Vision-Based Process Optimization for Hybrid Metal 3D Printing

Hybrid metal 3D printing combines Gas Metal Arc Welding (GMAW)-based deposition with CNC machining on a single machine, producing finished metal parts without separate post-processing equipment. This research focuses on using Wire Arc Additive Manufacturing (WAAM) for building metal parts layer by layer, offering high deposition rates and material efficiency. However, variations in weld-bead geometry due to process-parameter interactions result in uneven layer surfaces and increased machining requirements. The variation in height directly determines how much material machining must remove. Smaller height variation means tighter tolerances, shorter machining time, and less tool wear. Minimizing height variation requires careful process parameter selection, but existing measurement methods are either too complex or not suited for in-process use. This paper presents an algorithm that measures bead height variation from side-profile camera images and recommends parameter optimization to minimizes height deviations. A U-Net segmentation model pretrained on ImageNet extracts the bead surface profile from each photograph. Training on a small set of manually labeled images was satisfactory to achieve strong segmentation performance in bead height detection. Bead height values extracted from the images were validated against Coordinate Measuring Machine (CMM) measurements and showed close agreement on peak height (H_{max}) across the tested beads. These validated measurements were then used to train a Gaussian Process Regression (GPR) model, which identifies a candidate parameter combination predicted to minimize height variation and reports a confidence interval for each prediction. Unlike deterministic models, GPR communicates the reliability of each prediction, helping operators decide whether a recommended setting is ready for production. The algorithm runs automatically with only a fixed camera, requiring no additional specialized metrology equipment.

Keywords: *hybrid metal 3D printing, bead height variation, Gaussian Process Regression, U-Net segmentation, process parameter optimization*

Introduction

The WAAM method deposits metal wire layer by layer using an electric arc. This method allows building near-net-shape parts at deposition rates and material costs that powder-bed additive manufacturing cannot match for medium-to-large components in titanium, steel, and aluminum (Williams et al. 2016). That cost advantage has driven growing adoption across aerospace, defense, automotive, and heavy industry.

The hybrid 3D metal printing system at Western Michigan University advances WAAM by integrating CNC machining on the same platform. After each bead layer is deposited by the GMAW welding head, a cutting tool mills the surface before the next layer is deposited (Ikonomov 2020). This alternating

cycle produces dimensionally accurate, fully finished parts without relocating the workpiece.

Within this hybrid cycle, the properties of the welding bead affect the final part quality. When bead height varies, the CNC machining step must remove more material to reach the required surface finish. That extra step increases machining time and accelerates tool wear. The key metric is the difference between the maximum and minimum bead height within a single deposit, $\Delta H = H_{max} - H_{min}$. Minimizing ΔH reduces machining requirements, which is one of the central economic drivers of the hybrid process.

The deposited bead height can be measured using metrology tools such as Coordinate Measuring Machines (CMMs), structured-light scanners, and laser profilometers. These tools are accurate but slow and expensive, and interrupt the deposition workflow as each measurement requires the machine to stop, the part to cool, and in some cases, the workpiece to be repositioned or transferred to a separate measurement station before deposition can resume. This research takes a different approach, to in-process measurement, using a fixed camera to capture side-profile images of each bead immediately after deposition, without stopping the machine or moving the workpiece. Next, a trained neural network extracts the height profile directly from the image. The substrate plate, the flat metal base on which the bead is deposited, is visible in every frame and serves as an in-image calibration reference.

The welding bead height depends on process parameters like machine deposition speed (feed rate), welding wire feed rate, nozzle standoff height from the substrate, and welding power trim (voltage and amperage control). In this study, four parameters were varied across defined ranges of feed rate from 203 to 406 mm/min (8 to 16 in/min), wire feed rate at 5,588 and 6,350 mm/min (220 and 250 in/min), nozzle standoff height at three levels between 9.525 and 12.70 mm (0.375 and 0.500 in), and trim at three levels between 0.9 and 1.1. These ranges yield 162 possible parameter combinations in total. Testing all 162 experimentally is impractical, so 69 beads were deposited across representative combinations, and a model trained on those measurements can predict ΔH for any combination and find the setting that minimizes it.

To identify the optimal parameter combination, Gaussian Process Regression (GPR) was chosen as the model because it returns a calibrated confidence interval alongside every prediction. Other methods like Random Forest or neural network approaches, do not provide this directly for the given sample sizes. In practice with GPR, a prediction paired with a quantified interval is far more useful to an operator than a bare number. It tells the operator directly whether the recommended setting is reliable enough to run in production.

Literature Review

Wire Arc Additive Manufacturing and Hybrid Processing

WAAM has attracted significant research attention since the early 2000s as a lower-cost alternative to powder-bed fusion for medium and large metal components. Gas Metal Arc Welding (GMAW), Gas Tungsten Arc Welding (GTAW), and plasma arc process variants have been demonstrated across titanium alloys, aluminum, and steel (Williams et al., 2016). Plasma arc deposition in particular has shown consistently good results for titanium alloys, producing Ti-6Al-4V components with mechanical properties that meet aerospace structural requirements (Martina et al., 2012).

Wire-feed processes when compared with other additive manufacturing methods like powder-bed fusion and subtractive manufacturing use material more efficiently and are better suited for building medium-to-large components where minimizing waste and machining time matters (Ding et al., 2015; Guo and Leu, 2013). Despite these advantages, controlling the geometry of each deposited layer remains a persistent challenge. Thermal gradients between successive passes induce distortion and residual stress that accumulate throughout the build and ultimately affect the dimensional accuracy of the finished part (Denlinger et al., 2015; Li et al., 2018). The same thermal history also drives microstructure changes within the deposit, which is why monitoring what happens during and immediately after each pass is important rather than waiting until the part is complete (Xiong et al., 2020).

The hybrid 3D metal printing machine used in this study operates on a five-axis CNC platform that alternates between GMAW deposition and milling within each layer (Ikonomov, 2020). Subsequent work from the same laboratory demonstrated that forward-reverse and zigzag deposition path strategies reduce height inconsistencies at layer transitions, achieving build heights above 8 inches on the Z-axis (Ikonomov, 2025). With deposition path strategies established, the remaining lever for controlling bead geometry is the welding parameter combination within each pass, which is the focus of this research. Even with deposition path strategies in place, maintaining part quality and dimensional accuracy across a full build remains sensitive to how well process parameters are selected and controlled at each pass (Jafari et al., 2021).

Bead Geometry Prediction: Classical and Machine Learning (ML) Approaches

Predicting bead geometry from welding parameters has been studied across several modeling approaches. Response Surface Methodology (RSM) is a classical method, fitting a polynomial to data collected at a designed set of parameter combinations and optimizing the fitted surface to identify optimal settings. RSM works well for simple cases but breaks down when parameter interactions are nonlinear or when more than a handful of inputs are involved, both of which apply in the present study (Geng et al., 2017). Neural network models applied to GMAW bead geometry showed early on that data-driven approaches could capture these nonlinear relationships without assuming a fixed functional form, though getting reliable predictions from small experimental datasets was still a challenge (Xiong et al., 2014).

Artificial neural networks/ML handle more complex parameter-to-geometry relationships than RSM without assuming a fixed functional form. This makes them better suited for cases where multiple welding parameters interact in ways a polynomial cannot fit. However, they overfit easily on small datasets and return only a point estimate with no indication of how reliable that prediction is (Wang et al., 2021).

Deep neural networks push accuracy further by incorporating additional input features such as prior-layer geometry measurements, but this requires stopping to measure each layer before the next deposit, which is impractical in most WAAM setups and negates the speed advantage of the process. Combining support vector machines with response surface methods has also been explored as a way to identify optimal process windows more reliably (Shin, 2024; Xiao et al., 2022). When the parameter space is continuous or too large for exhaustive testing, Bayesian optimization offers a more principled way to find good settings by learning from each experiment and focusing subsequent trials on the most promising regions (Snoek et al., 2012). More broadly, the range of machine learning tools available for manufacturing process control has grown considerably, making it increasingly practical to adapt general-purpose frameworks for specific deposition tasks (Müller and Guido, 2016).

Gaussian Process Regression (GPR) provides a distinct aproach to this ML space. It is non-parametric, making no assumption about the functional form of the parameter-to-height deviation (ΔH) relationship, and returns a full predictive distribution rather than a point estimate. This probabilistic output is particularly valuable at low sample counts, which are common in WAAM experiments, and has been demonstrated to perform well in metal AM contexts (Rasmussen and Williams, 2006; Tapia et al., 2016). The present work applies the same GPR method to WAAM bead height variation, with the key distinction that the training targets are derived from an image rather than contact measurement.

Vision-Based Measurement of Welding Bead Geometry

Non-contact measurement of WAAM bead geometry covers a wide range of accuracy, cost, and complexity. Among the more established techniques, structured-light scanning projects a pattern of light onto the bead surface and captures the distortion of that pattern with a camera to reconstruct a full 3D surface map, while laser line scanning sweeps a laser line across the surface and records the reflected profile to build up a point-by-point height map. Both techniques produce detailed geometry data with high precision and have been used in several research systems for in-process monitoring. However, that capability comes at a cost with dedicated hardware, calibration fixtures, and mandatory pauses between layers all adding overhead that slows down the build cycle and makes these methods difficult to deploy in a production WAAM environment.

A two-dimensional (2D) side-profile captured by a camera provides a simpler and far more rapid measurement approach. The bead image is captured with a high-resolution camera positioned perpendicular to the deposition

direction. Because the camera angle and distance are fixed, each pixel in the image corresponds to a known physical dimension. A single calibration step, measuring a reference object of known size visible in the frame, in this case, the substrate plate. The thickness of the substrate plate establishes the pixel-to-millimeter calibration factor that applies to every image in the dataset. The practical difficulty is the feature extraction. Classical approaches like adaptive thresholding work well under stable lighting but breaks down when contrast at the bead edge is low or when a background object bleeds into the detection zone.

Deep learning segmentation handles these edge cases more gracefully. A practical solution to this annotation problem came from transfer learning, where a network already trained on a large general-purpose image dataset is adapted to a new domain with far fewer labeled examples. The pretrained network brings general visual knowledge that transfers well even when the new domain looks very different from the original training data (Pan and Yang, 2010). One such model, U-Net, originally developed by Ronneberger, Fischer, and Brox (2015) for biomedical image segmentation, has become the standard architecture for pixel-level tasks with limited labeled data. Its encoder-decoder structure with skip connections preserves spatial resolution all the way to the output. That matters here because the bead surface boundary is what the measurement actually depends on.

Using a ResNet34 encoder pretrained on ImageNet reduces the annotation burden further. ResNet architectures were built around the idea of residual connections, which allow very deep networks to train stably by letting gradients flow directly through shortcut paths rather than only through the learned layers. That depth is what makes them effective feature extractors across a wide range of visual tasks (He and Sun, 2015). The encoder already captures general edge and texture features, so only the decoder needs domain-specific training. In this work, 22 labeled bead images were enough to reach 93% pixel-level agreement between the predicted bead region and the hand-drawn ground truth. Other segmentation architectures were considered during development. DeepLabV3+ uses a different approach called Atrous Spatial Pyramid Pooling to capture context at multiple scales simultaneously, which works well for natural images but can struggle to preserve spatial resolution at small batch sizes or non-standard image dimensions (Chen et al., 2018).

Methodology

Experimental Setup and Dataset

The welding beads in this study were deposited onto substrate plates using a hybrid 3D metal printing machine. The GMAW welding head is mounted on a five-axis CNC machine and deposits each bead along a straight path at a controlled machine feed rate, with the wire feed rate and nozzle height standoff from the substrate set for each bead. Four parameters were varied across the experiment. Table 1 lists the parameter ranges. A total of 69 beads were

deposited across 12 plates, identified by the naming convention PlateID_BeadID (e.g., P01_BC01 through P12_BC06).

Table 1. *Experimental Parameter Ranges*

Parameter	Min	Max	Values / Levels
Feed Rate mm/min (in/min)	203 (8)	406 (16)	203, 229, 254, 279, 305, 330, 356, 381, 406 (8, 9, 10, 11, 12, 13, 14, 15, 16)
Wire Feed Rate mm/min (in/min)	5,588 (220)	6,350 (250)	5,588, 6,350 (220, 250)
Nozzle Height mm (in)	9.525 (0.375)	12.700 (0.500)	9.525, 11.113, 12.700 (0.375, 0.4375, 0.500)
Trim	0.9	1.1	0.9, 1.0, 1.1

Image Acquisition and Calibration

After each bead cooled to ambient temperature, it was photographed with a high-resolution camera fixed perpendicular to the bead's long axis. The camera position, lens focal length, and manual focus were held constant across the entire dataset. Between captures, the substrate plate was slid laterally to center the target bead in the frame. This ensures the pixel-to-millimeter calibration is identical for every image in the dataset.

The calibration reference is the substrate plate visible at the bottom of every frame. The plate top surface defines Z = 0, which is the base of the deposited bead, and the plate bottom edge provides the second reference point. The plate thickness was measured to be 6.15 mm. For each image, the algorithm detects both plate edges and computes the pixel-to-millimeter scale using the known plate thickness. This gives a physical resolution of 0.015 mm per pixel, which is adequate for measuring ΔH across the dataset.

Figure 1 shows a representative raw side-profile photograph of a WAAM bead on the substrate plate. The bright region at the top of the frame is the background above the bead crown. The dark central body is the deposited bead itself. The plate top edge, visible at the bottom of the bead, defines the Z = 0 reference from which all height measurements are taken.

Figure 1. *Raw side-profile photograph of a steel WAAM bead on substrate plate*

Vision Algorithm

Classical Detection Baseline (Versions 1–5)

Five versions of a classical computer vision algorithm were developed and tested before deep learning was adopted. These results of the five algorithms are

discussed here, as they identify the failure modes that led to the adoption of the CNN approach rather than relying on an off-the-shelf solution.

Version 1 of vision algorithm used GrabCut, a classical interactive segmentation algorithm, to separate the bead from the background using a bounding box approach. At the macro level it correctly identified the bead region, but the detected top edge tracked the silhouette rather than the actual surface contour. This produced heights 0.5–1.5 mm above the true bead crown. Version 2 replaced GrabCut with adaptive thresholding. For each image column, background brightness was sampled from a fixed reference row range, and the detection threshold was set at 70% of that local background. The first pixel darker than the threshold was taken as the bead top. This tracked the actual surface rather than the silhouette and gave the first physically meaningful measurements. A Savitzky-Golay smoothing filter (window = 61, polynomial order = 3) was applied to reduce noise in the extracted profile.

Versions 3 through 5 of vision algorithm added to Version 1 median-filter spike removal (Version 3), background-patch clamping via NaN interpolation (Version 4), and dynamic right-trim detection based on the edge profile gradient (Version 5). The background-patch clamping addressed a recurring artifact where a bright region caused the detector to jump to the image top boundary. By Version 5, the classical algorithm correctly processed 51 of 69 images (74%). The remaining 18 failures shared three root causes. The plate top row varied from row 279 to row 350 across images, making a fixed reference unusable. Some beads had low contrast at the bead-plate junction, where the threshold could not separate the two surfaces. Several images also had a background bead to the right of the target, confusing the boundary detector. Heuristic fixes for any one of these problems tend to break the others. This motivated the transition to deep learning segmentation.

CNN Segmentation Model

For each image, the trained U-Net will produce a binary segmentation mask. Rather than assuming a fixed plate row, the algorithm identifies the median row of the bottom plate region in the mask, then scans upward from that point to find the sharpest brightness transition. That transition corresponds to the plate top surface. This image-specific plate row detection is the main reason the CNN algorithm succeeds on the 18 images where the classical algorithm failed.

The segmentation model is a U-Net with a ResNet34 encoder pretrained on ImageNet, implemented using the segmentation-models-pytorch library. The U-Net architecture was chosen for three reasons. First, its encoder-decoder structure with skip connections preserves spatial resolution at the output. The bead surface edge is a one-pixel-wide boundary that must be precisely localized, so full output resolution is essential. Second, U-Net generalizes well from small training datasets when a pretrained encoder is used. The encoder captures general visual features and only the decoder needs to learn domain-specific patterns. Third, U-Net avoids the spatial size collapse encountered with DeepLabV3+ at this image resolution and batch size of one. The Atrous Spatial

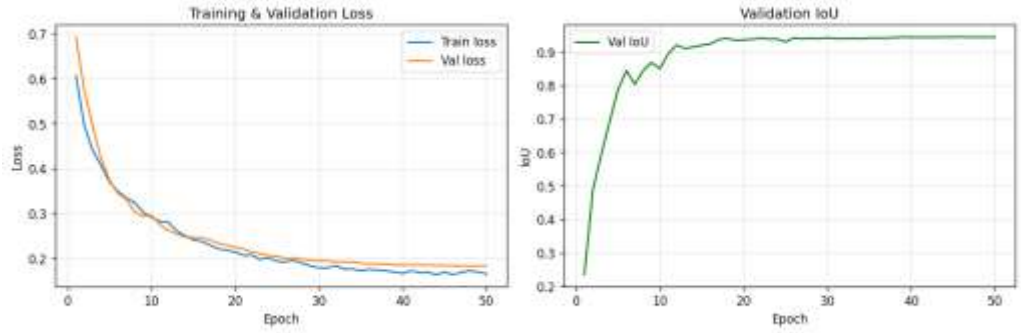
Pyramid Pooling module in DeepLabV3+ compressed spatial dimensions to a 1×1 map, destroying all positional information (Chen et al., 2018).

Twenty-two bead images were manually annotated using LabelMe (Wada, 2016), an open-source Python tool for adding polygon labels to images. A single polygon was drawn tracing the bead outline in each image. The annotations were converted to binary PNG masks and split into 18 training and 4 validation images using a fixed random seed. Training augmentations included horizontal flip (probability 0.5), random brightness and contrast adjustment (0.4), Gaussian noise (0.3), and Contrast Limited Adaptive Histogram Equalization (CLAHE) normalization.

Training ran for 50 epochs with an AdamW optimizer at learning rate 1×10^{-4}, cosine annealing schedule, and a combined Binary Cross-Entropy and Dice loss weighted equally. The region-of-interest image size used during training was 384 × 768 pixels, with a batch size of 4. The best validation Intersection over Union (IoU) of 0.93, a measure of how well the predicted bead region overlaps with the hand-labeled ground truth, where 1.0 means perfect overlap, was reached at epoch 20 and held through epoch 50 with no sign of overfitting. This indicates that 22 labeled images are adequate for this task when starting from an ImageNet-pretrained encoder.

Figure 2 shows the training convergence curves. The left plot tracks training loss and validation loss across all 50 epochs while both curves decrease steadily and converge by epoch 30 with no divergence between them, confirming that the model did not overfit to the training images. The right plot shows validation IoU across epochs, which climbs quickly and stabilizes at 0.93 by epoch 10, remaining consistent through the end of training.

Figure 2. *U-Net training convergence*



Bead Edge Extraction and Measurement

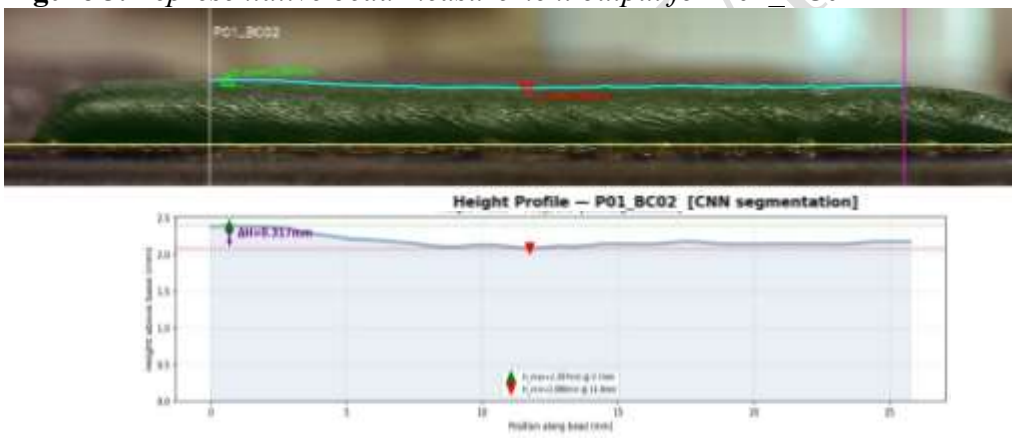
The bead top edge is extracted as the topmost white pixel per column of the mask, above the detected plate row. The raw edge undergoes three-pass spike removal, pass 1 uses a rolling median deviation tolerance of 20 pixels, pass 2 tightens to 12 pixels, and pass 3 to 7 pixels. After spike removal, a Savitzky-Golay filter (window = 41, polynomial order = 3) smooths the profile. Per-bead measurement boundaries are stored in trim_config.json as a left limit and a right limit. These exclude the bead start ramp, where height builds up to steady state, and the bead end zone, where spatter or wire retraction drops the height. The

limits were set interactively for each of the 69 beads using an OpenCV tool. The operator views the detected edge overlaid on the image and clicks the correct boundaries. Figure 3 shows a representative annotated output.

From the trimmed, smoothed profile, H_{max} is the maximum pixel height converted to millimeters, H_{min} is the minimum, and $\Delta H = H_{max} - H_{min}$. The position of H_{max} and H_{min} along the bead axis is also recorded.

Figure 3 shows the measurement output for a representative bead, P01_BC02. The top panel displays the original side-profile photograph with the CNN segmentation overlay highlighted in color, showing the detected bead boundary and the measurement annotations for H_{max} and H_{min} marked at their respective positions along the bead. The bottom panel presents an enlarged graphical representation of the extracted height profile in millimeters, plotted along the bead length, with H_{max}, H_{min}, and the resulting ΔH clearly labeled. This enlarged view makes it easier to see how height varies across the bead and where the peak and minimum points fall relative to each other.

Figure 3. *Representative bead measurement output for P01_BC02*



Cross-Batch Validation

CMM measurements were taken directly on all 69 deposited steel beads. The probe was run along each bead to record H_{max} and H_{min} at consistent intervals, producing a physical reference dataset tied to the same beads photographed by the vision algorithm. Vision and CMM outputs were then compared bead by bead. Match quality was classified as high (H_{max} error below 0.05 mm), medium (0.05 to 0.183 mm), or low (above 0.183 mm). 62 of 69 beads (90%) fell in the high category, confirming that the vision algorithm produces H_{max} values consistent with direct physical measurement across the full parameter range.

Gaussian Process Regression Model

Model Formulation

GPR models the unknown function $f(x)$ by mapping process parameters x = [feed rate, wire feed rate, nozzle height, trim] to ΔH as a sample from a Gaussian process with mean zero and covariance described by a kernel function $k(x, x')$. Given n training observations $\{X, y\}$, the predictive distribution at a new point x^* is Gaussian with mean and variance, as shown in equation 1 bellow:

$$\mu(x^*) = k(x^*, X)[K(X, X) + \sigma_n^2 I]^{-1} y \quad (1)$$

$$\sigma^2(x^*) = k(x^*, x^*) - k(x^*, X)[K(X, X) + \sigma_n^2 I]^{-1} k(X, x^*) \quad (2)$$

where $[K(X, X)$ is the $n \times n$ training kernel matrix and σ_n^2 is the noise variance, see equation 2. Both equations have closed-form solutions, so prediction requires no iterative inference (Rasmussen and Williams 2006). The 95% confidence interval at x^* is $\mu(x^*) \pm 1.96\ \sigma(x^*)$. This interval is the primary output reported for the optimal parameter recommendation.

Kernel and Hyperparameters

The GPR model uses a Matérn kernel with smoothness parameter $\nu = 2.5$. This controls how the model interpolates between observed data points. It was chosen over the squared exponential kernel because welding processes are not perfectly smooth. A small change in feed rate or nozzle height can produce a noticeable shift in bead geometry, and the Matérn kernel captures this more realistically. All four input parameters were scaled to the same range before fitting, so no single parameter dominates due to its units. The three kernel hyperparameters controlling prediction amplitude, length scale, and noise level were tuned automatically from the training data, using 15 independent starting points to avoid local optima.

Training and Parameter Search

The three beads classified as LOW confidence in the cross-batch validation, specifically P04_BC05, P05_BC05, and P05_BC06, all had H_{max} match errors above 0.183 mm and were excluded from the training set, leaving 66 samples. Model performance was evaluated on the training set (MAE, R^2, Pearson r) and by five-fold cross-validation (CV), in which the 66 samples were split into five roughly equal folds and each fold was predicted by a model trained on the remaining four. CV provides an estimate of out-of-sample prediction error that is independent of the training set fit.

After training, the model was evaluated at all 162 combinations in the discrete parameter grid (9 feed rates × 2 wire feeds × 3 nozzle heights × 3 trims). Each combination received a predicted mean ΔH and a 95% confidence interval. The combination with the lowest predicted mean is reported as the recommended parameter setting.

How We Used Parameters to Predict Variation in Height (ΔH)

Each bead in the dataset has four input parameters and one measured ΔH value from the vision algorithm. We used these as input-output pairs to train the GPR model. The four inputs are feed rate, wire feed rate, nozzle height, and trim. The output is the ΔH measured from the bead photograph. So, the model learns what combinations of those four parameters tend to produce flat beads and which ones tend to produce variable ones.

Before fitting the model, all four parameters were normalized to a common scale. Feed rate, wire feed rate, nozzle height, and trim each have different units and ranges. Normalizing puts them on equal footing so the model does not treat feed rate as more important just because its raw numbers are larger. Scikit-learn's StandardScaler was used to do this, which shifts each input to zero mean and unit variance.

The GPR model was then fit using GaussianProcessRegressor package from scikit-learn (Pedregosa et al., 2011). The kernel was set to a constant multiplied by a Matérn kernel with $\nu = 2.5$, plus a White kernel for noise. Hyperparameters were tuned by maximizing log marginal likelihood, with 15 restarts to avoid getting stuck in a local optimum. Once the model is trained, predicting ΔH for any new parameter combination is a single function call that returns both a mean prediction and a standard deviation.

To find the best setting, we built a full grid of all 162 parameter combinations from the experimental ranges in Table 1. We passed every combination through the trained model and collected the predicted ΔH and the 95% confidence interval for each one. The combination with the lowest predicted mean ΔH was selected as the recommendation. This grid search approach is straightforward and works well here because the parameter space is discrete and not too large. A more complex search strategy like Bayesian optimization would be useful if the parameter space were continuous or much larger (Snoek et al., 2012).

Results

Vision Algorithm Measurement Outputs

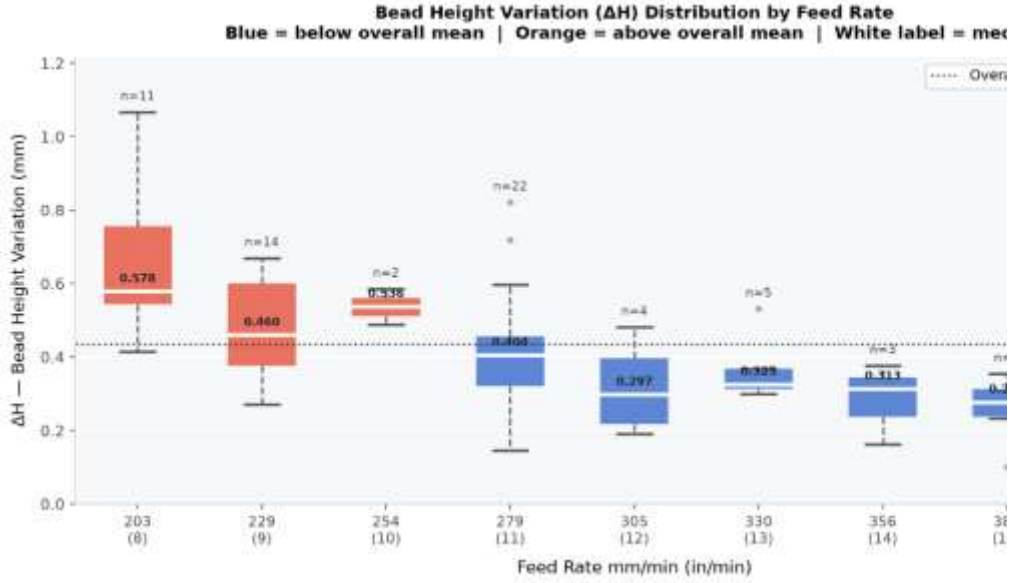
The algorithm successfully processed all 69 bead images. Table 2 summarizes the measurement statistics. Height variation (ΔH) ranged from 0.100 mm for the flattest bead (P10_BC04) to 1.067 mm for the most variable (P05_BC06). The distribution is right-skewed, with most beads falling between 0.2 and 0.6 mm and a smaller group above 0.7 mm. The calibration scale of 66.34 px/mm was consistent across all images, confirming that the fixed camera protocol maintained reproducibility across all 69 beads.

Table 2. *Vision Algorithm Measurement Summary*

Metric	Value	Remarks
Beads processed	69	Plates P01–P12
Calibration scale	66.34 px/mm	Plate thickness = 6.15 mm
H_{max} range (mm)	1.991 – 3.878	
H_{min} range (mm)	1.695 – 3.457	
ΔH range (mm)	0.100 – 1.067	
Mean ΔH (mm)	0.434	SD = 0.184 mm
Flattest bead	P10_BC04: ΔH = 0.100 mm	
Most variable bead	P05_BC06: ΔH = 1.067 mm	
CNN segmentation IoU	0.93	Best validation epoch 20/50
Classical algorithm success rate	51/69 (74%)	Before CNN adoption

Figure 4 shows the ΔH distribution across all 69 beads grouped by feed rate. Median ΔH drops from 0.578 mm at 203 mm/min (8 in/min) to 0.348 mm at 406 mm/min (16 in/min). The spread also tightens at higher feed rates, meaning not just lower average variation but more consistent bead geometry. Feed rate is clearly the dominant driver of bead flatness.

Figure 4. *Variation in Height (ΔH) distribution by feed rate across all 69 beads*



Cross-Batch Validation Results

Table 3 presents the cross-batch validation metrics. The most operationally relevant result is H_{max} agreement. 62 of 69 bead pairs (90%) fell in the HIGH confidence category with a mean match error of 0.017 mm against the physical reference data, which is well within the tolerance needed for machining

allowance decisions. The Pearson correlation between vision-derived ΔH and physical ΔH was r = 0.361 (p = 0.0023), indicating a statistically significant but moderate linear relationship between the two datasets. A correlation this low does not mean the algorithm is unreliable, but it reflects the fact that ΔH was measured across two separate deposition campaigns under slightly different conditions, introducing genuine variability in the bead profiles that neither dataset can fully account for. Both datasets agree closely on H_{max}, but the full height profiles differ between campaigns, producing a +0.212 mm mean offset in ΔH. That offset reflects genuine deposition variability, not a systematic algorithm bias. For machining purposes, H_{max} is the metric that matters, and on that measure the algorithm performs well.

Table 3. *Cross-Batch Validation Metrics*

Metric	Value	Interpretation
Total matched bead pairs	69	
HIGH confidence (H_max error < 0.05 mm)	62 (90%)	Excellent H_{max} agreement
MEDIUM confidence (0.05–0.183 mm)	4 (6%)	
LOW confidence (> 0.183 mm)	3 (4%)	Excluded from GPR training
Mean H_max match error	0.017 mm	
Max H_max match error	0.183 mm	3 bead pairs
Pearson r (ΔH: vision vs. physical)	0.361	p = 0.0023
Mean ΔH offset (vision − physical)	+0.212 mm	Reflects batch differences

GPR Model Performance

Table 4 presents the GPR model performance metrics. After hyperparameter optimization by log marginal likelihood maximization, the kernel settled on a Matérn length scale of 4.46 and a white noise level of 0.500, which is at the upper bound. That high noise level indicates the training labels contain variability that the four input parameters alone cannot explain. Some of that reflects real process scatter. Some reflects the cross-batch offset discussed above. Training MAE is 0.096 mm and R^2 is 0.375 on the training set. The five-fold cross validation MAE of 0.117 ± 0.029 mm is the more useful number, as it estimates out-of-sample error on unseen beads. The cross validation R^2 of −0.104 is discussed in Section 5.

Table 4. *GPR Model Performance Metrics*

Metric	Value	Notes
Training samples	66	HIGH + MEDIUM confidence
Excluded (LOW confidence)	3	P04_BC05, P05_BC05, P05_BC06
Kernel (optimized)	Constant × Matérn(ν=2.5) + White	
Length scale ℓ	4.46 (normalized)	

Noise level σ^2_n	0.500	At upper bound
Log marginal likelihood	−91.59	
Training MAE	0.096 mm	
Training R^2	0.375	
Training Pearson r	0.625	$p < 0.001$
5-Fold CV MAE	0.117 ± 0.029 mm	
5-Fold CV R^2	−0.104 ± 0.133	See Section 5; negative CV R^2 expected when label noise ≈ target variance
Mean 95% CI width	±0.233 mm	

The five-fold cross-validation R^2 of −0.104 means that when the model is tested on beads it has not seen during training, it does not predict ΔH better than simply guessing the dataset average every time. This sounds like a poor result but it is actually expected given the nature of the data. The ΔH values used to train the model came from two separate deposition campaigns, the vision-derived measurements and the CMM reference measurements, and these two campaigns produced a mean offset of roughly 0.2 mm between them. That offset is not a measurement error but a reflection of real differences in how the beads were deposited across the two sessions. The problem is that this cross-batch variability is about the same size as the natural spread of ΔH values in the dataset, which has a standard deviation of 0.184 mm. When the noise in the training labels is as large as the variation the model is trying to predict, no model, regardless of how it is built, can produce a reliable cross-validation R^2. The GPR model recognizes this situation on its own. During training, the kernel optimization pushed the noise parameter to its maximum allowed value, which is the model's way of signaling that the labels contain variability it cannot explain from the four input parameters alone. This is not a failure, it is the model being honest about the limits of the available data.

Figure 5 plots the measured ΔH for all 69 beads against feed rate, with the GPR predicted trend overlaid. The green line drops from a predicted 0.578 mm at 8 in/min to 0.235 mm at 16 in/min, confirming that higher feed rates produce more uniform beads. The shaded band shows the 95% confidence interval, which widens at the parameter space boundaries where training data is sparse.

Recommended Parameter Combination

Figure 5. *ΔH vs. feed rate — measured beads and GPR predicted trend*

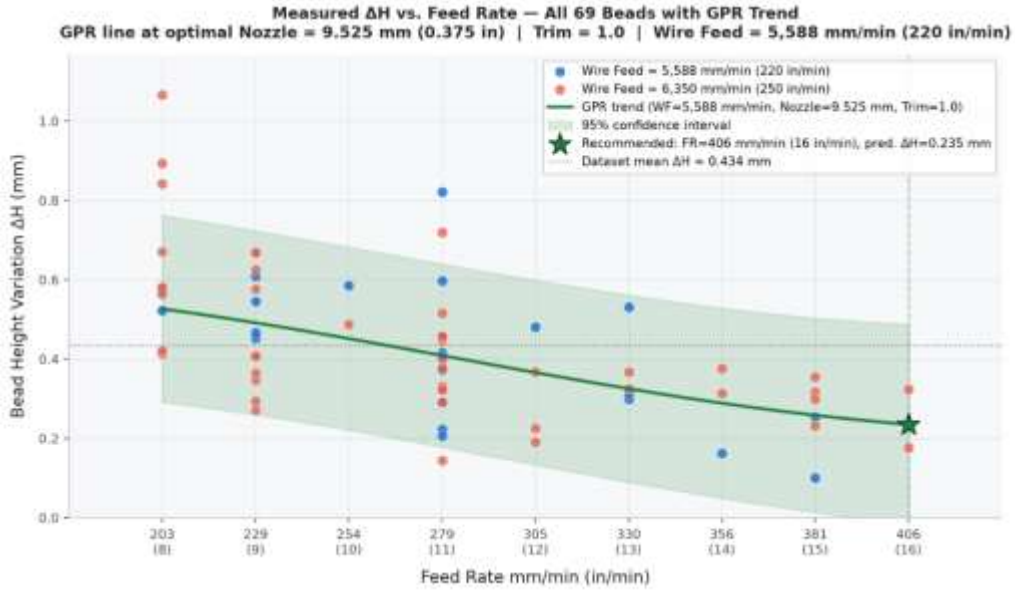


Table 5 presents the candidate parameter combination identified by grid search across all 162 combinations in the experimental parameter space. At Feed Rate = 406 mm/min (16 in/min), Wire Feed Rate = 5,588 mm/min (220 in/min), Nozzle Height = 9.525 mm (0.375 in), and Trim = 1.0, the model predicts ΔH = 0.235 mm with a 95% confidence interval of [−0.017, 0.487] mm. This represents a 46% reduction from the dataset mean of 0.434 mm. This 95% CI spans roughly 0.5 mm, comparable to the full range of ΔH observed across the dataset (0.100–1.067 mm, Table 2), indicating that this recommendation should be treated as a hypothesis for the validation beads proposed in the Discussion rather than a confirmed result. The lower CI bound crossing zero reflects prediction uncertainty at the edge of the training data distribution. It should be read as indicating that a near-flat bead (ΔH ≈ 0) is plausible at these settings, not that the model predicts an impossible negative height variation.

Table 5. *Recommended Parameter Combination*

Parameter	Recommended Value	Unit
Feed Rate	406 (16)	mm/min (in/min)
Wire Feed Rate	5,588 (220)	mm/min (in/min)
Nozzle Height	9.525 (0.375)	mm (in)
Trim	1.0	dimensionless
Predicted ΔH (mean)	0.235	mm
95% CI	[−0.017, 0.487]	mm
Prediction uncertainty σ	0.129	mm
Reduction vs. dataset mean	46%	0.434 mm → 0.235 mm

Discussion

The most practically significant result from the vision algorithm is the 0.017 mm mean H_{max} error against the physical reference dataset. The machining step on the hybrid system typically removes 0.5 to 1.5 mm per pass. Peak bead height is known to within 0.017 mm, which is well within the tolerance needed for machining allowance planning. That accuracy comes from a fixed high-resolution camera and polygon annotation across 22 images.

The classical algorithm's 74% success rate means the basic imaging geometry works in three out of four cases with no machine learning at all. The remaining 26% is not random noise, but failures cluster in beads caused by varied plate row position, a low-contrast bead, or a background bead bleeding into the detection zone. Heuristic patches for any one of those failure modes tend to break the others. A CNN trained on a diverse image set handles all three without separate fixes. To apply this method on a new WAAM process, it is recommended to start with the classical algorithm, then train the CNN once 20 or more labeled images are annotated.

An IoU of 0.93 from 22 training images is a strong result for a domain-specific segmentation task. It means 93% of each bead's pixel area is correctly classified on held-out validation images. Misclassification mostly affects isolated boundary pixels, which are averaged out by the Savitzky-Golay smoothing step. The key enabler is ImageNet pretraining on the ResNet34 encoder. General edge and texture knowledge from that pretraining transfers well enough that only 22 domain-specific annotations were needed.

The response surface shows physical consistency and practical usefulness. We found that feed rate has the dominant effect amongst all. Higher travel speed means less energy deposited per unit length, which produces a flatter and more uniform bead profile. That trend runs across the full parameter range and is robust to label noise because it is a global pattern rather than a local one. An operator can read the response surface and immediately know to stay in the high-feed-rate zone without parsing any statistical output.

Unlike deterministic models such as neural networks or Random Forest, GPR reports not just a predicted ΔH value but also how confident that prediction is at every point in the parameter space (Rasmussen and Williams, 2006). This confidence information is useful because it tells the operator where the model's recommendations can be trusted and where more experimental data is needed. Confidence is highest near feed rates of 254 to 305 mm/min (10 to 12 in/min), where the training dataset has the most beads and the model has seen enough examples to make reliable predictions. The recommended optimum sits at 406 mm/min (16 in/min), which is at the edge of that well-sampled zone. Because fewer training beads exist at that setting, the model is less certain there and reports a wider confidence interval — a prediction uncertainty of $\sigma = 0.129$ mm and a 95% CI width of 0.254 mm. In practical terms this means the recommended setting is promising but not yet fully confirmed by the available data. Depositing three to five validation beads at that setting and comparing the measured ΔH against the predicted range would directly confirm whether the

recommendation holds in practice and would also narrow the confidence interval for that region of the parameter space.

The recommended setting of Feed Rate = 406 mm/min (16 in/min), Wire Feed Rate = 5,588 mm/min (220 in/min), Nozzle Height = 9.525 mm (0.375 in), and Trim = 1.0 pairs the highest feed rate tested in this study, sitting at the boundary of the experimental range, with a lower wire feed and smallest nozzle standoff. Higher feed rate means the torch covers more path per unit time, which reduces heat input per unit length and physically explains the flatness gain. As a secondary benefit, the higher travel speed also improves throughput. The trained GPR model supports interactive use beyond the grid search. Given any parameter combination, it returns a predicted ΔH with a 95% confidence interval in seconds, which is useful when material constraints or joint geometry force a setting away from the recommended optimum and the operator needs to know the expected flatness penalty before committing to a build.

Future work should include the deposition of additional beads at high-uncertainty parameter combinations, particularly at low feed rates and the extremes of nozzle height, which would substantially reduce confidence interval width across the response surface. Also, moving the system toward the real-time closed-loop control by capturing the bead image immediately after deposition while the machine repositions for the next pass within a build.

Conclusions

A U-Net segmentation model with a ResNet34 encoder trained on only 22 manually labeled images successfully achieves a segmentation IoU of 0.93, confirming that ImageNet-pretrained encoders enable remarkably effective manufacturing segmentation with limited annotation effort. This is a strong result compared to classical segmentation approaches, which typically require far larger labeled datasets to reach comparable accuracy and still struggle with variable imaging conditions. The vision algorithm derived from this model reliably measures bead peak height with a consistently low mean error against physical reference data across 90% of matched bead pairs, achieving a level of measurement accuracy that is well suited for machining allowance planning in a production WAAM environment.

Classical computer vision methods achieve 74% success on the 69-bead dataset but fail systematically on images with variable plate position, low contrast, or background interference. The CNN handles all three of these effectively and without separate fixes because it learned directly from labeled bead images rather than relying on fixed rules, demonstrating clear superiority over classical approaches in challenging imaging conditions.

The GPR model results are consistent with higher feed rates producing lower ΔH, which lines up well with the physics of reduced heat input per unit length. The model also clearly shows that the high-feed-rate region has the widest prediction intervals, meaning that depositing a few more beads there would yield the most reliable and confident parameter recommendations for that zone.

The recommended parameter combination of Feed Rate = 406 mm/min (16 in/min), Wire Feed Rate = 5,588 mm/min (220 in/min), Nozzle Height = 9.525 mm (0.375 in), and Trim = 1.0 predicts ΔH = 0.235 mm, a considerable 46% reduction from the dataset mean, pending experimental validation. The accompanying 95% confidence interval communicates this prediction uncertainty directly and transparently to the operator rather than presenting a bare point estimate.

The complete algorithm runs autonomously without manual intervention and requires no specialized metrology equipment beyond a high-resolution fixed camera, making it readily and practically deployable in WAAM environments where structured-light or CMM-based measurement is impractical.

This work considerably advances the hybrid 3D metal printing research program at Western Michigan University (Ikonomov, 2020; Ikonomov, 2025) with a robust, data-driven measurement and optimization layer that brings the process closer to fully automated unattended operation. Expansion of the training dataset to boundary parameter regions is identified as a promising next step toward achieving comprehensive in-process optimization.

References

Chen LC, Zhu Y, Papandreou G, Schroff F and Adam H (2018) Encoder-decoder with atrous separable convolution for semantic image segmentation. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018, pp. 801–818.

Denlinger ER, Heigel JC, Michaleris P and Palmer TA (2015) Effect of inter-layer dwell time on distortion and residual stress in additive manufacturing of titanium and nickel alloys. Journal of Materials Processing Technology 215: 123–131.

Ding D, Pan Z, Cuiuri D and Li H (2015) Wire-feed additive manufacturing of metal components. International Journal of Advanced Manufacturing Technology 81(1–4): 465–481.

Geng H, Xiong J, Huang D, Lin X and Li J (2017) A prediction model of layer geometrical size in wire and arc additive manufacture using response surface methodology. International Journal of Advanced Manufacturing Technology 93(1–4): 175–186.

Guo N and Leu MC (2013) Additive manufacturing: technology, applications and research needs. Frontiers of Mechanical Engineering 8(3): 215–243.

He K and Sun J (2015) Convolutional neural networks at constrained time cost. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015.

Ikonomov P (2020) Hybrid manufacturing system. U.S. Patent No. 10,688,581 B2. Washington, DC: U.S. Patent and Trademark Office.

Ikonomov P (2025) Hybrid 3D metal printing process optimization using ML and AI. Athens Journal of Technology and Engineering 12(3): 181–194.

Jafari D, Vaneker THJ and Gibson I (2021) Wire and arc additive manufacturing: opportunities and challenges to control the quality and accuracy of manufactured parts. Materials and Design 202: 109471.

Kingma DP and Ba J (2015) Adam: a method for stochastic optimization. In Proceedings of the International Conference on Learning Representations (ICLR), San Diego, CA, USA, 7–9 May 2015.

Li Y, Han Q, Zhang G and Horváth I (2018) A layers-based approach for reducing distortion in WAAM. International Journal of Advanced Manufacturing Technology 96(9–12): 3809–3820.

Loshchilov I and Hutter F (2019) Decoupled weight decay regularization. In Proceedings of the International Conference on Learning Representations (ICLR), New Orleans, LA, USA, 6–9 May 2019.

Martina F, Mehnen J, Williams SW, Colegrove P and Wang F (2012) Investigation of the benefits of plasma deposition for the additive layer manufacture of Ti-6Al-4V. Journal of Materials Processing Technology 212(6): 1377–1386.

Müller AC and Guido S (2016) Introduction to Machine Learning with Python. Sebastopol, CA: O'Reilly Media.

Pan SJ and Yang Q (2010) A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering 22(10): 1345–1359.

Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M and Duchesnay E (2011) Scikit-learn: machine learning in Python. Journal of Machine Learning Research 12: 2825–2830.

Rasmussen CE and Williams CKI (2006) Gaussian Processes for Machine Learning. Cambridge, MA: MIT Press.

Ribeiro MT, Singh S and Guestrin C (2016) Why should I trust you? Explaining the predictions of any classifier. In Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016, pp. 1135–1144.

Ronneberger O, Fischer P and Brox T (2015) U-Net: convolutional networks for biomedical image segmentation. In Proceedings of the Medical Image Computing and Computer-Assisted Intervention (MICCAI), Munich, Germany, 5–9 October 2015, Lecture Notes in Computer Science, vol. 9351. Springer, Cham.

Shin JH (2024) Prediction of metal additively manufactured bead geometry using deep neural network. Sensors 24(19): 6250.

Snoek J, Larochelle H and Adams RP (2012) Practical Bayesian optimization of machine learning algorithms. In Advances in Neural Information Processing Systems (NIPS), Lake Tahoe, NV, USA, 3–6 December 2012, vol. 25, pp. 2951–2959.

Tapia G, Elwany AH and Sang H (2016) Prediction of porosity in metal-based additive manufacturing using spatial Gaussian process models. Additive Manufacturing 12: 282–290.

Wada K (2016) LabelMe: image annotation tool. GitHub repository. Available at: https://github.com/wkentaro/labelme (accessed June 2026).

Wang Z, Zimmer-Chevret S, Léonard F and Abba G (2021) Prediction of bead geometry with consideration of interlayer temperature effect for CMT-based wire-arc additive manufacturing. Welding in the World 65(12): 2255–2266.

Williams SW, Martina F, Addison AC, Ding J, Pardal G and Colegrove P (2016) Wire + arc additive manufacturing. Materials Science and Technology 32(7): 641–647.

Xiao X, Waddell C, Hamilton C and Xiao H (2022) Quality prediction and control in wire arc additive manufacturing via novel machine learning framework. Micromachines 13(1): 137.

Xiong J, Lei Y, Chen H and Zhang G (2020) Thermal process and microstructure evolution of thin-walled parts fabricated by WAAM with flux-cored wires. Journal of Manufacturing Processes 52: 431–442.

Xiong J, Zhang G, Hu J and Wu L (2014) Bead geometry prediction for robotic GMAW-based rapid manufacturing through a neural network and a second-order regression analysis. Journal of Intelligent Manufacturing 25(1): 157–163.

1 Zuiderveld K (1994) Contrast limited adaptive histogram equalization. In Heckbert PS (ed)
2 Graphics Gems IV. San Diego, CA: Academic Press, pp. 474–485.