Online Thermal Profile Prediction for Large Format Additive Manufacturing: A Hybrid CNN-LSTM based Approach

Lu Liu[1], Feng Ju*[1], and Seokpum Kim[2]

[1]*School of Computing and Augmented Intelligence, Arizona State University, Tempe, AZ*
[2]*Manufacturing Science Division, Oak Ridge National Laboratory, Oak Ridge, TN*

Abstract

Large format additive manufacturing (LFAM) is an advanced 3D printing technique that efficiently fabricates large-scale components through a layer-by-layer extrusion and deposition process. Accurate surface layer temperature monitoring is essential to prevent manufacturing failures and ensure final product quality. Traditional physics-based offline approaches for simulating thermal behavior are often inefficient and complex, posing challenges on real-time, in-situ monitoring. To address this, we propose a data-driven hybrid CNN-LSTM model to predict sequential thermal images of arbitrary length using real-time infrared thermal imaging. In this approach, a CNN is trained offline to capture spatial features, reduce dimensional complexity, and enhance time efficiency, while a stacked LSTM is applied online to capture temporal information for improved prediction of future thermal behavior in subsequent printing layers. Model performance is evaluated using MSE, SSIM, and PSNR metrics and is benchmarked against stacked LSTM and convolutional LSTM models, demonstrating superior accuracy and applicability. Additionally, to mitigate noise from moving extruders and gantry backgrounds in thermal images, a fine-tuned semantic segmentation model is impletemented offline to extract printing geometry, enabling precise temperature tracking along the tool path for further thermal analysis. The frameworks developed in this study significantly advance temperature monitoring, thermal analysis, and in-situ manufacturing control for LFAM, bridging the gap between theoretical modeling and practical application.

Keywords: Large format additive manufacturing; thermal image prediction; geometry extraction; hybrid CNN-LSTM; semantic segmentation

1 Introduction

Large Format Additive Manufacturing (LFAM) is an advanced subset of additive manufacturing (AM) that enables the layer-by-layer fabrication of large, complex structures through precise material deposition [1]. In LFAM, a physical object is built by following a G-code, which translates a Computer-Aided Design (CAD) model—a three-dimensional geometric design—into step-by-step instructions for the printing process. A nozzle, equipped with pelletized feedstock, commonly carbon fiber reinforced polycarbonate (CF/PC) for its strength and durability [2], moves to extrude and deposit material layer

*Corresponding author email: fengju@asu.edu

by layer, forming the desired geometries. This technology is widely used in industries requiring large, customized components, such as aerospace, automotive, and construction, offering benefits like minimized material waste and the ability to produce complex geometries that are challenging with traditional manufacturing methods [3].

Given the scale of production and characteristics of the thermal plastic material used in LFAM, temperature monitoring and real-time control are critical, as they directly affect the quality, mechanical properties, and overall cost efficiency of the printed structures. Quality issues such as debonding, cracking, and warping frequently occur when the surface layer temperature of the geometry cools excessively, while an overheated surface layer can lack sufficient stiffness to support subsequent layers, leading to collapse, as confirmed in [4, 5]. Compared to anomaly detection in post-adjustment quality control, preventing defects through real-time temperature monitoring offers a more effective approach to reducing manufacturing costs. By addressing temperature irregularities early through thermal prediction, this method can prevent significant defects, reduce costly reprinting, and maintain the structural integrity and precision of components manufactured through LFAM.

In extrusion additive manufacturing, thermal predictions are typically approached in two primary ways: physics-based models and data-driven models. Finite element analysis (FEA) is a widely used physics-based approach. Zhang et al. [6] proposed a three-dimensional FEA model incorporating element activation to simulate the mechanical and thermal behavior in the fused deposition modeling (FDM) process. Additionally, the Additive3D toolset was developed to model temperature history during deposition, along with residual stresses and deformations of printed parts in extrusion, by considering polymer crystallization, thermoviscoelastic stress relaxation, and combined thermomechanical and crystallization shrinkage behavior of materials [7]. To reduce the time demands of complex 3D FEA simulations for LFAM [8], a simplified 1D heat transfer model that ignores geometric influences has been studied, achieving accurate thermal profiles by accounting for conduction, natural convection, and radiation [9]. However, physics-based models are typically highly complex, time-consuming, and require extensive knowledge of the underlying physical processes. Additionally, since most physics-based models rely on specific assumptions and conditions, their generalizability and accuracy can be limited when exposed to scenarios involving environmental uncertainties. In response, an increasing number of researchers are leveraging data-driven approaches, including machine learning and deep learning. For temperature prediction, data-driven models offer significant advantages over physics-based models [10], as they excel in capturing complex, non-linear processes with enhanced accuracy and efficiency.

In large format additive manufacturing, an infrared (IR) camera can be employed to monitor surface-layer temperature profiles, providing both top and side-view perspectives. The thermal data captured by IR cameras is affected by both spatial and time-series dependencies, as temperature observations at each monitored position are correlated over time. Traditional time-series models, such as ARIMA, handle temporal correlations but often overlook spatial dependencies. In related work on Direct Metal Deposition (DMD), an ensemble of bagged decision trees was used to predict subsequent voxel temperatures, capturing these dependencies to some extent [11]. A regression model was also used in [12] to capture the relationship between temperature cooling rate and temperature differences from ambient temperature, achieving real-time surface layer temperature prediction in LFAM for simple, single-bead geometries. However, in the case of multi-bead structures, the thermal profile of one printed position can rebound when an adjacent position begins printing, and heat transfer dynamics shift due to changes in free surface area and ambient temperature. Consequently, the complexity of cooling behavior increases, limiting the effectiveness of simple regression models for predicting thermal profiles in complex, multi-bead geometries. Therefore, in addition to machine learning methods, deep learning techniques, particularly neural network-based architectures, have demonstrated remarkable capabilities in handling noisy and correlated time-series imaging data in recent years [13]. These specialized neural network architectures, such as

recurrent neural networks (RNNs) and convolutional neural networks (CNNs) [14], can significantly enhance forecasting accuracy, especially in highly complex scenarios. Therefore, utilizing deep learning methods is likely to be a more effective approach for predicting layer temperatures in complex printing structures.

Convolutional Neural Networks, among the most widely used deep learning techniques, have traditionally been applied to tasks such as image classification, recognition, and semantic segmentation in computer vision due to their capability to effectively extract spatial features and reduce data dimensionality through pooling convolutions [15]. Recently, CNNs have seen increased use in additive manufacturing, particularly for defect and anomaly detection, to help maintain product quality [16, 17]. However, CNNs alone typically do not capture sequential dependencies in data. In contrast, Long Short-Term Memory (LSTM) networks, a form of RNN, excel in handling time-series data [18]. This has led researchers to develop hybrid CNN-LSTM models to integrate both spatial and temporal characteristics for improved prediction accuracy. Studies comparing CNN-LSTM models with other machine learning approaches, such as Support Vector Machines (SVM) and Random Forests (RF), report that CNN-LSTM models outperform these methods in predictive performance [19, 20].

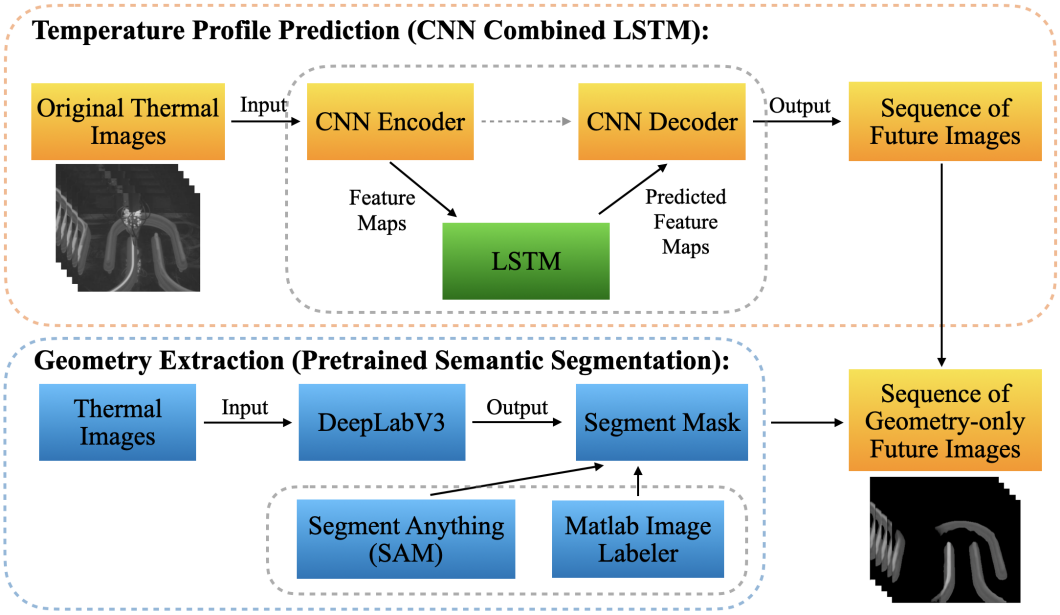


Figure 1: The framework of the proposed temperature profile prediction.

To the best of our knowledge, there has been little research on predicting temperature profiles for entire captured thermal images, rather than individual pixel locations, in the context of large format additive manufacturing. To address this gap, we take 3D thermal videos as input data and propose a combination of deep CNN and stacked LSTM architecture that utilizes both temporal and spatial information from thermal frames. This approach aims to develop a more generalized thermal prediction model, suited to the complex structures in LFAM. The model's performance is evaluated using three commonly used metrics: Mean Squared Error (MSE), Structural Similarity Index Measure (SSIM), and Peak Signal-to-Noise Ratio (PSNR). Final prediction results are then compared against other deep learning methods effective in handling sequential data, including stacked LSTM and convolutional LSTM (ConvLSTM) architectures. Additionally, in real-world applications, understanding the temperature profile along the entire printing path is crucial for those concerned with manufacturing quality and efficiency. However, the movement of the printing head and the noisy background in thermal images complicate the efficient and

automatic generation of temperature cooling dynamics for various locations along the path. To mitigate this issue and reduce human intervention, we fine-tune a pre-trained semantic segmentation model using thermal images to accurately capture the printing geometry. This approach allows for the generation of preferred temperature profiles along the printing path for analytical purposes. The overview framework of the proposed methods in this paper is shown in Figure 1. Ultimately, this research aims to enhance future application-level analysis and manufacturing control.

The rest of paper is structured as follows. Section 2 reviews applications of deep learning methods and hybrid CNN-LSTM architectures. In Section 3, a detailed description of the proposed temperature profile prediction model (3.3) and geometry extraction model (3.4) is presented. Section 4 provides case studies evaluating the proposed model on two different datasets, and finally, Section 5 offers conclusions and directions for future research.

2 Literature Review

The development of deep learning has revolutionized numerous industries, particularly in computer vision, by enabling models to directly learn complex features from large image and video datasets. This advancement has led to unprecedented accuracy in tasks such as image classification, object detection, segmentation and video prediction. In CNN applications, the choice of convolutional filter size, number of filters, and pooling layers significantly impacts model performance [21]. Notable deep CNN architectures like VGG16 [22] and ResNet [23] were developed to address image classification and recognition challenges, respectively. For image segmentation tasks, the UNet architecture, viewed as a convolutional autoencoder (CAE), is a fundamental model known for its accuracy [24]. UNet employs skip connections that link each encoder layer directly to its corresponding decoder layer at the same resolution level, mitigating spatial resolution loss from pooling operations and enhancing segmentation accuracy. Unlike image classification, recognition, and segmentation, video prediction is a spatiotemporal problem that requires analyzing both spatial and temporal features within sequential images. This complexity makes it necessary to develop hybrid models that integrate multiple network types to enhance prediction accuracy. Shi et al. [25] designed the ConvLSTM model specifically for spatiotemporal data, allowing both spatial and temporal processing to be integrated directly within each LSTM cell. Lotter et al. [26] developed the PredNet architecture, which leverages ConvLSTM within a predictive coding framework to propagate image prediction errors throughout the network for video prediction tasks. Liang et al. [27] combined a CNN autoencoder with ConvLSTM and two generative adversarial networks (GANs) to encode motion representations from images, enabling the future-frame and future-flow generators to work collaboratively for improving video prediction. Villegas et al. [28] integrated an encoder-decoder LSTM with CNN layers to generate the next frame in a sequence, enabling more effective long-term video prediction. Yu et al. [29] proposed CrevNet, which employs a 3D CNN as a bidirectional autoencoder to prevent information loss, coupled with a spatiotemporal LSTM that bridges the autoencoder, achieving state-of-the-art results in video prediction. Gao et al. [30] developed a streamlined video prediction model by utilizing a CNN autoencoder with inception modules as translators to capture temporal information, outperforming many more complex network architectures in predictive accuracy. In recent years, emerging techniques such as video transformers and video diffusion models have also gained significant attention for their ability to generate and predict videos with high accuracy [31, 32, 33]. While most complex models often achieve impressive accuracy, they are computationally intensive, typically requiring tens of hours or even days to train due to their complexity and large-scale data requirements.

In additive manufacturing, many in-situ monitoring systems rely on thermal images captured by IR cameras and surface images collected by 3D scanner, a setup that closely resembles traditional computer

vision applications involving the analysis of images and videos. Due to the success of deep learning in computer vision, researchers are increasingly exploring deep learning for real-time defect detection and thermal prediction in AM, leveraging its image analysis capabilities to detect anomalies, predict thermal patterns, and enhance process control and product quality. In in-situ monitoring of 3D printing process, most defect detection tasks are framed as classification problems. For instance, in FDM, Jin et al. [34] fine-tuned a CNN classification model based on the ResNet50 architecture to identify in-situ extrusion issues, such as over-extrusion or under-extrusion, enabling real-time adjustments to printing parameters. In powder bed fusion AM (PBFAM), a spectral convolutional neural network (SCNN) was proposed to classify print quality into three categories online by detecting acoustic emission (AE) signals [35]. Additionally, Chen et al. [36] developed an artificial neural network (ANN) model to predict warping issues in fused filament fabrication (FFF) as a binary classification task. Li et al. [37] further explored a residual attention network to monitor FDM product quality by predicting dimensional deviations between printed parts and their CAD designs.

In addition, hybrid networks are increasingly used to leverage the strengths of multiple techniques for improved performance. For example, Guo et al. [38] combined Kernel Principal Component Analysis (Kernel PCA) and ARIMA with a stacked LSTM to predict temperatures in the heat-affected zone (HAZ) during FFF printing. This hybrid approach outperformed standalone LSTM and traditional RNN models in prediction accuracy. Ren et al. [39] developed an integrated model combining an LSTM-Autoencoder with K-means clustering to extract spectral features and classify deposition quality in directed energy deposition (DED) printing. A hybrid CNN-Transformer model was investigated in [40] to predict the 2D melt pool cross-sectional morphology in metal additive manufacturing. In this study, the Transformer component is used specifically to capture temporal information, and the proposed architecture is shown to outperform the conventional Vision Transformer (ViT) model in terms of prediction accuracy. In [41], a hybrid CNN-LSTM model was proposed to detect defects by predicting the next layer's surface and monitoring surface morphology in real-time during the FFF process. This study confirms that the hybrid CNN-LSTM model generally outperforms both LSTM and ConvLSTM models, particularly under limited training epochs. Additionally, rather than using this hybrid approach for defect detection, Nalajam et al. [19] applied it to predict melt pool temperatures in metal additive manufacturing using 1D input data. This approach demonstrates superior performance, not only over traditional machine learning methods like SVM and RF but also over other deep learning models, such as CNN, Bi-LSTM, and Attention-LSTM.

Although several studies in additive manufacturing have applied hybrid networks for in-situ monitoring of defect detection and thermal prediction, there is a notable gap in research focusing on 3D video-level thermal predictions. Specifically, there is a lack of studies addressing this application within large format additive manufacturing, where thermal images encompass more complex geometries and exhibit significantly varied temperature cooling dynamics across the monitored layers. Therefore, this paper proposes a hybrid CNN-LSTM architecture tailored for monitoring thermal images of large-scale geometries in LFAM. This approach aims to enable in-situ monitoring for preventing manufacturing failures and ensuring product quality in large format additive manufacturing.

3 Model Description

In LFAM, predicting temperature profiles with the proposed CNN-LSTM model presents two key challenges. First, a simple CNN struggles to capture the intricate spatial features in high-detail thermal images. Second, using large images as input ignificantly increases the node count in the LSTM, leading to higher computational complexity, greater memory demands, and potential redundancy—ultimately com-

promising predictive accuracy. To address these issues, we propose a deep convolutional autoencoder (CAE) to capture richer spatial features and reduce input dimensionality for the LSTM. The detailed architecture of the deep CNN combined with the LSTM encoder-decoder network will be discussed in Section 3.3.

From the application perspective, monitoring the temperature of recently printed paths is essential, yet noisy thermal images and a constantly moving print head complicate data extraction, often requiring continuous manual intervention. For each unique geometry, obtaining layer temperatures often demands custom image processing, incurring significant time and labor costs. Therefore, it is essential to develop a model capable of automatically extracting the temperatures of printed objects from thermal images, irrespective of their shape. The developed semantic segmentation model for geometry extraction is discussed in Section 3.4.

3.1 *Deep Convolutional AutoEncoders (CAE)*

The autoencoder consists of two main components: the encoder and the decoder. The encoder maps high-dimensional inputs to lower-dimensional latent representations, helping to ignore irrelevant features and reduce computational costs. Conversely, the decoder reverses this process, reconstructing the original input from the latent space representations. The convolutional autoencoder (CAE), which leverages the benefits of CNN, is one of the most popular autoencoder models for generating spatial information in images while reducing input dimensionality. In the CAE, convolutional layers replace the fully connected layers found in traditional autoencoders, allowing weights to be shared across different locations in the input data [42]. The latent representation of the lth feature map can be represented as:

$$h^l = \sigma(x * W^l + b^l) \tag{1}$$

where x is the gray-scale input, σ represents the activation function. Denote $*$ as the convolution operation, W as weights, and b as the bias value for the lth layer. The decoding part of reconstructing original images from latent representations can be described as:

$$y = \sigma(\sum_{l \in H} h^l * \tilde{W}^l + b) \tag{2}$$

where H is all feature maps, and $\tilde{W}^l$ is the weight for lth feature map which is flipped both horizontally and vertically.

When dealing with complex and detailed image inputs, deep convolutional networks (CNNs) have been shown to achieve higher accuracy in tasks such as image classification and object detection in computer vision. More abstract features are extracted. Therefore, inspired by very deep convolutional networks [22], a ConvNet architecture with 14 weight layers is applied as the encoder and feature extractor in the proposed model, and a reverse configuration is build as the decoder part. The Mean Squared Error (MSE) is used as the loss function to train the autoencoder by updating the network weights through backpropagation. The detailed configuration of the proposed convolutional autoencoder will be discussed in Section 3.3

3.2 *Long Short-Term Memory*

The generated IR thermal images during the printing process of large format additive manufacturing exhibit strong temporal correlations between each frame. These temporal features are crucial for accurately predicting future temperature cooling profiles. The Recurrent Neural Network (RNN) is well

known for the effectiveness in capturing temporal dependencies between time series samples. To overcome the exploding and vanishing gradient limitations in traditional RNNs, Long Short-Term Memory (LSTM) networks are proposed for modeling sequential data, providing enhanced prediction performance [43]. A LSTM model contains three gates at each time step t: the input gate i_t, which decides the potential new memories to retain; the forget gate f_t, which determines which old memories to discard; and the output gate o_t, which updates the proportion of short-term information to extract from the cell block. Let x_t represent the input latent representation at time t, h_t denote the hidden vector (short-term memory), c_t represent the long-term memory from the cell block, and W indicate the various weights in the process. The composite function is shown in Equation (3). At each time step, the model determines how much old information to retain, updates the new combined memories from the current cell block, and refreshes the short-term memory. After a sequence of iterations, the latest h_t serves as the final output of the entire LSTM model.

$$
\begin{aligned}
f_t &= \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf} \odot c_{t-1} + b_f) \\
i_t &= \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci} \odot c_{t-1} + b_i) \\
c_t &= f_t \odot c_{t-1} + i_t \odot tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \\
o_t &= \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co} \odot c_t + b_o) \\
h_t &= o_t \odot tanh(c_t)
\end{aligned}
\tag{3}
$$

where $\odot$ represents the Hadamard product. The activation function σ is typically set to the sigmoid function, which projects values into the interval $[0,1]$. Additionally, the hyperbolic tangent (tanh) activation function is commonly used to map memory values to the range $[-1,1]$. During training, the LSTM uses backpropagation through time to update the weights based on the loss calculated from the outputs.

3.3 *Deep CNN Combined LSTM Encoder-Decoder Network*

Traditionally in LFAM, the trend of layer temperature variation changes based on the geometric structure, including factors such as the number of adjacent beads and the width of the angles. This variation is fundamentally due to internal physical interactions within the geometries and influences from the external environment, which necessitate complex calculations and expertise from field specialists on a case-by-case basis. Therefore, the development of a data-driven generic temperature prediction model applicable to various printed objects is essential. As the temperature cooling process of a printed layer exhibits time series characteristics, a sequence-to-sequence prediction technique is employed to predict future temperature dynamics of arbitrary length. In the proposed model, a deep CNN architecture is utilized as a convolutional autoencoder to capture spatial features from thermal images, while a stacked LSTM encoder-decoder model is applied to the bottleneck layer to learn temporal features within the latent representation space. This combination of CNN and LSTM enhances computational efficiency and improves prediction accuracy.

In the proposed model, the temperature profile of all pixels in the thermal images will be predicted simultaneously, rather than selecting specific positions on the geometry. Each pixel value represents a corresponding temperature value. While monitoring the temperature of the newest printing path on the geometry is more important, this proposed approach enhances the flexibility of temperature prediction for future applications by considering the entire thermal image. The overview framework of the proposed model is illustrated in Figure 2.

In this framework, a sequence of thermal images $\{X_1, X_2, ..., X_T\}$ with a resolution 256×320 is provided as input, and a sequence of images $\{\hat{X}_{T+1}, \hat{X}_{T+2}, ..., \hat{X}_{T+K}\}$ of length K is predicted as the final output. In stage one, a deep convolutional autoencoder (CAE) model is trained offline to extract

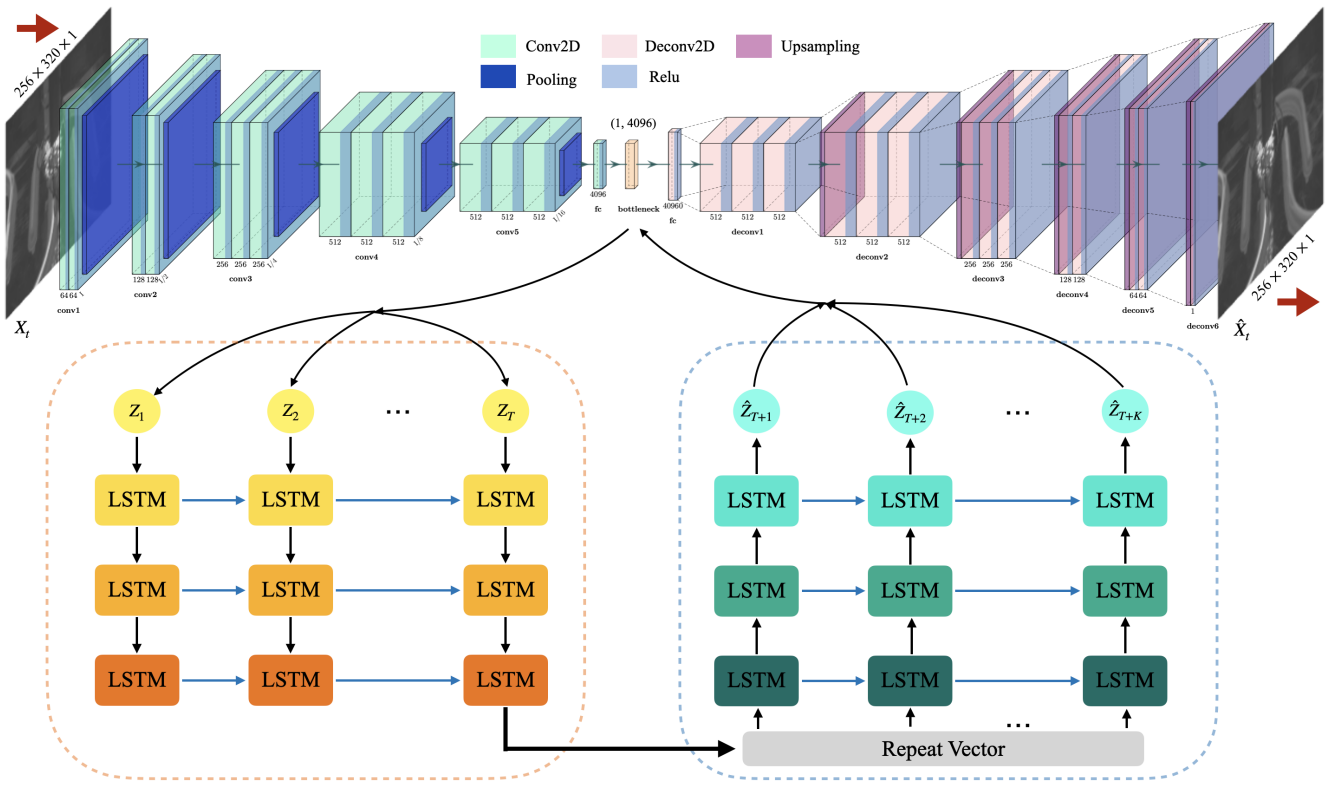


Figure 2: The Overview Framework of the CNN Combined LSTM Encoder-Decoder Network.

significant feature maps, thereby reducing the time required for spatial feature generation as well as the subsequent model training in stage two. During training, the input and output of the CNN autoencoder are identical thermal images, denoted as X_t and $\hat{X}_t$, $t = 1, 2, ..., T + K$. Denote H as the image height, W as the image width, and C as the number of color channels. As a thermal image can be considered as a grayscale image, where the color pixel values are replaced by corresponding temperature pixel values, the input tensor can be represented as (batch, H, W, C), specifically (batch, 256, 320, 1). In each convolutional block, there is a max-pooling layer after all convolution layers in the encoder, and while an upsampling layer precedes all convolutional layers in the decoder. Each convolution layer employs the ReLU activation function to introduce nonlinearity into the network. The output latent space of the fifth convolutional block in the encoder has dimensions of (batch, 8, 10, 512), which is large and inefficient for input into the subsequent stacked LSTM model. Therefore, the dimensionality is reduced by a fully connected dense layer to (batch, 4096). In the decoder, another fully connected layer with 40960 nodes is applied and reshaped back to (batch, 8, 10, 512) after the bottleneck, enabling the subsequent convolutional blocks to effectively reverse and mirror the encoder. The detailed configuration of the proposed convolutional autoencoder is shown in Table 3 of the Appendix.

In stage two, the stacked LSTM Encoder-Decoder model is integrated between the CNN encoder and decoder. The CNN encoder compresses each image X_t into a lower-dimensional latent representation Z_t, referred to as the bottleneck, with a shape of (1, 4096). Consequently, a sequence of 256 $\times$ 320 images with an original dimension of (num_frames, 81920) can be efficiently represented as (num_frames, 4096). Here, num_frames represents the number of images in the sequence and corresponds to the timeline. The stacked LSTM encoder-decoder then takes this bottleneck block as input, utilizing a specified number of lookback feature maps $\{Z_1, Z_2, ..., Z_T\}$ to predict a sequence of future feature maps $\{\hat{Z}_{T+1}, \hat{Z}_{T+2}, ..., \hat{Z}_{T+K}\}$. The input tensor has the shape (batch, T, 4096), while the out-

put tensor is represented as (batch, K, 4096). As illustrated in Figure 2, the LSTM model is structured as a stacked configuration with three encoding layers, mirrored by three decoding layers for symmetry and completeness. The three stacked LSTM layers contain 4096, 2048, and 1024 nodes, respectively. This architecture enables the generation of future prediction sequences of arbitrary length. Finally, the predicted future feature maps are reconstructed into thermal images by the CNN decoder, facilitating temperature profile prediction.

3.4 *Semantic Segmentation for Adaptive Geometry Extraction*

In the large format additive manufacturing, it is common to generate the layer temperature of a printed object by using an infrared (IR) camera positioned at a fixed observation angle. This setup typically captures not only the printed geometry but also the moving print head and noisy backgrounds. However, our primary focus is on the temperature of the geometry, particularly the temperature along the most recent print path. The process of capturing these temperatures often requires costly manual intervention due to challenges posed by noise in the images. This becomes even more challenging when attempting real-time capture and generating temperature data on geometries from thermal prediction images for future prints that have not yet been produced. Starting from the need for monitoring during the manufacturing process, it is essential to develop a model capable of automatically capturing temperatures on arbitrary geometries.

Inspired by similar applications in computer vision, we first employ semantic segmentation to predict a mask for the arbitrary geometry, identifying the geometric parts, and then set the temperature of all non-geometric areas to zero. This step not only helps isolate the geometry but also facilitates more accurate temperature capture along the printing path in the next step. Since the melted material stored in the print head is either hotter than or has a similar temperature to the most recently printed position, distinguishing the printing path from the print head in the original thermal image can be challenging. In the next step, we automatically generate the printing path by tracking the pixels with the highest temperatures; the highest temperature recorded in the processed geometry-only thermal image should correspond to the current printing position along the path.

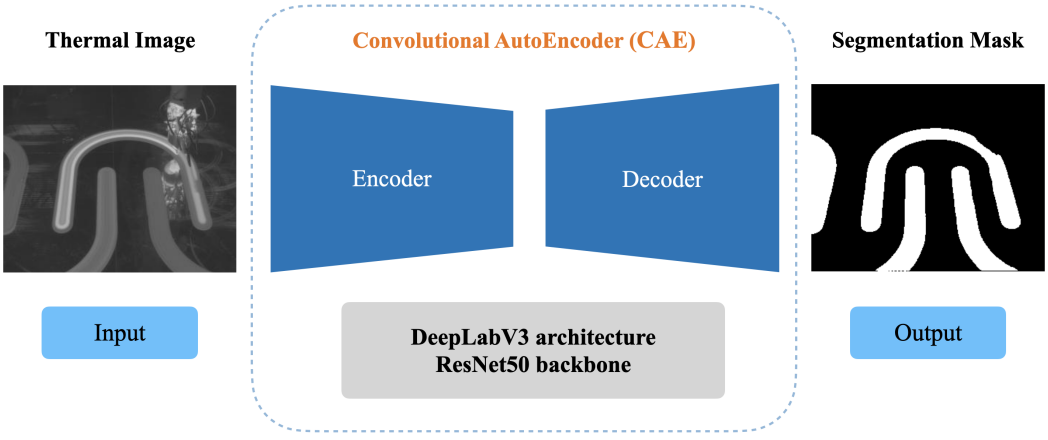


Figure 3: The Architecture of the Semantic Segmentation.

In this application, we have IR images with a size of (256, 320) depicting a part of the complex geometry constructed with multiple beads. The thermal image can be viewed as a grayscale image; instead of color channel values, each pixel represents the corresponding temperature at a specific position.

The highest temperature value is approximately 200, and the lowest value is around 25. As shown in Figure 3, the semantic segmentation model takes an arbitrary thermal image as its training input and the corresponding geometry mask as its output. Specifically, we fine-tune the DeepLabV3 architecture with a ResNet50 backbone [44], utilizing pretrained weights from ImageNet [45] to enhance mask prediction on our thermal images. During the training process, it is necessary to generate masks for the geometry in each image. To alleviate the heavy workload of manual labeling, the geometry masks are created using a combination of the Segment Anything (SAM) model [46] and the Matlab Image Labeler. The process of labeling the training data is illustrated in Figure 4. The SAM model can automatically generate multiple segmentation masks, from which the correct masks for the geometry will be selected. The purple mask in the figure represents a part of the extruder rather than the geometry; therefore, this section is ignored. However, it is clear that part of the geometry below the extruder is not masked. Consequently, if not all areas of the geometry are covered, a manually drawn mask for the missing sections will be combined with the generated masks to create an accurate overall mask for the geometry.

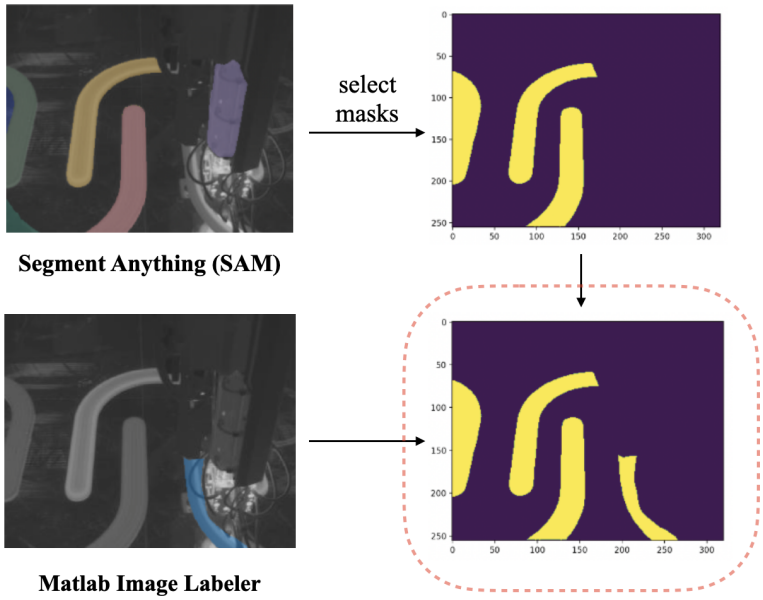


Figure 4: The process of applying the Segment Anything (SAM) model and the Matlab Image Labeler to generate mask.

With the preprocessed clean image, which contains only the geometry temperatures, the next step involves generating the printing path temperatures. Although the printed geometry will gradually and consistently move upward in the thermal image during the printing process due to the fixed IR camera settings, the path coordinate shifts between the current finished layer and the upcoming layer are minor and can be disregarded. Therefore, to generate the temperature along the path, we only need to track the highest temperature coordinates in a sequence of images from the current layer, which will allow us to obtain the entire printing path for the upcoming layer.

4 Case Study

The performance of the proposed 3D video-level temperature profile prediction model will be evaluated on two sequential datasets recorded during the LFAM process. By comparing its results with those

from stacked LSTM and ConvLSTM models, the proposed model is expected to demonstrate superior performance. Additionally, the application-oriented geometry extraction model will be fine-tuned to facilitate efficient thermal analysis in LFAM, enhancing its utility for real-time manufacturing processes.

4.1 *Data Collection*

The large format additive manufacturing machine utilizes 30% glass fiber reinforced PETG as its material, constructing products by melting and extruding out the material layer by layer along a guided printing path. The newly printed hot and soft layer should cool down sufficiently to become stiff enough to support the upcoming layer. Additionally, a compression wheel applies pressure to each freshly deposited bead of material, effectively fusing it to the layer below by collapsing any air pockets. This ensures a strong, seamless bond between layers, resulting in a high-quality, virtually void-free printed structure. The experimental setup is illustrated in Table 1. During the manufacturing process, the print head (the extruder) moves at a constant speed, and the layer deposition time represents the duration required to complete the printing of all geometries in the current layer before beginning the printing of the new layer. The final dimensions of the C-shaped table are a depth of 30 inches, a width of 14.5 inches, and a height of 25 inches. The tree totem has a final diameter of 13 inches and a height of 76 inches.

Table 1: Experimental setting

Printing conditions	Setting
Deposition Temperature, [°C]	190
Bed Temperature, [°C]	22
Ambient Temperature, [°C]	22
Bead height, [mm]	5.08
Bead width, [mm]	20.32
Speed, [in/min]	535
Layer deposition time (Table), [s]	400
Layer deposition time (Tree totem), [s]	236

Two thermal video datasets are collected by a fixed-angle IR camera during the printing of C-shaped tables and tree-totem geometries, named "Table" and "Tree totem", respectively. Both thermal video include either part of or the entire printed object, as well as the background printing bed and the moving extruder. Each frame in the thermal video has a resolution of 256×320. The frames collected from each thermal video have a frequency of 1 frame every 2 seconds. The C-shaped table takes approximately 200 frames (400 seconds) per layer, while the tree totem typically has a layer time of around 115 frames (230 seconds).

The thermal frames of the "Tree totem" dataset contain four identical single bead objects, printed at the same layer level, one after another. A single tree-totem features curved, sharp, and wide corners. Each frame of the "Table" dataset also includes multiple printed objects, with only one exhibiting a complete shape. Specifically, the C-shaped table has a double bead structure, which presents a more complex temperature cooling dynamic compared to the single bead geometry. Furthermore, the corner shape of the C-shaped table changes over time, indicating that this geometry is not homogeneous. In Figure 5, the objects are printed following the numerical symbols 1-4 in the left image, with the deposition direction labeled along the tool path. Additionally, the red rectangles in the right image illustrate the geometric changes at the corner positions after 12 layers of printing have been completed.

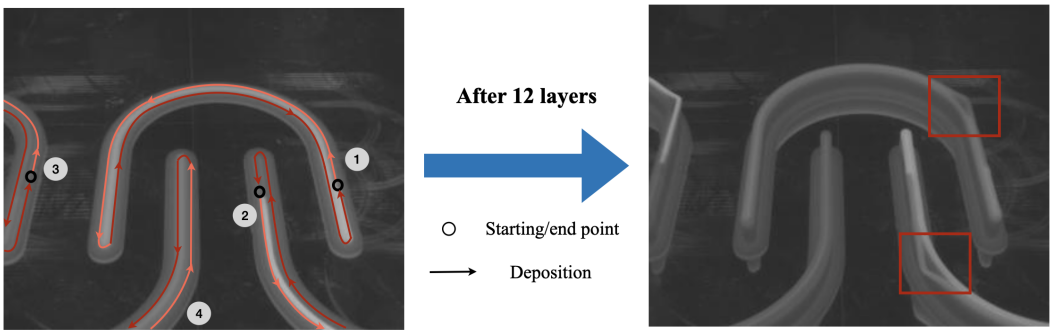


Figure 5: C-shaped table with double bead tool path. The left image shows the deposition direction and order, indicated by the numerical symbols 1-4. The right image illustrates the geometric changes at the corner positions as the number of printed layers increases.

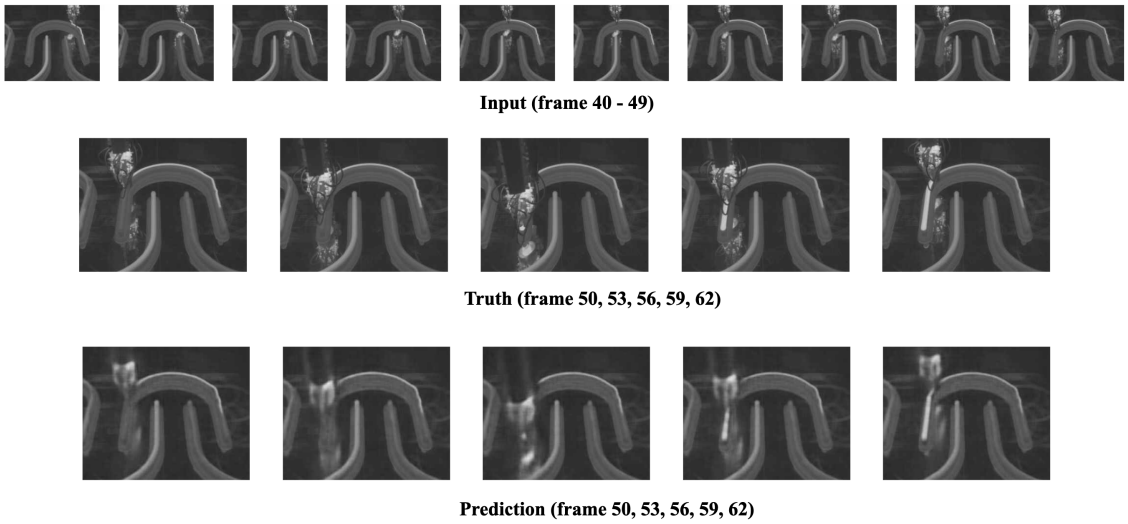
4.2 *Model Training*

In this case study, all thermal prediction models are trained on a NVIDIA A100 GPU. The "Table" dataset contains 2600 frames, of which 2415 frames are used for training. The "Tree Totem" dataset consists of 6000 frames, with 5600 frames used for training. All input images are normalized before the training process begins. In the proposed model, a deep CNN autoencoder is trained offline to reduce input dimensionality and extract significant spatial features. The bottleneck layer of the autoencoder is then used as the input for a stacked LSTM model, which incorporates temporal information for sequential data prediction. The detailed architecture of the deep CNN autoencoder is provided in Table 3. The mean squared error (MSE) is used as the loss function, and the Adam optimizer is applied with a reduced learning rate of $[1e^{-4}, 1e^{-5}, 1e^{-6}, 1e^{-7}]$. The model converges after 150 epochs with a batch size of 16. Next, a stacked LSTM encoder-decoder network is constructed to predict sequential future frames of arbitrary length. The encoder consists of three LSTM layers with 4096, 2048, and 1024 nodes, respectively, and is trained using input tensors of shape (batch, T, 4096), where T represents the number of lookback frames. Each layer includes a ReLU activation function to introduce nonlinearity into the network. The decoder mirrors the encoder to predict sequences of arbitrary length K. Finally, the predicted latent representations are reconstructed by the CNN decoder to produce thermal frames.

To evaluate the performance of the proposed model, we compare it against a stacked LSTM-only model as the baseline, given its established ability to predict time series data. Due to the limitations in extracting spatial features from sequential thermal images, the prediction performance tends to be conservative. Hence, we employ ConvLSTM [25], one of the most popular techniques for handling both temporal and spatial features in sequential frame prediction, as a benchmark for comparison. The architecture of the stacked LSTM-only model is configured identically to the LSTM component of the proposed model for comparison purposes. Instead of latent representations, this model takes a sequence of flattened original thermal images as input, with a tensor shape of (batch, T, 81920). The output tensor, with a shape of (batch, K, 81920), will be reshaped into the format (batch, K, 256, 320, 1). In this paper, the ConvLSTM model consists of three convlstm2D layers, each with 128 filters, followed by a batch normalization layer. The kernel sizes for these layers are (3, 3), (3, 3), and (1, 1), respectively. A conv3D layer with a single filter is then added to reconstruct the one-channel input image. The activation function used for all convolutional layers is ReLU. The tensor input for the ConvLSTM model has the shape (batch, T, 256, 320, 1). During the training process of the three sequential models, the Adam optimizer is applied with the mean absolute error (MAE) loss function and a reduced learning rate of $[1e^{-5}, 1e^{-6}]$. The computation time is limited to 5 hours for the "Table" dataset and 12 hours for the

"Tree totem" dataset. Each model is trained until convergence or the time limit is reached. Three combinations of (T, K) are tested on both datasets for each model: (2, 2), (10, 5), and (10, 15). Due to GPU memory limitations, the constructed ConvLSTM model cannot support long-term predictions with our current hardware. As a result, the lookback frame for this model is set to two, and two future frames are predicted at each step.

(a) Thermal prediction for the "Table" geometry.



(b) Thermal prediction for the "Tree totem" geometry.

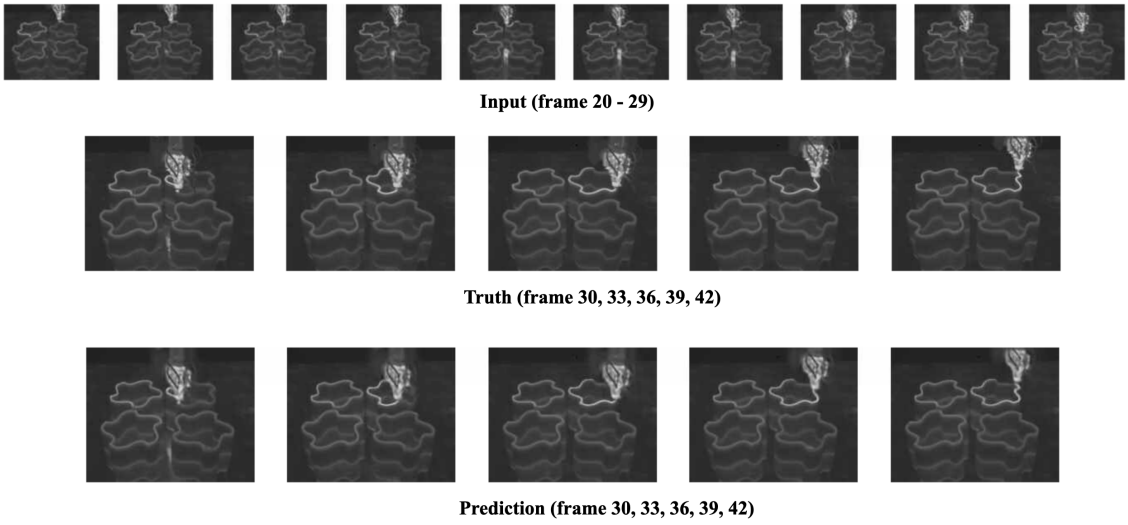


Figure 6: The comparison between truth and prediction frames obtained from the proposed CNN+LSTM model. For both dataset "Table" (a) and "Tree totme" (b), the lookback frames are 10, and the future predicted frames are consecutive 15 frames (every third frames is shown in the image). The consecutive frames are captured with 2s time gap.

Figure 6 illustrates the experimental results of the proposed model when predicting 15 future frames based on the "Table" and "Tree totem" datasets. The input frames consist of a sequence of lookback frames, and the ground truth frames represent the actual frames that follow the input sequence. The predicted frames are compared to the ground truth frames one by one, displayed vertically. To provide

a clearer view, only every third frame from the sequence is shown, rather than displaying all frames. In the images shown in (a), the extruder is printing from right to left along the top printing path in the input frames. In the predicted future frames, it completes the remaining right-to-left movement and starts a new path from left to right, positioned immediately adjacent to the previous one. It is evident that the prediction for the C-shaped table appears relatively blurred around the extruder. Fortunately, this blurriness does not significantly impact our objectives, as the focus is primarily on the temperature values of the printed geometry rather than the boundary of the printing head. It is important to note that the goals of prediction in this context differ from those in general computer vision applications. In computer vision, generating increasingly clear images is crucial, particularly for defining object boundaries. Typically, images consist of three color channels, and the precise value of a single pixel is less critical. However, in large format additive manufacturing, the emphasis is on monitoring temperature values at each geometry layer. Since each pixel represents the actual temperature at a specific position on the geometry, the accuracy of these pixel values is more relevant to our application. In addition, the prediction results based on the "Tree Totem" dataset are shown in (b). In these images, the extruder is printing a tree totem in the top-right section along a counterclockwise single-bead path. The prediction frames are clear, with only minor blurriness observed in this dataset. A more detailed comparison of the models, evaluation metrics, and prediction analysis is discussed in the following section.

4.3 *Model Validation*

4.3.1 Performance evaluation metrics

In this paper, the model's performance is evaluated and compared using three metrics: Mean Squared Error (MSE), Structural Similarity Index Measure (SSIM) [47], and Peak Signal-to-Noise Ratio (PSNR). The MSE quantifies the average of squared pixel differences between a ground truth image and a predicted image, as shown in the equation below:

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}(\hat{y}(i,j) - y(i,j))^2 \tag{4}$$

where $\hat{y}(i,j)$ and $y(i,j)$ represent the pixel values at the coordinates (i,j) of the predicted frame and the corresponding ground truth frame, respectively. The grayscale frame has a resolution of $m \times n$. A lower MSE value indicates better prediction performance. In addition, the SSIM is a popular metric utilized for image similarity comparison, as it accounts for changes in structural information, luminance, and contrast. It is calculated using Equation (5):

$$SSIM(\hat{y},y) = \frac{(2\mu_{\hat{y}}\mu_y + c_1)(2\sigma_{\hat{y}y} + c_2)}{(\mu_{\hat{y}}^2 + \mu_y^2 + c_1)(\sigma_{\hat{y}}^2 + \sigma_y^2 + c_2)} \tag{5}$$

where $\mu_{\hat{y}}$ and μ_y represent the average luminance of the predicted image and the ground truth image, respectively. The variances $\sigma_{\hat{y}}$ and σ_y represent the contrast, while the covariance $\sigma_{\hat{y}y}$ measures the structural similarity. Denote $L = 255$, which is the range of pixel values in the image, and $c_1 = (K_1 L)^2$ and $c_2 = (K_2 L)^2$ are small constants used to prevent division by zero, where K_1 and K_2 are typically set to 0.01 and 0.03, respectively. The predicted image is more similar to the corresponding ground truth image when the SSIM value is closer to 1 (100%). Another widely used metric to evaluate the quality of the predicted image, PSNR, measures the ratio between the maximum possible pixel value (L) of the image and the image's square root MSE value:

$$PSNR = 20\log_{10}(\frac{L}{\sqrt{MSE}}) \tag{6}$$

where higher PSNR values indicates better image quality and less distortion. Typically, values higher than 30 means acceptable and good image quality.

4.3.2 *Results discussion*

The performance comparison based on testing data between the stacked LSTM, ConvLSTM, and the proposed hybrid CNN-LSTM model, using two different thermal image datasets recorded during the large format additive manufacturing process, is presented in Table 2. To evaluate the proposed model's potential for predicting as many future frames as possible while maintaining acceptable accuracy, three sets of experiments were conducted for each dataset, predicting an increasing number of future frames: 2, 5, and 15. Each comparison set uses the same batch size to ensure a fair evaluation, as smaller batch sizes typically lead to more accurate but slower predictions, while larger batch sizes result in less accurate but faster predictions. In the comparison table, missing values indicate that the corresponding model can not be trained with the current settings due to limitations of available GPU memory. Therefore, the ConvLSTM model is trained and evaluated only for predicting future frames of two. Due to the larger training dataset for the "Tree Totem," the stacked LSTM model encountered an out-of-GPU-memory error during training. Compared to the stacked LSTM and ConvLSTM models, the proposed hybrid CNN-LSTM model demonstrates the ability to save computational costs on both CPU and GPU memory.

Table 2: The model performance evaluation metric based on testing data.

Data	Lookback#	Future#	Model	Avg. MSE ↓	Avg. SSIM ↑	Avg. PSNR ↑	Training time (s)
Table	2	2	ConvLSTM	170.65	90.96%	28.66	16600
			LSTM	250.55	58.31%	24.48	3154
			CNN+LSTM	46.84	91.65%	32.83	2687
	10	5	ConvLSTM	-	-	-	-
			LSTM	183.02	76.25%	26.22	7791
			CNN+LSTM	51.93	91.61%	32.56	2366
	10	15	ConvLSTM	-	-	-	-
			LSTM	345.24	48.30%	23.06	5529
			CNN+LSTM	61.47	91.40%	31.95	2577
Tree totem	2	2	ConvLSTM	213.44	91.78%	25.04	41280
			LSTM	120.62	92.05%	27.44	2608
			CNN+LSTM	46.56	93.83%	31.92	2284
	10	5	ConvLSTM	-	-	-	-
			LSTM	162.44	90.25%	26.11	10806
			CNN+LSTM	39.56	94.29%	32.54	1574
	10	15	ConvLSTM	-	-	-	-
			LSTM	-	-	-	-
			CNN+LSTM	44.87	94.04%	32.13	3315

* Note: - indicates that corresponding model can not be trained with the current settings due to limitations of available GPU memory.

Since the objective of applying the proposed model in large format additive manufacturing is to achieve real-time temperature prediction, an optimal model should effectively balance lower computation

time, higher prediction accuracy, and the ability to predict over longer time frames. In the comparison table, the average values of MSE, SSIM, and PSNR are calculated across 175 sequential frames of prediction. For both datasets, the proposed model achieves the lowest MSE value, the highest SSIM value, and the highest PSNR value, indicating superior prediction accuracy. Both the stacked LSTM model and the ConvLSTM model have average MSE values exceeding hundreds across all experiments, which are considered excessively large. In contrast, even when predicting 15 future frames, the proposed model achieves acceptable average MSE values of 61.47 for the "Table" dataset and 44.87 for the "Tree totem" dataset. Moreover, the proposed model achieves an average SSIM value exceeding 90% and an average PSNR value exceeding 30 across all experiments, confirming that it provides sufficiently high image quality for the predicted frames. However, the average value can sometimes be misleading; therefore, to compare and explore the details of prediction accuracy, the MSE comparison of each frame based on the three different models will be analyzed later.

In the experiments, a very lenient time limit of training is set as 5 hours for the "Table" and 12 hours for the "Tree totem". As the deep CNN autoencoder in the proposed model is trained offline, its training time is excluded from the training times recorded in the comparison table. In comparison, the proposed CNN-LSTM model requires the least training time, being 6 to 18 times faster than the ConvLSTM model. Typically, the ConvLSTM is recognized as an accurate model for predicting sequential images; however, it is extremely time-consuming to achieve accurate results. Therefore, under the time constraints of this case study, the ConvLSTM model does not perform well in terms of accuracy, even with very long training times. Due to the limitations of the stacked LSTM model in addressing spatial features in images, it cannot guarantee accurate predictions. Additionally, because of the high dimensionality of the inputs, the stacked LSTM model must be trained with a large number of parameters, making it more time-consuming than the proposed model. Hence, the proposed model is confirmed to have superior performance, successfully meeting the objective of achieving real-time temperature prediction in large format additive manufacturing.

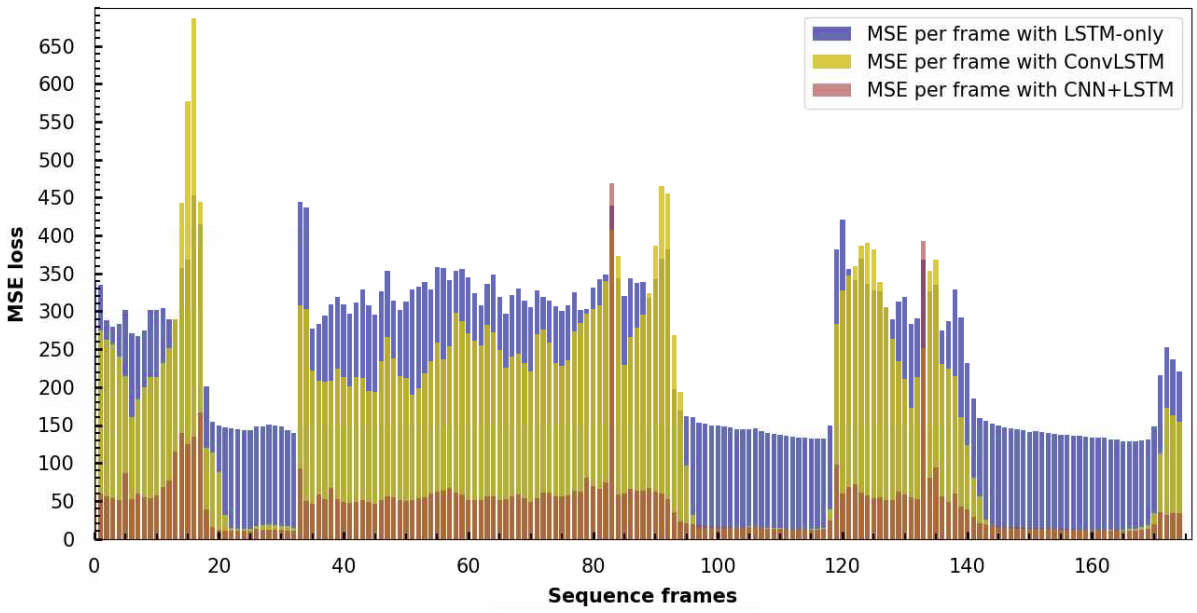


Figure 7: MSE comparison between CNN+LSTM, ConvLSTM and LSTM base on the "Table" data.

Furthermore, all models tend to perform better on the "Tree totem" dataset than on the "Table" dataset. In addition to the difference in the number of training samples, another possible reason is

explored through the comparison of each frame's MSE on testing data in Figures 7, 9. The Figure 7 illustrates the experiment of predicting two future frames based on the "Table" dataset. The x-axis represents the sequential frames from 0 to 174, and the y-axis shows the MSE value for each predicted frame, comparing different prediction models. In most frames, the MSE is highest with the stacked LSTM model, followed by the ConvLSTM model, and lowest with the CNN-LSTM model. Frames in the ranges of 20 to 32, 95 to 117, and 144 to 169 exhibit significantly lower MSE than other predicted frames because the extruder moves out of the frame. The ConvLSTM model demonstrates similar predictive performance to the proposed model during these periods, with MSE values of approximately 14 and 11, respectively. A sample frame during this period, frame 25, is shown in Figure 8. In these frames, there is no movement, and only the luminance of the geometry is visible, which reflects the temperature cooling dynamics of the geometry. However, the prediction model should be capable of performing well even when the extruder is moving within the frame, which presents a much more challenging scenario. In these frames, the proposed model achieves an MSE of around 50 for each, which is significantly lower than that of the other two models, except for two outlier frames, 83 and 133. It is evident that all three models perform poorly, having around 400 mse value, on these two frames. Comparisons between the predicted and corresponding ground truth frames are shown in Figure 8. In both cases, the predicted frames lag behind the ground truth by one frame, coinciding with the moment when the extruder moves to a new deposition start position. The predictions for these two frames can be considered outliers and do not significantly affect the overall accuracy of predicting the temperature cooling dynamics across the entire layer.

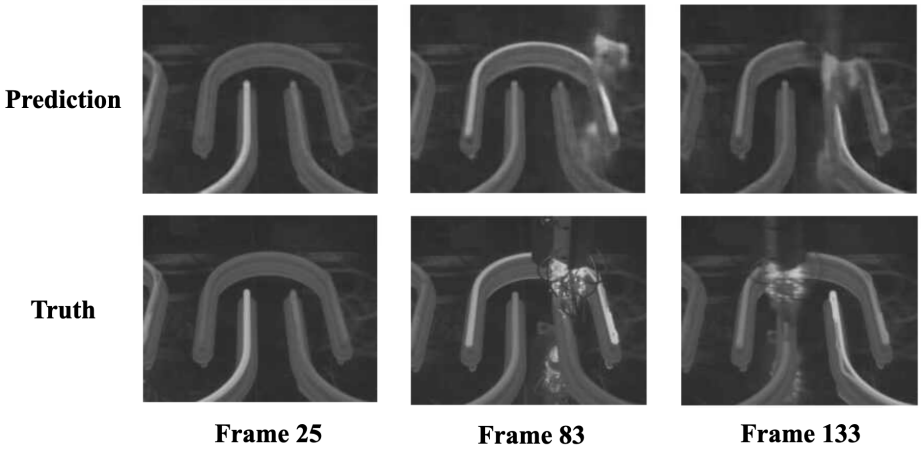


Figure 8: Image comparison between prediction (CNN+LSTM) and truth on frames with relative large MSE loss.

For the "Tree totem" dataset, the comparison of MSE values for each frame, based on the setting of predicting two future frames, is shown in Figure 9. In contrast to the "Table" dataset, the predictions on the "Tree totem" dataset are relatively consistent across sequential frames for all models. As shown in the figure, the proposed CNN-LSTM model consistently achieves the lowest MSE values, around 40, for nearly all frames. In this dataset, the extruder is constantly moving within the frames, which may explain why the prediction results remain more stable across frames. Additionally, with more frames containing the moving extruder used in training, the overall MSE values are lower for this dataset. A few frames exhibit anomaly higher MSE values, which occur when the extruder moves to a new deposition start position. However, in these outlier frames, there is no lag issue; the predictions are simply a little more blurred than other frames.

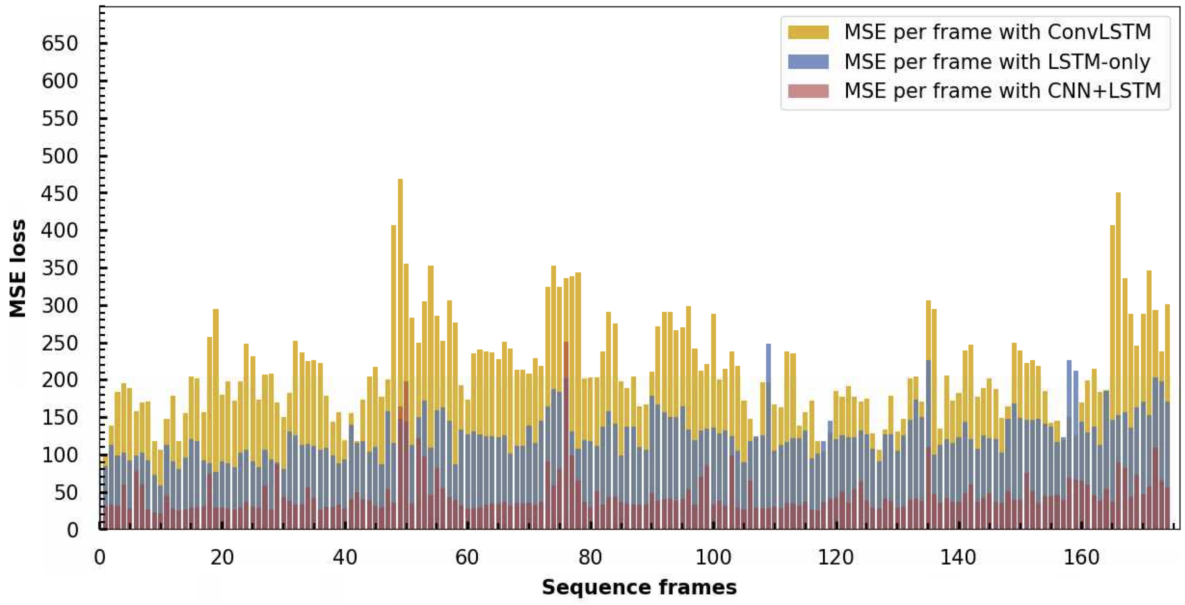


Figure 9: MSE comparison between CNN+LSTM, ConvLSTM and LSTM base on the "Tree totem" data.

4.4 *Adaptive Geometry Extraction*

Model validation demonstrates that the proposed hybrid CNN-LSTM model reliably predicts thermal behavior on the monitored layer, a promising step toward effective in-situ monitoring. However, an important remaining challenge for real-time thermal analysis is the efficient, automatic extraction of temperature data from thermal images, which are often complicated by noisy gantry backgrounds and the movement of the printing head. In LFAM, the primary focus for temperature monitoring and prediction is on the geometry's temperature, particularly along the most recently printed path. By applying the semantic segmentation model described in Section 3.4, we fine-tune this model on 500 sequential samples from the "Table" dataset. In this paper, we define the non-geometry components, including the extruder and background, as a single class: background. Therefore, the segmentation task is framed as a binary classification problem with two classes: geometry and background. The model employs cross-entropy as the loss function and is trained using the Adam optimizer with a learning rate of 0.0001. The model converges after 200 epochs with a batch size of 16. Pixel accuracy, defined as the ratio of correctly predicted pixels to the total number of pixels in the image, is used as the evaluation metric, achieving 98.9% accuracy on 100 validation samples. Finally, accurate mask predictions are generated for an additional 2400 sequential frames. The performance of the semantic segmentation model is anticipated to be improved and generalized across thermal images featuring various geometries with sufficient training. This enhancement would enable the model to effectively capture arbitrary printed objects within the images, thereby benefiting the subsequent temperature monitoring and analysis processes.

When future thermal images are predicted by the proposed temperature prediction model, the predicted frames can be combined with their corresponding predicted segmentation masks. This approach generates a clean image where all geometry regions retain their predicted temperature values, while all background regions, as defined, are assigned a value of zero. Figure 10 shows the preprocessed clean image for the "Table" dataset. Moreover, based on the thermal images predicted from the experiments of forecasting 15 future frames using the proposed hybrid CNN-LSTM model, the temperature cooling curve at a specific position located on the printing path is illustrated in Figure 11. The x-axis repre-

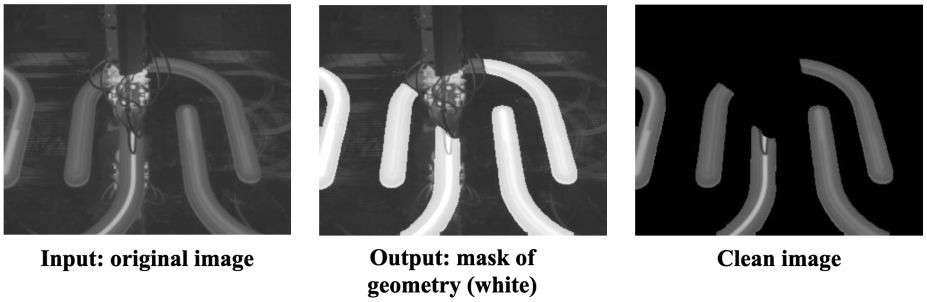


Figure 10: The comparison of original image, predicted mask and clean image.

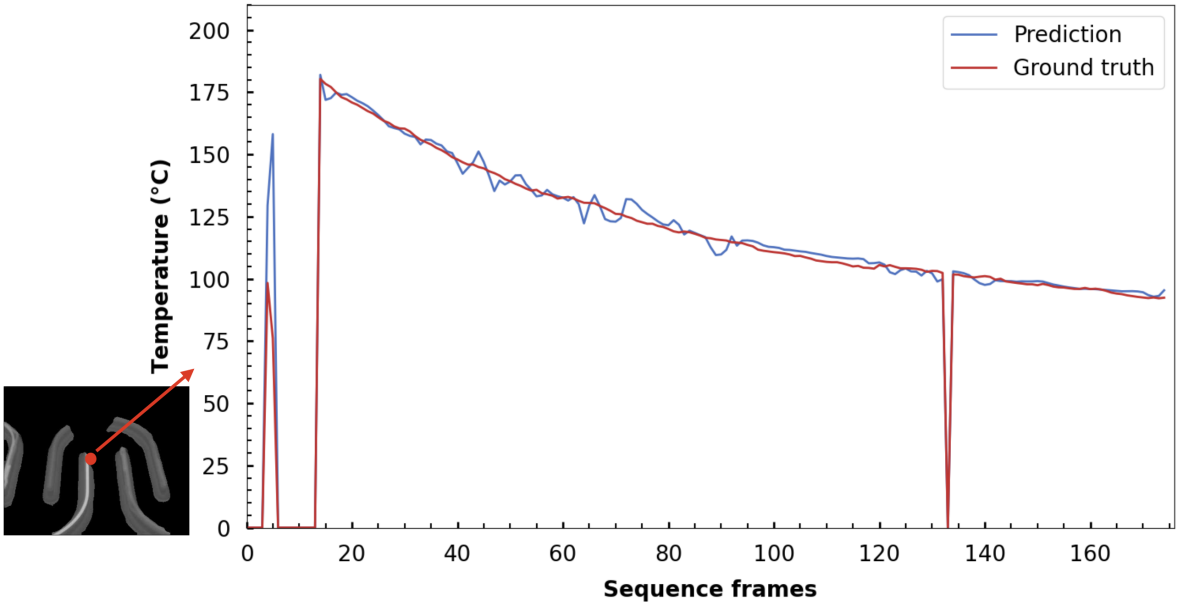


Figure 11: The temperature cooling curve for position coordinate (123, 143) with the CNN+LSTM model prediction based on the "Table" data.

sents the sequence of frames, which can be interpreted as the timeline, while the y-axis denotes the temperature. In the figure, the predicted temperature at this position closely matches the ground truth, demonstrating that the proposed model's performance is acceptable.

5 Conclusion

In LFAM, maintaining appropriate surface layer temperatures is essential to avoid manufacturing failures and defects in the final product. This necessity underscores the importance of developing in-situ temperature monitoring models capable of predicting thermal behavior for upcoming layers and efficiently extracting temperature data at relevant positions from noisy thermal images. In response, this paper proposes a data-driven hybrid CNN-LSTM model for accurate, generic, and efficient prediction of future thermal images in products with complex geometries. This model integrates an offline deep CNN autoencoder, designed to capture essential spatial features within thermal images, with a real-time stacked LSTM encoder-decoder to bridge the CNN autoencoder, incorporating temporal information for arbitrary-length future thermal image predictions. By balancing time efficiency and prediction accuracy,

this model reduces training time by more than half while maintaining high prediction accuracy. Furthermore, a semantic segmentation model is fine-tuned to address image noise from moving extruders and gantry backgrounds, isolating relevant printing geometry and enhancing temperature tracking reliability in real-time for subsequent thermal analysis.

Future work will focus on extending the current approach to support longer-term thermal predictions and addressing blur issues caused by moving elements. Additionally, a digital-twin alike thermal image prediction model could be developed to forecast thermal dynamics based on current conditions while simulating potential temperature profile variations resulting from adjustments in printing parameters. This extension may include the integration of interpolation blocks to allow the model to generate predictive profiles for different parameter scenarios, further enhancing its adaptability and utility in in-situ monitoring and control.

6 Acknowledgement

This research was partially supported by National Science Foundation Grant CMMI-1922739 and a US Department of Energy High Performance Computing for Energy Innovation (HPC4EI) grant and by the Vehicle Technologies Office in the U.S. Department of Energy, Office of Energy Efficiency and Renewable Energy, Industrial Technologies Program, under contract DE-AC05-00OR22725 with UT-Battelle, LLC.

References

[1] S. Fathizadan, F. Ju, F. Wang, K. Rowe, and N. Hofmann, "Dynamic material deposition control for large-scale additive manufacturing," *IISE Transactions*, vol. 54, no. 9, pp. 817–831, 2022.

[2] A. A. Hassen, R. Springfield, J. Lindahl, B. Post, L. Love, C. Duty, U. Vaidya, R. B. Pipes, and V. Kunc, "The durability of large-scale additive manufacturing composite molds," *CAMX*, vol. 2016, no. 26, pp. 26–29, 2016.

[3] F. Wang, S. Fathizadan, F. Ju, K. Rowe, and N. Hofmann, "Print surface thermal modeling and layer time control for large-scale additive manufacturing," *IEEE Transactions on automation science and engineering*, vol. 18, no. 1, pp. 244–254, 2020.

[4] V. Kishore, A. Nycz, J. Lindahl, C. Duty, C. Carnal, and V. Kunc, "Effect of infrared preheating on the mechanical properties of large format 3d printed parts," Oak Ridge National Laboratory (ORNL), Oak Ridge, TN (United States), Tech. Rep., 2019.

[5] S. Fathizadan, F. Ju, K. Rowe, A. Fiechter, and N. Hofmann, "A novel real-time thermal analysis and layer time control framework for large-scale additive manufacturing," *Journal of Manufacturing Science and Engineering*, vol. 143, no. 1, p. 011009, 2021.

[6] Y. Zhang and Y. Chou, "Three-dimensional finite element analysis simulations of the fused deposition modelling process," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 220, no. 10, pp. 1663–1671, 2006.

[7] B. Brenken, E. Barocio, A. Favaloro, V. Kunc, and R. B. Pipes, "Development and validation of extrusion deposition additive manufacturing process simulations," *Additive manufacturing*, vol. 25, pp. 218–226, 2019.

[8] E. Jo, L. Liu, F. Ju, D. Hoskins, D. Pokkalla, V. Kunc, U. Vaidya, and P. Kim, "The design of layer time optimization in large scale additive manufacturing with fiber reinforced polymer composites," Oak Ridge National Lab.(ORNL), Oak Ridge, TN (United States), Tech. Rep., 2022.

[9] L. Liu, E. Jo, D. Hoskins, U. Vaidya, S. Ozcan, F. Ju, and S. Kim, "Layer time optimization in large scale additive manufacturing via a reduced physics-based model," *Additive Manufacturing*, vol. 72, p. 103597, 2023.

[10] T. Wuest, D. Weimer, C. Irgens, and K.-D. Thoben, "Machine learning in manufacturing: advantages, challenges, and applications," *Production & Manufacturing Research*, vol. 4, no. 1, pp. 23–45, 2016.

[11] A. Paul, M. Mozaffar, Z. Yang, W.-k. Liao, A. Choudhary, J. Cao, and A. Agrawal, "A real-time iterative machine learning approach for temperature profile prediction in additive manufacturing processes," in *2019 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*. IEEE, 2019, pp. 541–550.

[12] F. Wang, F. Ju, K. Rowe, and N. Hofmann, "Real-time control for large scale additive manufacturing using thermal images," in *2019 IEEE 15th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2019, pp. 36–41.

[13] Y. Jiang, H. Dong, and A. El Saddik, "Baidu meizu deep learning competition: Arithmetic operation recognition using end-to-end learning ocr technologies," *IEEE Access*, vol. 6, pp. 60 128–60 136, 2018.

[14] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[15] N. Murray and F. Perronnin, "Generalized max pooling," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 2473–2480.

[16] Y. Wang, J. Huang, Y. Wang, S. Feng, T. Peng, H. Yang, and J. Zou, "A cnn-based adaptive surface monitoring system for fused deposition modeling," *IEEE/ASME Transactions on Mechatronics*, vol. 25, no. 5, pp. 2287–2296, 2020.

[17] S. Fathizadan, F. Ju, and Y. Lu, "Deep representation learning for process variation management in laser powder bed fusion," *Additive Manufacturing*, vol. 42, p. 101961, 2021.

[18] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "Lstm: A search space odyssey," *IEEE transactions on neural networks and learning systems*, vol. 28, no. 10, pp. 2222–2232, 2016.

[19] P. K. Nalajam and R. Varadarajan, "A hybrid deep learning model for layer-wise melt pool temperature forecasting in wire-arc additive manufacturing process," *IEEE Access*, vol. 9, pp. 100 652–100 664, 2021.

[20] A. Agga, A. Abbou, M. Labbadi, Y. El Houm, and I. H. O. Ali, "Cnn-lstm: An efficient hybrid deep learning architecture for predicting short-term photovoltaic power production," *Electric Power Systems Research*, vol. 208, p. 107908, 2022.

[21] W. S. Ahmed *et al.*, "The impact of filter size and number of filters on classification accuracy in cnn," in *2020 International conference on computer science and software engineering (CSASE)*. IEEE, 2020, pp. 88–93.

[22] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[23] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[24] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*. Springer, 2015, pp. 234–241.

[25] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo, "Convolutional lstm network: A machine learning approach for precipitation nowcasting," *Advances in neural information processing systems*, vol. 28, 2015.

[26] W. Lotter, G. Kreiman, and D. Cox, "Deep predictive coding networks for video prediction and unsupervised learning," *arXiv preprint arXiv:1605.08104*, 2016.

[27] X. Liang, L. Lee, W. Dai, and E. P. Xing, "Dual motion gan for future-flow embedded video prediction," in *proceedings of the IEEE international conference on computer vision*, 2017, pp. 1744–1752.

[28] R. Villegas, J. Yang, Y. Zou, S. Sohn, X. Lin, and H. Lee, "Learning to generate long-term future via hierarchical prediction," in *international conference on machine learning*. PMLR, 2017, pp. 3560–3569.

[29] W. Yu, Y. Lu, S. Easterbrook, and S. Fidler, "Efficient and information-preserving future frame prediction and beyond," in *International Conference on Learning Representations*, 2020.

[30] Z. Gao, C. Tan, L. Wu, and S. Z. Li, "Simvp: Simpler yet better video prediction," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 3170–3180.

[31] X. Ye and G.-A. Bilodeau, "Vptr: Efficient transformers for video prediction," in *2022 26th International Conference on Pattern Recognition (ICPR)*. IEEE, 2022, pp. 3492–3499.

[32] J. Ho, T. Salimans, A. Gritsenko, W. Chan, M. Norouzi, and D. J. Fleet, "Video diffusion models," *Advances in Neural Information Processing Systems*, vol. 35, pp. 8633–8646, 2022.

[33] S. Yu, K. Sohn, S. Kim, and J. Shin, "Video probabilistic diffusion models in projected latent space," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 18 456–18 466.

[34] Z. Jin, Z. Zhang, and G. X. Gu, "Autonomous in-situ correction of fused deposition modeling printers using computer vision and deep learning," *Manufacturing Letters*, vol. 22, pp. 11–15, 2019.

[35] S. A. Shevchik, G. Masinelli, C. Kenel, C. Leinenbach, and K. Wasmer, "Deep learning for in situ and real-time quality monitoring in additive manufacturing using acoustic emission," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 9, pp. 5194–5203, 2019.

[36] Z. Chen, P. Santhakumar, K. Granland, C. Troeung, C. Chen, and Y. Tang, "Predicting future warping from the first layer: A vision-based deep learning method for 3d printing monitoring," in *2023 IEEE 19th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2023, pp. 1–6.

[37] X. Li, M. Zhang, M. Zhou, J. Wang, W. Zhu, C. Wu, and X. Zhang, "Qualify assessment for extrusion-based additive manufacturing with 3d scan and machine learning," *Journal of Manufacturing Processes*, vol. 90, pp. 274–285, 2023.

[38] S. Guo, R. Dai, H. Sun, and Q. Nian, "pts-lstm: Temperature prediction for fused filament fabrication using thermal image time series," *Journal of Manufacturing Processes*, vol. 106, pp. 316–327, 2023.

[39] W. Ren, G. Wen, Z. Zhang, and J. Mazumder, "Quality monitoring in additive manufacturing using emission spectroscopy and unsupervised deep learning," *Materials and Manufacturing Processes*, vol. 37, no. 11, pp. 1339–1346, 2022.

[40] F. Ogoke, P. Pak, A. Myers, G. Quirarte, J. Beuth, J. Malen, and A. B. Farimani, "Deep learning for melt pool depth contour prediction from surface thermal images via vision transformers," *Additive Manufacturing Letters*, vol. 11, p. 100243, 2024.

[41] E. Yangue, Z. Ye, C. Kan, and C. Liu, "Integrated deep learning-based online layer-wise surface prediction of additive manufacturing," *Manufacturing Letters*, vol. 35, pp. 760–769, 2023.

[42] J. Masci, U. Meier, D. Cireşan, and J. Schmidhuber, "Stacked convolutional auto-encoders for hierarchical feature extraction," in *Artificial Neural Networks and Machine Learning–ICANN 2011: 21st International Conference on Artificial Neural Networks, Espoo, Finland, June 14-17, 2011, Proceedings, Part I 21*. Springer, 2011, pp. 52–59.

[43] A. Graves, "Generating sequences with recurrent neural networks," *arXiv preprint arXiv:1308.0850*, 2013.

[44] L.-C. Chen, "Rethinking atrous convolution for semantic image segmentation," *arXiv preprint arXiv:1706.05587*, 2017.

[45] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[46] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo *et al.*, "Segment anything," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 4015–4026.

[47] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

Appendix

Table 3: The detailed architecture of the deep convolutional autoencoder (CAE).

Type	Layer name	Operation	Number of filters	Filter size	Padding	Nodes
Encoder	conv2D_e1	Convolution + ReLU	64	(3,3)	same	
	conv2D_e2	Convolution + ReLU	64	(3,3)	same	
	maxpool_e3	Maxpooling	1	(2,2)		
	conv2D_e4	Convolution + ReLU	128	(3,3)	same	
	conv2D_e5	Convolution + ReLU	128	(3,3)	same	
	maxpool_e6	Maxpooling	1	(2,2)		
	conv2D_e7	Convolution + ReLU	256	(3,3)	same	
	conv2D_e8	Convolution + ReLU	256	(3,3)	same	
	conv2D_e9	Convolution + ReLU	256	(3,3)	same	
	maxpool_e10	Maxpooling	1	(2,2)		
	conv2D_e11	Convolution + ReLU	512	(3,3)	same	
	conv2D_e12	Convolution + ReLU	512	(3,3)	same	
	conv2D_e13	Convolution + ReLU	512	(3,3)	same	
	maxpool_e14	Maxpooling	1	(2,2)		
	conv2D_e15	Convolution + ReLU	512	(3,3)	same	
	conv2D_e16	Convolution + ReLU	512	(3,3)	same	
	conv2D_e17	Convolution + ReLU	512	(3,3)	same	
	maxpool_e18	Maxpooling	1	(2,2)		
Bottleneck	full_connect	Dense + ReLU				4096
Decoder	conv2D_d1	Convolution + ReLU	512	(3,3)	same	
	conv2D_d2	Convolution + ReLU	512	(3,3)	same	
	conv2D_d3	Convolution + ReLU	512	(3,3)	same	
	upSamp_d4	UpSampling	1	(2,2)		
	conv2D_d5	Convolution + ReLU	512	(3,3)	same	
	conv2D_d6	Convolution + ReLU	512	(3,3)	same	
	conv2D_d7	Convolution + ReLU	512	(3,3)	same	
	upSamp_d8	UpSampling	1	(2,2)		
	conv2D_d9	Convolution + ReLU	256	(3,3)	same	
	conv2D_d10	Convolution + ReLU	256	(3,3)	same	
	conv2D_d11	Convolution + ReLU	256	(3,3)	same	
	upSamp_d12	UpSampling	1	(2,2)		
	conv2D_d13	Convolution + ReLU	128	(3,3)	same	
	conv2D_d14	Convolution + ReLU	128	(3,3)	same	
	upSamp_d15	UpSampling	1	(2,2)		
	conv2D_d16	Convolution + ReLU	64	(3,3)	same	
	conv2D_d17	Convolution + ReLU	64	(3,3)	same	
	upSamp_d18	UpSampling	1	(2,2)		
	conv2D_d19	Convolution + ReLU	1	(3,3)	same	