Computer Vision Task for LPBF Defect Detection

Yuye Tang, 2/8/2024

1. Background and Objective of this Study

Additive manufacturing (AM, aka. 3D printing) builds a complex geometry layer by layer from bottom up based on given 3D CAD models. Because of its versatility, easy customization, and less material waste, AM has attracted a lot of interests in many different industries, such as defense, aerospace, medical, consumer products and so on. Due to the complex thermal history during printing, the quality control of additively manufactured parts is very challenging. The printed part may fail because of poor material properties, material irregularities (e.g., inclusion, porosities), large distortion leading to out-of-spec dimensions, etc. Both in-process monitoring and non-destructive evaluation after built are promising techniques for screening parts.

I am interested in a specific AM process, Laser Powder Bed Fusion (LPBF), which utilizes laser as the energy source for fusing material. There are not many shared in-process monitoring datasets. One available dataset is from video recordings of selective laser sintering process of plastics [1]. This dataset has been used in two papers [1][2]. The first paper utilized the idea of transfer learning (VGG16, Xception convolutional neural network (CNN) models with pretrained weights from the ImageNet dataset), while the second uses off-the-shelf CNN ensemble models. The objective of this report is to realize image classification using transfer learning technique by following the parameter setup in reference [1], providing a framework for in-situ quality control of LPBF products.

2. Dataset Description

The dataset from reference [1] can be downloaded from https://data.mendeley.com/datasets/2yzjmp52fw/1. Two datasets are shared, data and data_balanced folders:

```
dataset/
└── data
└── data_balanced
    ├── train
    |   ├── DEF
    |   └── OK
    ├── validation
    |   ├── DEF
    |   └── OK
    └── test
        ├── DEF
        └── OK
```

- The original dataset (from data folder) was extracted from video recordings of an off-axis powder bed surface of SLS printing system. The dataset (containing 8514 images in total) was divided into two classes, OK (i.e., good quality, 7808 images) and DEF (i.e., poor quality, 706 images) (Figure 1). It should be noted that this dataset is imbalanced. The training accuracy will be dominated by the majority class. Hence, special metrics are required to evaluate imbalanced classification tasks.

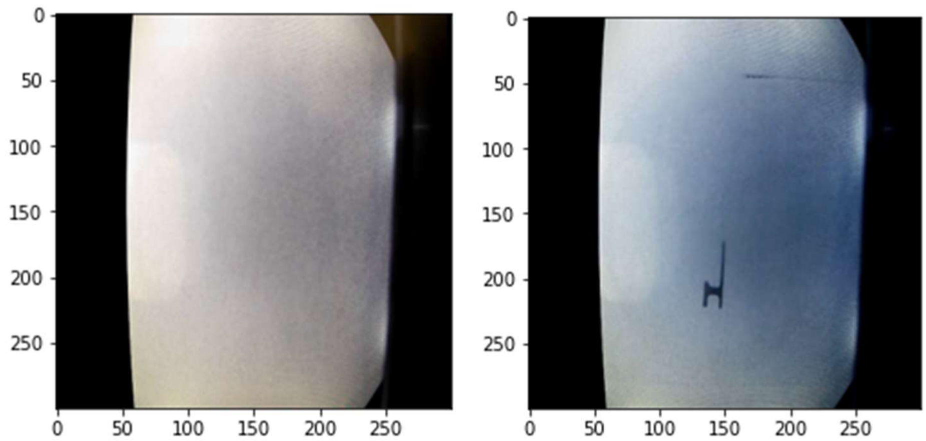

Figure 1 Examples from original dataset OK image (on the left) and DEF image (on the right)

- The data_balanced folder has three subfolders, training, validation, and test. The images in each subfolder were randomly selected from the data folder. Each subfolder has images for the corresponding classes, OK and DEF (Figure 2). The dataset has already been balanced. The training subfolder contains 1000 images in each class, the validation subgroup contains 500 images in each class and the test subgroup also contains 500 images in each class. The detailed methods for generating the subgroups were detailed in reference [1], including removing the image boundaries, rescale the images, and random under sampling for the OK class and over sampling for the DEF class.

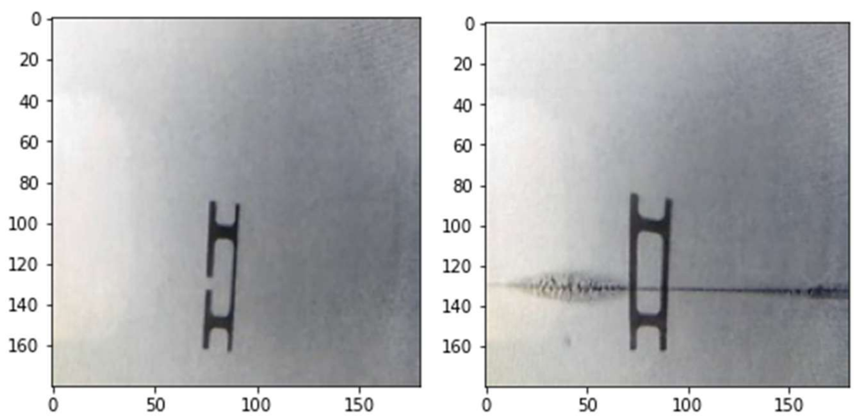

Figure 2 Examples from data_balanced dataset OK image (on the left) and DEF image (on the right)

The data_balanced folder has the processed RGB images that have been already chopped and focused on the printing region, image size reduces from 300px to 180px. The dataset is balanced by the author of paper [1]. In this report, the processed images in the data_balanced folder will be directly applied for training.

3. Data Augmentation

In paper [1], the author has performed data augmentation to avoid an immediate overfitting of the model to the training data. This will improve model performance. In this report, following the procedures and examples from Coursera, ImageDataGenerators were created for training, validation, and testing. These operations include:

- Rescaling factor of 1/255 for normalizing the image data
- Horizontal flip of the images
- Zoom range of image area 0.15
- Width shift 0.20
- Height shift 0.20
- Shear rate 0.15
- Fill mode "nearest"
- Random image rotation of 20 degrees

Examples of the augmented images are shown in Figure 3 and Figure 4.

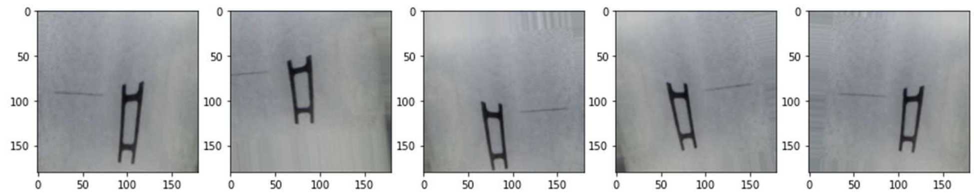

Figure 3 Examples for Data Augmentation

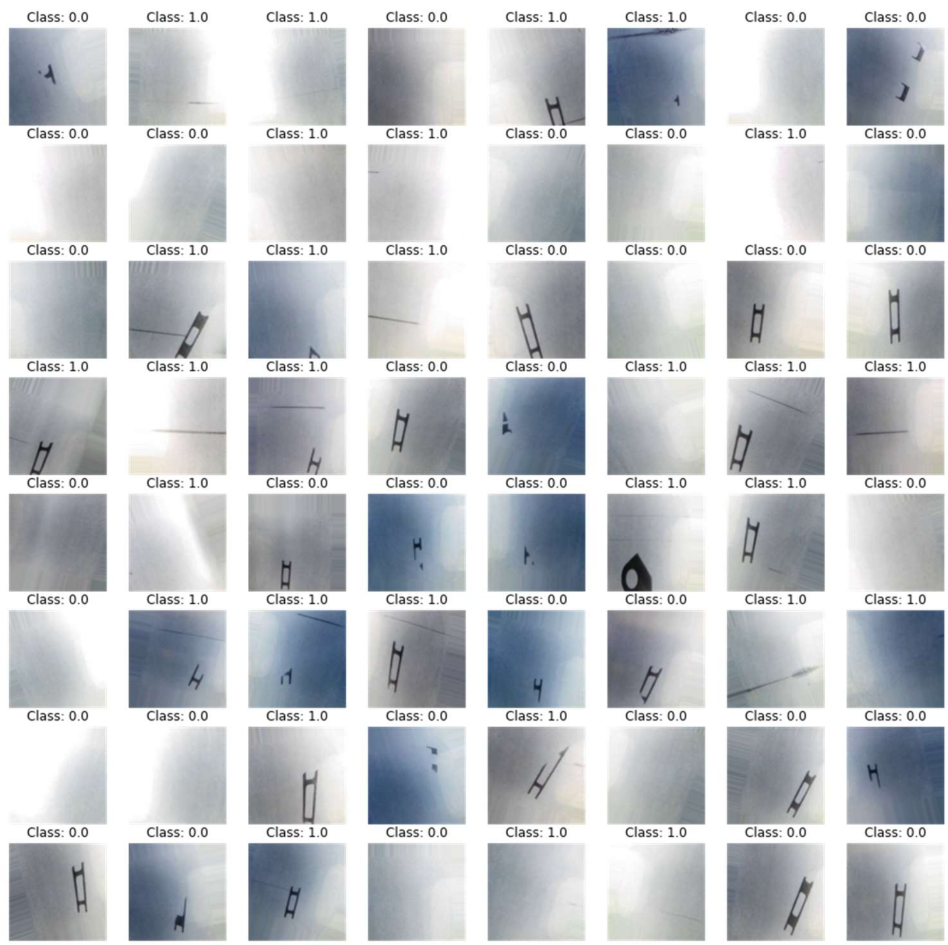

Figure 4 Batch 1 data from Training Dataset after Augmentation. Class 0.0 represents OK condition; class 1.0 represents DEF condition.

4. Transfer Learning Models and Training Results

CNN (convolutional neural network) is a kind of deep neural networks that have been widely used in computer vision problems. The CNNs can automatically learn the features from input data and make a prediction (i.e., classification). The CNN model requires large datasets during the training process to reach desirable accuracies. However, often insufficient data is provided. In these cases, transfer learning can be applied. Transfer learning leverages prior learnings based on another task and applies the same deep neural network structures to a similar problem. The assumption is that the first few layers of the deep neural network perform the same feature extraction functions. Both VGG16 and the Xception architectures were adopted in reference [1].

In this report, due to limited time, only three models were explored:

(1) VGG16 with nontrainable parameters from ImageNet and additional layers for training (Figure 5). This model has trainable parameters, 12,804,001.

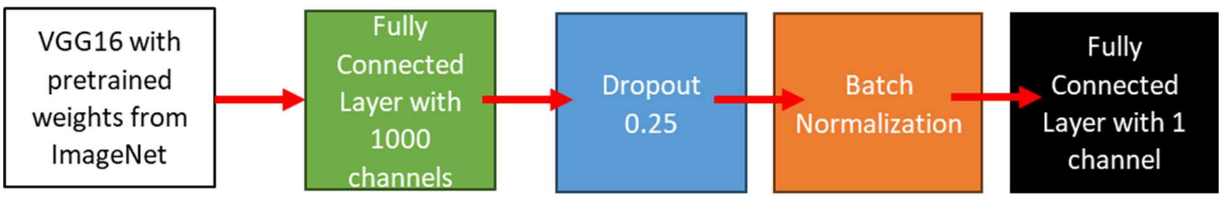

Figure 5 Transfer Learning by applying VGG16 model with pretrained weights from ImageNet.

(2) VGG16 with trainable top layers and additional layers for training (Figure 6). This model has trainable parameters, 25,783,201.

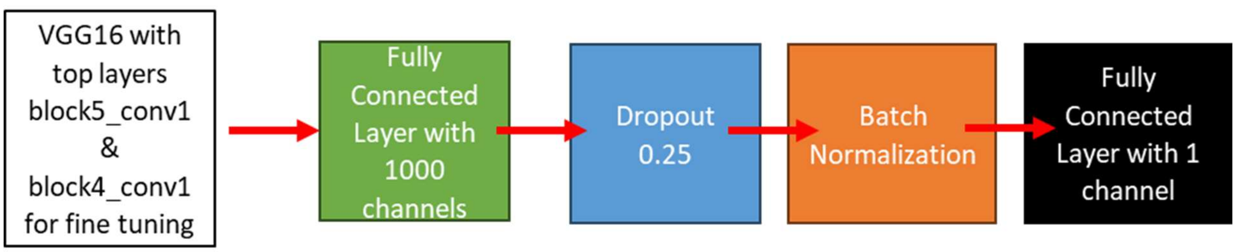

Figure 6 Transfer Learning by applying fine tuning VGG16 top layers.

(3) Xception with nontrainable parameters from ImageNet and additional layers for training (Figure 7). This model has trainable parameters, 73,732,001.

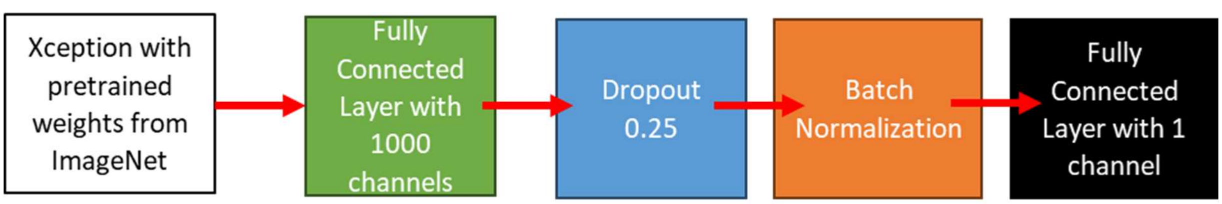

Figure 7 Transfer Learning by applying Xception model with pretrained weights from ImageNet.

For all three models above, the similar hyperparameters are applied:

- The total training epochs is set as 30.
- Binary cross entropy is the loss function.
- Learning rate is $1 \times 10{-}3$, Lr decay patience = 5
- Dropout 0.25
- Optimizer is Adam ß1 = 0.9 ß2 = 0.999
- Early stopping, patience = 20

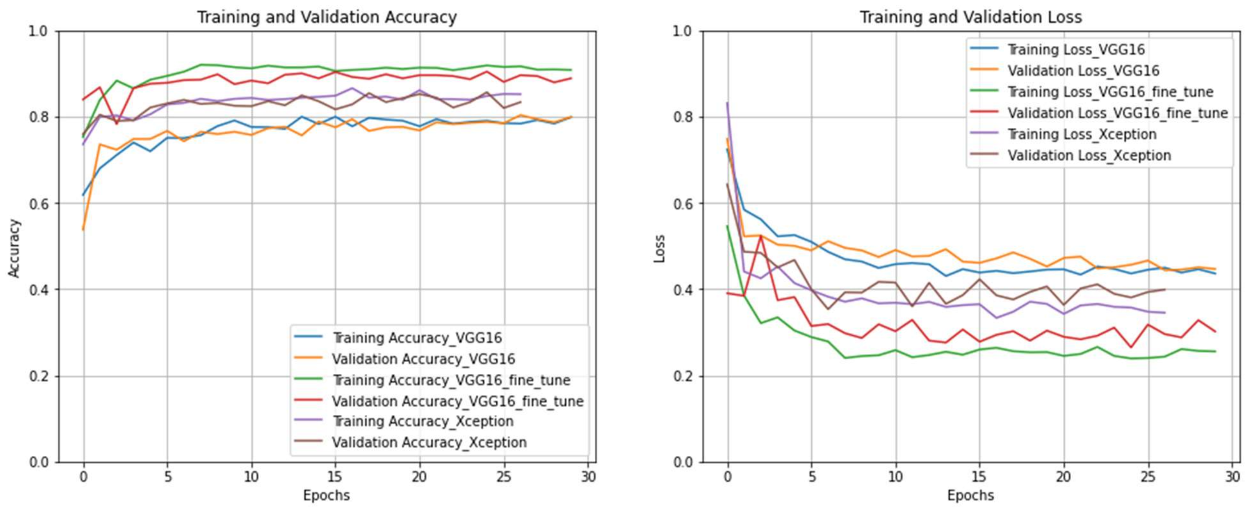

Figure 8 Training and Validation Accuracy and Loss.

Figure 8 shows the accuracy and loss histories of training and validation datasets for all three models during the training process. Training and validation accuracies are similar, meaning there is no overfitting of the model. Among the three models, the first model (VGG16 with nontrainable ImageNet weights) performed worst, and its prediction accuracy is about 80%. The third model (Xception with nontrainable ImageNet weights) performed slightly better than the first model, reaching an accuracy of about 85%. The second model (VGG16 with fine-tuned top layer weights) performed best with an accuracy of 90%. Hence, based on accuracy, the second model best fits the need to capture the defects present in LPBF in-situ monitoring.

After deciding the best model, we evaluate the prediction accuracy of the test dataset by using the second model (VGG16 with fine tuned top layer weights). The test accuracy is 0.9158. Figure 9 demonstrates the examples of final predictions by using the test dataset. For each subgraph, the title indicates the ground truth class of the image and the prediction from the transfer learning model. The ground truth is 0.0 or 1.0, corresponding to the label OK and DEF. Among the 16 example images, we observed one sample that is predicted wrong (highlighted in red box). This clearly shows that the trained transfer learning model has served the purpose of predicting the defects.

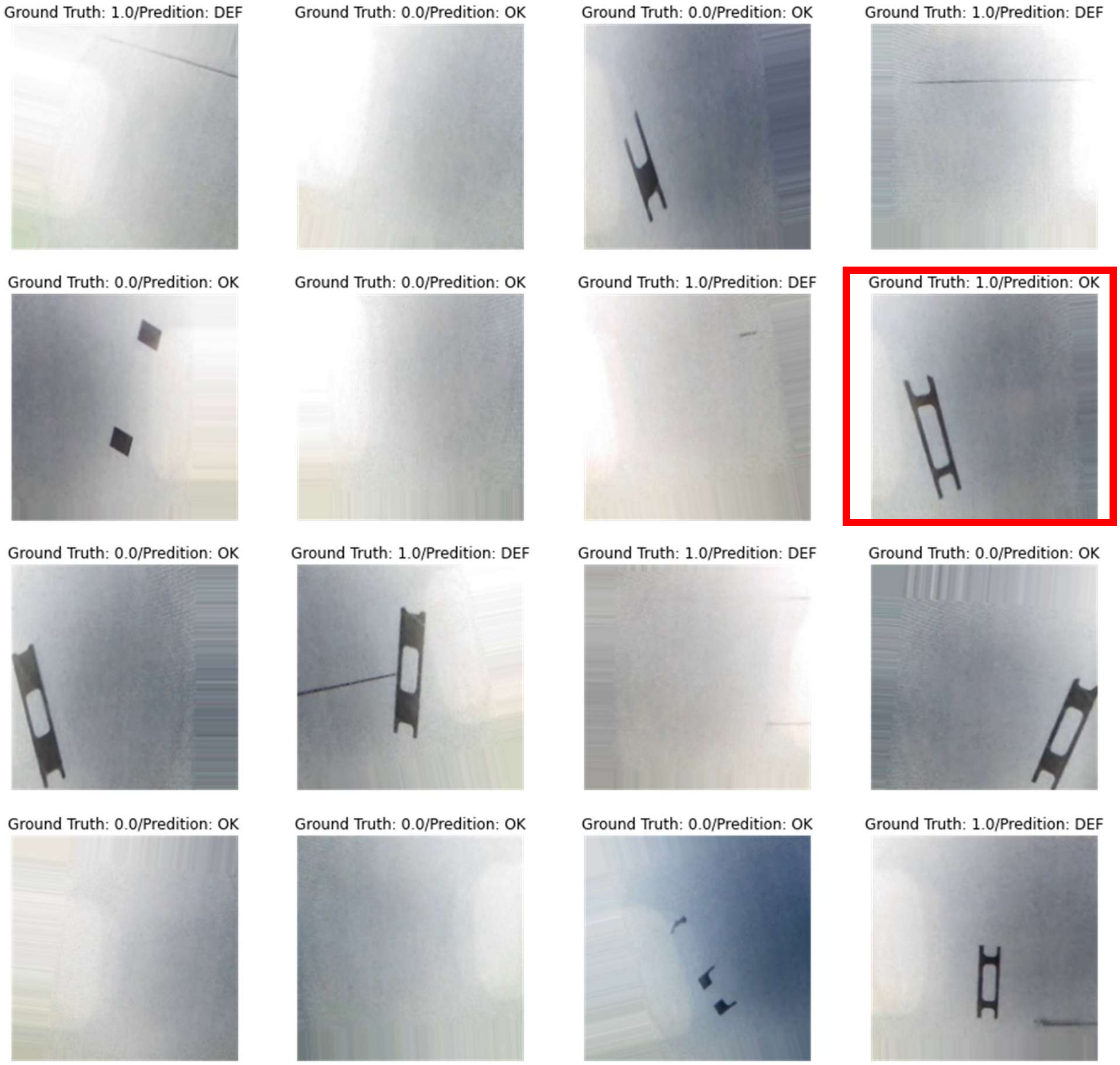

Figure 9 Prediction results from VGG16 fine-tuned Transfer Learning Model.

5. Summary of Key Findings

- In order to demonstrate the capability of applying deep learning approach for classifying the defects in LPBF process, an online dataset was explored.

- The online dataset has two directories. One directory contains the original images with an image data Imbalanced Ratio: 11.059. The other directory contains processed RGB images that have been already chopped and focused on the printing region. The dataset is balanced by the author of paper [1] based on down and over sampling technique, and has already been divided into training, validation, and test subgroups. In this report, we directly train our models based on the balanced dataset.

- Data augmentation has been applied to process the data to avoid overfitting and ensure better model performance. The images undergo operations such as shift, mirror, flip, zoom, etc.

- Three transfer learning models have been set up and trained, including: (1) VGG16 with nontrainable parameters from ImageNet and additional layers for training; (2) VGG16 with trainable top layers and additional layers for training; (3) Xception with nontrainable parameters from ImageNet and additional layers for training. Other hyperparameters are fixed, such as dropout rate, learning rate, etc.

- Results have shown that model (2) is the best model for predicting the defects during LPBF printing process in terms of accuracy. Its testing accuracy reaches 0.9158. But the trained models in the current report do not meet the accuracy presented in the original paper. This indicates that there is still room for improvement.

6. Recommendations for Next Steps

In order to obtain better accuracies, we can try the following options in future:

- Try other transfer learning models, such as InceptionV3, ResNet50, etc.
- Increasing model complexity by adding more fully connected layers in the classifier. Or set more weights in the top layers of the base model for fine tuning.
- Varying other hyperparameters, such as optimizer, dropout rate, learning rate, etc.
- Adopt ensemble learning approach: train multiple models with different architectures or initializations and combine them to improve performance.

Additionally, for a better representation and evaluation of the results, we can consider measuring AUC, F1-score, precision, and recall.

References

[1] Westphal, Erik, and Hermann Seitz. "A machine learning method for defect detection and visualization in selective laser sintering based on convolutional neural networks." Additive Manufacturing 41 (2021): 101965. https://doi.org/10.1016/j.addma.2021.101965

[2] Xin, Junyi, et al. "Ensemble learning based defect detection of laser sintering." IET Optoelectronics 17.6 (2023): 273-283 https://doi.org/10.1049/ote2.12108