AI-POWERED AUGMENTED REALITY TRAINING FOR METAL ADDITIVE MANUFACTURING

Donald Palomino[1,2], Shijie Bian[1], Pedro Salcedo[2], Erick Navarro[3], Bharath Gopalaiah[2], Richard Otis4, Bingbing Li[1,2]
[1]Autonomy Research Center for STEAHM, California State University, Northridge, CA 91324
[2]Department of Manufacturing Systems Engineering and Management, California State University, Northridge, CA 91330-8332
[3]SimInsights Inc, Irvine, CA 92618
[4]Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA 91109, USA

Abstract
Metal additive manufacturing (AM) training can cost a company the considerable amount of time, cost, and resources. To resolve this challenge, the NSF funded HyperSkill platform was applied to create an innovative, immersive training program that integrates AI-powered object detection and text recognition into a comprehensive digital twin of the metal AM machine operation. This immersive digital twin will support the delivery of just-in-time guidance to trainees while also monitoring their actions and providing contextual and personalized feedback to accelerate training, foster retention, and maximize transfer to the actual job. The augmented reality (AR) training supports the import of 3D assets, no-code authoring of workflows, standard operating procedures (SOP), step-by-step instructions, and delivery across a wide variety of AR devices. For this study, the specific metal AM operation is based on the Renishaw AM400 in both its full and reduced build volume (RBV) configuration.

Keywords: Augmented Reality Training; Metal Additive Manufacturing; Text Recognition; Object Detection

1. Introduction

Metal AM is a developing technology with huge potential in major industries like the aerospace, defense, and healthcare sectors. While relatively limited in build volume relative to traditional manufacturing, metal AM is capable of producing parts with intricate geometries and detail unobtainable by CNC machining and die casting. Additionally, unlike subtractive manufacturing, metal AM eliminates the need for expensive tooling and excessive waste. Metal AM is also able to print from a variety of different metals, including titanium which is notoriously known to be difficult to machine conventionally. From heat exchangers with complex patterns to biomedical micro-structures, metal AM is flexible and robust enough to fulfill the needs of any manufacturing facility. Due to these benefits, metal 3D printers have seen a sharp increase in market share in recent years with the global metal 3D printing market expected to grow from $378.0 million in 2019 to $738.8 million by 2025[1].

As the number of metal 3D printers proliferate, so too does the number of employees needed to operate these printers. Currently, the training process requires an individual experienced in the operation of the metal 3D printer to teach the trainee. This experienced individual is either a veteran operator at the company, or a professional trainer from the metal 3D printer's company requested by the trainee's company to come on-site. In both cases, this is a major cost as either the company will have to temporarily sacrifice the work productivity of the

veteran operator, or they will have to pay thousands of dollars for the on-site professional trainer. In addition to this, the training process can take weeks and requires the use of a metal 3D printer for demonstration and training purposes, leaving the printer out of commission for its intended use. These costs from time, money, and resources quickly add up to become a significant expense for companies in adopting metal AM. Lastly, this training process makes it inconvenient for companies to own multiple metal 3D printers from different manufacturers, as a veteran operator specializing in one printer would be unable to teach a trainee how to operate another printer, making companies resort to having to pay for an on-site professional trainer.

A system that is capable of standardizing the metal AM workflow in a comprehensive and easy to understand manner would greatly alleviate these associated costs for companies. The system would also give companies more flexibility in the different types and brands of metal 3D printers they can use.

In this study, an Artificial Intelligent (AI)-powered AR training simulation for metal AM was developed to simplify the training process and reduce its costs. The NSF funded Hyperskill platform will be applied to create an innovative, immersive training environment that will integrate AI-powered text recognition and object detection into a comprehensive digital twin of the metal AM processes. This immersive digital twin will support the delivery of just-in-time guidance to trainees while also monitoring their actions and providing contextual and personalized feedback to accelerate training, foster retention, and maximize transfer to the actual job. The AR training supports the import of 3D assets, no-code authoring of workflows, standard operating procedures, step-by-step instructions, and delivery across a wide variety of AR and VR devices.

For this case study, the digital twin will be created using Solidworks and Fusion 360 CAD software, the Renishaw AM400 metal 3D printer in both its full and RBV configuration will be used as the digital twin's real-world counterpart, and the HoloLens 2 will be used as the AR headset to immerse the trainee in the training environment. An RBV is a printer add-on that allows it to produce smaller builds relative to its full configuration. The smaller build volume takes up significantly less resources than the latter, requires less metal powder to print, and reduces the amount of time it takes to change between different metal powders.

1.1. Current Status of Metal AM, AR Training, and AI Systems

1.1.1. Metal AM

Despite the major advantages that metal AM offers, it currently suffers from a number of issues. One of these problems is the lack of a workforce with the proficiency and qualifications necessary to operate metal 3D printers[2]. This stems from the larger issue of a deficiency in the general education for AM. Universities and training programs have not kept up with the development of new AM technologies. Therefore, more training, education, and awareness needs to be accomplished for the current workforce and the new generation of students[3]. Progress towards this is already underway. For example, the Center for Innovative Materials Processing through Direct Digital Deposition at Penn State offers combined lecture and hands-on training in their AM facilities. However, they only trained 240 industry practitioners over three years due to

space constraints and safety concerns[4]. Additionally, several universities and companies including Virginia Polytechnic Institute, Virginia Tech, University of Texas Austin, Society of Manufacturing Engineers, and Stratasys are implementing courses and training specializing in AM[5]. However, despite these efforts, it is not fast enough to meet the growing demand of metal AM. They also only cover AM in a broad scope, not the specific operation of metal 3D printers. The need for a cost-effective and versatile training program for metal AM still exists.

1.1.2. AR Training

AR technology has proven itself to be effective in providing more capabilities to workers and in training. Through case studies performed by Microsoft on their HoloLens2, the same AR headset used in this study, it was found that a 177% return on investment can be achieved by implementing the AR headset into their manufacturing process[6]. AR technology has been used in the automotive industry to assist drivers in replacing a flat tire by tracking the car wheel and relevant tools[7]. Based on the work done in implementing AR technology to a training simulation for an industrial setting, it was found that tasks asked of the trainees must be of a high enough complexity to make the AR system feel justified. Additionally, "selling" the benefits of the AR system to the trainees in its improved work efficiency and build-in error control is important in the trainee's reception to it[8]. An overall advantage in using AR technology in training is the integration of multiple digital formats and capabilities into one function process. Even complex assemblies can be broken down into easy to follow steps that an inexperienced employee can follow[9].

1.1.3. AI Systems

AI systems are capable of recognizing both body and hand gestures of a human. These gestures can be used as a means of communication between the human and computer[10]. This can eliminate the need for peripheral devices like a mouse and keyboard which limit a user to the desk on which they are placed. Implementing AI systems into AR or VR simulations allows the user to work in a virtual environment while being able to freely stand and walk around, not being constrained by fixed input devices. AI-powered object detection can be used in developing a real-time system to monitor machine states and perform automatic fault detection on 3D printers[11]. Using character-region awareness for text detection (CRAFT)[12], AI systems can also identify and recognize the text and finger positions of a user operating a 3D printer with a control panel. This is done by processing images recorded on a wearable camera[13].

2. AI Algorithm

2.1. Text Recognition

2.1.1. Contour Detection

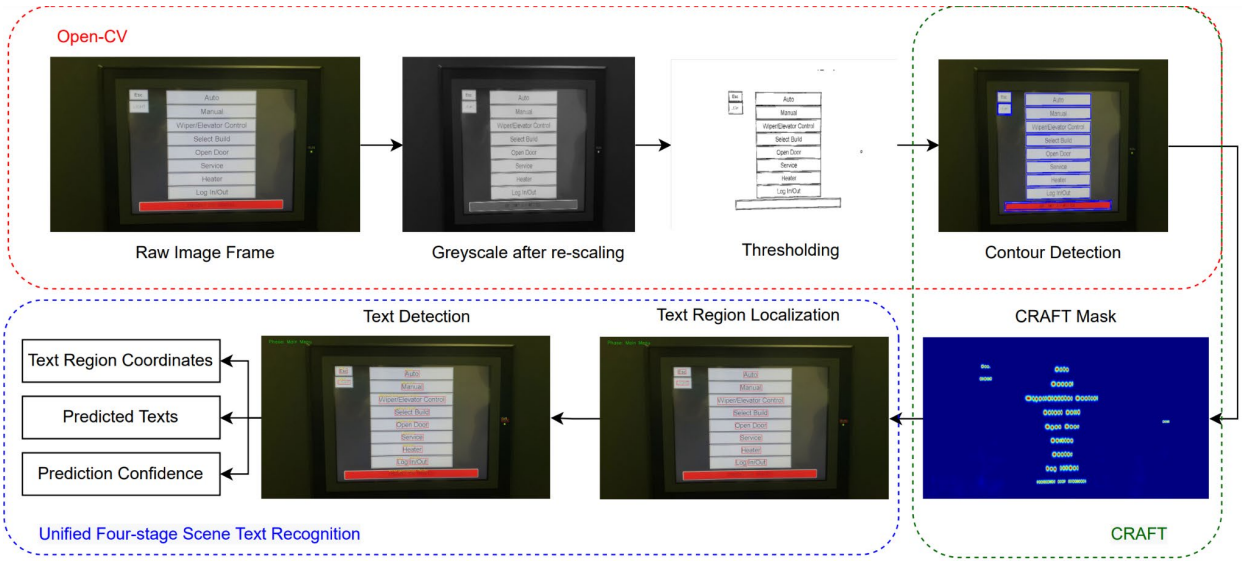

Figure 1: Flowchart for contour recognition

Before the substantial step of text recognition, preprocessing procedures are required to localize the text regions to narrow down the search space and discard the noisy background contexts. Considering that a majority of text displayed on the screen panel are bounded within a colored region, a contour detection step is performed to segregate them. As shown in **Figure 1**, the input image is first transformed into its grayscale representation after being re-scaled to a suitable resolution, followed by a binary pixel threshold of RGB value (0, 255, 150) and a neighborhood-oriented adaptive threshold screening to find contours. Utilizing OpenCV's functionalities of calculating contour area and perimeter, the contours that map to a closed region are filtered out and considered the regions of interest for the subsequent text detection step.

2.1.2. Text Region Localization

Character region awareness for text detection (CRAFT), a machine learning model that supports the localization of characters and phrases, is adopted to restrict the search space for text recognition further. Specifically, the regions of interest obtained from the previous step are passed through a pre-trained CRAFT model, which localizes individual characters within the said region by evaluating the probability that a given pixel is the center of the character. Subsequently, CRAFT merges localized character regions into instances by inference of the probability of adjacent characters belonging to the same word. The output of CRAFT consists of the bounding box coordinates of localized text regions, which are then passed to the final text recognition module. Note that the contour-detecting preprocessing step that narrows down the regions of interest, together with CRAFT which empowers accurate detection of text regions, allows the proposed framework to function effectively in diverse working environments.

2.1.3. Recognition Algorithm

The localized text regions of interest, as represented by bounding box coordinates, are passed through a unified four-stage scene text recognition framework, which includes a thin-plate spline (TPS) for feature transformation, a pre-trained residual network (ResNet) for feature extraction, a bi-directional long short-term memory (BLSTM) for sequence modeling, and an attention mechanism for prediction generation. The output from the recognition framework consists of the predicted texts for each region of interest, along with the corresponding prediction confidence score. The predicted texts are used in subsequent steps for the real-time identification of the machine's operation phase and the instructions as displayed on the output screen.

2.2. Object Detection

One traditional approach for detecting the operator's interaction with the machine would be to use automated finger detection models to locate the position of the operator's fingers on the interactive panels. However, directly adopting and integrating a finger detection model into the proposed framework will likely render the framework cumbersome, which may affect the real-time performance during work-scenario testing. Therefore, a color-change method is devised to utilize the localized text regions of interest together with the detected closed contours as generated from the previous text recognition section. Specifically, for each detected closed contour region, the averaged pixel color is calculated and compared with a pre-defined metric. If a finger intrudes the region, the algorithm would be able to automatically detect the anomaly in region colors, thereby identifying the location of the finger.

3. Creation of AR Training

3.1. Creation of Digital Twin using CAD Software

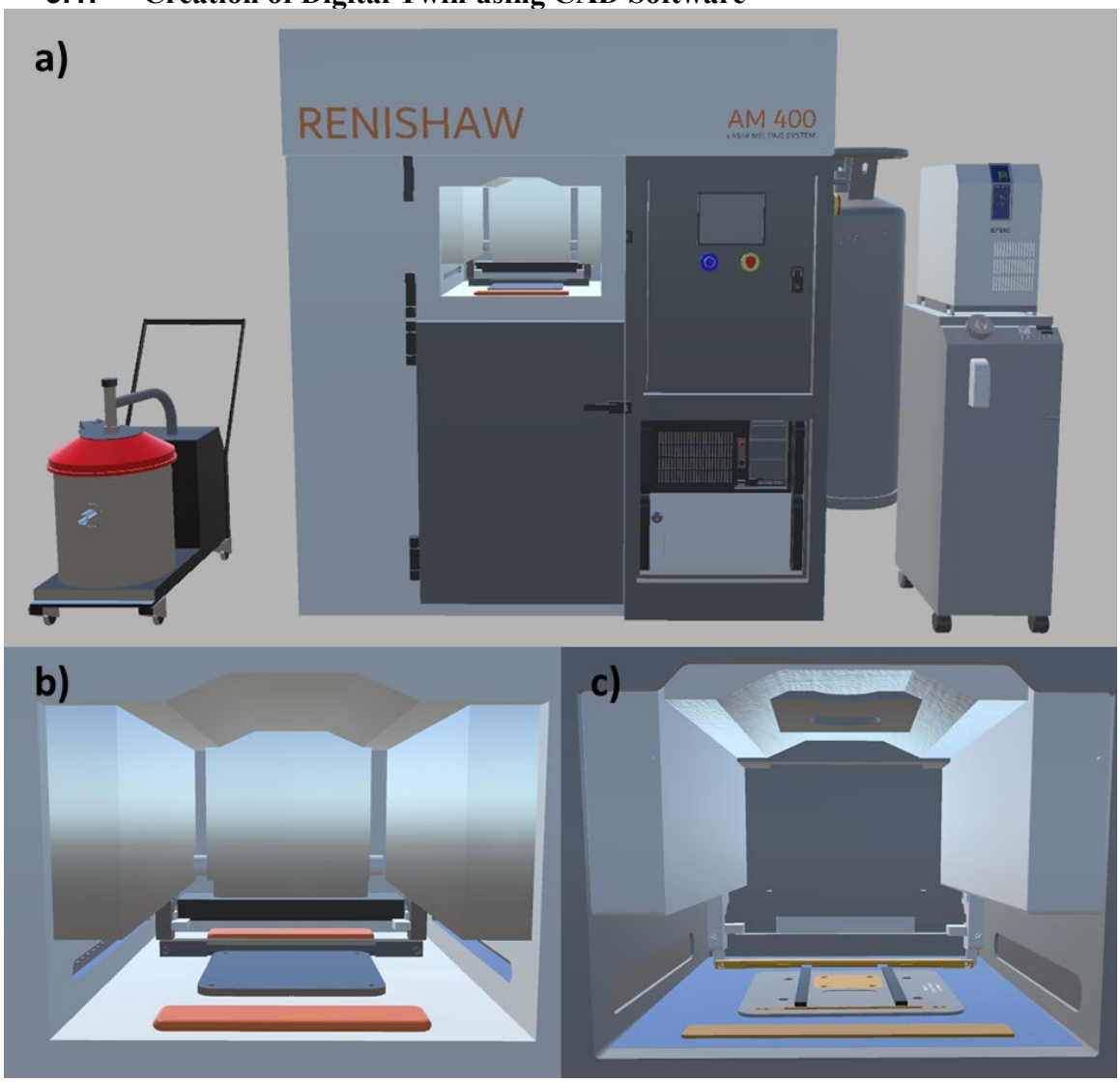

Figure 2: a) Renishaw AM400 with accompanying hardware, **b)** Build area in full configuration, **c)** Build area in RBV configuration

3.1.1. Full Configuration

The first step in developing the AR training simulation is to use computer aided design (CAD) modeling to create a digital twin of the metal 3D printer. All components of the metal 3D printer relevant and necessary to its manual operation were created, as well as components for facilitating ease in part recognition. Parts hidden behind panels such as electronics and other hardware components used in machine maintenance were not created.

To begin the CAD design process using Solidworks, components that are critical to the operation of the metal 3D printer had their measurements taken manually with calipers and

measuring tape. This was done to represent the digital twin as close as possible to its real-world counterpart. This includes the metal 3D printer's build area and its internal components such as the build plate, the wiper, and the wiper adjustment knobs. Unique to the full configuration is the build plate which is a large, flat, and solid piece of metal. Some of the non-critical parts had their measurements approximated as they are not required for the general operation of the metal 3D printer and only serve as visual representations to aid in the learning experience. An example of this was the computer as not every feature it included was necessary for operation, only the on/off switch. The safe-change filter area is another example of this since although many components are visible within the area, most are not needed in general operation, such as the pumps, generators, and compressors. The metal 3D printer also requires the following accompanying hardware in its operation process: the chiller, the dryer, the immersion separation vacuum, and the argon tank. As such, these components were also made into CAD models. The full digital twin model of the Renishaw AM400 and its accompanying equipment is shown in **Figure 2a)**, and a close up of its build area in full configuration is shown in **Figure 2b)**.

A key part of the AR training simulation is highlighting parts relevant to the current step. To accomplish this, all interactable components like the knobs, handles, and buttons were created as separate parts using Solidworks, then added to their respective CAD models as an assembly. Once the CAD assembly was finished, the model was converted into a file that was compatible with HyperSkill. This was done by using an add-in called Extended Reality (XR) Exporter in Solidworks. This converted the default .SLDPRT files into the .glb (extended reality) files required by HyperSkill. These .glb files were then imported into HyperSkill to complete the digital twin of the Renishaw AM400 in its full configuration.

3.1.2. RBV Configuration

To create the digital twin of the metal 3D printer in its RBV configuration, a similar process to its full configuration is performed. All components of the metal 3D printer relevant and necessary to its manual operation were created, as well as components for facilitating ease in part recognition. Parts hidden behind panels such as electronics and other hardware components used in machine maintenance were not created.

Unlike the full configuration digital twin process, Fusion 360 is the software used to create the CAD models for the RBV configuration. Similar to the full configuration process, all components that are critical to the operation of the metal 3D printer had their measurements taken manually with calipers and measuring tape. This includes the metal 3D printer's build area and its internal components such as the RBV unit, the wiper, and the wiper adjustment knobs. Unlike the full configuration's build plate, which was large, flat, and solid piece of metal, the RBV unit is much more complex and consists of several components including the RBV recoater, cover, adjustment blocks, top plate, cover plate, build plate, and powder compacting tool. Similar to the previous process, some of the non-critical parts had their measurements approximated as they are not required for the general operation of the metal 3D printer and only serve as visual representations to aid in the learning experience. The following accompanying hardware to the metal 3D printer were also made as CAD models using Fusion 360: the chiller, the dryer, the immersion separation vacuum, and the argon tank. The full digital twin model of the RBV configuration is identical to the full configuration with the exception of the build area,

which features an RBV build plate as opposed to a full build plate. A close up of its build area in RBV configuration is shown in **Figure 2c)**.

Similar to the full configuration process, all interactable components were created as separate parts using Fusion 360, then added to their respective CAD models as an assembly. Once the CAD assembly was finished, an add-in for Fusion 360 was used to convert the default .F3D files into the .glb files required by Hyperskill. These .glb files were then imported into Hyperskill to complete the digital twin of the Renishaw AM400 in its RBV configuration.

3.2. Training Procedure and Flowchart

3.2.1. Full Configuration

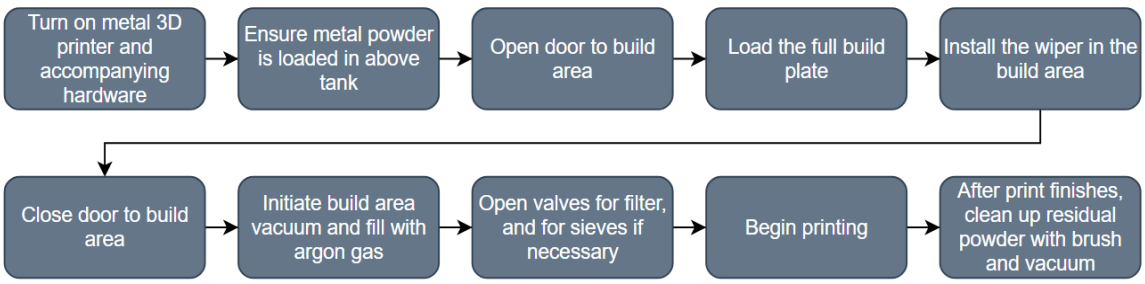

Figure 3: High-level flowchart for full configuration

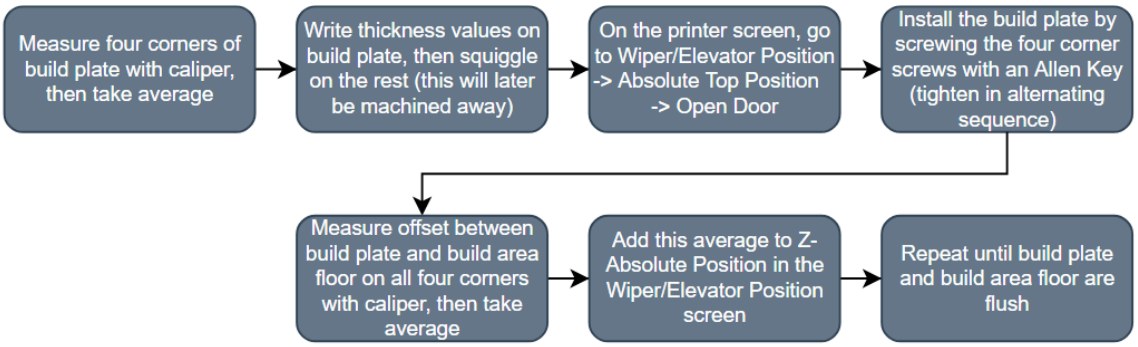

Figure 4: Low-level flowchart for full configuration detailing loading the full build plate

It is critical to the training simulation's effectiveness that the training procedure for operating the metal 3D printer in its full configuration is as comprehensive and easy to follow as possible. To create the training procedure, it is necessary to be trained by an experienced operator. During this training, each step is documented in detail and video of the process is captured for reference. From these notes, two step-by-step procedures can be created: one high-level and one low-level. The high-level procedure lists 10 generalized steps, as shown in **Figure 3**, while the low-level procedure breaks each of these steps into many more and in far greater detail. A segment of the low-level procedure is shown in **Figure 4**. This will then be reviewed by the experienced operator to ensure that all of the steps were documented correctly and that the steps were in the proper order. Once the training procedure has been verified, it can be implemented into HyperSkill.

3.2.2. RBV Configuration

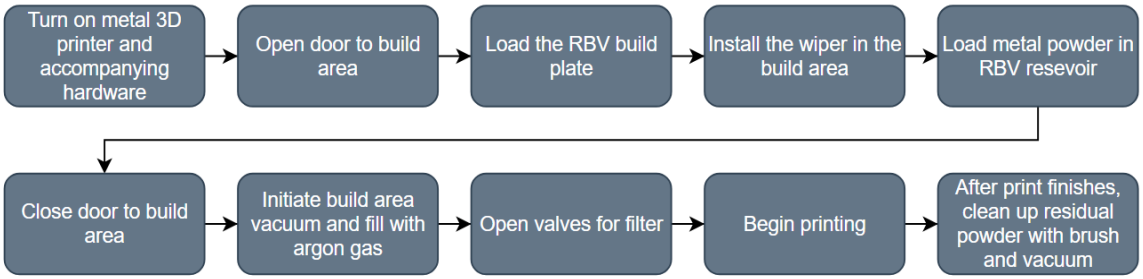

Figure 5: High-level flowchart for RBV configuration

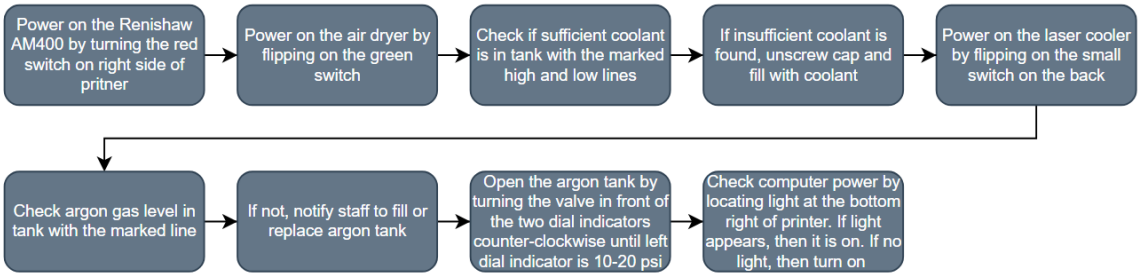

Figure 6: Low-level flowchart for RBV configuration detailing turning on all the hardware

Similar to the creation of the training procedure for the metal 3D printer's full configuration, it is also critical that the RBV configuration procedure is comprehensive and easy to follow. As an experienced operator goes through the operation, each step is documented in detail and video of the process is captured for reference. Two step-by-step procedures are then created based on these notes: one high-level and one low-level. While both the full and RBV configuration procedures are similar to each other, they differ greatly in loading the build plate and metal powder. The high-level procedure for the RBV configuration is shown in **Figure 5**, while a segment of its low-level procedure is shown in **Figure 6**. This will then be reviewed by the experienced operator to ensure that all of the steps were documented correctly and that the steps were in the proper order. Once the training procedure has been verified, it can be implemented into HyperSkill.

3.3. HyperSkill Procedure

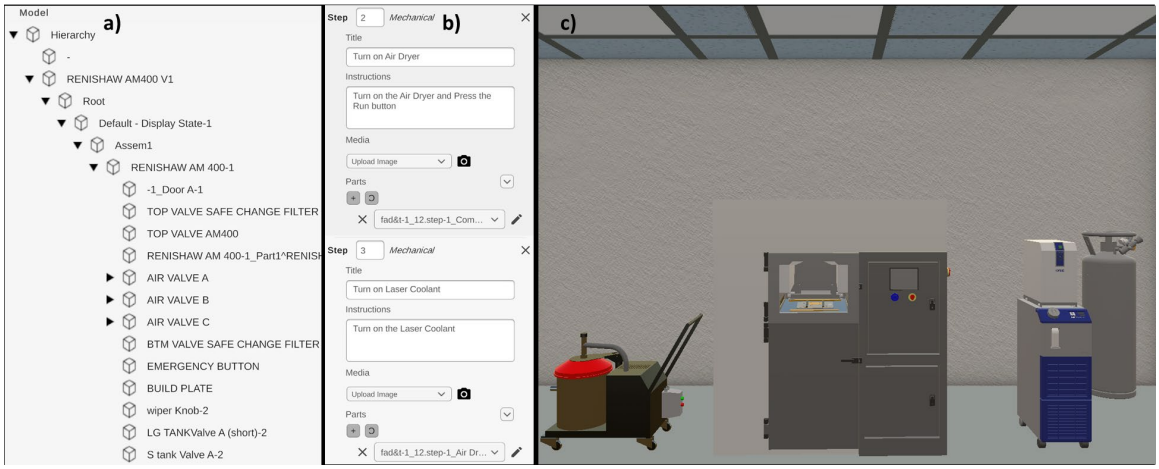

Figure 7: a) Hierarchy tree within HyperSkill, **b)** Assembly tab, **c)** Simulation with digital twin

HyperSkill, developed by SimInsights, is a no-code 3D simulation software for both AR and VR purposes. HyperSkill enables users with little to no programming experience to create immersive training content. It also enables non-programmers to author AR and VR content and to collect and visualize experience data. It features a cloud service to store interactive virtual objects converted from existing CAD and 3D assets.

After the converted .glb files of the metal 3D printer's digital twin are imported into HyperSkill, they are then opened within the Virtual Objects tab of the software. To the left side of the panel is the hierarchy tree, as shown in **Figure 7a)**. When selecting a feature in the hierarchy tree, its corresponding object or surface in the digital twin will be selected and turned bold. The features within the hierarchy tree can also be renamed for convenience when assigning them to steps in the training procedure.

On the right side of the panel is the assembly tab, as shown in **Figure 7b)**. The assembly tab is where the step-by-step instructions for the training simulation are written. Available options for defining the assembly are mechanical, wiring, and snapshot. For this study, only the mechanical feature will be used. Each step added to the assembly tab is numbered and kept in numerical order. When creating a new step, first the title and instructions must be added. There is an option to add media, such as images or videos, that can be viewed during the training simulation to aid the trainee. Lastly, features within the hierarchy tree that are used in the current step are selected. These selected features correspond to its respective object in the digital twin and will highlight it during the step within the training simulation. This helps to direct the trainee's attention to areas of significance regarding their current step. Consider adding an example as a figure.

Once all the steps necessary to the operation of the metal 3D printer have been completed, the Virtual Objects tab can be exited and the Simulations tab is selected. A new simulation is created by giving it a name and choosing a premade scenario as the virtual environment. This name is important as it will be used to later call the training simulation within the AR headset. Lastly, the virtual object with the .glb file and step-by-step instructions are

simply dropped into the simulation as shown in **Figure 7c)**. The training simulation is now ready to be used within the AR headset.

3.4. Implementation of AR Headset

Figure 8: Digital twin of Renishaw AM400 being scaled down to fit user environment

After the digital twin is created and imported into the HyperSkill simulation, it is ready to be imported into the AR headset. First, the HyperSkill app must be installed. Because the software is considered 3rd party and not affiliated with Microsoft, the manufacturer of HoloLens 2, HyperSkill must be manually downloaded. This is done by first connecting the AR headset to a computer via the AR headset's personal IP address. The device then needs to be put into developer mode, found within the onboard settings of the HoloLens 2. Being in developer mode allows the user to add 3rd party software and apps to the device. The HyperSkill app is then downloaded from the connected computer to the AR headset. Once it has been downloaded, the computer may be disconnected and the HoloLens 2 can then be worn. Within the menu of the AR headset, the HyperSkill app can be opened, which then prompts the user to select the simulation they would like to run. After selecting a simulation, the training commences with the digital twin appearing in a 1:1 ratio. Depending on the size of the user's environment, they may wish to scale the digital twin down to a more convenient size, as can be seen in **Figure 8**. Once an appropriate scale is set and its positioned anchored, the training simulation will begin.

3.5. Integration of AI-Powered Text Recognition and Object Detection

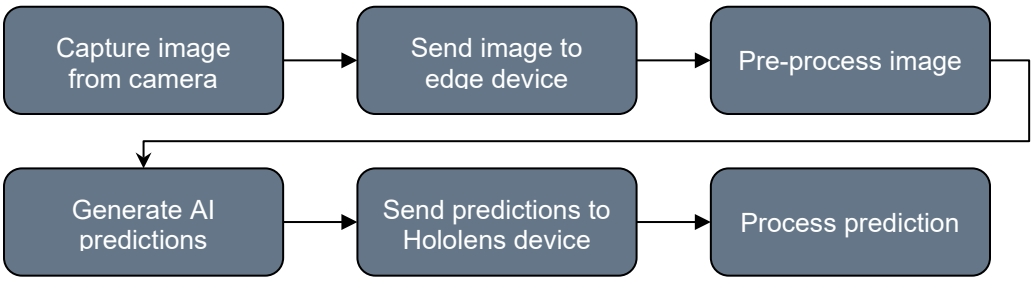

Figure 9: High-level flowchart of prediction process

The AI system will interface with the HyperSkill HoloLens app through a representational state transfer application programming interface (REST API). The REST API will be deployed on a local edge device using a Python Flask web application. We use a Flask application instead of deploying the AI system on the Hololens due to the limited battery and processing power of the Hololens. Deploying the AI system on an edge device further simplifies the process of validating the AI performance and accuracy. Should there be a need, the Flask application can also be deployed in the cloud at scale.

One the AI system is deployed on the edge device, the HoloLens application will be configured to communicate with the flask web app. After a certain set of frames or at the request of the Hololens user, HyperSkill will capture an image from the front-facing Hololens camera. The image will be sent to the flask web app through the REST API. The web app will then process the image and feed it into the AI system. The AI system will then process and image and return the corresponding predictions. The predictions are sent back to the Hololens device for further processing. A high-level flowchart of the prediction process of the AI system once integrated into the HyperSkill HoloLens app is shown in **Figure 9.**

4. Discussion

4.1. Accuracy of AI-Powered Text Recognition

4.1.1. Baseline: all lights on scenario

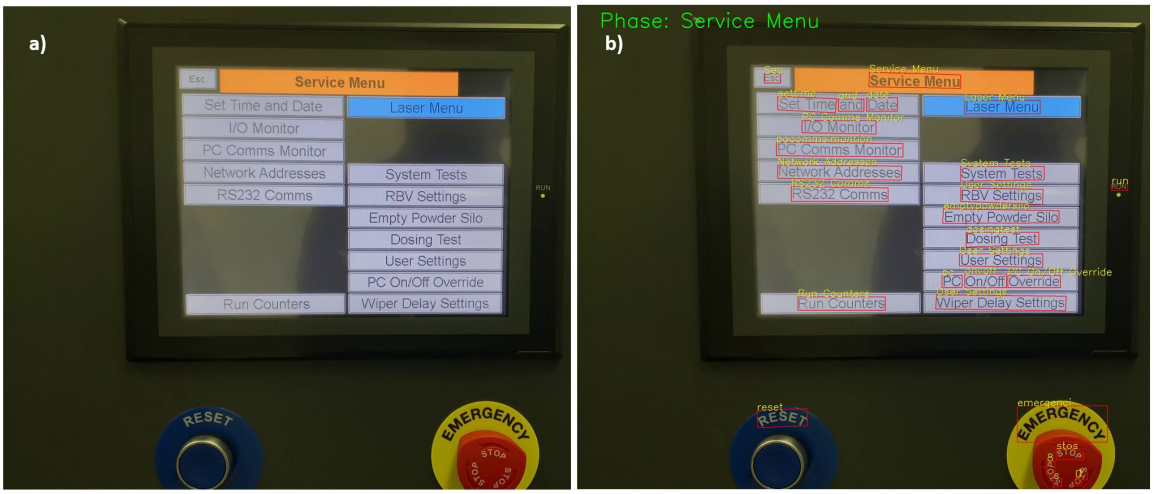

Figure 10: Baseline condition with all lights on; **a)** Raw image frame, **b)** Image frame with predicted phase and text

The accuracy of the text recognition for the AI system is an extremely important metric in creating a smooth, immersive experience for the trainee. It allows for the training simulation to recognize the screen and phase that the trainee is on using text recognition. If poor accuracy by the AI system occurs, this may result in frustration over the AR training simulation which would be detrimental to its effectiveness. Therefore, it is vital to the project that text recognition is as accurate as possible.

To test the accuracy of the text recognition, a video of all available phases for the Renishaw AM400 was taken. To ensure the screen panel, as well as the reset button and emergency off button were in frame at all times, a tripod was used for stabilization. A user pressed buttons on the screen to go through each phase used during normal operation of the printer. The user also paused for a few seconds in between each button press to allow an unobstructed view of the screen for the AI system to later process. To set a baseline for future testing scenarios, all lights in the room were turned on and the operator did not wear any gloves during filming. Once all screens and menus were captured on video, it was separated by individual frames. These images were then manually processed to remove images that were blurry, showed a blank screen during phase transition, or had the user's hand blocking the screen. Images with the user's hand in the frame were relocated to a separate folder for future testing of object detection. The remaining images were then processed by the AI system to recognize the current phase and all text on the screen, as shown in **Figure 10**. Successful recognition is identified when a red box is formed around text. Once all unobstructed frames of the video have been processed, manual verification by a user was then performed to compare the results by the AI system to the ground truth.

Once the accuracy of the AI system's ability to recognize text was verified to be acceptable under nominal conditions, it is then tested under different working conditions and scenarios. These external factors in the environment can affect the AI system's performance during the operation of the AR simulation. Such factors include low-light conditions in the training room or area and harsh light reflecting off the metal 3D printer screen.

4.1.2. Low-light Scenario

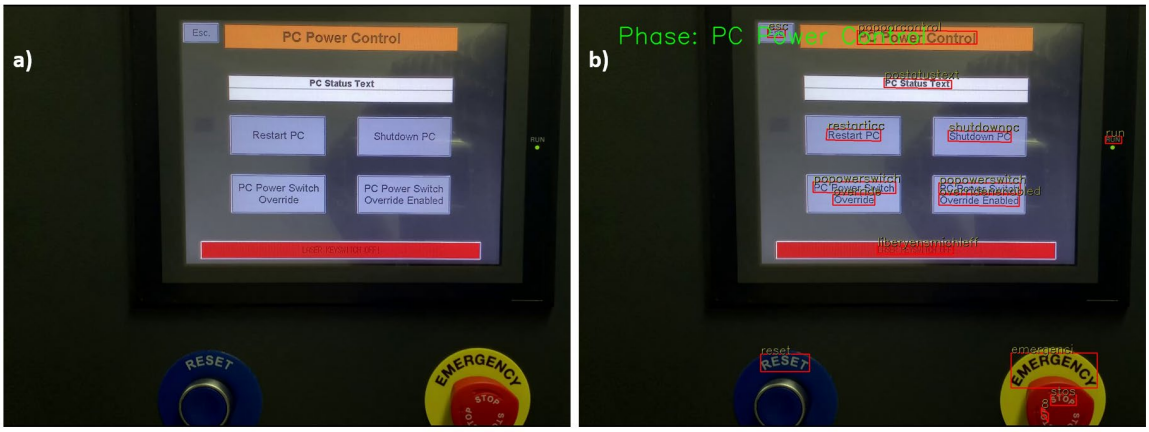

Figure 11: Condition with half-lights on; **a)** Raw image file, **b)** Image frame with predicted phase and text

Low-light conditions may affect the ability for the AI system to recognize text. While most factory environments are required to have a sufficient amount of footcandles determined by their local law, shadows from equipment or structures may be cast on the area of the metal 3D printer. Additionally, in more rare instances, the metal 3D printer may be located in a room with below-regulation lighting conditions. In both cases, low-light conditions are simulated to gauge the effect it has on the AI system.

To test low-light conditions, a video of all available phases for the Renishaw AM400 was taken. Similar to the baseline test, a tripod is used to capture the screen, reset button, and emergency off button. However, unlike the baseline test which had all lights in the room on, this test had only half of the lights on. Once all screens and menus were captured on video, it was separated by individual frames based on the screen shown. These images were then manually processed to remove images that were blurry, showed a blank screen during phase transition, or had the user's hand blocking the screen. Images with the user's hand in the frame were relocated to a separate folder for future testing of object detection. The remaining images were then processed by the AI system to recognize the current phase and all text on the screen, as shown in **Figure 11**. Successful recognition is identified when a red box is formed around text. Once all unobstructed frames of the video have been processed, manual verification by a user was then performed to compare the results by the AI system to the ground truth.

4.1.3. Screen Reflection Scenario

Figure 12: Condition with harsh, direct light; **a)** Raw image file, **b)** Image frame with predicted phase and text

The screen of the metal 3D printer is highly susceptible to reflection as it has no anti-glare film or treatment. While the reflection of the user does not interfere with the manual operation of the printer, it may be an issue for the AI system's ability to detect the trainee's hand or recognize text. The screen of the metal 3D printer does not come with an anti-glare film, making it susceptible to reflection. Both excessive lighting and the operator can be reflected off the screen, adding noise and interference to the AI system's image processing.

To test the effect of harsh reflection off of the screen, a video of all available phases for the Renishaw AM400 was taken. All lights in the room were kept on, as well as two bright flashlights pointed directly towards the screen. While the video recorded, the flashlights would slowly move around the screen, making sure to go over the text. Once all screens and menus were captured on video, it was separated by individual frames based on the screen shown. These images were then manually processed to remove images that were blurry, showed a blank screen during phase transition, or had the user's hand blocking the screen. Images with the user's hand in the frame were relocated to a separate folder for future testing of object detection. The remaining images were then processed by the AI system to recognize the current phase and all text on the screen, as shown in **Figure 12**. Successful recognition is identified when a red box is formed around text. Once all unobstructed frames of the video have been processed, manual verification by a user was then performed to compare the results by the AI system to the ground truth.

4.1.4. Scenario Test Results

Table 1: Text recognition accuracy results for test scenarios

Test Scenario	Phase Identification Accuracy	Text Recognition Accuracy	Text Recognition Confidence
Baseline (All lights on)	100%	94.24%	0.7180% (± 0.086%)
Half-lights on	88.83%	89.94%	0.7098% (± 0.091%)
Reflection	91.43%	91.37%	0.6923% (± 0.069%)

Recorded in **Table 1** are the test results for the text recognition module. Note that all the numerical results are recorded in the format of average ± standard deviation. Specifically, three test scenarios are adopted to simulate the diverse working conditions of the machine: the baseline scenario in which the camera is fixed and all lights in the room are on; a low-light scenario in which the camera is fixed and only half the lights in the rooms are on; and a screen reflection scenario where harsh, direct light flow moves around the screen causing glare. For each scenario, numerous image frames are extracted from the captured test videos and then run through the text recognition framework to generate text predictions. The predicted texts for each image frame are utilized to identify the operation phase that the machine is in, and the final phase identification accuracy for each scenario is calculated as the proportion of correctly predicted phases over the total number of image frames in that scenario. For each frame of each scenario, the proportion of correctly predicted texts of interest over the total number of texts of interest is calculated to be the text recognition accuracy of that specific frame, which is averaged together with the recognition accuracy of all other frames to be the final accuracy for that scenario. The confidence score is obtained in a similar procedure.

The test results show that the baseline scenario's averaged performances for phase and text recognition are the most satisfactory. Specifically, when abundant light is provided, the image as captured by the camera shows the most apparent contour distinctions, serving as a premise for subsequent detection and recognition steps. Since most real-life working scenarios support normal lighting conditions, the test results assure the effectiveness of the proposed design. Performances under insufficient or excessive illumination are less satisfactory. When the light intensity is low, text that is close together and bound inside dark-colored display regions tend to be particularly challenging to correctly identify due to the difficulty for the CRAFT model to localize the regions of interest. Likewise, when harsh lights are on, significant reflections on the display screen will affect the text region localization step as performed by CRAFT, thereby reducing the accuracy and confidence of text and phase recognition. Therefore, these working scenarios should be avoided if possible. The confidence scores follow a similar pattern. In the baseline scenario, where the camera's movement is minimal and the lights are abundant, the text prediction confidence is the highest. In comparison, the scenarios in which only half of the lights are provided reflect less confidence in the text detection results. Harsh lighting conditions that result in strong reflections also restrict the text detection performances, which is something to avoid or ameliorate.

4.2. Validation of Integration

The complete AR training simulation comprises both hardware and software including: the digital twin, the AR headset, and the Hyperskill app. It is essential that all elements integrate in a cohesive way for the best user experience. Before integration is performed, each system was individually tested to ensure that they were performing as intended. Once each system is verified, a full run of the training simulation was performed.

To verify that the digital twin is functional, all interactable components that are referenced in the training procedure, like the knobs, handles, and buttons, are checked within the digital twin assembly. Additionally, all non-interactable components, like the body of the metal 3D printer, are checked to make sure that it visually resembles the real-world printer to aid in the trainee's experience. This was done by opening the hierarchy tree within HyperSkill's Virtual Object tab, then going through the individual parts with the training procedure to ensure all used components have been made and are included.

To verify that the AR headset is operational, it first must be able to turn on and recognize basic gestures. These include gestures to open the main menu and to select buttons close up and far away. If the AR headset is unable to recognize any one of these gestures, then it will not be capable of running the AR training simulation. As an additional measure, the calibration app will also be used to adjust the AR headset to the user's eyes. These are basic features of the AR headset and should be highly reliable in its standard operation.

To verify that the HyperSkill app is working properly, the digital twin is first checked to see that it is successfully able to be dropped into the HyperSkill's virtual simulation on a computer. Once this is achieved, the HyperSkill app on the AR headset is checked for its ability to open correctly. Lastly, the training simulation is selected and its ability to progress through its step-by-step instructions while highlighting relevant components is checked. This procedure requires the verification of the previous systems and results in the final integration of one system for the AR training simulation. After the simulation has been run through in its entirety with no errors or bugs, testing with volunteers can be conducted to gauge reaction and learning effectiveness.

4.3. Validation of AR Training Effectiveness

After the training simulation and its individual components are validated, its effectiveness must also be validated. To accomplish this, the Kirkpatrick evaluation model was used. Normally this is a four step process, however as the last two steps pertain to evaluating results over a long period of time, only the first two steps will be utilized for this project. This first step is Reaction: to what degree do participants react favorably to the learning event. The second step is Learning: to what degree do participants acquire the intended knowledge, skills, attitude, confidence, and commitment based on their participation in the training[14]. For this study, two volunteers participated in the training simulation. These volunteers were chosen based on having an engineering background with no prior experience in operating a metal 3D printer or using an AR headset.

Table 2: Results of Reaction Survey After Completion of AR Training Simulation

Questions (scored 1-5; 1=strongly disagree, 5=strongly agree)	Volunteer 1	Volunteer 2
Training simulation was enjoyable	4	5
AR technology is exciting	5	5
AR finger and hand gesturing is intuitive	4	4
Step-by-step instructions were easy to follow	5	5
Teaching style is time-effective	4	5

For this first step, Reaction, a proctor administered the training simulation to the participant and assisted when necessary. Prior to beginning the simulation, a tutorial by the AR headset is taken to familiarize the volunteer with how to interact in the augmented environment and with finger and hand gestures for navigation. An eye calibration was also performed to ensure the headset was optimized for each volunteer. Afterwards, the AR training simulation was given to the volunteers. Upon completion of the simulation, which took around 30 minutes, a survey was conducted to gauge the degree to which the participant found the training favorable and engaging. The results of the survey for both volunteers are shown in **Table 2**. As can be seen, both volunteers found the training simulation to be enjoyable, intuitive, easy to follow, and time-effective. Verbal feedback was positive and mostly praised the novelty of AR technology and its potential.

For the second step, Learning, a short quiz was given to the volunteers before and after the training simulation to gauge their retention of the information. All of the questions were simple in nature, mainly locating points of interest within the metal 3D printer. However, the initial test scores were low, with one volunteer scoring a 0/5 and the other scoring a 3/5. This shows the complexity of the printer, especially for someone inexperienced in its operation. After the AR training simulation was completed, the volunteers took the same quiz again, however this time the volunteers scored a 4/5 and a 5/5 respectively. Therefore, the simulation indeed improved the volunteers' knowledge of the printer's operation.

5. Conclusion and Future Work

The rate at which companies invest in metal AM far exceeds the existing education and awareness for a capable workforce to operate metal 3D printers. Conventional methods in hiring new operators involve intensive, long, and expensive training, costing these companies noticeable expenditures. To solve this problem, an AI-powered AR training simulation for metal AM was developed to simplify the training process, reduce its associated costs, and reduce its time frame. This was achieved by using CAD software to 3D model a digital twin of a metal 3D printer, importing it into HyperSkill with the printer's training procedure, then running the training simulation using an AR headset. Leveraging advanced machine learning models, an end-to-end workflow for the automated processing of images mimicking the operator's vision is developed. This workflow utilizes contour detection and character-region awareness for the

localization of regions of interest, enhancing the subsequent modules' robustness in removing the unnecessary background details within captured images. Subsequently, a unified four-stage scene text recognition module is adapted to perform text recognition on the previously filtered regions of interest. Utilizing the detected texts, the AI system integrated into the application is capable of recognizing the machine's current phase and the text displayed on the machine's screen panel, all done in real-time and automation, without any human-labored processing. Test results obtained from numerous scenario settings reflect the competent performance of the proposed methodology to function in diverse environments. Feedback using the Kirkpatrick evaluation model showed that volunteer reaction to the AR training simulation was positive and excited at the prospect of AR technology. It also showed that the volunteers' knowledge test scores improved significantly, a testament to the learning efficiency of the training simulation.

Future work may be done to further enhance the robustness and accuracy of text detection by integrating rectification through similarity search with the aid of the predicted screen phase. Specifically, a data set of mappings between the phases to the ground-truths texts displayed during that phase should be constructed. After obtaining the text predictions through real-time inference, the results are compared to the possible ground-truth texts recorded in the dataset as mapped from the detected phase, and the most similar ground truths are retrieved, which may serve as the rectified results of the predicted texts. Another aspect of future work would be combining the screen phase and text detection with object and finger detection, which empowers the algorithm to fully supervise and analyze human-machine interactions. If realized, the AI system for real-time automated inference can be integrated with existing AR technologies for vision capturing, thereby producing AI-powered systems to serve as an auxiliary unit to support the operators' operations of complex devices. Additionally, these capabilities will be expanded to not only include screen phase and text recognition, but also recognize user interaction with the following areas of the printer: filter area, build area door, glove door, laser chiller, dryer, argon tank, and vacuum cleaner. Once implemented, the AR training will be capable of identifying and verifying the actions of a user with any metal 3D printer.

6. Acknowledgements

This research was supported by the U.S. NASA MUREP Institutional Research Opportunity (MIRO) program (80NSSC19M0200), NASA MUREP High Volume (80NSSC22M0132), and partially carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration (80NM0018D0004).

7. References

[1] "Metal 3D Printing Market Accounted for $378.0 Million in 2019 and Is Expected to Be $738.8 Million by 2025: Global Metal 3D Printing Market Anticipated to Reach $738. 8 Million by 2025. Market Report Coverage - Metal 3D Printing. Market Segmentation." *NASDAQ OMX's News Release Distribution Channel*, NASDAQ OMX Corporate Solutions, Inc, 2020.

[2] Vafadar, Ana, et al. "Advances in Metal Additive Manufacturing: A Review of Common Processes, Industrial Applications, and Current Challenges." *Applied Sciences*, vol. 11, no. 3, MDPI AG, 2021, pp. 1–33, https://doi.org/10.3390/app11031213.

[3] Pei, E.; Loh, G.H. Future challenges in functionally graded additive manufacturing. In *Additive Manufacturing—Developments in Training and Education*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 219–228.
[4] Simpson, T.W.; Williams, C.B.; Hripko, M. Preparing industry for additive manufacturing and its applications: Summary & recommendations from a National Science Foundation workshop. *Addit. Manuf.* 2017, *13*, 166–178.
[5] Go, J.; Hart, A.J. A framework for teaching the fundamentals of additive manufacturing and enabling rapid innovation. *Addit. Manuf.* 2016, *10*, 76–87.
[6] Brown, Benjamin, et al. The Total Economic Impact™ of Mixed Reality Using Microsoft HoloLens 2, Microsoft, https://tools.totaleconomicimpact.com/go/microsoft/HoloLens2/.
[7] J. Lima, R. Roberto, F. Simoes, M. Alemida, L. Figueiredo, J. Teixeria, and V. Teichrieb, "Markerless tracking system for augmented reality in the automobile industry," Expert systems with applications, vol. 82, pp. 100-114, 2017.
[8] A. Syberfeldt, O. Danielsson, M. Holm, and L. Wang, "Visual assembling guidance using augmented reality," Procedia Manufacturing, vol. 1, pp. 98-109, 2015.
[9] Osborne, Mark, and Scott Mavers. "Integrating Augmented Reality in Training and Industrial Applications." 2019 Eighth International Conference on Educational Innovation through Technology (EITT), IEEE, 2019, pp. 142–46, https://doi.org/10.1109/EITT.2019.00035.
[10] Chen, Zhi-hua, et al. "Real-Time Hand Gesture Recognition Using Finger Segmentation." *TheScientificWorld*, vol. 2014, Hindawi Publishing Corporation, 2014, pp. 267872–79, https://doi.org/10.1155/2014/267872.
[11] Bian, S., Lin, T., Li, C., Fu, Y., Jiang, M., Wu, T., Hang, X., Li, B., 2021b. Real-time object detection for smart connected worker in 3d printing. In: Paszynski, M., Kranzlmüller, D., Krzhizhanovskaya, V. V., Dongarra, J. J., Sloot, P. M. (Eds.), Computational Science – ICCS 2021. Springer International Publishing, Cham, pp. 554–567.
[12] Baek Y, Lee B, Han D, Yun S, Lee H. Character region awareness for text detection. 2019. arXiv:1904.01941.
[13] Bian, Shijie, et al. "Machine Learning-Based Real-Time Monitoring System for Smart Connected Worker to Improve Energy Efficiency." *Journal of Manufacturing Systems*, vol. 61, Elsevier Ltd, 2021, pp. 66–76, https://doi.org/10.1016/j.jmsy.2021.08.009.
[14] Kirkpatrick, J. & Kayser-Kirkpatrick,W. (2014). *The Kirkpatrick four levels: A fresh look after 55 years*. Ocean City: Kirkpatrick Partners.