Institute for Software Research
University of California, Irvine

Just-In-Time AR-Based Learning in the Advanced Manufacturing Context

Bryce Tham
University of California, Irvine
btham@uci.edu

Walt Scacchi
University of California, Irvine
wscacchi@ics.uci.edu

June 2017
ISR Technical Report # UCI-ISR-17-2

Institute for Software Research
ICS2 221
University of California, Irvine
Irvine, CA 92697-3455
isr.uci.edu

Just-In-Time AR-Based Learning in the Advanced Manufacturing Context

Bryce Tham and Walt Scacchi

Donald Bren School of Information & Computer Sciences, University of California, Irvine
UCI Institute for Software Research (ISR)
California Institute for Telecommunications and Information Technology (Calit2)

June 2017

ISR Technical Report #UCI-ISR-17-2

Abstract

The use of augmented reality (AR) applications has the potential to greatly reduce errors and increase productivity in the manufacturing space. The application I developed, in particular, focuses on the problem of how to provide effective training, which refers to issues that are related to when workers are required to learn new or updated skills, using AR. In a technique called just-in-time augmented learning, these "smart" workers can use AR to access valuable training resources, which play an important role in informing them of their tasks and responsibilities. This can be done without the need to refer to a secondary laptop computer or tablet, because the resources are loaded directly onto the workers' AR device, integrating the training process into the workflow and reducing diversion. By using various game-based systems and marker-based AR libraries, I designed and developed a prototype for a head-mounted AR application that demonstrates the feasibility of just-in-time augmented learning for use in the assembly line manufacturing context.

Keywords: assembly line, augmented reality, digital twin, manufacturing, smart factory, smart worker, training

Overview

In a typical assembly line setting, a product moves from the beginning to the end, and along the way, operations are performed on the product to iteratively transform it into the desired final result. This process takes place across potentially hundreds of steps, each of which may have dozens upon dozens of variations that make it nearly impossible for a single worker to recall each step easily and without error. The beauty of the assembly line is, of course, that each individual step is rather simple and straightforward, requiring very little cognitive and physical ability, so anyone can perform it given the proper guidance and training. Given the high complexity of the overall process, however, and the tendency for factory settings to rotate workers between multiple steps on any given day, the challenge of training workers just-in-time is one that should not be taken for granted.

Motivation

Approximately 12 million people in the United States work directly in the manufacturing sector, and a majority of these workers have only a modest education[1]. By using computer game and augmented reality technologies, it is possible to enable smart workers to learn more efficiently. Augmented reality, or AR, refers to the superimposition of virtual objects in the real world and has many possible benefits when it comes to solving issues that are related to when workers are required to learn new or updated skills. Equipping manufacturing workers with AR-enabled headsets will allow them to utilize AR-game techniques and receive just-in-time training and workflow support, maximizing their output, which is beneficial to both the worker, who gains a higher job satisfaction, and the manufacturer. In addition, the assembly line process, which gained popularity during the industrial revolution for its high efficiency, is not completely error-free, as in some cases a step may be missed or a malfunction in the assembly process may result in a flawed product[2]. Oftentimes, when a product is tested at the end of an assembly line and a defect is found, the whole batch goes to waste. AR can make the training process more effective, thereby reducing wasted resources and mistakes.

Aside from the productivity differences that an AR-based solution might make, there are many other benefits to take into consideration as well. One implication is regarding repetitive motions. It has long been understood that repetitive motions are detrimental to the physical being[3], as it may result in serious, irreparable injuries such as tendinitis (when the tendon in the worker's arm becomes inflamed) and bursitis (when the area between the tendon and the bone becomes inflamed). Repetitive motions can also have mental implications as well; performing the same task over a long period of time may result in decreased mental awareness, which often leads to carelessness and therefore an increased chance of accidents (which in some cases can be fatal). Common job rotation strategies attempt to relieve this problem, but doing so wastes valuable time as the worker needs to be retrained every time a new task is given. An AR solution would not only make training quicker and more efficient but make learning more fun and straightforward.

Goals

The goal of this project is to design and develop a prototype for a head-mounted AR application that demonstrates just-in-time augmented learning for use in the assembly line manufacturing context. For the past year, I have been studying and analyzing existing use-cases for AR applications in the workplace as well as how AR can be used in the context of training and performance support, with some inspiration taken from popular video games that exhibit complex object-oriented systems. In particular, this report aims to answer the following important questions:

- How can AR be used to enable smart workers?
- What techniques exist that leverages the advantages of AR in the assembly line context?
- What type of prototype best demonstrates the feasibility of just-in-time augmented learning?

The prototype described in this report uses techniques from various game-based and AR systems to successfully recreate the assembly line experience while offering an enhanced training solution for smart workers. By leveraging the unique capabilities of AR combined with a physical simulation of the assembly line process, the prototype demonstrates a working approach to delivering AR-based training for a complex product across multiple assembly steps. It is without a doubt that AR can change people's lives, from the applications we see today to the smart factories that affect the workers of tomorrow. Technology like this aims to better the everyday lives of human beings to create a better and happier place to work.

Background

Before discussing the details regarding the design and development of the prototype, it is important to first understand the basics of the manufacturing process and the context in which the problems this prototype addresses exists. In particular, it is imperative to delve into the definition of the "smart worker" and how access to AR-based tools distinguishes them from the assembly line workers of today. It is also useful to analyze the different aspects of the manufacturing process and understand how they interact with each other, especially when it comes to training and the effects AR might have on existing training techniques.

The Smart Worker

Many of those who work in a factory are blue-collar workers with only a modest education, but they use their job as an opportunity to move up in the social ladder. Equipping these workers with smart headgear processors fitted with AR-enabled goggles, audio headphones, and video cameras, among other things, will empower them to accomplish tasks more efficiently and at a much higher rate[4]. There are dozens of benefits that this smart technology can enable, from safety features such as automated access restriction and emergency notifications, to risk mitigation features like defect identification.

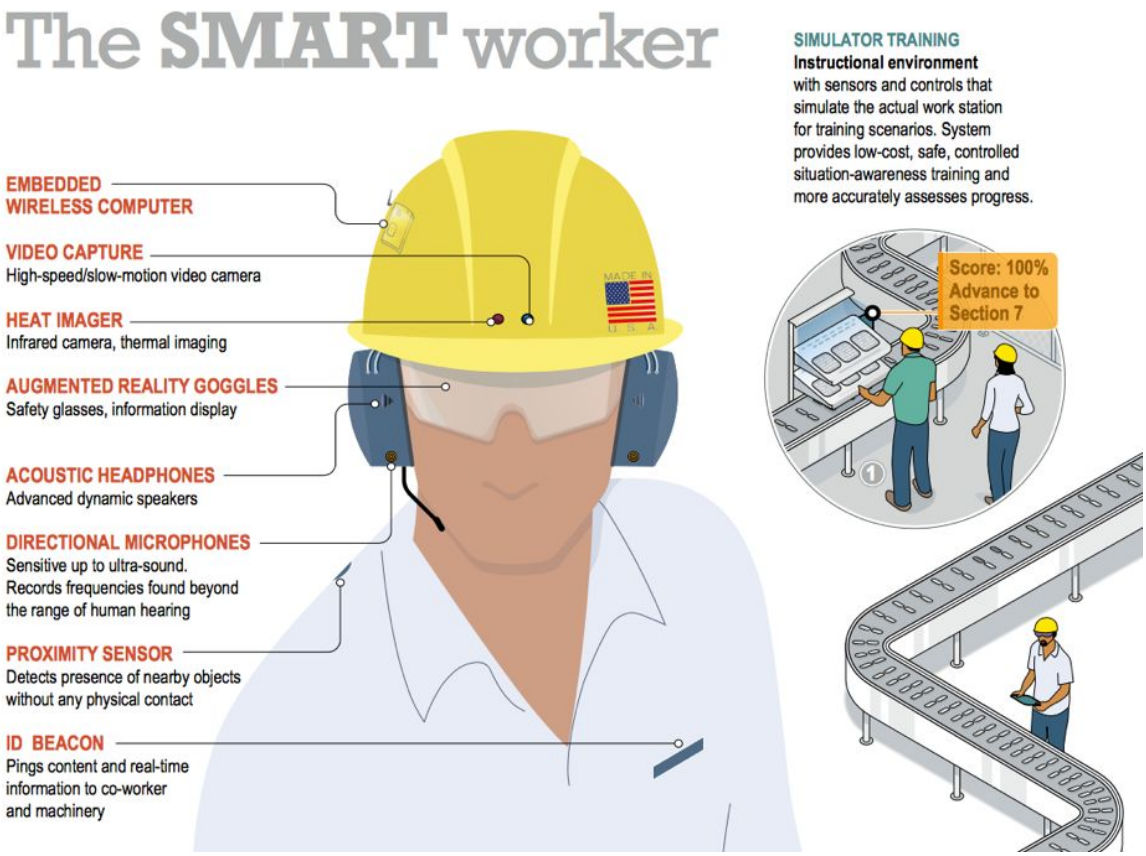

Figure 1. The smart worker and simulator training (Calit2).

The focus of this prototype is strictly training, so while sensor data such as location and infrared can be useful in other applications, they will not be discussed below. It is important to note, however, that such data can be useful in conjunction with AR-enabled devices in the future.

The Manufacturing Process

The manufacturing process consists of a series of steps as the product moves from its initial state to its final form. There are four primary stages of the manufacturing process: preparation, assembly, testing, and packaging. The focus of this prototype is assembly, but it is important to understand how each of these stages feeds into and affects each other.

Preparation refers to the protection, storage, and control of materials before it is used in the assembly stage of the manufacturing process. In a step known as material handling, these materials are kept maintained in a storage facility or warehouse until they are needed, after which they are transported to the assembly line itself. The assembly, which refers to the series of steps between the procurement of materials to the creation of the final product, typically involves an assembly line that consists of several stations, each of which contributes in some way to the assembly of the product itself. Afterward comes testing (also known as quality assurance or QA), which refers to the practice of identifying mistakes or defects during the preparation and assembly processes. Testing may occur during or after the two previous stages and ensures that the items in question meet some threshold of quality before proceeding, and follows the principles of “fit for purpose” and “right first time.” Finally, packaging and labeling prepare the final product for distribution by enclosing and protecting it before it is transported to consumers.

In each of these stages, training plays a major role in informing the worker of their tasks[5]. The flow of this process is very linear, and without the proper training, a worker might unintentionally create a bottleneck in the workflow and slow down the entire assembly process. Any of these steps pose some risk of error, and the best way to create a successful manufacturing workflow is to prevent errors from happening in the first place. While this prototype focuses on the assembly stage, training can affect all stages and is thus an important problem to consider.

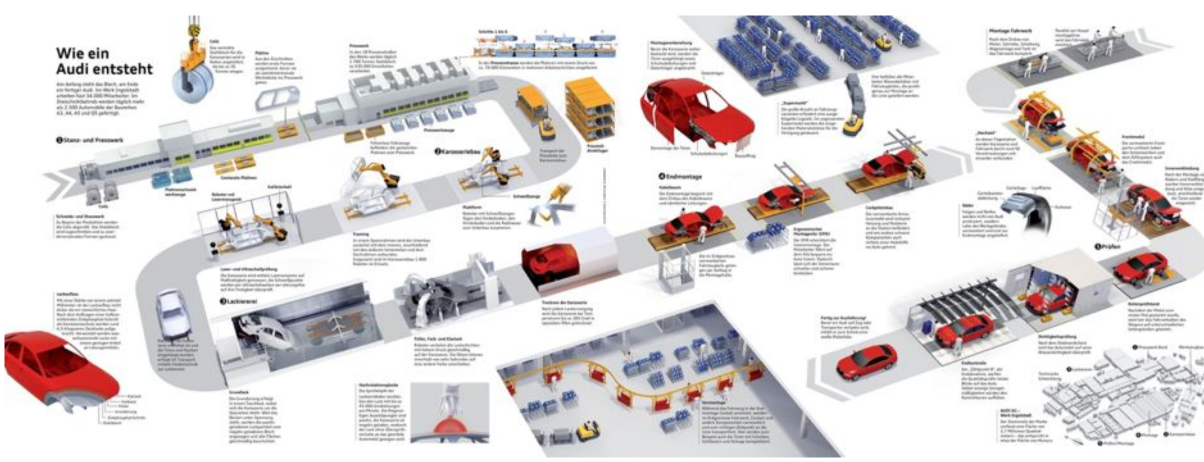

Figure 2. An artist's rendition of an automobile manufacturing line (Audi).

It is also important to identify the specific setting for which this prototype will be modeled. There are four different types of manufacturing settings, each of which have unique features that set them apart from each other.

Small job shop manufacturing settings are most often used in the production of customizable consumer products in which larger, more automated manufacturing settings are inefficient; these systems typically employ several workers who must manually assemble each individual product, possibly with the help of machines, to unique specifications given by the customer. Mechanized mass manufacturing settings are used in the production of small consumer products; these systems are capable of producing many identical products concurrently. Large-scale assembly manufacturing settings are responsible for the mass production of large consumer products, including automobiles; in these settings, the base components of the product are usually similar or identical, while additional components are customizable. Hazardous material transfer manufacturing settings differ from other manufacturing settings in that they typically involve the production of raw materials through a series of high-danger, high-hazard transformations; in these manufacturing settings, materials are often heated and melted in order to produce a consistent final product that can be used in other manufacturing settings.

This prototype will focus on the large-scale assembly manufacturing setting.

Problems in Manufacturing

There are many recurring problems or issues in manufacturing that may occur throughout any given day. In the context of manufacturing, issues regarding efficiency, quality assurance, safety, and effective training are commonplace and require utmost attention. By focusing on a worker-centric solution that enhances the abilities of employees in the manufacturing field using AR, there exists a potential to increase not only the overall productivity of the manufacturing process as a whole but also the well-being of the workers themselves.

Of the four categories of manufacturing problems mentioned above, problems in effective training are perhaps the most important and the focus of this prototype. Nevertheless, it is valuable to understand other problems in manufacturing to make sense of the context in which these training issues occur. Problems in performance optimization refer to issues that are related to the manufacturing process' ability to produce products as quickly as possible; a manufacturing system with high efficiency is one that maximizes production by optimizing its resources. Problems in quality improvement refer to issues that are related to the detection and resolution of defective products; a manufacturing system with high quality assurance is one that is able to detect defects, identify the source of the defects, and resolve issues in as little time as possible.

Problems in safety refer to issues that are related to the health and well-being of the worker; a manufacturing system with high safety ensures that workers are aware of common safety hazards and procedures as well as emergency protocols.

It should be noted that these problems may not occur at all if effective training is in place. Problems in effective training refer to issues that are related to when workers are required to learn new or updated skills. A manufacturing system with high effective training is able to inform the worker of their tasks and integrate the training process into the workflow. Consider the scenario in which a worker is given a new task but is unable to learn it. Irene and John are co-workers at a manufacturing plant. One afternoon, Irene is not feeling well and asks John to take over her task for the remainder of the day. Jon agrees, and Irene proceeds to teach him the task. However, John, who is unfamiliar with the type of task he is being assigned to do, struggles to learn it. After twenty minutes, Irene determines that John knows “just enough” to complete the task and returns home. John is unable to perform the task effectively and production is halted. Irene’s urgency to return home combined with John’s failure to learn the task results in a halt in production for the remainder of the day. However, even if Irene was able to spend more time teaching John, the problem rests mainly on John’s ability to learn it. The best approach would be to take Irene out of the picture and focus on John; if John had been able to learn the task quickly, this problem would never have occurred. A solution would be able to directly teach John the task at hand at his own pace, without the need for someone else to intervene.

Imagine what this scenario could have looked like had an AR-based just-in-time training system been put in place. Irene is not feeling well and asks John to cover for her. John agrees, and Irene returns home. John is unfamiliar with the task, but he uses his AR goggles to display a tutorial while he is performing the task, using it as a reference. Although it takes some time for him to completely grasp it, John is able to adequately perform the task for the remainder of the day. In order for this solution to be made possible, there must exist a reliable and well-designed AR solution that takes advantage of the most recent developments in technology.

Performance Support Model

The graph below describes the flow of work between stages of the manufacturing process and covers the four categories of scenarios: performance optimization, quality improvement, safety, and effective training. It is an attributed directed graph, with each node representing either an agent, a tool, an action, or a resource. An agent is denoted in the bold text without a surrounding shape (e.g. “smart worker”) and is defined as a person who is capable of performing an action. A tool is denoted with a surrounding yellow hexagon (e.g. “smart headgear processor”) and is itself a processor in a system of resources that is used by an agent to perform an action. An action is denoted with a surrounding blue rectangle (e.g. “perform”) and is performed by an agent to produce one or more output resources; an action with an infinity (∞) symbol beside its name represents an action that is continuous. A resource denoted with a surrounding red oval (e.g. “skills”) and can either be used as an input for an action and produced as an output of an action.

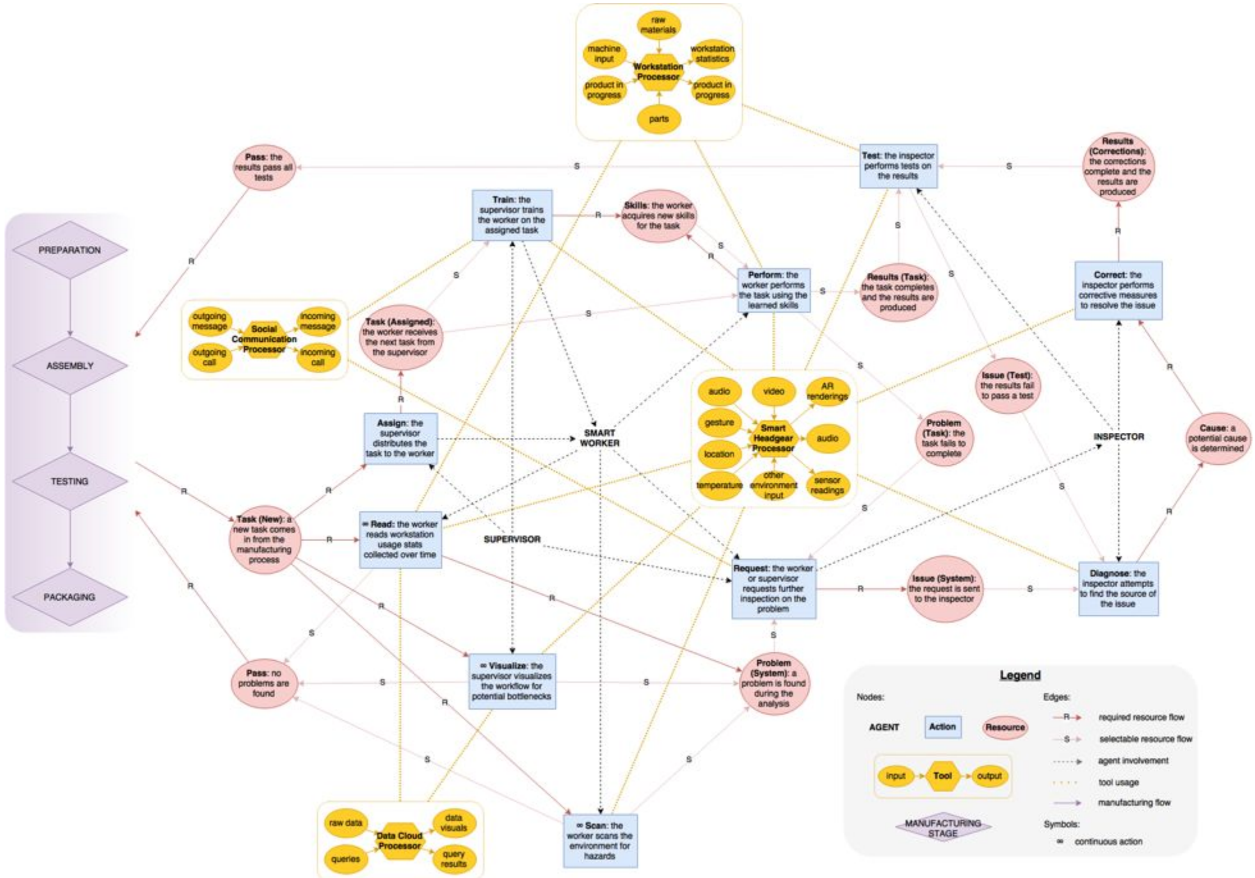

Figure 3. The attributed directed graph for generalized performance support (https://drive.google.com/file/d/0By6c-eOPVBLCYldaQUF6OXFqa0k/view?usp=sharing).

The nodes are connected via a series of edges represented using lines. Each edge symbolizes a course of flow between nodes. A dashed black edge represents an agent’s involvement in an action. An edge directed towards an action represents an agent’s direct involvement; an edge directed away from an agent represents an agent’s indirect involvement. A solid red edge represents a resource’s flow from one action to another. A dotted yellow edge represents a tool’s usage by an agent in an action.

The purple nodes and edges on the left of the graph represent the manufacturing process.

The letters “R” and “S” on the red edges signify the flow condition of resources needed by or produced from an action (a la a Petri net). The symbols are with respect to the actions, which serve as transitions. Input edges labeled “R” denote a required resource that must be used for the action to fire; input edges labeled “S” denote a resource in which exactly one of the “S” resources must be selected for the action to fire. Output edges labeled “R” denote a required resource that must be produced from the action; output edges labeled “S” denote a resource in which exactly one of the “S” resources must be produced from the action.

The primary node of interest in the graph above for the prototype described in this report is the action node "Train". Notice how in this graph the supervisor is the one who trains the smart worker through a social communication processor. By implementing an AR-based learning solution that utilizes modern training techniques, both the supervisor and social communication processor can be eliminated from the training process, creating a solo learning experience for the smart worker that may actually prove to be more efficient.

Just-In-Time Training

In any workplace, the job training process plays an important role in informing the worker of their tasks and responsibilities. Workers are trained to perform effectively and safely in their work environment as well as other aspects of their work including proper workplace behavior, policies, and other regulations. Oftentimes, job training focuses on how to respond to different scenarios that can range from trivial (e.g. how to interact with coworkers) to life-threatening (e.g. what to do when there is a factory fire). Training can be either off-the-job or on-the-job[6].

A job training technique known as training and development (or human resource development) emphasizes not only the learning process but also the efficiency factor[7]. Training and development cover three major activities of job training: training, education, and development. The goal of these activities is to motivate the worker in believing in their long-term potential in hopes that their performance would improve under the mindset that opportunities for growth exist. It is thus important to keep workers engaged with their work by implementing techniques that boost productivity and growth.

Job rotation strategies prevent workers from becoming disengaged from their work by adding a variety of different tasks to their work day. It is also useful to train multiple people on a single task so that options exists if one needs to miss a day of work. In order to implement a successful job rotation strategy, a training technique known as just-in-time training may be used.
Just-in-time learning sets aside traditional job training techniques in favor of a more on-demand approach[8]. Instead of undergoing hours of training (either on-the-job or off-the-job) prior to the worker’s employment, just-in-time training runs under the principle of “training when needed,” or training workers only when the need to do so arises (e.g. in temporary worker replacement). For simple jobs like manufacturing, just-in-time training can be more effective than traditional training.

Modern just-in-time training systems use visual media and shared technologies in order to facilitate the learning process. Using a laptop computer or tablet, workers can learn new skills on the fly or in their workstations. The use of modern search engines and wiki technologies allow for the ability to find and share solutions to common problems or tutorials for common tasks. Although skills can be learned just by reading the materials, visual learning techniques using videos or diagrams have proven to be faster and much easier to understand, especially among blue-collar workers.

Just-in-time training can be improved using AR in a technique known as just-in-time augmented learning[9]. In this technique, workers can access just-in-time training resources without the need to refer to a laptop computer or tablet by loading them directly onto their AR device. This integrates the training process into the workflow and reduces diversion. Just-in-time augmented learning can be implemented in several different ways. In an overlay video demonstration, a pre-recorded demonstration is played on the side as the worker completes the task. In an overlay animated demonstration, a pre-rendered animation is played as the worker completes the task. In an interactive animated demonstration, multiple short pre-rendered animations are played as the worker completes each step of the task rather than a single, longer animation that shows the entire task. In interactive a telestrated demonstration, an off-site support person draws animations as the worker completes each step of the task.

Design

Even with the problem statement narrowed down to the focus of effective training, there are still many things that need to be considered before developing the actual prototype. For one, there is valuable insight to be gained from examining popular video games on the market today that leverage object-oriented systems, some of which even mimic the assembly line process quite closely. It is also important to analyze different approaches to AR, as the technology has been widely available for several years on multiple different development suites and platforms. In terms of the design of the application itself, there are many options that lend themselves well to an AR application for manufacturing, so it is vital to analyze these different options and choose the best tools and software libraries that would allow for the quick development and deployment of the prototype.

Game-Based Systems

The prototype takes the form of an interactive building game in which the user wearing an AR headset incrementally builds a product from start to finish using visual cues superimposed onto the physical world. Before designing the prototype, it is important to first analyze interactive game-based systems with object-oriented functionality that exist on the market today. The games below span multiple genres and play quite differently, but they all share one trait with each other: they all showcase the complex, interactive nature of object-oriented game-based systems.

Figure 4. Screenshots of Contraption Maker (top-left), Minecraft (top-right), Infinifactory (bottom-left), and Fallout 4 (bottom-right).

One of the key features of any assembly is the use of parts to form a whole. Treated individually, these objects have their own characteristics, but when working conjunctively they become something greater, obtaining functionality that was they did not have before. Many games simulate this object-oriented structure by applying properties to individual objects that interact with each other to accomplish some goal. For example, the puzzle game Contraption Maker (a spiritual successor to the 1993 classic The Incredible Machine) requires the player to use objects provided to complete a Rube Goldberg machine and accomplish a simple, predetermined objective (e.g. put the ball in the hoop). It features multiple objects, each of which has their own unique characteristics and properties, and although the interaction between any two individual objects is usually quite simple, it is the combination of these simple interactions that make the system as a whole so complex. It is useful to view the entire system by parts, with each part dependent on the outcome of the previous processes, and if the outcome of the system is not what is expected, it is easy to quickly identify the part that failed.

Many games have some sort of scoring system in order to track player progress and motivate them to perform better at a particular task. In the context of manufacturing, some means of evaluating efficiency can provide meaningful feedback to the worker in terms of how well they are performing. The puzzle game Infinifactory developed by Zachtronics Industries showcases such a concept. Infinifactory is a factory designing game in which input blocks must be transformed into the desired output and then transported to the goal. Input blocks can be transformed using special blocks like welders and eviscerators, and they can be transported using moving blocks like conveyors and pushers. Solutions are scored by space used (footprint), time (cycles), and the number of blocks used (blocks).

One important aspect of video games that is often overlooked is user interface design, particularly because good user interface design succeeds when it is least noticeable. This is especially true when applied to AR, as the need to identify and interact with virtual objects in the real world mandates a proper user interface. The following two games are not AR games but utilize techniques that translate well to the AR space. In sandbox world-building exploration game Minecraft, the first-person perspective of the player makes it difficult to have an all-encompassing view of the complete system, and thus accessibility to every part of the system, including every workstation, is required to successfully maintain its complexity. In popular role-playing action game Fallout 4, a modern graphical user interface informs the user about the various in-game objects onscreen, feeding as much relevant information to the user as possible and keeping them up to date with their environment. The player can focus on an object by hovering the reticle over it, thereby highlighting the object, indicating that it is selectable, and selecting an object reveals information about it, including its description and what resources it needs.

Not all of these systems were used in the final prototype, but they did provide some valuable information as to how to approach designing a proper game-based AR prototype.

AR Systems

Most of today's AR applications use the smart device as an interface[10]. The ubiquity of smartphones and tablets have facilitated the development and distribution of AR applications in the mobile market. Typically, the smart device uses its back-facing camera to locate and identify a 2D marker in the real world and superimposes a virtual image on the device's screen over the video captured by the camera. These marker-based solutions have existed in the development community for several years, and many development platforms, including mobile and headset options, have native support for AR toolkits.

The 2D marker, however, is restrictive. Markerless options exist in order to free the developer from the limitations of marker-based AR like occlusion, range, orientation, and lighting conditions. Instead of attempting to locate a 2D marker in 3D space, markerless solutions attempt to identify patterns or entire objects. While significantly more complex, object recognition tools are starting to become widely available and make markerless AR development relatively straightforward, and other pseudo-markerless solutions (such as fiducial marker placement) are currently being investigated that can provide the benefits of markerless AR while retaining the simplicity of marker-based AR.

Figure 5. A simple marker-based AR system for manufacturing.

Recently, new 3D infrared cameras have enabled developers to use object tracking technologies to identify objects in the real world by creating depth maps that define the 3D space around the camera[11]. These objects can, therefore, be saved as these 3D maps, which allows them to be used in other applications. While this technology is still relatively new, support has been integrated into many popular development platforms. This should make development relatively simple, if not for the issues concerning cost (the cheapest 3D infrared camera costs at least several hundred dollars) and robustness (3D mapping remains imperfect at this stage).

It is important to recall the goal of this prototype: to demonstrate a working AR solution a model of the manufacturing environment. This demonstration should test assumptions regarding creating an AR application and illustrate the feasibility of such a solution when applied to the real world. The use of markerless solutions, such as a 3D infrared camera, is enticing, but the fact that it is so costly and that the technology is still relatively primitive prevents it from being a viable solution. In fact, upon further testing, it was found that even the most basic of object-recognition libraries can identify the correct object less than 10% of the time, even after attempting to control the scanning environment (e.g. by diffusing light sources and normalizing the scanning procedure). On the other hand, marker-based AR works with exceptional accuracy and is, therefore, a perfect fit for a prototype that is meant to demonstrate the usefulness of AR rather than solve an object-recognition problem. More about how this decision was made can be found in the initial design described below.

The Digital Twin

A relatively new concept called digital twin technology refers to the idea of modeling physical objects, settings, and processes virtually such that they can be used in various software scenarios to train, demonstrate, and educate[12]. They do not replace their physical counterparts, but rather complement them, supplementing the user with unprecedented access to information and analytics from the physical world while leveraging the unique advantages that only a virtual model can provide. This one-to-one relationship between the physical and the virtual delivers a means of unique, simultaneous interactions between two separate but interconnected systems, and in the context of manufacturing, is not only able to simulate the manufacturing process but also improve it with a meaningful AR-based training solution.

Figure 6. An example of a digital twin (Siemens).

In terms of the specifics of constructing a prototype of the manufacturing line, there are two primary options. The first forgoes the idea of a physical prototype and instead opts to model everything in a virtual 3D space. This creates a virtual prototype rather than a physical one, allowing for a more detailed world where the environment can accurately mimic that of a real-world setting. As everything would be built using 3D modeling software, there would be no cost in creating this virtual environment, outside of a means to display the content. Furthermore, objects in the virtual world can be manipulated with ease, making modeling state changes especially easy.

As simple as a virtual prototype may seem, there are some intrinsic problems with it, particularly when it comes to whether or not it will actually provide a sufficient proof of concept. As the AR application is supposed to work in the real world, the dichotomy between the virtual and the physical world is inescapable. Another approach then would be to create a physical prototype, one that is grounded in reality inasmuch as it demonstrates the feasibility of an AR platform in a manufacturing setting. There needs to exist a physical parallel for each manufacturing concept, namely the assembly line, the product-in-progress, and parts.

Modeling a representation of the assembly line is relatively trivial, as any type of track (even an imaginary one) will suffice. Creating physical representations of the product-in-progress and product parts is less straightforward. Two intuitive methodologies arise when it comes to solving this problem: the replacement method and the iterative method. Both methodologies require parts of the product body to be placed atop a rolling chassis. The difference is that the replacement method swaps out the body in its entirety at every state change, while the iterative method simply adds parts onto it. There is, however, a third option to represent the assembly line, and that is to completely forgo the idea of a dynamically changing product-in-progress and opt for a more static approach. That is, rather than have a product that is constantly transforming from one state to another, there can instead be multiple instances of the product, one at each station. This better resembles a real-life assembly line in which each station is performing an operation on the product-in-progress at all times, and eliminates the need to program state transitions between steps, simplifying the physical model yet maintaining the illusion of progress that a dynamically-changing product would give.

Initial Design

The initial design of the prototype differs greatly from that of the final product, but it is still worth noting here.

The idea of a smart advanced manufacturing system can be modeled as an AR "building game." In this game, the player is asked to perform a sequence of tasks that ultimately builds towards a final product called the "objective." Each objective is made up of multiple parts; thus each step is designed to incrementally build towards said objective such that the player can progress through each level regardless of complexity. The addition of AR not only enables players to physically interact with the game environment, but it also offers new tools for interfacing with the game and providing user agency. The physicality of the objects in play makes player actions and consequences feel more direct as opposed to playing a game in a completely virtual space.

The goal of designing this game is to make it both simple and intuitive. Accomplishing this requires an easy-to-use user interface. On startup, the main menu is launched, asking players to choose one of two options: "Level Select" or "Settings". The former brings the player to the level selection screen (described below); the latter brings the player to the settings screen in which various options (such as in-game volume) can be adjusted. With the addition of AR, the level selection screen is unlike any other traditional video game. The player is asked to bring a QR marker (which represents a level) into view. When the game recognizes the QR marker, it shows the following options: “Start Level,” “Difficulty,” and “Leaderboard.” The first option begins the level. The second option allows the player to adjust the level’s difficulty (described below). The third option shows a list of high scores for that level. During each level, the user interface will show the following information: level name, difficulty, steps left, and time elapsed. This basic user interface will remain static throughout the entirety of the level. In order to prevent the screen from being cluttered, all AR elements and text will be minimalistic, using simple, bright colors that pop out against the physical environment.

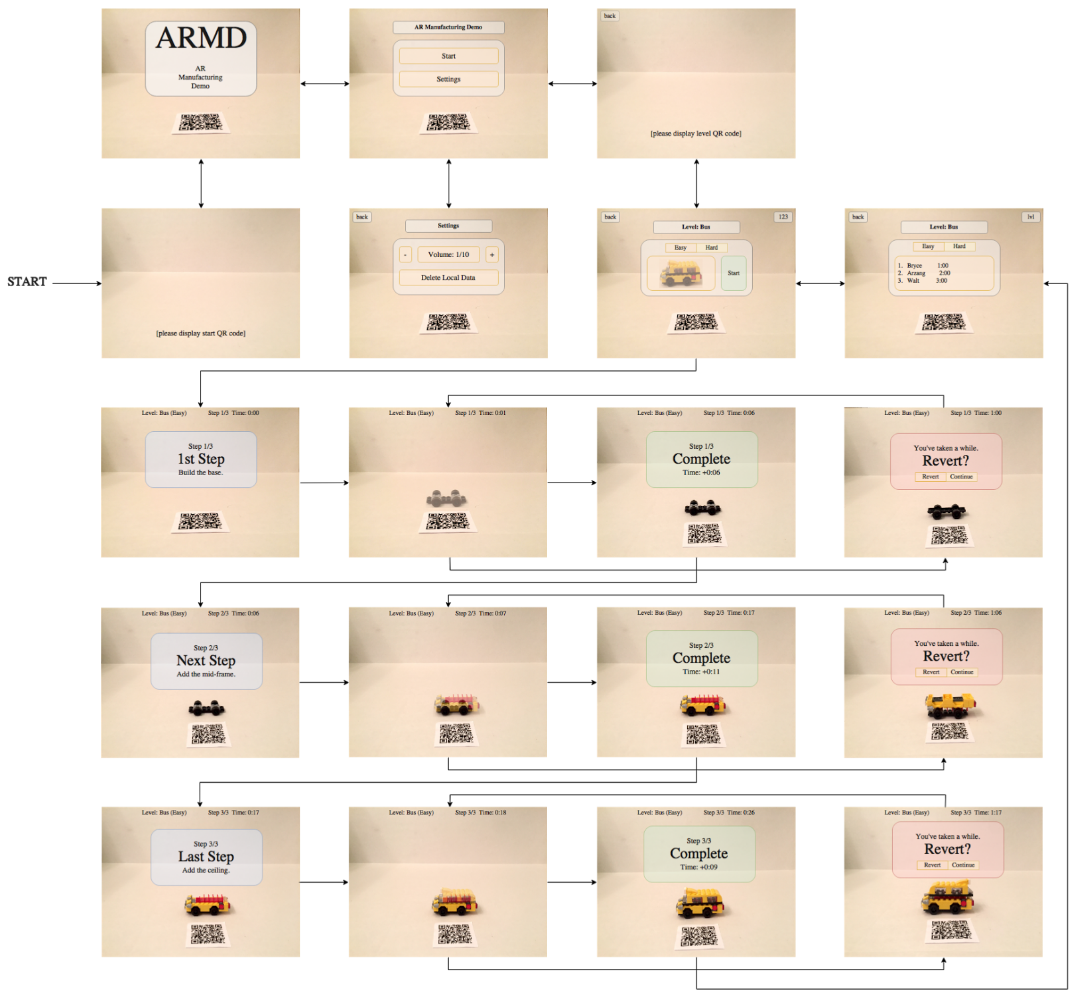

Figure 7. A storyboard concept flow for the initial prototype design (https://drive.google.com/file/d/0By6c-eOPVBLCTDlNN3hUVWlqZFU/view?usp=sharing).

A level is made up of multiple steps that build towards the objective. The choice of objective depends on the QR marker scanned at startup. Before gameplay commences, the game attempts to locate and identify a QR marker that corresponds to a game level. Upon identifying the QR marker, the game begins to instruct the player to complete the first step. Usually, this step is relatively straightforward, such as finding the first piece or assembling a base component. In order to communicate the instructions to the player, a faint outline of the desired outcome of that step is superimposed atop the QR marker (in subsequent steps, this overlay is displayed atop the objective-in-progress). When the game recognizes that the first step is complete, it moves on to the next step. Note that the QR marker remains in place until the end of the level. The subsequent steps follow the same general pattern as the first: instruct the player to add onto the current objective-in-progress and move on to the next step once complete. Whenever the game fails to identify the state of the game (e.g. the QR code moves out-of-view or the objective- in-progress is no longer recognizable), an AR pop-up window will appear asking the player to remedy the situation or to revert to a previous step. When the final step is complete, the game congratulates the player and returns to the main menu.

Redesign

Significant changes were made to the initial prototype design since its conception described above. These changes were made in an effort to make the prototype more demonstrable and push the project forward at a feasible pace while retaining the core aspects of the prototype design and goals. Among these changes was a significant narrowing of scope, especially when it comes to the menu design (which plays only a minor role in the prototype). The prototype was also rescoped to focus only on a single level and difficulty, as well as other changes described below.

The biggest change is the omission of object recognition functionality from the initial prototype. Early versions of the prototype utilized object recognition functionality to identify each step of the manufacturing process and quickly detect errors when found, but as mentioned before, the correct object was identified less than 10% of the time. Because it was nearly impossible to get object recognition to work consistently in even the simplest of scenarios, it was decided that such functionality would be temporarily set aside in order to get essential parts of the prototype working as soon as possible.

Therefore, the focus of the prototype design was shifted from "object recognition-based error detection" to "simulation-based training methodologies". In other words, the goal of the new prototype would be to create a simulated environment that reasonably reflects a hypothetical AR training setting in the context of advanced manufacturing systems. The first prototype redesign involved creating a miniature model version of a factory floor, with virtual representations of each manufacturing step moving down an imaginary assembly line. The user would use a QR code to spawn this simulated environment in the real world and build each step of the product as their respective virtual representations appeared on the assembly line. This would all be done in a single view, with the user building the product within the virtual factory as if they were working in a real one.

The biggest problem with the factory prototype redesign was that there were major limitations when it came to using AR to create sophisticated superficial features that interact with real-world objects. For one, there was no intuitive place to overlay the virtual factory setup relative to the QR code required to display it, as any simple setup could obscure the QR code itself and cause the AR application to lose track of it (e.g. by inadvertently covering the QR code physically). Furthermore, because the AR layer is always on top of any object in physical space, all virtual factory assets would have to be positioned away from the product-in-progress insomuch as to not get in the way of the user's view, undermining the need of using a virtual factory setup in the first place. Finally, virtual objects do not scale with real-world objects, especially with regards to perspective angles.

Therefore, a second prototype redesign was created that would strip down the AR-based features but retain the same feel of a simulated training environment. This redesign completely omitted the superficial virtual factory features of the previous redesign attempt but kept the idea of having an imaginary assembly line as a training paradigm. With this idea in mind, a two-view approach was created—in addition to the AR application itself, a second system would run a simulation of the assembly line on a different display, and the two systems would work together in order to create an immersive training experience. To simulate an assembly line, the second system would allow the user to scroll through several assembly steps and build the product at their own pace by viewing a virtual model atop the virtual assembly area. New "expansion" and "annotation" features would be included to let the user easily see which pieces were being added at each step.

This second redesign attempt would serve as the foundation for the final handheld prototype implementation described below.

Implementation

Two separate versions of the prototype were created, both of which share the same AR application but differ in the implementation of the assembly line itself. Namely, the virtual prototype simulates the assembly line process on a flat multi-touch screen while the physical prototype uses real-life objects to recreate a miniature, working model of an assembly line. The conjunctive use of the AR application and the respective assembly line models make for a valuable training experience that demonstrates the smart worker headset processor as a user interface to AR-based manufacturing training and operations process support. In both prototypes, Lego bricks were chosen as the building blocks of the product-in-progress not only because they are familiar and easily distinguishable, but because they are naturally designed for the purpose of assembly.

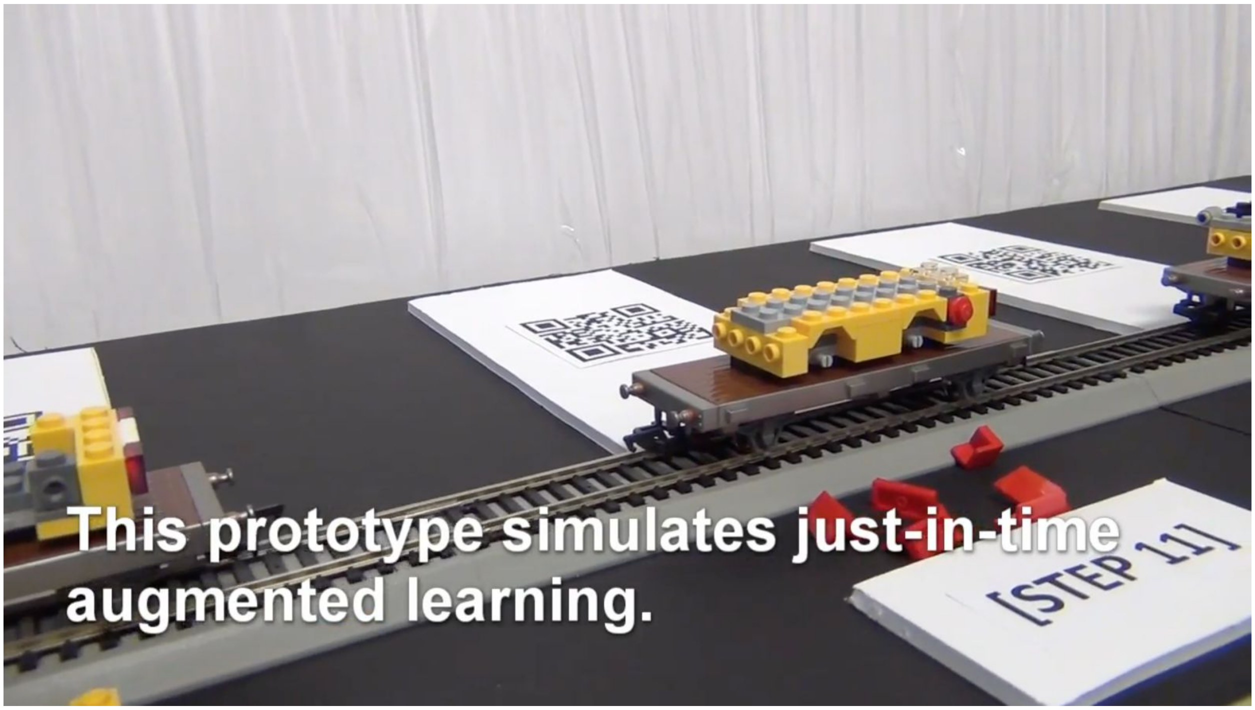

Figure 8. An overview of the final prototype implementation (https://www.youtube.com/watch?v=PFJrOvf40wk).

AR Application

Both the physical and virtual prototypes share the same AR application, which forms the core the AR experience through which the smart worker will train. The application itself was developed in the C# programming language using the Unity game engine, which is popular among game developers because of its ease of use and robust functionality[13]. All AR operations were handled by the Vuforia Augmented Reality SDK, which allows for many flexible options when it comes to adapting AR into any device. Therefore, this application runs on any Android device, as well as any Google Cardboard headset with Android compatibility. The device chosen for this prototype is the Lenovo Phab 2 Pro Android smartphone for its large screen and excellent camera quality. The AR application uses the phone's back-facing camera to detect and recognize QR codes that appear in the assembly line model (details about how this is implemented can be found below) and superimposes the respective virtual models on the screen.

The user interface is designed to be simple and non-distracting, focusing on providing the most relevant information using as little text and screen space as possible. This is because it is important for the worker to focus on the task at hand rather than be distracted by unnecessary user interface elements. Across the top of the application is the header bar, which shows the level objective, step number, and time elapsed since the simulation started. The latter is an example of efficiency tracking, which gives the worker some idea as to how well they are doing in performing the task. Notice how the header bar is translucent, allowing the user to have a complete, unobstructed view of the environment. Directly below the header bar are the detailed instructions for that step, outlined in white to increase the visibility of the small text over an unpredictable background. On the sides are buttons for expansion and annotation, both of which are described later below.

The virtual models shown on the screen were created using Unity's built-in object creation tools, but they could have easily been done via a third-party 3D modeling software and imported into Unity. Each model is actually an amalgamation of multiple models, as they build atop each other with every assembly step. This allows each additional part to be treated as separate objects, giving them individual functionality that is key to the core parts of the AR application. For example, at each step, the "new" parts to be added to the product-in-progress are highlighted in flashing white to easily distinguish them from the rest of the model, which is already assembled, simultaneous showing what parts are needed to complete the step and what the final product should look like.

Two outstanding characteristics of this application are the aforementioned expansion and annotation features. The expansion feature gives an exploded view of the current step—that is, the new parts expand off the current model so they are easily seen and identifiable. Performing the expansion command again will "put" the pieces back to where they were originally, which incidentally demonstrates the physical action required to add the part to the current product. The annotation feature applies floating labels to each part and identifies each by name. This feature is especially useful when used in conjunction with the detailed written instructions on the screen. These labels are translucent and viewable from any angle regardless of the orientation of the AR device.

One more feature added to the prototype is audio. This was implemented very late in the development process, but plays an important role in providing valuable user feedback. For example, when a QR code for a step is detected, a sound that denotes "success" is played, and the instructions for that step are narrated to the user. This is particularly important in the manufacturing context, where most factory workers have limited education and many of them are illiterate. By providing audio as well as visual feedback to the user, they are not able to see exactly what parts are needed, where they go, and what actions need to take place in order to complete a task, all within seconds.

Virtual Prototype

In the virtual prototype, the assembly line is simulated via a separate software application on a second screen. This simulated interactive manufacturing line is run on a Dell touchscreen LED monitor, rotated 90 degrees such that is in portrait (vertical) orientation. The screen is laid down flat such that objects can be placed atop it to simulate the assembly process. The view on the screen is divided into two distinct subareas: the demonstration (or "demo") area on top and the workspace (or "work") area on the bottom.

Figure 8. A screenshot of the virtual prototype showing the demonstration area and the work area.

The demonstration area is where the QR codes that correspond to each step appear—one at a time and in sequence—as well as virtual models for each that can be seen using the AR application. The user can scroll through each QR code by tapping the "PREV" or "NEXT" buttons on the bottom of the screen, which toggles between the different steps of the product. This is somewhat different from one might expect from a typical assembly line, as the product-in-progress stays in place while the "stations" (represented by QR codes) are the ones that are moving over time.

The workspace area is where the user will build their physical version of the product. This allows the worker to view a virtual model of the product in the demonstration area where working on the product itself directly below it. As previously mentioned, the product-in-progress is stationary here, and the worker builds the upon each step of the product away from the demonstration area so that occlusion is no longer an issue.

Much like the AR application, the virtual prototype was developed via Unity.

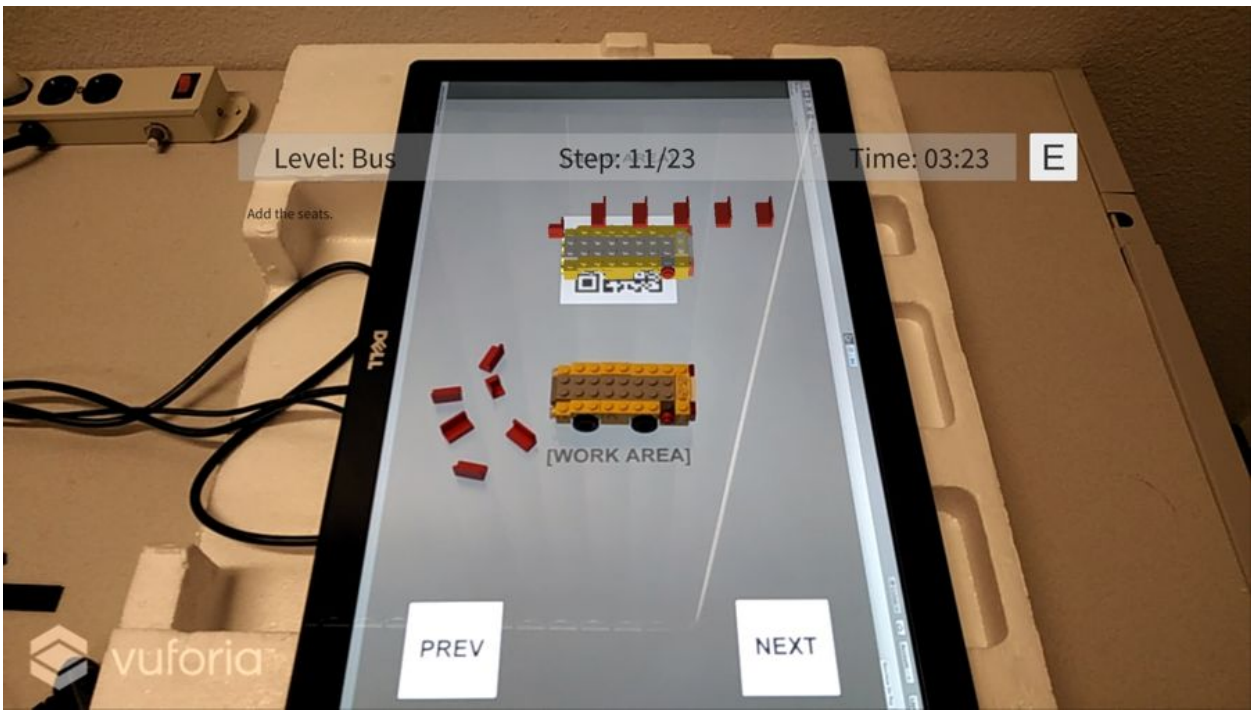

Figure 9. A screenshot of the AR application working in conjunction with the virtual prototype.

The virtual prototype was developed in order primarily because it did not require much effort to put together, as the QR codes were already imported for AR functionality and so needed only to be placed in the scene view to complete the view. However, it proposes an interesting portable training solution that smart workers can take on-the-go with any touch-enabled mobile device, which might be useful for more complex jobs that cannot be learned as easily just-in-time.

Physical Prototype

In the physical prototype, an assembly line was constructed using HO-model railroad tracks and cars. This was the best way to simulate a miniature version of a real-life assembly line, as the cars would carry the product-in-progress down the assembly line track across multiple steps. Along each segment of track is a QR code for the AR application as well as a label that tells the user what step that segment corresponds to.

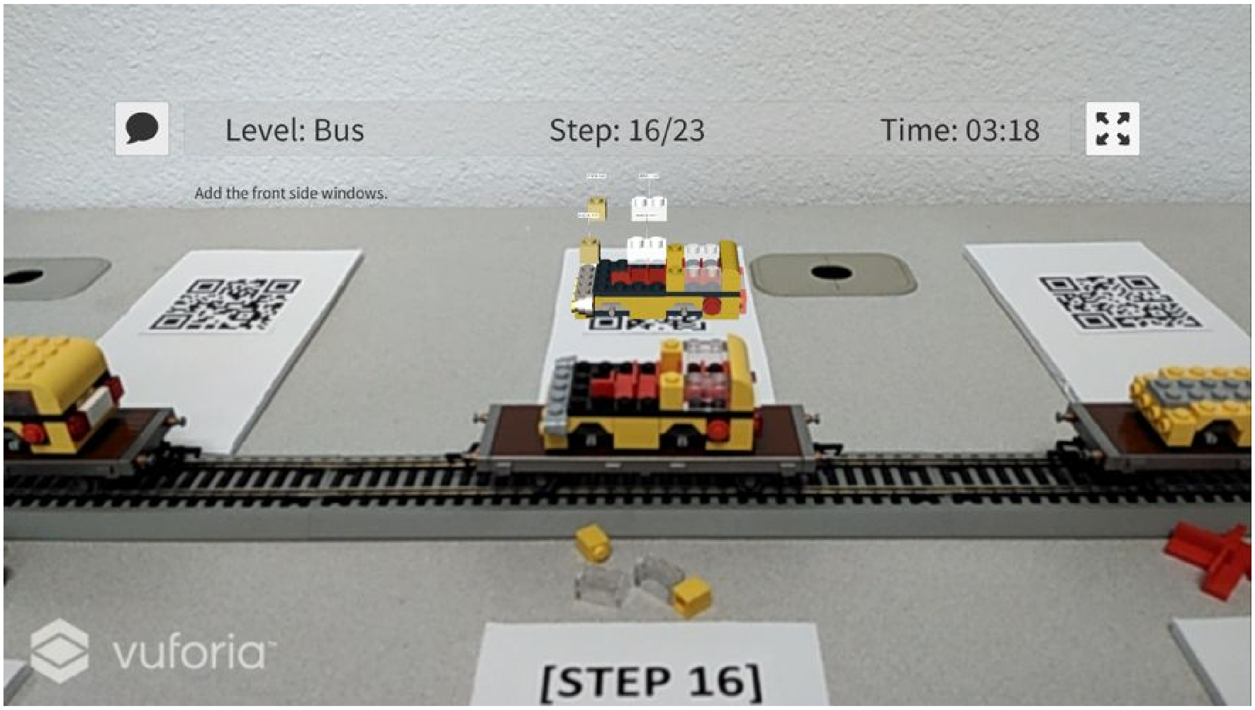

Figure 10. A screenshot of the AR application working in conjunction with the physical prototype.

For demonstration purposes, the physical prototype uses the aforementioned static approach to the assembly process (i.e. each station has its own car and product-in-progress). Only 4-5 steps were chosen in order to represent the physical prototype in a compact manner. In a sense, this setup is more realistic because a worker is more likely to jump between random stations rather than go through them sequentially, and it helped illustrate the overall picture of how the AR application can affect the assembly line process from start to finish.

The physical prototype is preferred over the virtual prototype because it makes for a more convincing argument that such an AR solution can feasibly exist in the real world. In addition to it following closely to the concept of the digital twin, the physical layout of the assembly line model looks and feels like one that might be found in a typical factory, and the use of real-world objects sparks the imagination for where else this type of technology might be used in other fields.

Further Discussion

The potential for just-in-time AR-based learning has far-reaching implications that go beyond just the scope of advanced manufacturing. In fact, after discussing this prototype with others, I have discovered that there is much interest in this sort of technology for applications across multiple different fields. As AR-enabled devices become increasingly commonplace, the market for AR applications will grow, begging the following questions. What comes next? How can this technology be improved? What potential applications exist for AR-based learning in the future?

Assessment & Feedback

Upon the completion of the physical prototype, I had the opportunity to present and demonstrate the AR application at various research forums, symposiums, and industry events. Feedback was generally positive, and many of those who viewed the demo and interacted with the prototype for themselves expressed a desire to bring just-in-time AR-based learning into their respective fields. Current industry professionals were particularly interested in how this technology could be applied in the workplace, while students and professors seemed more interested in how the learning aspect of AR could be used to enhance education. These potential applications are discussed in detail below. The prototype also proved, perhaps unsurprisingly, very popular among children, who gravitated towards the physical prototype due to its toy-like appearance.

Figure 11. Demonstrating the AR application to industry professionals.

In terms of the AR application itself, the primary goal was ease of use. This was tested by handing the AR application to random individuals who stopped by during demonstrations. Giving these "new" users an opportunity to interact with the prototype for themselves supplied valuable feedback in terms of usability and proved that the prototype was indeed intuitive and easy to use. In fact, even among those who had not even heard of the concept of AR, very few had any trouble understanding the concept of the prototype and using it for themselves. This shows just how capable the application is, especially when thinking about uneducated factory workers, and succeeds at being just-in-time.

By far the most common feedback received was the suggestion of commercializing the prototype into a marketable product and partnering with The Lego Group, who recently have been investing in new ways to promote their toys. This is an interesting proposition that can be further investigated if commercialization is an interest of all parties involved.

Future Revisions

There is still much room for improvement when it comes to the current prototype.

As mentioned previously, markerless AR solutions, along with 3D infrared scanning technology, would make for excellent options for developing any AR-based product. However, developer support for markerless options is currently very low, and some of these technologies (such as object recognition) are still highly error-prone and not very consistent. Computer vision and artificial intelligence are current topics that are being studied very closely by leading computer scientists around the world[14], and it is only a matter of time before these technologies will be viable for widespread use in AR applications such as this one.

In a similar vein, due to the nature of the AR application and the limitations of the current state of technology, there is currently no way to reasonably implement this AR-based learning solution on a massive scale. Current AR-enabled headsets with high enough processing power to render complex 3D images are very expensive (the Microsoft Hololens costs between $3000 and $5000) and computer chips of equivalent power are not small enough to fit into smaller AR-enabled glasses. The current prototype uses an Android device on a Google Cardboard headset, which is as affordable as technology like this comes, but the field of view is too narrow for work use, and the device itself is clunky and uncomfortable. There is little choice then but to wait until better forms of wearable technology are developed. Luckily, this is more of a question of "when" rather than "if".

Potential Applications

The potential application for AR-based learning is not just limited to the advanced manufacturing context. In fact, any application that requires learning in some facet can benefit from AR. Below is just a small sample of the many potential applications for AR-based learning, from workflow support in other fields to general consumer products.

The video game industry has always been on the forefront of new technologies, and AR is no exception, especially given the various game-based AR training techniques described above. Specifically, simulation games would benefit greatly from AR-based learning because they oftentimes require the player to perform a series of complicated tasks in a short period of time, an aspect of these games that scare away many new players. Going beyond the context of games, real-life simulation-based training can also be enhanced by AR, notably those that attempt to replicate complex system such as military operations or flying an aircraft.

In the education context, learning in and of itself can be a difficult task. This is especially true for more abstract topics. For example, in computer networking, it may be difficult to visualize the flow of data between devices because data is ephemeral, or in biology and chemistry, it is nearly impossible to demonstrate the complex interactions of molecular structures without some sort of virtual simulation. AR-based learning has several benefits that enhance the educational experience, such as accelerating comprehension among visual learners, allowing for a more collaborative and fun learning environment, and providing access to virtual learning resources on-the-go.

At home, learning takes the form of do-it-yourself projects. From assembling furniture to fixing leaky pipes, the home economics sector is ripe for all sorts of AR-based learning solutions. Consider the simple act of cooking a meal, for example, which is a task that is strikingly similar to manufacturing in that the process follows a series of iterative steps that transform parts into a whole. Current solutions require following an online recipe that lacks visual detail or watching an online video tutorial on a secondary device. Having an AR-based learning solution would help give rise to the "smart homeowner," which is a concept that should be explored once wearable AR-enabled devices become more commonplace in the public sphere.

Conclusion

An assembly line is a sequential and iterative process that may consist of hundreds of steps and variations, and while straightforward it has ample room for errors and mistakes if factory workers are not properly trained. A just-in-time AR-based learning solution would not only increase worker productivity but also worker mental and physical health as well. The prototype described in the sections above hope to accomplish these goals in an effort to enable smart workers and leverage the unique features of AR to the workers' advantage.

The smart worker itself is a conceptual model consisting of various devices and sensors that make up a smart headgear processor. The focus of this prototype is just-in-time training using AR goggles, specifically in large-scale assembly manufacturing settings, but AR-enabled devices have widespread applications in various other stages in the manufacturing process regardless of setting. In designing this prototype, inspiration was taken from various game-based systems in order to create a fun and intuitive learning solution, and marker-based AR was chosen to expedite the development process and provide a consistent AR experience across all devices.

The final prototype effectively uses unique dual-systems "digital twin" approach that both simulates the manufacturing process and provides a meaningful AR-based training solution. The virtual prototype simulates the assembly line on a secondary application that allows for training on-the-go. The physical prototype recreates a real-life assembly line experience that adds tactility to the prototype model. The overall success of the prototype was tested at various events and the feedback received demonstrated the ease-of-use of the AR application. It remains to be seen where this technology will go and in what other fields a just-in-time AR-based learning solution might be useful.

Acknowledgements

This research project would not be possible without the help and guidance from the following groups and individuals. Debra A. Brodbeck, Anna Chang, and the rest of the faculty at the UCI Institute for Software Research (ISR): thanks for providing a location to design and develop the prototype as well as for additional guidance on the project. Said M. Shokair and others from the Undergraduate Research Opportunities Program (UROP): thanks for providing the funding and resources to make developing and sharing this research project possible.

The California Institute for Telecommunications and Information Technology (Calit2): thanks for providing general support for this project overall. Prof. Richard Lathrop, Jessica Shanahan, and others from the ICS Honors Program (ICSHP): thanks for giving me the opportunity to engage in this research opportunity. Finally, to fellow undergraduate research and close friend Arzang Kasiri: thanks for introducing me to this project and for being someone I can turn to for support.

References

[1] "Industries at a Glance: Manufacturing: NAICS 31-33." Bureau of Labor Statistics. United States Department of Labor, 07 June 2017. Web.

[2] Saptari, Adi, et al. "Human Error and Production Rate Correlation in Assembly Process of Electronics Goods." (2014).

[3] Yassi, Annalee. "Repetitive strain injuries." The Lancet 349.9056 (1997): 943-947.

[4] McCutcheon, Bob, Bob Pethick, and Bobby Bono. "For US Manufacturing, Virtual Reality Is for Real." Rep. PwC, Jan. 2016. Web.

[5] Broad, Mary L., and John W. Newstrom. "Transfer of Training: Action-Packed Strategies To Ensure High Payoff from Training Investments." Corporate and Professional Publishing Group, Addison-Wesley Publishing Co., One Jacob Way, Reading, MA 01867, 1992.

[6] Lynch, Lisa M. "The role of off-the-job vs. on-the-job training for the mobility of women workers." The American Economic Review 81.2 (1991): 151-156.

[7] Noe, Raymond A., et al. "Human resource management: Gaining a competitive advantage." (2006).

[8] Sombataro, Monica. "Just-in-Time Learning." Computerworld. N.p., 03 Apr. 2000. Web.

[9] Arshavskiy, Marina. "Augmented Learning: An Argument For Its Use." ELearning Industry. N.p., 21 Apr. 2016. Web.

[10] Reardon, Marguerite. "Augmented Reality Comes to Mobile Phones." CNET.com. CNET, 29 Sept. 2010. Web.

[11] Besl, Paul J., and Ramesh C. Jain. "Three-dimensional object recognition." ACM Computing Surveys (CSUR) 17.1 (1985): 75-145.

[12] Marr, Bernard. "What Is Digital Twin Technology - And Why Is It So Important?" Forbes.com. Forbes, 06 Mar. 2017. Web.

[13] Geig, Mike. "Why You Should Be Using the Unity Game Engine." InformIT. Pearson, 04 Apr. 2013. Web.

[14] Hartley, Richard, and Andrew Zisserman. "Multiple view geometry in computer vision." Cambridge University Press, 2003.