GEOMETRY PREDICTION IN WIRE ARC ADDITIVE MANUFACTURING USING MACHINE LEARNING

A Thesis
Presented to
The Academic Faculty

by

Marwin Gihr

In Partial Fulfillment
of the Requirements for the Degree
Master of Science in the
George W. Woodruff School of Mechanical Engineering and the University of Stuttgart

Georgia Institute of Technology
December 2022

GEOMETRY PREDICTION IN WIRE ARC ADDITIVE MANUFACTURING USING MACHINE LEARNING

Approved by:

Dr. Shreyes Melkote, Advisor
School of Mechanical Engineering
Georgia Institute of Technology

Dr. Christopher Saldana
School of Mechanical Engineering
Georgia Institute of Technology

Dr. Hans-Christin Möhring
Institute of Machine Tools
University Stuttgart

Date Approved: 12/06/2022

ACKNOWLEDGEMENTS

First of all, I would like to thank my supervisor, Professor Melkote, for giving me the opportunity to do research in an exciting field at Georgia Tech and for giving me the trust and freedom to follow my ideas. His guidance and open mind have allowed me to have an instructive and interesting year.

Secondly, I would like to thank my girlfriend who always takes my peculiarities with a laugh. I would like to thank my mother for giving me the confidence and composure to stand in society and my father for directing and tuning me toward success. I would like to thank my brothers for their unconditional love and all my friends for making my life special and accompanying me in my adventures. Last but not least, I would like to thank my lab mates for their support and time throughout my research and my room mates for ensuring that I wouldn’t starve during the last weeks.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

ANN	Artificial Neural Network
BH	Bead Height
BOM	Bead Overlap Model
BW	Bead Width
CA	Contact Angle
CTWD	Contact Tip to Work Distance
DP	Depth of Penetration
GFR	Gas Flow Rate
GMAW	Gas Metal Arc Welding
I	Current
ITP	Interpass Temperature
MAPE	Mean Absolute Percentage Error
MIG	Metal Inert Gas
OR	Overlap region feature
P	Power
PBF	Powder Bed Fusion
PG	process parameter – bead geometry
PIML	Physics-Informed Machine Learning
RMSE	Root Mean Squared Error
SLA	Stereolithography Apparatus
SLC	Selective Laser Cladding
TCap	Specific Heat Capacity
TCon	Heat Conductivity
TS	Travel Speed
TCP	Tool Center Point
U	Voltage
WAAM	Wire Arc Additive Manufacturing
WD	Wire Diameter
WFS	Wire Feed Speed

SUMMARY

Wire Arc Additive Manufacturing has disruptive potential for modern manufacturing. The technology comes with the flexibility and material efficiency of additive manufacturing processes while mitigating the disadvantages through high material output and high energy efficiency. The prevalence of the technology is inhibited by the large induced residual stresses and geometrical inaccuracy. This work tackles the latter by assessing the process parameter-geometry relationship using Machine Learning (ML) algorithms. To do so, multiple mild steel welding beads with varying shape features like corner angle are printed using a Metal Inert Gas (MIG) welding machine attached to an industrial robot. The cross-sectional profile of the printed beads is measured using a point laser sensor and correlated through different ML algorithms to input features such as travel speed (TS), wire feed speed (WFS), interlayer temperature, and shape features. By incorporating varying bead shapes, a holistic model, not limited to geometry prediction of straight beads only, is trained. Thus, the model holds the potential to learn the process parameter-geometry relationship for different shape features of a part. Using the model, excess material at the inner angle of corners determined by the overlapping regions of the two adjacent beads can be predicted. By generating a database of possible bead shapes a inverse algorithm was created, that suggests welding parameter combinations resulting in a smoother bead shape at corners. Additionally, a study on the transferability of common bead geometry prediction models on other research testbeds was conducted. The importance of input features for transferability is assessed and the potential to increase transferability by infusing the model training with mass conservation is examined.

CHAPTER 1. INTRODUCTION

Wire Arc Additive Manufacturing (WAAM) has disruptive potential for modern manufacturing (Israr et al., 2021; Xiong et al., 2022). The prevalence of the technology is mostly inhibited by high residual stresses and limited deposition accuracy (Ryan, 2020). This work is focusing on alleviating the latter by using a state-of-the-art Machine Learning (ML) approach. The first section of this chapter will give an overview of the applied technologies and point to gaps in the current research. From there, Section 1.2 will derive objectives for this work and cluster it into three fields of research. The final section of this chapter will introduce the approach in each of these fields to achieve the described objectives.

1.1 Overview

In recent years Wire Arc Additive Manufacturing (WAAM) gained increasing attention in the field of metal Additive Manufacturing (Lin et al., 2021). The process combines Gas Metal Arc Welding (GMAW) with a robot or portal system to systematically melt and add material to shape complex and thin-walled structures (Norrish et al., 2021). The high build speed, the high energy efficiency, the cheapness of material, and the potential to build large and dense components are promising process characteristics to disrupt modern manufacturing (Zhao et al., 2020). But the prevalence of WAAM is still limited due to the high induction of residual stresses and limited deposition accuracy, when compared to other metal, AM technologies like Selective Laser Melting (Ryan, 2020). The accuracy of WAAM is difficult to control due to the complex interaction of multiple process parameters and complex, multi-scale process physics (Zhu et al., 2021). These two characteristics led many researchers to analyze the “process parameter – bead geometry” (PG) relationship with ML. ML finds the relationship between input and output parameters through the statistical assessment of a multitude of datapoints and thus avoids complex physical modeling (Rai & Sahu, 2020). The downside of Machine Learning is the cost-intensive creation of datapoints as it usually includes real-world experiments (Rai & Sahu, 2020). This is the reason why the number of datapoints used in the literature usually doesn’t exceed 30 datapoints per research paper and why only a few input parameters are varied and analyzed (s. papers listed in table 1). While the trained algorithms model the relationship of these input parameters towards the bead geometry well, it is limited to the narrow scope of each individual paper. This work will for the first time combine the data of several papers and test the transferability of algorithms trained on the data of one paper toward the findings of another paper. This approach holds several advantages. The increase in data points can help prevent the overfitting of algorithms and may allow the training of more sophisticated ML algorithms. Furthermore, as different sets of input parameters are varied in the analyzed papers the combination of the datasets allows the creation of a more holistic model. A holistic model holds the potential to enable the calibration of different WAAM systems and applications, instead of requiring each researcher to study the PG relationship afresh. Superordinate process characteristics like utilized material, Gas Flow Rate (GFR), or wire diameter (WD) can be included in the data and enable predictions for diverse manufacturing and research environments. In addition, a holistic model enables the quantification of the importance of each input parameter and thus enables more sophisticated geometry control systems and can point future research in the most promising direction. Next to the fact, that the ML models are only trained on one research testbed, the current models are limited to straight, single-layer beads. But real-world applications necessitate deposition accuracy throughout complex product shapes. This includes the manufacturing of overhangs, hollows, corners, and other shape features. For the first time, this work will examine the influence on the bead geometry when depositing beads on top of each other. Real-world applications will always consist of multiple layers and there are reasons to believe that the distance to the substrate and the shape of the surface influence the bead geometry. On top of that, the bead geometry development along part corners will be analyzed with an ML algorithm. In the corner region, cross-sections of the bead partly overlap which leads to changes in the geometry (Kulkarni et al., 2021). These changes can accumulate when depositing multi-layer components and can lead to deformed, and uneven product shapes and even to process failure (Ding, Zhao, et al., 2021). Next to an overlap region the deposition of corner structures usually involves an area of missing material at the very tip of the corner (H. H. Liu et al., 2020). When depositing wide structures with multiple adjacent beads, these regions can lead to porosity and inferior mechanical properties (H. H. Liu et al., 2020). A few research groups have started to approach these two defects individually and their solutions to one problem usually aggravate the other (s. Ding, Zhao, et al., 2021; H. H. Liu et al., 2020). A final shortcoming identified in the available ML models on geometry prediction is the fact that the bead geometry is usually captured by taking the mean bead height (BH) and bead width (BW) over the length of a bead deposited with certain welding parameters. This approach ignores the process variance in the deposited bead geometry. This is limiting especially for subtractive processes as the prediction of the mean geometry does not determine at which position the surface of the part should be anticipated in the worst-case scenario. In the case of robot milling, an excessive, initial depth of cut can lead to vibration, tool overloading, and potentially process failure. The following two sections derive the objectives for this work and explain the approach to achieving them.

1.2 Objective

The overall objective of this thesis is to exploit the potential of ML algorithms to mitigate the hurdles to the application of WAAM for complex products. This objective can be divided into three fields of research. The first field aims at assessing and improving the applicability of trained ML models in other application environments. To do so, this work aims at creating an overarching dataset containing bead geometry data of multiple different research testbeds. From there ML algorithm can be used to quantify feature importance on the prediction accuracy and model transferability. Finally, this work aims at improving transferability through state-of-the-art ML technology involving the augmentation of model training with process physics.

The second research field targets to increase the application of ML algorithms in the field of geometry prediction to more complex product features, like corners and multi-layer structures. To do so, data needs to be generated that captures the geometry evolution within these product features. This involves the definition of a geometry paradigm for beads in corners and at upper layers, to make the geometry development quantifiable and thus usable for an ML model. After generating the data and training an ML model, the influence of the input features needs to be analyzed to assure that the model learned reasonable and transferable relations between input features and the bead geometry. Finally, this thesis will use the trained model to derive an algorithm that is capable of suggesting welding parameters in the corner region that will lead to an optimized product contour. With this, the second field of research provides a benchmark approach for studies on the optimization of other product shape features, like overhangs.

The final field of study aims at generating an understanding of the influence of welding parameters on the geometry variance of the process. As described in Section 1.1, the current studies in geometry prediction only predict mean geometry features and ignore the variance of the process. The cross-section-based data captured for the second research field enables the generation of the variance characteristic for different welding parameters. This work will exploit this opportunity by assessing the predictability of geometry variance based on welding parameters and thus aims at delivering a tolerance band for existing geometry prediction models. Good predictability on the variance could also help in reducing the process variance on bead geometry by suggesting optimized welding parameters with minimal variance similar to the approach in the second research field of this work.

1.3 Approach

The approach to each research field is visualized in figure 1. For the transferability study, an overarching dataset is generated by collecting data from different research testbeds. This is done by extracting depicted data from the publicly available literature on geometry prediction and by contacting research groups that are active in this field. Thereafter, the collected data of each group is verified by applying the principle of mass conservation to their dataset. Data, where the volume of the described bead geometry doesn’t approximate the supplied material as determined by the utilized WFS and TS is removed from the dataset. The resulting dataset is used to train a multitude of linear Regressors and Artificial

Neural Networks (ANN) with different sets of input features. This allows for the assessment of the feature's importance on prediction accuracy and model transferability. Finally, the model training is enhanced by incorporating mass conservation into the loss function of the ML model.

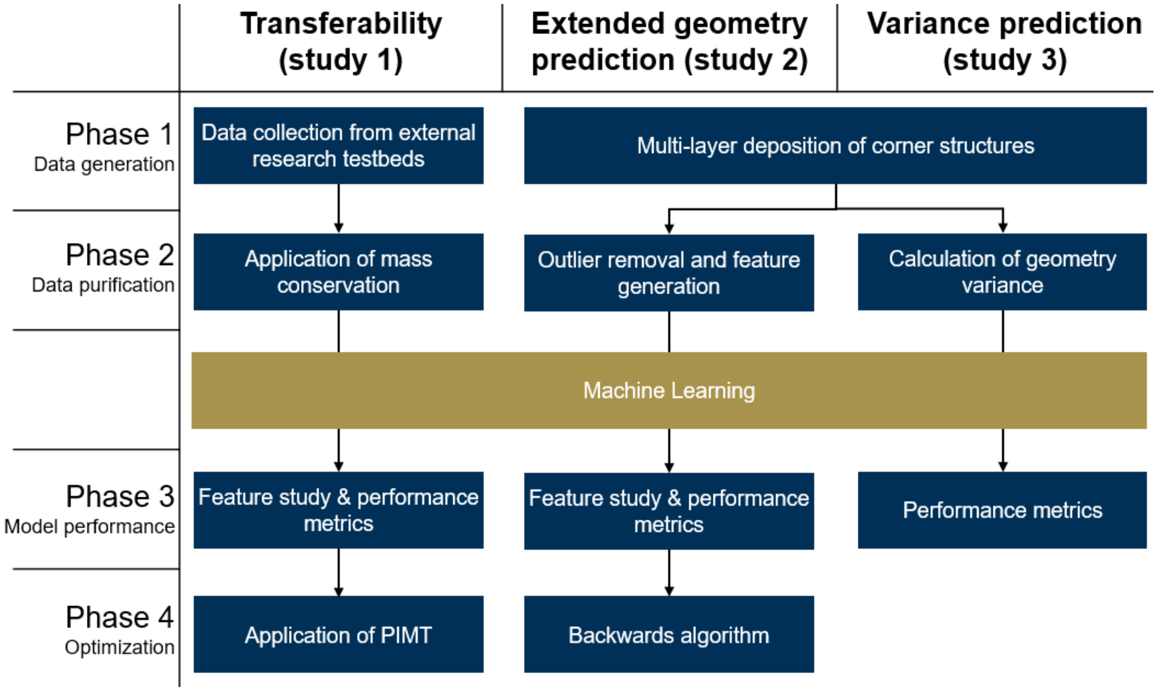

Transferability (study 1)
Extended geometry prediction (study 2)
Variance prediction (study 3)
Phase 1
Data generation
Data collection from external research testbeds
Multi-layer deposition of corner structures
Phase 2
Data purification
Application of mass conservation
Outlier removal and feature generation
Calculation of geometry variance
Machine Learning
Phase 3
Model performance
Feature study & performance metrics
Feature study & performance metrics
Performance metrics
Phase 4
Optimization
Application of PIMT
Backwards algorithm

Figure 1: Approach to research objectives in this work

As no data is available on the bead geometry of corners and multi-layer structures deposited by WAAM, experiments have to be conducted for the second research field. A Gas Metal Arc Welding (GMAW) system is attached to an industrial robot and used to deposit structures with varying corner angles and multiple layers. Each bead in every layer of these structures is scanned at multiple locations with a point laser sensor. From the resulting data streams, the point cloud of the bead surfaces is extracted. A geometry paradigm is defined that allows the geometry characterization of a bead at a corner and when deposited onto another bead. Quantifiable geometry features are derived from this paradigm and extracted from the purified data set containing the point cloud of the bead surfaces. An ANN is trained on the resulting dataset. The influence of the input features on the trained model is assessed by predicting on artificial datasets where only the feature of interest is varied. For the final step, the generation of a inverse algorithm, the trained ML model is used to generate a database with different bead geometries for varying welding parameters. This database allows for relating a desired bead geometry in the corner region with the welding parameters that would generate the most similar shape. The performance of the inverse algorithm is assessed by comparing the suggested deposition volume with the required volume according to the desired geometry.

The dataset for the third study is derived from the purified point clouds of the second study. The variance is calculated and an ANN is trained with the welding parameters as the input to the model. The performance of the model is assessed with different metrics.

The structure of this work is set-up to facilitate the presentation and understanding of the results. The section chapter will therefore introduce the basics of the applied technologies, WAAM and ML. The third chapter will detail the current state of the art in geometry prediction, corner modeling, and Physics-Informed Machine Learning (PIML). The fourth chapter will describe the data generation and data purification done for the three research fields and detail the approach to each topic. Chapter five will present the results and discuss the findings. The final chapter will summarize the work, draw a conclusion and point to the next steps on the geometry prediction frontier for WAAM.

CHAPTER 2. BACKGROUND

The following chapter lays the foundation of this work, by introducing the utilized technologies. Section 2.1 will describe the WAAM technology and delimit it from other metal AM processes. Section 2.2 will introduce the basic concept of ML algorithms and outline the steps of the ML process.

2.1 Wire Arc Additive Manufacturing

The generated and collected data for the three research fields of this work were produced with different set-ups of the WAAM technology. The following two sections will introduce the technology and describe its characteristics. Additionally, the main process parameters are presented. This lays the foundation for the input feature definition for the training of the ML models of this work.

2.1.1 Process

Wire Arc Additive Manufacturing (WAAM) is an Additive Manufacturing (AM) process of the type of Directed Energy Deposition (DED) (s. figure 2). In contrast to other AM process groups like Powder Bed Fusion (PBF) or Stereolithography Apparatus (SLA), the material is dynamically supplied during the process (Gebhardt, 2016). In contrast to technologies of the group Fused Deposition Modeling (FDM), the energy is supplied through an external heat source instead of heating the material before deposition (Gebhardt, 2016). In addition, the group of DED AM processes is the only group that encompasses technologies that use powder and wire-based material supply (Gebhardt, 2016). As WAAM makes use of traditional welding technologies, it belongs to the latter and involves feeding a wire into a traveling arc (Panda et al., 2019).

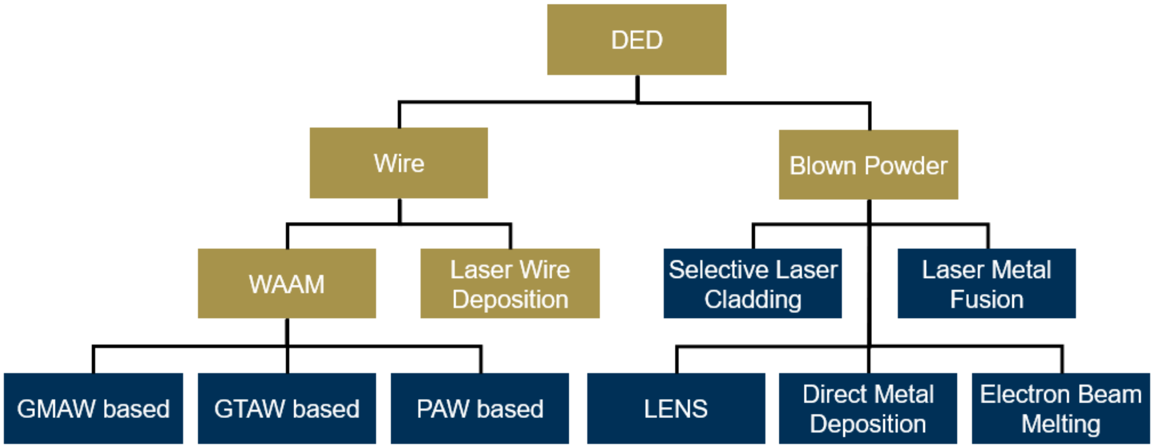


Figure 2: DED Processes (Lin et al., 2021)

The wire melts due to temperatures up to 30,000K (Phillips, 2016) in the arc and is deposited onto the surface of the object or the substrate. By using modern control software combined with a multi-axis machine tool or robot, the path of the welding torch can be controlled to subsequently build up a part layer by layer (Panda et al., 2019). In WAAM three different sub-technologies exist (s. figure 2). In contrast to Gas Metal Arc Welding (GMAW), the two sub-technologies Gas Tungsten Arc Welding and Plasma Arc Welding make use of a so-called “non-consumable” electrode to create the arc (Lin et al., 2021). This has the advantage of avoiding part quality defects caused by spatter (Phillips, 2016). Spatter involves molten material droplets getting ejected from the arc (Phillips, 2016). These droplets reduce the amount of material deposited in the bead and can stick to the part at an unwanted position (Phillips, 2016). Despite this disadvantage, GMAW has established itself as the leading technology in WAAM, due to its simplicity and good controllability (Phillips, 2016). Because of these advantages, the technology was also used for the experiments described in this thesis. The rest of this Section and the following

Section on arc-based welding fundamentals therefore focuses on GMAW. Figure 3 shows the basic principle of GMAW.

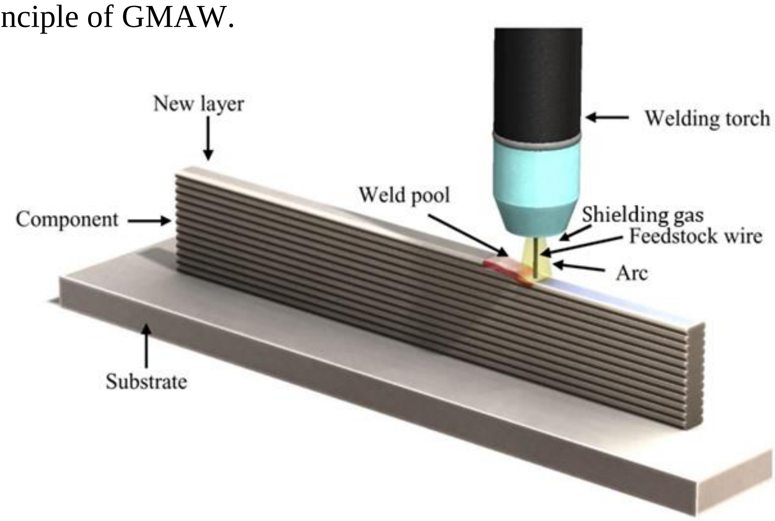


Figure 3: WAAM process schematic (Ryan, 2020)

As depicted in the figure, a welding torch supplies the wire as a consumable electrode, through the arc, and into the weld pool. The arc is surrounded by continuous inert gas flow, which avoids contamination of the metal bead with oxygen or nitrogen (Panda et al., 2019). For the first layer, the deposited material is fused with a substrate or build plate. Consecutive layers fuse with previously deposited layers of the component. This process leads to near-net shape parts with high density (compared to PBF). The deposition rate is up to 30 times larger than for PBF (Ryan, 2020), and the build volume scales with much lower cost (Panda et al., 2019; Ryan, 2020). The process is a comparably energy-efficient AM process and the feedstock material is a lot cheaper and easier to store and handle when compared to powder-based AM processes (Cunningham et al., 2018). See Ryan (2020) for a quantitative comparison of the different metal AM processes. The prevalence of WAAM is inhibited by the limited geometrical accuracy of the weld bead (Chen et al., 2022) and the introduction of high residual stresses and distortion due to a high thermal gradient produced during the process (Israr et al., 2021). The latter can lead to cracks and inferior mechanical part characteristics when compared to traditional manufacturing processes (Israr et al., 2021). Nonetheless, WAAM has been successfully applied for the manufacturing of a plethora of complex, large-scale products like a marine propeller-bracket, a landing gear rib, or even a metal bridge (Israr et al., 2021).

2.1.2 Welding Fundamentals

GMAW was introduced in the 1940s and continuously improved since then (Phillips, 2016). To establish the arc, a voltage in the range of 60-80V ("open-circuit voltage") is provided by a power supply (Phillips, 2016). If the gap between the feedstock wire and the substrate is small enough, the inert gas in the gap ionizes at this voltage and can thus close the electric circuit and transport electrons from the anode to the cathode (Phillips, 2016). Once the arc is established, the voltage can be reduced to a range between 10 to 40V (Phillips, 2016). The voltage drop in the gap is the largest right next to the electrodes (80-90%), which leads to an efficient introduction of heat into the wire and the substrate (Phillips, 2016). The GMAW power supply nowadays usually delivers a direct, pulsed current (Phillips, 2016). The pulsed current is achieved through a periodic increase and decrease in voltage, which helps to separate metal droplets from the wire (Phillips, 2016). In GMAW three different metal transfer modes exist, depending on the applied closed-circuit voltage. With a lower voltage, the arc length becomes negligible, and the circuit is shorted when the wire touches the substrate (Phillips, 2016). This mode is called short-circuit transfer mode and was historically prone to spatter and incomplete fusion problems as the current and heat input suddenly spike with each short circuit (Phillips, 2016). Modern power supplies bypass this problem with sophisticated current control, and thus, create a welding technology with less heat input and almost no spatter (s. Cold Metal Transfer)

(Phillips, 2016). A higher current flow leads to the globular transfer mode where large drops are separated from the wire (Phillips, 2016). This transfer mode is limited to horizontal welding positions, as the droplets mainly separate from the wire due to gravity (Phillips, 2016). Spray transfer, the most common transfer mode in industrial welding, is achieved with even higher closed-circuit voltages, where a large number of small droplets get ejected by electromagnetic forces and gravity onto the substrate (Phillips, 2016). The current thus highly influences the amount of supplied material and the heat input into the component. In GMAW heat is introduced through the hot material droplets on the one hand and the arc temperature on the other hand (Hu et al., 2018). According to Ohm's law, the heating of the wire and the substrate depends on the current squared, and thus, a higher current leads to a non-linearly larger and deeper melt pool (Phillips, 2016). The wire feed speed (WFS) controls the amount of supplied material, but as all supplied material needs to be melted by the arc, a higher WFS necessitates a higher current (Phillips, 2016). These two process parameters are thus highly connected.

The deposited bead shape and size are a result of a multitude of process and material characteristics. The volume of the provided material (V_{prov}) per millimeter is a function of the wire diameter (WD), the WFS, and the travel speed (TS) of the welding torch (s. eq. 1). A common way to approximate the deposited material per cross-section ($\frac{V_{dep}}{mm}$) is with an elliptic model (s. eq. 1).

$$\frac{WFS}{TS} * \pi * \frac{WD^2}{4} = \frac{V_{prov}}{mm} \overset{!}{=} \frac{V_{dep}}{mm} = \frac{BW}{2} * BH * \frac{\pi}{2} \qquad \text{(eq. 1)}$$

The geometrical distribution of the material is dependent on the size and depth of the weld pool before solidification (Bai et al., 2018). The size of the weld pool depends on the temperature distribution in the material, which in turn is a function of the heat input, the heat distribution in the arc, the heat conductivity of the material, the temperature of the deposition area before welding, the shielding gas flow rate and the material distribution and thus the convection in the weld pool. The forces acting on the material in the weld pool include gravity, electromagnetic force, surface tension force (Marangoni effect), and plasma arc force (Hu et al., 2018). The heat distribution in the arc depends on the contact-tip-to-work-distance (CTWD), the supplied voltage, and current, and the wire material due to its influence on the voltage drop in the wire and the torch angle. Thus, the simulation and prediction of the bead geometry is a highly complex problem and up to now has been impossible to solve analytically without major approximations (Bai et al., 2018). The introduced process parameters are visualized in figure 4.

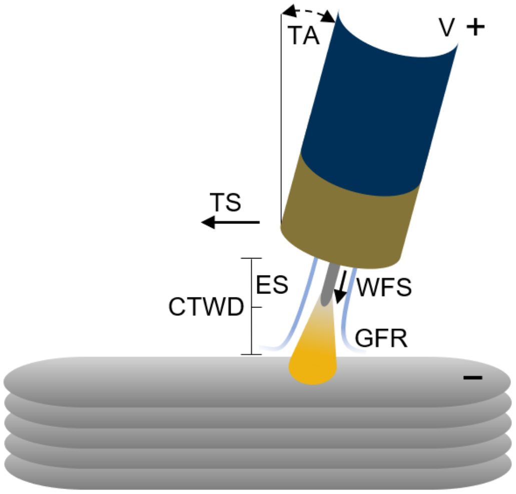


Figure 4: Process parameters in WAAM

2.2 Machine Learning

Machine Learning (ML) gained increasing attention throughout this century and is nowadays an imminent tool of modern society and industry. The largest companies of today are all directly connected to the application or provision of ML algorithms (Donnelly et al., 2017). In the context of manufacturing, countless applications for ML algorithms, like predictive maintenance or quality control, exist (Welsch et al., 2018). This work applies ML to tackle issues of inaccuracies in bead geometry deposition with WAAM. The following three sections will introduce basic concepts of ML by outlining the usual process of an ML project and by illustrating how "learning" is possible for a machine.

2.2.1 Overview

ML subsumes mathematical algorithms that discover patterns in data and use these patterns to categorize or predict based on new, unseen data points (Steiner & Welker, 2016). In the field of Artificial Intelligence, these algorithms are mostly used to establish capabilities in a machine for problems, which would otherwise be impossible or very hard to program (Müller & Guido, 2017). For example, it is hard for people to define what features facilitate the identification of a certain object in a picture, and thus it is very difficult to embed this capability in a computer program. With ML, AI was enabled to recognize and "learn" these features, without the need to explicitly program them (Welsch et al., 2018). In the field of Data Science or Data Mining ML is the tool to infer knowledge from data for application by humans (Provost & Fawcett, 2017). So, instead of creating capabilities for AI, the facilitation and improvement of human decision-making is the main objective. As stated in the introduction, this work lays the foundation for a data-driven algorithmic model that can enable the optimization of welding parameters in WAAM. While the generation of a deeper understanding of the welding process is (of course) desirable, it is not the objective of this work. Thus, while also applying technologies from the Data Science field (s. Section on algorithms), the focus of this thesis lies in the field of AI.

ML algorithms learn from a multitude of individual, self-contained data points. Each data point is a representation of the application and contains a numerical or qualitative characteristic embedded in the predefined, application-specific features (Müller & Guido, 2017). Depending on how the algorithm learns from these data points it is categorized according to figure 5. The first categorization distinguishes between supervised and unsupervised learning algorithms. In supervised learning, each data point for training the algorithm contains the features that will be predicted in the actual application (Müller & Guido, 2017). So, the algorithm learns to deduct the most likely value of certain features based on the other features of the data point. Depending on the ability of the algorithm to handle either quantitative or qualitative output features, they are categorized as a classification or regression algorithm (in some cases both) (Paper, 2020). Unsupervised algorithms on the other hand don't learn based on provided knowledge about the output but solely look for patterns in the dataset. These algorithms then use these patterns to simplify the dataset, as in the case of dimension reduction algorithms, or to group the data, as with clustering algorithms. Popular representatives of these respective groups are the Principle Component Analysis and K-Means clustering, respectively. The third and last group of ML algorithms contains the so-called reinforcement learning algorithms. These learn iteratively, based on trial and error, and have proven to be very proficient in recent years (Silver et al., 2017). As this learning approach implies many erroneous outputs during training, these algorithms are seldomly suitable for manufacturing applications (Wuest et al., 2016) that are characterized by their often demanding and cost-intensive processes and are therefore not used in the field of WAAM. This thesis makes use of ML algorithms that fall into the regression category. Two example algorithms will be introduced in Section 3.2.3.

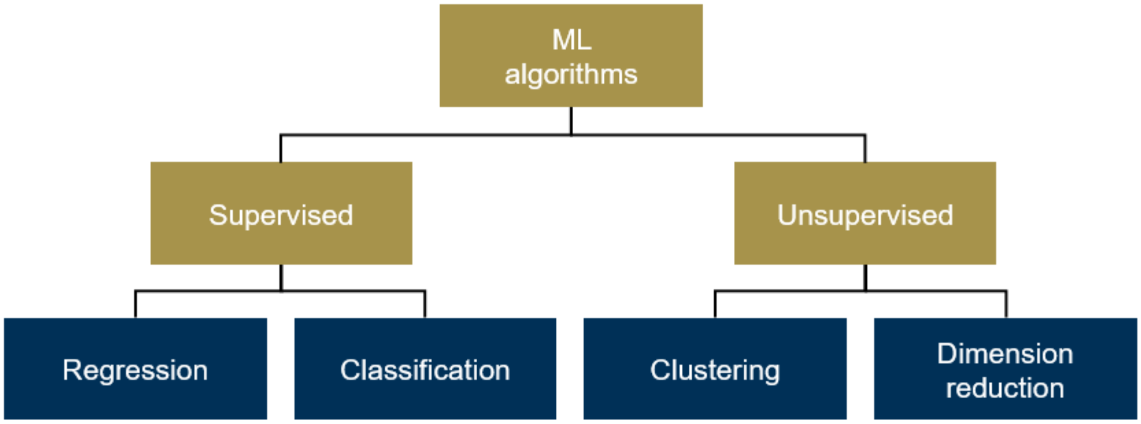
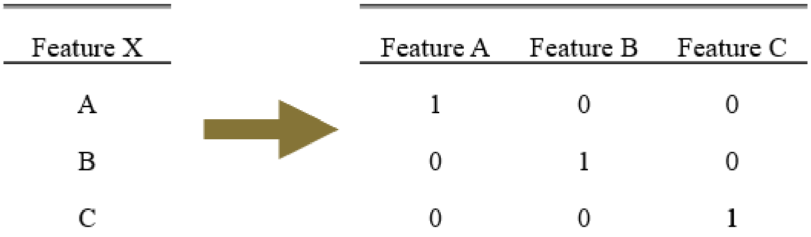


Figure 5: Categorization of ML algorithms

Nowadays various open-source libraries exist, which simplify the application of ML algorithms. The main libraries used in this work are scikit-learn (*Scikit-Learn.Org*, 2022) and PyTorch (*PyTorch.Org*, 2022). With these libraries, the biggest hurdles for successful ML projects become data quality and quantity (Buschbacher, 2016). Chapter 4 will describe the data collection and data processing steps utilized in this work.

2.2.2 Machine Learning Process

The ML process in the context of AI or Data Science is usually not just the application of an algorithm on a dataset but implies an iterative process with intensive pre- and postprocessing of the data (Provost & Fawcett, 2017). The first step in the ML process has to be an investigation of the cause-effect relationship(s) in the application of interest (Flath & Stein, 2018). From there, relevant features for the dataset have to be selected (Géron, 2019). This is a crucial step that determines the success of the project (Domingos, 2012). Additionally, it can be beneficial to refine or even mathematically combine features to make it easier for the algorithm to learn (Géron, 2019). By doing so, expert knowledge about process physics can be introduced into the dataset, and thereby the complexity of the cause-effect relationship reduced (Géron, 2019) (s. Section 3.3). The step involving feature creation also implies that the relevant characteristics are transferred into a machine-readable way. In this context, the so-called “one-hot encoding” is often applied, which translates qualitative into quantitative features (s. figure 6).



Feature X
A
B
C

Feature A	Feature B	Feature C
1	0	0
0	1	0
0	0	1

Figure 6: One-Hot Encoding (based on (Müller & Guido, 2017)

After the identification and extraction of features, the data usually has to be purified. In most applications, some data points have missing or illogical values for some of the features, which can cause ML algorithms to fail or perform worse. Thus, a common step is to impute new values for these features using statistical methods on the remaining data points (Mueller & Massaron, 2016). . After purification, in the case of supervised learning, the dataset is split into training and testing datasets. The testing set is used to evaluate the ML algorithm’s performance based on different statistical metrics. Popular metrics used to rate the performance of regression algorithms are the Root Mean Squared Error (RMSE), the Mean Average Percentage Error (MAPE), and the R-squared (R2). These three metrics are calculated according to eq. 2 to eq. 4 (Géron, 2017; Ruggeri et al., 2007), where m stands for the number of samples, y_i^* for the predicted value by the ML algorithm and y_i for the actual value. The R2 formula is a measure of the amount of variance in the output features that can be explained by the model (Ruggeri et al., 2007).

$$RMSE = \sqrt{\frac{\sum_{i=1}^{m}\left(y_i^* - y_i\right)^2}{m}} \qquad \text{(eq. 2)}$$

$$MAPE = \frac{100\%}{m} * \sum_{i=1}^{m} \left| \frac{\left(y_i^* - y_i\right)}{y_i^*} \right| \qquad \text{(eq. 3)}$$

$$R2 = 1 - \frac{\sum_{i=1}^{m}\left(y_i - y_i^*\right)^2}{\sum_{i=1}^{m}\left(y_i - mean(y_i)\right)^2} \qquad \text{(eq. 4)}$$

These metrics are usually also the baseline to optimize the so-called hyperparameters of the algorithm. These hyperparameters determine the way the particular algorithm is set up and the way it learns. An illustrative example would be the number of layers or the number of neurons in an Artificial Neural Network (ANN) (s. Section 2.2.3). Multiple systematic or heuristic approaches exist to optimize the parameter selection and these are still the subject of active research (Golovin et al., 2017). In this thesis, the so-called k-fold cross-validation approach is applied. Here the training dataset is split into k subgroups and the algorithm is trained on varying combinations of these subgroups with a systematic combination of different values for the hyperparameters (Géron, 2019). The trained algorithm is then tested on the remaining subgroups. The algorithm with the best performance on all subgroups is selected and finally evaluated on the test dataset.

2.2.3 Algorithms

In this section, the focus is on the way most regression-type ML algorithms learn the relationships underlying the data. First, the basic concepts of ML are described for a simple linear regression algorithm, before the concepts are transferred onto the ANN algorithm utilized in this work. The linear regression algorithm, like all ML models, contains several parameters that are used to transfer an input vector into an output vector (Bergstra &

Bengio, 2012). The values of these parameters are estimated from data, a process that is commonly referred to as learning (Stevens et al., 2020). For a linear model, the parameters that are learned are the offset (c) and the slope (a) of a linear function (s. eq. 5).

$$y(x) = a * x + c \qquad \text{(eq. 5)}$$

In the case of eq. 5, x represents a singular input parameter that gets transformed through the model parameters a and c. For the algorithm to learn a dataset of inputs and outputs must be available. To do so, the model gets initialized with random values for a and c, which are used to transform the input into a first guess for y (Stevens et al., 2020). As the actual y is represented in the dataset, the prediction error can be calculated. A common metric for the error is the Mean Squared Error, which, just like the RMSE (s. eq. 2), takes the square of the difference between the predicted values and the actual values, but saves the square root for computational efficiency (Stevens et al., 2020). By inserting eq. 5 into the calculation of the MSE and taking the derivative of the resulting function with respect to the model parameters, the effect of a change in the parameters on the loss function can be quantified (Stevens et al., 2020). This information is used to adapt the parameters in a way that the loss is minimized. This process is known as gradient descent and it scales up very well to many parameters in a model (Stevens et al., 2020). One advantage of the linear regression model is that the trained slope parameter is a quantification of the importance and the influence of the related input feature and can thus be interpreted (Paper, 2020). A popular representative of linear Regression is the Ridge algorithm. It combines the introduced concept with a regularization term in the loss function, which helps in avoiding overfitting (Géron, 2019).

An ANN has the same basic feature of transforming an input vector into an output vector. The transformation function is just a lot more complex. In contrast to the linear model, it is not only made up of a linear transformation but combines multiple linear transformations with non-linear transformations (Stevens et al., 2020). Each combination of a linear and non-linear transformation is called a neuron. The non-linear part is called the activation function and is usually a function that is (close to) 0 for negative inputs and around 1 for positive inputs with a small transition region. The equation for a neuron with the hyperbolic tangent as the activation function is depicted in eq. 6. An ANN is usually made up of multiple neurons that sequentially feed into each other. Thus, the output of multiple neurons can form the input of one subsequent or multiple subsequent neurons. An example architecture of an ANN is depicted in figure 7. This combination of linear and non-linear functions in a neuron can, in theory and when using enough of them, approximate any functional relation (Géron, 2019). So, in contrast to the linear model, with an ANN no prior assumption regarding the functional relation of the input vector to the output vector has to be made. As mentioned above, the principle of gradient descent transfers well onto the application with multiple parameters. So, when training an ANN, the influence of an alteration of each model parameter on the loss is calculated for each data point and then tweaked in a way to minimize the loss (Géron, 2019). The amount each parameter is adjusted by is determined by multiplying the gradient with the so-called learning rate. Next to the number of neurons and layers in the ANN architecture, the learning rate is one of the major hyperparameters of an ANN. Another important hyperparameter is the number of epochs, which describes how often each data point is presented to the algorithm for adjusting its weights (Nagesh & Datta, 2002). A common way to approach the optimization of an ANN is to increase the model complexity by increasing the number of neurons until it fits the training data well and then reducing the complexity until it stops overfitting (Stevens et al., 2020).

$$y(x) = \tanh(a * x + c) \qquad \text{(eq. 6)}$$

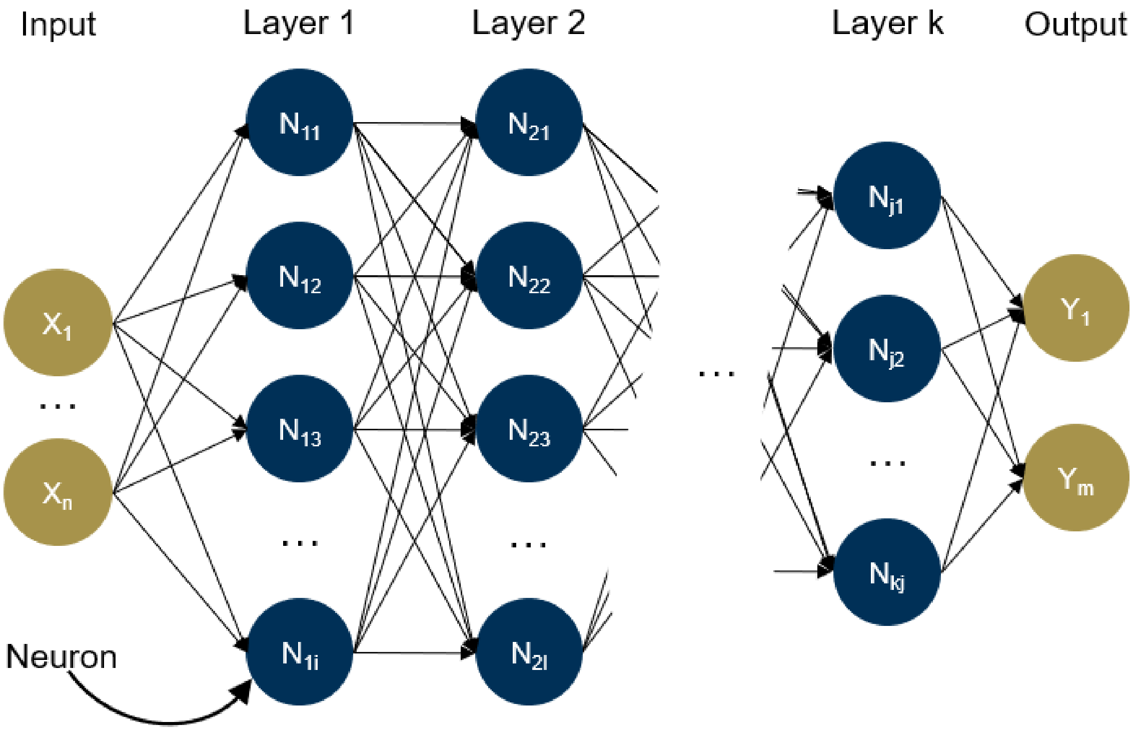


Figure 7: Visualization of an ANN with n inputs, m outputs, and k layers

CHAPTER 3. LITERATURE REVIEW

Chapter three will describe the current state of the art at the interface of ML and WAAM. Section 3.1 will describe the common approach to bead geometry prediction and point to its limitations. Additionally, the papers and research groups identified as potential data sources for the transferability study are introduced. Section 3.2 will outline the current state of specialized geometry prediction, which focuses on particular part features. Section 3.3 will introduce the concept of PIML, a current trend in the field of ML that combines the data-based approach with the conventional approach of physical formulation, to achieve superior performance in modeling real-world phenomenons. The chapter is concluded by deriving the research fields of this work from the identified gaps in the literature.

3.1 Prediction of Bead Geometry

Initially, the “process parameter – bead geometry” (PG) relationship was primarily studied by analyzing the influence of a single, isolated parameter (Zhao et al., 2020) or by trying to model the physics of the welding process (Venkata Rao et al., 2022). But with the advent of ML and the recognition of its superiority in predicting bead geometry, a plethora of PG studies using ML have emerged (s. excerpt in Table 1). The usual approach consists of depositing multiple, single-layer beads with different input parameter combinations and measuring bead height (BH) and bead width (BW) to generate data for supervised learning algorithms (s. for example Deng et al., 2019; Ding, He, et al., 2021; Zhao et al., 2020). The most common input factors considered in developing ML algorithms for bead geometry prediction are TS and WFS (s. Table 1). While TS describes the speed of the machine or robot depositing material, WFS represents the feed rate of the electrode. Lambiase et al.

(2022), Mu et al. (2021), and Ding et al (2021) solely predict the bead geometry based on these two input factors and achieve good results with a Mean Squared Error as low as 1.9 mm (MAPE~10%) (Ding et al., 2016). On top of these two input factors the Interpass Temperature (IPT) (Ding, He, et al., 2021; Wacker et al., 2021), the Contact Tip to Work Distance (CTWD) (Nouri et al., 2007; Zhao et al., 2020), the torch angle (Gokhale et al., 2019; Venkata Rao et al., 2022), the measured peak current (Nagesh & Datta, 2002) or the set current on the welder (I) (Deng et al., 2019) were found to influence bead geometry and used to improve the geometry prediction. Linear Regression models and ANN are the most common ML Algorithms used (s. Table 1). While ANNs yield accurate prediction results (Venkata Rao et al., 2022), Linear Regression allows for the assessment of the influence of the input factor on the geometry (Mu et al., 2021; Nouri et al., 2007). Apart from decision tree-based algorithms (Deng et al., 2019), genetic algorithms like Gene Expression Programming (Panda et al., 2019) and the Support Vector Machine Algorithm (Ding, He, et al., 2021) have also been applied. In general, the MAPE of the ML models reported in these papers is less than 10% and as low as 2% for the BW prediction by an ANN (Venkata Rao et al., 2022).

Table 1: Summary of ML-based research on the PG relationship

Author	Input Factors	Output	Algorithm
Deng et al. (2019)	TS, I, U, IPT	BW, BH	XGBoost
Ding et al. (2016)	TS, WFS	BW, BH	Neural Network
Ding, He, et al. (2021)	TS, WFS, IPT	BW, BH	Support Vector Machine
Dinovitzer et al. (2019)	TS, WFS, I, GFR	BW, BH, DP	-
Gokhale et al. (2019)	TS, WFS, I, TA	BW, BH	Linear Regression
Israr et al. (2021)	TS, P	BW, BH	Neural Network

Table 1 continued

Lambiase et al. (2022)	TS, WFS	BW, BH	Linear Regression
Mbodj et al. (2021)	TS, WFS, P	BW, BH	Linear Regression
Mu et al. (2021)	TS, WFS	BW, BH	Linear Regression
Nagesh and Datta (2002)	TS, WFS, P, U, I, L_A	BW, BH, DP	Neural Network
Nouri et al. (2007)	TS, WFS, CTWD, TA	BW; BH, CA	Linear Regression
Panda et al. (2019)	TS, WFS, I	BW, BH	Genetic (GEP)
Venkata Rao et al. (2022)	TS, WFS, TA	BW, BH, DP	Neural Network
Wacker et al. (2021)	TS, WFS, IPT, L	BW, BH	Neural Network
Zhao et al. (2020)	TS, WFS, CTWD, U	BW, BH, DP	Genetic (Grey Wolf)
…	…	…	…

Common to all these studies is the fact that their testing data is always generated using the same experimental setup used to generate the training data. Thus, overfitting to the respective research testbed instead of modeling the PG relationship can’t be avoided. It is like deducing the relationship between mass and gravitational force by only looking at falling apples on earth. It has value but can’t be transferred to applications outside the environment of the data source. In addition, the influence of superordinate process parameters like GFR or utilized material is not assessed even though these characteristics are expected to influence the PG relationship. Both characteristics alter the size of the melt pool, as the GFR impacts heat convection, and the heat transport characteristics of the material impact heat accumulation (Israr et al., 2021; Nagesh & Datta, 2002). Furthermore, it is evident that all studies only assess the bead shape of single-layer deposition. This is surprising as the influence of the layer number on the temperature cycle in the bead and thus on the melt pool characteristics has been widely established (Bai et al., 2018; Vora et al., 2022). The solidification time of the deposited material for lower layers has been proven to be shorter than for subsequent layers (Vora et al., 2022). This is because the heat is directly conducted into the substrate and doesn't have to go through the (potentially thin) deposited wall. The thermal resistivity is thus the smallest for the first layer. In addition, the substrate does not experience any thermal cycling before the deposition of the first layer and is thus colder than for subsequent layers and can conduct heat easier. This should not only influence the grain size and mechanical properties of the bead (Vora et al., 2022), but also the melt pool size and thus the bead geometry (Y. Li et al., 2018). Another factor that points to the importance of including the layer number as an input feature to the ML model is the fact that the shape of a droplet on a surface is influenced by the surface curvature (Wu et al., 2015). The surface tension of the droplet can lead to a different contact angle (CA) when deposited on a convex surface, as is the case for layers that are not deposited on the substrate (s. figure 8).

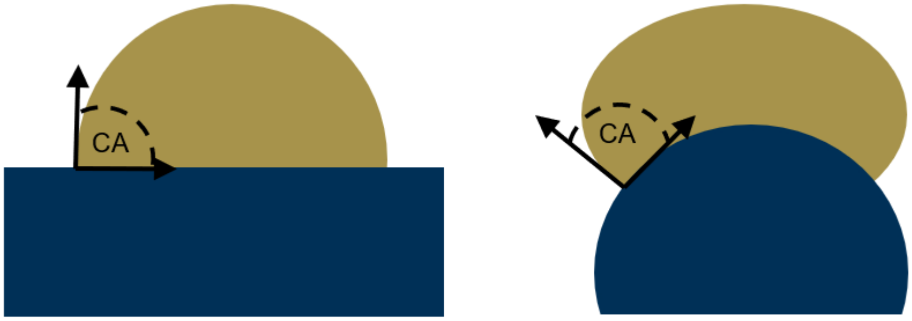


Figure 8: Contact angles for different surface shapes (based on Wu et al. (2015))

In addition to BH and BW, the depth of penetration (DP) (e.g. Dinovitzer et al., 2019; Zhao et al., 2020), as well as the contact angle of the bead to the substrate (Nouri et al., 2007) was studied in some papers. Thus, the visible shape of the bead is only approximated by two, or in one case, three parameters. To deduce the cross-sectional geometry of the deposited bead from these parameters different shape approximation models were tested by Lambiase et al. (2022). According to their study, an elliptic model best approximates the actual shape, when compared with parabolic and circular models. In the work of Ding et al. (2015) the parabolic model better approximated the bead cross-section. These contradictory findings show that only BH and BW can’t fully describe the actual cross-section geometry. For example, shifts of the bead height maximum relative to the cross-section can’t be captured when only using those two features. On the one hand, it is important to accurately predict the part shape formation for multiple layers and certain shape features like corners, circular shapes, or overhangs (s. Section 3.2), and on the other hand to capture geometrical defects. Chen et al. (2022) trained a classification-based ANN on geometrical defects like humps, path deviation, and deposition collapse. Especially the latter two geometrical defects cannot be captured in the ML training data if only BW and BH are used.

In summary, the identified shortcomings in the existing literature are the following:

1. The trained ML models are only validated on data gathered from the same research testbed. Thus, none of the papers can claim that their model captures the physical relationship between the input features and the bead cross-section. All models are overfitted to the research testbed. This calls for an overarching study on multiple research testbeds, which also takes hyperparameters like material characteristics or wire diameter into account.
2. The ML models are limited to training on single-layer beads. The effect of depositing on a previous layer instead of the substrate is not captured.
3. The bead shape is mostly approximated by only two shape characteristics (BH and BW), which can’t capture defects and shifts of the highest point with regard to the weld path.

4. The BH and BW are usually computed by taking the mean of these features over one bead. While creating a more stable output for the training of the model, this approach ignores the variance within a bead. When looking at hybrid manufacturing, it is of limited use to predict the mean BH of the layer to remove using a subtractive process (e.g., milling), when no knowledge about the variance of the set of utilized welding parameters exists.

3.2 Prediction of Shape Features

As introduced in section 3.1 a multitude of studies on single-layer, straight depositions exist. But as products manufactured by WAAM will be a lot more complex than straight walls, the bead geometry for shape features like corners, overhangs, and slopes need to be analyzed as well. In this area, only a few papers exist. The highest research density can be found for multiple, horizontally adjacent beads. In this field, the Bead Overlapping Model (BOM) (s. Ding et al., 2015) is a widely accepted model to determine the optimal distance between two adjacent beads to produce a flat layer surface (Y. Li et al., 2018). The BOM determines the optimal distance between the deposition trajectory of two adjacent beads by equalizing the overlapping area at the bottom of the two adjacent beads with the area of missing material in between the top of the two beads (s. figure 9). By approximating the bead cross-sections with a parabolic function the optimal center line distance can be analytically identified to be 0.783*BW (Ding et al., 2015). Y. Li et al. (2018) established that the center of the bead shifts when a bead is deposited next to an existing bead and thus enhances the BOM to adapt to those center line shifts. This underlines the importance of not only predicting the height of a bead but also the position of the bead center, as it can influence the optimal distance of adjacent weld beads.

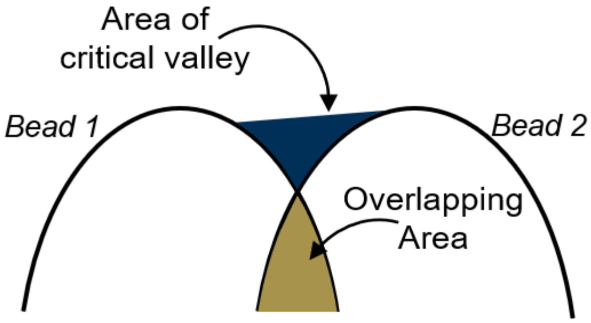


Figure 9: Important cross-sectional areas for the BOM (based on Ding et al. (2015))

Petrik et al. (2022) studied and predicted the geometry of beads when producing part shapes with a radius. Multiple circular beads were deposited with varying radii. It was found that the geometry of the bead gets altered up to a radius of 8mm (Petrik et al., 2022). For radii larger than 8mm the bead shape could be deduced from a straight bead deposition. For smaller radii, a shift of the bead center towards the center of the circular deposition was observed (Petrik et al., 2022). This study is also the only study that describes a bead based on multiple cross-sections and on multiple points per cross-section. This enables the analysis of the geometry development when approaching or exiting the circular part of the bead. To do so, the Euclidean distance to the middle of the circular part of the bead and an angle with respect to a perpendicular plane at the beginning of the circular part was used in addition to the radius and the travel speed as the input vector for the ANN model. Each cross-section was described using 16 measurement points, which served as the output for the training of the ANN.

A few studies were conducted on corners and how to avoid material accumulation in the overlap region of the two bead arms (s. figure 10). This material accumulation can aggregate when depositing multiple layers and can lead to defects in the part (Ding, Zhao, et al., 2021). Ding, Zhao, et al. (2021) adapted the BOM to two bead arms that overlap at a corner. They analytically derived the amount of excess material in the overlapping region and used the model to propose a (faster) TS at the corners. This effectively reduces the deposited material per cross-section and can thus create a more uniform surface of the part.

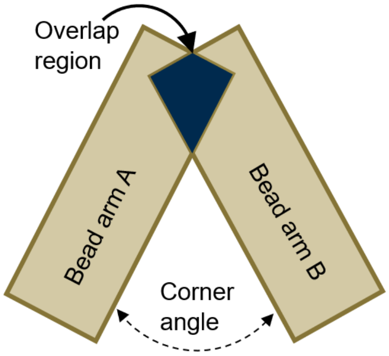


Figure 10: Bead overlap region at corners (based on Ding, Zhao, et al. (2021))

F. Li et al. (2018) consider the dynamic constraint of the robot when depositing in corners. As the travel direction of the robot is drastically changed at these locations the robot can't keep a continuous TS. The inertia of the robot combined with a finite jerk necessitates a reduction of TS at the corners. This enhances the material accumulation in the overlap region and can thus also lead to an uneven surface and process failure when depositing multiple layers (F. Li et al., 2018). F. Li et al. (2018) thus train an ANN on straight, single-layer beads with TS and WFS as the input and use this model to deduce a WFS that generates the same BH and BW with the reduced TS at the corners. The excess material at the corners was reduced by an average of 65% (F. Li et al., 2018).

H. H. Liu et al. (2020) don’t study the reduction of the vertical material accumulation at corners but point out the fact that corners can lead to porosity in the part when producing a corner with multiple adjacent beads. Here the round, outer-contour of the inner bead can’t fill the angular inner contour of the outer bead (s. figure 11). H. H. Liu et al. (2020) compensate for the missing material by adapting the tool path of the inner bead to produce a more acute outer contour. Unfortunately, to do so the amount of material at the corner will inevitably increase as well which in turn leads to an increased material accumulation and potentially to process failure when multiple layers are deposited. On the other hand, the reduction of deposited material as proposed by the other two studies on corner manufacturing would worsen pore creation.

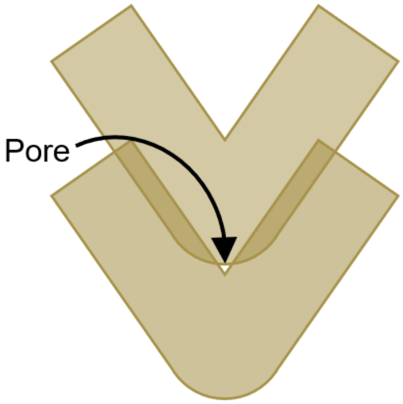


Figure 11: Formation of pores at corners with acute angles

This shows the fifth shortcoming in the current research on WAAM geometries:

5. The elimination of the two defects of over-deposition and porosity at the corners is connected and neither of the suggested methods can solve both issues at the same time. To tackle both issues simultaneously a deeper understanding of the bead contour development along the corner, as well as the material accumulation in the overlap region have to be established. An ML model can capture and represent this understanding and help in creating the capability to tackle both corner-related defects at once.

3.3 Physics-Informed Machine-Learning

The advantages of data-driven models for metal AM over pure physical models have been widely recognized by the research community (S. Guo et al., 2022). At this time their capability of including a wide spectrum of process parameters and the dispensability of process assumptions and approximations often make them a superior tool for complex processes (S. Guo et al., 2022). The downside of these models is the lack of interpretability and thus also the difficulty to transfer the models to new applications (Daw et al., 2017). As already pointed out in Section 3.1 the models overfit to their data source instead of learning a physical relationship. This lack of interpretability makes the models prone to errors in the dataset and the lack of transferability limits the application range of a trained model (Daw et al., 2017). This is why the combination of physics with data-driven models has gained increasing attention in the area of metal AM (S. Guo et al., 2022). The field of "Physics-Informed Machine Learning" (PIML) can be grouped into five different methods of how to include physics in data-driven models. These five groups are introduced in the following and visualized in figure 12 before specific examples from the field of AM are given. *Physics-Informed Model Input* (PIMI) subsumes the most intuitive form of PIML, where collected data is preprocessed to provide the model with physically more relevant data (S. Guo et al., 2022). This also includes the collection of input data from a simulation of the analyzed problem. This kind of richer data can reduce the number of necessary experiments and facilitate the learning for the ML model (S. Guo et al., 2022). This method is strongly connected to the feature extraction phase of the ML process (s. Section 2.2.2). The second method of PIML, *Physics-Informed Model Training* (PIMT), refers to adapting the way a model learns. In Section 2.2.3 the loss function of supervised regression models was introduced. Instead of only including a metric that penalizes wrong predictions when compared to the training data it also includes a loss term when known and process-relevant laws of physics are violated. *Physics-Informed Model Components* (PIMC) aims at enriching the activation function or the initialization process of a ML model (S. Guo et al., 2022). This requires understanding of the information representation of the ML model at intermediate levels. When this understanding exists the activation function or the initial values of the model parameters can be adapted to better represent the analyzed application (S. Guo et al., 2022). *Physical-Informed Model Architecture* (PIMA) is somewhat connected to PIMI. But instead of transforming the input data statically with a known physical relationship, the model architecture is adapted in a way that the model can easily represent this physical relationship, while still being able to learn and adapt certain parameters of this relationship. Finally, *Physics-Informed Model Output* (PIMO) refers to the application of additional algorithms on the model itself to uncover the learned relationship within the trained ML model.

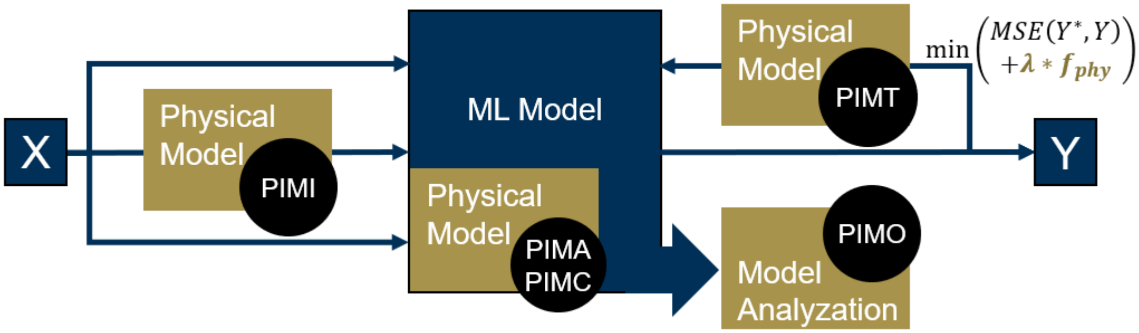


Figure 12: Illustration of PIML options in the application of data-driven algorithms

For examples of PIMO and PIMC the reader is referred to Grezmak et al. (2020) and Howland and Dabiri (2019), respectively, as no work in the field of AM could be found for these methods. As for the other three categories, PIMI is the one that is most often represented in the AM literature. An overview of the identified papers is given in table 2. A popular example of PIMI in the field of AM is the work of Du et al. (2021). Here the data from six different research environments that study the balling of powder in the PBF process is combined and their input features are transformed using process physics. Richer input data was, for example, created by transforming the laser power, scanning speed, and laser radius into an energy density parameter. On top of that, the process was simulated using a 3D, transient heat transfer, and fluid flow model, and additional parameters like the solidification time or the pool aspect ratio were extracted from the simulation and used as input to the ML model. By using such features, the algorithm achieved an accuracy of over 90% in all six research environments that were used to form the data set. Another example of PIMI can be found in the work of Ness et al. (2022). Here a surrogate model for thermal history prediction is created by deducing physically relevant features from a FEM simulation. By abstracting the modeled part with features like the distance of the welding torch to the node of interest or the amount of surrounding material to the node, the surrogate model can predict the thermal history for unseen deposition patterns and part geometries

(Ness et al., 2022). The only study identified for PIMA was conducted by Nagarajan et al. (2019). While the data was collected for an FDM and thus a polymer process, the example is still presented here as the approach can be directly transferred to metal AM and WAAM in particular. Here the process physics that determines the wall thickness and part height based on independent variables like the temperature of the heat source and filament feed speed are graphically approximated. This graphic visualization helps to identify parts where prior knowledge about process physics can be infused into the transformation of inputs to the output of the ANN. To do so, the ANN was divided into four ANN sections and intermediate physically relevant parameters for those sections were defined. For example, the input of filament feed speed and heat source temperature was used to train a section of the ANN to predict the viscosity of the filament. This parameter is fed into the next section of the ANN together with other transformed input parameters. By doing so, Nagarajan et al. (2019) were able to reduce the amount of necessary training data and increase the transferability of the ANN. An example of PIMT is given in the work of Zhu et al. (2021). Here the temperature, velocity, pressure, and melt pool dimensions are predicted based on data generated by a FEM simulation. To accelerate training and reduce the amount of necessary data the ANN's loss function is enhanced by adding terms that penalize violations of mass, momentum, and energy conservation. This approach also helped in reducing the amount of necessary data when compared to solely training on a conventional ANN loss function.

Table 2: Collection of PIML studies found in metal AM

Author	AM Process	PIML Category	Objective (prediction of)
Du et al. (2021)	PBF	PIMI	Powder balling
Gaikwad et al. (2020)	PBF & SLC	PIMI	Porosity and distortion
Herriott and Spear (2020)	SLC	PIMI	Yield strength
Kouraytem et al. (2021)	Review	Review	Review

Table 2 continued

Nagarajan et al. (2019)	FDM	PIMA	Geometry
Ness et al. (2022)	WAAM	PIMI	Thermal history
Ren et al. (2020)	SLC	PIMI	Thermal history
R. Liu et al. (2021)	PBF	PIMI	Pore Diameter
S. Guo et al. (2022)	Review	Review	Review
W. Guo et al. (2020)	SLC	PIMI	Pore Diameter
Xie et al. (2021)	PBF & SLC	PIMI	Tensile & yield strength
Zhou et al. (2022)	WAAM	PIMI	Residual stress
Zhu et al. (2021)	PBF	PIMT	Thermal history

With this a sixth shortcoming can be identified in the literature on WAAM:

6. So far none of the five PIML categories have been applied to facilitate the transferability and reduce the amount of required data for bead geometry predictions in WAAM.

These six identified gaps in the literature can be grouped into three different areas of study. Gaps 1 and 6 relate to the transferability of geometry prediction models onto other research testbeds. Gaps 2,3 and 5 identify the need for a more sophisticated geometry model that takes layer number and corner features into account and is able to predict shifts in the bead maximum. Gap 4 refers to the need to not only predict bead geometry based on welding parameters but to also predict the variance.

CHAPTER 4. METHODOLOGY

From the identified gaps in the literature three areas of study were identified for this work. The first area analyses the transferability of WAAM bead geometry prediction models on data from different research environments. The second area examines the influence of layer number and corners on the bead geometry. The third area investigates the predictability of bead geometry variance based on welding parameters. The following three sections describe the approach to each study, the data collection procedure, and the data purification methods employed.

4.1 Methodology for the transferability study (study 1)

The objective of the transferability study is to generate a WAAM bead geometry prediction model that can predict on data from unseen research environments. Section 4.1.1 introduces the steps required to train such a model and assess its transferability. Section 4.1.2 details the data collection and data preparation for this study.

4.1.1 Approach for the transferability study

As introduced in Chapter 1 the objective of the transferability study is to test and improve the transferability of ML algorithms trained on data from one research testbed onto other research testbeds. This would have the potential to eliminate the need to study the bead geometry formation of new WAAM setups, that utilize different equipment, different welding settings, and potentially even different materials. To achieve this objective, the bead geometry data of multiple research testbeds were collected (s. Section 4.1.3) and the following five studies conducted:

1. Train, optimize, and test ML models on each dataset's full feature space individually:

   In each research testbed, different combinations of experiment settings were varied. While some only assess the changes in bead geometry when the WFS or the TS is changed, others vary features like GFR or the voltage. The first study will train an ML algorithm on the full feature space of every research testbed individually. This will ensure that the results for each research testbed are reproducible with the available resources for this work.

2. Train, optimize, and test ML models on each dataset individually and use only fully transferable input features:

   In the second study, the feature space is limited to WFS and TS for every research environment as only these features are present and varied in all studies. By doing so, the trained ML model can be tested on all other research environments. The performance on the same dataset is also assessed and compared with the metric scores obtained in study 1. This study is intended to show the importance of the excluded input features on model prediction accuracy.

3. Train, optimize, and test ML models on a subset of the available datasets that include additional input features:

   As some datasets share additional input features, the influence of these features on model transferability can be assessed. To do so, ML algorithms are trained on datasets in which these additional features are varied, once including these features and once without these features, and then tested on the remaining datasets, where the features are kept constant.

4. Train, optimize and test ML models on all but one dataset and assess the transferability onto the remaining dataset:

   To assess the influence of aggregating multiple research environments on the transferability to a new research environment, the ML algorithms are trained on all but one dataset and tested on the remaining dataset. Furthermore, the influence of superordinate features like WD, GFR, or material characteristics can be assessed. These superordinate features are kept constant in each of the research testbeds reported in the literature, but their influence can be examined by combining multiple sources.

5. Train, optimize, and test ML models on all but one dataset under the inclusion of process physics:

   In this study, the ML algorithm is augmented with process physics. This study investigates the PIMT approach introduced in Section 3.3. To do so, the loss function of the trained ANN takes mass conservation into account. This enables tweaking the model weights in a way that the deposited material approximates the input material (s. eq. 1). Opposite to the relationship between BW and BH, which can vary from research environment to research environment based on the experiment characteristics, the principle of mass conservation has to be valid across all research testbeds. A model that incorporates this principle should thus perform better when predicting on research environments with other settings and produce physically consistent results when extrapolating (s. Section 3.3).

To conduct the first four studies an overarching optimization and evaluation function is established using Python and the ML library scikit-learn. Within this function, the training data is split into training and testing datasets by a ratio of four to one. Thereafter, the input data of the whole dataset is normalized (mean of 0 and variance of 1) by the mean and variance of the training data. A Ridge algorithm and an ANN are trained on the scaled training data. The Ridge algorithm is a type of multiple linear regression model, which allows to assess the influence of each input feature. The ANN is a popular choice in the presented paper and has the potential to capture more complex relationships between inputs and outputs. To optimize the algorithms on the individual application and to avoid overfitting a hyperparameter, grid search with cross-validation is applied. This means that the training data is split using a ratio of four to one and the algorithms are trained with different hyperparameters on every combination of the five parts of the training dataset. The algorithm and hyperparameters with the best average score on all five validation datasets are determined. The algorithm is then tested on the test dataset, which was set aside at the beginning of the optimization function. The Root Mean Squared Error (RMSE), the Mean Average Percentage Error (MAPE), and the R2 score are used as model performance metrics. The RMSE score has the advantage of returning a physically interpretable score as it describes the deviation of the actual BH and BW in mm. The MAPE allows for direct comparison over the different research environments and for averaging the algorithm’s performance on BH and BW as it is unitless. The R2 score assesses the ability of the model to explain the variance in the data by comparing it to the constant prediction of a line through the mean of the data. Thus, the R2 score can also be negative if the algorithm performs worse than this constant prediction. For the transferability test in studies 2,3 and 4, the performance of the optimized algorithm is additionally evaluated on the remaining datasets using the aforementioned metrics. For study 5, the PyTorch library is used, as sci-kit learn doesn’t allow the adaption of the loss function. To optimize the hyperparameter selection, cross-validation is used in combination with the Optuna library. Optuna trains multiple models with different sets of hyperparameters (Akiba et al., 2019). The particular hyperparameters are not randomly but strategically chosen from a predefined hyperparameter space based on the performance history of the trained models (Akiba et al., 2019). As in study 4, the training and validation datasets are constructed from the data of all but one research environment, while the remaining research environment serves as the test case.

4.1.2 Data preparation for the transferability study

To conduct the transferability studies, publicly accessible data from research on geometry prediction in WAAM is retrieved from the literature. Researchers were contacted if not all data was available in the literature. The complete data of Venkata Rao et al. (2022), Nagesh and Datta (2002), Zhao et al. (2020), Panda et al. (2019), Gokhale et al. (2019), and Nouri et al. (2007) could be extracted from tables reported in their papers, whereas the data of Deng et al. (2019) and Israr et al. (2021) are only partially reported in their papers. Upon request, Mbodj et al. (2021) provided access to their data. Thereafter, the input factors which weren’t varied and thus are not included in the reported tables were gathered from the papers and added to the dataset. This includes parameters, like current or voltage (U), that are only varied in some studies, as well as superordinate process parameters like GFR, WD, or the feedstock material. Subsequently, each dataset was tested for physical coherence to ensure the quality of the final dataset. To do so, the mass conservation theorem was applied to the input and deposited materials reported in each paper. Figure 13 plots the provided material (s. eq. 1 in Section 2.1.2) versus the deposited material according to the bead geometry for each study. As this should roughly be the same, the data points from each study should be located around the diagonal in the diagram. Out of the nine datasets obtained from the literature, three didn’t pass this test and one couldn’t be tested as two factors of eq. 1 were not reported. For two of the papers (Mbodj et al. (2021) and Panda et al. (2019)), it became obvious that the TS and WFS were swapped. Panda et al. even concluded that a decrease in WFS leads to higher and wider beads, which cannot be the case as less material is input into the process (s. Panda et al., 2019). This underlines the need to take measures against overfitting to the research testbed and test algorithms on foreign datasets, as ML algorithms also perform well on physically incorrect datasets.

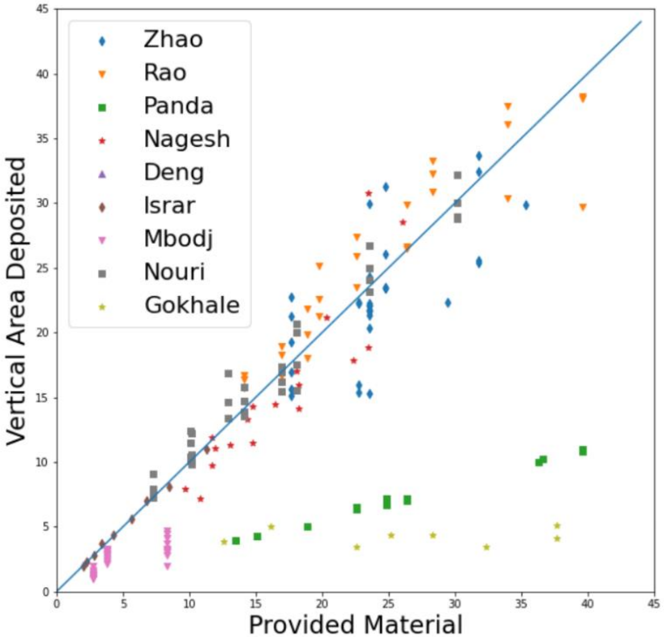
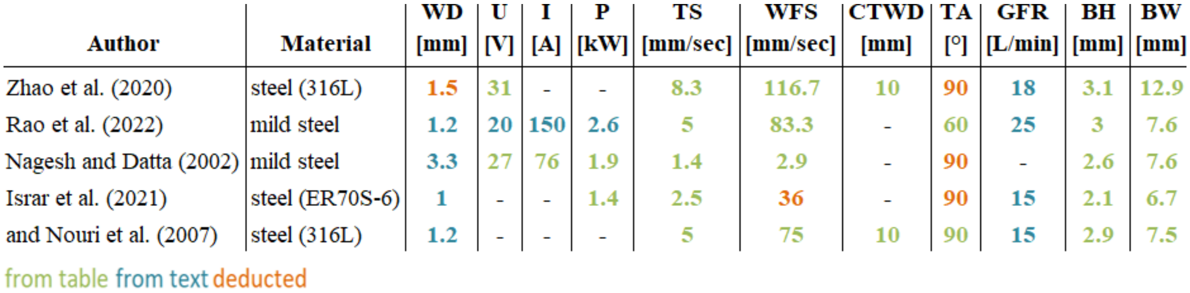


Figure 13: Physical coherence check by using mass conservation (s. eq. 1)

The datasets obtained from the remaining five papers were completed by applying the mass conservation theorem of eq. 1 where possible. By transforming eq. 1 a WD of 1.5 mm for the data of Zhao et al. (2020) and a WFS of 36 $\frac{mm}{sec}$ for Israr et al. were deduced from the mass conservation principle. In addition, the TA for the data of Israr et al., Zhao et al., and Nagesh et al. were assumed to be the standard 90° angle as no other information was provided in their papers. This constitutes the dataset for the transferability study. The dataset is reproduced in Table 3.

Table 3: Dataset for transferability study



Author	Material	WD [mm]	U [V]	I [A]	P [kW]	TS [mm/sec]	WFS [mm/sec]	CTWD [mm]	TA [°]	GFR [L/min]	BH [mm]	BW [mm]
Zhao et al. (2020)	steel (316L)	1.5	31	-	-	8.3	116.7	10	90	18	3.1	12.9
Rao et al. (2022)	mild steel	1.2	20	150	2.6	5	83.3	-	60	25	3	7.6
Nagesh and Datta (2002)	mild steel	3.3	27	76	1.9	1.4	2.9	-	90	-	2.6	7.6
Israr et al. (2021)	steel (ER70S-6)	1	-	-	1.4	2.5	36	-	90	15	2.1	6.7
and Nouri et al. (2007)	steel (316L)	1.2	-	-	-	5	75	10	90	15	2.9	7.5

from table from text deducted

4.2 Methodology for extended bead geometry prediction (study 2)

The objective of the extended bead geometry prediction study is to capture an understanding of the development of the bead geometry over multiple layers and around corners through an ML model. To conduct this study the following six steps were followed. Step one is the experiment design, where the constant and varied features for the experiments have to be defined. This step is described in Section 4.2.1. Thereafter, the data has to be collected according to step one. Section 4.2.2 describes the experimental setup and workpiece material. In the third step, the data is purified and prepared for application in an ML algorithm. This step is described in Section 4.2.3. The data is used for training, validation, and testing in step four. Step 5 uses the trained model to assess the influence of the individual features on the bead geometry. Finally, Step 6 utilizes the trained model and the incorporated understanding of the bead geometry development for corners, varying layers, and varying welding parameters to propose welding parameters that create a uniform bead geometry across different layers and without porosity or accumulated material at corners. Steps 4, 5, and 6 present the results of this study and are described in Chapter 5.

4.2.1 Experiment design for the extended geometry prediction study

To capture the influence of layer number and corner features, data has to be generated by varying these parameters. Ding, He, et al. (2021) only collect data for beads from layer four in their study since, according to the authors, the bead shape changes up to this layer. To capture this transition phase and validate their claim, ten layers for each wall geometry will be deposited and characterized in the current study. To capture the influence of corners on the bead geometry, beads with different corner angles are produced. In this study, the angle is varied in steps of 30° starting with a straight bead (s. table 4). To be able to adapt the cross-section of the bead geometry with the welding parameter settings in the backward model, an understanding of the influence of these parameters on the bead geometry has to be incorporated into the model. To do so, the welding parameters TS and WFS are varied as well. This constitutes the feature matrix of the experiments (s. table 4) wherein every possible parameter combination is captured in the experiments. While this full feature combination is not necessary for an ANN to capture the influence of each parameter, it simplifies the visualization of the data (s. Section 5.2.1).

Table 4: Feature levels for experiments

Feature	Units	Feature Levels									
TS	$\frac{mm}{sec}$	3	4.3	5.6	7						
WFS	$\frac{mm}{sec}$	60	72	83							
Corner Angle	$\circ$	180	150	120	90	60	30				
Layer number	-	1	2	3	4	5	6	7	8	9	10

The geometry of the beads is captured by measuring the bead cross-section every 1.5 mm perpendicular to the deposition trajectory (s. figure 14). Each cross-section is described by the BW and five BH measurements along the cross-section. The BH measurements are taken at five equally spaced locations along the cross-section. This approach has the advantage of being applicable to different BW values when compared to taking measurements at fixed locations in each cross-section. A potential alternative would be to predict different geometric descriptors like the contact angle or curvature together with the maximum BH. For ease of interpretability and comparison, the BH measurements relative to the BW are used. The potential downsides of this approach are discussed in Chapter 5. The choice of five BH measurements along the cross-section is not arbitrary. The mean BH is roughly a fifth of the measured BW throughout the experiment. Thus, by using five measurements of the BH and by assuming the same error distribution over the BH and BW measurements, the BH and BW prediction error will be weighted the same in the loss function of the ANN. Thus, the ANN will choose its model parameters in a way that BW and all the BH errors are equally reduced. The following example illustrates this line of thought. If ten BH measurements are used with one BW measurement to describe the bead geometry and the model predicts these characteristics with an error of 10% during training, then the absolute combined error of BH predictions would be almost twice as large as the error in the single BW measurements. The RMSE loss function in the ANN would then attribute greater importance to the combined BH errors and thus adjust the weights with a focus on BH prediction.

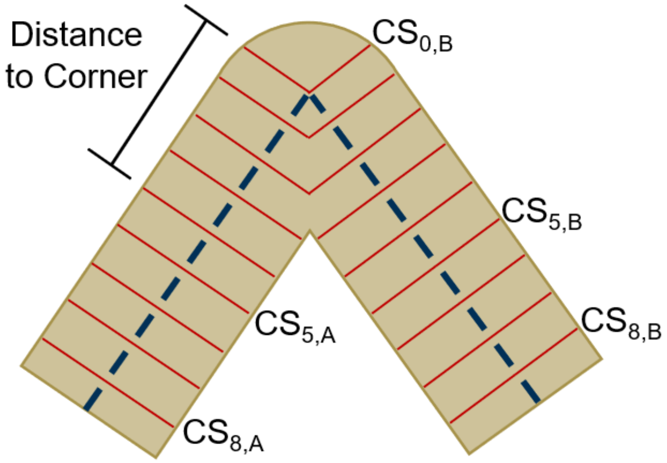


Figure 14: Bead geometry measurements along deposition trajectory

The BH measurements of the first layer are taken with respect to the baseplate surface. The measurements of subsequent layers are referenced to the volumetric mean of the preceding layer(s). This follows the assumption, that the preceding layer is remolten during the deposition of subsequent layers and that the molten material redistributes over the BW of the bead (s. figure 15).

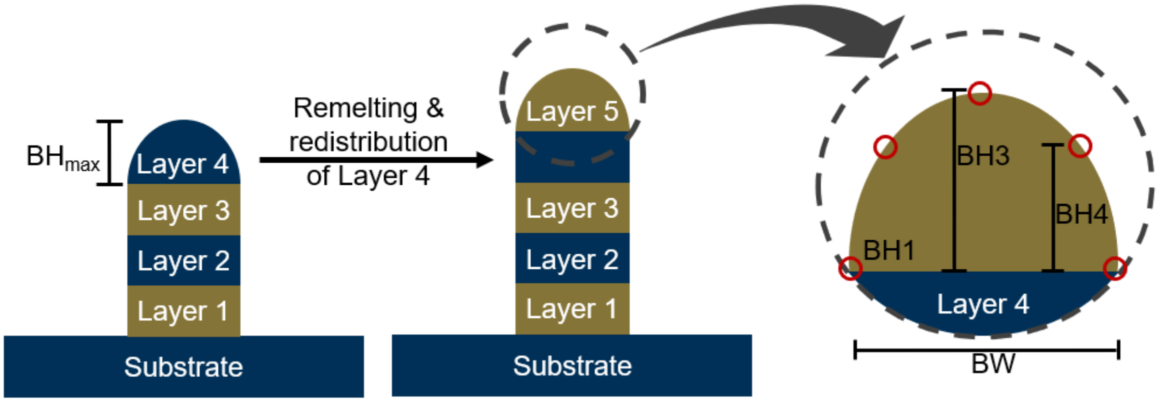


Figure 15: Visualization of output parameter creation

These input and output features constitute the data generated in the experiments. Thereafter, two additional input features are generated that help the ANN to learn the formation of the bead geometry. The first such input feature is the Euclidean distance of the cross-section measurement towards the center of the corner (s. figure 14). This allows the algorithm to learn the development of the bead geometry along the deposition trajectory around a corner. The other additional input feature is a Boolean descriptor, which captures if parts of the measured cross-section are in the overlap region of the corner. So, this input feature is 1, when parts of the cross-section are overlapping with another cross-section in the corner, and 0 if not. Following Ding, Zhao, et al. (2021) this overlap region can be determined when an assumption about the width of the bead for the applied welding parameters exists. The two bead arms of a corner overlap, when the distance perpendicular to the deposition path (s. $\underline{AB}$ in figure 16) of one bead arm towards the middle between the two bead arms (s. $\underline{CA}$ in figure 16) is smaller than BW/2. In this case, the distance to the corner (DtC) for which the bead arms overlap can be calculated from eq. 7. With this, all cross sections where the DtC is smaller than $D_{Overlap}$ are labelled as 1 in the overlap region feature (OR), whereas the other cross-sections are labelled as 0. As mentioned and as can be seen in eq. 7, an assumption about the BW is needed to calculated this input feature. To get an assumption for the BW the trained transferability model from study 1 is being used (s. Section 5.1.4). These two features complement the input space for the ANN depicted in table 4.

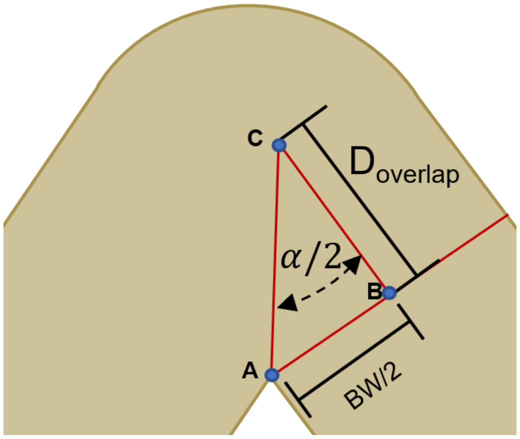


Figure 16: Illustration of the determination of the overlap region

$$D_{Overlap} <= \frac{BW}{2*\tan\left(\frac{\alpha}{2}\right)} \qquad \text{(eq. 7)}$$

After the definition of the feature space, the collected data needs to be purified (s. Section 2.2.2). The steps for purification are described in Section 4.2.3. Thereafter, a subset of the dataset is set aside for testing, whereas cross-validation is performed on the remaining data. As with the transferability study, the PyTorch library is used to create the ANN and the Optuna library for hyperparameter optimization. The performance results on the test set are reported in Section 5.2.2. The trained model is then used to assess the influence of the different corner angles and the different layer numbers. To do so, a fictional dataset is created wherein all but the feature of interest is held constant. This fictional dataset is fed to the trained model and the changes in the predictions, depending on the varied feature, are assessed. This approach can’t capture interdependencies between the feature of interest and the other features but provides a basic idea of its influence on the bead geometry.

Finally, the trained ANN is used to generate an optimized parameter set to create uniform beads in the corner region. To do so, many different TS and WFS combinations for different angles and different DtCs are fed to the trained model. The predicted geometries for these parameter combinations are stored in a database. Then, an optimal cross-section geometry is defined for the different angles and the different DtCs. Based on the desired geometry, the RMSE of all possible predicted cross-section geometries for the given angle and the particular DtC can be calculated. The welding parameters of the geometry from the database that is closest to the optimal geometry are then proposed for the deposition of the corner. The results of this approach are illustrated in Section 5.2.4.

4.2.2 Data creation and collection for the extended geometry prediction study

For data creation, the GMAW process is the WAAM technology in this work. The Power MIG 360 of Lincoln Electric Holding, Inc. is utilized as the MIG power supply. A 90-10 argon-carbon-dioxide-blend serves as the inert gas during the welding process. A Keyence LK-G35 point laser with a repeatability of 0.05µm (Keyence, 2022) is used to measure the cross-section of the deposited beads. The laser sensor and the Magnum Pro 300 welding gun are mounted onto the KUKA KR 210 L180 industrial robot (s. figure 17).

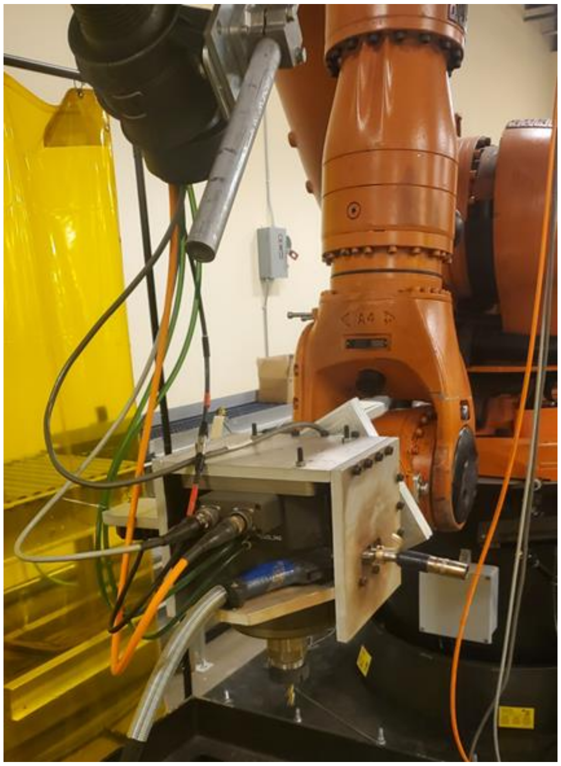


Figure 17: Robotic WAAM setup

The robot is controlled by a KRC2 controller. The robot trajectory is programmed using the Python API of RoboDK (s. RoboDK, 2022). Additionally, a spindle is attached to the robot flange as seen in figure 17 but is not used in this work. An ER70S-6 mild steel wire with a WD of 1.134 mm is utilized as the feedstock material to deposit onto a low-carbon, 30x30 cm^2 steel plate (A36) that serves as the substrate. The data sheets for the wire and substrate material composition are included in the Appendix. The KRC2 controller operates the outputs of a Beckhoff BK5200 bus coupler (Beckhoff Automation GmbH, 2022) via the DeviceNet interface, which in turn controls a relay. The relay closes the normally manually triggered control switch of the welding gun. By doing so the deposition by the welding machine can be controlled via the robot control program and thus triggered according to the position of the welding machine relative to the baseplate. The robot's Tool Center Point (TCP) positioning data stream during deposition is communicated via the Kuka Robot Serial Interface (RSI) to a personal computer for storage. The laser sensor bead cross-section measurement data is communicated over the USB interface onto the personal computer and stored as well. These two data streams serve as the basis for the data-driven modeling and analyses in this work. The combination and purification of these two data streams are described in Section 4.3. The laser measures at a rate of 200 Hz and a repeatability of 0.05 μm (Keyence, 2022). The robot speed during measurement was set to 20 $\frac{mm}{sec}$, whereas the repeatability was determined to be 0.02 mm. In total three deposition experiments were conducted, with each containing 24 beads à ten layers with a length of 80 mm (s. figure 18). The torch is positioned perpendicular to the substrate during deposition (torch angle of 90°) and the substrate is fastened to a work bench table with multiple screws to limit distortion (s. screw holes in figure 18). The CTWD is set to 15 mm during deposition but can vary during process within +/- 2 mm due to material accumulation in the overlap region and unpredictable process variation. The resulting range of CTWD lies within the recommended range for the utilized feedstock wire (12-19 mm) (The Lincoln Electric Company, 2022). The welding direction is altered after each layer to mitigate deposition errors due to the arc striking. In addition, the deposition sequence is chosen in a way to minimize heat accumulation in a particular area (s. sequence in figure 18). By doing so the effect of interlayer temperature on the bead shape is minimized. The best beads, characterized by minimum spatter and porosity, were produced using the pulsed mode of the welding machine. In this mode, the voltage and current can't be set manually but are controlled and varied automatically by the power supply of the welding machine.

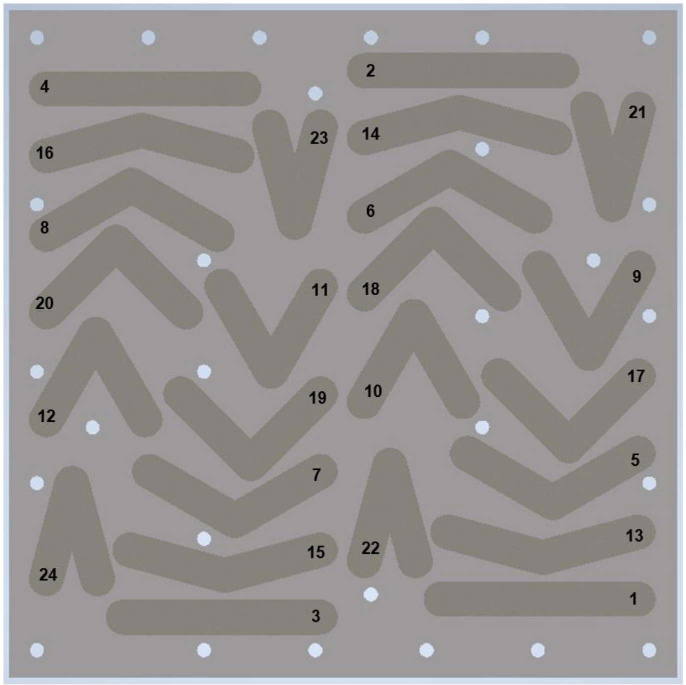
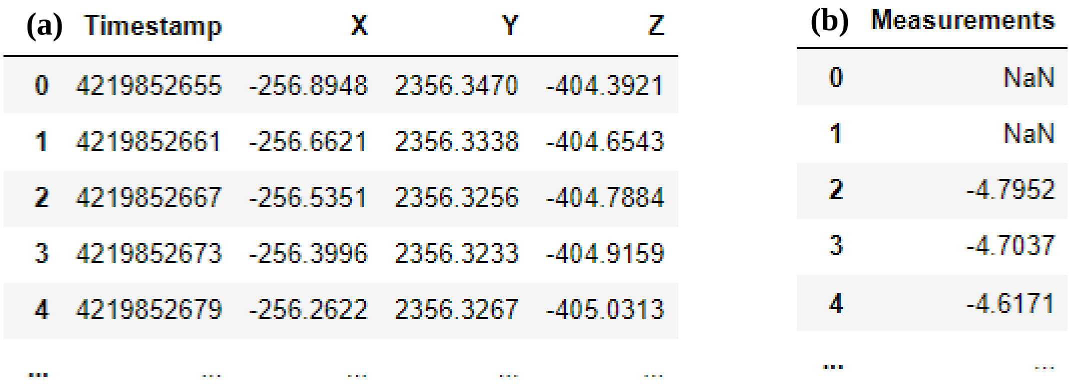


Figure 18: Deposition pattern for data generation

4.2.3 Data combination and purification for extended geometry prediction study

The data stream from the robot consists of the Cartesian positions of the TCP associated with timestamps (s. table 5). The laser sensor produces a continuous data stream without a timestamp, but at a reliable frequency (s. table 5).

Table 5: Excerpt of (a) robot's TCP trajectory and (b) laser sensor data stream



(a)

	Timestamp	X	Y	Z
0	4219852655	-256.8948	2356.3470	-404.3921
1	4219852661	-256.6621	2356.3338	-404.6543
2	4219852667	-256.5351	2356.3256	-404.7884
3	4219852673	-256.3996	2356.3233	-404.9159
4	4219852679	-256.2622	2356.3267	-405.0313
...	...	...	...	...

(b)

	Measurements
0	NaN
1	NaN
2	-4.7952
3	-4.7037
4	-4.6171
...	...

To combine the data streams of the robot's positional data and the laser sensor measurements, a synchronization movement is conducted at the beginning and the end of the consecutive measurement of six beads These synchronization movements allow the identification of data points in both datasets that were taken at the same point in time. From there, the timestamp of the identified datapoints in the robot data stream can be mapped to the laser sensor data stream. By using the known measurement frequency of the laser sensor, the time stamp can be extrapolated onto the remaining laser sensor dataset. The second synchronization point taken at the end of a measurement cycle can then be used to verify the synchronization of the two datasets. Each measurement cycle for one layer of six beads takes around 20 minutes. The error in the synchronization approach (determined by the second synchronization point) was only in the range of 30 milliseconds. Figure 19 illustrates the continuous data stream from the robot and the identified synchronization points. As a synchronization movement, the robot moved to a position on the table and then translated down and up, while the laser sensor took measurements. The lowest captured points at the beginning and end of both datasets constitute the synchronization points.

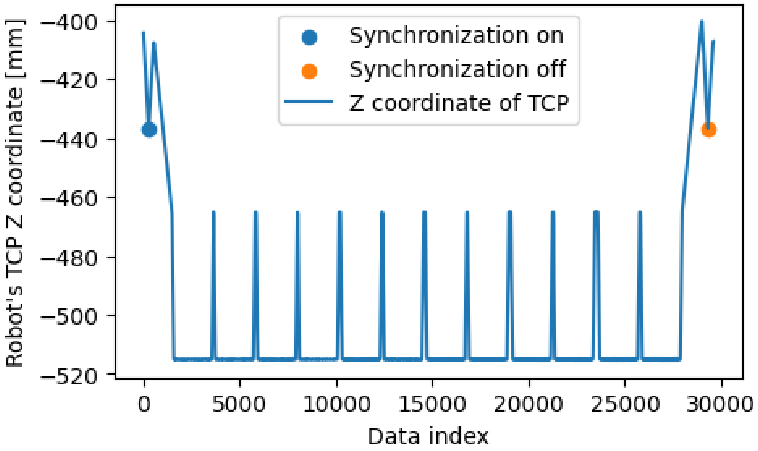


Figure 19: Example TCP data stream indicating the synchronization points

Thereafter, the measurements of the laser sensor can be added to the Z coordinates of the robot’s TCP position. This combination is referenced as Z* in the following sections. The resulting X, Y, and Z* dataset contains a point cloud of the measured surface.

From there, the sections of the dataset that contain the measurements of one cross-section have to be identified. To do so, the first step is to identify and assign data sections that measure a particular bead. This is needed to connect the bead’s cross-sections to the utilized welding settings in creating that bead. To help in identifying the data stream section that belongs to one particular bead, landmarks in the data stream are created by moving the robot upwards after the measurement of each bead. This constitutes an increase in the Z values of the robot’s dataset after each bead measurement cycle (s. spikes in Z in figure 19). So, the beginning and end of each bead measurement cycle can be found within the dataset by searching for local maxima in the Z data.

After the parts of the dataset that refer to one bead are identified, these parts have to be further divided into sections that belong to one cross-section. To do so, the robot movements in the XY plane can be used (s. figure 20). The bead data is rotated around the center of the corner to generate a data-set for each bead arm that is aligned with the X and Y axis of the robot coordinate system. By doing so, the local minima and maxima on the X-axis (s. red dots in figure 20) can be utilized to section the dataset into cross-section measurements (s. blue and yellow lines in figure 20). These cross-sectional dataset snippets constitute the part of the data from which the output features for the ANN are extracted.

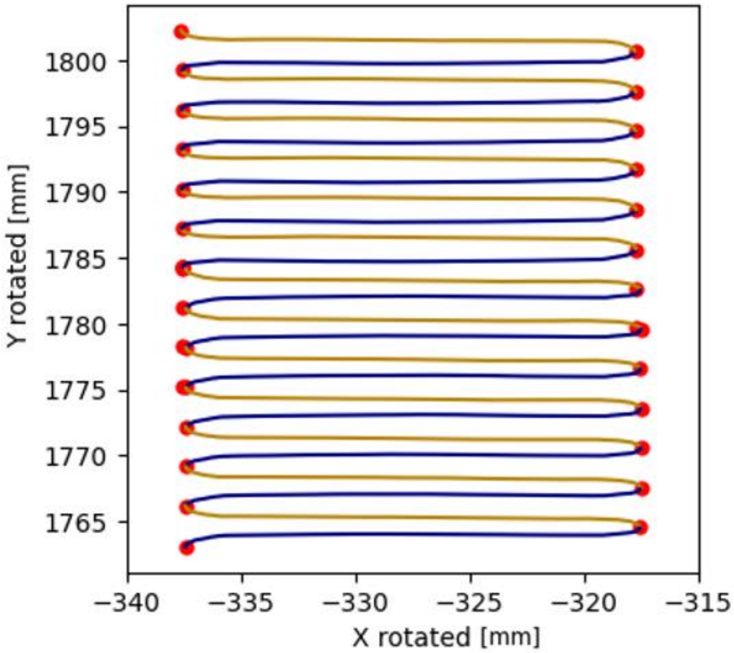


Figure 20: Identification of cross-sections in the data for one bead

Every step from this point onwards aims at purifying the bead cross-section data to create output features that resemble the actual geometry of the beads. As the laser sometimes fails to report a measurement, the first step in the data purification procedure is to interpolate the existing measurements for each cross-section. The result is a continuous dataset with a datapoint of the bead geometry every ~0.1 mm. Thereafter, measurements of the substrate to the side of each bead have to be removed from the continuous data stream belonging to one cross-section (s. figure 21 a)). This is done by fitting a cubic spline to the cross-section data, taking the derivative of the spline, and looking for rapid changes in the slope of the profile (s. figure 21 b)). These rapid changes indicate the beginning and end of the bead cross-section (s. grey points in figure 21 a.).

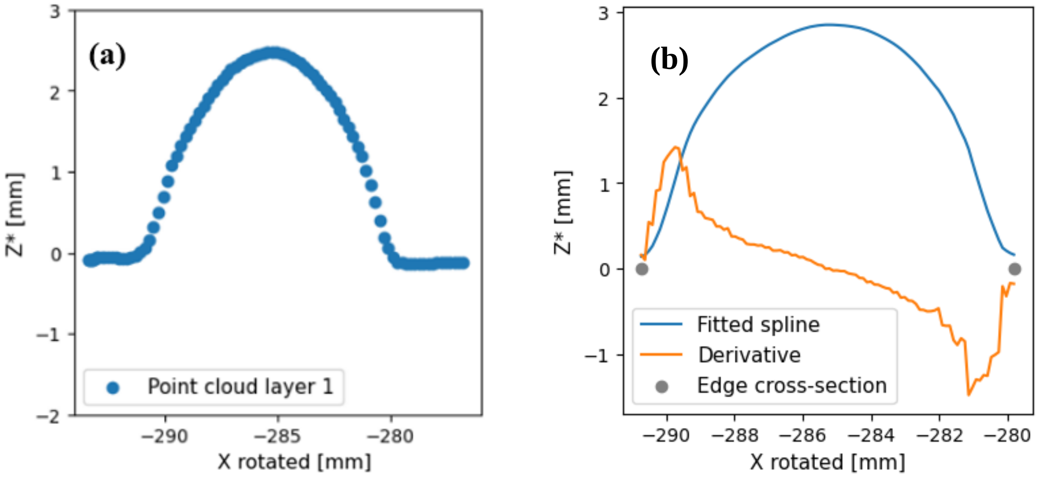
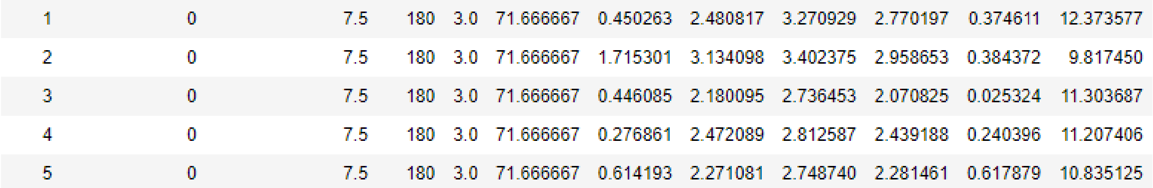


Figure 21: Allocation of data points to cross-section: a) shows the raw data, whereas b) demonstrates the spline fitting and edge identification

Thereafter, the center point at the corner of each bead is extracted from the robot trajectory program and used to cut cross-sections of each bead arm at the center line through the corner (s. figure 21). This ensures that cross-sections of one bead arm in the overlap region don’t contain data points of the other bead arm. Finally, after this step, the geometric features for the training of the ANN can be deduced. The BW is defined as the difference between the maximum and minimum in the rotated X coordinates of each cross-section. The BH is taken from the Z* profile data of each cross-section at 0%, 25%, 50%, 75%, and 100% of the BW and reduced by the volumetric BH of each previous layer. As the last step, the outliers and cross-sections with faulty measurements are removed. Outliers are identified by comparing one cross-section with the other nine cross-sections at the same location on the same bead but in a different layer. If a cross-section differs in its geometric features by more than 20% from the mean of the ten cross-sections at the particular location, it is labeled as an outlier and removed. For example, if the BW of the cross-section at a DtC of 20 mm is 30% larger than the mean BW of the cross-sections of the same bead at the same DtC, this cross-section is removed from the dataset. This approach integrates the observation that the BH and BW are not excessively changed by the layer number. Thus, cross-sections with excessive changes are more likely due to faulty measurements. Table 6 shows an excerpt of the final dataset.

Table 6: Excerpt of the dataset for the extended geometry prediction model



1	0	7.5	180	3.0	71.666667	0.450263	2.480817	3.270929	2.770197	0.374611	12.373577
2	0	7.5	180	3.0	71.666667	1.715301	3.134098	3.402375	2.958653	0.384372	9.817450
3	0	7.5	180	3.0	71.666667	0.446085	2.180095	2.736453	2.070825	0.025324	11.303687
4	0	7.5	180	3.0	71.666667	0.276861	2.472089	2.812587	2.439188	0.240396	11.207406
5	0	7.5	180	3.0	71.666667	0.614193	2.271081	2.748740	2.281461	0.617879	10.835125

4.3 Methodology for the variance prediction (study 3)

The variance prediction study makes use of the collected and purified data of study 2. By extracting the cross-sections that don’t lie in the corner region (OR=0), the standard deviation of each geometric feature for different welding settings and layers can be calculated. The calculated standard deviation for each feature serves as the output for the training of an ANN. This ANN is also trained using PyTorch and its hyperparameters are tuned through cross-validation and the Optuna library.

This concludes chapter 4 on the utilized methodology for the three studies. The data collection and data purification for study one are introduced and the approach is divided into 5 research steps. Section 4.2 described the data generation with the used WAAM setup, the data stream combination, and purification, as well as the research approach for this study. Section 4.3 outlines how the generated data is reused for study 3 to train an ANN for variance prediction.

CHAPTER 5. RESULTS

The following chapter presents and discusses the main findings of this work. Section 5.1 presents the findings of study 1 on the transferability of geometry prediction models. Section 5.2 deals with the results of the extended bead geometry prediction model. The generated dataset is described, the performance of the model is presented, the influence of the input features is assessed, and a inverse algorithm that uses the forward model to suggest optimized welding parameters is presented. Finally, Section 4.3 outlines the possibilities for predicting the variance of the WAAM process.

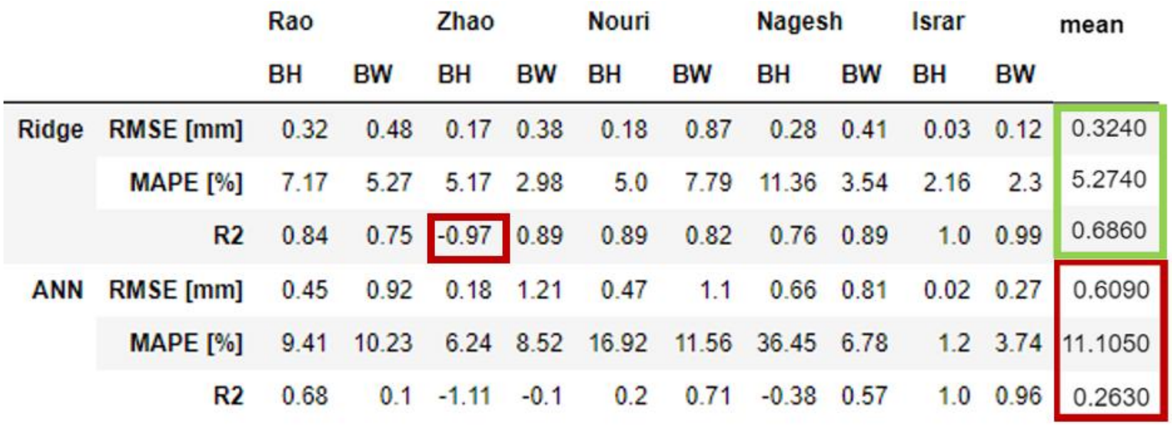
5.1 Transferability study (study 1)

The findings of the transferability study are presented according to the steps introduced in Section 4.1.1. The first step tests the performance of the bead geometry prediction models on each research testbed individually. Step two assesses the transferability of each individually trained model on the other research testbeds. Steps 3 and 4 try to improve the transferability by including additional welding settings and superordinate testbed characteristics. The final step extends the training loss function of the ANN with the principle of mass conservation and tests its influence on the transferability of the resulting model.

5.1.1 Train, optimize, and test ML models on each dataset individually

As a first step in the transferability study, a Ridge algorithm (s. Section 2.2.3) and an ANN (s. Section 2.2.3) were trained on the full feature space of every research testbed individually. This is done to verify the results obtained from each research testbed, to assess if similar results can be achieved with the available resources in this work, and to have a benchmark performance on each dataset to compare within the following investigations. The RMSE, MAPE, and the R2 score on each data set are listed in table 7.

Table 7: Performance of different algorithms on each dataset individually



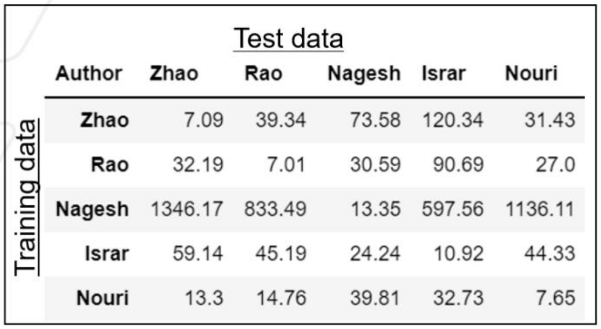
		Rao		Zhao		Nouri		Nagesh		Israr		mean
		BH	BW	BH	BW	BH	BW	BH	BW	BH	BW	
Ridge	RMSE [mm]	0.32	0.48	0.17	0.38	0.18	0.87	0.28	0.41	0.03	0.12	0.3240
	MAPE [%]	7.17	5.27	5.17	2.98	5.0	7.79	11.36	3.54	2.16	2.3	5.2740
	R2	0.84	0.75	-0.97	0.89	0.89	0.82	0.76	0.89	1.0	0.99	0.6860
ANN	RMSE [mm]	0.45	0.92	0.18	1.21	0.47	1.1	0.66	0.81	0.02	0.27	0.6090
	MAPE [%]	9.41	10.23	6.24	8.52	16.92	11.56	36.45	6.78	1.2	3.74	11.1050
	R2	0.68	0.1	-1.11	-0.1	0.2	0.71	-0.38	0.57	1.0	0.96	0.2630

The Ridge algorithm has the best overall performance with a mean MAPE of 5.3 % and an R2 score of 0.69. The algorithm shows good performance on all datasets, the only exception being the predictability of the BH in the dataset of Zhao et al. (2020). While the ANN still achieves acceptable results with an error of around 11 % and is thus in the range of results achieved in the papers, it performs worse than the linear model. In addition, it was found that the performance of the ANN is highly dependent on the part of the datasets used for training. This could explain the slight difference in the performance of the ANN trained by Venkata Rao et al. (2022) (6.45%) versus the ANN trained in this study (9.82%) while using the same hyperparameters.

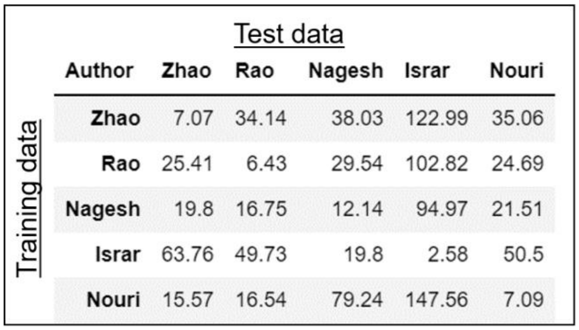
5.1.2 *Train, optimize, and test ML models on each dataset individually and use only fully transferable input features*

In the second step of the transferability study, the algorithms are trained and optimized on the transferable features of all datasets. Transferable features are features that are varied in all datasets, which, in the current work, are limited to TS and WFS. The algorithms are then used to predict the BH and BW of the remaining five datasets. The average MAPE on BH and BW for the Ridge and ANN models are listed in table 8.

Table 8: Average MAPE of Ridge (left) and ANN models trained on WFS and TS on one dataset



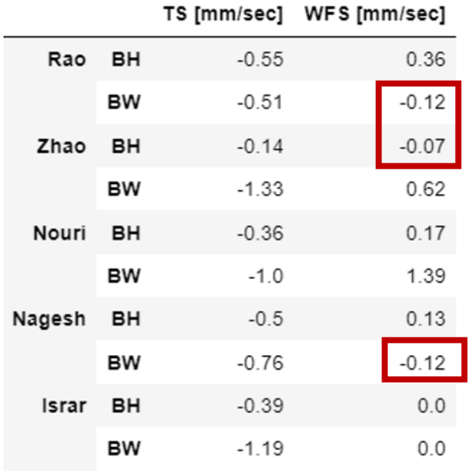
		Test data				
	Author	Zhao	Rao	Nagesh	Israr	Nouri
Training data	Zhao	7.09	39.34	73.58	120.34	31.43
	Rao	32.19	7.01	30.59	90.69	27.0
	Nagesh	1346.17	833.49	13.35	597.56	1136.11
	Israr	59.14	45.19	24.24	10.92	44.33
	Nouri	13.3	14.76	39.81	32.73	7.65



		Test data				
	Author	Zhao	Rao	Nagesh	Israr	Nouri
Training data	Zhao	7.07	34.14	38.03	122.99	35.06
	Rao	25.41	6.43	29.54	102.82	24.69
	Nagesh	19.8	16.75	12.14	94.97	21.51
	Israr	63.76	49.73	19.8	2.58	50.5
	Nouri	15.57	16.54	79.24	147.56	7.09

The diagonals of table 8 depict the performance of the algorithm on the test dataset from the same research testbed. When compared with the findings from study 1, it becomes apparent, that the performance didn’t deteriorate much by only using TS and WFS. It can be concluded that these two features have the highest influence on the prediction. This is reasonable, as these two features determine the deposited material volume per second and per location (Phillips, 2016). The scores next to the diagonal show the average MAPE on the other datasets. Here, the algorithms show mixed performance. The algorithm trained on the dataset of Nagesh and Datta (2002) performs poorly, with a MAPE as high as 1346 %. This extreme deviation can be explained by the small variance of WFS used in the experiments of Nagesh and Datta (2002). On top of the fact, that the prediction on the other datasets will be in the extrapolation range of the trained algorithms, the standard scaler of scikit-learn will increase the range of WFS in the other datasets and thus lead to very large predicted BH and BW. The linear algorithm trained on the dataset of Nouri on the other hand achieves an average MAPE on the other four datasets of ~26%. Table 5 shows the linear influence of the WFS and TS according to the optimized Ridge algorithms.

Table 9: Linear influence of WFS and TS on BH and BW when trained on one dataset



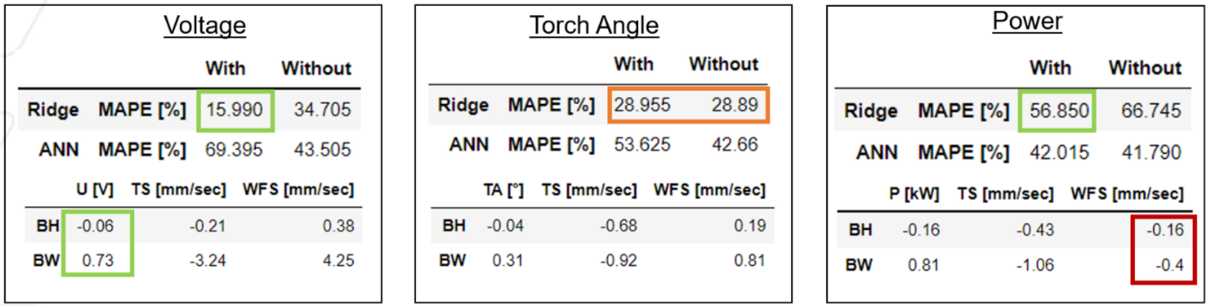
		TS [mm/sec]	WFS [mm/sec]
Rao	BH	-0.55	0.36
	BW	-0.51	-0.12
Zhao	BH	-0.14	-0.07
	BW	-1.33	0.62
Nouri	BH	-0.36	0.17
	BW	-1.0	1.39
Nagesh	BH	-0.5	0.13
	BW	-0.76	-0.12
Israr	BH	-0.39	0.0
	BW	-1.19	0.0

While TS has a clear negative influence as expected, WFS is negative as well in some instances. This is physically incoherent as a higher WFS means that more material is deposited (Phillips, 2016). This is a sign that the algorithms in those cases overfitted the data of the particular research testbed.

5.1.3 *Train, optimize, and test ML models on a subset of the available datasets to include additional input features*

The input features of voltage, torch angle, and power were varied in multiple research testbeds. This makes their influence on the transferability assessable. The third step of the transferability study is thus to include these features and train the algorithms on all research testbeds where the features were varied. The transferability of the trained model is then tested on the remaining datasets, where these features were kept constant. Table 6 shows the average MAPE score of the optimized Ridge and ANN models on the test dataset with and without the additional features. Only the addition of voltage significantly improved the transferability while keeping physically coherent feature coefficients. Here a higher voltage increases BW while decreasing BH as predicted in the literature (cp. Phillips, 2016). The inclusion of power as an input feature slightly improved the transferability of the trained models, but the feature coefficients in the linear model show a physically inconsistent relationship, as, according to the model, an increase in power would reduce both the BH and BW and thus the amount of deposited material. Based on these findings, this investigation recommends the inclusion of voltage in future studies on model transferability.

Table 10: Influence of additional features on transferability (top) and coefficients of the features in the Ridge algorithm (bottom)



Voltage

		With	Without
Ridge	MAPE [%]	15.990	34.705
ANN	MAPE [%]	69.395	43.505

	U [V]	TS [mm/sec]	WFS [mm/sec]
BH	-0.06	-0.21	0.38
BW	0.73	-3.24	4.25

Torch Angle

		With	Without
Ridge	MAPE [%]	28.955	28.89
ANN	MAPE [%]	53.625	42.66

	TA [°]	TS [mm/sec]	WFS [mm/sec]
BH	-0.04	-0.68	0.19
BW	0.31	-0.92	0.81

Power

		With	Without
Ridge	MAPE [%]	56.850	66.745
ANN	MAPE [%]	42.015	41.790

	P [kW]	TS [mm/sec]	WFS [mm/sec]
BH	-0.16	-0.43	-0.16
BW	0.81	-1.06	-0.4

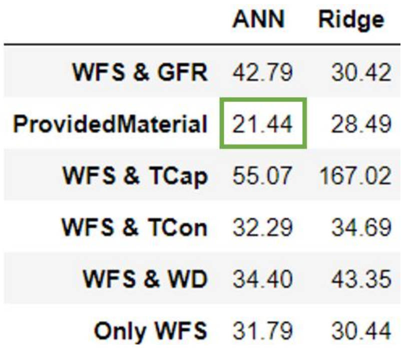
5.1.4 Train, optimize, and test ML models on all but one dataset and assess the transferability onto the remaining dataset

In the fourth step of the transferability study, the impact of superordinate features is studied. Superordinate features are experiment characteristics that are kept constant within all research testbeds reported in the literature (including the testbed used in the current work). By training the models on multiple research testbeds the influence of the features on the prediction can be assessed as well. Superordinate features include the wire diameter, the gas flow rate, and the utilized work material. As introduced in Section 2.1.2, the thermal characteristics of the material play a major role in the welding process. Thus, the assessability of the influence of the utilized material is enabled by using the specific heat capacity (TCap) and the heat conductivity (TCon) of the material as inputs to the model. The influence of WD is assessed once by using it directly as input and once by using it together with the WFS to calculate the input or provided material per second (PM) in each study. The provided material per unit of time is calculated according to eq. 8, where the cross-section of the wire is multiplied by the WFS.

$$\frac{Provided\ Material}{sec} = \frac{WD^2}{4} * \pi * WFS \qquad \text{(eq. 8)}$$

The models are trained on data from all but one research testbed and tested on the data from the remaining testbed. The superordinate features are each combined with TS and WFS. Table 11 shows the average MAPE of the trained algorithms.

Table 11: Average MAPE [%] of algorithms, trained on TS and the features shown in the table’s row index, on all but one dataset

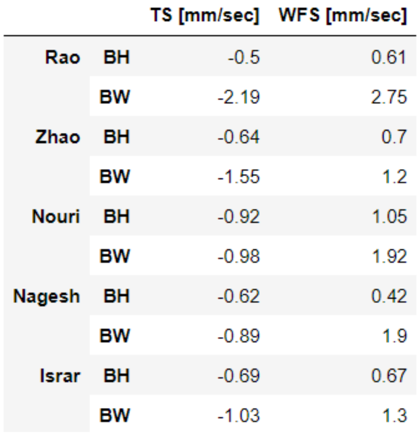


	ANN	Ridge
WFS & GFR	42.79	30.42
ProvidedMaterial	21.44	28.49
WFS & TCap	55.07	167.02
WFS & TCon	32.29	34.69
WFS & WD	34.40	43.35
Only WFS	31.79	30.44

Of all the feature combinations the one using the Provided Material feature scored the best average MAPE. As this feature includes the WD and is thus more meaningful to the bead geometry, it enhances model transferability. Because of this, PM will be used throughout this work instead of WFS. It is also seen that the ANN performed better than the Ridge algorithm for the first time throughout the transferability study. The larger amount of data made it possible to leverage the potential of the ANN to learn more complex relationships.

Finally, the feature coefficients of the Ridge algorithm trained on WFS and TS on all datasets are given in table 12. The coefficients are physically coherent over all dataset combinations when trained on multiple datasets. This speaks for better transferability and less overfitting of the algorithm when compared to the algorithms trained on only one dataset (s. step 2).

Table 12: Feature coefficients when trained on multiple datasets



		TS [mm/sec]	WFS [mm/sec]
Rao	BH	-0.5	0.61
	BW	-2.19	2.75
Zhao	BH	-0.64	0.7
	BW	-1.55	1.2
Nouri	BH	-0.92	1.05
	BW	-0.98	1.92
Nagesh	BH	-0.62	0.42
	BW	-0.89	1.9
Israr	BH	-0.69	0.67
	BW	-1.03	1.3

5.1.5 Train, optimize, and test ML models on all but one dataset with PIML and assess the transferability onto the remaining datasets

For the training of the model in the last step of the transferability study, the loss function was adapted to include the principle of mass conservation. Eq. 9 to eq. 11 depict the calculation of the loss function. N is the number of samples per training batch, and λ determines the weight for each loss. The parameter λ is an additional hyperparameter for training the model. $Loss_{classic}$ constitutes the RMSE for predicting BH and BW, whereas $Loss_{physics}$ calculates the RMSE of the predicted deposited material relative to the provided material per cross-section. The overall loss function is a function of $Loss_{classic}$ and $Loss_{physics}$ and is defined in eq. 11.

$$Loss_{classic} = \sum_i^n \left(BH_{i,pred} - BH_{i,act}\right)^2 + \left(BW_{i,pred} - BW_{i,act}\right)^2 \quad \text{(eq. 9)}$$

$$Loss_{physics} = \sum_i^n \left(BH_{i,pred} * \frac{BW_{i,pred}}{2} * \frac{\pi}{2} - \frac{PM_i}{TS_i}\right)^2 \quad \text{(eq. 10)}$$

$$Loss = \lambda * Loss_{classic} + (1 - \lambda) * Loss_{physics} \quad \text{(eq. 11)}$$

To study the influence of the extended loss function, each dataset from one research testbed was once put aside and cross-validation was performed on the remaining datasets. The cross-validation was used to determine the optimal hyperparameters and to choose the best value for λ. Within the cross-validation, each dataset was once put aside for validation, while the other datasets were used for training. It became apparent, that the performance on the test set was highly dependent on the chosen hyperparameters, which made it difficult to draw a clear conclusion regarding the influence of the extended loss function. For some parameter combinations, the model including mass conservation performed better for a particular research testbed as the test set, while for some it performed worse. So, to make the influence of mass conservation in the loss function assessable, 100 different models were trained for each research testbed in the test set - once with and once without the extended loss function. This means, that each testbed was once put aside and the remaining testbeds are used 200 times to train models with 100 sets of different hyperparameters. Each of these 100 sets of hyperparameters was once used for the training of a model with and for a model without mass conservation in the loss function. Figure 22 plots the error on one test set for each of the 100 hyperparameter sets. When looking at the plot it becomes evident, that the hyperparameters of the model play a bigger role than if mass conservation is applied or not.

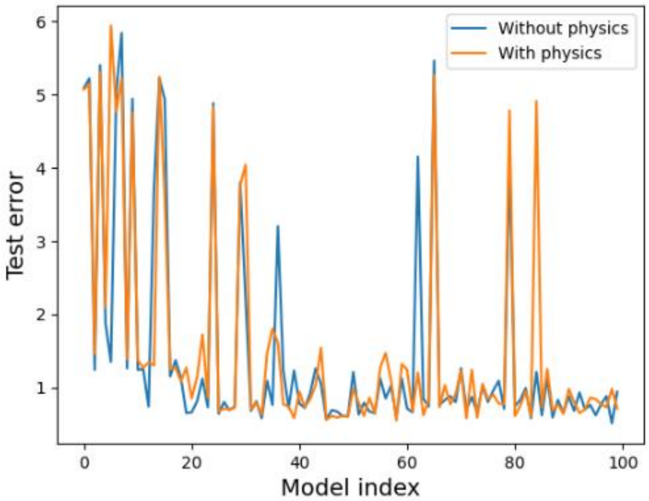
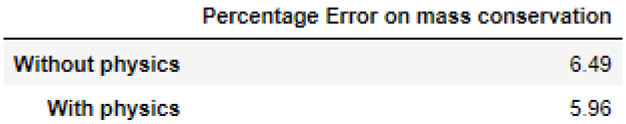


Figure 22: Comparison of prediction error on the test set for models trained with and without mass conservation in the loss function

That the inclusion of the mass conservation in the loss function doesn’t have a large impact can also be seen when comparing the volume of the predicted bead geometry to the provided material volume. In theory, the models with the extended loss function were trained in a way to minimize this error. Still, the error of the physical models turned out to be only 0.5% smaller compared to the models trained with only the BW and BH error in the loss function (s. table 13).

Table 13: Model performance regarding mass conservation



	Percentage Error on mass conservation
Without physics	6.49
With physics	5.96

A potential explanation could be that the error in mass conservation is very similar to the error in BH and BW, as the best way for a model to reduce both errors is to learn a very accurate prediction on BH and BW. This points to the conclusion that for PIMT to improve the transferability of a ML model the included process physics must describe a more complex relation with the predicted output parameters than is accounted for by the mass conservation principle applied here.

Finally, a model with and one without the extended loss function was trained and optimized on all research testbeds taken from the literature and used to predict the mean BH and BW of the straight beads produced in study two of this work. The error of the physical model proved to be 1% smaller than the error of the classical model, which is also assumed to be statistically insignificant. The error of the two models is in the range of 10% and thus a lot better than the results from study four. This might be because one more dataset was available for training. Figure 23 shows the predicted BH and BW of the two models versus the measured values. The travel speed increases with ascending sample number, whereas for each TS three different WFS are plotted. The plots show that both the physical and the standard ML models have learned the trend of increasing BW and BH with increasing WFS, and the decrease in the size of the geometric features with increasing TS. The model shows this good performance even though different types of steel were used in the training and testing data. If the model also performs well on other metals, like aluminum, still needs to be tested.

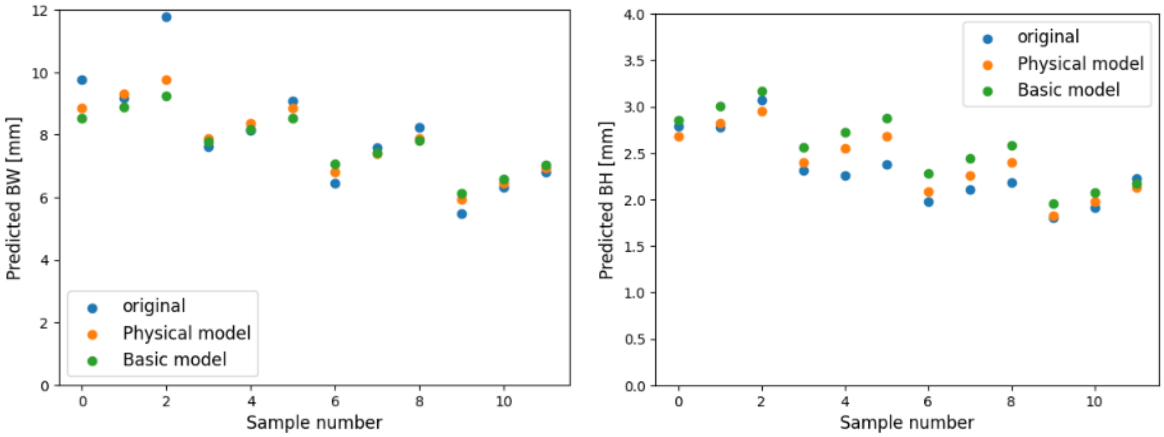


Figure 23: Predicted BH and BW versus measured on own data

The architecture and the weights of the final model with the best transferability on the beads produced for study 2 can be found in the appendix of this work. With this information, the ANN can be reproduced and used on new research testbeds and thus can avoid the need to execute the kind of geometry prediction studies conducted in the analyzed papers.

The study on transferability is summarized in the following. Step 1 of the transferability study showed that, in general, good results can be achieved for geometry prediction with ML. It was shown, that linear models tend to perform better if the available data is as limited as in the examined research testbeds reported in the literature. Step two proved the overfitting of each model to its particular research testbed, and showed the relevance of TS and WFS. In step three the input feature space was extended by voltage, power, and torch angle. Only voltage helped to improve the transferability while producing physically consistent results. Step four tested the inclusion of superordinate experiment characteristics. The only improvement that could be seen was for enhancing WFS with the wire diameter to match the provided material per second. The final step tested the inclusion of simple process physics on model transferability. Contrary to expectation, the use of the extended loss function didn’t lead to a significant improvement in model transferability. It has to be pointed out, that in total the model transferability study was only conducted on five external research testbeds. The code and the collected data can be provided upon request to facilitate other transferability studies when more bead geometry data becomes available.

5.2 Extended bead geometry prediction study (study 2)

The extended bead geometry prediction study analyzed the possibility to predict the shape of beads for different layers and different corner angles. This section is structured in the following way. At first, the collected research data is introduced and its limits and key characteristics are described. Section 5.2.2 describes how the model is trained and depicts its performance on the testing data. Thereafter, a feature study is conducted to analyze the influence of each input feature. Finally, in Section 5.2.4, a backward approach is described that extracts welding parameters from the trained model that can lead to an optimized, smoother bead surface at corners.

5.2.1 Data overview

As a first step, an error estimate of the combined robot and laser measurement system was obtained. To do so, the substrate before the deposition was measured with the laser in the regions of deposition. As the substrate can be assumed to be flat, the measured surface profile provides an error measurement for the utilized robot – laser sensor combination. Figure 24 a) shows the reported robot position while measuring the cross-sections of one bead and figure 24 b) the deviation of the measurements from a perfectly planar surface. While the absolute error of the reported robot position in the area of a cross-section was up to 0.4 mm (s. figure 24 a), the laser measurements could correct this error down to a maximum of 0.14 mm (s. figure 24 b). The remaining error is most likely due to inaccuracies in the reported positions of the robot, as it is in the same order of magnitude as the robot's repeatability.

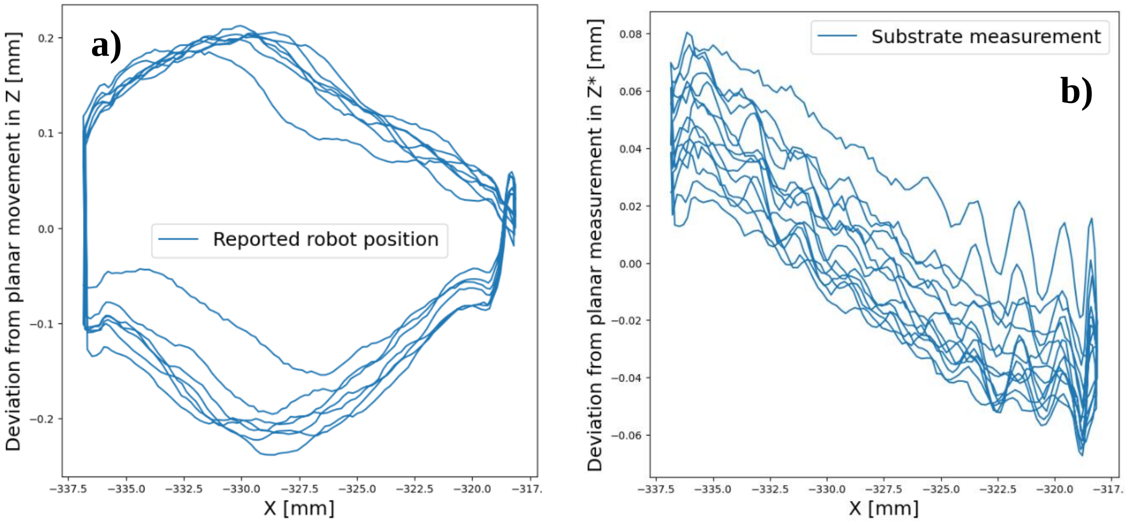


Figure 24: Reported robot position and surface profile when measuring the substrate in the region of a bead

Thereafter, the data set is visualized and thoroughly examined. As the dataset consists of over 28000 cross-sections, not the whole dataset could be checked. Figure 25 shows the process from raw data towards the geometric features used for training. The substrate measurements, depicted in the original data figure (s. figure 25 a), show signs of minor distortion. It is also evident, that in some cases the laser sensor has problems reporting data points at the edges of a bead cross-section (s. layers 7 & 10 in figure 25 b). This effect appears to become more frequent for the upper layers when no substrate measurements are possible anymore due to the limited range of the laser sensor. A picture of an extreme case can be found in the appendix of this work. Keyence confirmed that this can be the case under certain measurement conditions.

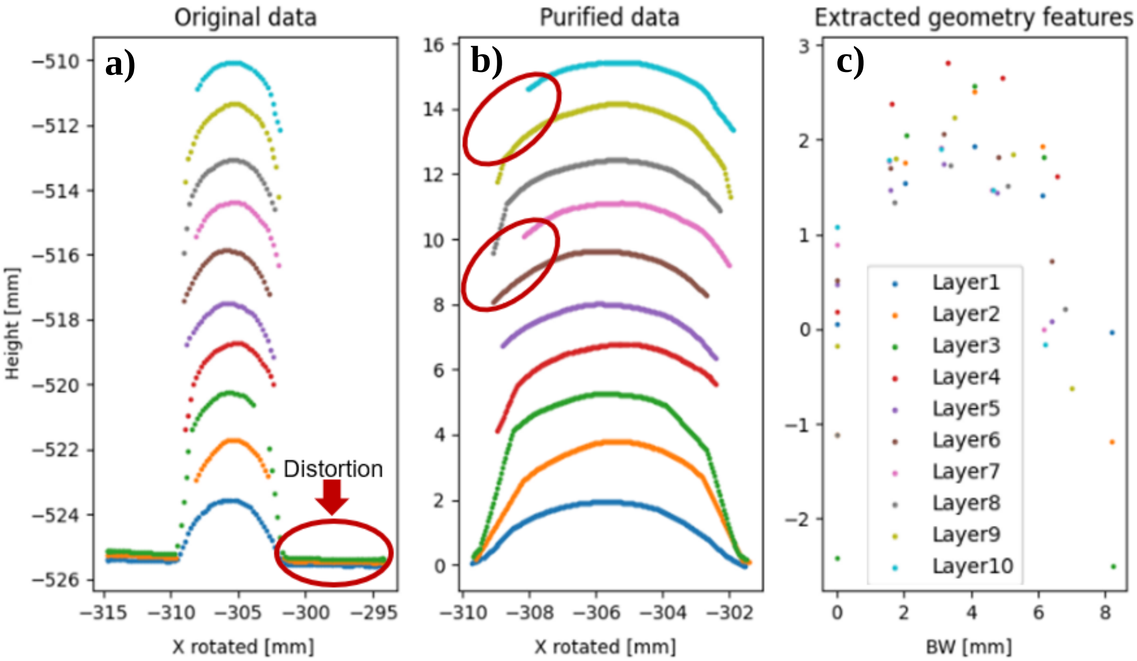


Figure 25: Visualization of data purification and feature extraction. The highlighted region in a) indicates substrate distortion, in b) the lack of laser sensor data at the edges of bead cross-sections.

Through the visualization of the data, an interesting effect at the corners is noticeable. For small corner angles and large deposition volumes (low TS and high WFS) the material in the overlap region of the corner shifts periodically with the layer number (s. figure 26). As described in Section 4.2.1, the deposition direction was altered in every layer to mitigate the influence of the arc-striking and arc-off. In figure 26 layers with an odd number (layer numbers 1,3,...) show the cross-section of the corner bead arm that was deposited first. It can be seen that the material here extends further away from the center of the bead. A reason for this material shift could be that for small angles the first deposited bead arm stays hot longer as the torch stays in its vicinity when depositing the second bead arm. This would give the melt pool more time to distribute. For smaller deposition volumes a change in the position of the corner maximum can be identified (s. figure 26 b). These effects of material shifting suggest the inclusion of deposition direction in future studies of corner deposition in WAAM.

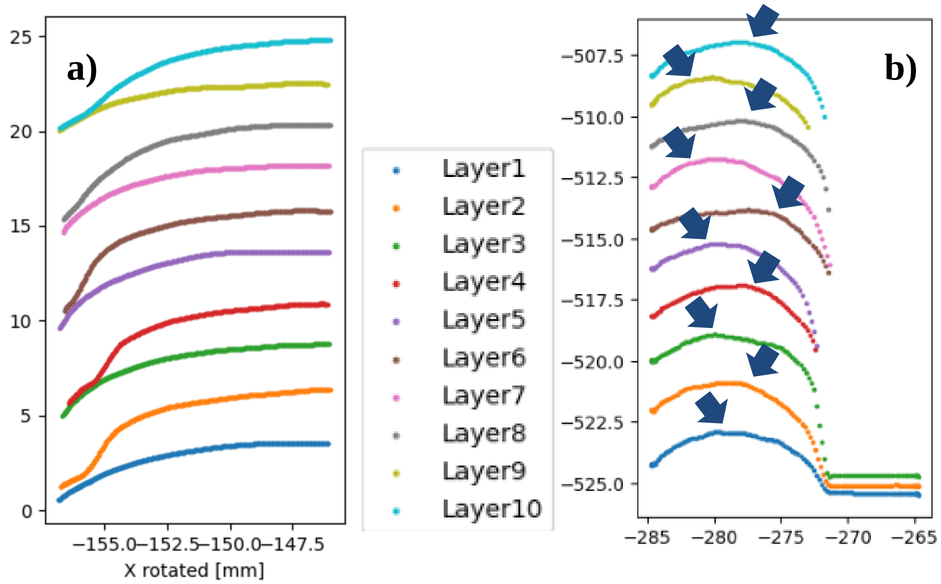
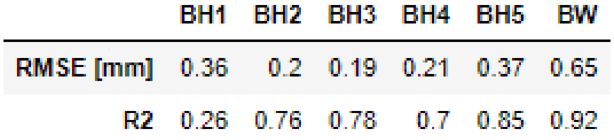


Figure 26: Periodic material shift in the bead corners depending on deposition direction

5.2.2 Performance

For the extended geometry prediction model, an ANN was trained with the layer number, TS, WFS, DtC, and the Boolean descriptor for the overlap region in the corners as input features. The output features used in the model consisted of the five BH measurements and the measurement of the BW. As described in Section 4.2.1, the hyperparameters of the ML model were trained and optimized using the Optuna library. To find the optimal hyperparameters, over 200 models were trained. For every model, Optuna varies the hyperparameters and records the performance of the model on the validation dataset. The best model had an R2 score of 0.65 and an RMSE as low as 0.42 mm on the test dataset. To assess the influence of the corner overlap region feature and the layer number, Optuna was used to optimize the same number of models excluding each feature at a time. While the R2 score was lowered by 0.06 for the best model when excluding the layer number, the predictability of the model increased by 0.05 when dropping the overlap region as an input feature. Correlated input features can decrease training efficiency for an ANN as the model weight determination can get unstable (Matignon, 2005). Apparently, the correlation of this feature with the input features of corner angle and DtC rather reduces training efficiency than adding new information. The model excluding the corner overlap region reduced the RMSE on BW predictions to 0.65 mm and the RMSE on all BH features to a mean of 0.27 mm (s. table 14). Especially the prediction of BH1 and BH5 increases the mean RMSE on BH prediction. Here a lot of variance is introduced by the previously described measurement error of the laser. Figure 27 shows the plot of the loss function over the number of epochs used for training. Small signs of overfitting can be seen, as the error of the test dataset is constantly above the error on the training dataset but overall the errors are in the same range.

Table 14: Model performance on output features



	BH1	BH2	BH3	BH4	BH5	BW
RMSE [mm]	0.36	0.2	0.19	0.21	0.37	0.65
R2	0.26	0.76	0.78	0.7	0.85	0.92

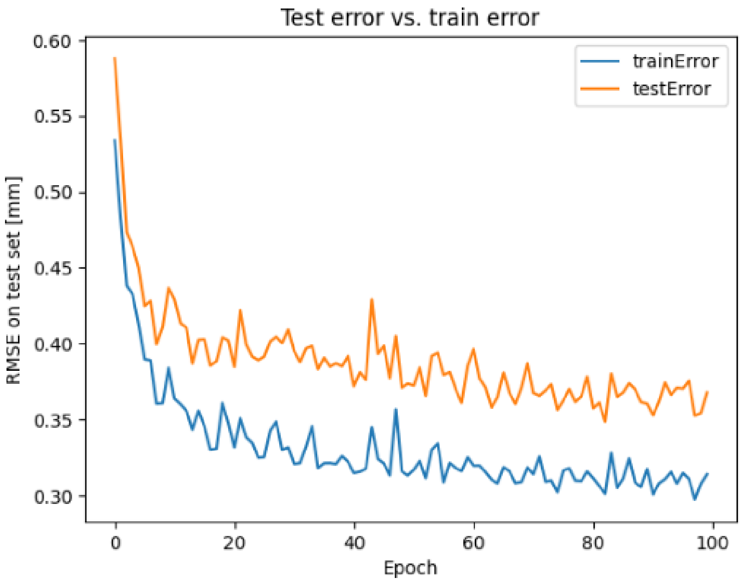


Figure 27: Training and testing errors of the extended geometry prediction model during training

According to the mean R2 score, 70% of the data variance can be explained by the model. The remaining 30% can partly be explained by the inaccuracies of the robot – laser sensor combination for measurements (s. Section 5.2.1). Another factor is certainly the depicted material shifting effect in Section 5.1.1 that is not accounted for in the current feature space. In addition, the variability of the GMAW process itself certainly contributes to the remaining 30% of the unexplained variance. Finally, the fact that the edge of the bead is sometimes not captured by the laser sensor (s. Section 5.2.1) contributes two-fold to the unexplainable variance. On the one hand, it directly influences the recorded values of BW, BH0, and BH5. On the other hand, as BH is measured relative to BW, the reduction in BW entails a measurement of the BH features at different points of the cross-section. In future studies, this correlation between the output features might be avoidable by using independent geometric features like contact angle, BH maximum, or bead curvature.

5.2.3 *Feature study*

In the featured study, the influence of each input feature is assessed by creating a fictional data set and predicting on this data with the trained model. Each dataset is made up in a way that all features but the feature of interest are kept constant. The influences of TS and PM are depicted in figure 28. The basic influences of each feature are correctly learned by the model. An increase in PM increases the cross-section area whereas TS reduces it. In both cases, some overfitting can be noticed for BH5. Apparently, in the dataset, a high TS or high PM appears more often with lower values of BH5.

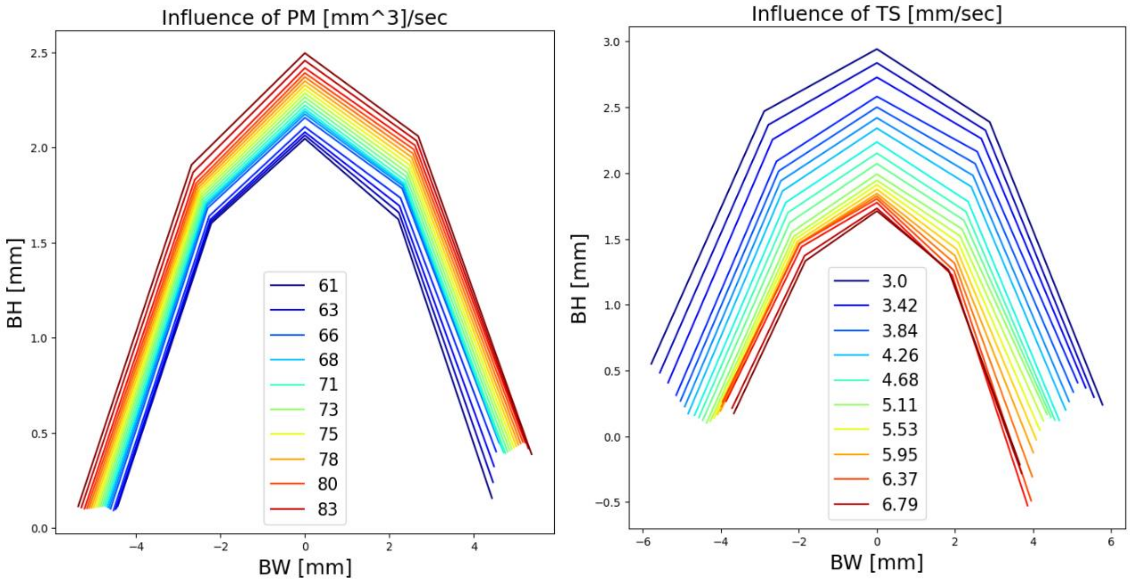


Figure 28: Influence of PM and TS on the bead cross-section geometry

The influence of DtC is illustrated in figure 29. The three graphs plot the predicted development of the cross-section geometry in the corner region for three different angles. It can be seen, that the influence of DtC decreases with increasing corner angles and vanishes for straight beads. The model also accurately represents the gradual decline in BW and the increase in BH5 when approaching a corner. Interestingly, the model learned a decrease in BH for the last few cross-sections in the corner region for angles smaller than

60°. This correlates with the creation of porosity described in Section 3.2. The smaller the angle, the larger the overlap region and the larger the area of missing material behind the corner (s. figure 11 in Section 3.2). So, it is logical that for small corner angles more material needs to flow in the area of missing material, which will reduce the BH in the last cross-sections.

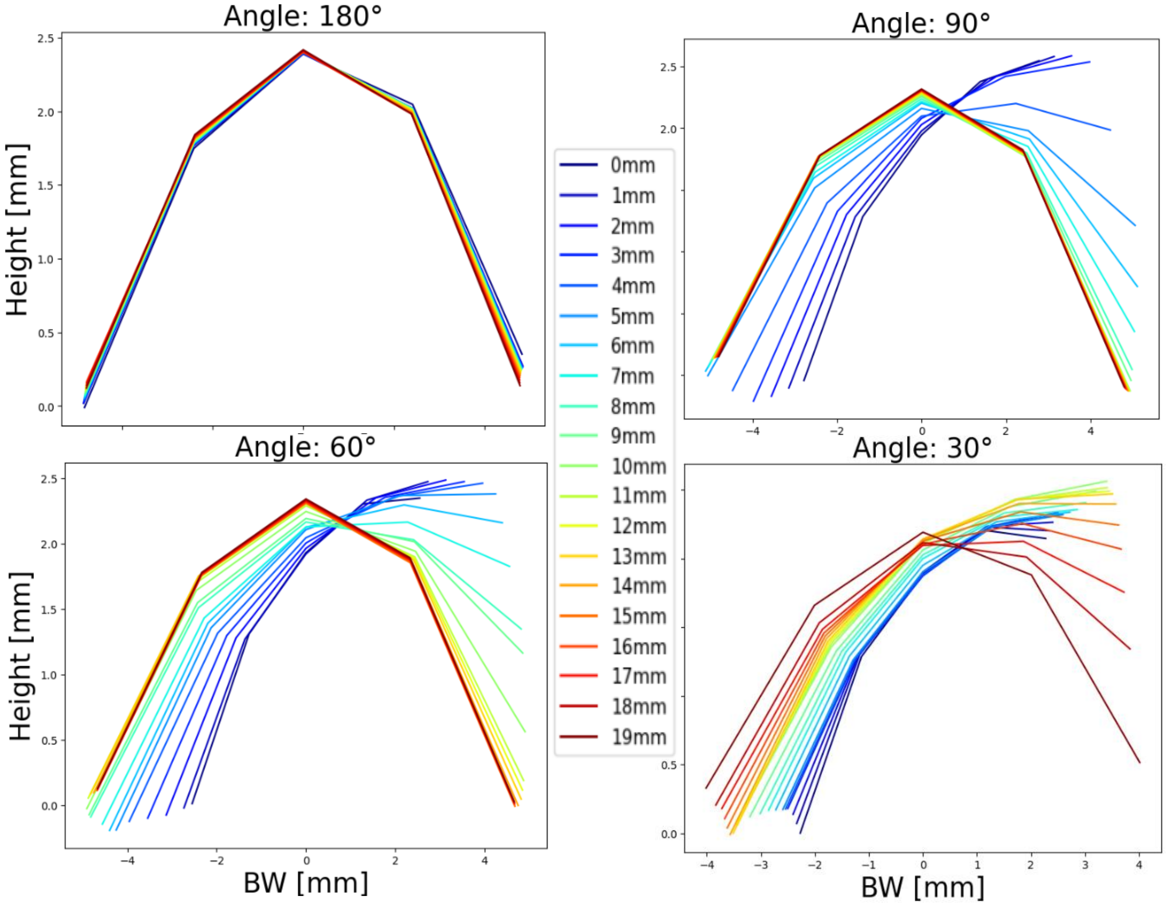


Figure 29: Influence of different angles and "distance to corner"

Finally, the influence of the layer number was analyzed. Unfortunately, the findings here are highly affected by the laser sensor measurement error. Layer three is usually the last layer where the substrate is within the range of the laser sensor. For layers above layer three, no measurements are obtained before the sensor arrives at the edge of the bead surface, and then sometimes the first few points at the edge of the bead surface aren't recorded. Thus, the model highly underestimates BW for these layers (s figure 30).

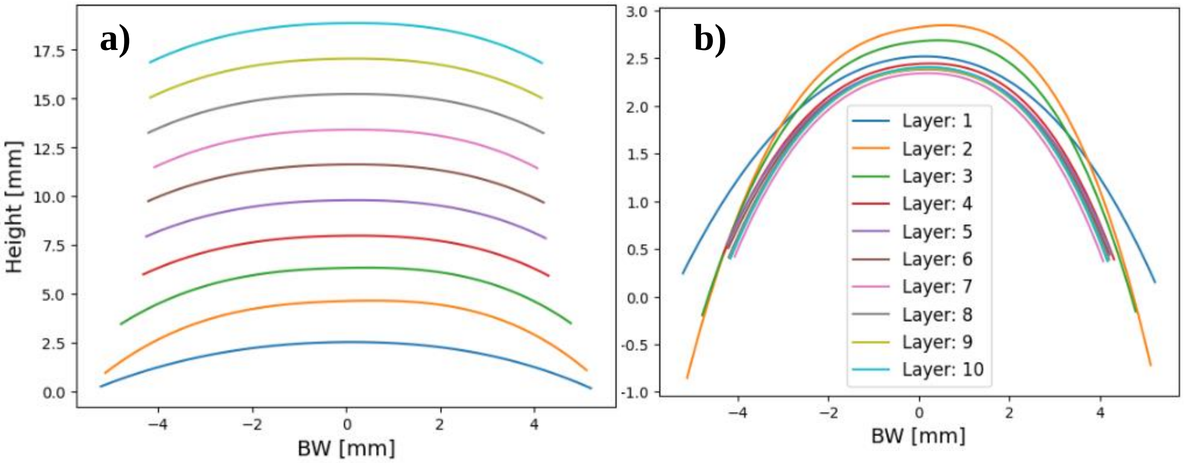


Figure 30: BH and BW prediction of the extended geometry prediction model – stacked (by placing predicted cross-section on top of the volumetric mean of the previous layer) (a) and unstacked as predicted (b)

Still, some relevant findings can be deduced from the feature influence plots. For once it becomes apparent that for the layers with accurate BW measurements, the widest point of the layer lies below the volumetric mean of the previous layer. This can be seen in the negative BH measurement at the edges. The BH measurements in the middle of the cross-section show a peak at layer number 2 before transitioning to a relatively constant value for the layers after layer number 4. This coincides with the approach of Ding, He, et al. (2021) to measure the cross-section of beads after layer number four. Unfortunately, the correct quantitative change of the bead shape, and thus an enhancement of the existing geometrical models of other research testbeds, can't be determined from these findings.

5.2.4 Backward algorithm

The objective of the backward algorithm is to extract WAAM process parameter suggestions from the trained ML model. To do so, a dataset with the predicted geometries for a large set of different input features was created. To test the approach and to reduce the feature space, the initial dataset was created for an angle of 60º and layer number 1. The feature range of the input features PM and TS was partitioned into 100 levels, whereas the resolution of DtC was chosen to be 0.5 mm for a range of 0 to 20 mm. All possible combinations of these three features together with the angle of 60º and the layer number 1 were fed to the trained model. Thereafter, the desired geometry is defined for the corner. To do so, the corner is divided into three regions following Ding, Zhao, et al. (2021). The first region describes the straight parts of the bead before entering the overlap region. The second region contains the parts of the overlap region where the material between the two bead arms is not enough to fill the enclosed valley. According to the BOM (s. Section 3.2), the valley is filled when the distance between the two bead arms is less than 0.738 times the BW (Ding et al., 2015). The BW of the desired geometry is arbitrarily set to 10mm. This results in the following three regions along the corner (s. eq. 7):

- Region 1: straight beads [20 mm, 18.5 mm]
- Region 2: overlap region with unfilled valley [18.5 mm, 14.5 mm]
- Region 3: overlap region with filled valley [14.5 mm, 0 mm]

The maximum BH of the intended geometry is set to 2 mm. The cross-sections in region 1 are approximated by a half-ellipse. BH4 and BH5 of the cross-sections in region 1 are set to be equal to a maximum BH of 2 mm, which would result in a flat surface in the corner region. The BH features of the cross-section in region 2 linearly interpolate the values of regions 1 and 3. The BW in regions 1 and 2 is reduced as the bead arms partially overlap. To calculate the reduced BW eq. 12 is used. The resulting cross-sections are shown in figure 32 a).

$$BW_{Overlap} = \tan\left(\frac{angle}{2}\right) * DtC + \frac{BW}{2} \qquad \text{(eq. 12)}$$

After the desired geometry is defined, the geometry features of the cross-sections are compared with the data set. The inverse algorithm first compares the cross-section of the desired geometry at DtC = 0 with the data set. The RMSE for each of the predicted geometries is calculated and five predictions with the lowest deviation are chosen. From these five predictions, the parameter combination with the lowest TS is chosen as a suggestion for this cross-section. This is done to minimize the dynamic constraint of the robot at the corner, which results from the inertia of the robot and the attached tools. Thereafter, the algorithm propagates through the other cross-sections and calculates the difference between all predicted geometries and the desired geometry. Here, from the five parameter combinations with low deviation, the parameter combination closest to the previous parameter combination is chosen. This is done to minimize the need for sudden changes in the TS of the robot. Figure 31 illustrates the suggested parameter combination by calculating the provided material per cross-section ($\frac{PM}{TS}$). The same figure also plots the material in the predicted and the material in the desired geometry cross-section. An interesting observation is that for the cross-sections in region 1 ($DtC > 18.5$) the provided material agrees with the deposited material according to the model. This can be seen as a sign that the model has learned the relationship between the bead deposition parameters and the deposited material fairly accurately for straight beads. For the overlap region 2, the area of the predicted and the desired geometry slightly increases due to linear interpolation in generating the desired geometry. Here the algorithm seems to understate the volume of necessary material. This can be because the excess material from region 1 partly flows into the enclosed valley in region 2. Finally, for region 3 the algorithm suggests combinations of TS and PM in a way that at first less material is provided than needed ($5\ mm < DtC < 14.5\ mm$) and then increases the material provided for the remaining cross-sections. Here, an increase in material deposition makes sense to fill the area of missing material at the peak of the corner. So, the algorithm seems to have learned that the material in the last few cross-sections partly flows in the region of the missing material. So, the reduction of material at the beginning of corner region 3 compensates for material accumulation, whereas the increase at the end has the potential to mitigate the porosity problem for acute corner angles.

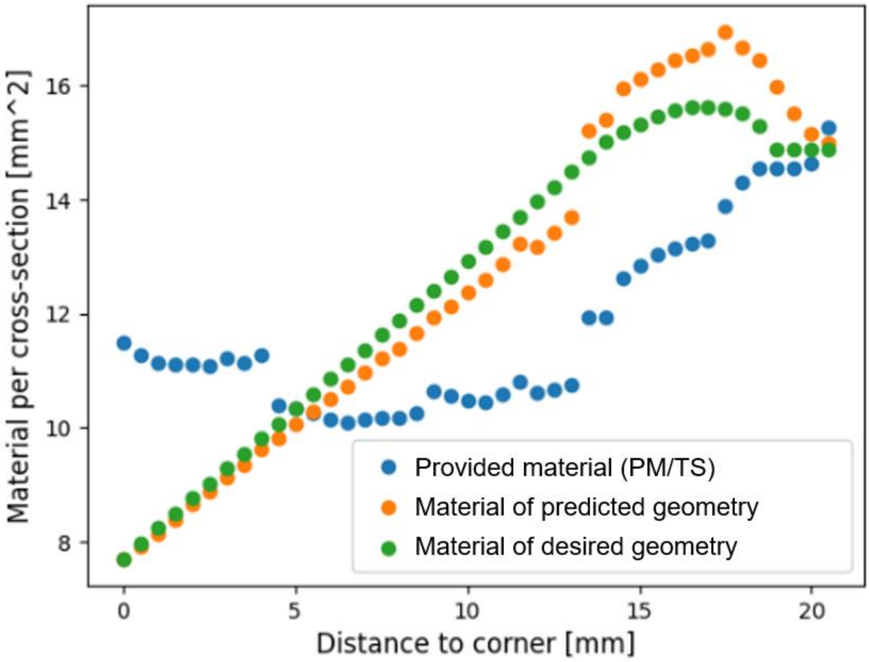


Figure 31: Visualization of the comparison of provided material and deposited material for suggested deposition parameters

Finally, by feeding the suggested parameters back into the trained model, the predicted geometry can be visualized and compared to the desired geometry. The figure depicts good congruence of the predicted and the desired geometries with a slight surplus in BH. The surplus might be because BH and BW are chosen arbitrarily and not every combination of these geometric features is producible. The backward algorithm shows good performance for many different desired geometries. In some cases, the relation of provided material to deposited cross-section material is not as interpretable as in the provided example. In these cases, it's possible that the model overfitted to the faulty experimental data for some cross-sections that are similar to the desired geometry. To make the inverse model more robust, more data is needed for training the forward model. Additionally, the inverse algorithm needs to be validated through real-world experiments in a future study.

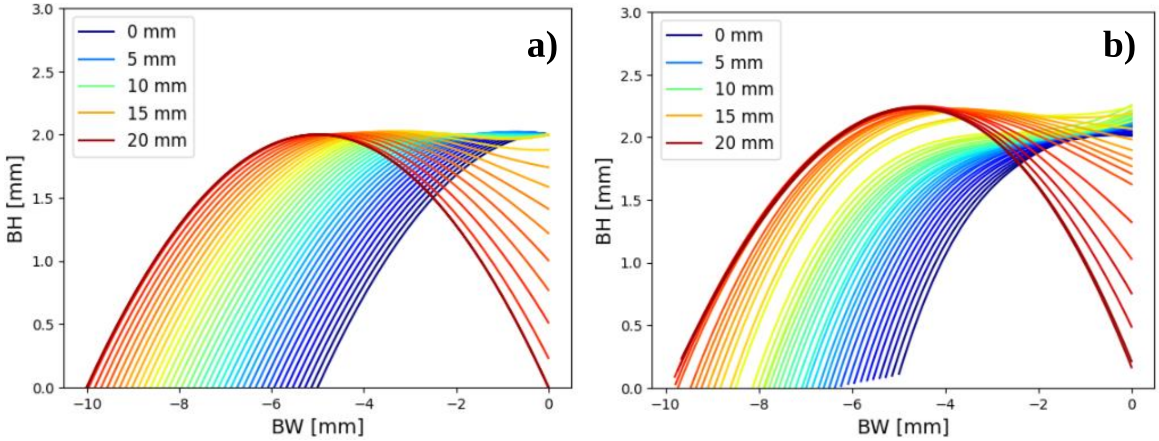


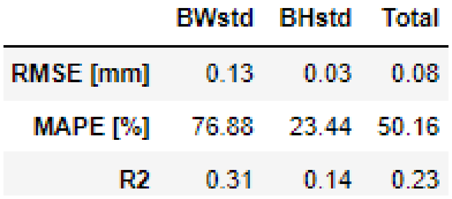
Figure 32: Comparison of the (a) desired geometry and (b) predicted geometry based on the suggested process parameters from the inverse algorithm

5.3 Variance prediction study (study 3)

The objective of the last study was to assess the possibility of predicting the variance of a bead based on the welding parameters. This can provide a tolerance band for the prediction of bead geometry features. This tolerance band could especially be helpful for machining (e.g., milling) operations after the deposition of weld beads in a hybrid manufacturing scenario. The knowledge of the mean BH or the mean BW together with the tolerance band could be used to choose the initial milling trajectory while minimizing the risk of an excessive axial depth of cut.

To study the predictability of variance, the experimental data for straight beads is used. The data for corners had to be excluded from this study as not enough data was collected to calculate a variance for each combination of DtC and corner angle. For each layer of the straight beads, the variance in the maximum BH and in the BW was calculated. The input of the model is the PM, the TS, and the layer number. An ANN was trained using the PyTorch library and its hyperparameter was optimized by cross-validation with Optuna. The performance of the trained model is shown in table 15. With a very low mean R2 score of 0.23, only some of the variance can be explained. An RMSE of 0.03 mm on the variance prediction of BH has the potential to reduce some uncertainty concerning bead geometry variance and thus help determine the initial milling trajectory for hybrid manufacturing. But the overall model performance is still seen to be rather poor. More data could help in improving predictability. After all, it is fair to assume that a variation in deposited material caused by a change in the bead deposition parameters also changes the magnitude of variance in the bead geometry.

Table 15: Performance of variance prediction ANN



	BWstd	BHstd	Total
RMSE [mm]	0.13	0.03	0.08
MAPE [%]	76.88	23.44	50.16
R2	0.31	0.14	0.23

CHAPTER 6. CONCLUSION

Deposition accuracy is one factor that limits the prevalence of WAAM in modern manufacturing. This work conducted three studies that help in mitigating this hurdle and that point future research in the most promising directions.

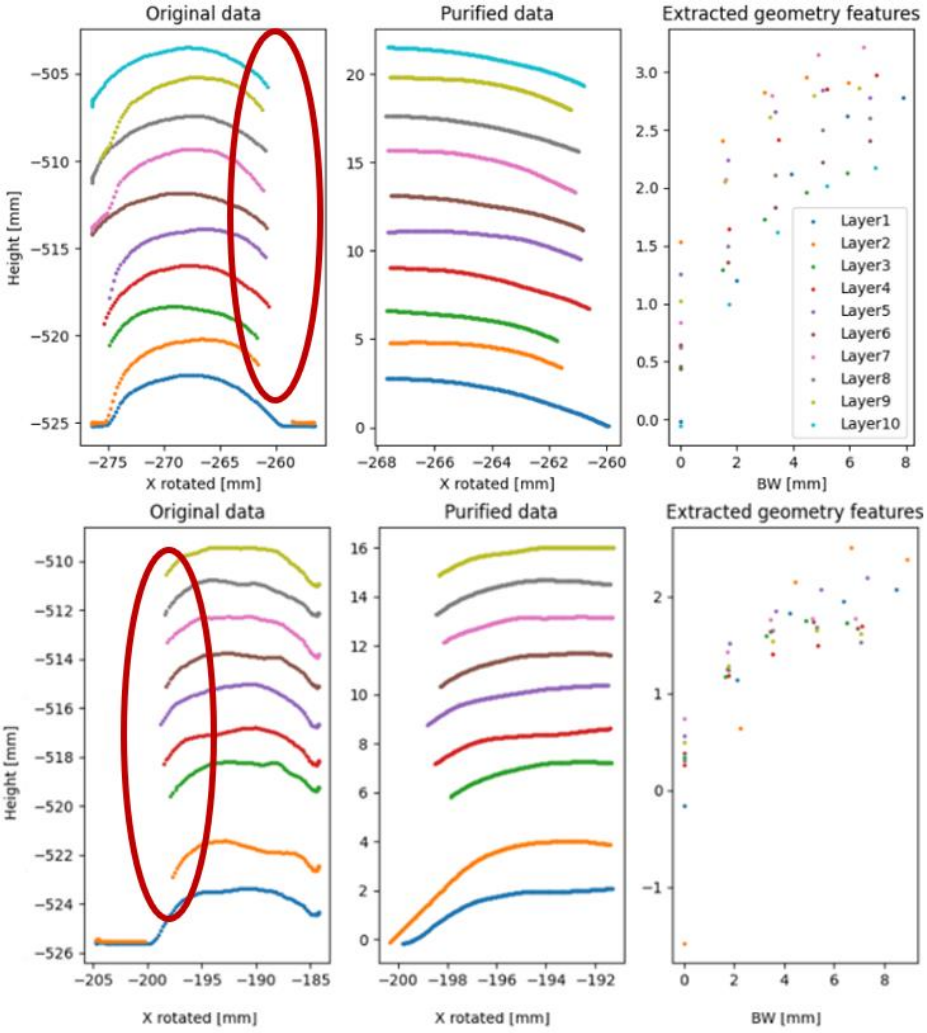
The first study assesses the transferability of trained ML models from one testbed onto other testbeds. A holistic model covering the complex multi-physics of the WAAM process could help WAAM applicants to profit from conducted experiments on multiple other WAAM setups. In the long run, this will be an inevitable step for sophisticated deposition control as a plethora of testbed settings and testbed characteristics influence the bead shape. To study the influence of all these testbed characteristics in one set-up would entail an unmanageable number of necessary experiments. In this work, the transferability study was initialized by first ensuring the quality of available data. The quality of almost a third of the collected studies was insufficient, as the data showed a large discrepancy between deposited material (according to the bead cross-section) versus provided material (according to the welding settings). By training an ANN and a linear regressor on the remaining datasets individually, it became obvious, that the limited amount of available data per testbed prevented the ANN from developing its full potential. Nevertheless, it was possible to achieve a similar performance as in each particular paper. Thereafter, several models were trained on the data of multiple testbeds. First with only the fully transferable welding parameters WFS and TS, which were present and varied in all collected datasets. The performance of these models didn’t deteriorate much, which showed the dominant influence of TS and WFS on geometry prediction. But the transferability of each model, trained on one testbed and tested on another, proved to be poor. To study the influence of other testbed characteristics on the transferability, the models were subsequentially trained with a different combination of additional input features, and the performance was assessed. Only the inclusion of voltage as an input feature and the inclusion of the utilized wire diameter improved the transferability. This is most likely due to the fact, that not enough different testbeds were available yet, for the model to properly learn the influence of superordinate features, like utilized material or the gas flow rate. This underlines the importance for research groups to make their data publicly accessible and to provide the full range of the used experiment settings. In this step of the study, the ANN performed better than the linear regressor. In the final step, the training of the ANN was altered by including the principle of mass conservation in the loss function. While the relation of BH and BW might change with the testbed settings of different research environments, the mass conservation principle of deposited material and provided material had to hold throughout all testbeds. By including this principle, the hope was to teach the algorithm this universal principle and improve its transferability performance. A thorough hyperparameter study, involving the training of hundreds of models including and excluding mass conservation in the loss function, showed that the transferability wasn’t affected by the mass conservation principle. The fact that the performance didn’t improve or worsen might be seen as a sign, that the principle is simple enough to be incorporated by both types of models. Finally, the best model from the hyperparameter study was used to predict on the unseen data of the research testbed used for the second study of this work. A mean error of around 10% in predicting BH and BW was achieved. This accuracy might be good enough for many applications and could avoid the need for other research groups or companies venturing into WAAM to conduct a similar study. The weights and parameters of the trained model are provided in this work. To further reduce the error below the 10% more data including more process characteristics is needed, which underlines the plea for research groups to publish their data.

The second study of this work targeted the extension of geometry prediction onto shape features that go beyond the single-layer straight bead geometry that is usually studied in the literature. For this work, the shape features of corners and multi-layer parts were chosen, whereas a similar need for research can be seen for other shape features like overhangs or hollows. To study the influence of corners and multi-layer deposition, experiments were conducted which included 10-layered parts with varying corner angles. A shape geometry paradigm was developed to capture the geometry development along a corner and over multiple layers. The generated data was used to train an ANN with a tolerable performance of 0.7 in the R2 metric and an RMSE of 0.27mm on BH prediction. Additionally, the model performance was assessed by conducting a feature influence study, by creating a fictional dataset where all input features are kept constant except the feature of interest. The model learned a reasonable relationship between TS and WFS, as well as the influence of distance-to-corner and the corner angle. The only obvious discrepancy existed for the influence of layer number. Here the model overfitted to a measurement error of the laser sensor for the edges of the upper bead layers, which lead to a consistent underestimation of the BW. Finally, the trained ANN was used to infer welding parameters which would lead to an optimized bead geometry in the corner region. The inverse algorithm suggests welding parameters for every point along the deposition direction in a way that a smooth and uniform part surface is created. To tackle the problem of porosity in multi-bead corner structures, H. H. Liu et al. (2020) adapt the deposition tool path of inner beads to produce a more acute outer contour. While this approach eliminated the problem of porosity it increases the problem of material accumulation. The inverse model is set up in a way, that the suggested approach of H. H. Liu et al. (2020) can be applied while maintaining a smooth corner surface. Nevertheless, the inverse model does not yet provide reasonable results for every kind of desired geometry. Wherever the trained ANN overfitted to flaws in the dataset, the inverse model tends to predict unreasonable parameter combinations. The model needs to be made more robust, by training the ANN on more data. Additionally, future models should be trained with the deposition direction as an input feature, as its influence on the bead geometry in the corner region became vivid when plotting the data. Finally, it is recommended to adapt the capturing of the bead geometry. Instead of using BH measurements relative to the BW, a description of the shape with independent geometry characteristics like contact angle or maximum BH should prove to be more robust against measurement errors. With the depicted approach of experiment design, geometry characterization paradigm, model training, feature study, and inverse model, research study two provides a blueprint for other studies on the geometry prediction of different shape features.

The final study conducted in this work targeted the prediction of geometry variance based on welding settings. This study results from the identified shortcoming in current literature to only predict on mean geometry features of straight beads. With this, an approximation of the expected bead geometry can be obtained from the model, but no knowledge about the tolerance can be obtained. While the trained model was able to show some correlation between the geometry variance and welding parameters, it performed poorly with an R2 score of 0.23. More good quality data, without any sensor measurement error as encountered in this study, should improve the model accuracy. A good model would also hold the potential to derive a inverse model similar to study two which provides optimized welding parameter combination leading to minimal geometry variance on deposited beads.

APPENDIX

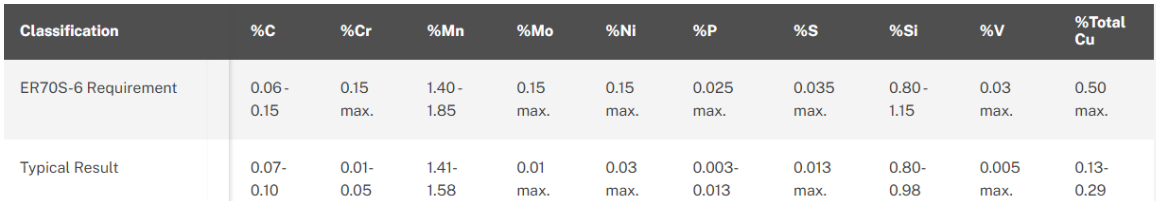
A.1 Visualization of measurement failure at bead edges



A.2 Data sheet of utilized feed stock and substrate material

Feedstock wire: Copper-coated mild steel (AWS ER70S-6)

Composition:

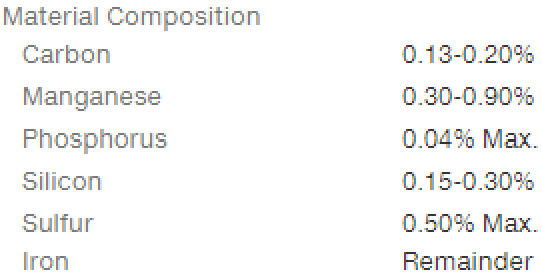


Classification	%C	%Cr	%Mn	%Mo	%Ni	%P	%S	%Si	%V	%Total Cu
ER70S-6 Requirement	0.06 - 0.15	0.15 max.	1.40 - 1.85	0.15 max.	0.15 max.	0.025 max.	0.035 max.	0.80 - 1.15	0.03 max.	0.50 max.
Typical Result	0.07- 0.10	0.01- 0.05	1.41- 1.58	0.01 max.	0.03 max.	0.003- 0.013	0.013 max.	0.80- 0.98	0.005 max.	0.13- 0.29

(The Lincoln Electric Company, 2022)

Substrate: Low-carbon steel (steel grade: 1006-1026)

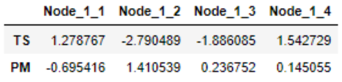
Composition: (*McMaster-Carr*, 2022)



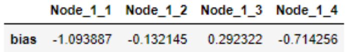
Material Composition	
Carbon	0.13-0.20%
Manganese	0.30-0.90%
Phosphorus	0.04% Max.
Silicon	0.15-0.30%
Sulfur	0.50% Max.
Iron	Remainder

A.3 Weights of ANN for geometry prediction on external research testbeds

Hidden Layer 1:

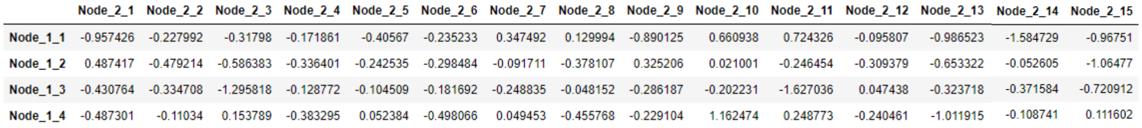


	Node_1_1	Node_1_2	Node_1_3	Node_1_4
TS	1.278767	-2.790489	-1.886085	1.542729
PM	-0.695416	1.410539	0.236752	0.145055

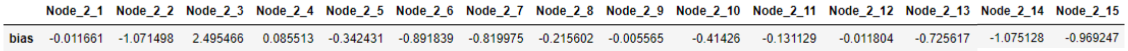


	Node_1_1	Node_1_2	Node_1_3	Node_1_4
bias	-1.093887	-0.132145	0.292322	-0.714256

Hidden Layer 2:

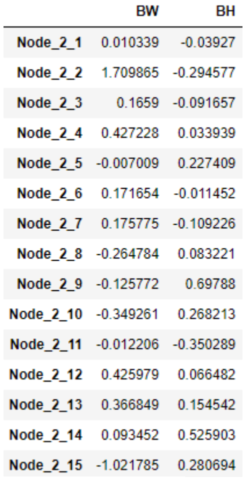


	Node_2_1	Node_2_2	Node_2_3	Node_2_4	Node_2_5	Node_2_6	Node_2_7	Node_2_8	Node_2_9	Node_2_10	Node_2_11	Node_2_12	Node_2_13	Node_2_14	Node_2_15
Node_1_1	-0.957426	-0.227992	-0.31798	-0.171861	-0.40567	-0.235233	0.347492	0.129994	-0.890125	0.660938	0.724326	-0.095807	-0.986523	-1.584729	-0.96751
Node_1_2	0.487417	-0.479214	-0.586383	-0.336401	-0.242535	-0.298484	-0.091711	-0.378107	0.325206	0.021001	-0.246454	-0.309379	-0.653322	-0.052605	-1.06477
Node_1_3	-0.430764	-0.334708	-1.295818	-0.128772	-0.104509	-0.181692	-0.248835	-0.048152	-0.286187	-0.202231	-1.627036	0.047438	-0.323718	-0.371584	-0.720912
Node_1_4	-0.487301	-0.11034	0.153789	-0.383295	0.052384	-0.498066	0.049453	-0.455768	-0.229104	1.162474	0.248773	-0.240461	-1.011915	-0.108741	0.111602

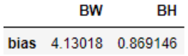


	Node_2_1	Node_2_2	Node_2_3	Node_2_4	Node_2_5	Node_2_6	Node_2_7	Node_2_8	Node_2_9	Node_2_10	Node_2_11	Node_2_12	Node_2_13	Node_2_14	Node_2_15
bias	-0.011661	-1.071498	2.495466	0.085513	-0.342431	-0.891839	-0.819975	-0.215602	-0.005565	-0.41426	-0.131129	-0.011804	-0.725617	-1.075128	-0.969247

Output Layer:



	BW	BH
Node_2_1	0.010339	-0.03927
Node_2_2	1.709865	-0.294577
Node_2_3	0.1659	-0.091657
Node_2_4	0.427228	0.033939
Node_2_5	-0.007009	0.227409
Node_2_6	0.171654	-0.011452
Node_2_7	0.175775	-0.109226
Node_2_8	-0.264784	0.083221
Node_2_9	-0.125772	0.69788
Node_2_10	-0.349261	0.268213
Node_2_11	-0.012206	-0.350289
Node_2_12	0.425979	0.066482
Node_2_13	0.366849	0.154542
Node_2_14	0.093452	0.525903
Node_2_15	-1.021785	0.280694



	BW	BH
bias	4.13018	0.869146

CHAPTER 7. REFERENCES

Akiba, T., Sano, S., Yanase, T., Ohta, T., & Koyama, M. (2019). Optuna. In A. Teredesai, V. Kumar, Y. Li, R. Rosales, E. Terzi, & G. Karypis (Eds.), *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (pp. 2623–2631). ACM. https://doi.org/10.1145/3292500.3330701

Bai, X., Colegrove, P., Ding, J., Zhou, X., Diao, C., Bridgeman, P., roman Hönnige, J., Zhang, H., & Williams, S. (2018). Numerical analysis of heat transfer and fluid flow in multilayer deposition of PAW-based wire and arc additive manufacturing. *International Journal of Heat and Mass Transfer*, *124*, 504–516.

Beckhoff Automation GmbH. (2022, November 10). *DeviceNet Bus Coupler*. https://www.beckhoff.com/en-us/products/i-o/bus-terminals/bkxxxx-bus-coupler/bk5200.html

Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, *13*, 281–305. http://www.jmlr.org/papers/volume13/bergstra12a/bergstra12a.pdf

Buschbacher, F. (2016). Wertschöpfung mit Big Data Analytics. *Controlling & Management Review*, *60*(S1), 40–45. https://doi.org/10.1007/s12176-016-0009-8

Chen, Fu, Y., Kong, F., Li, R., Xiao, Y., Hu, J., & Zhang, H. (2022). An in-process multi-feature data fusion nondestructive testing approach for wire arc additive manufacturing. *Rapid Prototyping Journal*, *28*(3), 573–584. https://doi.org/10.1108/RPJ-02-2021-0034

Cunningham, C. R., Flynn, J. M., Shokrani, A., Dhokia, V., & Newman, S. T. (2018). Invited review article: Strategies and processes for high quality wire arc additive manufacturing. *Additive Manufacturing*, *22*, 672–686. https://doi.org/10.1016/j.addma.2018.06.020

Daw, A., Karpatne, A., Watkins, W., Read, J., & Kumar, V. (2017, October 31). *Physics-guided Neural Networks (PGNN): An Application in Lake Temperature Modeling*. https://arxiv.org/pdf/1710.11431.pdf

Deng, J., Xu, Y., Zuo, Z., Hou, Z., & Chen, S [Shanben]. (2019). Bead Geometry Prediction for Multi-layer and Multi-bead Wire and Arc Additive Manufacturing Based on XGBoost. In S. Chen, Y. Zhang, & Z. Feng (Eds.), *Transactions on Intelligent Welding Manufacturing. Transactions on Intelligent Welding Manufacturing* (pp. 125–135). Springer Singapore. https://doi.org/10.1007/978-981-13-8668-8_7

Ding, D., He, F., Yuan, L., Pan, Z., Wang, L [Lei], & Ros, M. (2021). The first step towards intelligent wire arc additive manufacturing: An automatic bead modelling system using machine learning through industrial information integration. *Journal of Industrial Information Integration*, *23*, 1–16. https://doi.org/10.1016/j.jii.2021.100218

Ding, D., Pan, Z., Cuiuri, D., & Li, H. (2015). A multi-bead overlapping model for robotic wire and arc additive manufacturing (WAAM). *Robotics and Computer-Integrated Manufacturing*, *31*, 101–110. https://doi.org/10.1016/j.rcim.2014.08.008

Ding, D., Pan, Z., Cuiuri, D., Li, H., van Duin, S., & Larkin, N. (2016). Bead modelling and implementation of adaptive MAT path in wire and arc additive manufacturing. *Robotics and Computer-Integrated Manufacturing*, *39*, 32–42. https://doi.org/10.1016/j.rcim.2015.12.004

Ding, D., Zhao, R., Lu, Q., Pan, Z., Li, H., Wang, K., & He, K. (2021). A shape control strategy for wire arc additive manufacturing of thin-walled aluminium structures with sharp corners. *Journal of Manufacturing Processes*, *64,* 253–264. https://doi.org/10.1016/j.jmapro.2021.01.029

Dinovitzer, M., Chen, X., Laliberte, J., Huang, X., & Frei, H. (2019). Effect of wire and arc additive manufacturing (WAAM) process parameters on bead geometry and microstructure. *Additive Manufacturing*, *26*, 138–146. https://doi.org/10.1016/j.addma.2018.12.013

Domingos, P. (2012). A few useful things to know about machine learning. *Communications of the ACM*, *55*(10). https://doi.org/10.1145/2347736.2347755

Donnelly, P., Boden, M., Brownsword, R., Ghahramani, Z., Griffiths, N., Hassabis, D., & Hauert, S. (2017). *Machine learning: The power and promise of computers that learn by example*. The Royal Society.

Du, Y., Mukherjee, T., & DebRoy, T. (2021). Physics-informed machine learning and mechanistic modeling of additive manufacturing to reduce defects. *Applied Materials Today*, *24*, 101123. https://doi.org/10.1016/j.apmt.2021.101123

Flath, C. M., & Stein, N. (2018). Towards a data science toolbox for industrial analytics applications. *Computers in Industry*, *94*, 16–25. https://doi.org/10.1016/j.compind.2017.09.003

Gaikwad, A., Yavari, R., Montazeri, M., Cole, K., Bian, L., & Rao, P. (2020). Toward the digital twin of additive manufacturing: Integrating thermal simulations, sensing, and analytics to detect process faults. *IISE Transactions, 52*(11), 1204–1217. https://doi.org/10.1080/24725854.2019.1701753

Gebhardt, A. (2016). *Additive Fertigungsverfahren: Additive Manufacturing und 3D-Drucken für Prototyping - Tooling - Produktion* (5., neu bearbeitete und erweiterte Auflage). Hanser. http://www.hanser-fachbuch.de/9783446444010

Géron, A. (2017). *Hands-on machine learning with Scikit-Learn and TensorFlow: Concepts, tools, and techniques to build intelligent systems* (First edition). O'Reilly Media. http://proquest.tech.safaribooksonline.de/9781491962282

Géron, A. (2019). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems* (Second edition). *Covid-19 collection*. O'Reilly.

Gokhale, N. P., Kala, P., & Sharma, V. (2019). Thin-walled metal deposition with GTAW welding-based additive manufacturing process. *Journal of the Brazilian*

Society of Mechanical Sciences and Engineering, *41*(12). https://doi.org/10.1007/s40430-019-2078-z

Golovin, D., Solnik, B., Moitra, S., Kochanski, G., Karro, J., & Sculley, D. (2017). Google Vizier. In S. Matwin, S. Yu, & F. Farooq (Eds.), *Kdd2017: August 13-17, 2017, Halifax, NS, Canada* (pp. 1487–1495). Association for Computing Machinery Inc. (ACM). https://doi.org/10.1145/3097983.3098043

Grezmak, J., Zhang, J., Wang, P [Peng], Loparo, K. A., & Gao, R. X. (2020). Interpretable Convolutional Neural Network Through Layer-wise Relevance Propagation for Machine Fault Diagnosis. *IEEE Sensors Journal*, *20*(6), 3172–3181. https://doi.org/10.1109/JSEN.2019.2958787

Guo, S., Agarwal, M., Cooper, C., Tian, Q., Gao, R. X., Guo, W., & Guo, Y. B. (2022). Machine learning for metal additive manufacturing: Towards a physics-informed data-driven paradigm. *Journal of Manufacturing Systems*, *62*, 145–163. https://doi.org/10.1016/j.jmsy.2021.11.003

Guo, W., Tian, Q., Guo, S., & Guo, Y. (2020). A physics-driven deep learning model for process-porosity causal relationship and porosity prediction with interpretability in laser metal deposition. *CIRP Annals*(69), 205–208. https://doi.org/10.1016/j.cirp.2020.04.049

Herriott, C., & Spear, A. D. (2020). Predicting microstructure-dependent mechanical properties in additively manufactured metals with machine- and deep-learning methods. *Computational Materials Science*, *175*, 109599. https://doi.org/10.1016/j.commatsci.2020.109599

Howland, M. F., & Dabiri, J. O. (2019). Wind Farm Modeling with Interpretable Physics-Informed Machine Learning. *Energies*, *12*(14), 2716. https://doi.org/10.3390/en12142716

Hu, Z., Qin, X., Shao, T., & Liu, H. (2018). Understanding and overcoming of abnormity at start and end of the weld bead in additive manufacturing with GMAW. *The International Journal of Advanced Manufacturing Technology*, *95*(5-8), 2357–2368. https://doi.org/10.1007/s00170-017-1392-9

Israr, R., Buhl, J., & Bambach, M. (2021). A study on power-controlled wire-arc additive manufacturing using a data-driven surrogate model. *The International Journal of Advanced Manufacturing Technology*, *117*(7-8), 2133–2147. https://doi.org/10.1007/s00170-021-07358-y

Keyence. (2022, September 18). *LK-G35*. https://www.keyence.com/products/measure/laser-1d/lk-g3000/models/lk-g35/

Kouraytem, N., Li, X., Tan, W., Kappes, B., & Spear, A. D. (2021). Modeling process–structure–property relationships in metal additive manufacturing: a review on physics-driven versus data-driven approaches. *Journal of Physics: Materials*, *4*(3), 32002. https://doi.org/10.1088/2515-7639/abca7b

Kulkarni, A., Bhatt, P. M., Kanyuck, A., & Gupta, S. K. (2021). Using Unsupervised Learning for Regulating Deposition Speed During Robotic Wire Arc Additive

Manufacturing. In *Volume 2: 41st Computers and Information in Engineering Conference (CIE)*. American Society of Mechanical Engineers. https://doi.org/10.1115/DETC2021-71865

Lambiase, F., Scipioni, S. I., & Paoletti, A. (2022). Accurate prediction of the bead geometry in wire arc additive manufacturing process. *The International Journal of Advanced Manufacturing Technology*, *119*(11-12), 7629–7639. https://doi.org/10.1007/s00170-021-08588-w

Li, F., Chen, S [Shujun], Wu, Z., & Yan, Z. (2018). Adaptive process control of wire and arc additive manufacturing for fabricating complex-shaped components. *The International Journal of Advanced Manufacturing Technology*, *96*(1-4), 871–879. https://doi.org/10.1007/s00170-018-1590-0

Li, Y [Yongzhe], Sun, Y., Han, Q., Zhang, G., & Horváth, I. (2018). Enhanced beads overlapping model for wire and arc additive manufacturing of multi-layer multi-bead metallic parts. *Journal of Materials Processing Technology*, *252*, 838–848. https://doi.org/10.1016/j.jmatprotec.2017.10.017

Lin, Z., Song, K., & Yu, X. (2021). A review on wire and arc additive manufacturing of titanium alloy. *Journal of Manufacturing Processes*, *70*, 24–45. https://doi.org/10.1016/j.jmapro.2021.08.018

The Lincoln Electric Company. (2022, November 21). *SuperArc® L-56®*. https://www.lincolnelectric.com/en/products/superarcl56_gmaw

Liu, H. H., Zhao, T., Li, L. Y., Liu, W. J., Wang, T. Q., & Yue, J. F. (2020). A path planning and sharp corner correction strategy for wire and arc additive manufacturing of solid components with polygonal cross-sections. *The International Journal of Advanced Manufacturing Technology, 106*(11-12), 4879–4889. https://doi.org/10.1007/s00170-020-04960-4

Liu, R., Liu, S., & Zhang, X. (2021). A physics-informed machine learning model for porosity analysis in laser powder bed fusion additive manufacturing. *The International Journal of Advanced Manufacturing Technology, 113*(7-8), 1943–1958. https://doi.org/10.1007/s00170-021-06640-3

Matignon, R. (2005). *Neural Network Modeling using SAS Enterprise Miner*. Author House.

Mbodj, N. G., Abuabiah, M., Plapper, P., El Kandaoui, M., & Yaacoubi, S. (2021). Bead Geometry Prediction in Laser-Wire Additive Manufacturing Process Using Machine Learning: Case of Study. *Applied Sciences, 11*(24), 11949. https://doi.org/10.3390/app112411949

McMaster-Carr. (2022, November 21). https://www.mcmaster.com/6544K76/

Mu, H., Pan, Z., Li, Y [Yuxing], He, F., Polden, J., & Xia, C. (2021). MIMO Model Predictive Control of Bead Geometry in Wire Arc Additive Manufacturing. In *2021 IEEE 11th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER)* (pp. 169–174). IEEE. https://doi.org/10.1109/CYBER53097.2021.9588331

Mueller, J. P., & Massaron, L. (2016). *Machine learning for dummies*. *For dummies*. John Wiley & Sons Inc. http://gbv.eblib.com/patron/FullRecord.aspx?p=4526803

Müller, A. C., & Guido, S. (2017). *Einführung in Machine Learning mit Python: Praxiswissen Data Science* ((K. Rother, Trans.)) (1. Auflage). O'Reilly. http://proquestcombo.safaribooksonline.com/9781492064619

Nagarajan, H. P. N., Mokhtarian, H., Jafarian, H., Dimassi, S., Bakrani-Balani, S., Hamedi, A., Coatanéa, E., Gary Wang, G., & Haapala, K. R. (2019). Knowledge-Based Design of Artificial Neural Network Topology for Additive Manufacturing Process Modeling: A New Approach and Case Study for Fused Deposition Modeling. *Journal of Mechanical Design*, *141*(2), Article 021705. https://doi.org/10.1115/1.4042084

Nagesh, D. S., & Datta, G. L. (2002). Prediction of weld bead geometry and penetration in shielded metal-arc welding using artificial neural networks. *Journal of Materials Processing Technology*, *123*(2), 303–312. https://doi.org/10.1016/S0924-0136(02)00101-2

Ness, K. L., Paul, A., Sun, L., & Zhang, Z. (2022). Towards a generic physics-based machine learning model for geometry invariant thermal history prediction in additive manufacturing. *Journal of Materials Processing Technology*, *302*, 117472.

Norrish, J., Polden, J., & Richardson, I. (2021). A review of wire arc additive manufacturing: development, principles, process physics, implementation and current status. *Journal of Physics D: Applied Physics*, *54*(47), 1–29. https://doi.org/10.1088/1361-6463/ac1e4a

Nouri, M., Abdollah-zadeh, A., & Malek, F. (2007). Effect of Welding Parameters on Dilution and Weld Bead Geometry in Cladding. *Journal of Material Science and Technology*(23), Article 6, 817–822. https://www.researchgate.net/publication/266895217_Effect_of_Welding_Parameters_on_Dilution_and_Weld_Bead_Geometry_in_Cladding

Panda, B., Shankhwar, K., Garg, A., & Savalani, M. M. (2019). Evaluation of genetic programming-based models for simulating bead dimensions in wire and arc additive manufacturing. *Journal of Intelligent Manufacturing*, *30*(2), 809–820. https://doi.org/10.1007/s10845-016-1282-2

Paper, D. (2020). *Hands-on Scikit-Learn for Machine Learning Applications*. Apress. https://doi.org/10.1007/978-1-4842-5373-1

Petrik, J., Sydow, B., & Bambach, M. (2022). Beyond parabolic weld bead models: AI-based 3D reconstruction of weld beads under transient conditions in wire-arc additive manufacturing. *Journal of Materials Processing Technology*, *302*, 117457. https://doi.org/10.1016/j.jmatprotec.2021.117457

Phillips, D. H. (2016). *Welding engineering: An introduction*. John Wiley & Sons Ltd. https://onlinelibrary.wiley.com/doi/book/10.1002/9781119191407 https://doi.org/10.1002/9781119191407

Provost, F., & Fawcett, T. (2017). *Data Science für Unternehmen: Data Mining und datenanalytisches Denken praktisch anwenden* (1st ed.). *mitp Business*. MITP. https://ebookcentral.proquest.com/lib/gbv/detail.action?docID=5123640

PyTorch.org. (2022, October 9). https://pytorch.org/

Rai, R., & Sahu, C. K. (2020). Driven by Data or Derived Through Physics? A Review of Hybrid Physics Guided Machine Learning Techniques With Cyber-Physical System (CPS) Focus. *IEEE Access*, *8*, 71050–71073. https://doi.org/10.1109/ACCESS.2020.2987324

Ren, K., Chew, Y., Zhang, Y. F., Fuh, J., & Bi, G. J. (2020). Thermal field prediction for laser scanning paths in laser aided additive manufacturing by physics-based machine learning. *Computer Methods in Applied Mechanics and Engineering*, *362*, 112734. https://doi.org/10.1016/j.cma.2019.112734

RoboDK. (2022, September 18). *Simulator for industrial robots and offline programming*. https://robodk.com/

Ruggeri, F., Kenett, R., & Faltin, F. W. (2007). *Encyclopedia of statistics in quality and reliability*. John Wiley.

Ryan, E. M. (2020). *On wire and arc additive manufacture of aluminium.* https://doi.org/10.15126/THESIS.00850379

scikit-learn.org. (2022, October 10). https://scikit-learn.org/stable/about.html

Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, T., Baker, L., Lai, M., Bolton, A., Chen, Y., Lillicrap, T., Hui, F., Sifre, L., van den Driessche, G., Graepel, T., & Hassabis, D. (2017). Mastering the game of Go without human knowledge. *Nature*, *550*(7676), 354–359. https://doi.org/10.1038/nature24270

Steiner, H., & Welker, P. (2016). Wird der Controller zum Data Scientist? *Controlling & Management Review*, *60*(S1), 68–73. https://doi.org/10.1007/s12176-016-0003-1

Stevens, E., Antiga, L., & Viehmann, T. (2020). *Deep Learning with PyTorch*. Manning.

Venkata Rao, K., Parimi, S., Suvarna Raju, L., & Suresh, G. (2022). Modelling and optimization of weld bead geometry in robotic gas metal arc-based additive manufacturing using machine learning, finite-element modelling and graph theory and matrix approach. *Soft Computing*, *26*(7), 3385–3399. https://doi.org/10.1007/s00500-022-06749-x

Vora, J., Parmar, H., Chaudhari, R., Khanna, S., Doshi, M., & Patel, V. (2022). Experimental investigations on mechanical properties of multi-layered structure fabricated by GMAW-based WAAM of SS316L. *Journal of Materials Research and Technology*, *20*, 2748–2757. https://doi.org/10.1016/j.jmrt.2022.08.074

Wacker, C., Köhler, M., David, M., Aschersleben, F., Gabriel, F., Hensel, J., Dilger, K., & Dröder, K. (2021). Geometry and Distortion Prediction of Multiple Layers for Wire Arc Additive Manufacturing with Artificial Neural Networks. *Applied Sciences*, *11*(10), 4694. https://doi.org/10.3390/app11104694

Welsch, A., Eitle, V., & Buxmann, P. (2018). Maschinelles Lernen: Grundlagen und betriebswirtschaftliche Anwendungspotenziale am Beispiel von Kundenbindungsprozessen. *HMD Praxis Der Wirtschaftsinformatik*, *55*(2), 366–382. https://doi.org/10.1365/s40702-018-0404-z

Wu, D., Wang, P [Pengfei], Wu, P., Yang, Q., Liu, F., Han, Y., Xu, F., & Wang, L [Lin] (2015). Determination of contact angle of droplet on convex and concave spherical surfaces. *Chemical Physics*, *457*, 63–69. https://doi.org/10.1016/j.chemphys.2015.05.020

Wuest, T., Weimer, D., Irgens, C., & Thoben, K.-D. (2016). Machine learning in manufacturing: advantages, challenges, and applications. *Production & Manufacturing Research*, *4*(1), 23–45. https://doi.org/10.1080/21693277.2016.1192517

Xie, X., Bennett, J., Saha, S., Lu, Y., Cao, J., Liu, W. K., & Gan, Z. (2021). Mechanistic data-driven prediction of as-built mechanical properties in metal additive manufacturing. *Npj Computational Materials*, *7*(1). https://doi.org/10.1038/s41524-021-00555-z

Xiong, W., To, A., & Klecka, M. (2022). *Integrated Computational Materials and Mechanical Modeling for Additive Manufacturing of Alloys with Graded Structure Used in Fossil Fuel Power Plants.* https://doi.org/10.2172/1842580

Zhao, Y., Li, W., & Liu, A. (2020). Optimization of geometry quality model for wire and arc additive manufacture based on adaptive multi-objective grey wolf algorithm. *Soft Computing*, *24*(22), 1–16. https://doi.org/10.1007/s00500-020-05027-y

Zhou, Z., Shen, H., Liu, B., Du, W., Jin, J., & Lin, J. (2022). Residual thermal stress prediction for continuous tool-paths in wire-arc additive manufacturing: a three-level data-driven method. *Virtual and Physical Prototyping*, *17*(1), 105–124. https://doi.org/10.1080/17452759.2021.1997259

Zhu, Q., Liu, Z., & Yan, J. (2021). Machine learning for metal additive manufacturing: predicting temperature and melt pool fluid dynamics using physics-informed neural networks. *Computational Mechanics*, *67*(2), 619–635. https://doi.org/10.1007/s00466-020-01952-9