**Proceedings of the ASME 2022 17th International
Manufacturing Science and Engineering Conference
MSEC2022
June 27-July 1, 2022, West Lafayette, Indiana, USA**

MSEC2022-85422

LEARNING THE TEMPORAL EFFECT IN INFRARED THERMAL VIDEOS WITH LONG SHORT-TERM MEMORY FOR QUALITY PREDICTION IN RESISTANCE SPOT WELDING

Shenghan Guo
The School of Manufacturing
Systems and Networks, Arizona
State University, Mesa, AZ

Dali Wang, Jian Chen, Zhili Feng
Oak Ridge National Lab
Oak Ridge, TN

Weihong "Grace" Guo
Department of Industrial and
Systems Engineering, Rutgers
University
Piscataway, NJ

ABSTRACT

With the advances of sensing technology, in-situ infrared thermal videos can be collected from Resistance Spot Welding (RSW) processes. Each video records the formulation process of a weld nugget. The nugget evolution creates a "temporal effect" across the frames, which can be leveraged for real-time, nondestructive evaluation (NDE) of the weld quality. Currently, quality prediction with imaging data mainly focuses on optical feature extraction with Convolutional Neural Network (CNN) but does not make the most of such temporal effect. In this study, pixels corresponding to critical locations on the weld nugget surface are extracted from a video to form multivariate time series (MTS). Multivariate Adaptive Regression Splines (MARS) is used in MTS processing to remove noisy signals related to uninformative frames. A Stacked Long Short-Term Memory (LSTM) model is developed to learn from the processed MTS and then predicts weld nugget size and thickness in real-time NDE. Results from a case study on RSW of Boron steel demonstrates the improvement in prediction accuracy and computational time with the proposed method, as compared to CNN-based weld quality prediction.

Keywords: long short-term memory, resistance spot welding, infrared thermal video, temporal effect, quality prediction

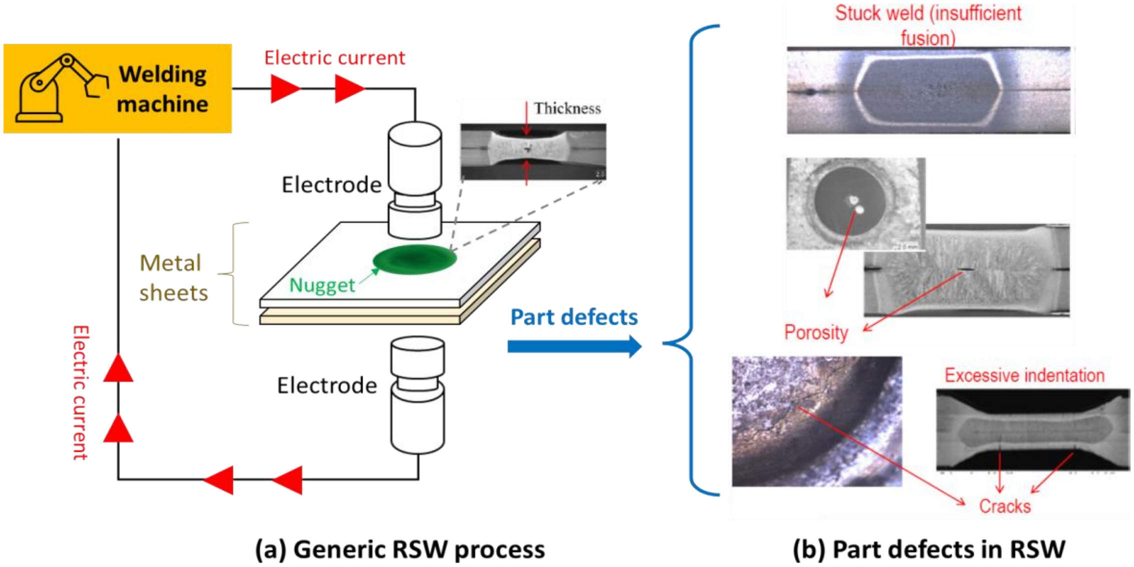
NOMENCLATURE

$\boldsymbol{y}$; $\boldsymbol{y}_1, \dots, \boldsymbol{y}_n$	Vector of pixels at the critical locations in a frame (or subsequent frames of a video)
x	Timestamps or index of frames
$\boldsymbol{D}$	MTS extracted from a video
$\hat{\boldsymbol{y}} = \hat{f}(x)$	Fitted value of $\boldsymbol{y}$ with MARS
$\boldsymbol{a}_1, \dots, \boldsymbol{a}_M$	Vectors of coefficients in MARS
$B_1, \dots, B_M$	Basis functions in MARS
$R_1, \dots, R_M$	Subregions representing the partitions of $\boldsymbol{D}$
$c_1, \dots, c_M$	Knots at the partitioning position of $\boldsymbol{D}$
c^*	Knot for the most abrupt change in $\boldsymbol{D}$
$\boldsymbol{D}^*$	MTS after c^*
L	Length of MTS segments
$\boldsymbol{d}_t$	MTS segments at time t (or the tth pixel vector) in $\boldsymbol{D}^*$
w_{ij}	Weight connecting unit j to i in LSTM
g_i	Differentiable function for activation in LSTM
$y^i(t)$	Activation of a noninput unit i at time t in LSTM
$net_i(t)$	Unit i's current input in LSTM
$v_j(t)$	Error signal of a nonoutput unit j
ξ_j	Memory cell at unit j in LSTM

1. INTRODUCTION

Resistance spot welding (RSW) is a popular technique for joining lightweight materials in automotive industry. Metal sheets are clamped together and placed between two water-cooled electrodes, as shown in **FIGURE 1**(a), which will concentrate electric currents and add pressure on the weld position. The resistance of metal against electric currents will generate heat and create a molten spot at the faying surface. The electrodes will be lifted after certain holding time, leaving the weld spot to solidify. Eventually, a *nugget* will be formulated at the weld spot [1].

RSW is economical, efficient, and operationally simple [1], which makes it favorable for joining lightweight steel and aluminum alloys in large-scale production [2]. Nonetheless, defects can be caused by unstable electric currents, insufficient/extra holding time, and other industrial uncertainties. **FIGURE 1**(b) shows the major defects in RSW, including insufficient/no fusion, porosity, and cracks. They are mainly reflected as abnormal *nugget size* and *thickness*. To ensure desirable mechanical properties of the weld, nondestructive



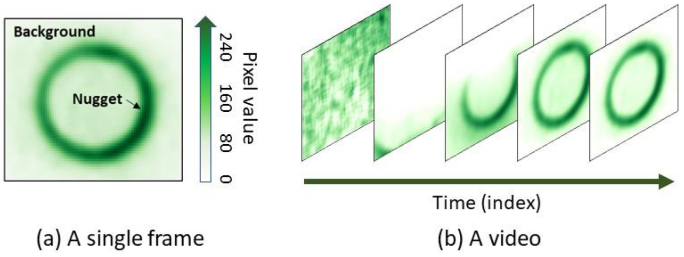
(a) Generic RSW process

(b) Part defects in RSW

FIGURE 1. (a) GENERIC RSW PROCESS AND (b) MAJOR PART DEFECTS [4].

evaluation (NDE) of nugget size and thickness, preferably done in real time, is of keen interest in industry.

In recent years, practitioners began to leverage inline sensing technology to capture in-situ data from RSW processes. Image sensors, e.g., high-speed infrared (IR) camera, can be mounted above or near the metal sheets and collect thermal video of the nugget during the welding process. An in-situ thermal video records the entire formulation of a nugget. The pixels represent the IR radiation from the nugget surface. Each individual frame (thermal image) reflects the nugget surface profile at a particular timestamp (**FIGURE 2**(a)), and consecutive frames reveal the evolution of nugget across time (**FIGURE 2**(b)). With a high speed of image recording, the nugget information is well preserved. In-situ thermal video therefore becomes a vital information source for real-time NDE in RSW.



(a) A single frame

(b) A video

FIGURE 2. (a) A SINGLE FRAME FROM AN IN-SITU THERMAL VIDEO AND (b) CONSTIUTION OF A VIDEO.

When it comes to quality prediction with imaging data, the prevalent methods are extracting optical features from images as the evidence of quality and then mapping the extracted features to quality metrics. Deep learning models designed for optical feature learning, e.g., Convolutional Neural Network (CNN), are widely adopted by these methods [3-6]. Despite the remarkable prediction accuracy, this methodology has two major drawbacks. First, working with high-dimensional data like image/video increases the data processing burden, which elevates the data handling cost and compromises the timeliness of NDE. Second, training CNNs and related DL models is computationally costly and time-consuming due to the image convolution [7]. An intriguing question to be asked, therefore, is whether an alternative method can be developed that alleviates the burden from data processing and model training? This study answers "YES" to this question.

Due to the reflection of nugget surface profile and nugget evolution across time, an in-situ thermal video from RSW contains both spatial and temporal characteristics in nugget formulation. CNN-based methods explore spatial features of nugget surface for quality prediction but do not fully leverage the temporal changes. This study proposes an alternative quality prediction method that explores the temporal effect in in-situ thermal videos to predict nugget size and thickness in RSW. **FIGURE 3** demonstrates the flow of method. In each video, pixels corresponding to the critical locations of weld nugget are extracted from the frames to form multivariate time series (MTS). Multivariate Adaptive Regression Splines (MARS) [8] is used to detect salient changes in the MTS, which correspond to the splitting points of informative frames containing stabilized nugget and those uninformative frames at the initial stage of nugget formulation. Uninformative signals are removed from the MTS. A *Stacked Long Short-Term Memory (LSTM)* model is trained with the filtered MTS, which enables real-time NDE of nugget size and thickness.

The proposed method substitutes the temporal characteristics for optical features from in-situ thermal videos as the evidence of weld quality. By doing so, a thermal video, which is a 3-dimensional tensor of pixels, is relaces by 2-dimensional MTS, which requires much less data storage and processing time. LSTM, as an advanced version of recurrent neural network

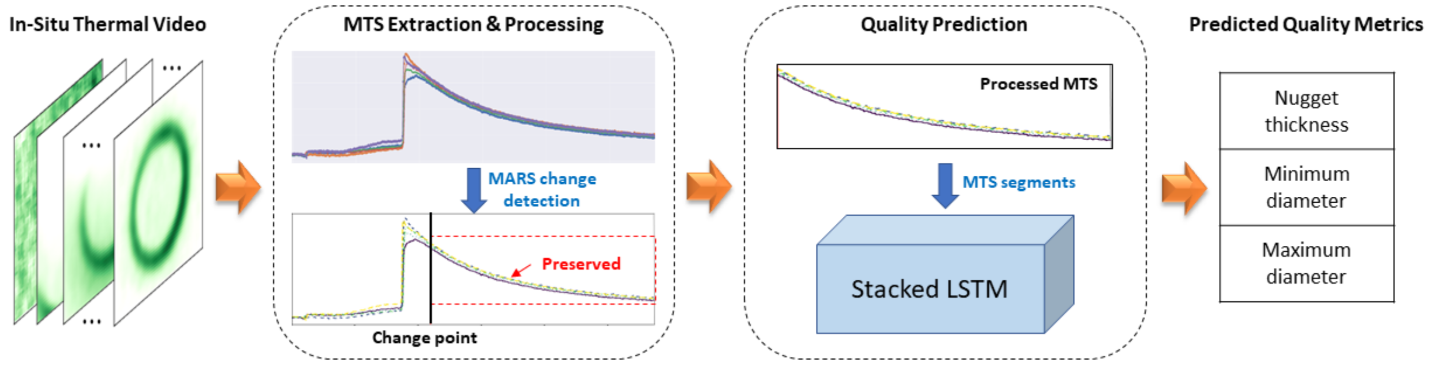


FIGURE 3. FLOWCHART OF THE PROPOSED METHOD.

(RNN), learns from MTS instead of images. The training time is substantially reduced due to lower input dimension and avoidance of optical feature extraction. A case study on an RSW application of Boron steel demonstrates that this method leads to a significant decrease in computing time while maintains the prediction accuracy in NDE at a favorable level, as compared with CNN-based methods.

The rest of this paper is organized as follows. Section 2 will review state-of-the-art for LSTM models and their use in manufacturing. Section 3 will elaborate on technical details of the proposed method. Section 4 will present a case study based on the in-situ thermal videos from an RSW application of Boron steel. Section 5 will conclude the paper and highlight future directions.

2. LITERATURE REVIEW

This section reviews LSTM and the model variants and discuss their current uses and potentials for NDE in manufacturing.

2.1 LSTM Models

First introduced in Hochreiter and Schmidhuber [9], LSTM was developed to handle the "vanishing gradient problem" in model training of conventional RNNs. When using gradient-based algorithms, e.g., stochastic gradient decent [10], to update the weights in conventional RNNs proportionally to the partial derivative of the error function per training iteration (epoch), the gradient can be vanishing and prevent the weights from updating [11]. LSTM incorporates *memory cells* and *gate units* in the RNN architecture to enforce constant error flow through constant error carousels within special units [9], therefore avoiding the "vanishing gradient problem". Due to this design, LSTM has a significantly improved learning capability for tasks that span a long period, e.g., thousands or even millions of discrete time steps. It is demonstrated for better model training performance and computational efficiency compared with conventional RNNs [12].

The simplest LSTM model consists of one LSTM layer. Recently, Deep LSTM, or Stacked LSTM [13, 14], has increasing presence in literature. Stacked LSTM models consist of multiple LSTM layers, with each layer adding one level of abstraction of the input at a different time scale [15]. Such model architecture was initially motivated by the need of representing flexible use of long range context in speech recognition [13, 14]. It turned out that the model depth could be a better compensation to the LSTM model skill than the number of memory cells in a given layer [13]. Consequently, Stacked LSTM became a most popular model for learning from sequence data or time series. Its applications have been extended to more subjects other than speech recognition and natural language processing [16]. An emerging field of use is manufacturing.

2.2 LSTM in Manufacturing

Many sensor data from manufacturing applications contain temporal changes that can be learned effectively with LSTM. For example, multi-sensory MTS of a functional component, e.g., lithium battery, bearing, reflect the degradation over time, which can be learned by LSTM models to predict the remaining useful life (RUL) [17-23]. With the aid of LSTM, data-driven RUL prediction has become a popular alternative to the traditional reliability analysis of manufacturing systems.

LSTM is also leveraged for tool condition monitoring and quality prediction. The rationale for using LSTM here is the temporal dependency of tool wear or quality downgrade. Multi-sensory MTS is, again, the dominant data type in this subject. Stacked LSTM [24-26], Bi-directional LSTM [27, 28], or customized LSTM models (e.g., AdaBoost-LSTM [29], LSTM-FFNN [30]) are developed to ensure a desirable prediction accuracy. The deeper and more complex model structures undoubtedly showed huge potentials for manufacturing quality prediction.

Recently, with the wide adoption of image sensors, imaging data are commonly acquired from manufacturing systems, e.g., in-situ thermal images from laser metal deposition [31]. These data reflect both the spatial characteristics and temporal changes of a part/product. A plausible thought is therefore applying LSTM to such imaging data to explore the temporal effect for quality prediction. The challenge, though, is that LSTM models intake sequence data or time series but not images. A specially designed LSTM model architecture emerged, i.e., CNN-LSTM [32, 33]. Optical feature extraction from the input images is done with the initial several convolutional layers, whose output becomes the input of the subsequent LSTM layers. LSTM layers treats the extracted features as MTS, learn the inherent temporal dependency, and eventually map them to quality metrics. This model design provides a strategy for applying LSTM on imaging data – an MTS extraction method can convert images to the input type for LSTM.

Inspired by the potentials of LSTM in learning from spatial-temporal images, this study attempts MTS extraction from in-situ thermal videos and training Stacked LSTM for RSW weld quality prediction. As an experimental study for building the bridge between thermal video and LSTM, this work lies the foundation for more advanced research along this direction.

2. METHOD

This section elaborates the method development. It first describes the characteristics and special properties of the motivating data type, in-situ thermal videos from RSW (Section 3.1). Then, it proceeds to MTS processing with MARS (Section 3.2), followed by model details of Stacked LSTM (Section 3.3).

3.1 In-Situ Thermal Videos from RSW

An inline data acquisition system was developed for a lab application of RSW for Boron steel [34]. A high-speed IR camera (100fps) was mounted above the metal sheets to be welded and near the electrodes. The camera started recording when the RSW process of one nugget began and continued until the nugget was completed. There are separate datasets for two different manufacturing modes: (i) RSW of 2 no-coat Boron steel sheets, 1mm thickness for each; (ii) RSW of 3 Al-coated Boron steel sheets, 1 mm thickness for the top and bottom sheets and 2mm thickness for the middle one. A thermal video in (i) consists of 600~602 frames and one in (ii) consists of 500~504 frames. Each frame (in either dataset) is a grayscale thermal image of size 61×81. Depending on the recording time, the pixel values in the image may have different ranges. At the beginning of a video, the nugget is not well formulated yet, so the pixel values in early frames tend to be small, e.g., all below 20 (for a grayscale color range of [0, 255], from light to dark); for frames captured after the nugget has fully formulated and stabilized, the pixel values are large and typically range in [20, 100]. The generic temporal changes of a weld nugget, as reflected in these videos, is that the nugget fully formulates at certain timestamp (frame) and then remains stable thereafter.

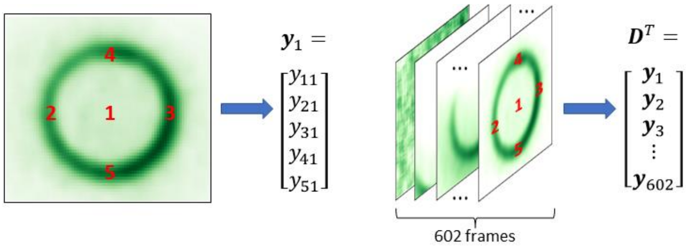
Dataset (i) has 25 raw videos and dataset (ii) has 22. For each video, there is a set of quality metrics obtained from offline, destructive testing, i.e., [*nugget thickness, minimum diameter, maximum diameter*]. **TABLE 1** shows the quality metrics for selected videos, where “Dmin” is minimal diameter and “Dmax” is maximal diameter, both measured for the same nugget. Each row in **TABLE 1** is for a nugget, and equivalently, a video. All the frames in a video have the same values of quality metrics.

TABLE 1. OFFLINE MEASUREMENTS FOR SELECTED VIDEOS IN DATASET (i). THICKNESS AND DIAMETER ARE IN MILLIMETER (mm).

Video	Thickness (mm)	Dmin (mm)	Dmax (mm)
1	1.899	3.135	3.311
2	1.905	3.135	3.289
3	1.871	4.923	4.923
4	1.875	4.875	4.945
5	1.861	5.740	5.762

3.2 MTS Processing with MARS

With the observation of nugget geometry, as well as the domain knowledge from destructive RSW testing [34], there are several *critical locations* in a weld nugget that are worth particular attention, e.g., nugget center and the boundary locations contouring the nugget size shown in **FIGURE 4**(a). The temporal evolution of these critical locations can largely reflect the geometric changes of nugget during the formulation process, which further reveals anomalies in nugget size and thickness. With the consideration of data processing burden, it is of interest to extract the pixel values for these critical locations as the input data for data-driven quality prediction, instead of using the entire video. The extracted pixels would form an MTS of 5 attributes, which is demonstrated in **FIGURE 4**(b).


4
2
1
3
5
$\boldsymbol{y}_1 =$
$\begin{bmatrix} y_{11} \\ y_{21} \\ y_{31} \\ y_{41} \\ y_{51} \end{bmatrix}$
...
...
...
4
3
1
2
5
602 frames
$\boldsymbol{D}^T =$
$\begin{bmatrix} \boldsymbol{y}_1 \\ \boldsymbol{y}_2 \\ \boldsymbol{y}_3 \\ \vdots \\ \boldsymbol{y}_{602} \end{bmatrix}$

(a) MTS extraction from a single frame

(b) MTS extraction from the entire video

FIGURE 4. MTS EXTRACTION: (a) CRITICAL LOCATIONS IN A SINGLE FRAME, AND (b) MTS EXTRACTION FROM THE ENTIRE VIDEO.

3.2.1 MTS extraction

The MTS extraction is as follows. Fix the coordinates, or row and column index (r, c), of the five critical locations in an image. For the data described in Section 3.1, they are (1) (30, 40), (2) (30, 20), (3) (30, 60), (4) (15, 40), and (5) (45, 40). Then, extract the pixel values for these coordinates from each frame of a video. The extracted values from a single frame is a vector of five elements, i.e., $\boldsymbol{y} \in \mathbb{R}^5$. Combining all the extracted vectors from a video, an MTS of 5 columns ($\boldsymbol{D}$) is obtained, $\boldsymbol{D} = [\boldsymbol{y}_1, \boldsymbol{y}_2, \dots, \boldsymbol{y}_n]$, each timestamp corresponds to one of the n frames in a video. Such MTS extraction from fixed coordinates works for the RSW data here because the nugget position in subsequent frames of a video did not change due to the fixed camera above the nugget. There was neither obvious recording error that caused position shift of the nugget across frames.

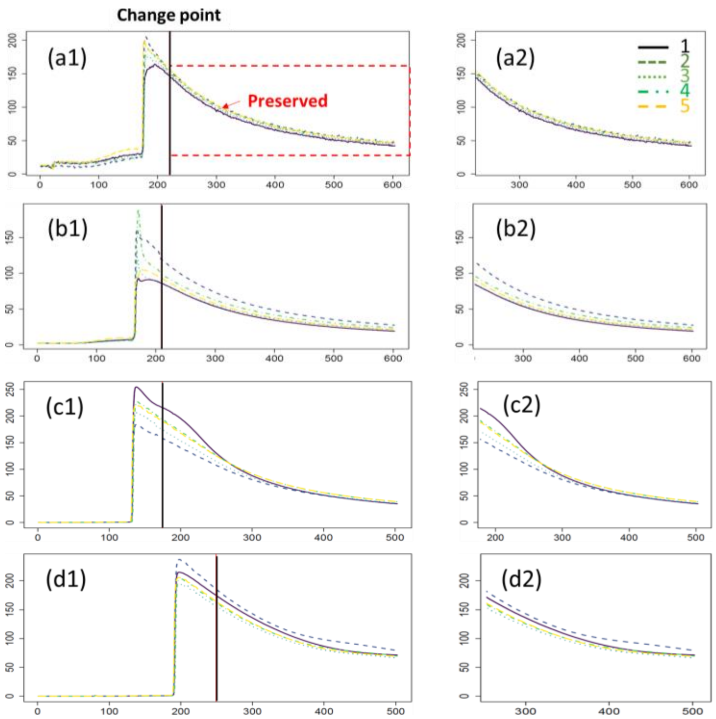


FIGURE 5. CHANGE-POINTS IN MTS AND MTS TRUNCTED AT THE CHANGE-POINT. LEFT COLUMN: CHANGE-POINT (c^*) DETECTED BY MARS IN MTS. RIGHT COLUMN: MTS AFTER PROCESSING. (a) VIDEO 1, DATASET (i); (b) VIDEO 6, DATASET (i); (c) VIDEO 1, DATASET (ii); (d) VIDEO 6, DATASET (ii). EACH COLOR AND LINE STYLE REPRESENTS ONE CIRTICIAL LOCATION IN IMAGE.

3.2.2 MTS processing

Now, before using the extracted MTS to train a data-driven quality prediction model, preliminary processing is necessary to remove signals corresponding to early frames. As seen in **FIGURE 2**, early frames in a video do not contain a fully formulated weld nugget because the electrodes have not been lifted away from the metal sheets. In the visualization of extracted MTS from a video in dataset (i), as shown in **FIGURE 5**(a1-d1), abrupt shift of pixel values is observed at a particular timestamp, which is because the electrodes are lifted, and a fully formulated but not-yet-stable nugget appears in the focal area of the camera. According to the domain knowledge, properties of a fully formulated nugget are the evidence for weld quality. An incomplete nugget is not quite informative to support the quality prediction, so signals related to the early frames in a video should be discarded.

Removing the uninformative signals from an extracted MTS, in this situation, is equivalent to detecting the change-point [35] in it. As revealed in **FIGURE 5**(a1-d1), a single, abrupt change-point split the early signals with the later ones associated with a stabilized nugget. For such abrupt-change-detection problem, MARS is a suitable method to use. Introduced in Friedman [8], MARS was designed to handle the underlying relationship in data sequence that involves interactions in at most a few variables. It models the data, $\boldsymbol{D}$, as following:

$$\hat{\boldsymbol{y}} = \hat{f}(x) = \sum_{m=1}^{M} \boldsymbol{a}_m B_m(x) \quad (1)$$

where $x = 1, 2, \dots, n$ is the timestamp or index of frames, $\hat{\boldsymbol{y}}$ is the model-fitted value of $\boldsymbol{y}$, $\boldsymbol{a}_m \in \mathbb{R}^5$ is the vector of coefficients, and B_m is the *basis* function that takes the general form of an indicator, $B_m(x) = \mathbb{I}\{x \in R_m\}$, with $\{R_m\}_1^M$being subregions representing the partitions of $\boldsymbol{D}$ [8]. Multiple basis functions can be (linearly) combined to form an underlying relationship piecewise. Commonly, the basis is written as Hinge functions in the form $B_m(x) = \max(0, x - c_m)$, where c_m is a *knot* at the partitioning position of $\boldsymbol{D}$. This basis function is adopted here, leading to:

$$\hat{\boldsymbol{y}} = \hat{f}(x) = \sum_{m=1}^{M} \boldsymbol{a}_m \max(0, x - c_m) \quad (2)$$

The identification of c_m can be achieved by MARS algorithm [8]. In model fitting, MARS algorithm uses stepwise (forward/backward) variable selection to find knots of data sequence. In an iteration of the algorithm, the model in Eq. (2) is fitted to the data, which reveals the knots for the iteration, and the squared error loss (SEL) is calculated. In subsequent iterations, more knots can be added, or certain existing knots can be pruned, to reduce the SEL. The algorithm continues until the SEL does not change significantly across iterations, or the allowed maximal number of knots is reached [8, 36].

The role of c_m can be related to change-points in data. The use of Eq. (2) as a change-point model has made some success [37]. In this study, the abrupt change in the MTS extracted from a video can be treated as a sole knot and identified with MARS algorithm. Signals after the abrupt change are preserved, which are related to a fully formulated nugget. The partition point, or identified knot, in MTS is marked by a solid vertical line in **FIGURE 5**(a1-d1). With only a single knot preserved as c^*, the MARS algorithm partitions the MTS right after the abrupt change-point. To be prudent, all early signals up to c^* are removed, resulting in the processed MTS, $\boldsymbol{D}^* = [\boldsymbol{y}_{c^*+1}, \boldsymbol{y}_{c^*+2}, \dots, \boldsymbol{y}_n]$, in **FIGURE 5**(a2-d2). The processed MTS will be further truncated into MTS segments of a fixed length (L) to become the input of Stacked LSTM. This will be elaborated in subsection 3.3.2.

3.3 Quality Prediction with Stacked LSTM

This section specifies the technical details of LSTM and the Stacked LSTM model used in quality prediction for RSW.

3.3.1 LSTM

The distinguishing property of LSTM is the prevention of "vanishing gradient". The central feature to enable this is *constant error carousel (CEC)*. Denote the activation of a noninput unit i at time t by $y^i(t) = g_i\big(net_i(t)\big)$, where g_i is a differentiable function, and $net_i(t) = \sum_j w_{ij} y^j(t-1)$ is unit i's current input with weight w_{ij} connecting unit j to i. The error signal of a nonoutput unit j is then

$$v_j(t) = g_j'\left(net_j(t)\right) \sum_i w_{ij} v_i(t+1) \tag{3}$$

Consider a single unit j with connection to itself. Its error signal is therefore $v_j(t) = g_j'\left(net_j(t)\right) v_i(t+1) w_{jj}$. To enforce constant error flow through j, it is required that

$$g_j'\left(net_j(t)\right) w_{jj} = 1 \tag{4}$$

which implies

$$g_j\left(net_j(t)\right) = net_j(t)/w_{jj}, \forall net_j(t) \tag{5}$$

A further implication is that g_j must be linear, and unit j's activation remains constant:

$$y_j(t+1) = g_j\left(net_j(t+1)\right) = g_j\left(w_{jj} y^j(t)\right) = y^j(t) \tag{6}$$

which is fulfilled by using the identity function, $g_j(t) = t, \forall t$, and setting $w_{jj} = 1$. This design is the CEC in LSTM [9].

Instead of using naïve CEC, LSTM incorporates *gate units* to allow for constant error flow through special, self-connected units. A multiplicative *input gate* unit is incorporated to protect the memory stored in j from perturbation by irrelevant inputs; a multiplicative *output gate* unit is added to protect other units from perturbation by currently irrelevant memory stored in j. The resulted unit is a *memory cell* [9], denoted by ξ_j. **FIGURE 6** demonstrates the memory cell topology. ξ_j gets input from net_{ξ_j}, which is $net_{\xi_j}(t) = \sum_u w_{\xi_j,u} y^u(t-1)$, a multiplicative unit in_j from the input gate (at the 2nd "x" from left), and a multiplicative unit out_j from the output gate (at the 3rd "x" from left). Their activations are respectively:

$$y^{out_j}(t) = g_{out_j}\left(net_{out_j}(t)\right)$$

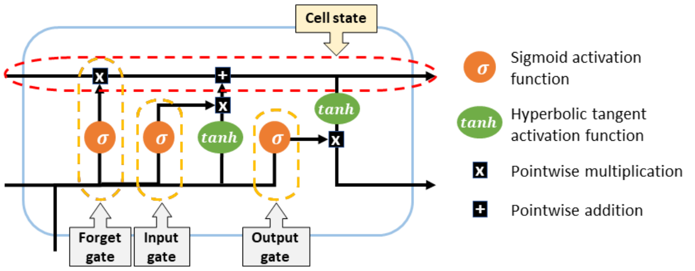


FIGURE 6. MEMORY CELL TOPOLOGY IN LSTM, WITH FORGET UNIT INCORPORATED.

$$y^{in_j}(t) = g_{in_j}\left(net_{in_j}(t)\right) \tag{7}$$

where $net_{out_j}(t) = \sum_u w_{out_j,u} y^u(t-1)$ and $net_{in_j}(t) = \sum_u w_{in_j,u} y^u(t-1)$ [9]. In recent literature [38], a *forget gate* can be added to the memory cell to allow memory reset (at the 1st "x" from left), thus preventing the LSTM internal memory from growing indefinitely and eventually break down. The cell state, as the "memory" of the network, carries relevant information throughout the processing of the sequence data. Even information from earlier time steps can be passed to later time steps, thus reducing the effects of "short-term memory", or equivalently the "vanishing gradient". In model training, information is added to or removed from the cell state via gates. Popular computational tools for LSTM, e.g., "keras" library written in Python [39], mostly incorporate a forget gate in the generic LSTM layer.

3.3.2 Stacked LSTM for in-situ thermal videos

For the simplest model structure of LSTM, a single layer consisting of multiple memory cells is used to map the input data to output values, which are quality metrics in the context of data-driven NDE. However, as mentioned in Section 2.1, adding LSTM layers can better enhance the model skills than increasing the number of memory cells in a single LSTM layer [13]. Deeper model architecture, e.g., Stacked LSTM, has demonstrated its good performance in use cases [24-26].

This work adopts Stacked LSTM to handle the processed MTS $\boldsymbol{D}^*$. The input data for Stacked LSTM, and other LSTM models, are sequence data for a time window of certain length,

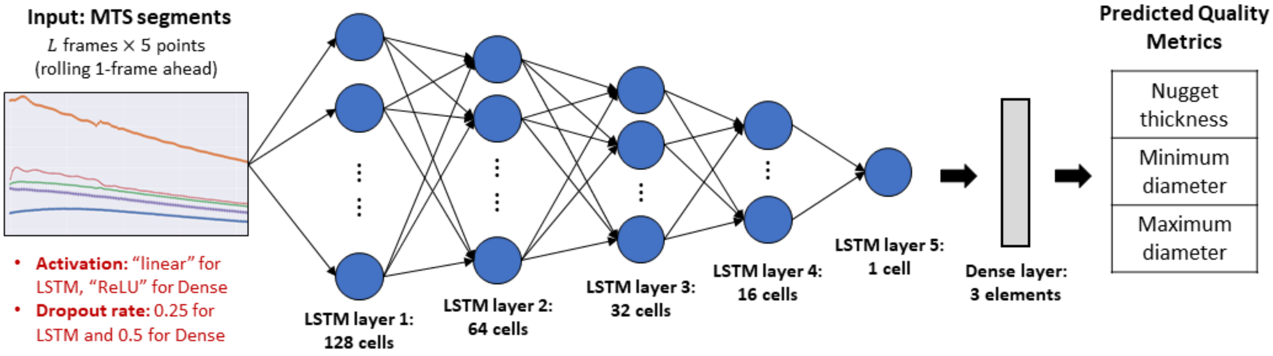
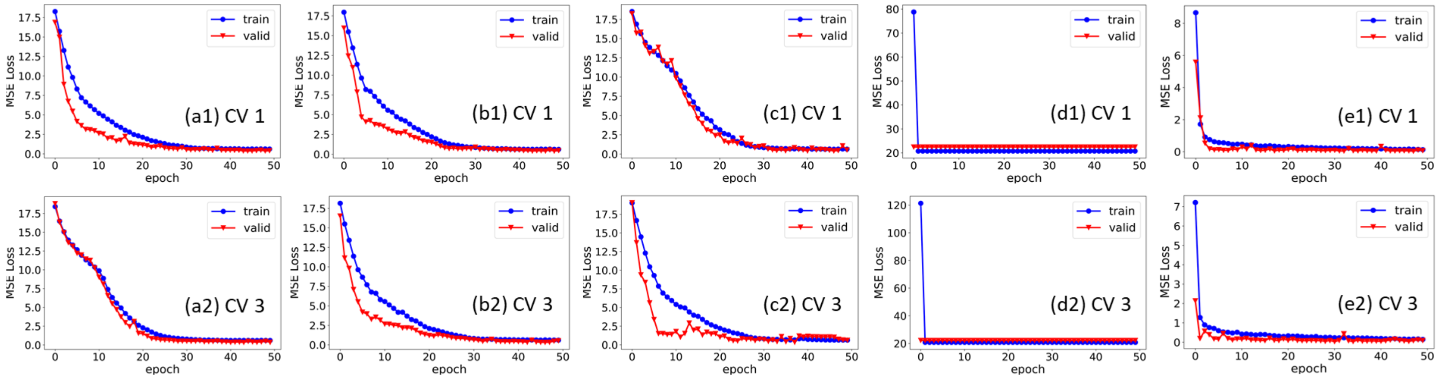
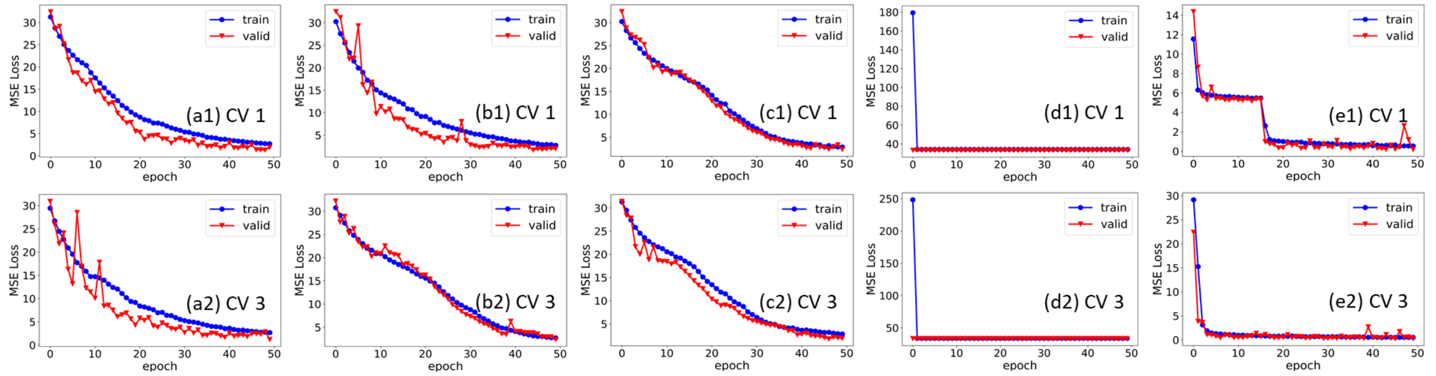


FIGURE 7. MODEL ARCHITECTURE OF THE STACKED LSTM USED IN THIS STUDY.



(a) Stacked LSTM, $L = 5$ (b) Stacked LSTM, $L = 10$ (c) Stacked LSTM, $L = 15$ (d) AlexNet (e) VGG7

FIGURE 8. TRAINING PERFORMANCE OF FIVE MODELS FOR DATASET (i). IN LEGEND, "TRAIN" FOR TRAINING LOSS, "VALID" FOR VALIDATION LOSS.



(a) Stacked LSTM, $L = 5$ (b) Stacked LSTM, $L = 10$ (c) Stacked LSTM, $L = 15$ (d) AlexNet (e) VGG7

FIGURE 9. TRAINING PERFORMANCE OF FIVE MODELS FOR DATASET (ii). IN LEGEND, "TRAIN" FOR TRAINING LOSS, "VALID" FOR VALIDATION LOSS.

or equivalently, MTS segments. It indicates that $\boldsymbol{D}^*$ must be further truncated into multiple segments of length L, i.e., $\boldsymbol{d}_t = [\boldsymbol{y}_{t-L+1}, \boldsymbol{y}_{t-L}, \dots, \boldsymbol{y}_t], t = 1, 2, \dots, \lfloor n/L \rfloor$. Also, to preserve the temporal order of signals, a window of length L should be rolled forward to exhaust all the signals in $\boldsymbol{D}^*$. The step size of rolling the window is typically set to a small number, e.g., 1, which creates overlap between subsequent segments.

Unlike CNNs, there are rarely benchmark architectures for Stacked LSTM. The number of memory cells in each layer and the number of LSTM layers are mainly tuned for the given dataset. The model structure used here is illustrated in **FIGURE 7**. It consists of 5 LSTM layers, with the layers containing 128, 64, 32, 16, and 1 memory cell, from left to right. A batch-normalization layer is appended after each LSTM layer to facilitate the training process, followed by a drop-out layer to avoid overfitting. Finally, a dense layer maps the output from the last drop-out layer to the quality metrics, i.e., [nugget thickness, minimum diameter, maximum diameter].

4. CASE STUDY

This section demonstrates the proposed method on in-situ thermal videos acquired from the RSW process of Boron steel, as described in Section 3.1. Both training and prediction performance are evaluated. Here, "performance" refers to the convergence of model in training and the accuracy in prediction. A performance comparison is done among two popular CNN models, i.e., AlexNet [40] and VGG7 [41], and the proposed Stacked LSTM. The computing efficiency is also compared. These models are trained for both dataset (i) and (ii), separately. Different lengths of segment, i.e., $L \in \{5, 10, 15\}$, are used to generate the results for Stacked LSTM. The corresponding total number of MTS segments, before any training/testing data split, is given in **TABLE 2**. To have a fair comparison, the raw frames used for training and testing all the models are identical, which are the frames in a video after c^*. All computation is done with a Dell Inspiron 5593 PC, with Intel(R) Core(TM) i7-1065G7 CPU, clock speed 1.30GHz~1.50 GHz.

TABLE 2. TOTAL NUMBER OF MTS SEGMENTS. ROLLING STEP SIZE IS 1.

Dataset	$L = 5$	$L = 10$	$L = 15$
(i)	8824	8712	8596
(ii)	5596	5484	5376

4-fold Cross-Validation (CV) is adopted. The instances of a dataset are randomly shuffled and assigned to four equally sized folds without replacement. In one run of implementation, one fold is preserved as the testing data. For the rest three folds, 80% MTS segments in them are used for model training and 20% for training-phase model validation. For all the models, including the benchmark CNNs, 50 epochs are completed to train the model, without batching, using "ADAM" optimizer [42]. Since the quality metrics of interest, i.e., [thickness, Dmin, Dmax], are numeric values, the model fitness in training and accuracy in prediction can both be measured by mean squared error (MSE) loss [43], which quantifies the discrepancy between model-fitted values (or prediction) and the ground truth, and a smaller value implies a better result.

4.1 Model Training

The training performance for dataset (i) and (ii) is demonstrated in **FIGURE 8** and **9**, respectively. In each subplot, the horizontal axis shows the index of training epoch, and the vertical axis shows the MSE loss between the model-fitted values and the true value of quality metrics. The (blue) curve with dot markers represents training loss, and the (red) curve with triangle markers represents training-phase validation loss. Each column is associated with one model, and the 1st and 2nd rows are for CV1 and CV3, respectively.

For both datasets, the tendency of MSE loss curves regarding a specific model is similar. The proposed method, Stacked LSTM, has converging training and validation loss for all the L values (**FIGURE 8**(a-c) and **FIGURE 9**(a-c)). The MSE loss after convergence is small, with (i) almost equals to 0 and (ii) approaches 0. Such well-converged MSE loss indicates the effectiveness of model training – Stacked LSTM learned sufficient information from the data to reduce the MSE loss over training epoch, and the learning speed is decent, considering the fast convergence within 50 epochs.

TABLE 3. AVERAGE COMPUTING TIME (IN SECOND) FOR MODEL TRAINING AND TESTING WITH 4-FOLD CV.

Dataset	Stacked LSTM			CNNs	
	$L = 5$	$L = 10$	$L = 15$	AlexNet	VGG7
(i)	238.25	325	393	15346.75	1329.5
(ii)	132.25	182.25	259	10049.25	822.5

The training performance for benchmark CNNs are more complicated. The two architectures adopted here are relatively shallow, with VGG7 even shallower than AlexNet. The MSE loss curves for AlexNet (**FIGURE 8**(d) and **FIGURE 9**(d)) converged to a fixed, nontrivial value and then stay there ever since, implying that the training algorithm stuck to a value and do not optimize properly. Though not very deep, AlexNet still has considerate complexity (equivalently, too many parameters) in model architecture, which is inappropriate for the RSW data. The training performance of VGG7 (**FIGURE 8**(e) and **FIGURE 9**(e)), on the other hand, is the best among five models. The convergence speed is the fastest and the MSE loss upon convergence is nearly 0. It indicates that VGG7 happens to have an appropriate architecture for the given RSW data.

Nonetheless, whether the training performance of the benchmark CNNs is good or not, they consume much longer time for data processing and model training. **TABLE 3** shows the computing time (in second) for all the five models. The time consumed by Stacked LSTM is generally way less than CNNs. This is critical for manufacturing practice. It is not worth the effort to spend so much time training CNNs and finding the suitable CNN architecture. Even with a shallow architecture like VGG7, the model training can be rather time-consuming. Stacked LSTM is a better option with its acceptable training performance and the superior computing efficiency.

4.2 Quality Prediction

The prediction performance is also evaluated for Stacked LSTM and benchmark CNNs. **TABLE 4** shows the MSE loss from prediction (model testing) for both dataset (i) and (ii). The performance metrics, since 4-fold CV is adopted, are the averaged value of minimum, mean, median, and maximum MSE loss across CVs. Each cell shows the mean value of metric and the standard deviation (std) in bracket.

TABLE 4. QUALITY PREDICTION PERFORMANCE: AVERAGE (*STANDARD DEVIATION*) OF MSE LOSS ACROSS 4-FOLD CV.

Data set	MSE loss	Stacked LSTM			CNNs	
		$L = 5$	$L = 10$	$L = 15$	AlexNet	VGG7
(i)	Min	0.0006 (*0.0002*)	0.0016 (*0.0018*)	0.0006 (*0.0004*)	7.8554 (*0*)	0.0000 (*0*)
	Mean	0.3875 (*0.0587*)	0.4090 (*0.1084*)	0.6452 (*0.2562*)	21.0404 (*0.0474*)	0.0426 (*0.0202*)
	Median	0.2623 (*0.0319*)	0.2786 (*0.0423*)	0.6817 (*0.4067*)	23.1990 (*0*)	0.0230 (*0.0175*)
	Max	5.2534 (*6.1644*)	7.5843 (*9.3358*)	3.9532 (*1.8785*)	32.3619 (*0*)	0.6078 (*0.1085*)
(ii)	Min	0.0023 (*0.0001*)	0.0132 (*0.0158*)	0.0221 (*0.0115*)	8.4609 (*0*)	0.0006 (*0.0003*)
	Mean	1.6407 (*0.4186*)	2.0306 (*0.3568*)	2.2591 (*0.7992*)	33.7814 (*0.5055*)	0.2394 (*0.2220*)
	Median	0.8142 (*0.1999*)	1.2296 (*0.3498*)	1.4455 (*0.2554*)	34.5470 (*0.4327*)	0.0911 (*0.0577*)
	Max	6.8040 (*0.8234*)	9.0132 (*1.6172*)	8.7249 (*2.1158*)	67.0311 (*0*)	2.6778 (*2.6153*)

Stacked LSTM models generally have small MSE loss, as revealed by the low values of the metrics. For both datasets, $L = 5$ leads to the best result, but the discrepancy associated with different L values is not salient. Dataset (i) has slightly lower metric values, or equivalently better prediction performance, than dataset (ii), but can be associated with larger std when L is small.

The prediction performance of benchmark CNNs are at the extremes. AlexNet does a poor job in prediction – it tends to predict the same results for all testing data, which is likely an outcome of the trapped training performance. VGG7, again, does the best job among the five models. It has extremely low values for all the metrics, indicating a remarkable prediction accuracy.

4.3 Discussion

To summarize, Stacked LSTM has favorable performance for both training and prediction. It consumes at least 3-time less of the time for data processing and model training than benchmark CNNs. Such superior computing efficiency and relatively high prediction accuracy make it a better option in practice when computational burden is a concern. CNNs, as a contrast, generally require a long time to train. It is challenging to find the suitable CNN architecture, especially when experimenting with different architectures is so time-consuming.

For RSW applications, the proposed Stacked LSTM will substantially facilitate real-time NDE. First, the low data processing burden associated with Stacked LSTM will reduce the potential cost of NDE. This is especially important in the phase of model development (equivalently, model training), because the requirement for computing capability is lower for handling MTS data, as compared to videos. Second, the hurdle of utilizing DL-based NDE may be mitigated by Stacked LSTM. As revealed in Section 4, the raw training data contains only a few videos, yet the mechanism of Stacked LSTM allows a natural augmentation of the data (a few videos → thousands of MTS segments). The prediction accuracy of Stacked LSTM trained with such data, after only 50 epochs, has reached a high level already. It implies the possibility of widening the adoption of DL-based, real-time NDE in RSW and other welding applications, especially when the training data is sparse.

5. CONCLUSION

In this study, a quality prediction method based on MARS and Stacked LSTM was proposed for in-situ thermal videos from RSW. By using Stacked LSTM to learn the temporal effect in data as the evidence for quality prediction, the proposed method has remarkably improved the computing efficiency of DL-based NDE, at the expense of none or trivial decrease in prediction accuracy. The data storage requirement is also lowered due to the use of MTS instead of videos. This method can be readily applied on spatial-temporal imaging data from other manufacturing or engineering applications.

In future, this study can be extended along several directions. First, the model parameters and architecture of the LSTM model can be further tuned, preferably on larger datasets of in-situ thermal videos, to improve the accuracy of quality prediction. Second, the method can be customized for spatial-temporal data with more complex structure, e.g., in-situ thermal videos with multiple abrupt changes or unfixed object position in image. Third, the current MTS extraction only considers a handful of critical locations. It is of interest to have more locations considered in an image, and meanwhile develop an adaptive method to identify the critical locations in the object based on the (training-stage) learning performance of DL models.

ACKNOWLEDGEMENTS

This article was supported in part by the US Department of Energy, in part by the Office of Nuclear Energy (Advanced Methods for Manufacturing Program), and in part by the AI Initiative at Oak Ridge National Laboratory.

REFERENCES

[1] Manladan, S., F. Yusof, S. Ramesh, M. Fadzil, Z. Luo, and S. Ao, "A review on resistance spot welding of aluminum alloys". *The International Journal of Advanced Manufacturing Technology*, 90(1-4): p. 605-634. 2017.

[2] Chen, C., L. Kong, M. Wang, A.S. Haselhuhn, D.R. Sigler, H.-P. Wang, and B.E. Carlson, "The robustness of Al-steel resistance spot welding process". *Journal of Manufacturing Processes*, 43: p. 300-310. 2019. https://doi.org/10.1016/j.jmapro.2019.02.030

[3] Tian, Q., S. Guo, E. Melder, L. Bian, and W.G. Guo, "Deep Learning-Based Data Fusion Method for In Situ Porosity Detection in Laser-Based Additive Manufacturing". *Journal of Manufacturing Science and Engineering*, 143(4). 2020. 10/31/2021. 10.1115/1.4048957

[4] Guo, S., D. Wang, J. Chen, Z. Feng, and W.G. Guo, "Predicting Nugget Size of Resistance Spot Welds Using Infrared Thermal Videos With Image Segmentation and Convolutional Neural Network". *Journal of Manufacturing Science and Engineering*, 144(2). 2021. 10/31/2021. 10.1115/1.4051829

[5] Janssens, O., R.V.d. Walle, M. Loccufier, and S.V. Hoecke, "Deep Learning for Infrared Thermal Image Based Machine Health Monitoring". *IEEE/ASME Transactions on Mechatronics*, 23(1): p. 151-159. 2018. 10.1109/TMECH.2017.2722479

[6] Li, X., S. Siahpour, J. Lee, Y. Wang, and J. Shi, "Deep Learning-Based Intelligent Process Monitoring of Directed Energy Deposition in Additive Manufacturing with Thermal Images". *Procedia Manufacturing*, 48: p. 643-649. 2020. https://doi.org/10.1016/j.promfg.2020.05.093

[7] Chen, X. and X. Lin, "Big Data Deep Learning: Challenges and Perspectives". *IEEE Access*, 2: p. 514-525. 2014. 10.1109/ACCESS.2014.2325029

[8] Friedman, J.H., "Multivariate Adaptive Regression Splines". *The Annals of Statistics*, 19(1): p. 1-67. 1991. 2021/11/04/.

[9] Hochreiter, S. and J. Schmidhuber, "Long Short-Term Memory". *Neural Computation*, 9(8): p. 1735-1780. 1997. 10.1162/neco.1997.9.8.1735

[10] Bottou, L., "Stochastic gradient learning in neural networks". *Proceedings of Neuro-Nımes*, 91(8): p. 12. 1991.

[11] Hochreiter, S., "The Vanishing Gradient Problem During Learning Recurrent Neural Nets and Problem Solutions". *International Journal of Uncertainty, Fuzziness and*

Knowledge-Based Systems, 06(02): p. 107-116. 1998. 2021/11/01. 10.1142/S0218488598000094

[12] Schmidhuber, J., "Deep learning in neural networks: An overview". *Neural Networks*, 61: p. 85-117. 2015. https://doi.org/10.1016/j.neunet.2014.09.003

[13] Graves, A., A.-r. Mohamed, and G. Hinton. "Speech recognition with deep recurrent neural networks". in *2013 IEEE international conference on acoustics, speech and signal processing*. 2013. Ieee.

[14] Fernández, S., A. Graves, and J. Schmidhuber. "Sequence labelling in structured domains with hierarchical recurrent neural networks". in *Proceedings of the 20th International Joint Conference on Artificial Intelligence, IJCAI 2007*. 2007.

[15] Pascanu, R., C. Gulcehre, K. Cho, and Y. Bengio, "How to construct deep recurrent neural networks". *arXiv preprint arXiv:1312.6026*. 2013.

[16] Van Houdt, G., C. Mosquera, and G. Nápoles, "A review on the long short-term memory model". *Artif. Intell. Rev.*, 53(8): p. 5929-5955. 2020.

[17] Zheng, S., K. Ristovski, A. Farahat, and C. Gupta. "Long Short-Term Memory Network for Remaining Useful Life estimation". in *2017 IEEE International Conference on Prognostics and Health Management (ICPHM)*. 2017.

[18] Zhang, J., P. Wang, R. Yan, and R.X. Gao, "Long short-term memory for machine remaining life prediction". *Journal of Manufacturing Systems*, 48: p. 78-86. 2018. https://doi.org/10.1016/j.jmsy.2018.05.011

[19] Zhang, Y., R. Xiong, H. He, and M.G. Pecht, "Long Short-Term Memory Recurrent Neural Network for Remaining Useful Life Prediction of Lithium-Ion Batteries". *IEEE Transactions on Vehicular Technology*, 67(7): p. 5695-5705. 2018. 10.1109/TVT.2018.2805189

[20] Wu, J., K. Hu, Y. Cheng, H. Zhu, X. Shao, and Y. Wang, "Data-driven remaining useful life prediction via multiple sensor signals and deep long short-term memory neural network". *ISA Transactions*, 97: p. 241-250. 2020. https://doi.org/10.1016/j.isatra.2019.07.004

[21] Zhang, B., S. Zhang, and W. Li, "Bearing performance degradation assessment using long short-term memory recurrent network". *Computers in Industry*, 106: p. 14-29. 2019. https://doi.org/10.1016/j.compind.2018.12.016

[22] Li, X., L. Zhang, Z. Wang, and P. Dong, "Remaining useful life prediction for lithium-ion batteries based on a hybrid model combining the long short-term memory and Elman neural networks". *Journal of Energy Storage*, 21: p. 510-518. 2019. https://doi.org/10.1016/j.est.2018.12.011

[23] Tao, Z., Q. An, G. Liu, and M. Chen, "A novel method for tool condition monitoring based on long short-term memory and hidden Markov model hybrid framework in high-speed milling Ti-6Al-4V". *The International Journal of Advanced Manufacturing Technology*, 105(7): p. 3165-3182. 2019. 10.1007/s00170-019-04464-w

[24] Cai, W., W. Zhang, X. Hu, and Y. Liu, "A hybrid information model based on long short-term memory network for tool condition monitoring". *Journal of Intelligent Manufacturing*, 31(6): p. 1497-1510. 2020. 10.1007/s10845-019-01526-4

[25] Lindemann, B., N. Jazdi, and M. Weyrich. "Anomaly detection and prediction in discrete manufacturing based on cooperative LSTM networks". in *2020 IEEE 16th International Conference on Automation Science and Engineering (CASE)*. 2020.

[26] Guo, W., C. Wu, Z. Ding, and Q. Zhou, "Prediction of surface roughness based on a hybrid feature selection method and long short-term memory network in grinding". *The International Journal of Advanced Manufacturing Technology*, 112(9): p. 2853-2871. 2021. 10.1007/s00170-020-06523-z

[27] Habbouche, H., T. Benkedjouh, and N. Zerhouni, "Intelligent prognostics of bearings based on bidirectional long short-term memory and wavelet packet decomposition". *The International Journal of Advanced Manufacturing Technology*, 114(1): p. 145-157. 2021. 10.1007/s00170-021-06814-z

[28] Liu, X., S. Liu, X. Li, B. Zhang, C. Yue, and S.Y. Liang, "Intelligent tool wear monitoring based on parallel residual and stacked bidirectional long short-term memory network". *Journal of Manufacturing Systems*, 60: p. 608-619. 2021. https://doi.org/10.1016/j.jmsy.2021.06.006

[29] Bai, Y., J. Xie, D. Wang, W. Zhang, and C. Li, "A manufacturing quality prediction model based on AdaBoost-LSTM with rough knowledge". *Computers & Industrial Engineering*, 155: p. 107227. 2021. https://doi.org/10.1016/j.cie.2021.107227

[30] Kim, D., M. Kim, and W. Kim, "Wafer Edge Yield Prediction Using a Combined Long Short-Term Memory and Feed- Forward Neural Network Model for Semiconductor Manufacturing". *IEEE Access*, 8: p. 215125-215132. 2020. 10.1109/ACCESS.2020.3040426

[31] Guo, S., W.G. Guo, and L. Bain, "Hierarchical spatial-temporal modeling and monitoring of melt pool evolution in laser-based additive manufacturing". *IISE Transactions*, 52(9): p. 977-997. 2020. 10.1080/24725854.2019.1704465

[32] Essien, A. and C. Giannetti, "A Deep Learning Model for Smart Manufacturing Using Convolutional LSTM Neural Network Autoencoders". *IEEE Transactions on Industrial Informatics*, 16(9): p. 6069-6078. 2020. 10.1109/TII.2020.2967556

[33] Ghimire, S., R.C. Deo, N. Raj, and J. Mi, "Deep solar radiation forecasting with convolutional neural network and long short-term memory network algorithms". *Applied Energy*, 253: p. 113541. 2019. https://doi.org/10.1016/j.apenergy.2019.113541

[34] Carlson, B.E., A.S. Haselhuhn, J. Chen, and Z. Feng, "Nondestructive evaluation of resistance spot-welded Al-steel joints". *MRS Bulletin*, 44(8): p. 619-624. 2019. 10.1557/mrs.2019.179

[35] Yao, Y.-C., "Estimating the number of change-points via Schwarz' criterion". *Statistics & Probability Letters*, 6(3): p. 181-189. 1988. https://doi.org/10.1016/0167-7152(88)90118-6

[36] Friedman, J.H. and C.B. Roosen, "An introduction to multivariate adaptive regression splines". 1995, Sage Publications Sage CA: Thousand Oaks, CA.

[37] Guo, S., W.G. Guo, A. Abolhassani, and R. Kalamdani, "Nonparametric, real-time detection of process deteriorations in manufacturing with parsimonious smoothing". *IISE Transactions*, 53(5): p. 568-581. 2021. 10.1080/24725854.2020.1786195

[38] Gers, F.A., J. Schmidhuber, and F. Cummins, "Learning to Forget: Continual Prediction with LSTM". *Neural Computation*, 12(10): p. 2451-2471. 2000. 10.1162/089976600300015015

[39] Ketkar, N., "Introduction to keras", in *Deep learning with Python*. 2017, Springer. p. 97-111.

[40] Krizhevsky, A., I. Sutskever, and G.E. Hinton, "Imagenet classification with deep convolutional neural networks". *Advances in neural information processing systems*, 25: p. 1097-1105. 2012.

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

[42] Kingma, D.P. and J. Ba, "Adam: A method for stochastic optimization". *arXiv preprint arXiv:1412.6980*. 2014.

[43] Allen, D.M., "Mean Square Error of Prediction as a Criterion for Selecting Variables". *Technometrics*, 13(3): p. 469-475. 1971. 10.1080/00401706.1971.10488811

[44] Chen, J., Y.-C. Lim, H. Huang, Z. Feng, and X. Sun, "Ultrasonic welding of AZ31B magnesium alloy". *MRS Bulletin*, 44(8): p. 630-636. 2019. 10.1557/mrs.2019.182

[45] Jian, C. and F. Zhili. "IR-based spot weld NDT in automotive applications". in *Proc.SPIE*. 2015.

APPENDIX

A1. Acquisition of In-situ Videos

A high-speed camera (FLIR SC655) was positioned stationary around 15-20cm away from the weld area with a roughly 45 deg tilted angle. The camera captured the whole welding process for about 3s including both heating and cooling stages.

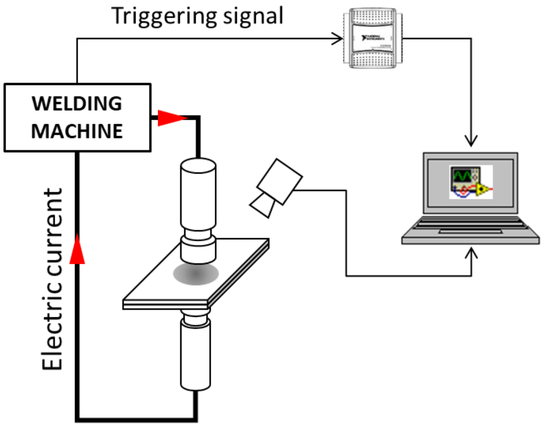


FIGURE A1. IN-SITU ACQUISITION OF VIDEOS FOR WELD NUGGET.

A2. Acquisition of Measurements

Each weld was cross-sectioned to reveal the weld nugget for measurement. Each tested weld is associated with three measurements, i.e., "Thickness", "Dmin", and "Dmax" (all in mm). "Thickness" is the measurement in the weld center using micrometer. "Dmin" and "Dmax" are the minimum and maximum diameter of the nugget measured nondestructively by ultrasonic C-scan method. A weld nugget is not standard round so the diameter varied slightly even for the same one, thus "Dmin" and "Dmax". Chen *et al.* [44] innovated the procedure that was used here to acquire the measurement data.

This paper focuses on the "nugget size" (i.e., "Dmin" and "Dmax") and "thickness", since they are the most important weld attributes related to the physical appearance as well as defects of welds. Defects can be lack-of-fusion, cracks, porosity, etc. Among all weld attributes, nugget size and thickness are closely related to the mechanical performance of the weld [45]. They are effective indicators of the weld quality. This study does not relate all defects to nugget size and thickness but focus on DL-based NDE of these weld attributes. By enabling real-time NDE of nugget size and thickness with the proposed method, this study strived to facilitate the efficiency improvement and cost reduction in RSW quality control.

A3. Full Records of Measurements

TABLE A1. FULL MEASUREMENT RECORDS: "THICKNESS", "DMIN", AND "DMAX" (ALL IN mm). CURRENT INTENSITY IS THE EXPERIMENTAL PARAMETER, WHOSE LEVEL CHANGED AMONG: "COLD", "MIN", "MID", "MAX250", "EXP".

Data	(i)				(ii)			
Sample	Current intensity	Thickness	Dmin	Dmax	Current intensity	Thickness	Dmin	Dmax
1	cold	1.899	3.1	3.311	cold	3.99	2.3	2.3
2	cold	1.905	3.1	3.289	cold	3.939	2.1	2.4
3	cold	1.905	3.1	3.201	cold	3.997	2.1	2.3
4	cold	1.903	3.2	3.245	cold	4.008	2.6	2.6
5	cold	1.917	3.2	3.422	cold	4.012	2.1	2.3
6	min	1.871	4.9	4.923	min	3.837	5.6	5.9
7	min	1.875	4.9	4.945	min	3.904	5.7	5.9
8	min	1.869	4.9	4.879	min	3.855	5.9	6.1
9	min	1.871	4.9	4.923	min	3.87	6.1	6.2
10	min	1.863	4.9	4.923	mid	3.794	6.6	6.7
11	mid	1.861	5.7	5.762	mid	3.845	6.5	6.6
12	mid	1.863	5.7	5.784	mid	3.815	6.6	6.9
13	mid	1.857	5.7	5.828	mid	3.879	6.7	6.9
14	mid	1.853	5.7	5.784	max250	3.817	7.4	7.6
15	mid	1.859	NA	NA	max251	3.81	7.2	7.4
16	max250	1.811	6.3	6.424	max252	3.804	7.3	7.6
17	max251	1.787	6.3	6.446	max253	3.814	7.2	7.6
18	max252	1.819	6.3	6.623	max254	3.817	7.1	7.2
19	max253	1.806	6.4	6.534	exp	3.545	8.8	9.4
20	max254	1.805	6.4	6.446	exp	3.556	9	9.6
21	exp	1.651	6.5	6.578	exp	3.546	8.9	9.8
22	exp	1.681	6.6	6.755	exp	3.531	9.9	9.5
23	exp	1.682	6.9	6.865				
24	exp	1.686	6.8	6.843				
25	exp	1.646	6.6	6.623				

In the design of experiments, the parameter "current intensity", which controlled the intensity of weld currents, was purposely varied to produce different weld nugget size and thickness. "cold" represented the situation when the nugget size was below the requirement. “min”, “mid” and “max250” created three situations when the nugget size varied but stayed within the acceptable range. “exp” represented overly strong weld current that produced material expulsion during welding.

Each weld investigated was associated with an in-situ thermal video of the nugget formulation and a record of measurements. Either the video or the measurement record was saved to file that was named with experiment parameter and sample ID. In data processing, the in-situ video and the measurements were matched based on experiment parameter and sample ID to ensure correct analysis. For the weld whose measurement record was incomplete, i.e., with “NA” values, the sample was removed from the analysis.