



LAWRENCE  
LIVERMORE  
NATIONAL  
LABORATORY

# Semi-supervised Learning for Selective Laser Melting Monitoring via In-situ Videos

B. Yuan, B. Giera, G. Guss, S. McMains

July 2, 2018

IEEE Winter Conf. on Applications of Computer Vision  
Waikoloa Village, HI, United States  
January 7, 2019 through January 11, 2019

# Semi-Supervised Convolutional Neural Networks for In-Situ Video Monitoring of Selective Laser Melting

Bodi Yuan<sup>1</sup>, Brian Giera<sup>2</sup>, Gabe Guss<sup>2</sup>, Manyalibo Matthews<sup>2</sup>, and Sara McMains<sup>1</sup>

<sup>1</sup>UC Berkeley

<sup>2</sup>Lawrence Livermore National Laboratory

## Abstract

*Selective Laser Melting (SLM) is a metal additive manufacturing technique. The lack of SLM process repeatability is a barrier for industrial progression. SLM product quality is hard to control, even when using fixed system settings. Thus SLM could benefit from a monitoring system that provides quality assessments in real-time. Since there is no publicly available SLM dataset, we ran experiments to collect over one thousand SLM videos, measured the physical output via height map images, and applied a proposed image processing algorithm to them to produce a dataset for semi-supervised learning. Then we trained convolutional neural networks (CNNs) to recognize desired quality metrics from videos. Experimental results demonstrate the effectiveness of our proposed monitoring approach and also show that the semi-supervised model can mitigate the time and expense of labeling an entire SLM dataset.*

## 1. Introduction

Selective Laser Melting (SLM, or alternatively laser powder bed fusion) is an advanced additive manufacturing technique that uses a high power laser to melt metal powders layer-by-layer to produce an object with any desired shape. One challenge of SLM is that the quality of SLM products is sensitive to many factors, such as laser characteristics (power, speed, spot size, etc.), metal powder characteristics (size, purity, thickness, etc.), and build environment (ambient temperature, oxygen level, etc.) [38]. Given the many, often inter-related, physical and system variables, it can be challenging to choose the best SLM operating parameters and adapt them to changing conditions to produce precise products repeatably. We envision that a real-time SLM monitoring system could provide helpful feedback for

identifying defects, adjusting laser parameters, and compensating for variability in the powder and build environment.

As a step towards providing such a system, we propose to take in-situ videos taken during the SLM build process to use as input data points and train convolutional neural networks (CNNs) to recognize desired process metrics from those videos in real time. In previous work, we described an SLM experimental setup [40] with which we collected video while creating single linear SLM track welds. We labeled these SLM tracks with metrics of interest and used supervised learning to train regression and classification CNN models to predict several metrics from real-time videos.

Generally speaking, to train a well-performing CNN, the prerequisite is to have a sufficiently large amount of well-labeled data. However, for monitoring SLM (and other additive manufacturing) systems, it is challenging to collect labeled video data, though relatively easy to collect unlabeled data. Therefore, in this paper, we explore using a semi-supervised learning technique to reduce the effect of having relatively little labeled data. We also describe the details of our image processing algorithm that we use to label (a portion of) our data using structured light measurements. We train a semi-supervised CNN model based on the temporal ensemble method [25] with a small amount of labeled data and a large amount of unlabeled data. We evaluate our trained model on a labeled test dataset, using coefficient of determination for evaluating regression and accuracy for evaluating classification; in all cases the addition of unlabeled data improved performance.

We collect 1200 SLM in-situ videos, and measure and label 700 of them. The labeled data is used for the test set and labeled portion of the semi-supervised training dataset. To obtain labels for those in-situ videos, we take a height map measurement for each laser printed track using a Keyence VR3000 structured light microscope. In this paper, we describe details of our height map image processing algo-

<sup>1</sup>{bodiuyan, mcmain}@berkeley.edu, Berkeley, CA

<sup>2</sup>{giera1, guss2, matthews11}@llnl.gov, Livermore, CA

rithm that is used for automatically labeling portions of our video data corresponding to each separate track. Our image processing algorithm can measure the laser printed track metrics in terms of average width, continuity, and other surface map derived track properties. To recognize these track metrics for SLM in-situ monitoring, we design a semi-supervised in-situ video monitoring framework with convolutional neural networks, which does not require a significant amount of labeled data to train. Note that the data we collect and interpret can also be used for monitoring other metrics, such as track etching, roughness, etc. Our overall framework is easily transferable for monitoring such other metrics as long as a portion of the data is labeled for the metric. Furthermore, our approach is not limited to track width regression or continuity classification. To our knowledge, we are the first to successfully apply semi-supervised learning to SLM monitoring. Our experimental results demonstrate that a semi-supervised approach is promising and easy-to-implement for SLM monitoring.

Overall, in this paper our contributions are:

- We design and implement an entire semi-supervised framework for SLM monitoring, which includes data collection, partial data labeling, CNN model training, and results verification.
- We collect a dataset comprised of SLM in-situ videos for semi-supervised machine learning.
- We describe details of our height map image processing algorithm for labeling.
- We successfully apply semi-supervised CNNs to SLM in-situ monitoring.

## 2. Related Work

Our work is interdisciplinary and relates to: Selective Laser Melting, additive manufacturing, image processing, deep learning, convolutional neural networks, and semi-supervised learning.

### 2.1. Selective Laser Melting (SLM)

SLM is an additive manufacturing technology that shows promise to revolutionize certain aspects of the fabrication of metal parts. However, the physical process of SLM is complex, making it hard to control product properties by adjusting the system settings [12, 21]. Regan et al. [32] focuses on correlations between build environment and product quality, but the results only describe the rough trend of the correlations rather than a clear mathematical relation.

SLM product quality is affected by a variety of system build parameters. Many researchers identify the key input variables and metrics to analyze the quality of finished products. Spears, et al. [38] state that there are over 50 different SLM process input variables (grouped into four categories:

laser and scanning parameters, powder material properties, powder bed properties and recoat parameters, and build environment parameters) that affect the quality of the finished product. Given so many input variables, it is not only hard to analyze the impact of each variable, it is also hard to run experiments to test each variable independently. Berumen et al. [4] argue that there are five determining areas of the building process: powder quality, temperature management, process gas atmosphere, melt pool behavior, and documentation process. This variety also suggests the complexity of SLM product quality control. Kusuma [24] studies only the effect of laser power, scan speed, and laser energy density, showing that even if only these three variables change, the relation between the finished product metrics and the studied variables is complex and not easy to predict. Slotwinski [37] et al. focus on the effect of powder properties and study two different powder materials: stainless steel and cobalt-chrome. Their experimental results show that a variety of characteristics of powder properties, such as powder size, shape, density, crystallographic form, etc., have effects on the finished product quality.

Moreover, with improper system build parameters, SLM finished products might have lots of different kinds of defects [1, 5, 14, 20, 31] or unsatisfied product properties [27]. These issues can cause serious problems when the product is being used.

These works collectively suggest the difficulty of SLM product quality control. Therefore, addressing these issues is crucial to improve the quality of an SLM product.

### 2.2. SLM monitoring

Although physics-based models offer deep insights into optimized parameter selection and underlying mechanisms of SLM, the required computational expense limits their application for real-time uses. This suggests the critical necessity for in-situ SLM monitoring.

SLM monitoring methods typically fall into two categories: in-situ that occur in real-time, or ex-situ [30, 2, 13] that provide precise, automatic, robust quality detection and measurement from off-line data in certain circumstances. However, an unavoidable problem of ex-situ methods is that they require an entire part to be fully built before it is assessed for quality. Even if there is an obvious defect that arises during the build process, an ex-situ method inherently cannot detect it in time to correct it. Therefore, compared to in-situ methods, ex-situ methods are not only slow, but also incur additional time and material losses due to failed builds. For these reasons, we focus only on in-situ methods.

There are several different strategies to monitor the SLM process in real time. Grasso et al. [15] gives four major categories for in-situ monitoring depending on the feature of interest: (i) the powder bed, (ii) the layer slice, (iii) the track along the scan path, and (iv) the melt pool.

Li et al. [28] proposed a stereo vision based method for powder bed monitoring, which can calculate the powder bed 3D surface topology from two aligned cameras. Compared to a single camera monitoring system, Li's method will be more computationally expensive and there might be 3D reconstruction errors when calibrating two cameras. Erler et al. [10] proposed monitoring the thickness of the powder bed to recognize structural defects. This approach requires an expensive surface height sensor that can measure surface height in-situ, and the height map generation is slow. Scime, et. al. [36] leverages CNN architectures, as we do here, to identify the location of possible powder spreading anomalies for a variety of material systems.

For layer slice monitoring, Foster, et al. [11] proposed a layer-wise powder bed monitoring method using a single camera and regular color images. It takes and analyzes images for each powder layer before the layer building process. Kleszczynski, et al. [22] proposed using high resolution gray-scale images to recognize unsatisfied material properties of the entire product in the build process. However, both methods can only provide high-level monitoring rather than detailed information during the SLM process.

Track scanning methods [19, 3, 9] take temperature data as input signals. However, the common problems of this approach are that a temperature sensor's sensitivity is low and the data is noisy, resulting in imprecise monitoring results.

The predominant approach is in-situ melt pool monitoring, which is the method we use in this paper. Lott et al. [29] designed an optical system for in-situ melt pool monitoring, but without experimental verification. Craeghs [8] proposed analyzing the measured melt pool image data to find the position of the frame with maximum average intensity, which may correspond to overheating. Clijsters et al. [7] proposed using high-speed optical sensors to record the intensity of melt pool, and recognize the overheating of the melt pool and detect pores position. Repossini et al. [35] focuses on a specific feature of the melt pool, spatter analysis, and develops spatter-related descriptors to classify different energy density conditions. Grasso et al. [17] sets up a monitoring system similar to ours, which also includes in-situ monitoring and ex-situ labeling methods, and then uses conventional image processing methods and a statistical learning technique, Principle Component Analysis (PCA). Grasso et al. [16] integrates the acquisition of infrared images with conventional image processing methods. However, all of these approaches have many experimental parameters whose values are likely tied to their specific system and dataset and may not be transferable to other tasks and systems.

### 2.3. Convolutional neural networks

In recent years, CNNs [26, 23] have shown exceptional performance on many image and video recognition, seg-

mentation, and classification problems. In addition to improved performance compared to conventional computer vision methods, CNNs also are convenient because relevant image features are automatically extracted by the model itself [41]. This allows us to address several different problems from the same dataset with the same CNN architecture just by training with different labels, whereas conventional computer vision methods typically require significant manual oversight to extract different features and tune algorithm parameters. In this paper, we demonstrate using CNNs with the same architecture to perform both regression and classification.

### 2.4. Semi-supervised learning

Current convolutional neural networks rely critically on large labeled training datasets. Several popular examples benefit from vast publicly-accessible labeled datasets. Collecting data is generally relatively cheap and easy; however, labeling data requires time-consuming human labor [33]. The same is true for our problem: collecting in-situ videos is automated and the laborious part has traditionally been labeling video data. Therefore, semi-supervised learning [42, 6] is a promising approach since it reduces the need for large labeled datasets while not suffering from the level of performance loss expected from unsupervised learning. In semi-supervised learning, the dataset can still be huge but only a very small portion of the dataset might be labeled. This will significantly reduce the expense of the labeling task and increase the generalizability of deep learning for practical use. We will discuss the most closely related semi-supervised works to our approach, ladder networks [34] and the temporal ensemble method [25].

The ladder network [34] is a powerful semi-supervised neural network for image classification tasks. The ladder network makes use of both supervised learning with the labeled portion of data and unsupervised learning with unlabeled data simultaneously. The architecture of ladder networks is an autoencoder [18] applied to every layer, not just the inputs. For the labeled data, the ladder network trains in a standard supervised fashion. For the unlabeled data, it trains with an autoencoder in an unsupervised fashion to preserve all the details needed for reconstructing the image. Laine et al., inspired by ladder networks, proposed the temporal ensemble method [25], which is the approach we adapt for our application. The temporal ensemble method can be seen as a simplified version of ladder networks, without the autoencoder. Instead, it makes two passes over the same input with different data augmentation and different dropout [39]. The supervised loss is still the standard cross entropy loss, and the unsupervised loss is the mean square difference between two passes' output layers. The unsupervised portion helps the model to extract more important features and reduces the effect of overfitting to the small

portion of training data that is labeled. It works like an autoencoder but is much more efficient and robust.

### 3. Proposed Framework for SLM Monitoring

Our proposed framework for SLM monitoring includes SLM data collection, partial data labeling, semi-supervised training, and result verification. We address data labeling by taking height map measurements with structured light, followed by image processing. However, alternate labeling approaches using manual measurements could be used by researchers without access to height maps.

#### 3.1. SLM dataset collection

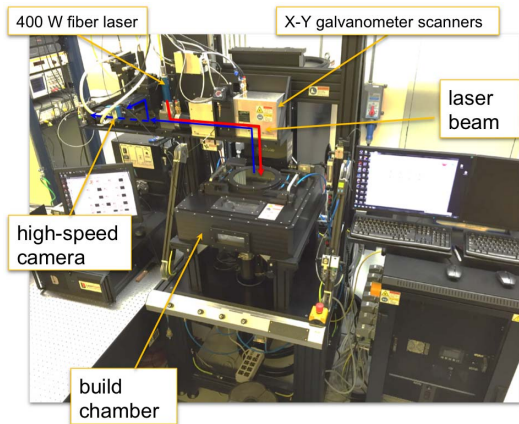


Figure 1: SLM machine and data collection system

For our SLM experiments, we use 316L stainless steel powder. The powder is spread to  $\sim 50 \mu m$  thickness on top of a 180 mm diameter stainless-steel plate (only a single layer is printed). The laser spot size is a fixed  $106 \mu m$  diameter. We vary the laser power between 50-375 W and the laser scan speed (the speed of moving the laser spot) between 100-400 mm/s.

We mount a high-speed 1 kHz video camera on the SLM machine such that its focal point moves together with the laser spot to collect in-situ video data. We printed 1200 individual 5 mm tracks over the 180 mm plates (shown in Figure 2) and collected the corresponding in-situ videos, while the laser scans the tracks. The experimental machine is shown in Figure 1.

The video for each individual track has 12-50 frames depending on the scan speed, each frame of size  $256 \times 256$ . Due to the video collection software, a portion of the video records the process of moving the laser toward and away from the desired location on the powder bed, while the laser is off. For this work, in-situ monitoring is only relevant when the laser is on. To extract the relevant sequence of frames within each video, we first compare the

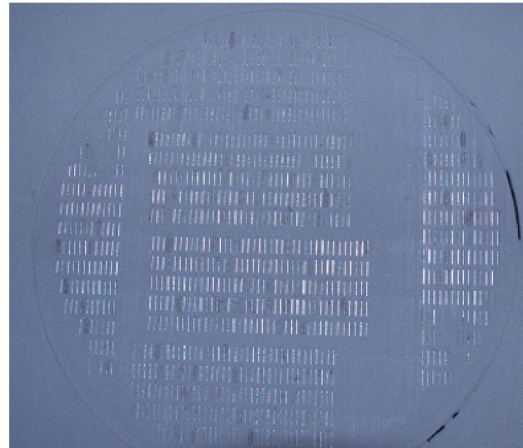


Figure 2: Examples of experimental tracks on one of the two build plates

average pixel intensity of an  $80 \times 80$  region of interest (ROI) in the center of the frame,  $\bar{\epsilon}_{80 \times 80}$ , against the average intensity of pixels not in that region,  $\bar{\zeta}_{80 \times 80}$ . For each video, we then consider the sub-selection of frames where  $\bar{\epsilon}_{80 \times 80} > \bar{\zeta}_{80 \times 80}$ , which occurs when the laser is on and, in rare cases, when reflections from the plate or previously-welded tracks appear sporadically in the ROI. From this sub-selection, we identify the frames with the laser on as the longest chronological sequence of frames. We center-crop each frame to  $80 \times 80$  and take the 10 continuous middle frames (Figure 4 (a)) from each such sequence (using 10 frames to make sure all frames contain the laser spot). Then we concatenate these 10 frames as a single data point as shown in Figure 4 (b). Each video data point is labeled with the measured average width value and the continuity of the corresponding laser printed track, using the procedure described next.

#### 3.2. Height map measurement

To label a portion of our videos, we performed ex-situ measurements of corresponding laser printed tracks on the top of the steel plate. After all tracks were completed, unmelted powder was removed so that we could measure just the tracks using a Keyence VR3000 3D microscope, which can precisely measure the surface height as shown in Figure 3 (b) via a structured light scan.

A problem with this ex-situ measurement is that it relies on an expensive structured light microscope that is not yet widely available. Other techniques, e.g. traditional confocal microscopy, are too time-consuming to generate sufficiently large area height maps for meaningful ex-situ assessments. Thus, we explore the effectiveness of a semi-supervised method for SLM monitoring, which does not require as much labeled data. Researchers without direct ac-

cess to such a microscope could send a single plate of track experiments to a facility that had such an instrument for labeling, or could even manually label a small portion of their data to use our proposed semi-supervised framework.

### 3.3. Image processing for height maps

From the raw height map measurement, we propose a two-round image processing algorithm that can automatically calculate metrics for each laser track for generating labels for our video data.

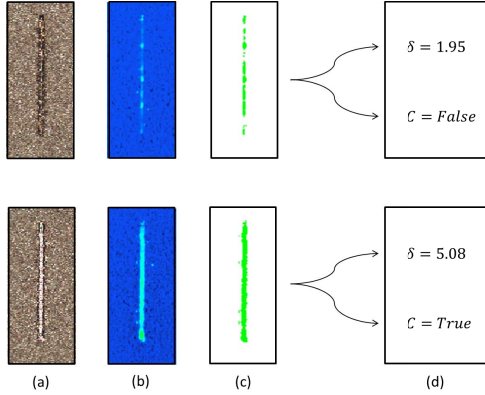


Figure 3: *Example tracks: (a) regular electro-optical images, (b) height maps measured by structured light, (c) segmentation results by proposed image processing algorithm, (d) labels from calculated track metrics*

Since we know the physical spacing between tracks, we subdivide the height map into patches that contain a single vertical track. Our algorithm takes each patch as an input and provides pixel level classification results as outputs (Track/Non-track). Knowing the pixel level information, we can then easily calculate a variety of track metrics. The key idea here is to remove the background and then analyze each track. Since the background is noisy in terms of height, and there are global variations in the height of the underlying plate (it is not completely flat), we propose a two-round filtering method that is robust to background noise and able to provide a clean segmentation result as shown in Figure 3 (c). The details of our algorithm are summarized below; pseudo code is given in Algorithm 1.

- (a) Calculate the average height  $\bar{h}_1$  of the height map patch  $\mathcal{H}$ . This will generally be just a bit higher than the height of the background plate since track pixels are at most 10% in our experiments.
- (b) Re-calculate the average height  $\bar{h}_2$  of pixels with height in the range  $(\bar{h}_1 - \tau_1, \bar{h}_1 + \tau_1)$ . These are likely to be background pixels rather than track pixels.
- (c) Binarize the height map  $\mathcal{H}$  as  $\mathcal{B}$  with the threshold  $\bar{h}_2 + \tau_2$ , in preparation for final denoising.

- (d) Calculate the summation  $S_j$  of each column  $j$  in the binary map  $\mathcal{B}$ .
- (e) Set all the pixels in column  $j$  to the “non-track” (zero) value if  $S_j$  is less than a noise threshold  $T$  expressed as a percentage of the desired track length.
- (f) Return the output binary map  $\mathcal{B}$ , which has the same size as the original height map patch  $\mathcal{H}$ , with the “Track/Non-track pixel” classification result on each pixel.

---

#### Algorithm 1 Height map segmentation

---

```

1:  $\mathcal{H} \leftarrow$  input height map patch
2:  $H, W \leftarrow$  vertical and horizontal sizes of the height map patch  $\mathcal{H}$ 
3:  $\tau_1, \tau_2, T \leftarrow$  constants for threshold
4: ## Round 1:
5:  $h_1 \leftarrow \sum_i^H \sum_j^W \mathcal{H}_{ij}$ 
6:  $\bar{h}_1 \leftarrow h_1 / (\sum_i^H \sum_j^W 1)$ 
7: ## Round 2:
8:  $h_2 \leftarrow \sum_i^H \sum_j^W \mathcal{H}_{i,j} \cdot (\bar{h}_1 - \tau_1 < \mathcal{H}_{i,j} < \bar{h}_1 + \tau_1)$ 
9:  $\bar{h}_2 \leftarrow h_2 / (\sum_i^H \sum_j^W 1 \cdot (\bar{h}_1 - \tau_1 < \mathcal{H}_{i,j} < \bar{h}_1 + \tau_1))$ 
10: ## Post-filtering
11:  $\mathcal{B} \leftarrow$  binarize  $\mathcal{H}$  with threshold  $\bar{h}_2 + \tau_2$ 
12: for  $j \leftarrow 1$  to  $W$  do
13:    $S_j \leftarrow \sum_{i=1}^H \mathcal{B}_{ij}$ 
14:   if  $S_j < T$  then
15:     for  $i \leftarrow 1$  to  $H$  do
16:        $\mathcal{B}_{ij} \leftarrow 0$ 
17:     end for
18:   end if
19: end for
20: return  $\mathcal{B}$ 

```

---

The appropriate values of the constants  $\tau_1, \tau_2$ , and  $T$  will depend upon the height of the track pixels relative to the background pixels, and on the variability. We use  $\tau_1 = 0.01$  and  $\tau_2 = 0.01$  (all heights are in units of pixels) for our setup, in which the typical standard deviation of the background height within a patch is around 0.002, and the difference between track pixel height and background pixel height is over 0.2 (100 times the standard deviation of the background). In other words, these constants are set at about 5% of the height of the tracks above the background, which is also equal to about 5 standard deviations of the background height. We set  $T = 5\%$  of the track length, which is 10 pixels out of 200 in our experimental

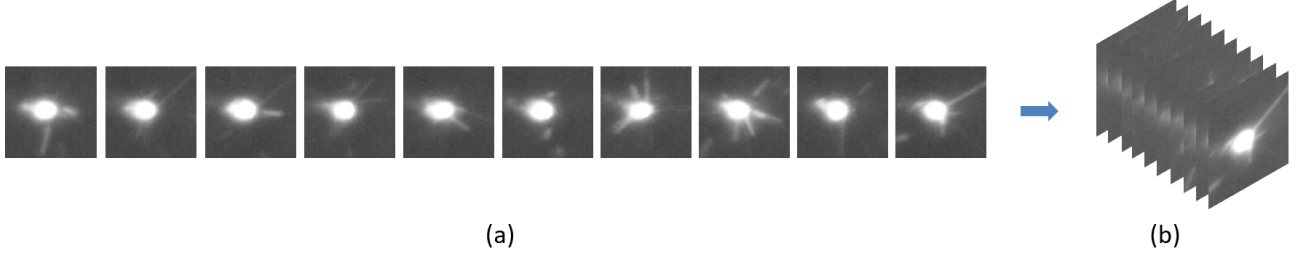


Figure 4: Example of in-situ video data (a) ten continuous  $80 \times 80$  frames of in-situ videos (b) single data point from concatenation of ten continuous frames

setup. Any automated labeling process must rely upon a relatively precise machining and polishing process for the background steel plate compared to the track height; with the two-orders-of magnitude height:stdev ratio in our setup, there is less sensitivity to the choice of these constants.

Note that sometimes there are pixels with heights lower than the background height. This is caused by the laser melting not only the powder but also the powder bed, which results in etching, where the surface height will be lower than that of the underlying plate. For the current work, these are classified as non-track pixels; identification of etching is left to future research.

Having identified the track pixels within the binary map  $\mathcal{B}$ , it is straightforward to calculate the relevant track metrics (Figure 3 (d)), such as the length  $l$ , the average width  $\delta$ , both in pixel units, and the continuity  $\mathcal{C}$  of the track. The length  $l$  is given by:

$$l = I' - I^*,$$

where  $I^*$  is the index of the first row that has track pixels, and  $I'$  is the index of the last row that has track pixels.

For the CNNs, we treat the average width  $\delta$  as a regression problem, so their metrics are given as positive real numbers:

$$\delta = \frac{1}{l} \sum_{i=I^*}^{I'} \sum_{j=1}^W \mathcal{B}_{ij}.$$

As for the continuity  $\mathcal{C}$ , we define it as a Boolean and treat it as a binary classification problem. If there exists any row between  $I^*$  and  $I'$  such that the row has no track pixel, then this track has a break.

Although we can measure other laser track metrics from the height map, e.g. roughness, average height, etc., we only explore width and continuity for this work.

### 3.4. Semi-supervised learning for video regression

Building on previous research on semi-supervised learning algorithms, we make use of the  $\Pi$  model from the Temporal Ensemble method [25] for our CNN model. The  $\Pi$  model has both a supervised training flow and an unsupervised training flow. It is designed for a semi-supervised classification problem. But in our work, rather than only

a semi-supervised classification task, we also extend it to a regression problem (predicting track width).

For our problem, the training data  $\mathcal{X}$  has  $N$  inputs, only  $M$  of which have a regression target value (labels  $\mathcal{Y}$ ).  $L$  is the set of labeled data indices; for each  $i \in L$ ,  $x_i$  has a positive real number label  $y_i$ , where  $|L| = |\mathcal{Y}| = M$ .  $\theta$  is the set of weights for our convolutional neural networks and the regression value of each data point  $x_i$  is  $\tilde{z}_i = \theta(x_i)$ . During training, data augmentation  $\mathcal{A}(x)$  and dropout  $\mathcal{D}(x)$  are used, where  $\tilde{z}_i = \mathcal{D}(\theta(\mathcal{A}(x_i)))$ .

Similar to the  $\Pi$  model, we feed each training data point (video)  $x_i$ , whether labeled or unlabeled, into the neural network twice, evaluating it with different data augmentation  $\mathcal{A}_1(x)$ ,  $\mathcal{A}_2(x)$ , and different dropout  $\mathcal{D}_1(x)$ ,  $\mathcal{D}_2(x)$ , which will produce different regression outputs  $\tilde{z}_{i1}$  and  $\tilde{z}_{i2}$ , where

$$\tilde{z}_{i1} = \mathcal{D}_1(\theta(\mathcal{A}_1(x_i))), \quad \tilde{z}_{i2} = \mathcal{D}_2(\theta(\mathcal{A}_2(x_i))).$$

The  $\Pi$  model has two passes but only calculates loss from one pass. To make the training more efficient, we instead make the slight change of using both passes' output to calculate the supervised loss  $\mathcal{L}_s$  symmetrically (see Figure 5). This will take hardly any additional time, since we have to run two passes whether we use the symmetric loss or the original loss, but will work like doubling the training batch size, making our training more efficient overall (it was almost twice as fast in our tests). When we address a regression problem instead of a classification problem, we also make the change of using mean squared difference for the loss instead of cross entropy loss. Specifically, for regression our supervised loss  $\mathcal{L}_s$  for each “mini-batch” subset  $B$  of data indices is calculated by:

$$\mathcal{L}_s = \frac{1}{|B|} \sum_{i \in (B \cap L)} (\|y_i - \tilde{z}_{i1}\|^2 + \|y_i - \tilde{z}_{i2}\|^2).$$

The unsupervised loss  $\mathcal{L}_u$  is still calculated by mean squared difference:

$$\mathcal{L}_u = \frac{1}{|B|} \sum_{i \in B} \|\tilde{z}_{i1} - \tilde{z}_{i2}\|^2.$$



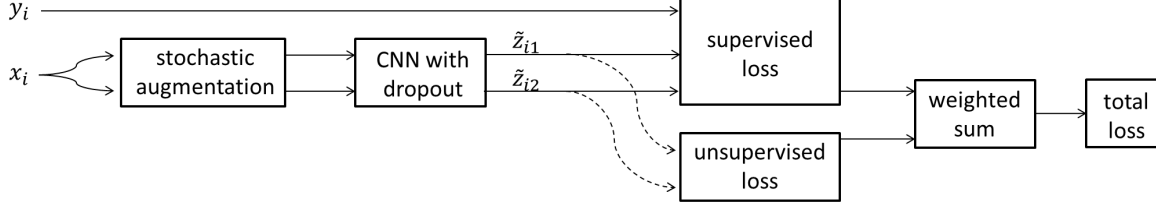


Figure 5: *Semi-supervised CNN architecture*

Finally, the total loss  $\mathcal{L}$  is the weighted summation of the supervised loss and the unsupervised loss:

$$\mathcal{L} = \mathcal{L}_s + w\mathcal{L}_u,$$

where  $w$  is the weight for the unsupervised loss.

### 3.5. Semi-supervised learning for classification

We use the same CNN architecture for classification as for regression, but although we use the same symmetric supervised loss, for the classification supervised loss we use cross-entropy loss rather than squared difference:

$$\mathcal{L}_s = -\frac{1}{|B|} \sum_{i \in (B \cap L)} y_i \log\left(\frac{\tilde{z}_{i1}\tilde{z}_{i2}}{\tilde{z}_{i1}^*\tilde{z}_{i2}^*}\right) + \log(\tilde{z}_{i1}^*\tilde{z}_{i2}^*),$$

where  $\tilde{z}_{i1}^* = 1 - \tilde{z}_{i1}$  and  $\tilde{z}_{i2}^* = 1 - \tilde{z}_{i2}$ .

We use the same unsupervised loss as for the regression problems, as described in the previous section.

## 4. Experimental Results and Discussion

We test the semi-supervised learning approach on our collected dataset, consisting of 1000 training data points with 500 labeled and 500 unlabeled, and 200 test data points. Labeled data has two separate labels, average track width (positive real number) and track continuity (Boolean value). We train our model with 500, 100, 50, and 10 labeled data points in both semi-supervised and fully supervised fashion; for all instances of unsupervised learning we also use the entire set of 500 unlabeled videos as additional training data. The 10/50/100 labeled data points are chosen randomly from the full labeled dataset, with the same subset used to train the fully supervised and semi-supervised models we compare. We ran all experiments five times, and report the average performance and the standard deviation. The model architectures are the same (when training the fully supervised model, we just use the supervised loss).

We use the same CNN architecture to address both regression and classification tasks. The CNN has 3 conv-layers and 3 dense layers. The conv-layers have 32 feature maps of  $3 \times 3$  kernels, 64 feature maps of  $3 \times 3$  kernels, and 64 feature maps of  $3 \times 3$  kernels respectively. The first two conv-layers, CONV-1 and CONV-2, are each followed by ReLU activation and a  $2 \times 2$  mean-pooling layer. The CONV-3 layer is followed by ReLU activation and connected to the first dense layer, FC-1, with 1000 hidden units,

followed by a dropout layer with 0.5 dropout rate. FC-2 and FC-3 (each followed by sigmoid activation), with 200 and 20 hidden units respectively, follow the dropout layer. FC-3 is fully connected with the final output layer of a single unit, which corresponds to the relevant track metric as output. We set batch size to 20, and max training epoch number to 300. We choose the Adam optimizer with a learning rate of  $10^{-4}$  for the regression task, and  $10^{-5}$  for the classification task.

Similar to the original  $\Pi$  model's implementation, we ramped up the unsupervised loss component weight  $w$  during the first 50 epochs with a Gaussian ramp-up curve

$$w = W_{max}e^{-5(1-T)},$$

where  $T$  varies linearly from 0 to 1 during the ramp-up period and  $W_{max}$  is set to 100 for all the experiments.

Each input data video is 10 grayscale frames of  $80 \times 80$  pixels whose values are in the range  $[0, 1]$ . Standard data augmentation (applying translation, mirroring, brightness and contrast) is applied to all the training data. We crop the frames to  $64 \times 64$ , with pixels shifted vertically and horizontally, by an amount chosen from a uniform random distribution from  $[-8, +8]$ . We randomly apply horizontal and/or vertical flips. For robustness to changes in lighting conditions (which may even vary from run to run depending on experimental settings), as well as to improve generalizability to other SLM machines, we also performed random brightness and contrast data augmentation for each input video. For brightness, we add a constant chosen randomly from a uniform distribution in the range  $[-0.5, 0.5]$  to all pixel values; for contrast, we similarly choose a uniform random factor  $c$  in the range  $[0.2, 1.8]$ , using it to adjust the contrast  $\mathcal{I}_{aug} = c * (\mathcal{I} - \bar{\mathcal{I}}) + \bar{\mathcal{I}}$ , where  $\mathcal{I}$  is the original image,  $\bar{\mathcal{I}}$  is its mean pixel value, and  $\mathcal{I}_{aug}$  is the image after augmentation. After these adjustments, we clamp pixel values lower than 0 or larger than 1 to the range limits.

Table 1 shows our average track width regression experimental results. We report mean and standard deviation of  $R^2$ , the coefficient of determination, for randomly selected subsets of 10, 50, 100, and 500 labeled training videos measured on  $N$  test data points (in our case,  $N = 200$ ).

Comparing the semi-supervised and fully supervised results, we see that the semi-supervised approach often performs much better than the fully supervised approach when



Number of labeled videos	supervised-only	semi-supervised
500	0.90 (0.02)	0.92 (0.01)
100	0.74 (0.04)	0.81 (0.02)
50	0.56 (0.05)	0.68 (0.03)
10	0.15 (0.07)	0.13 (0.06)

Table 1: Mean  $R^2$  score for average width regression; standard deviations in parentheses

only a small portion of the data is labeled, which demonstrates the viability and data-efficiency of a semi-supervised SLM monitoring system via in-situ videos.

Especially when the number of labeled videos is 50 or 100, the semi-supervised approach significantly improves the performance over supervised, mitigating the disadvantage of having fewer labeled training videos. With only 100 labeled videos, our model can still achieve an  $R^2$  score of 0.81, a 0.07  $R^2$  score improvement from fully supervised training. Compared to supervised training with five times as many labeled videos, the performance is not much worse. Even with only 50 labeled training videos, we still push the  $R^2$  score from 0.56 to 0.68, which is fairly high for this regression problem. These results imply that people may be able to get away with manually labeling just a small portion of data and training a CNN model in a semi-supervised fashion to address the problem of lack of labels. Moreover, this also demonstrates that semi-supervised learning with temporal ensembles can not only work well for classification, but also work well for regression problems.

For the experiments training with 500 labeled videos, the semi-supervised training result is still better than the fully supervised result, but only slightly. For this dataset, 500 labeled videos is already 50% of the entire training set and it is already enough for supervised training to learn useful features. Thus, the semi-supervised approach does not provide too much improvement. Our semi-supervised approach targets problem when we only have a relatively small amount of labeled data.

However, when we have just 10 labeled training videos, the performance of both semi-supervised and fully supervised models is unsurprisingly poor. With such little labeled data, the training set will be overfit and the model is likely only going to learn very basic and inaccurate features. The  $R^2$  score is close to 0, implying that the model prediction is almost a constant prediction.

Table 2 similarly gives the mean and standard deviation of prediction accuracy of our binary classification for continuity, also evaluated on the same 200 test data points. About one fifth of the data is “FALSE” for continuity, both for labeled and unlabeled.

The semi-supervised algorithm still performed well for this binary classification problem. It is a remarkable fact that the semi-supervised performance when we have only 100 labeled videos even outperforms the fully supervised

Number of labeled videos	supervised-only	semi-supervised
500	92.2% (0.2)	93.8% (0.2)
100	86.3% (1.1)	92.5% (0.9)
50	85.2% (2.2)	89.8% (1.7)
10	69.1% (4.0)	68.8% (4.5)

Table 2: Mean accuracy for continuity classification; standard deviations in parentheses

performance with 500 labeled videos. Similarly, the semi-supervised performance when we have only 50 labeled videos outperforms the fully supervised one with 100 labeled videos. The semi-supervised approach is particularly advantageous when the amount of labeled data is small.

Overall, our semi-supervised approach performs better than the fully supervised approach no matter if we are solving a regression or classification problem.

The speed of our trained CNN is more than adequate for real-time monitoring. Measuring its inference time on 100 videos, the average inference time was 1.4ms, and the maximum was 2.9ms.

## 5. Conclusion

In our work, we ran a set of experiments to print laser tracks, collected over one thousand in-situ videos, labeled a portion of that data, trained semi-supervised CNN models, and verified our results on a test dataset. We demonstrate that this entire framework enables SLM in-situ monitoring to detect desired metrics in real time. Our framework and the entire approach should be transferable to other SLM systems. To make our approach work well on different systems, it is only required to re-collect video data, measure a small portion of track data and run the track pixel segmentation and classification algorithm, and re-train the model.

We focus on monitoring two characteristics of our experimental tracks, average width and continuity. The experimental results show that our approach recognizes these metrics very well, and also demonstrate that the semi-supervised method requires less labeled data for training.

Besides the metrics discussed in this paper, this approach can also be easily extended to predict a wider range of defects and metrics, such as etching detection, roughness regression, and even to AM technologies beyond SLM. In this work, we only did monitoring for separated SLM tracks with a single layer. Future work will address monitoring for SLM with contiguous tracks and multi-layer printing.

## 6. Acknowledgements

Thanks to the anonymous reviewers for their helpful feedback. This work was performed partially under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344, LLNL-PROC-754101.

## References

- [1] N. T. Aboulkhair, N. M. Everitt, I. Ashcroft, and C. Tuck. Reducing porosity in AlSi10Mg parts processed by selective laser melting. *Additive Manufacturing*, 1:77–86, 2014.
- [2] M. Aminzadeh and T. Kurfess. Vision-based inspection system for dimensional accuracy in powder-bed additive manufacturing. In *ASME 2016 11th International Manufacturing Science and Engineering Conference*, pages V002T04A042–V002T04A042. American Society of Mechanical Engineers, 2016.
- [3] F. Bayle and M. Doubenskaia. Selective laser melting process monitoring with high speed infra-red camera and pyrometer - art. no. 698505. 6985, 01 2008.
- [4] S. Berumen, F. Bechmann, S. Lindner, J.-P. Kruth, and T. Craeghs. Quality control of laser-and powder bed-based additive manufacturing (am) technologies. *Physics procedia*, 5:617–622, 2010.
- [5] N. P. Calta, J. Wang, A. M. Kiss, A. A. Martin, P. J. Depond, G. M. Guss, V. Thampy, A. Y. Fong, J. N. Weker, K. H. Stone, et al. An instrument for in situ time-resolved x-ray imaging and diffraction of laser powder bed fusion additive manufacturing processes. *Review of Scientific Instruments*, 89(5):055101, 2018.
- [6] O. Chapelle, B. Scholkopf, and A. Zien. Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]. *IEEE Transactions on Neural Networks*, 20(3):542–542, 2009.
- [7] S. Clijsters, T. Craeghs, S. Buls, K. Kempen, and J.-P. Kruth. In situ quality control of the selective laser melting process using a high-speed, real-time melt pool monitoring system. *The International Journal of Advanced Manufacturing Technology*, 75(5-8):1089–1101, 2014.
- [8] T. Craeghs, S. Clijsters, J.-P. Kruth, F. Bechmann, and M.-C. Ebert. Detection of process failures in layerwise laser melting with optical process monitoring. *Physics Procedia*, 39:753–759, 2012.
- [9] P. J. DePond, G. Guss, S. Ly, N. P. Calta, D. Deane, S. Khairallah, and M. J. Matthews. In situ measurements of layer roughness during laser powder bed fusion additive manufacturing using low coherence scanning interferometry. *Materials Design*, 154:347 – 359, 2018.
- [10] M. Erler, A. Streek, C. Schulze, and H. Exner. Novel machine and measurement concept for micro machining by selective laser sintering.
- [11] B. Foster, E. Reutzel, A. Nassar, B. Hall, S. Brown, and C. Dickman. Optical, layerwise monitoring of powder bed fusion. In *Solid Free. Fabr. Symp. Proc.*, pages 295–307, 2015.
- [12] M. M. Francois, A. Sun, W. E. King, N. J. Henson, D. Tourret, C. A. Bronkhorst, N. N. Carlson, C. K. Newman, T. S. Haut, J. Bakosi, et al. Modeling of additive manufacturing processes for metals: Challenges and opportunities. *Current Opinion in Solid State and Materials Science*, 21(LA-UR-16-24513), 2017.
- [13] A. Gebhardt, F.-M. Schmidt, J.-S. Hötter, W. Sokalla, and P. Sokalla. Additive manufacturing by selective laser melting the realizer desktop machine and its application for the dental industry. *Physics Procedia*, 5:543–549, 2010.
- [14] H. Gong, K. Rafi, H. Gu, G. J. Ram, T. Starr, and B. Stucker. Influence of defects on mechanical properties of ti–6al–4 v components produced by selective laser melting and electron beam melting. *Materials & Design*, 86:545–554, 2015.
- [15] M. Grasso and B. M. Colosimo. Process defects and in situ monitoring methods in metal powder bed fusion: a review. *Measurement Science and Technology*, 28(4):044005, 2017.
- [16] M. Grasso, A. Demir, B. Previtali, and B. Colosimo. In situ monitoring of selective laser melting of zinc powder via infrared imaging of the process plume. *Robotics and Computer-Integrated Manufacturing*, 49:229–239, 2018.
- [17] M. Grasso, V. Laguzza, Q. Semeraro, and B. M. Colosimo. In-process monitoring of selective laser melting: Spatial detection of defects via image data analysis. *Journal of Manufacturing Science and Engineering*, 139(5):051001, 2017.
- [18] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006.
- [19] J. A. Kanko, A. P. Sibley, and J. M. Fraser. In situ morphology-based defect detection of selective laser melting through inline coherent imaging. *Journal of Materials Processing Technology*, 231:488–500, 2016.
- [20] S. A. Khairallah, A. T. Anderson, A. Rubenchik, and W. E. King. Laser powder-bed fusion additive manufacturing: Physics of complex melt flow and formation mechanisms of pores, spatter, and denudation zones. *Acta Materialia*, 108:36 – 45, 2016.
- [21] W. E. King, A. T. Anderson, R. Ferencz, N. Hodge, C. Kamath, S. A. Khairallah, and A. M. Rubenchik. Laser powder bed fusion additive manufacturing of metals; physics, computational, and materials challenges. *Applied Physics Reviews*, 2(4):041304, 2015.
- [22] S. Kleszczynski, J. Zur Jacobsmühlen, J. Sehart, and G. Witt. Error detection in laser beam melting systems by high resolution imaging. In *Proceedings of the Twenty Third Annual International Solid Freeform Fabrication Symposium*, volume 2012, 2012.
- [23] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.
- [24] C. Kusuma. The effect of laser power and scan speed on melt pool characteristics of pure titanium and ti-6al-4v alloy for selective laser melting. 2016.
- [25] S. Laine and T. Aila. Temporal ensembling for semi-supervised learning. *arXiv preprint arXiv:1610.02242*, 2016.
- [26] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.
- [27] S. Leuders, M. Thöne, A. Riemer, T. Niendorf, T. Tröster, H. Richard, and H. Maier. On the mechanical behaviour of titanium alloy ti6v4 manufactured by selective laser melting: Fatigue resistance and crack growth performance. *International Journal of Fatigue*, 48:300–307, 2013.
- [28] Z. Li, X. Liu, S. Wen, P. He, K. Zhong, Q. Wei, Y. Shi, and S. Liu. In situ 3d monitoring of geometric signatures in the powder-bed-fusion additive manufacturing process via vision sensing methods. *Sensors*, 18(4), 2018.

- [29] P. Lott, H. Schleifenbaum, W. Meiners, K. Wissenbach, C. Hinke, and J. Bültmann. Design of an optical system for the in situ process monitoring of selective laser melting (slm). *Physics Procedia*, 12:683–690, 2011.
- [30] A. M. Luiz, L. P. Flávio, and E. A. Paulo. Automatic detection of surface defects on rolled steel using computer vision and artificial neural networks. In *IECON 2010-36th Annual Conference on IEEE Industrial Electronics Society*, pages 1081–1086. IEEE, 2010.
- [31] S. Ly, A. M. Rubenchik, S. A. Khairallah, G. Guss, and M. J. Matthews. Metal vapor micro-jet controls material redistribution in laser powder bed fusion additive manufacturing. *Scientific reports*, 7(1):4085, 2017.
- [32] P. ORegan, P. Prickett, R. Setchi, G. Hankins, and N. Jones. Metal based additive layer manufacturing: variations, correlations and process control. *Procedia Computer Science*, 96:216–224, 2016.
- [33] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*, 2016.
- [34] A. Rasmus, M. Berglund, M. Honkala, H. Valpola, and T. Raiko. Semi-supervised learning with ladder networks. In *Advances in Neural Information Processing Systems*, pages 3546–3554, 2015.
- [35] G. Repossini, V. Laguzza, M. Grasso, and B. M. Colosimo. On the use of spatter signature for in-situ monitoring of laser powder bed fusion. *Additive Manufacturing*, 16:35–48, 2017.
- [36] L. Scime and J. Beuth. A multi-scale convolutional neural network for autonomous anomaly detection and classification in a laser powder bed fusion additive manufacturing process. *Additive Manufacturing*, 24(5786):504–507, 2018.
- [37] J. A. Slotwinski, E. J. Garboczi, P. E. Stutzman, C. F. Ferraris, S. S. Watson, and M. A. Peltz. Characterization of metal powders used for additive manufacturing. *Journal of research of the National Institute of Standards and Technology*, 119:460, 2014.
- [38] T. G. Spears and S. A. Gold. In-process sensing in selective laser melting (slm) additive manufacturing. *Integrating Materials and Manufacturing Innovation*, 5(1):2, 2016.
- [39] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.
- [40] B. Yuan, G. M. Guss, A. C. Wilson, S. P. HauRiege, P. J. DePond, S. McMains, M. Matthews, and B. Giera. Machine learning based monitoring of laser powder bed fusion. *Advanced Materials Technology*, 2018.
- [41] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.
- [42] X. Zhu. Semi-supervised learning literature survey. Technical Report 1530, Computer Sciences, University of Wisconsin-Madison, 2005.