International Journal on Advanced Science Engineering Information Technology

Vol.14 (2024) No. 2
ISSN: 2088-5334

Performance Analysis of ResNet50 and Inception-V3 Image Classification for Defect Detection in 3D Food Printing

Cholid Mawardi [a,*], Agus Buono [a], Karlisa Priandana [a], Herianto [b]

[a] Department of Computer Science Faculty of Mathematics and Natural Sciences, Institut Pertanian Bogor, Bogor, 16680, Indonesia
[b] Department of Mechanical & Industrial Engineering, Gadjah Mada University, Yogyakarta, 55281, Indonesia
Corresponding author: *mawardicholid@apps.ipb.ac.id

Abstract—**In the future, 3D food printing may be a fruitful area for the food industry. The presence of this technology opens the public's eyes to a revolution in the food sector, where food processing is limited to the conventional. The existence of this technology will also make food production much more effective and efficient. However, there is a problem when the 3D Food Printing process is carried out; when object defects occur, there will be material waste. This research proposes deep learning-based defect detection for defect and non-defect objects. This model mainly uses CNN as a deep learning method. The dataset is taken from the image of the print process performed at the time of object creation to be trained and validated to check the effectiveness of the proposed model. The architecture used uses pre-trained CNN models namely Inception-V3 and ResNet50 with the hope of classifying images that have a higher viscosity of the material. Where the model has been tested with previous datasets and applied with 3D Food Printing datasets with a dataset division ratio of 85% which is training data and 15% is validation data. After testing the two proposed scenarios, the accuracy result obtained in the test model scenario 1, Inception-V3, is 84.62% and for the test model scenario 2, ResNet50, the accuracy is 93.83%. The outcomes also demonstrate that improved accuracy, loss, and classification time may be obtained by applying CNN in conjunction with data augmentation.**

Keywords— **3D Food Printing; defect; CNN; Inception-V3; ResNet50.**

Manuscript received 18 Jan. 2024; revised 12 Feb. 2024; accepted 1 Apr. 2024. Date of publication 30 Apr. 2024.

I. INTRODUCTION

Perfection in the form of food is essential for every industry in the food industry. Good service in terms of food perfection is a crucial factor in getting the primary response from customers. 3D Food Printing is here to support food technology innovation in all elements. Experts have predicted that 3D food printing will become a part of culinary life, both in restaurants and at home [1].

The manufacturer also claims that, unlike mass-produced industrial food products, 3D Food Printing will truly improve human nutrition, allowing every human to print fresh and healthy food right in the kitchen and with maximum transparency with minimum time and effort [2]. The shape characteristics of 3D Food Printing objects are also important for object recognition. Various deep learning algorithms can be used to recognize 3D Food Printing objects with shape, texture, and other features to have good detection capabilities and high accuracy. The remote monitoring system in the 3D printing process is also a problem in addition to the perfection of the shape of the food printed by the 3D Food Printer machine [3]. When the automation process is executed, the tendency to avoid defect detection can be done. Although extrusion-based 3D food printing technology has advanced in the past decade, food layer interlayer imperfections such as delamination and warping are still dominant when printing complex parts.

A self-monitoring system was created to categorize various degrees of delamination in food printing using real-time camera images and deep learning algorithms. In addition, a new method incorporating strain measurement was established to measure and predict the onset of warping [4]. Without human intervention, this self-monitoring system can also evaluate other production processes to achieve autonomous calibration and pre-diagnose defects. Materials are the focus when a monitoring system for defect detection is needed—early detection of food-printed objects, making the efficient provision of food goods. Several studies show that deep learning models can detect various levels of delamination [5] and successfully determine the level and trend of defects before they occur in food object prints. Many studies have been done on defect detection in conventional 3D printing, especially with the help of artificial intelligence technology that begins with computer vision methods [6]. Baumann and Roller [6] conducted research with computer vision to detect fault diagnosis; Baumann divided the defect classification into 5 parts, of which 3 were successfully detected with false positives with a 60 to 80 percent detection rate.

Research conducted by Baumgartl et al. [7], observing the printing process with delamination features obtained an accuracy of 96.80%. Then in research conducted by Kadam et al [3], a machine learning model was used to detect faults with SVM, KNN, Random Forest, Decision Tree, and naïve Bayes algorithm models where the highest accuracy uses the SVM model with 99.70%. Research conducted by [8] improved print results were achieved by analyzing 3D printing fault detection using augmentation data on a Convolutional Neural network (CNN), which had an accuracy value of 95.45% using private datasets.

So, from this, various research studies were conducted on detecting new defects carried out with conventional 3D printing, and no one has used a 3D food printing machine. Also, because of the voids that occur in 3D Printing [9], voids can also be included in the types of defects in 3D Food Printing. In 3D Food Printing, defects obtained in the printing process are less than in ordinary 3D Printing. This is due to the different types of materials used. In 3D Printing, with the kind of Polylactic Acid (PLA) material, defect problems often occur, namely warping [10], under extrusion, and over extrusion. Etc. Research conducted by [11] conducted defect detection in 3D Food Printing, but in detail, the technology used to detect defects does not yet exist. The research undertook only mitigation strategies and characteristic properties of defects in 3D Food Printing. Therefore, our research will perform a defect detection process with deep learning to obtain classification results and accuracy values, which can help monitor the printing process in 3D food printing.

II. Materials and Method

The research method in this study is shown in Figure 1. This study has seven main stages, starting with making the formation setup and then performing the printing process on 3D food printing with chocolate material. Next, we take the image dataset during printing with defect labeling detection from the Obico plugin on the camera captured at one side of the corner. After that, the image is preprocessed by resizing the dataset and dividing the process data by the training and validation data. In the next stage, the image will be trained using Inception-V3 and ResNet50 models to get maximum results, and in the final stage, the model evaluation will be carried out.

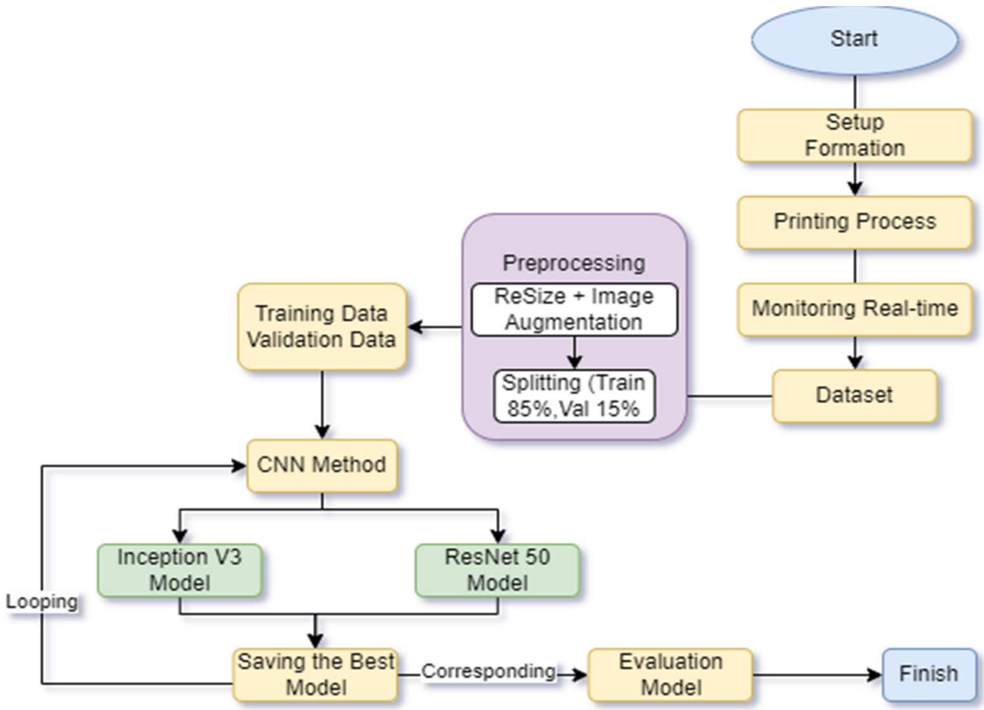

Fig. 1 Flowchart of the research method

A. *3D Food Printing Material*

By adding fruits, vegetables, and other vitamin sources in between the layers created, 3D printing can also encourage the general public to consume these foods. This way, the food products will be strengthened with nutrients without the consumer noticing them [12]Though 3D food printing has progressed significantly recently, many obstacles remain in the areas of printable ingredients (the printing technique limits the types of food that can be printed), process (the process is prolonged for mass production), and consumer perception (very few people know about food printing and tend to reject it) [13]. The search for suitable food materials and the adjustment of their printing parameters are the main topics of many of the published studies on 3D food printing [14], [15] The experiment was conducted using a 3D Printer machine the Ender 3, plus a food extruder tool that can remove printed objects, namely the luckybot extruder. A camera is used to monitor defect detection that occurs in the printing process [16].

B. Setup Formation

Fused Deposition Modeling (FDM) is the 3D printing technology employed in this research, and the food 3D printer used is the 3D Printer Ender 3. The Luckybot food extruder, an additional food printing device, is combined with the 3D printer. The printer head, nozzle replacement, and motor cable transfer on the Ender 3D Printer machine's motherboard are among the components disassembled throughout the machine construction process. It should be noted that our suggested approach is currently limited to the printer and printing procedure described above. Using cameras to take photographs at certain checkpoints, the pieces that are 3D printed are monitored [17],[18]. Raspberry Pi also helps the connection between monitoring devices to detect PCs with LAN/Wi-Fi networks, from camera capture directly controlled through octoprint software.

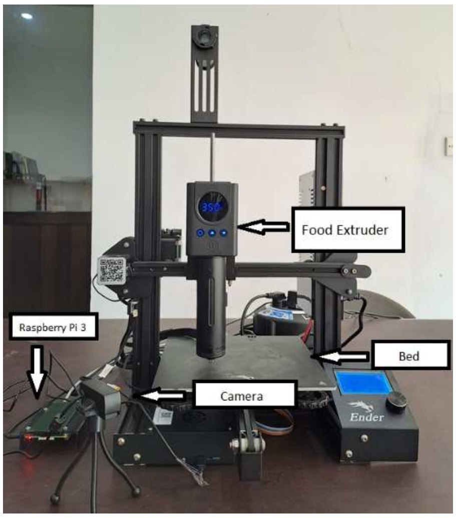

Fig. 2 Experimental Setup 3D Food Printing Monitoring

The Internet of Things is integrated with the camera setup procedure, as depicted in Figure 3, and brown printing material was employed for this investigation. Chocolate is an excellent substance for the 3D food printing machine's extrusion concept [19]. The differences significantly influence the quality of the product printed with both materials in their mechanical qualities [20]. Code to automate real-time printing process monitoring has been developed using the Obico and Octoprint applications [21]Throughout this investigation, we kept the same setup for 3D food printing. Some tests were carried out first by taking images of the 3D printing process, taking into account the impacts of lighting conditions.

C. Printing Process

The images taken for analysis underwent a molding process with a nozzle temperature of 200°C, sufficient to extract the required chocolate ingredients, and a food extruder temperature of 35°C. For the bed temperature, a temperature of 50°C is required so that the attachment process between the chocolate material and the bed is strong. So that the printing can have a good shape according to the design. Human resources and operators have a significant impact on the printing process since different operator skills will result in higher-quality prints [22].

TABLE I
PRINTING PROCESS PARAMETER TEST

Manufacturing Parameter	Value	Manufacturing Parameter	Value
Print direction	XYZ	Nozzle diameter (mm)	0,4
Material	Chocolate Food	Nozzle temperature (°C)	200
Raster angle	0	Cooling	No fan cooling
Layer height (mm)	0,1	Infill (%)	30
Bed temperature (°C)	50	Filament diameter (mm)	2,00
Print speed (mm/min)	2400	Number of layers	20

As shown in Table 1 using the XYZ direction pattern, 3D Food printing follows the direction of the Gcode command during the printing process, so alignment between the design command and the Gcode command is needed. In this printing process, defects occur that can cause material waste. The accuracy level of these defects will be analyzed.

This research takes the dataset from scratch by capturing data from the ongoing printing process. With the Obico plugin tool and eye observation, the dataset is captured by video recording. Then, the image is broken into frames and analyzed by deep learning. Therefore, transfer learning techniques are used in the classification process. Transfer learning is one of the solutions to overcome data limitations when classifying images using deep learning [23]. One transfer learning process is using a pre-trained model to build a new CNN architecture. In this research, two pre-trained models were used, namely Inception-V3 and ResNet50. The two pre-trained models were chosen because they can build CNN models on devices with limited resources [24].

This study compares the performances of the Inception-V3 and ResNet 50 models. To increase its accuracy, the architecture of the pre-trained model was altered. The neural network layer's hyperparameter tuning was used to make changes. The research's difficult task is choosing the correct parameter values to create a reliable model for mobile device picture categorization.

2085 data is taken to perform the defect detection process on 3D Food Printing. 15% of the dataset is used for testing, while 85% is used for training. This percentage indicates 1773 images in the training set and 312 in the testing set. Datasets are grouped into defects and no defects. The skin disease dataset is displayed. The defective 3D Food Printing dataset can be seen in Figure 3.

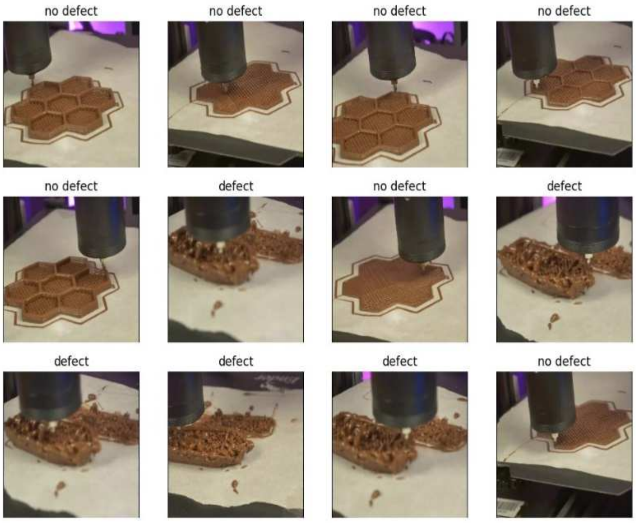

Fig. 3 Defect Class: Defect & No Defect

The dataset is taken while the print process runs and detected by the Obico plugin if a defective object occurs. For every image recording that arises with a defect label, the researcher takes the data from the frames of the recording to be used as a dataset. The 2085 data taken are heterogeneous data from the chocolate printing process in 3D Food Printing. In this research, the dataset consists of several variables formed from materials and is grouped into several classes. The surface shape and object shape, as in Table 2, are very influential on image detection because they can affect the precision and accuracy of results when optimized.

TABLE II
VARIABLE DATASET

Variable	Data	Description
Materials	Couverture Callebaut Chocolate	It can be supplemented with other materials according to the characteristics of the method used
Class	Defect, Non-Defect	In the research experiment, the dataset was divided into 2 classes
Surface	Solid, Flat, Perforated	Object printout adjusts print parameters
Object shape	Spaghetti, Warping, Under Extrusion, Poor Initial Layer	Variable over image scrapping based on the print object that appears

D. Pre-processing Process

The most challenging step in the procedure is pre-processing because the data must be normalized. Uniform image resolution is how the normalizing process is completed. The steps completed in mage Resizing, localization, augmentation, and labeling are examples of pre-processing [25]. An image prepared for vector processing is the end product of pre-processing.

1) Resize Image: The dataset consists of various-sized images. As a result, the image size must be equalized to facilitate robot object recognition. This study resizes the image so that all processed images have the same 224x224 size.

2) Image Augmentation: Five augmentation strategies have been used to expand the quantity of datasets. Some augmentation techniques include rescaled, rotation, zoom, horizontal flipping, and random shift. Through augmentation, a new image with the same features as the original is created [26].

3) Image Splitting: Following the pre-processing phase, the image has a consistent size and shape format, ready for further processing throughout the feature extraction procedure. Photos must be divided into training and validation sets to speed up the model validation procedure and begin feature extraction. The training and validation sets are compared at 85% versus 15%.

Strong normalization is needed to overcome imbalanced data in the pre-processing stages above. In this research, every data captured from the camera, the camera capture process indeed uses good angle capture techniques to get high-quality image data. The 3 points above, namely resizing the image, image augmentation image, and splitting, are enough to overcome the slightly imbalanced data.

E. Model Architecture

This research uses the pre-trained learning CNN model, using model because the pre-trained model can detect previous models that at least have knowledge that can be utilized to detect other objects [27]. For example, the model already knows which objects are background which are trees, and which are not. We transfer that knowledge to new needs. Therefore, we still need to train again (although most training processes are much faster because the model is already a bit smarter).

Transfer learning is a deep learning technique that reuses a previously prepared model to train a new model for a related problem [28]A convolutional neural network with 50 layers of depth is called ResNet-50. The three picture blocks that makeup RestNet-50 are the architecture: a convolution block (in the middle) that modifies the input dimensions and an identity block (on the right) that does not. [29] CNN is a kind of selective deep architecture made up of a pooling layer and a convolution layer on top of each other to create a deep model [30].

TABLE III
INCEPTION-V3 BASELINE NETWORK ARCHITECTURE

Stage	Operator	Resolution	Stride
1	Convolution	299 × 299 × 3	3 × 3/2
2	Convolution	149 × 149 × 32	3 × 3/1
3	Convolution	147 × 147 × 32	3 × 3/1
4	Pooling	147 × 147 × 64	3 × 3/2
5	Convolution	73 × 73 × 64	3 × 3/1
6	Convolution	71 × 71 × 80	3 × 3/2
7	Convolution	35 × 35 × 192	3 × 3/1
8	Inception module	35 × 35 × 288	Three modules
9	Inception module	17 × 17 × 768	Five modules
10	Inception module	8 × 8 × 1,280	Two modules
11	Pooling	8 × 8 × 2,048	8×8
12	Linear	1 × 1 × 2,048	Logits
13	SoftMax	1 × 1 × 1,000	Output

Including a crucial element known as an inception module sets the Inception-v3 apart from networks like LeNet and

VGG [31]. This module makes use of receptive kernels in various sizes. Zero padding maintains consistency in the convolution operation's output size. By using filter concatenation, the final feature maps are produced. Richer features are extracted from the input image with the aid of the inception technique. An inception network's structure is displayed in Table 3. Deeper networks can be trained with the residual network pretty effectively. As Table 4 illustrates, the concept of "identity shortcut connection" was established by ResNet [32].

$$y = F(x, \{W_i\}) + x \quad (1)$$

Consequently, we can create a residual form of a deep network by injecting shortcuts. Eq. (1) can be used to introduce identity shortcuts when the input and output have the same dimension where $F(x, \{W_i\})$ represents the residual mapping and *x, y* represents the input and output vectors. Gradients in ResNet can backpropagate to the prior layers via skip connections.

TABLE IV
RESNET50 ARCHITECTURE

Stage	Operator	Resolution	50 Layer
1	Conv1	112 × 112	7 x 7, 64 stride 2
2	Conv2_x	56 × 56	3 x max pool stride 2 $\begin{bmatrix} 1x1, 64 \\ 3x3, 64 \\ 1x1, 256 \end{bmatrix} \times 3$
3	Conv3_x	28 × 28	$\begin{bmatrix} 1x1, 128 \\ 3x3, 128 \\ 1x1, 512 \end{bmatrix} \times 4$
4	Conv4_x	14 × 14	$\begin{bmatrix} 1x1, 256 \\ 3x3, 256 \\ 1x1, 1024 \end{bmatrix} \times 6$
5	Conv5_x	7 × 7	$\begin{bmatrix} 1x1, 512 \\ 3x3, 512 \\ 1x1, 2048 \end{bmatrix} \times 3$
6	Conv3x3, average pool, 1000-d FC	1 × 1	Conv3x3

The numbers of the building blocks for each of these structures are layered and displayed in brackets. ResNet50 is a residual network that consists of 50 layers.

F. Model Evaluation

Accuracy, precision, recall, and F1 scores are computed using the confusion matrix in this study to assess model performance. Confusion matrix normalization offers more details on the relationships between classes and kinds of classification mistakes [33].

$$Accuracy = \frac{(TP+TN)}{(TN+FP+FN+TN)} \quad (2)$$

$$Precision = \frac{(TP)}{(TP+FP)} \quad (3)$$

$$Recall = \frac{(TP)}{(TP+FN)} \quad (4)$$

$$F1\ Score = 2\ x\ \frac{Precision\ x\ Recall}{Precision\ x\ Recall} \quad (5)$$

Furthermore, all experiments were recorded. The experimental results were analyzed to determine the most accurate (highest accuracy), lightest (fastest training and testing time), and most efficient (best combination of all metrics) model, as well as the image augmentation that most significantly improved the model's performance.

III. RESULTS AND DISCUSSION

A. Testing The Use of Learning Rate

The learning rate is one of the training parameters to calculate the weight correction value during the training process [34]. This learning rate value ranges from zero (0) to (1). The training process will proceed more quickly the higher the learning rate value. The network will be less accurate the higher the learning rate, and vice versa; if the learning rate is lower, the network's accuracy will be higher or rise, which will lengthen the training process. Tables 5 and 6 are the results of comparing models generated from 4 different learning rates.

TABLE V
LEARNING RATE COMPARISON OF THE INCEPTION-V3 MODEL

Learning Rate	**Inception-V3**		
	Accuracy (%)	**Precision (%)**	**Sensitivity (%)**
0.005	76.92	85.45	84.33
0.001	79.49	87.51	86.53
0.0005	83.97	90.79	90.04
0.0001	81.09	88.73	87.83

Table 5 shows data indicating that the Inception-V3 model's learning rate value of 0.0005 when employing the Adam optimizer yields the best results, with an accuracy value of 83.97%. The learning rate in the ResNet50 model yields the most remarkable results; however, it likewise employs a learning rate value of 0.0001 and achieves an accuracy value of 89.73%. The outcomes of the ResNet50 learning rate comparison are displayed in Table 6.

TABLE VI
LEARNING RATE COMPARISON OF RESNET 50 MODEL

Learning Rate	**ResNet50**		
	Accuracy (%)	**Precision (%)**	**Sensitivity (%)**
0.005	89.20	94.18	93.68
0.001	79.69	87,78	86,81
0.0005	87.15	92.90	92.31
0.0001	89.73	94.50	94.03

Based on Table 6 above, the learning rate value is not too small to get good accuracy. The dataset is very influential based on the test results, from image preprocessing and algorithm analysis.

B. Training and Validation Results using Inception V3

Inception-V3 architecture was used in the model's development for the training and validation phases. The data is divided into training and validation sets with a learning rate of 0.0005, with a total of 2085 training sets comprising both defect and non-defect images. There are 1773 images utilized for training in the training and validation stage, where the ratio of training data to validation data is 85:15. To validate the relationship between epochs and the accuracy and loss that resulted from the training stage, a total of 312 pictures were employed. The curves displayed in Figures 4 and 5 display the corresponding validations.

Fig. 4 The Validation Accuracy using Inception-V3

Inception-V3 accuracy results on training data are shown on the blue line, while validation uses red. These results show that the best learning rate gets an accuracy value of 84.62% with 26 epochs.

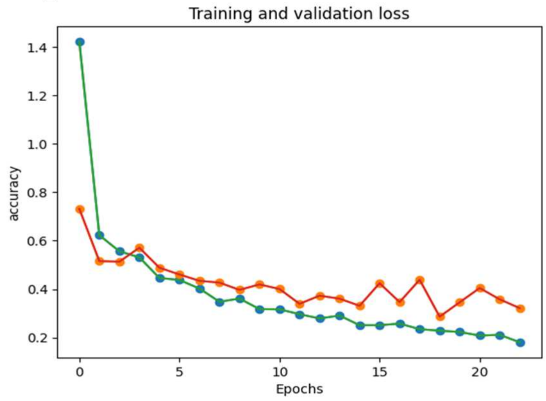

Fig. 5 The Validation Loss using Inception-V3

C. Training and Validation Results Using ResNet 50

The model further uses the ResNet50 architecture in the training and validation stages. With a learning rate 0.0001, the data is also grouped into training and validation data, each totaling 2085 training data consisting of defect and non-defect images. Like the Inception-V3 model, the training and validation data ratio is 85:15, and 1773 images were used for training. As demonstrated in Figures 6 and 7, 312 images were used for validation to examine the link between epochs and the accuracy and loss that resulted from the training and validation stages.

Fig. 6 The Validation Accuracy using ResNet50

ResNet50 accuracy results on training data are shown on the blue line, while validation uses red. These results show the best learning rate using 0.0005 with epochs 40 and an accuracy value of 93.83%.

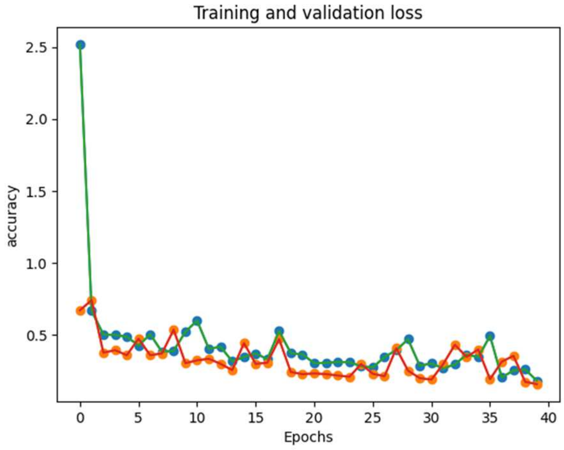

Fig. 7 The Validation Loss using ResNet50

D. Comparison of Accuracy Results

Based on the results of accuracy, Precision, Recall, and F-1 Score, the ResNet50 Model has the highest accuracy of 93.83% compared to Inception-V3, which only has 84.62%. In Table 7, ResNet50 also achieves the highest Precision with 96.84%. Another influencing factor is the shape of objects, which is difficult to analyze by Inception-V3 and very easy to analyze by ResNet50, especially with datasets sourced from printed materials with high viscosity.

TABLE VII
PERFORMANCE COMPARISON

Architecture CNN	**Performance**			
	Accuracy (%)	**Precision (%)**	**Recall (%)**	**F1-Score (%)**
Inception V3	84.62	91.24	90.51	90.97
ResNet50	93.83	96.84	96.56	96.70

TABLE VIII
PERFORMANCE COMPARISON OF SEVERAL METHOD OF 3D PRINTING

Method	**Accuracy (%)**	**Ref**
A deep learning-based model for defect detection in laser-powder bed fusion using in-situ thermographic monitoring	96.8%	[7]
Fine-Tuned CNN with Data Augmentation for 3D Printer Fault Detection	95.45%	[8]
Proposed Method (3D Food Printing)	93.83%	

Table 8 shows the results of this study compared to previous research with a higher accuracy value than last research, with a value of 96.80% [7]. This is because the materials and methods used are very different. Suppose Hermann Baumgartl's research uses conventional 3D printing with laser powder bed fusion. This research uses 3D food printing with a distinct chocolate material and a higher viscosity than PLA or ABS 3D printing materials.

Also, in the concept research of [8], with an accuracy value of 95.45%, the printout was perfected by analyzing 3D printing fault detection using data augmentation on Convolutional Neural network (CNN) using private datasets. However, it also uses ordinary filament material, which is materially quite good with a level of difficulty, unlike 3D Food Printing.

IV. CONCLUSION

Based on the results, the first test model, ResNet50, obtained an accuracy of 93.83%, higher than the test model scenario of test model 2, Inception-V3, which is 84.62%. The shape of 3D food printing between defects and non-defects looks identical. Still, in this study, they can be distinguished with the help of a trained system using the Convolution Neural Network (CNN) algorithm, even though the dataset image material has a high viscosity. Numerous factors affect the outcome, such as the distribution of the splitting ratio, the number of layers, the number of inputs, and the balance of the datasets used. The same study topic and dataset should be used in future research, and the CNN algorithm should be modified by attempting to combine the CNN model with transformers to achieve the best possible outcomes. Another research opportunity is the application of this pre-trained learning algorithm to other 3D printing objects, such as 3D buildings, which also need optimization techniques for defect detection in the future.

ACKNOWLEDGMENT

The BIMA-PDD Ministry of Education, Culture, and Research of the Republic of Indonesia funded this research.

REFERENCES

[1] P. Pitayachaval, N. Sanklong, and A. Thongrak, "A Review of 3D Food Printing Technology," *MATEC Web Conf.*, vol. 213, pp. 1–5, 2018, doi: 10.1051/matecconf/201821301012.

[2] J. Sun, Z. Peng, L. Yan, J. Y. H. Fuh, and G. S. Hong, "3D food printing-An innovative way of mass customization in food fabrication," *Int. J. Bioprinting*, vol. 1, no. 1, pp. 27–38, 2015, doi:10.18063/IJB.2015.01.006.

[3] V. Kadam, S. Kumar, A. Bongale, S. Wazarkar, P. Kamat, and S. Patil, "Enhancing surface fault detection using machine learning for 3d printed products," *Appl. Syst. Innov.*, vol. 4, no. 2, 2021, doi:10.3390/asi4020034.

[4] G. X. G. Zeqing Jin, Zhizhou Zhang, "Autonomous in-situ correction of fused deposition modeling printers using computer vision and deep learning," *Manuf. Lett.*, vol. 22, pp. 11–15, 2019, doi:10.1016/j.mfglet.2019.09.005.

[5] E. Polyzos, A. Katalagarianakis, D. Van Hemelrijck, and L. Pyl, "Delamination analysis of 3D-printed nylon reinforced with continuous carbon fibers," *Addit. Manuf.*, vol. 46, no. June 2023, 2021, doi: 10.1016/j.addma.2021.102144.

[6] F. Baumann and D. Roller, "Vision based error detection for 3D printing processes," *MATEC Web Conf.*, vol. 59, pp. 3–9, 2016, doi:10.1051/matecconf/20165906003.

[7] H. Baumgartl, J. Tomas, R. Buettner, and M. Merkel, "A deep learning-based model for defect detection in laser-powder bed fusion using in-situ thermographic monitoring," *Prog. Addit. Manuf.*, vol. 5, no. 3, pp. 277–285, 2020, doi: 10.1007/s40964-019-00108-3.

[8] S. M. Rachmawati, M. A. Paramartha Putra, T. Jun, D. S. Kim, and J. M. Lee, "Fine-Tuned CNN with Data Augmentation for 3D Printer Fault Detection," *Int. Conf. ICT Converg.*, vol. 2022-Octob, pp. 902–905, 2022, doi: 10.1109/ICTC55196.2022.9952484.

[9] Y. Tao *et al.*, "A review on voids of 3D printed parts by fused filament fabrication," *J. Mater. Res. Technol.*, vol. 15, pp. 4860–4879, 2021, doi: 10.1016/j.jmrt.2021.10.108.

[10] A. Saluja, J. Xie, and K. Fayazbakhsh, "A closed-loop in-process warping detection system for fused filament fabrication using convolutional neural networks," *J. Manuf. Process.*, vol. 58, no. November 2019, pp. 407–415, 2020, doi:10.1016/j.jmapro.2020.08.036.

[11] A. S. M. Pattarapon Phuhongsung, Min Zhang, Sakamon Devahastin, "Defects in 3D/4D food printing and their possible solutions: A comprehensive review," *Compr. Rev. Food Sci. Food Saf.*, vol. 21, no. 4, pp. 3455–3479, 2022, doi: 10.1111/1541-4337.12984.

[12] M. Z. Sangeeta Prakash, Bhesh R. Bhandari, Fernanda C. Godoi, *Future Outlook of 3D Food Printing*. 2019. doi: 10.1016/B978-0-12-814564-7.00013-4.

[13] T. Pereira, S. Barroso, and M. M. Gil, "Food texture design by 3d printing: A review," *Foods*, vol. 10, no. 2, pp. 1–26, 2021, doi:10.3390/foods10020320.

[14] K. K. and T. A. and J. A. M. and C. Anandharamakrishnan, "Development of fiber-enriched 3D printed snacks from alternative foods: A study on button mushroom," *J. Food Eng.*, vol. 287, p. 110116, 2020, doi: 10.1016/j.jfoodeng.2020.110116.

[15] C. Egg *et al.*, "Optimization of the Formulation and Properties of 3D-Printed Complex Egg White Protein Objects," 2020.

[16] S. Garfo, M. A. Muktadir, and S. Yi, "Defect Detection on 3D Print Products and in Concrete Structures Using Image Processing and Convolution Neural Network," *J. Mechatronics Robot.*, vol. 4, no. 1, pp. 74–84, 2020, doi: 10.3844/jmrsp.2020.74.84.

[17] M. A. Abdullah and T. F. Abbas, "Numerical Developing the Internet of Things To Remotely Monitor the Performance of a Three Dimensions Printer for Free-Form Surface," *J. Eng. Sci. Technol.*, vol. 18, no. 6, pp. 2809–2822, 2023.

[18] T. Ludwig, A. Boden, and V. Pipek, "3D printers as sociable technologies: Taking appropriation infrastructures to the Internet of Things," *ACM Trans. Comput. Interact.*, vol. 24, no. 2, 2017, doi:10.1145/3007205.

[19] R. Theagarajan, J. A. Moses, and C. Anandharamakrishnan, "3D Extrusion Printability of Rice Starch and Optimization of Process Variables," *Food Bioprocess Technol.*, vol. 13, no. 6, pp. 1048–1062, 2020, doi: 10.1007/s11947-020-02453-6.

[20] Y. Liu *et al.*, "Applicability of Rice Doughs as Promising Food Materials in Extrusion-Based 3D Printing," *Food Bioprocess Technol.*, vol. 13, no. 3, pp. 548–563, 2020, doi: 10.1007/s11947-020-02415-y.

[21] J. Xie, A. Saluja, A. Rahimizadeh, and K. Fayazbakhsh, "Development of automated feature extraction and convolutional neural network optimization for real-time warping monitoring in 3D printing," *Int. J. Comput. Integr. Manuf.*, vol. 35, no. 8, pp. 813–830, 2022, doi:10.1080/0951192X.2022.2025621.

[22] C. Mawardi, G. Suryadi, E. Elviana, and F. Yakob, "3D Printing Object Using Recycled Disposable Masks Filament," 2022, doi:10.4108/eai.16-11-2022.2326056.

[23] F. Zhuang *et al.*, "A Comprehensive Survey on Transfer Learning," *Proc. IEEE*, vol. 109, no. 1, pp. 43–76, 2021, doi:10.1109/JPROC.2020.3004555.

[24] W. G. Pamungkas, M. Iqbal, P. Wardhana, Z. Sari, and Y. Azhar, "Leaf Image Identification: CNN with EfficientNet-B0 and ResNet-50 Used to Classified Corn Disease," *J. RESTI*, vol. 5, no. 158, pp. 326–333, 2023, doi: 10.29207/resti.v7i2.4736.

[25] I. Hestiningsih, A. N. A. Thohari, Kurnianingsih, and N. D. Kamarudin, "Mobile Skin Disease Classification using MobileNetV2 and NASNetMobile," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 13, no. 4, pp. 1472–1479, 2023, doi: 10.18517/ijaseit.13.4.18290.

[26] R. L. Galvez, E. P. Dadios, A. A. Bandala, and R. R. P. Vicerra, "Object detection in x-ray images using transfer learning with data augmentation," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 9, no. 6, pp. 2147–2153, 2019, doi: 10.18517/ijaseit.9.6.9960.

[27] M. S. Fairuz, M. H. Habaebi, and E. M. A. Elsheikh, "Pre-trained based CNN model to identify finger vein," *Bull. Electr. Eng. Informatics*, vol. 8, no. 3, pp. 855–862, 2019, doi:10.11591/eei.v8i3.1505.

[28] Asesh. A, Causal Inference - Time Series. In: Biele, C., Kacprzyk, J., Kopeć, W., Owsiński, J.W., Romanowski, A., Sikorski, M. (eds) *Digital Interaction and Machine Intelligence*. MIDI 2021. Lecture Notes in Networks and Systems, vol 440. Springer, Cham. doi:10.1007/978-3-031-11432-8_4.

[29] L. Wen, X. Li, and L. Gao, "A transfer convolutional neural network for fault diagnosis based on ResNet-50," *Neural Comput. Appl.*, vol. 32, no. 10, pp. 6111–6124, 2020, doi: 10.1007/s00521-019-04097-w.

[30] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, "A survey of the recent architectures of deep convolutional neural networks," *Artif. Intell. Rev.*, vol. 53, no. 8, pp. 5455–5516, 2020, doi: 10.1007/s10462-020-09825-6.

[31] C. Feng, H. Zhang, S. Wang, Y. Li, H. Wang, and F. Yan, "Structural Damage Detection using Deep Convolutional Neural Network and Transfer Learning," *KSCE J. Civ. Eng.*, vol. 23, no. 10, pp. 4493–4502, 2019, doi: 10.1007/s12205-019-0437-z.

[32] S. Y. Chaganti, I. Nanda, K. R. Pandi, T. Gnrsn Prudhvith, and N. Kumar, "Image Classification using SVM and CNN," *2020 Int. Conf. Comput. Sci. Eng. Appl. ICCSEA 2020*, no. March 2020, 2020, doi:10.1109/ICCSEA49143.2020.9132851.

[33] D. Krstinić, M. Braović, L. Šerić, and D. Božić-Štulić, "Multi-label Classifier Performance Evaluation with Confusion Matrix," pp. 01–14, 2020, doi: 10.5121/csit.2020.100801.

[34] D. Granziol, S. Zohren, and S. Roberts, "Learning Rates as a Function of Batch Size: A Random Matrix Theory Approach to Neural Network Training," *J. Mach. Learn. Res.*, vol. 23, pp. 1–65, 2022.