Assembly Guidance in Augmented Reality Environments Using a Virtual Interactive Tool

M.L. Yuan*, S.K. Ong§ and A.Y.C. Nee*§
*Singapore-MIT Alliance
§Mechanical Engineering Department
10 Kent Ridge Crescent, Singapore 119260
National University of Singapore

***Abstract*— The application of augmented reality (AR) technology for assembly guidance is a novel approach in the traditional manufacturing domain. In this paper, we propose an AR approach for assembly guidance using a virtual interactive tool that is intuitive and easy to use. The virtual interactive tool, termed the Virtual Interaction Panel (VirIP), involves two tasks: the design of the VirIPs and the real-time tracking of an interaction pen using a Restricted Coulomb Energy (RCE) neural network. The VirIP includes virtual buttons, which have meaningful assembly information that can be activated by an interaction pen during the assembly process. A visual assembly tree structure (VATS) is used for information management and assembly instructions retrieval in this AR environment. VATS is a hierarchical tree structure that can be easily maintained via a visual interface. This paper describes a typical scenario for assembly guidance using VirIP and VATS. The main characteristic of the proposed AR system is the intuitive way in which an assembly operator can easily step through a pre-defined assembly plan/sequence without the need of any sensor schemes or markers attached on the assembly components.**

***Index Terms* Augmented reality, assembly guidance, assembly tree, real-time tracking**

I. INTRODUCTION

AUGMENTED reality (AR) attempts to integrate virtual information/models, such as computer graphics, text, sound and other modalities, into the physical environment so that the users can perceive that information as existing in real-time [1, 2]. One of the most promising applications of AR is in the traditional manufacturing assembly domain. A typical assembly process involves the grouping of individual parts that fit together to form a self-contained unit. With the development of advanced computer-aided tools and technologies, many assembly operations are automated. However, there are still a significant number of assembly operations that require assistance from human assemblers. Presently, the assembly information used to guide the human assemblers is often detached from the equipments and exists either as hard or soft copies. As a result, the operators will often have to alternate their attention between the actual assembly and the assembly instructions available as paper manuals, soft copies on external computers, or on websites. This divergence of attention will obviously consume much valuable time, especially when the instructions are not conveniently placed relative to the operators. This resultant workers fatigue may cause reduced productivity, errors, increased assembly time, repetitive motion and sometimes strain injuries. AR can provide useful assembly instructions in the real environment in the operator's field of view. This allows time saving, and convenient retrieving and sending of information vital to support the users' assembly tasks. The users can thus concentrate on the task at hand without having to physically move (change head or body position) to receive the next set of assembly instructions.

Engineers at Boeing demonstrated an AR-based system to aid workers in the assembly of airplanes [3, 4]. A test using AR resulted in a 20-50% improvement in wiring assembly performance even though the application was limited by the lack of the resolution accuracy and long-range head position tracking [5]. Molineros *et al.* [6, 11] addressed AR-related issues for the assembly domain where a multimedia augmentation guides humans in assembling an industrial object. Haniff *et al.* [7] used the AR technology for assembly training. Zauner *et al.* [9] developed a mixed-reality based step-by step furniture assembly system. Reiners *et al.* [8] described an AR demonstrator for the task of door-lock assembly in a car door. This demonstrator was developed as a practical and realistic application that teaches users how to assemble a door-lock into a car.

Some researchers have investigated the effectiveness of AR-assisted assembly methods for handling assembly tasks [10, 12, 13]. Several experiments were conducted using

M. L. Yuan is with Singapore-MIT Alliance, Singapore, 9 Engineering Drive 1, 117576. E-mail: smayml @ nus.edu.sg.

S.K. Ong is with Mechanical Engineering Department, the National University of Singapore, Singapore, 9 Engineering Drive 1,117576, E-mail: mpeongsk@nus.edu.sg.

A.Y.C. Nee is with Singapore-MIT Alliance and Mechanical Engineering Department, the National University of Singapore, Singapore, 9 Engineering Drive 1, E-mail: mpeneeyc@nus.edu.sg.

printed manuals and AR-assisted instructions to examine how effectively AR displays could be used to aid an operator in performing a manual assembly task. The results showed that the AR-assisted conditions were more effective for the assembly task than the paper-based instructions. In addition, operators made fewer errors under the AR-assisted conditions as compared to the paper-based instructional media. Moreover, AR proved to be more suitable for difficult tasks as compared to the paper manual, whereas for easier tasks, the use of a paper manual did not differ significantly from the AR-assisted conditions.

The main constraint in using AR for assembly guidance is to determine *when*, *what* and *where* to display the virtual information in the augmented world. It requires at least a partial understanding of the surrounding scene. This understanding requires sufficient sensor modality and interpretation that can communicate with the AR systems and the relevant changes in the state of the surrounding world with which the user interacts. In this paper, we developed an interactive tool called Virtual Interaction Panels (VirIPs), which can be used to directly acquire a relevant understanding of the surrounding scene from the human assemblers' perception. The main characteristic of the proposed AR system is the novel and intuitive way in which an assembly operator can step through a pre-defined assembly plan (assembly sequence) easily without the need of sensor schemes. The proposed approach uses a visual assembly tree structure (VATS) to manage the assembly information and retrieve the relevant instructions for the assembly operators in the AR environment. VATS is a hierarchical tree structure and can be maintained easily via a visual interface.

The paper is organized as follows. Section 2 describes the concept of the VirIP. Section 3 describes the VATS for assembly information management and retrieval in the AR environment. Section 4 presents the architecture of the AR-assisted assembly guidance system. System implementation will be discussed in Section 5. Section 6 presents two experiments to examine the performance of the proposed approach. Finally, conclusions and future works are presented in the last section.

II. VIRTUAL INTERACTION PANEL

The VirIP primarily comprises of two tasks: the design of the VirIPs and the real-time tracking of an interaction pen (referred to as an input device) using a RCE neural network. The input device is a pen-like object with a certain color distribution that can be used to trigger the buttons in the VirIPs. Such input devices occupy little space and can be utilized anywhere because they are common in the everyday environment.

A. Image Segmentation Using RCE Neural Network

The RCE neural network is known to be a specific design of the hyper-spherical classifier that can serve as a general adaptive pattern classification engine [14, 15]. In this research, it is used for image segmentation to track the input device (*pen*) in real-time. The RCE network consists of three layers of neuron cells, with a full set of connections between the first and second layers, and a partial set of connections between the second and third layers, as shown in Figure 1. Three cells in the input layer contain the color values: *R*, *G* and *B*, together representing a pixel in the image. The cells in the middle layer cells are called the prototype cells. Each prototype cell contains the color information, i.e., a learned color class in the training data, and each cell in the output layer corresponds to a different color class presented in the training data set.

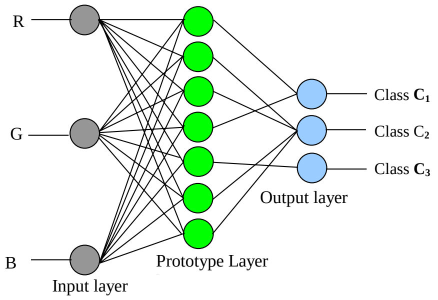


Figure 1 Architecture of a RCE neural network

The image segmentation using a RCE network includes two procedures, namely, the training and the image segmentation procedures.

1. Select a small area on the pen from the whole image to obtain the training data set *X*.
2. For an input color signal $x \in X$ belonging to a class C_k, if it does not trigger any response from the existing prototype cells, a new prototype cell *i* will be created and this new cell is then connected to the output cell *k* representing the class C_k.
3. For an input color signal $x \in X$ belonging to a class C_k, if it does trigger a response from an existing prototype cell *i* belonging to the same class C_k, the *t* counter of this existing prototype cell *i* is incremented by 1.
4. For an input color signal $x \in X$ belonging to a class C_k, if it triggers a response from an existing prototype cell *i* that does not belong to C_k, the radius of this prototype cell of the hyper-spherical influence field is reduced according to Equation (1).

$$d_i = \sqrt{\sum_{j=1}^{3} (\omega_{ij} - x_j)^2} \qquad (1)$$

After the training procedure, the segmentation procedure will be executed. The color images to be segmented are first acquired. Next, the set of color values of the images, referred to as *Y*, is input into the RCE network. In response to an input color signal $x \in Y$, a prototype cell uses a radial basis function to determine a trigger signal d_i as shown in Equation (1). If d_i is less than or equal to a cell threshold λ, the prototype cell will become active to trigger its associated color class *C*. Otherwise, the prototype will not respond to the input signal. When $d_i < \lambda$, a prototype cell is triggered, its parameter *t* will be incremented by 1.

A prototype cell can respond in a fast way. The prototype cell will simply output the value 1 to the connected cells in the output layer, that is:

$$p_i = \begin{cases} 1 & if\ d_i < \lambda_i \\ 0 & if\ d_i \geq \lambda_i \end{cases} \tag{2}$$

The pixels with the output value 1 represent the segmentation result. After the segmentation, the surrounding region of the segmented pen is computed automatically. Examples of the color image segmentation using the RCE network are shown in Figures 2(a) and (b).

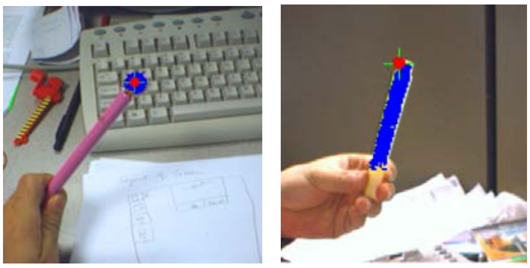

(a) (b)

Figure 2 Examples of RCE-based color image segmentation.

After the image segmentation, a feature point, called the interactive point, will be extracted as an input device. In the authors' AR system, the following two simple cases are used:

1. Use the center of the mass of the ball at the tip of the pen as the interactive point, as shown in Figure 2(a); and
2. Use the approximate tip of the pen as the interactive point, as shown in Figure 2(b).

B. *Tracking*

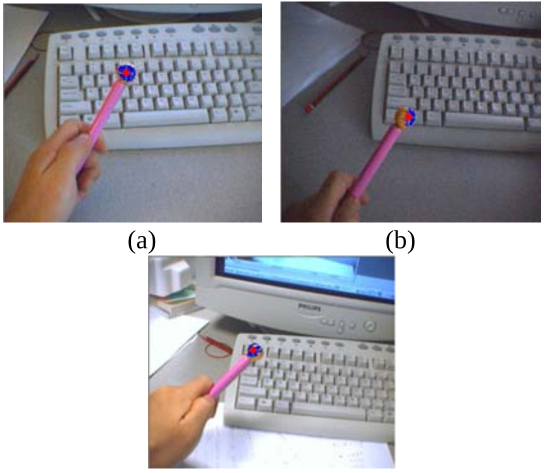
The above-mentioned segmentation method using a RCE network can be used to track the pen in real-time, as the RCE network is only performed on a small region surrounding the pen. A continuity of the position of the pen is assumed during tracking. The segmentation process using the RCE network will be automatically performed within a small region surrounding the pen in a live video. The average time elapsed for tracking the head of the pen during the augmentation process is 0.03147s, which can meet the real-time performance. Two issues addressed below show the robustness of the RCE-based tracking technique:

(1) *Movement of the pen*

Currently, the pen is moved at a normal speed. However, when there is a fast movement of the pen, the tracking information may be lost. Thus, the previous interaction point is preserved. The users can move the tip of the pen near the position of the previous interaction point to easily re-track the pen.

(2) *Lighting condition*

Under different lighting conditions, the pen can still be tracked if the lighting conditions do not change abruptly. For example, considering the three different lighting conditions as shown in Figures 3(a)-(c), part of the head of the pen can still be tracked. If lighting conditions change abruptly, the tracking information may be lost. In such case, the training data will be reset and the training procedure will be re-executed through loading other different training files from the systems. This re-initialization process is automatic.


(a) (b)

(c)

Figure 3 Examples of real-time tracking of a pen**.**

C. *Interaction Mechanism*

Users can interact with the authors' AR system through an intuitive interface via the tracked pen. This interface is called the VirIP. It is typically composed of some virtual buttons. Each virtual button represents some meaningful assembly information. These buttons can be activated using the interaction point of the tracked pen. In the authors' AR system, 2D interfaces are initially used. For example, the buttons on the left in Figure 4(a) and on the white paper in Figure 4(b) form the VirIPs, representing a set of assembly components. They are termed as assembly VirIPs. Such VirIPs can be used for assembly guidance without the need for object recognition.

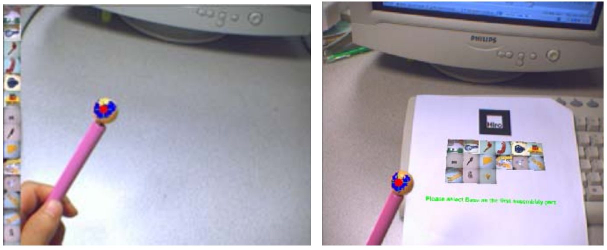

(a) (b)

Figure 4 Two examples of the VirIPs.

The VirIPs can be positioned in the following two ways: (1) in the screen coordinate system as shown in Figure

4(a), and (2) in the world coordinate system as shown in Figure 4(b). For the first case, the 2D image position of the tracked pen can be used to directly trigger the virtual buttons on the VirIPs. This kind of VirIPs will be used for monitor-based assembly guidance. For the second case, the VirIPs will be placed vertically to one of the *X*-, *Y*-, or *Z*-axes for easy calculation of the projection of the interaction points in the *XY*-, *YZ*-, and *XZ*- planes. In the AR assembly guidance system presented in this paper, the VirIPs are placed on the *XY*-plane in the world coordinate system, as shown in Figure 4(b). The 2D information is sufficient for the interactive operations because it is only necessary to determine whether the projection of the interaction point lies in the projection of the buttons of the VirIP in the world coordinate system. For the first case, it is intuitive to determine whether the tracked interaction point is on the virtual buttons in the screen coordinate system. For the second case, the projection of the buttons of the VirIP is designated as the sensing area. The projection on the *XY*-plane of the interaction point can be calculated easily from the pinhole camera model, assuming the intrinsic camera parameters are known.

A method to trigger the virtual button is required through determining that the interaction point is in the sensing area. A triggering mechanism has been designed as follows: the users place a pen in the real world, and use the above-mentioned interaction method to compute the interaction point in real-time. When the projection of the interaction point is in the sensing area continuously for 2~3 seconds, the virtual buttons are activated, which are otherwise in a sleep mode. Additional virtual control buttons for confirming whether to trigger the assembly buttons are marked as "OK" and "Cancel" buttons. The VirIPs with the "OK" and "Cancel" button are denoted as the confirmation VirIPs. Such a triggering mechanism for interactive control in an AR system can be easily performed through keeping the interactive point on the sensing area for a couple of seconds (e.g., 2 seconds).

III. ASSEMBLY AND INFORMATION MANAGEMENT

In this research, VirIPs are used for assembly guidance. One critical issue is the manageability of the assembly information and the retrieval of the assembly instructions/pre-defined assembly sequences from a database for assembly guidance.

As commonly known, assembly trees can be used to represent the assembly sequences of complex products [16]. In the authors' AR assembly system, an assembly tree is used to represent the sequences of a product assembly. A visual interface is used to build an assembly tree of a product. This assembly tree is denoted as a visual assembly tree structure (VATS). VATS represents the possible assembly sequences and is used for assembly information management and retrieval for the AR-assisted assembly guidance system. Every node in a VATS includes all the necessary assembly information, such as the geometric properties of an assembly part, sub-assembly relationships, etc.

It should be noted that a node in the VATS could represent an assembly part or a sub-assembly. If a node represents a sub-assembly, termed as a "*subassembly0*", then it is also a type of VATS, and denoted as VATS$_{0.}$ VATS$_0$ is a sub-tree of VATS, i.e. VATS$_0 \subset$VATS. If a sub-assembly exists, another node with the same assembly information but with the name suffixed with "*_Copy*", i.e., "*subassembly0_Copy*", will be added in the VATS automatically, as shown in Figure 5. The node "*subassembly0_Copy*" has no child nodes. It is not a VATS, but as an intermediate node representing the sub-tree VATS$_0$. This simplifies the representation of the assembly sequences of a product. The VATSs can be directly integrated into an AR system, or it can be implemented as an independent central control station on a remote computer to control the relevant assembly data flow during the whole process.

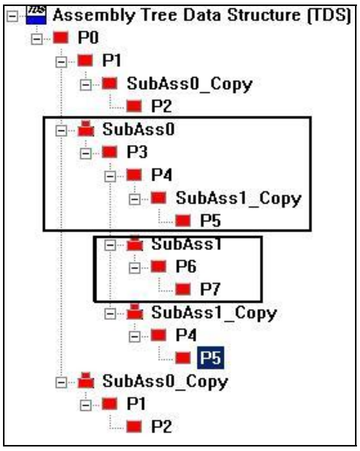


Figure 5 Examples of the VATS.

Assuming that the complete assembly comprises eight individual parts, i.e. $\{P_0,...,P_7\}$, there are two sub-assemblies, i.e., the nodes with the names of "*SubAss0*" and "*SubAss1*". These two nodes should be pre-assembled before they are being assembled with other parts, as shown in Figure 5. *SubAss0* is denoted as VATS$_0$ that includes another subassembly VATS$_{1.}$ VATS$_1$ represents another sub-assembly "*SubAss1*". Each part name or sub-assembly name of the nodes in the VATS corresponds to the name on the buttons in the assembly VirIP. Let P_0 be the base of the whole assembly. If the "*Base*" button in the VirIP is activated, two children "P_1" and "*SubAss0*" are searched for from the VATS. The corresponding instructions will be obtained from the database according to their names and the related node. The operator can pre-select one of the nodes using the VirIP. The instruction will be displayed in the AR environment and the operator assembles the selected real part. If P_1 is selected, the instruction between P_0 and P_1 will be displayed in the AR environment.

Otherwise, the information between the P_0 and VATS_0 will be displayed. If the part "*SubAss0_Copy*" is selected, the AR system will go back to the sibling node of its parent, namely, "*SubAss0*", which has it its own assembly tree, i.e., VATS_0. The nodes in VATS_0 will first be pre-assembled before being assembled with P_0. The sub-assembly procedure is the same as the main assembly. In this example, there is another sub-assembly, namely, VATS_1 in VATS_0. The same operations should be repeated to assemble VAST_1 before assembling all nodes in VAST_0. After VATS_1 has been assembled, P_5 is selected to complete the nodes in VAST_0. When VAST_0 has been assembled, the next part of the instructions relating to "*SubAss0_Copy*" will be automatically retrieved from the instruction database according to its child node P_2 in the VATS. Next, P_2 is selected via the assembly VirIP. The corresponding real part is then assembled. This procedure will be repeated until all parts are successfully assembled.

Based on the operator's experience, different assembly information can be retrieved to guide the assembly operation. It should be noted that when a node is retrieved from a VATS, the assembly instruction would be searched for from an existing assembly instruction database according to the names of the current node and its parent's node (it can be extended to retrieve more information from the nodes, such as geometric constraint types, etc.). Simple image-based instructions indicating the assembly operations can be stored in the assembly instruction database, as shown in Figure 6.

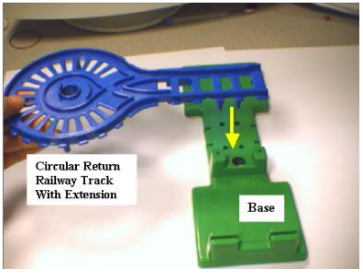


Figure 6 Example of assembly instructions.

IV. SYSTEM ARCHITECTURE

In an AR-assisted assembly environment, human involvements will greatly help the assembly guidance. At the same time, the assembly instructions can be retrieved from the instruction database according to the node information in the VATS. The instructions that have been retrieved guide the human operators through each assembly step required. The architecture of the AR-assisted assembly guidance system using VirIP and VATS is shown in Figure 7. It includes the human operator in the loop. The human operator uses the assembly VirIP to select the part or sub-assembly to be assembled. The VATS and instruction database provide information for display in the augmentation system. For example, information can be provided on the types of operations to be executed (insert, snap, pressure fit, etc.). Besides, the human operators use the confirmation VirIP to confirm if an assembly operation has been executed and thus retrieving the next piece of assembly instruction. There is a critical assembly information data flow between the assembly tree, the human operator and the AR system, described as follows:

1. *Information visualization from the VATS*

   If there is only one part related to the current node from the VATS, the AR system will prompt the user to select that part via the assembly VirIP. According to the related node and its parent node, the relevant instructions will be retrieved from the instruction database and displayed in the AR environment. If there are several child nodes related to the current node in the VATS, the AR system will display the corresponding instructions retrieved from the database in the AR environment and prompt the user to select one part. The relevant instructions will be displayed upon the human operator selecting a part via the assembly VirIP.

2. *To the VATS from the Human*

   While the user manually carries out the assembly steps, he/she uses the confirmation VirIP to confirm that the operations have been executed. After the confirmation, the system will retrieve the next part or sub-assembly to be assembled from the VATS. The system then provides the user with some instructions, such as "*Please select (P1), then press "OK" button!*". Next, the user selects the part via the assembly VirIP and the relevant instruction will be retrieved from the instruction database and displayed in the system.

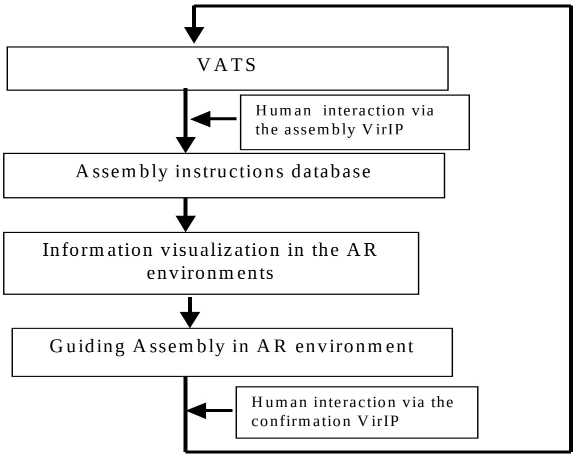


Figure 7 The Architecture of AR-assisted Assembly Guidance System.

From the above discussion, it can be concluded that the proposed approach enables a convenient flow of information between the assembly operator and the VATS. Two related issues are summarized as follows:

1. The VATS can be implemented as an independent system in a remote computer to act as a central control station to control the assembly information data flow during the whole assembly process.

2. The VirIPs allow the AR system to automatically monitor and record difficulties, if any, encountered during the assembly operations performed on real parts for future assembly evaluations.

V. IMPLEMENTION

The video sequences were captured using an IEEE 1394 FireFly camera. The basic system set-up consists of a camera, an i-glass display, a helmet and a pen. Both monitor- and head-mounted display (HMD)-based assembly guidance methods were developed according to the positions of the VirIPs in the screen coordinate system and the world coordinate system respectively. It can be said that a monitor-based method would be more practical and realistic for the end-users. Both methods have their advantages and disadvantages. A HMD with its lower resolution allows the user to interact more freely and gives him/her the possibility to use his/her hands together with the real tools during assembly. In contrast, the monitor-based method, although having a higher resolution, hinders the degree of interaction of the user, because the user is required to focus his/her attention on the monitor to see the instructions. However, the choice of the device solely depends on the user's requirements.

A VATS was initially built from the possible pre-defined assembly sequences prior to any actual assembly operations in the AR environment. Using the pre-defined VATS and VirIPs, assembly instructions can be retrieved to guide the human operators. The assembly guidance method is defined as ***Assembly*** (VATS) and described as follows:

1. First, select an assembly part as the "*Base*" and search its child nodes in VATS. Next, retrieve the relevant assembly instructions for the "*Base*" from the instruction database. There may be several individual pieces of assembly instructions, which means that there may be several child nodes in the VATS and their corresponding parts or sub-assemblies can be assembled with the "*Base*" at the same precedence level. Next, the user selects one part or sub-assembly via the assembly VirIP. If the selected object is a part, the operator can then select the real assembly component in front of him/her, for which the assembly instructions will be retrieved from the instruction database according to the current node and its parent's node. If the selected object is a sub-assembly, the sub-tree VATS_0 of this sub-assembly should first be pre-assembled together. The system will prompt the user to pre-assemble this sub-assembly. This procedure is similar to the overall assembly, and hence ***Assembly*** (VATS_0) would be executed to guide the sub-assembly process.
2. After the assembly operation has been executed, the system would need to be updated with this information and the status of the assembly process so that the next possible assembly information can be retrieved from the VATS and the instruction database to continue the assembly process. Completion of an assembly operation and updating of the status of the assembly process recognition can be achieved using the confirmation VirIP. After the confirmation, the system will search the child nodes of the current node in the VATS, followed by the instructions from the instruction database for the next assembly operation to be executed.
3. Repeat **Step 1** and **Step 2** until all the assembly components have been assembled. During the whole process, if the operator does not wish to follow the assembly operation according to the information provided by the VATS, he/she can click an assembly part button in the assembly VirIP for a new instruction to be displayed if it exists.

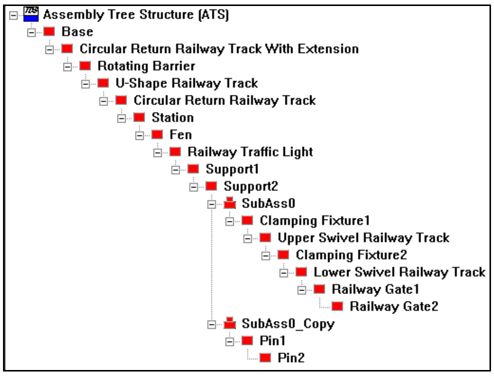
Following the above steps, the assembly operations can be easily carried out in an AR environment solely based on human experience without the need for object recognition and markers attached on the assembly components.

VI. EXPERIMENTAL RUN

Two experiments were conducted to examine the performance of the AR-assisted assembly guidance system using the VirIPs. A toy "Fun Train" was assembled using the proposed AR-assisted assembly guidance approach. Any engineering products could be assembled using this same approach. The difference between the two experiments is that, the first experiment is based on the monitor-based assembly guidance method, and the second is the HMD-based method. The assembly VirIP represents the components of the "Fun Train", and the confirmation VirIP is used to confirm the assembly operations. The reason for using the confirmation VirIP with "OK" and "Cancel" buttons is to confirm that an assembly operation has been completed and the next assembly step can be carried out. Upon confirmation, the child nodes of the current node will be searched for from the VATS and the relevant assembly instruction will be retrieved from the instruction database. Figure 8 shows a part of the VATS, which represents a sequence of the "Fun Train" used in the two experiments. In the experiments, 19 steps are required for guiding the assembly process. The pen with a ball-tip will be tracked in real-time using the RCE network during the assembly process.

A. *Experiment 1 – Monitor-based Assembly Guidance*

In the first experiment, the VirIPs are directly placed in the screen coordinate system for guiding the assembly operations.

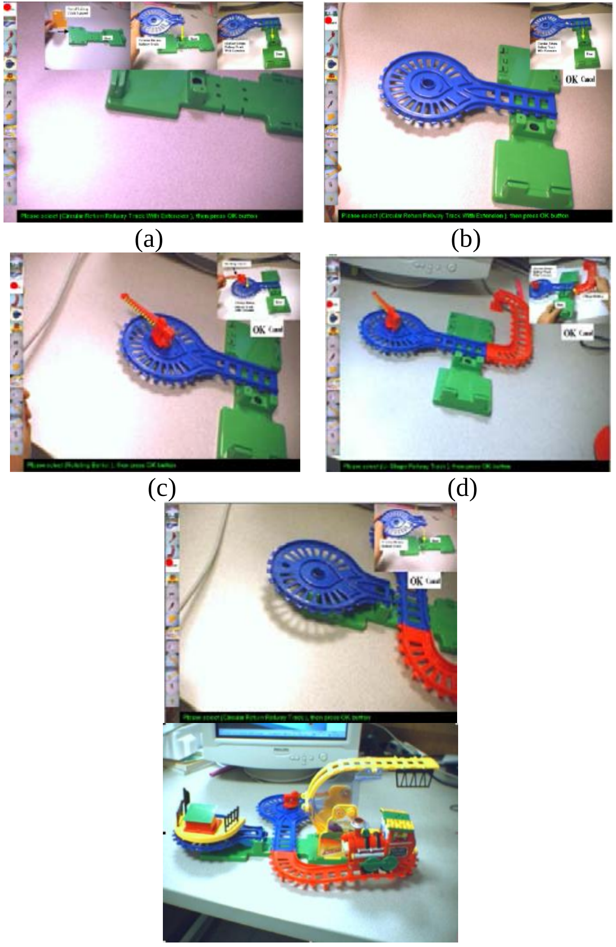


- Assembly Tree Structure (ATS)
  - Base
    - Circular Return Railway Track With Extension
      - Rotating Barrier
        - U-Shape Railway Track
          - Circular Return Railway Track
            - Station
              - Fen
                - Railway Traffic Light
                  - Support1
                    - Support2
                      - SubAss0
                        - Clamping Fixture1
                          - Upper Swivel Railway Track
                            - Clamping Fixture2
                              - Lower Swivel Railway Track
                                - Railway Gate1
                                  - Railway Gate2
                      - SubAss0_Copy
                        - Pin1
                          - Pin2

Figure 8 The VATS of "Fun Train".

The first step is to use the tracked interaction point to specify the base of the whole "Fun Train" assembly, as shown in Figure 9(a). When the "*Base*" has been specified, the user places the real "Base" in front of him/her. Next, the system will retrieve the assembly information related to the child nodes of the "*Base*" in the VATS from the instruction database. In this case, there are three pieces of instructions that have been retrieved for the "*Base*". The user selects the next assembly components via the assembly VirIP. The part "*Circular Return Railway Track with Extension (CRRTE)*" is then selected and the assembly information of the relationship between the "*Base*" and the "*CRRTE*" will be displayed at the top right corner of the screen, as shown in Figure 9(b). Below this information, the confirmation VirIP is displayed. The user can assemble the two selected parts together. After this operation, the completion of the operation is confirmed by clicking the "OK" button in the confirmation VirIP. After this operation, the system will provide the instruction "*Please select Rotating Barrier (RB), then press "OK" button!*" at the bottom of the screen. The name "*RB*" is retrieved from the VATS, as shown in Figure 9. The relevant instruction retrieved from the instruction database will next be displayed, as shown in Figure 9(c). Next, the system will provide the user with the instruction from the VATS, i.e., "*Please select U-shaped Railway Track (URT), then press "OK" button!*". The process is same as the operation of the "*RB*", as shown in Figure 9(d), after which the "*Circular Return Railway Track (CRRT)*" is searched for from the VATS and selected by the user directly to be assembled. This part is related to the "*Base*" and hence the instructions between the "*CRRT*" and the "*Base*" will be retrieved and displayed, as shown in Figure 9(e). On the completion of this operation, the user would press the "OK" button to confirm that the operation has been completed. There is a sub-assembly in this experiment, i.e., "Support Platform". After the "Support2" has been assembled, the system will prompt the users to assemble this sub-assembly, i.e., the sub-tree VATS$_0$. The process is similar to the whole assembly. The above process is repeated until the whole assembly process has been completed, as shown in Figure 9(f).

From this experiment, it can be observed that the monitor-based interactive assembly guidance method is a very useful and serves as an easy guide for assembly operations, especially using wearable computers.



(a) (b)

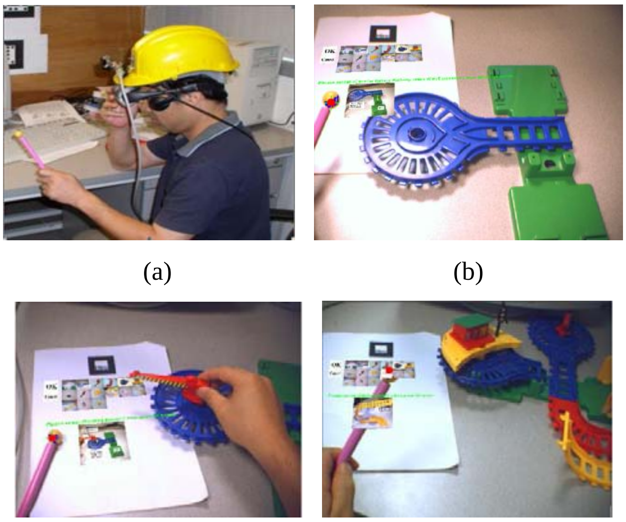
(c) (d)

(e) (f)

Figure 9 Examples of monitor-based "Fun Train" assembly.

B. *Experiment 2 – HMD-based Assembly Guidance*

The second experiment is the HMD -based experiment, as shown in Figure 10(a). The VirIPs will be positioned in the world coordinate system. A white paper on which there is a marker will be used to display the assembly instructions, as can be seen in Figures 10(b)-(d). The paper can be moved freely or placed at a convenient place near the assembly site to guide the assembly operations. This paper can be regarded as a "*virtual assembly instruction book*". The complete assembly process is the same as the first experiment. In particular, this method is useful for assembling small complex products as the white paper and the assembly site can be shown in the same image. The HMD-based method allows time saving, and easy retrieving and sending of information to help the user with the assembly task.



(a) (b)

(c) (d)

Figure 10 Examples of monitor-based “Fun Train” assembly.

As mentioned in Section 5, the monitor-based and HMD-based assembly guidance methods have their own advantages and disadvantages. In addition, these proposed guidance methods can be used for assembly sequence planning. Assembly sequence planning using the interactive tool in an AR environment is part of the authors’ future work. Raghavan *et al.* [11] have reported the use of AR for interactive assembly sequence evaluation. The main drawback in their method is that markers have to be placed on every assembly components for recognition purpose. It is cumbersome to place markers on every assembly component, and the recognition results will cause confusion in real-time performance. Hence, the interactive tool proposed in this research is more suitable for interactive assembly sequence generation and evaluation, without the need for object recognition and markers attached on the components. The VirIPs can be used for assembly sequence planning and evaluation in the following three ways:

1. The VirIPs can be used for object recognition in AR-assisted assembly planning and evaluation;
2. The VirIPs can be used to interactively control an online assembly process and the assembly information; and
3. The VirIPs can be used to interactively control a mixed prototyping model used for assembly sequence planning and evaluation.

VII. CONCLUSION AND FUTURE WORK

In this paper, a simple and user-friendly method for assembly guidance using VirIP and VATS in an AR environment has been proposed and implemented. The VirIP can be interactively controlled using an input-tracking device, which is as simple as a common pen, hand or any pen-like object. VATS is used for information management and retrieval of assembly instructions. Two experiments were conducted to show the efficiency of the proposed approach for assembly guidance. Future work will be focused on assembly sequence planning and interactive evaluation without any markers attached on assembly parts. This process will be incorporated with the assembly guidance process so that the assembly sequence can be generated in real-time during assembly guidance.

REFERENCES

[1] Azuma, R.T., 1997, “A Survey of Augmented Reality”, *Presence: Teleoperators and Virtual Environments*, pp 355-385.

[2] Azuma, R., Baillot, Y., Behringer, R., Feiner, S., Julier, S., MacIntyre, B., 2001, “Recent Advances in Augmented Reality”, *IEEE Computer Graphics and Applications*, 21(6), pp 34-47.

[3] Caudell, T.P., Mizell, D.W., 1992, “Augmented Reality: An Application of Heads-Up Display Technology to Manual Manufacturing Processes”, *Proceedings of Hawaii International Conference on System Sciences*, pp 659-669.

[4] Sims, D., 1994, “New Realities in Aircraft Design and Manufacture”, *IEEE Computer Graphics and Applications*, 14(2), p 91.

[5] “Wiring the Jet Set”, http://www.wired.com/wired/5.10/wiring.html, (2004-11-15)

[6] Molineros, J.M., 2002, “Computer vision and augment reality for guiding assembly”, *PhD Dissertation*, The Pennsylvania State University.

[7] Haniff, D.J., Boud, A., Baber, C., 1999, “Assembly training with augmented reality and virtual reality”, *International Conference on Human-Computer Interaction*, 2, pp 35-36.

[8] Reiners, D., Didier, S., Gudrun, K., Stefan, M., 1998, “Augmented reality for construction tasks: doorlock assembly, Augmented Reality. Placing Artificial Objects in Real Sciences*”, Proceedings of IWAR'98, 1998*, pp 31-46.

[9] Zauner, J., Haller, M., Brandl, A., 2003, “Authoring of a Mixed Reality Assembly Instructor for Hierarchical Structures”, *Proceedings of ISMAR’03*, October, Tokyo, pp 237-246.

[10] Tang, A., Owen, C., Biocca, F., Mou, W.M., 2003, “Comparative Effectiveness of Augmented Reality in Object Assembly”, *CHI 003*, Florida, USA, pp 73-80.

[11] Raghavan, V., Molineros, J., Sharma, R., 1999, “Interactive evaluation of assembly sequences using augmented reality”, *IEEE Transaction on Robotics and Automation*, 15(3), pp 435-449.

[12] Wiedenmaier S., Oehme, O., Schmidt, L., Luczak, H., 2003, “Augmented Reality (AR) for Assembly Processes Design and Experimental Evaluation”, *International Journal of Human-Computer Interaction*, 16, pp497-514.

[13] Baird, K.M., Barfield, W., 1999, “Evaluating the effectiveness of augmented reality displays for a manual assembly task”, *Virtual Reality*, 4(4), pp 250-259.

[14] Yin, X.M., Xie, M., 2001, “Hand Image Segmentation Using Color and RCE Neural Network”, *Journal of Robotics and Autonomous Systems*, 34(4), pp 235-250.

[15] Reilly, D.L., Cooper, L.N., Elbaum, C., 1982, “A Neural Network Mode for Category Learning”, *Biological Cybernetics*, 45, pp 35-41.

[16] Yuan, X.B., 2002, “An Interactive Approach of Assembly Planning”, *IEEE Transaction on Systems, Man, Cybernetics - Part A: Systems and Humans*, 32(4), pp 522-526.

M.L. Yuan received his BS degree in Mathematics from Hangzhou Normal College in 1992 and his PhD degree in Mechanical Engineering from Huazhong University of Science and Technology (HUST), China in 1997. During 1997-1999, he worked in HUST as a lecturer. During 1999-2000, he worked as a senior software engineer in Asahi Hi-tech Co, Ltd, Japan. From 2000 to present, he is a research fellow in Singapore-MIT Alliance, Singapore. His research interests include computer vision, spatial layout, geometric reasoning, augmented reality and its applications in manufacturing.

S.K. Ong has been an assistant professor in the Manufacturing Division, Department of Mechanical Engineering, at the National University of Singapore since 1996. Her research interests are intelligent and distributed manufacturing systems, computer-aided set-up planning, life cycle engineering, environment impact assessment, and more recently, virtual and augmented reality applications in manufacturing. She has attended and made presentations in many international conferences, including two general assemblies of the CIRP. She has published two books, and over 78 international refereed journals and conference papers. She is the recipient of the Norman Dudley Award in 2002 for the best paper published in the International Journal of Production Research. She has served on several international conference committees, invited reviewer of many journals in manufacturing and is an assistant editor of a book series on Manufacturing Systems and Technology published by World Scientific.

A.Y.C. Nee is a professor of manufacturing engineering, Department of Mechanical Engineering, National University of Singapore since 1989. He received his PhD and DEng from UMIST in 1973 and 2002 respectively. His research interest is in computer applications to tool, die, fixture design and planning, intelligent and distributed manufacturing systems, virtual and augmented reality applications in manufacturing. He has published 5 books and over 450 papers in refereed journals and conference presentations. He currently held regional editorship, department editorship, associate editorship and member of editorial board of 15 international journals in the field of manufacturing engineering. He is an active member of CIRP and an elected Fellow of the Society of Manufacturing Engineers, both in 1990.