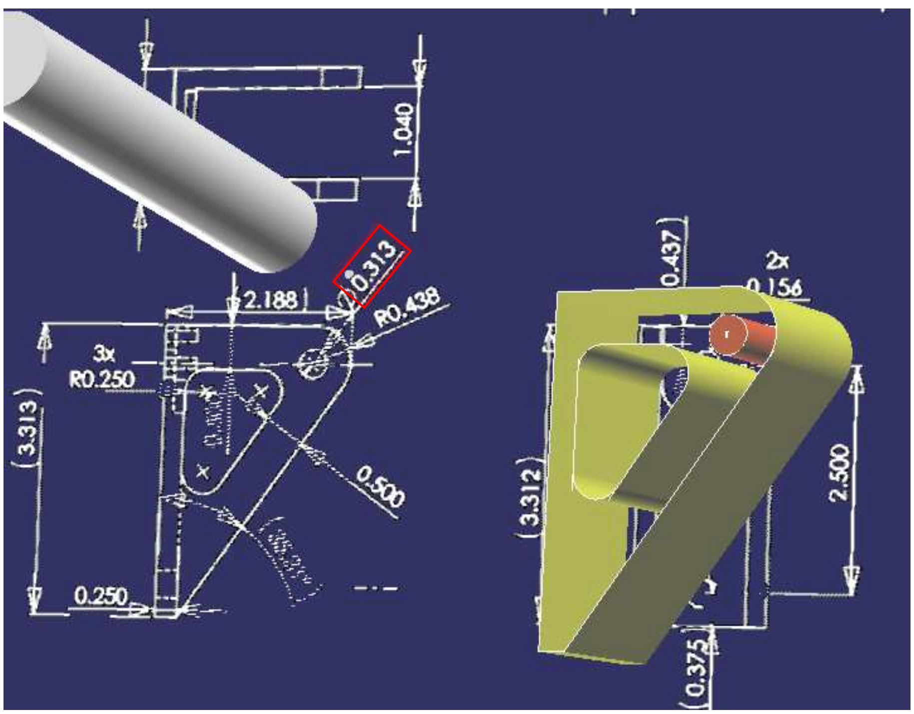

Figure 13. Picking dimension from drawing. A dimension is highlighted when the wand is pointed toward it.

neering drawings, dependency graph representations, and CAD models. Each set of information is fully visualized in a virtual 3D stereo environment, and further, augmented via the use of agents and visualization techniques to transform them from static objects into dynamic data sources which intuitively (visually) provide the user with the necessary information to understand, manipulate, and modify them such that the final CAD model is more accurate and achieved in less time than previously possible.

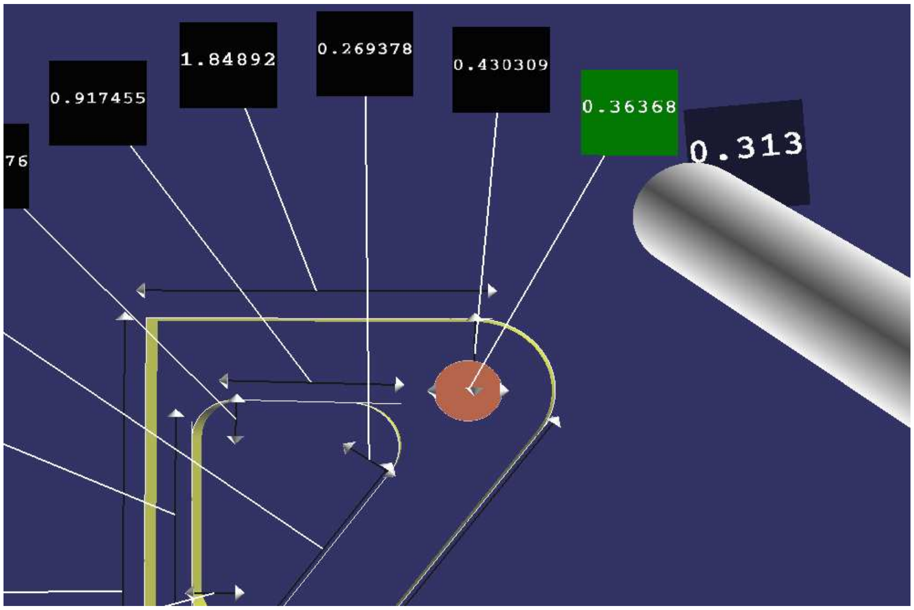

Figure 14. Selecting target feature parameter for modification via its label. The value picked from the drawing is displayed alongside the wand.

Key abilities include the display and modification of feature parameters in the regions of interest on the CAD model; projection of the dependency graph along side the 3D model; registration of multiple data sources such as drawings and scanned point clouds; and the folding of the engineering drawing such that each view is registered along side the appropriate view of the CAD model. Thus the user is allowed to directly focus on the reverse engineering process in the 3D virtual environment without the encumbrance of constantly updating the menus and text-boxes associated with the traditional WIMP interface.

We have constructed each element of the MERGE system specifically for the use in computer-aided reverse engineering and redesign. Although not all of the individual components of the system are significant technical advances in themselves, when utilized together in a unified environment, they combine to simplify, speed up, and improve the reverse engineering process. Given the appropriate tools, the reverse engineer can more quickly evaluate and build new models, identify and resolve inconsistencies between different data sources, and perform more effectively in legacy engineering scenarios. The findings and experiences reported herewith should be useful to projects with related goals, and to those considering developing adjunct reverse engineering components in their work.

FUTURE WORK

Ongoing work on the MERGE system includes providing more functionality, incorporating other interaction techniques and providing interactive analysis tools. Interaction techniques such as gestures and speech can make the interface more intuitive. Further work needs to be done on interface management for multiple parts, assemblies and large mechanical systems.

Currently, in the given scenario, engineering drawings of the legacy parts are available. If they are not available, the reverse engineering process has to be performed using only the point cloud. In addition, if the physical part is worn out or broken, we will require information about the rest of the assembly where the part fits in. Work needs to be done to address these issues.

ACKNOWLEDGMENT

This work was supported in part by DAAD19-01-1-0013. All opinions, findings, conclusions or recommendations expressed in this document are those of the authors and do not necessarily reflect the views of the sponsoring agencies.

REFERENCES

[1] Wikipedia, 2005. Boeing 707. URL `http://en.wikipedia.org/wiki/Boeing_707`.

[2] Hoppe, H., DeRose, T., Duchamp, T., McDonald, J., and Stuetzle, W., 1992. "Surface reconstruction from unorganized points". *Computer Graphics,* **26**(2), pp. 71–78.

[3] de St. Germain, H. J., Johnson, D. E., and Cohen, E., 2004. "Integrating freeform and feature-based fitting methods". In ASME Design Engineering Technical Conference. DETC2004-57298.

[4] de St. Germain, H. J., 2002. “Reverse engineering utilizing domain specific knowledge”. PhD thesis, University of Utah, Salt Lake City, Utah, December.

[5] Arangarasan, R., and Gadh, R., 2000. “Geometric modeling and collaborative design in a multi-modal multi-sensory virtual environment”. In DETC 2000.

[6] Daily, M., Howard, M., Jerald, J., Lee, C., Martin, K., McInnes, D., Tinker, P., and Smith, R., 2000. “Distributed design review in virtual environments”. In Proceedings of ACM Conference on Collaborative Virtual Environments, ACM Press, pp. 57–63.

[7] Yeh, T., and Vance, J., 1998. “Applying virtual reality techniques to sensitivity-based structural shape design”. *ASME Journal of Mechanical Design,* **120**(4), Dec, pp. 612–619.

[8] Jayaram, S., Jayaram, U., Wang, Y., Tirumali, H., Lyons, K., and Hart, P., 1999. “Vade: A virtual assembly design environment”. *IEEE Computer Graphics and Applications,* **19**(6), pp. 44–50.

[9] Johnson, T., 2000. “A General Virtual Environment for Part Assembly Method Evaluation”. MS Thesis, Depatmant of Mechanical Engineering, Iowa State University, Ames, Iowa.

[10] Bronsvoort, W. F., Bidarra, R., and Noort, A., 2002. “Feature model visualization”. *Computer Graphics Forum,* **29**(4), pp. 661–673.

[11] Swaminathan, L., 2003. “Agent-Based Engineering Drawing Analysis”. Master’s thesis, University of Utah, Salt Lake City, Utah, May.

[12] Tombre, K., and Dore, D., 1997. “Interpretation of engineering drawings”. In Handbook of Character Recognition and Document Image Analysi, pp. 457–484.

[13] Y.Yu, A.Samal, and S.C.Seth, 1997. “A system for recognizing a large class of engineering drawings”. *IEEE Transactions on PAMI,* **19**(8), pp. 868–890.

[14] Dori, D., 1992. “Dimensioning analysis toward automatic understanding of engineering drawings”. *Communications of the ACM,* **35**(10), October, pp. 92–103.

[15] Feng Su, Jiqiang Song, Chiew-Lan Tai and Shijie Cai, 2001. “Dimension Recognition and Geometry Reconstruction in Vectorization of Engineering Drawings”. In Computer Vision and Pattern Recognition, 2001 (CVPR 2001), Vol. 1, pp. 682–688.

[16] Henderson, T. C., 2003. Explicit and Persistent Knowledge in Engineering Drawing Analysis. Research Report UUCS-03-018, School of Computing, University of Utah, Salt Lake City, Utah.

[17] Edmondson, S., Christensen, J., Marks, J., and Shieber, S., 1997. “A general cartographic labeling algorithm”. *Cartographica,* **33**(4), pp. 13–23.

[18] Fekete, J.-D., and Plaisant, C., 1999. “Excentric labeling: dynamic neighborhood labeling for data visualization”. In CHI ’99: Proceedings of the SIGCHI conference on Human factors in computing systems, ACM Press, pp. 512–519.

[19] Agrawala, M., and Stolte, C., 2001. “Rendering effective route maps: improving usability through generalization”. In SIGGRAPH ’01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques, ACM Press, pp. 241–249.

[20] Rose, E., Breen, D., Ahlers, K. H., Crampton, C., Tuceryan, M., Whitaker, R., and Greer, D., 1995. “Annotating real-world objects using augmented reality”. *Computer graphics: developments in virtual environments*, pp. 357–370.

[21] Bell, B., Feiner, S., and Höllerer, T., 2001. “View management for virtual and augmented reality”. In UIST ’01: Proceedings of the 14th annual ACM symposium on User interface software and technology, ACM Press, pp. 101–110.

[22] Ritter, F., Sonnet, H., Hartmann, K., and Strothotte, T., 2003. “Illustrative shadows: integrating 3d and 2d information displays”. In IUI ’03: Proceedings of the 8th international conference on Intelligent user interfaces, ACM Press, pp. 166–173.

[23] Sonnet, H., Carpendale, S., and Strothotte, T., 2004. “Integrating expanding annotations with a 3d explosion probe”. In AVI ’04: Proceedings of the working conference on Advanced visual interfaces, ACM Press, pp. 63–70.

[24] University of Utah, Department of Computer Science, 1999. *Alpha1 User’s Manual Version 99.01*. Online HTML Document. URL: http://www.cs.utah.edu/gdc/projects/alpha1help/man/a1.html.

[25] Jacobs, T., and Cohen, E., 1998. “Aggregation and controlled interaction: Automated mechanisms for managing design complexity”. In Proceedings of ASME Design Engineering Technical Conference in Design Theory and Methodology.