Contents lists available at ScienceDirect

Journal of Food Engineering

journal homepage: www.elsevier.com/locate/jfoodeng

Comparison of deep learning models for 3D food printing monitoring system

Eunji Ju[a,c], Jaehwi Seol[a,c], Sarang Kim[a], Sol Kim[b], Soo-Jung Kim[b,d,*], Hyoung Il Son[a,c,d,**]

[a] Department of Convergence Biosystems Engineering, Chonnam National University, Yongbong-ro 77, Buk-gu, 61186, Gwangju, Republic of Korea
[b] Department of Integrative Food, Bioscience and Biotechnology, Chonnam National University, Yongbong-ro 77, Buk-gu, 61186, Gwangju, Republic of Korea
[c] Interdisciplinary Program in IT-Bio Convergence System, Chonnam National University, Yongbong-ro 77, Buk-gu, 61186, Gwangju, Republic of Korea
[d] Research Center for Biological Cybernetics, Chonnam National University, Yongbong-ro 77, Buk-gu, 61186, Gwangju, Republic of Korea

ARTICLE INFO

Keywords:
3D food printing
Monitoring
Deep learning
Defect detection

ABSTRACT

Three-dimensional food printing (3DFP) is receiving increasing attention for its ability to provide personalized nutrition and utilize alternative food sources, such as insects. However, food materials are highly sensitive to environmental factors such as temperature and humidity. This often causes defects such as nozzle clogging and uneven layer formation. These problems result in significant waste of materials and time, so continuous monitoring during the printing is essential. However, manual monitoring involves limitations owing to subjectivity and fatigue, monitoring must be done through an automated system. This paper proposes a deep learning-based monitoring system that can operate in real time and demonstrate robust performance despite environmental variations. The proposed system uses a camera to monitor the printing process in real time and detect both the type and location of defects. Since deep learning model has different characteristics, it is essential to compare models to select the most suitable model. In this study, three widely used segmentation models, U-Net, YOLOv8, and SegNet, were compared in terms of their performance and segmentation quality. A dataset was constructed using chocolate and three designs were printed three times in 27 cases. Using the dataset, the model learns sagging and thinning defects. The experimental results showed that U-Net demonstrated the best performance in defect detection. YOLOv8 displayed moderate performance with low sensitivity, highlighting its suitability for applications where speed is more important than accuracy. SegNet achieved the highest AUC value, suggesting that its performance can be enhanced via further optimization.

1. Introduction

Three-dimensional printing (3DP) is a manufacturing technology that uses computer-aided design software and digital processing equipment to produce 3D products (Lv et al., 2023). 3DP technology is being utilized in various fields. In particular, the application of this technology in the food industry, known as 3D food printing (3DFP), has attracted considerable attention owing to its potential for development. 3DFP offers significant advantages in terms of nutritional customization and resource utilization. From the perspective of nutrition, 3DFP enables the printing of food tailored to the health status and body type of an individual, supplying personalized nutrition and addressing health problems such as malnutrition. In terms of resource utilization, 3DFP can reduce food waste by utilizing undervalued materials and expanding the use of existing resources. Additionally, this technology supports low-volume production and a simplified supply chain, reducing food preparation and transportation costs. Also, 3DFP enables customized food design through process digitization, allowing consumers to create personalized, high-value products by taking advantage of low-volume production (Jiang et al., 2019; Portanguen et al., 2019; Niakousari et al., 2019; Sun et al., 2015; Wan et al., 2015). 3DFP is highly versatile owing to its ability to use a variety of materials, but the available materials vary slightly depending on the type of 3DFP.

According to the deposition mechanism, 3DFP can be classified into extrusion-based printing, inkjet printing, and powder-binding deposition (Godoi et al., 2016). Among these, extrusion-based 3DFP is the most commonly used method (Sun et al., 2018). In the extrusion process, edible food ink is placed in the extruder and extruded through a nozzle using hydraulic piston force. The layers are deposited continuously by directing the cylinder to specific locations determined by

[*] Corresponding author at: Department of Integrative Food, Bioscience and Biotechnology, Chonnam National University, Yongbong-ro 77, Buk-gu, 61186, Gwangju, Republic of Korea.
[**] Corresponding author at: Department of Convergence Biosystems Engineering, Chonnam National University, Yongbong-ro 77, Buk-gu, 61186, Gwangju, Republic of Korea.

E-mail addresses: bioksj@jnu.ac.kr (S.-J. Kim), hison@jnu.ac.kr (H.I. Son).

https://doi.org/10.1016/j.jfoodeng.2025.112903
Received 18 July 2025; Received in revised form 17 November 2025; Accepted 3 December 2025
Available online 8 December 2025

the 3D model. The binding mechanism in extrusion depends on the material properties, with extrusion typically classified into soft material extrusion, melting extrusion, and hydrogel-forming extrusion. Soft material extrusion is commonly applied for materials such as processed cheese and dough, while melting extrusion is selected for chocolate. Materials such as xanthan gum and gelatin are used in hydrogel-forming extrusion (Godoi et al., 2016). However, owing to the nature of food materials, 3DFP is sensitive to changes in environmental factors, such as humidity and temperature, which can cause printing defects.

Product quality in 3DFP is affected by the hardware, material properties, printing parameters, and environmental conditions such as humidity and temperature. Defects can occur at various stages of the printing process, including material preparation, extrusion, and deposition, as well as after the printed object is placed on the platform (Phuhongsung et al., 2022). Common types of defects include layer shifting, stringing, over-extrusion, under-extrusion, and banding. Layer shifting occurs when mechanical issues or complex print geometries interfere with axial movement, leading to misalignment of the extruder during printing (Hsiang Loh et al., 2020). Stringing results from excessive material temperature, which causes material to flow out and create unwanted thin threads between printed parts (Khan et al., 2021). Over-extrusion refers to the deposition of excess material during printing, causing the printed object to deviate from the intended digital design. Under-extrusion refers to insufficient material deposition, with less material extruded than specified by the digital design. In the case of banding, the vertical or lateral surfaces of a printed object are uneven, resulting in noticeable convex lines or a wavy appearance (Erokhin et al., 2023).

Despite the severity of these defects, most current 3D printers lack real-time monitoring systems that can detect defects during the printing process (Sani et al., 2024). To prevent defects, traditionally, a skilled operator monitors the additive manufacturing process, identifies defects, halts printing, removes the defective parts, and adjusts parameters before restarting (Brion and Pattinson, 2022). However, the limitation is that humans make judgments based on subjective criteria, and monitoring over extended periods can lead to fatigue and incorrect judgments. Therefore, an effective monitoring system that can accurately detect defects is required to improve product quality (Bhandarkar et al., 2025). This led to the development of monitoring based on computer vision (CV), which classifies images by manually extracted features. However, this approach had limitations in that manual feature extraction became increasingly cumbersome as the number of classes increased, and the selection of features depended on the judgment of the CV engineer and a long trial and error process (O'Mahony et al., 2020). These limitations led to the adoption of deep learning, which could enable automatic feature extraction, and active research is being conducted on monitoring through deep learning.

Deep learning, a subset of artificial intelligence, uses deep neural networks to learn data representations through multiple levels of abstraction (LeCun et al., 2015). Deep learning has demonstrated significant potential for defect detection owing to its robustness and generalizability to different materials, geometries, and printers (Brion and Pattinson, 2022). One key advantage of deep learning is that it can automatically extract meaningful features from data, minimizing the need for manual intervention (Shin et al., 2017). Moreover, it is highly adaptive, with continuous improvements possible through retraining, making it robust in dynamic environments. Unlike traditional monitoring methods that rely on predefined rules or manually generated features, deep learning models can learn and adapt to various printing conditions, enhancing the reliability of real-time quality evaluation (Chan et al., 2020). Based on these strengths, deep learning enables automated defect detection with high accuracy, providing significant benefits for 3DFP process monitoring.

This paper proposes a 3DFP monitoring method based on deep learning. This system can effectively identify the occurrence of a defect, as well as its type and location. Users can use this information to devise appropriate countermeasures. In addition, we compare several deep learning models to identify the most suitable algorithm for 3DFP. Unlike prior studies focused on specific materials or printing methods, our method is applicable to a wide range of materials and printers.

The contributions of this paper can be summarized as follows:

(1) We develop a 3DFP real-time monitoring system based on deep learning to detect sagging and thinning defects.

(2) Our method identifies specific defect information, such as the type and location, allowing users to establish effective countermeasures.

(3) We compare three deep learning models through experiments in the 3DFP process using chocolate to identify the most suitable deep learning model for 3DFP monitoring.

2. Related work

2.1. Monitoring and control in general 3DP

Various research studies have been conducted to detect defects in 3DP to improve output quality and reduce material waste. Vaibhav et al. developed a real-time surface defect detection system for fused deposition modeling (FDM) using machine learning and CV. Layer-by-layer images captured by the Raspberry Pi camera were analyzed using a combination of pre-trained models and algorithms. As a result, the combination of AlexNet and SVM demonstrated the best performance with an accuracy of 99.7% (Kadam et al., 2021). However, this system could only detect the occurrence of defects, without identifying their types. Luca et al. developed a deep learning-based system that detects defects in real time in 3DP using convolutional neural network (CNN). The ResNet-18 model was trained on a dataset consisting of 34 geometric shapes and 3 defect classes along with normal prints captured by a single camera. The testing results demonstrated that a defect detection accuracy was over 83%, although the system performance varied depending on the print bed color and camera angle (Rettenberger et al., 2024). Syifa et al. focused on early defect detection in FDM 3D printers using a sensor-data-driven fault diagnosis system based on a lightweight CNN and Digital Twin technology. This system achieved an F1-score of 99.81% and average latency of 995.43 ms, rendering it suitable for real-time monitoring and control (Rachmawati et al., 2023).

However, the above mentioned researches have limitations in that they can detect defects but cannot correct them. Douglas et al. trained a multihead deep neural network for real-time detection and correction of defects in 3DP. The network was trained using approximately 1.2 million automatically labeled images and could simultaneously predict and adjust key printing parameters such as flow rate, lateral speed, and Z offset to correct defects. In testing, this system achieved an overall accuracy of 84.3% (Brion and Pattinson, 2022).

2.2. Monitoring and control in 3DFP

Food is highly sensitive to environmental conditions, such as temperature, rendering 3DFP more prone to printing failures compared with 3DP. However, research in this domain remains limited compared with 3DP. Cholid et al. developed an IoT-based monitoring system using Raspberry Pi and cameras to detect defects in real-time during 3DFP. This system used the Obico plugin and supervised machine learning to analyze images captured at key stages of the process, successfully detecting 20 defect states in experiments with chocolate (Mawardi et al., 2023). Cholid et al. used a CNN to detect defects by analyzing images captured during object creation in 3DP. They used pretrained Inception-V3 and ResNet50 model architectures, along with data augmentation techniques. The experiments used couverture callebaut chocolate, and the Inception-V3 model achieved an accuracy of 84.62%, while ResNet50 displayed a higher accuracy of 93.83% (Mawardi et al., 2024).

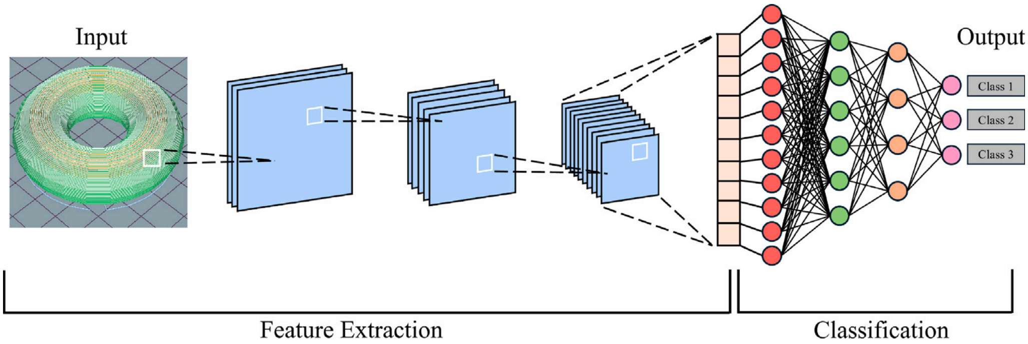

Fig. 1. Representative CNN architecture. This is a classification architecture, divided into feature extraction and classification stages, and consists of convolution layers, pooling layers, and FC layers.

Nevertheless, these systems also lacked the capability to correct defects, so research on control has been conducted. Yizhou et al. addressed under-extrusion and over-extrusion in 3DFP using CV and feedforward control. Specifically, they applied CV to measure extrusion speed and extrusion line width in real-time and implemented feedforward control to automatically adjust the nozzle movement of a printer. The experiments used white chocolate spread, lemon curd, cookie dough, and processed cheese. As a result, this system enhanced the printing accuracy to 97.6%–100%, and extrusion line width customization could be achieved with less than 8% deviation from the target (Ma et al., 2023). Ichimori et al. developed deep learning-based real-time material extrusion volume control system for 3DFP. A camera monitored the extrusion quality and a feedback control system automatically adjusted the extrusion volume during printing. ResNet18 was trained to classify the extrusion state as “Good ”, “Under”, or “Over”. The food inks used in the machine learning dataset are alpha rice flour paste, pumpkin paste, carrot paste, broccoli paste, three types of thickeners for nursing food, sweet potato paste, and nursing food pumpkin. The materials used in the extrusion amount test are caramel, cheese, sweet potato paste, pumpkin paste, and broccoli paste. The system achieved a classification accuracy of approximately 90% and effectively stabilized the printing process, reducing the need for manual intervention (Ichimori et al., 2024).

3. Deep learning

Deep learning includes various architectures, such as CNNs, recurrent neural networks, temporal convolutional networks, and transformers. Among these, CNNs are the most widely used, especially in image processing. CNNs offer various advantages, such as weight sharing, which reduces the number of trainable parameters, leading to improved generalization performance and reduced overfitting compared with other neural networks. Additionally, CNNs perform feature extraction and classification simultaneously, making the model output more structured and optimized, while facilitating large-scale network implementation (Alzubaidi et al., 2021).

3.1. CNN-based deep learning architecture

A typical CNN architecture, commonly used for image classification, is illustrated in Fig. 1. This model is structurally similar to a multilayer perceptron. It consists of multiple convolutional layers followed by pooling layers and ends with a fully connected (FC) layer that performs the final prediction. The convolutional layer consists of filters called kernels (Alzubaidi et al., 2021). In the case of the kernel, it is usually applied with a size of 3 × 3 or 5 × 5, and performs convolution by moving sequentially according to a specified interval called the stride. This generates a feature map that includes essential local feature information such as edges, patterns, and textures in the input image. Pooling layers typically use operations such as max pooling or average pooling. These layers reduce the spatial dimension of the feature map by summarizing information within local regions. After passing through several convolutional and pooling layers, the extracted features are passed to the FC layer. This layer integrates information to classify images into defined labels.

Depending on the task, CNN models can be categorized into classification, object detection, and segmentation architectures. Classification labels the entire image, while object detection uses bounding boxes to identify the presence and location of objects. In contrast, segmentation performs pixel-level analysis, enabling highly precise detection and location estimation. Therefore, in this study, we use segmentation techniques to obtain specific information about the target.

Segmentation models are built on the basic structure of CNNs used for classification, but with modifications to enable pixel-wise prediction. A typical architecture of the segmentation models is shown in Fig. 2. These models typically adopt an encoder–decoder architecture. Similar to classification models, the encoder initially uses a series of convolutional and pooling layers. These layers extract hierarchical features while gradually reducing the spatial resolution of the input image. Subsequently, the decoder takes the compressed feature and gradually reconstructs the spatial dimensions through upsampling operations, such as interpolation, unpooling, or transposed convolution. Through this reconstruction process, the decoder generates a pixel-wise segment map that assigns semantic labels to individual pixels in the original image space.

3.2. Comparison of segmentation model architectures

The performance of the segmentation model is significantly affected by structural factors such as backbone, neck, skip connection, decoder upsampling method, network depth, and activation function. These factors are summarized in Table 1. An example of the architecture of a model with a backbone, neck, and skip connection is shown in Fig. 3, which can help understand the factors in Table 1. The backbone extracts features from the input image. A deeper and more expressive architecture generally enables more precise segmentation, but increases computational complexity and learning difficulty. The neck structure aggregates multi-resolution features and passes them to the head module, thereby enhancing segmentation performance for extremely small or large objects (Zhao et al., 2017). A skip connection is a network mechanism that connects previous feature maps to later layers. Short skip connections help stabilize training by improving gradient flow, while long skip connections preserve spatial details and enhance segmentation accuracy. The upsampling method of the decoder is an important factor in restoring the output resolution. Learnable approaches such as transposition convolution can effectively reconstruct detailed structures, while simpler methods such as interpolation or index-based unpooling are computationally efficient but may have reduced boundary precision. Network depth determines the receptive field size, allowing the model to utilize richer contextual

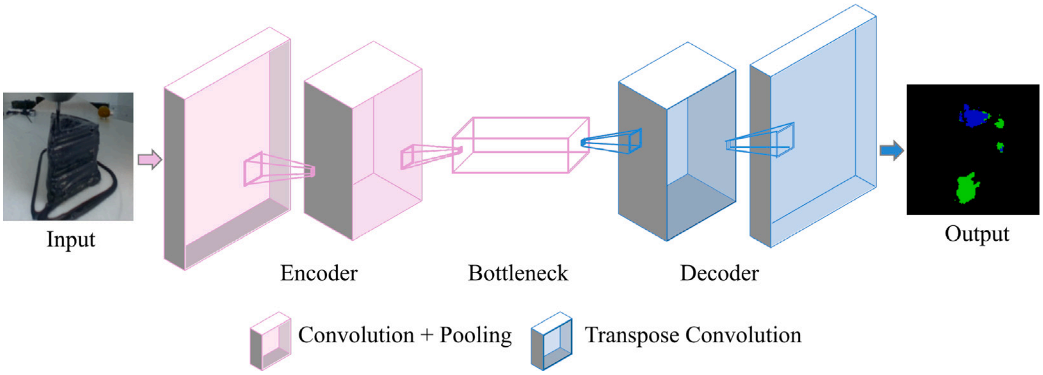

Fig. 2. Segmentation architecture. Similar to the classification architecture, but modified for pixel-wise prediction. Uses an encoder–decoder architecture. In Output, blue indicates thinning defect, and green indicates sagging defect.

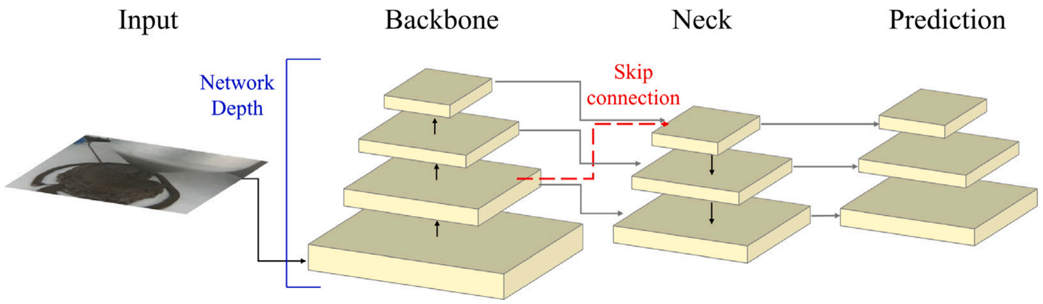

Fig. 3. A representative architecture of a model with a backbone, neck, and skip connections. This is a visualization of the factors in Table 1 and is similar to the YOLOv8 architecture.

information. Moderate depth generally improves performance, but excessive depth can lead to unstable training owing to problems such as gradient vanishing or overfitting. Lastly, the activation function directly affects the nonlinear representation capacity of the model. Recent studies have shown that SiLU, a smooth and non-monotonic activation function, improves information and gradient propagation in deep networks, thereby improving learning stability and accuracy (Seo et al., 2024).

The structural characteristics of a model significantly affect its performance and suitability for a given dataset. Therefore, comparing architectures is essential for selecting the most suitable model for a specific task. In this study, three widely used segmentation models (U-Net, YOLOv8, and SegNet) are compared. Their architectural characteristics are summarized in Table 2. U-Net uses VGG16 pretrained with ImageNet as its backbone and features long skip connections. YOLOv8 uses a deep architecture based on C2f blocks and SPPF module, and includes PAN-FPN necks for multi-scale feature fusion. Upsampling is achieved through nearest neighbor interpolation followed by convolutional refinement, enabling efficient real-time processing. In contrast, SegNet uses VGG16 backbone and reconstructs spatial information using index-based max unpooling without explicit necks or skip connections. The three models also differ in network depth and activation functions. YOLOv8 uses SiLU, while both U-Net and SegNet use ReLU. These structural differences considerably affect the ability of each model to capture spatial detail and semantic information, as well as its computational efficiency.

4. Deep learning-based monitoring system

4.1. 3DFP monitoring system

The proposed real-time monitoring system uses an RGB camera to continuously capture images during the 3DFP process and detects defects based on the acquired images. The flow chart of the proposed system is shown in Fig. 4. The core of this system is a pretrained segmentation model that receives the acquired images as input and analyzes them to detect defects. When a defect is detected, the system provides detailed information about the type and exact location of the defect. With this information, users can promptly identify the type and location of the defect and adjust printing parameters or take other corrective actions immediately. Consequently, this system contributes to improving the overall quality of printed products and reducing the defect rate.

4.2. Deep learning-based defect detection

A key objective of this study is to compare three segmentation models, U-Net, YOLOv8, and SegNet, to determine the most suitable model for defect detection in 3DFP. Table 3 compares the application areas, advantages, and disadvantages of these three models. The following sections describe each model in detail based on the structural factors summarized in Table 2 and model-specific characteristics listed in Table 3.

4.2.1. U-Net

U-Net was introduced by researchers at the University of Freiburg, Germany, in 2015 for efficient segmentation of biomedical images. This model demonstrated high performance by winning the ISBI 2015 Challenge (Ronneberger et al., 2015). Since then, U-Net has been widely used as an image segmentation architecture, particularly in medical image modalities (Azad et al., 2024). This is because U-Net consists of a contracting path that captures context and a symmetric expanding path that facilitates accurate localization, providing pixel-level accuracy. Additionally, it achieves high performance even with limited data and is robust to noise through data augmentation. The network is fast, and can segment a 512 × 512 image in less than a second on a modern GPU (Ronneberger et al., 2015). However, the skip connections of U-Net require storing feature maps in memory, which increases memory

Table 1
Factors affecting segmentation model performance.

Factor	Options	Effect on performance
Backbone	- Shallow - Deep - Lightweight	- Deeper backbones facilitate complex feature extraction, leading to higher accuracy. - Lightweight models offer faster inference but may reduce precision.
Neck	- Feature Pyramid Network (FPN) - Path Aggregation Network -Bi-directional FPN	- Multiscale fusion supports recognition of objects with diverse sizes. - Particularly effective for segmenting small or large objects.
Skip Connection	- Long - Short	- Long skip connections preserve spatial detail and assist boundary reconstruction by linking encoder and decoder stages. - Short skip connections improve gradient flow and training stability, especially in deep networks. - Lack of skip connections may reduce reconstruction accuracy and hinder learning.
Decoder Upsampling Method	- Nearest Neighbor - Max Unpooling - Transposed Convolution	- Nearest neighbor is fast and simple but may produce blocky artifacts and lacks learnable detail restoration. - Max unpooling restores spatial position using encoder pooling indices but cannot recover lost semantic information. - Transposed convolution enables learnable upsampling and accurate boundary reconstruction but may introduce checkerboard artifacts.
Network Depth	- Shallow - Medium - Deep	- Moderate depth increases the receptive field, improving performance. - Excessive depth may cause overfitting or unstable training.
Activation Function	- Rectified Linear Unit (ReLU) - Leaky ReLU - Sigmoid Linear Unit (SiLU)	- SiLU improves accuracy and training stability by allowing smooth gradient flow. - ReLU is simple and stable but less expressive.

Table 2
Structural comparison of U-Net, YOLOv8, and SegNet.

	U-Net	YOLOv8	SegNet
Backbone	VGG16 pretrained on ImageNet	C2f blocks and SPPf	VGG16
Neck	None	Path Aggregation Network-Feature Pyramind Network structure	None
Skip Connection	Long skip connection	Short skip connection	None
Decoder Upsampling Method	Nearest Neighbor + Convolution	Nearest Neighbor + Convolution	Max Unpooling (Index-based)
Network Depth	29	More than 40	26
Activation Function	Rectified Linear Unit (ReLU)	Sigmoid Linear Unit	ReLU

usage and makes it less suitable for resource-constrained devices (Yin et al., 2024).

In this study, a modified U-Net architecture is used, in which the encoder is replaced by a VGG16 network pretrained on ImageNet. The U-Net architecture is shown in Fig. 5. Unlike the original U-Net that uses a custom contracting path composed of repeated convolution and pooling operations, the encoder in this version follows the standard VGG16 configuration with 13 convolutional layers. This model does not include explicit neck modules or multiscale fusion structures such as FPN or PAN. Instead, we directly connect the encoder and decoder via long skip connections. These skip connections concatenate the high-resolution feature maps of the encoder with the corresponding decoder layers, preserving spatial details and improving segmentation accuracy. The decoder restores the spatial resolution through a five-step upsampling process using nearest neighbor interpolation, and then refines the feature maps via convolution, batch normalization, and ReLU activation. This approach enables efficient feature refinement while maintaining structural simplicity. The entire network consists of 29 convolutional layers, with 13 in the encoder, 15 in the decoder, and a final 1×1 convolutional layer for pixel-wise classification. ReLU is used as the activation function throughout the network. ReLU is simple and stable, but may be less expressive than newer alternatives such as SiLU. Overall, this VGG16-based U-Net maintains the core advantages of the original design, such as precise localization and robust performance on limited data, while improving learning efficiency and generalization.

4.2.2. YOLOv8

You Only Look Once (YOLO) is an object detection algorithm that was first introduced in 2015. YOLOv8, developed by Ultralytics, is an advanced version of YOLOv5 with overall enhancements in performance. It supports various tasks such as object detection, classification, segmentation, tracking, and pose estimation, and is thus widely used in various fields such as healthcare, agriculture, and transportation. This is because YOLO provides pretrained models tailored to specific use

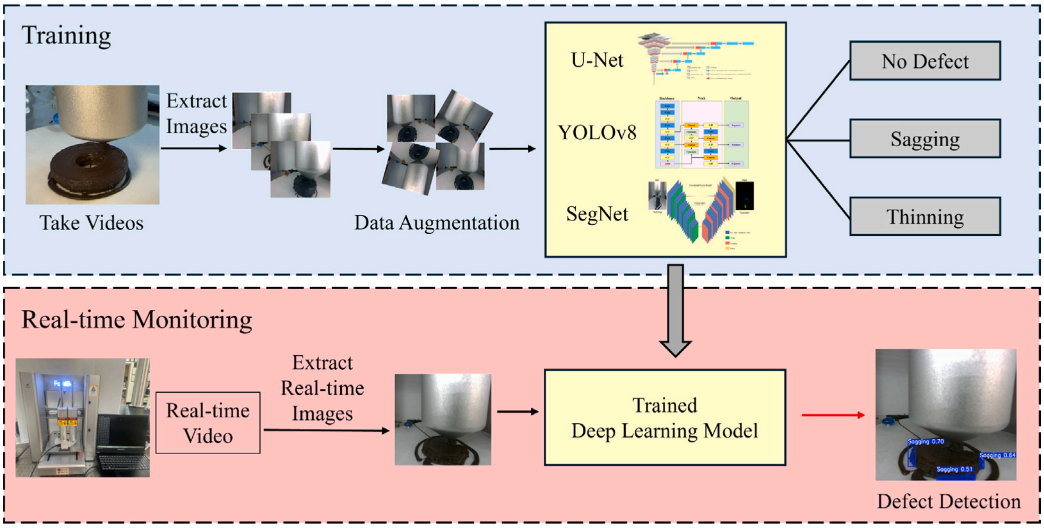

Fig. 4. Flow chart of the proposed 3DFP Monitoring System. It consists of a training stage that trains a deep learning model and real-time monitoring that performs real-time predictions using the trained model. Through training, the model learns sagging and thinning defects.

Table 3
Comparison of deep learning models.

Model	Application area	Advantages	Disadvantages
U-Net	Medical image segmentation	- High accuracy at the pixel level - Excellent performance even with limited training data - Fast training - Robustness to noise	- Increased memory consumption
YOLOv8	Applicable across various fields such as healthcare, transportation, and agriculture	- High accuracy and efficient detection - Suitable for real-time object detection - Optimized balance between accuracy and speed	- Difficulty in detecting small objects in complex scenes - High computational load and a large number of parameters
SegNet	Road scene understanding	- Uses pooling indices from the encoder to perform nonlinear upsampling in the decoder - Reduces computational cost by avoiding the need to learn upsampling filters	- Sensitive to low-resolution inputs, affecting segmentation accuracy - Limited performance in complex or noisy scenes

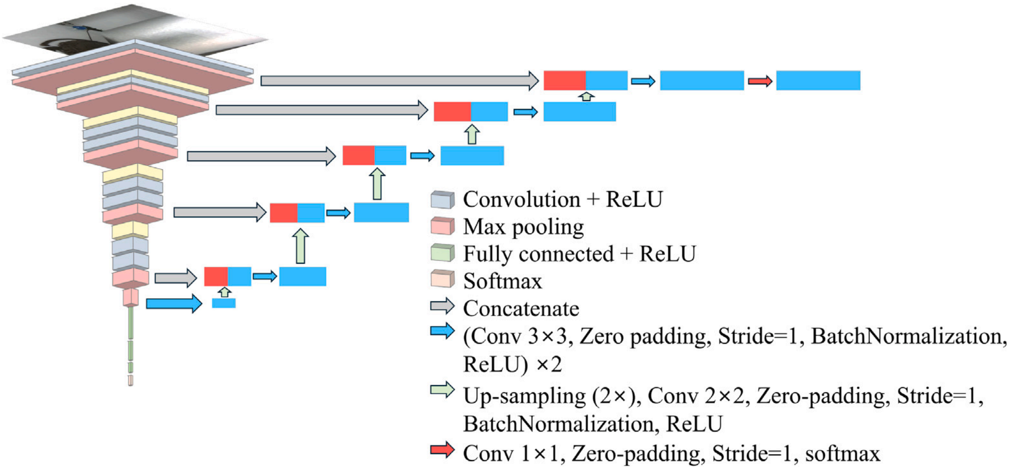
Convolution + ReLU
Max pooling
Fully connected + ReLU
Softmax
Concatenate
(Conv 3×3, Zero padding, Stride=1, BatchNormalization, ReLU) ×2
Up-sampling (2×), Conv 2×2, Zero-padding, Stride=1, BatchNormalization, ReLU
Conv 1×1, Zero-padding, Stride=1, softmax

Fig. 5. Architecture of U-Net. In this study, VGG16 pretrained with ImageNet was used as a backbone instead of an encoder.

cases and provides high accuracy and efficient detection capabilities. YOLOv8 is optimized to achieve a balance between accuracy and speed, making it suitable for real-time applications. However, it is computationally expensive owing to its deep architecture and large number

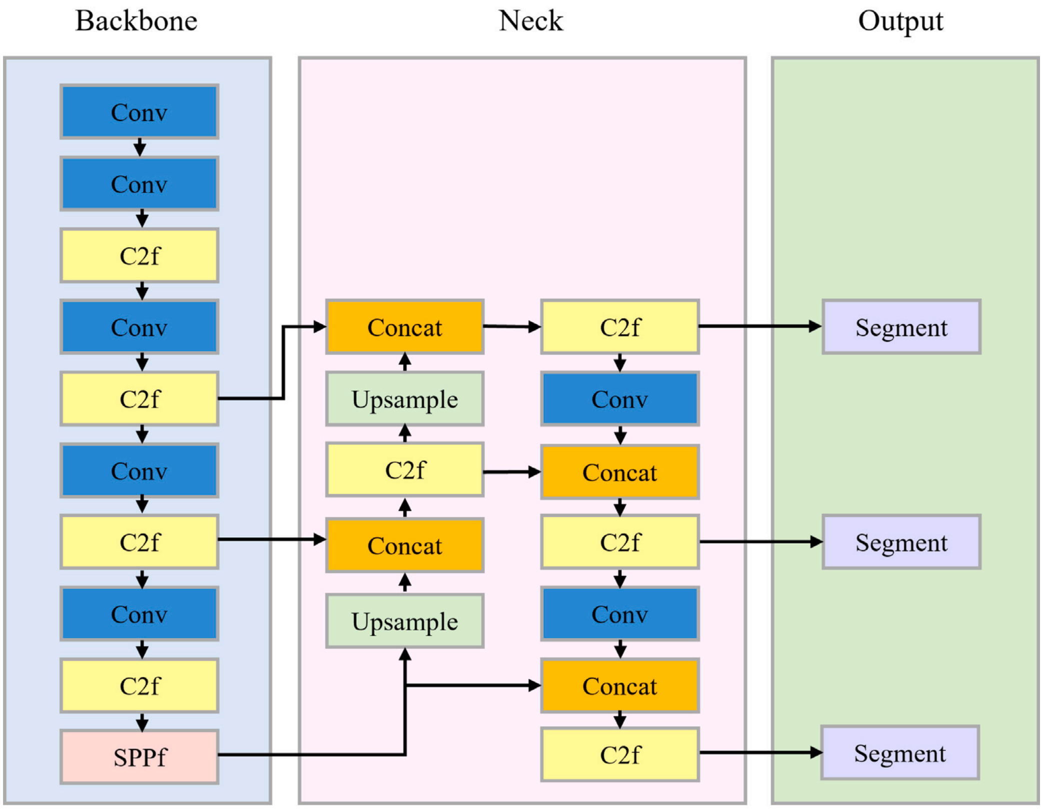

Fig. 6. Architecture of YOLOv8. It uses an asymmetric encoder–decoder architecture optimized for real-time performance.

of parameters. In addition, its performance may degrade in complex scenes, especially when detecting small objects, resulting in reduced accuracy and slower processing.

YOLOv8 is a deep CNN architecture designed primarily for object detection, but can also be applied to segmentation tasks. The YOLOv8 architecture is shown in Fig. 6. The model adopts an asymmetric encoder–decoder architecture optimized for real-time performance. The encoder extracts hierarchical features using C2f blocks and SPPF modules, efficiently capturing both local and global contexts. The decoder uses lightweight components for fast inference and integrates upsampling and skip connections to refine spatial resolution. The YOLOv8 backbone is based on C2f blocks and SPPF, which enable efficient feature extraction by reducing redundancy and enhancing the receptive field. YOLOv8 includes a dedicated neck module that adopts the PAN-FPN architecture to facilitate multiscale feature fusion. This fusion is particularly effective for detecting or segmenting objects of different sizes. YOLOv8 uses short skip connections, which is especially helpful for gradient flow and learning stability in deep networks. However, this design may be less effective in preserving spatial details compared with long skip connections. The decoder upsampling method combines nearest neighbor interpolation with convolutional layers, offering a balance between computational efficiency and boundary reconstruction accuracy. Although this strategy is fast, it may produce checkerboard artifacts in fine structures. With over 40 layers, YOLOv8 is considered a deep model that benefits from enhanced feature learning capacity. It uses the SiLU activation function, which has been shown to improve learning stability and accuracy by enabling smooth gradient flow. Overall, YOLOv8 combines short skip connections, PAN-FPN neck architecture, and the SiLU activation function, capable of efficiently handling multiscale information, rendering it promising for real-time visual recognition.

4.2.3. *SegNet*

SegNet is a deep convolutional encoder–decoder architecture proposed by researchers at the University of Cambridge in 2015, specifically designed for semantic segmentation tasks (Badrinarayanan et al., 2017). Since its introduction, SegNet has been widely applied in various practical fields such as road scene understanding for autonomous vehicles, medical image analysis, agriculture, and geographic information systems (Zhang et al., 2024). One of its key advantages is the use of pooling indices, which are stored during the max pooling stage of the encoder and reused during the decoder upsampling process. This approach improves both memory efficiency and computational speed while preserving boundary information, enabling more precise object descriptions. Benchmark evaluations on the CamVid and SUN RGB-D datasets have shown that SegNet achieves comparable mean intersection over union scores to models such as FCN and DeconvNet, while enabling faster inference with much less computational and memory resources (Badrinarayanan et al., 2017). However, its sensitivity to low-resolution feature maps during unpooling may lead to the loss of contextual information between neighboring pixels and affect segmentation accuracy. Additionally, the model tends to show limited performance in highly complex or noisy scenes, where fine structural details are essential (Zhang et al., 2024).

The SegNet architecture is shown in Fig. 7. The backbone of SegNet is derived from the convolutional layers of the pretrained VGG16 network, while the FC layers are removed to preserve spatial information. This design allows SegNet to achieve precise pixel-level segmentation while using the feature extraction capability of VGG16. SegNet does not use a neck structure. Instead, it uses the saved pooling indices from each max pooling operation in the encoder to help the decoder reconstruct spatial information. The decoder performs unpooling using the saved pooling indices, and then gradually restores the spatial resolution through the convolutional layers. This approach enables accurate segmentation while reducing computational costs. The network architecture inherits 13 convolutional layers from the VGG16 encoder

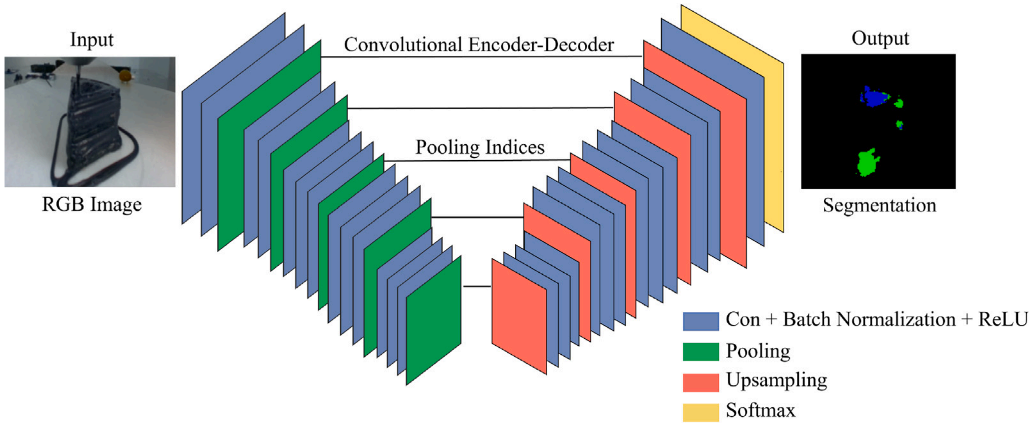

Fig. 7. Architecture of SegNet. It consists of a convolutional encoder–decoder architecture and uses VGG16 as a backbone like U-Net. In Output, blue indicates thinning defect, and green indicates sagging defect.

Table 4
Hardware specifications.

FoodBot 2CH	Dimensions (Length × Depth × Height)	420 mm × 381 mm × 400 mm
	Weight	20 kg
	Output Size	100 mm × 100 mm × 70 mm
	Nozzle Size	0.3-1.2 mm
	Printing Speed	15-70 mm/s
RealSense D405	Dimensions (Length × Depth × Height)	36.5 mm × 19.4 mm × 10.5 mm
	RGB FOV (H × V)	87 ° × 58 °
	RGB Frame Resolution	Up to 90 fps
	Ideal Range	7 cm to 50 cm

and incorporates an additional 13 mirrored convolutional layers in the decoder, resulting in a total of 26 convolutional layers. The ReLU activation function is used throughout the network to provide simple and stable nonlinearity. SegNet is known for its efficient memory usage and fast inference speed. It uses index-based unpooling to preserve spatial information and accurately reconstruct the segmentation map.

5. Experiment

The objectives of the experiment can be summarized as follows: (1) Evaluate the accuracy of defect detection during 3DFP using a deep learning model. (2) Compare the defect detection performances of the three deep-learning models to identify the most suitable model for 3DFP.

5.1. Experimental setup

In this study, a two-nozzle parallel-extrusion-based FOODBOT 2CH model and Intel's RealSense D405 model were used. The hardware specifications are summarized in Table 4. Various food materials, such as chocolate, cheese, and sweet potato, are supported. For this study, a nozzle size of 0.8 mm was used.

The environmental setup of the 3DFP process is shown in Fig. 8. To ensure stable image acquisition and minimize motion artifacts during printing, the 3DFP device was placed on a welding fixture table. This setup helps reduce mechanical vibrations that could occur during printer operation, particularly when the print head and plate are in motion. CNN-based multi-sensor fusion approaches can improve accuracy, despite increased computational cost, compared to individual sensor-based approaches (Li et al., 2022). However, in this paper, because the print design are low-height, a single camera was sufficient to secure a sufficient field of view, so we used single camera. Since the printer plate moves along the y-axis and the print head moves along the x-axis, it is difficult to maintain a consistent field of view with an external, fixed

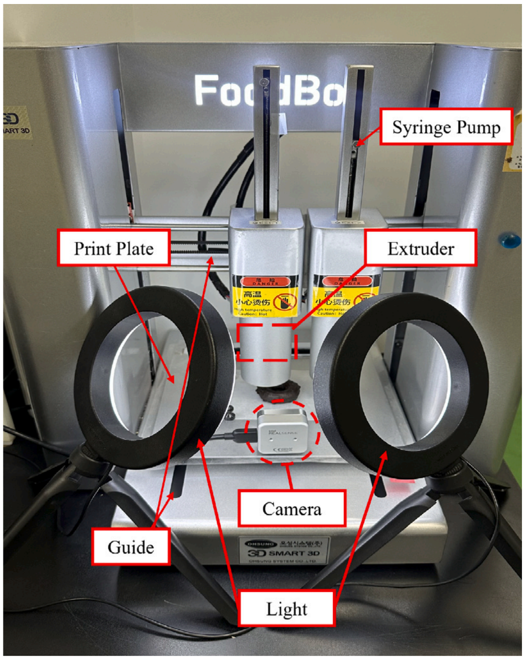

Fig. 8. 3DFP setup for dataset collection and experiments. The camera was attached to the printer plate to minimize interference from the head, and white LEDs were installed on both sides to reduce illumination influence.

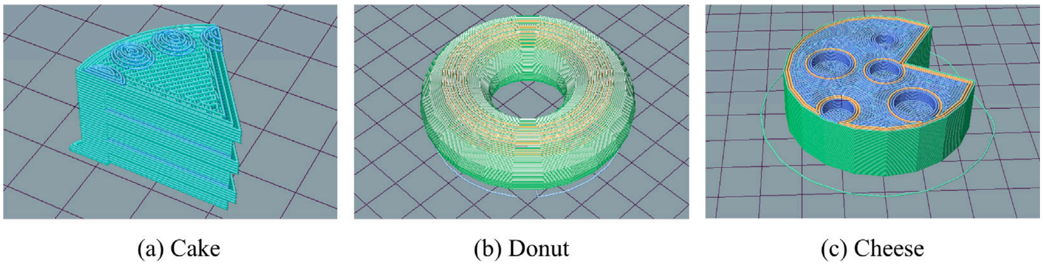

Fig. 9. Designs used in dataset collection and experiments: (a) Cake, (b) Donut, (c) Cheese. These designs were used to print realistic food models.

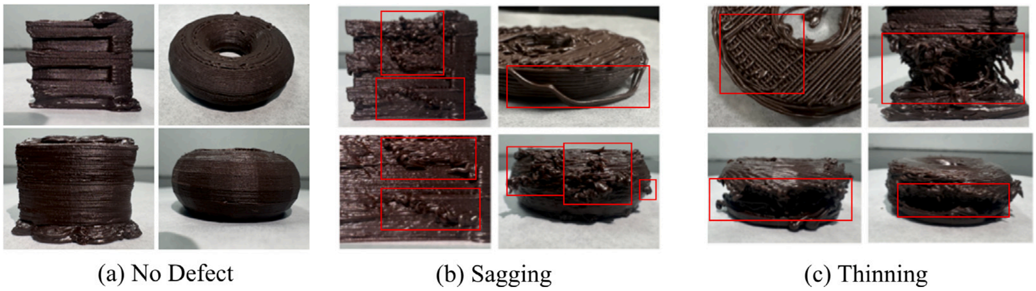

Fig. 10. Defined defect types: (a) No Defect, (b) Sagging, (c) Thinning. Sagging and thinning are the two most common defects in 3D food printing and are sensitive to temperature and printing speed.

camera. Therefore, the camera was mounted on the plate to ensure the print was positioned consistently within the camera's coordinate system. This configuration helps maintain a constant distance between the camera and printed object, while minimizing visual obstruction caused by the movement of the print head. The camera continuously captured images of the printing process in real time and transferred them to a computer via USB connection, where a monitoring algorithm processed the data. To ensure consistent illumination and high visibility of the printed material within the captured frames, two white LED were installed symmetrically on both sides of the printer. This illumination setup helps eliminate shadows and reflections, allowing the segmentation model to better identify fine details and printing defects.

5.2. Material selection and preparation

The material used in the experiment was dark coating chocolate. Chocolate remains solid at room temperature (20 to 25 °C) and melts at body temperature, allowing it to dissolve easily in the mouth (37 °C) (Windhab, 2006). Consequently, chocolate is a highly suitable material for extrusion-based 3DFP (Mawardi et al., 2023) and frequently used to demonstrate the capabilities of 3DFP (Karyappa and Hashimoto, 2019). However, chocolate is highly sensitive to temperature owing to changes in its rheological properties and crystallization behavior depending on the tempering process. Therefore, maintaining a constant temperature and pre-tempering the chocolate are necessary to achieve uniform printing (Kim et al., 2017). For the above reasons, chocolate, which is prone to defects during 3DFP, was selected as the experimental material in this study. Specifically, this study used coating chocolate, which replaced cocoa butter with a fat such as palm kernel oil. Therefore, the tempering required for cocoa butter to obtain the β -V polymorph was unnecessary for coating chocolate (Lipkin et al., 2025). The experiments were conducted after melting the chocolate at 34 °C for 1 h before printing.

5.3. Experimental design

As the objective of this experiment was to print realistic food models, the designs used for the experiment included a cake, donut, and cheese, as shown in Fig. 9. The designs were printed using Repetier-Host software and Slic3r slicer. Fig. 10a is a normal output without defects. Defects were classified into two categories: sagging (Fig. 10b) and thinning (Fig. 10c). Sagging occurs when the food material flows unevenly from the nozzle or the output is extracted beyond the target range, resulting in an uneven surface. This defect mainly arises when the heating temperature of the food material is excessively high. Thinning occurs when the food material is cut-off or printed very thinly. This interferes with the stable settlement of the next layer, and in severe cases, may lead to the collapse of the printed matter. This defect usually occurs when the food materials are insufficiently heated.

The experiment was conducted under three varying parameters: extruder temperature, printing speed, and infill pattern. Among the printing speed parameters, the infill speed refers to the speed used for filling the infill pattern, while the default speed indicates the speed applied to all other movements. Each parameter was tested at three values as shown in Fig. 11, resulting in 27 experimental cases. As each of the three designs was printed under all 27 cases, 81 samples were created. To minimize potential biases in the experimental results, the 81 samples were printed in triplicate in random order. The chocolate was melted at 34 °C for 1 h and then heated to the extruder temperature corresponding to each case for printing. All other conditions, including the infill pattern's filling density, were set to be identical.

5.4. Dataset

The first step in training a deep-learning model is the generation of a dataset, which is critical for ensuring robust model performance. To create an accurate dataset, image data were collected by direct printing. This approach allows training data to capture a wide range of defects that occur under practical operating conditions. The final collected dataset included 608 images in the training set and 173 images for validation. Subsequently, the dataset was subjected to manual labeling, where a person directly annotated the image with defect locations and names. Features were automatically extracted and learned from the labeled data, and each model learned the defect and normal classes.

To enhance the diversity and size of the dataset, data augmentation techniques were used. These techniques improve the generalizability of

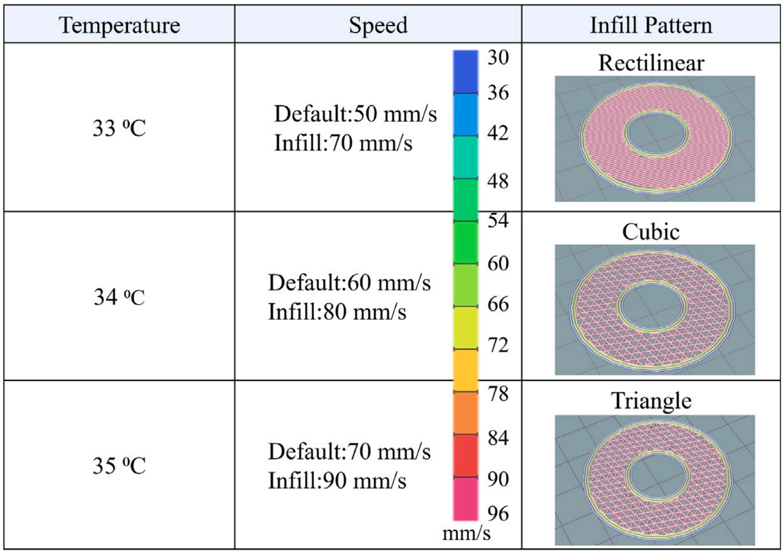

Fig. 11. Experimental cases according to temperature, speed, and infill pattern. Since these varying parameters each have three values, a total of 27 cases are created. Since the three designs are printed under the 27 cases, a total of 81 samples are created. This enables the collection of various datasets. Speeds in the infill pattern image are set to default: 70 mm/s, infill: 90 mm/s.

Table 5
Evaluation metrics.

Task	Item	Explanation	Metrics
Model Performance	1. Accuracy	Proportion of correctly predicted values over the entire dataset	Acc
	2. Precision	Proportion of predicted positives that are actually positive	Prec
	3. F1-Score	Balance between Precision and Recall	F1
	4. Sensitivity	Proportion of actual positives that are correctly predicted	TPR
	5. Specificity	Proportion of actual negatives that are correctly predicted	TNR
	6. ROC Curve	Plot illustrating the trade-off between true positive rate and false positive rate	
Segmentation Quality	7. Intersection over Union	Extent of overlap between the predicted and actual regions	IoU
	8. Hausdorff Distance	Maximum distance between the predicted and actual boundaries	HD

the model by simulating various environmental and operational conditions. In this study, three data augmentation techniques, i.e., horizontal flipping, rotation, and cropping were applied. Horizontal flipping was applied with a probability of 50%, enabling the model to learn symmetric features. Rotation was also applied with a 50% probability, with images randomly rotated within a range of −15 ° to +15 °. This process enhanced the model's ability to recognize slightly rotated objects and improved robustness to camera tilt during image acquisition. Lastly, the cropping technique randomly selected more than 80% of the original image area and resized it to a fixed size using nearest neighbor interpolation. This allowed the model to learn objects at various position, thereby improving robustness to spatial variations.

5.5. Evaluation metrics

The model was evaluated in terms of performance and segmentation quality. The evaluation metrics are summarized in Table 5.

Accuracy (Acc) represents the proportion of correct predictions and is one of the most widely used evaluation metrics in multiclass classification. In multiclass segmentation, each pixel is treated as an individual classification instance, with accuracy calculated as the proportion of pixels for which the predicted class label matches the ground truth label. It is calculated directly from the confusion matrix (Grandini et al., 2020). True positive (TP) means that the model correctly predicts a positive outcome for a positive instance. True negative (TN) means that the model correctly predicts a negative outcome for a negative instance. False positive (FP) means that the model incorrectly predicts a positive outcome for a negative instance. False negative (FN) means that the model incorrectly predicts a negative outcome for a positive instance. The accuracy is calculated using Eq. (1).

$$Acc = \frac{TP + TN}{TP + FP + FN + TN} \tag{1}$$

Precision (Prec) represents the proportion of elements predicted as positive by the model that are actually positive. In other words, precision indicates the reliability of the model's predictions when it identifies an instance as positive. This metric is especially important in situations where FPs must be minimized, such as in medical diagnoses or quality inspection systems. Precision is calculated as the number of TPs divided by the total number of elements predicted as positive by the model, as indicated in Eq. (2).

$$Prec = \frac{TP}{TP + FP} \tag{2}$$

The F1-score (F1) combines precision and recall through the harmonic mean, as defined in Eq. (3). It can be interpreted as a balanced measure that reflects both precision and recall. The F1-score ranges from 0 to 1, where 1 indicates perfect precision and recall, and 0

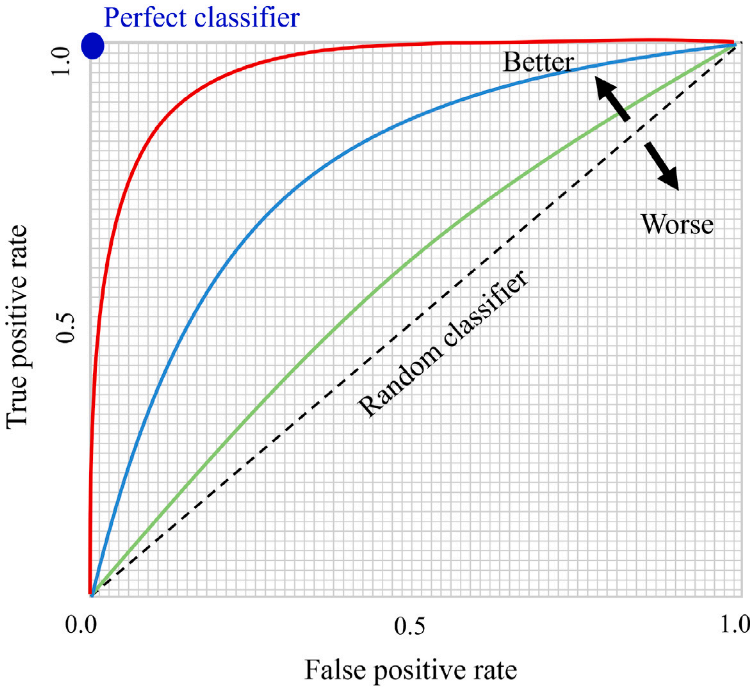

Fig. 12. Interpretation of the ROC curve. This compares classifiers with different performances. The closer to the upper left corner, the better the performance. The area under the curve can have values between 0 and 1, with closer to 1 indicating better performance.

indicates the worst performance.

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{3}$$

Sensitivity (TPR), also known as recall or the TP rate, measures the model's ability to correctly identify positive instances. It reflects how well the model detects actual positive cases without missing them. Sensitivity is particularly important in applications where failing to detect a positive case can be costly or dangerous, such as in medical diagnostics or industrial automation systems. It is calculated as the number of TPs divided by the total number of actual positive elements, as described in Eq. (4).

$$TPR = \frac{TP}{TP + FN} \tag{4}$$

Specificity (TNR) refers to the model's ability to correctly identify negative cases. This metric is particularly important when distinguishing normal regions or nondefective outputs in applications such as medical imaging or anomaly detection. Specificity is calculated as the number of TNs divided by the total number of actual negatives, as indicated in Eq. (5). High specificity indicates that the model effectively avoids false alarms by correctly rejecting negative cases.

$$TNR = \frac{TN}{TN + FP} \tag{5}$$

The receiver operating characteristic (ROC) curve measures the performance of a model at all possible thresholds, indicating the theoretical classification ability of a model. The curve is suitable for evaluating the discriminatory power of a model by visualizing the trade-off between TPR (Eq. (4)) and false positive rate (FPR, Eq. (6)). As shown in Fig. 12, the diagonal line represents the performance of a random classifier. Curves above this line represent models with some degree of discriminatory power. A model with the ROC curve closer to the top-left corner demonstrates superior performance, as it achieves a high TPR while maintaining a low FPR. The area under the curve (AUC) represents the area under the curve obtained by integrating the ROC curve. It ranges from 0 to 1, with a value closer to 1 indicating superior performance. An AUC of 0.5 corresponds to a performance similar to that of random classification.

$$FPR = \frac{FP}{FP + TN} \tag{6}$$

The segmentation performance of models is traditionally reported using overlap-based metrics such as the intersection over union (IoU) (Kolbinger et al., 2024). IoU, also known as the Jaccard index or Jaccard similarity coefficient, assesses the region similarity and is thus highly sensitive to under-segmentation and over-segmentation. It is defined as the intersection area of the predicted segmentation map and ground truth divided by their union, as indicated in Eq. (7), where A represents the set of pixels predicted to belong to a certain class, and B denotes the corresponding ground truth set.

$$IoU = \frac{|A \cap B|}{|A \cup B|} \tag{7}$$

The Hausdorff distance (HD) is considered one of the most valuable evaluation metrics, as it effectively reflects the maximum segmentation error (Karimi and Salcudean, 2019). HD evaluates the boundary accuracy by measuring the distance between the predicted and ground truth boundaries. As shown in Fig. 13, the IoU only shows the overlapping part of two regions and thus cannot account for cases where the boundary error is large. However, the HD can effectively identify such instances, as shown in Fig. 13. Therefore, when the size of the actual boundary is extremely small, HD is used as an effective evaluation metric because it highlights scenarios where models may achieve high accuracy by predicting background classes but fail to properly segment the region of interest. Equations (8) and (9) define the one-sided HD distances. X and Y denote the sets of points on the predicted and ground truth boundaries, respectively. $x \in X$ represents an individual point on the predicted boundary, and $y \in Y$ denotes a point on the ground truth boundary. Eq. (8) calculates the minimum distance from each point x in set X to the closest point y in set Y and returns the maximum of the minimum distances. Similarly, Eq. (9) calculates the minimum distance from each point y in set Y to the closest point x in set X and returns the maximum of the minimum distances. Eq. (10) defines

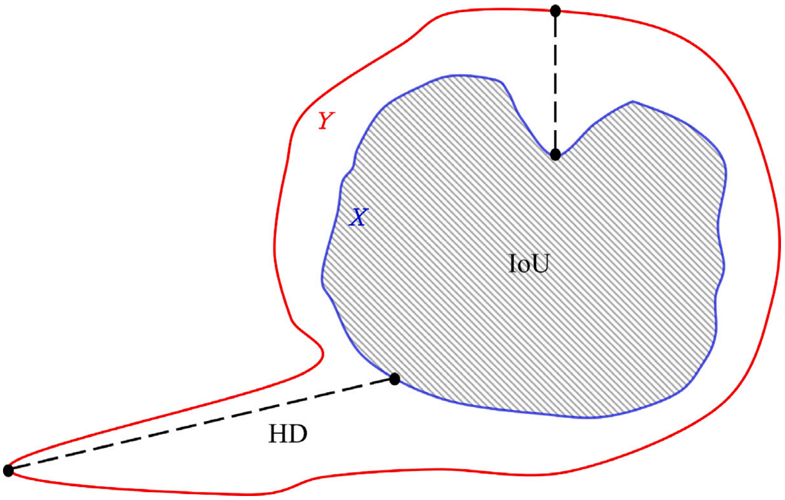

Fig. 13. Schematic representations of IoU and HD. IoU represents area accuracy, and HD represents boundary accuracy. HD can be useful when the prediction area is very wide compared to the ground truth, as shown in this figure.

Table 6
AUC by model and class.

	AUC		
	Sagging	Thinning	Mean
U-Net	0.973	0.968	0.971
YOLOv8	0.872	0.859	0.866
SegNet	0.968	0.980	0.974

the bidirectional HD distance, which takes the larger value between the two one-sided HD distances.

$$\mathrm{hd}(X,Y) = \max_{x\in X}\min_{y\in Y}\|x-y\|_2, \tag{8}$$

$$\mathrm{hd}(Y,X) = \max_{y\in Y}\min_{x\in X}\|x-y\|_2. \tag{9}$$

$$\mathrm{HD}(X,Y) = \max(\mathrm{hd}(X,Y),\mathrm{hd}(Y,X)) \tag{10}$$

5.6. Experimental results

5.6.1. Model performance

Fig. 14 shows the ROC curves of each model for the two defect types, sagging and thinning. Table 6 lists the AUC values of each model. Based on average AUC, the theoretical classification ability exhibited the following decreasing order: SegNet, U-Net, and YOLOv8. However, as the average AUCs of SegNet (0.974) and U-Net (0.971) were similar, both models were considered to have excellent theoretical classification ability. YOLOv8 displayed a lower AUC, indicating weaker theoretical classification ability. Notably, U-Net demonstrated the best performance in the sagging class, while SegNet outperformed the other two models in the thinning class.

However, the actual performance was different from the theoretical classification ability. Fig. 15 shows the prediction results over the test dataset. The predictions of all models were similar to the ground truth. Fig. 16 shows the change in model accuracy over the training process. All three models showed a high accuracy of over 0.98, with U-Net attaining the highest accuracy. YOLOv8 showed an overall stable accuracy trend, while SegNet showed more significant fluctuations. This is because SegNet uses VGG16, which was not pretrained, as its backbone and trains with a significantly higher weight on the thinning class to address the imbalanced dataset.

The assessment of model performance based on the evaluation metrics is outlined in Table 7. Overall, all models detected sagging better than thinning. This is because the number of thinning samples in the training dataset was considerably smaller than that of the sagging samples. U-Net recorded the best performance across all indices. In contrast, SegNet displayed the lowest performance across all indices except TPR, with significant variations from the other models. As the ROC curve shows the results at all possible thresholds instead of a specific threshold, it is likely that the threshold value used for prediction was not optimal. Therefore, to use SegNet for real-time monitoring, threshold optimization must be performed. YOLOv8 had extremely low TPR but overall average performance. This means that although it could effectively distinguish normal classes, it was more likely to misclassify defects as normal.

5.6.2. Segmentation quality

Table 8 presents the results of the segmentation quality evaluation metrics for each model. Similar to the model performance evaluation results, U-Net demonstrated the best performance across all indices. In particular, U-Net achieved an HD of 15.36, less than half those of YOLOv8 and SegNet. This indicates that the boundary accuracy of U-Net was considerably higher than that of other models. SegNet recorded the lowest IoU in both the sagging and thinning classes, while YOLOv8 displayed the largest HD value.

6. Discussion

6.1. Explainable deep learning for 3DFP

One potential approach to enhance the performance of deep learning models is to integrate of explainable artificial intelligence (XAI) techniques. A persistent challenge in deep learning is the lack of transparency in its decision-making processes, which is caused by its black-box nature. XAI addresses this issue by offering explainability, allowing users to evaluate how and why a model makes a particular decision (Budhkar et al., 2025). Rather than relying solely on accuracy-based comparisons to determine the suitable model, understanding the reasoning behind each model's predictions can allow for a more informed evaluation of the logic that is best suited for a particular task. By analyzing feature importance and decision paths, XAI provides transparency, facilitating the selection of a model that not only achieves high accuracy but also demonstrates logical consistency and reliability. Applying XAI to 3DFP monitoring makes it possible to evaluate how each model determines the defect class. This can facilitate the selection of the optimal model for 3DFP monitoring by comparing specific decision-making processes rather than simply focusing on evaluation indices.

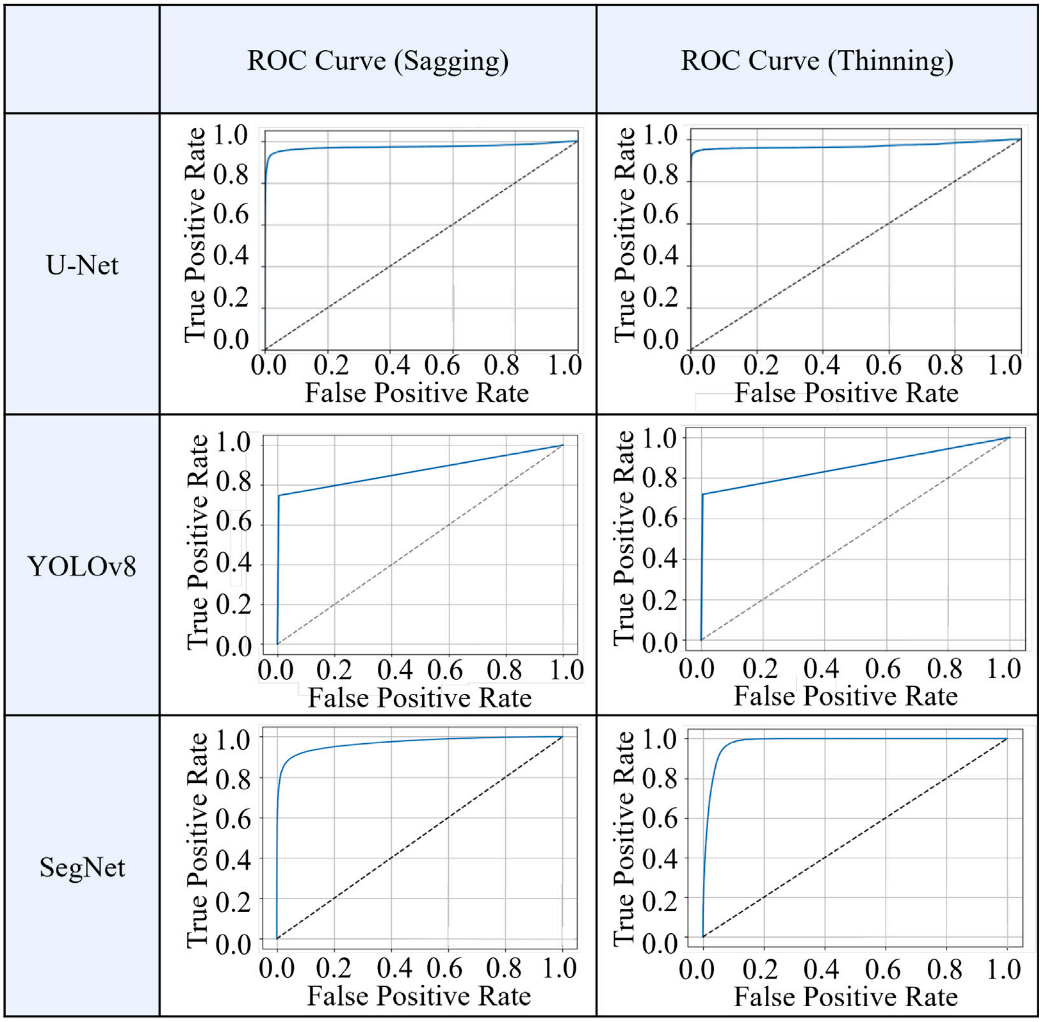

Fig. 14. ROC curves by model and class. This demonstrates theoretical classification ability.

Table 7
Comparison of model performance.

Model	Acc	Prec		F1		TPR		TNR	
		Sagging	Thinning	Sagging	Thinning	Sagging	Thinning	Sagging	Thinning
U-Net	0.998	0.736	0.701	0.837	0.798	0.970	0.926	0.998	1.000
YOLOv8	0.996	0.655	0.613	0.703	0.648	0.757	0.688	0.998	1.000
SegNet	0.993	0.455	0.329	0.570	0.446	0.762	0.694	0.995	0.999

Table 8
Comparison of segmentation quality across models.

Model	IoU		HD
	Sagging	Thinning	
U-Net	0.720	0.664	15.362
YOLOv8	0.541	0.479	49.821
SegNet	0.399	0.446	34.084

Post hoc explanation allows for the analysis of already trained models to gain insights into the internal relationships and patterns they have learned. XAI techniques applicable to CNNs can be broadly categorized into backpropagation-based and perturbation-based methods. Backpropagation-based methods generate explanations by performing a single forward and backward pass of the network. Grad-CAM is one of the most representative techniques in this category. In contrast, perturbation-based methods evaluate the contribution of input features by selectively removing, masking, or changing part of the original input and measuring the resulting changes in the model output. A representative example of this approach is occlusion sensitivity (Van der Velden et al., 2022).

6.2. Generalizable deep learning for 3DFP

Although the proposed deep learning-based monitoring system was trained and evaluated exclusively using chocolate as the printing material, the methodology has potential for broader generalization. Chocolate is commonly used as a food ink owing to its temperature sensitivity and well-studied rheological properties. However, to be considered generalizable, the model should be evaluated on a wider range of food materials with diverse physical properties. Future work should expand the dataset to include food inks such as surimi and plant-based meat analogue that exhibit significant differences in viscosity and extrusion properties. By incorporating multiple food ink types during model

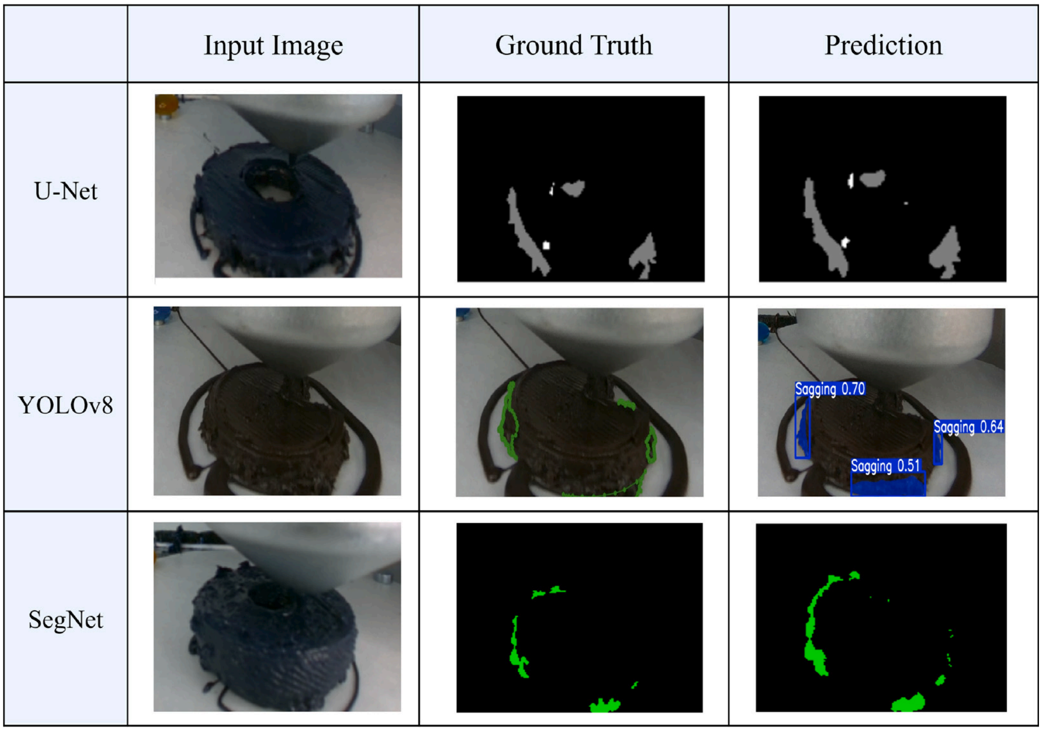

Fig. 15. Test dataset prediction results. Ground truth is the defect region that was directly labeled by a human. All three models predicted similarly to the ground truth.

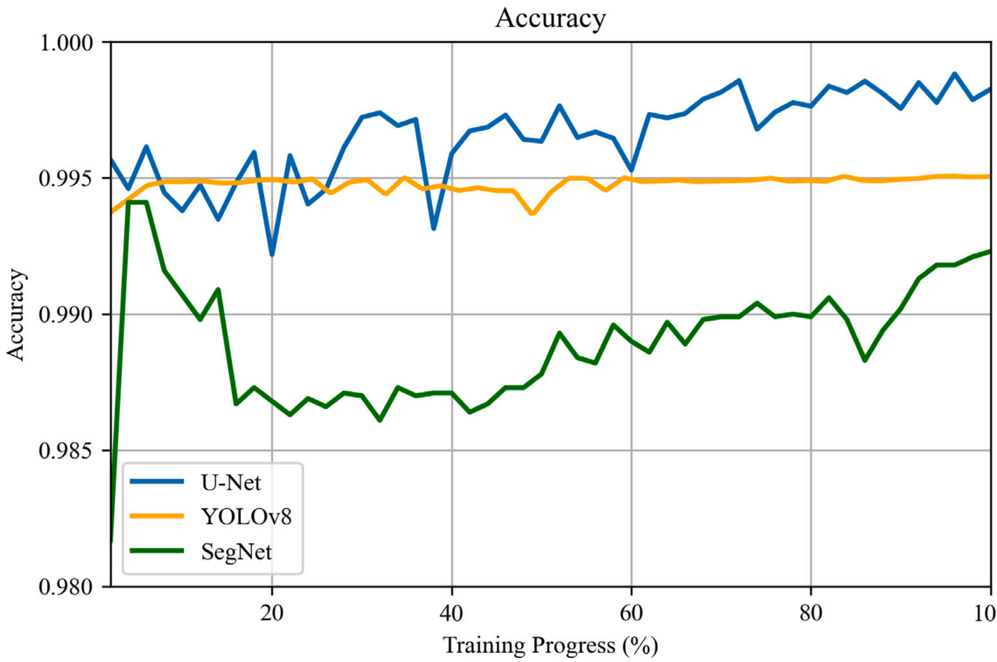

Fig. 16. Comparison of accuracy trends during training. All three models show high accuracy, but U-Net is the best and SegNet is the most unstable.

training and evaluation, a more robust and generalizable segmentation model can be developed that can adapt to diverse 3DFP environments.

6.3. *Feedback-based printer control system*

Although various monitoring systems have been developed to reduce the material and time waste due to printing defects, most 3DFP systems currently operate in an open-loop control mode. In such systems, even if a defect is detected, the printer cannot autonomously adjust its parameters or recover from the error. Consequently, human intervention is required to modify settings or restart the printing process, which causes delays and increases human dependency. Real-time feedback control is essential to overcome these limitations.

Fig. 17 illustrates an example of a feedback control system in a 3DFP context, also known as an Intelligent 3DFP system. The intelligent 3D food printing system can improve the quality of 3D food

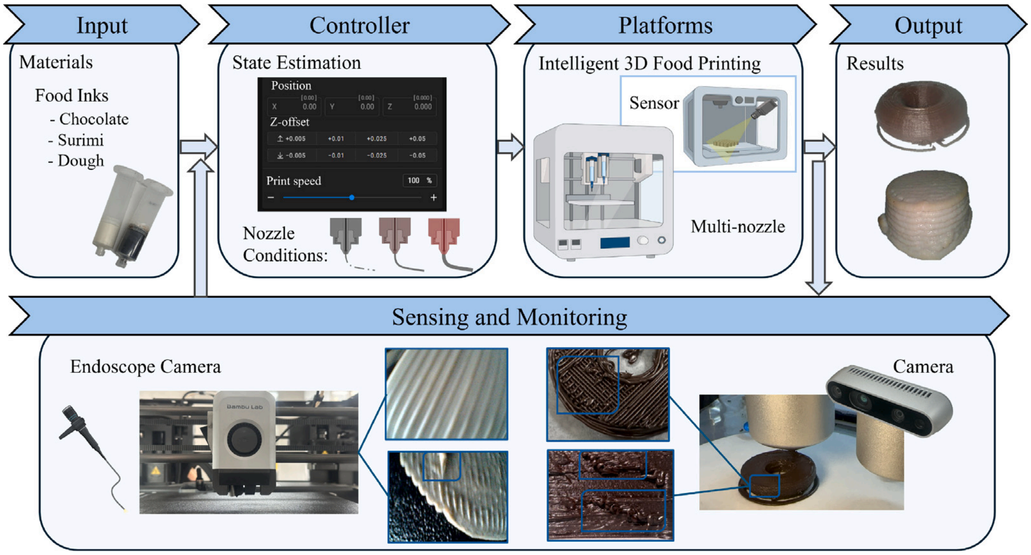

Fig. 17. Feedback-based printer control system. In other words, it is called an intelligent 3D food printing system. When a defect is detected, it automatically adjusts the parameters to enable stable printing.

printing by optimizing multiple parameters simultaneously through a real-time feedback system. Our group has written a review paper on our knowledge of intelligent 3DFP (Seol et al., 2025). The Feedback system enables the printer to continuously monitor the printing process and immediately adjust relevant parameters when a defect is detected. Typical control mechanisms include nozzle control and device motion control, with nozzle control being the most commonly applied in extrusion-based 3DFP. Integrating a deep learning-based defect detection system with a feedback control system could facilitate fully automated correction processes. For example, if a model identifies a particular type of defect, the system can infer optimal corrective action, such as adjusting extrusion pressure, nozzle temperature or printing speed, without human intervention. This closed-loop integration would not only improve accuracy and efficiency, but also lay the foundation for a truly autonomous 3DFP system.

7. Conclusion

This paper proposes a real-time monitoring system using deep learning for defect detection in 3DFP. A segmentation model was used to inform users of specific defect locations and types, and the performance of three models, U-Net, YOLOv8, and SegNet, was compared to identify the suitable model. A dataset was constructed by repeatedly printing 27 chocolate samples under different environmental conditions, and the models were compared using performance indices and segmentation quality. The results of a comprehensive analysis revealed that all three models were more effective at detecting sagging defects than thinning defects, likely owing to the smaller number of thinning defects in the dataset. U-Net showed the best and most stable performance across all indices. Notably, it demonstrated excellent performance for thinning defects, unlike other models. Also, it recorded the highest IoU and lowest HD value, indicating excellent area and boundary accuracies. Thus U-Net was selected as the suitable model for 3DFP monitoring. YOLOv8 showed moderate performance overall but recorded low sensitivity and boundary accuracy. Therefore, YOLOv8 may be preferable in scenarios where detection speed is prioritized over accuracy. SegNet showed the best theoretical classification ability with the highest AUC value.

CRediT authorship contribution statement

Eunji Ju: Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Jaehwi Seol:** Writing – review & editing, Methodology, Conceptualization. **Sarang Kim:** Writing – original draft, Software, Formal analysis. **Sol Kim:** Writing – review & editing. **Soo-Jung Kim:** Writing – review & editing, Supervision, Funding acquisition. **Hyoung Il Son:** Writing – review & editing, Supervision, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by Korea Institute of Planning and Evaluation for Technology in Food, Agriculture and Forestry(IPET) through Agriculture and Food Convergence Technologies Program for Research Manpower development, funded by Ministry of Agriculture, Food and Rural Affairs(MAFRA) (RS-2024-00402136).

Data availability

Data will be made available on request.

References

Alzubaidi, L., Zhang, J., Humaidi, A.J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M.A., Al-Amidie, M., Farhan, L., 2021. Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. J. Big Data 8, 1–74.

Azad, R., Aghdam, E.K., Rauland, A., Jia, Y., Avval, A.H., Bozorgpour, A., Karimijafarbigloo, S., Cohen, J.P., Adeli, E., Merhof, D., 2024. Medical image segmentation review: The success of u-net. IEEE Trans. Pattern Anal. Mach. Intell.

Badrinarayanan, V., Kendall, A., Cipolla, R., 2017. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 39 (12), 2481–2495.

Bhandarkar, V.V., Das, B., Tandon, P., 2025. Real-time remote monitoring and defect detection in smart additive manufacturing for reduced material wastage. Measurement 252, 117362.

Brion, D.A., Pattinson, S.W., 2022. Generalisable 3D printing error detection and correction via multi-head neural networks. Nat. Commun. 13 (1), 4654.

Budhkar, A., Song, Q., Su, J., Zhang, X., 2025. Demystifying the black box: A survey on explainable artificial intelligence (XAI) in bioinformatics. Comput. Struct. Biotechnol. J.

Chan, H.-P., Samala, R.K., Hadjiiski, L.M., Zhou, C., 2020. Deep learning in medical image analysis. In: Deep Learning in Medical Image Analysis: Challenges and Applications. Springer, pp. 3–21.

Erokhin, K., Naumov, S., Ananikov, V., 2023. Defects in 3D printing and strategies to enhance quality of FFF additive manufacturing. A review.

Godoi, F.C., Prakash, S., Bhandari, B.R., 2016. 3D printing technologies applied for food design: Status and prospects. J. Food Eng. 179, 44–54.

Grandini, M., Bagli, E., Visani, G., 2020. Metrics for multi-class classification: an overview. arXiv preprint arXiv:2008.05756.

Hsiang Loh, G., Pei, E., Gonzalez-Gutierrez, J., Monzón, M., 2020. An overview of material extrusion troubleshooting. Appl. Sci. 10 (14), 4776.

Ichimori, I., Watanabe, Y., Fujiwara, K., Ogawa, J., Furukawa, H., 2024. Real-time material extrusion volume control system for 3D food printers.

Jiang, H., Zheng, L., Zou, Y., Tong, Z., Han, S., Wang, S., 2019. 3D food printing: Main components selection by considering rheological properties. Crit. Rev. Food Sci. Nutr. 59 (14), 2335–2347.

Kadam, V., Kumar, S., Bongale, A., Wazarkar, S., Kamat, P., Patil, S., 2021. Enhancing surface fault detection using machine learning for 3D printed products. Appl. Syst. Innov. 4 (2), 34.

Karimi, D., Salcudean, S.E., 2019. Reducing the hausdorff distance in medical image segmentation with convolutional neural networks. IEEE Trans. Med. Imaging 39 (2), 499–513.

Karyappa, R., Hashimoto, M., 2019. Chocolate-based ink three-dimensional printing (Ci3DP). Sci. Rep. 9 (1), 14178.

Khan, M.F., Alam, A., Siddiqui, M.A., Alam, M.S., Rafat, Y., Salik, N., Al-Saidan, I., 2021. Real-time defect detection in 3D printing using machine learning. Mater. Today: Proc. 42, 521–528.

Kim, K.E., Park, K., Lee, C., 2017. A study on development of three-dimensional chocolate printer. J. Korean Soc. Precis. Eng. 34 (4), 293–298.

Kolbinger, F.R., He, J., Ma, J., Zhu, F., 2024. Strategies to improve real-world applicability of laparoscopic anatomy segmentation models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2275–2284.

LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. Nature 521 (7553), 436–444.

Li, J., Zhou, Q., Cao, L., Wang, Y., Hu, J., 2022. A convolutional neural network-based multi-sensor fusion approach for in-situ quality monitoring of selective laser melting. J. Manuf. Syst. 64, 429–442.

Lipkin, E.M., Cohen, M.L., Hartel, R.W., 2025. Microstructure and rheology of chocolate. Annu. Rev. Mater. Res. 55.

Lv, Y., Lv, W., Li, G., Zhong, Y., 2023. The research progress of physical regulation techniques in 3D food printing. Trends Food Sci. Technol. 133, 231–243.

Ma, Y., Potappel, J., Chauhan, A., Schutyser, M.A., Boom, R.M., Zhang, L., 2023. Improving 3D food printing performance using computer vision and feedforward nozzle motion control. J. Food Eng. 339, 111277.

Mawardi, C., Buono, A., Priandana, K., et al., 2023. Real-time defect detection monitoring system for extrusion-based 3D food printing using internet of things. In: 2023 3rd International Conference on Smart Cities, Automation & Intelligent Computing Systems. ICON-SONICS, IEEE, pp. 109–113.

Mawardi, C., Buono, A., Priandana, K., et al., 2024. Performance analysis of ResNet50 and inception-V3 image classification for defect detection in 3D food printing. Int. J. Adv. Sci. Eng. Inf. Technol. 14 (2).

Niakousari, M., Hedayati, S., Tahsiri, Z., Mirzaee, H., 2019. Overview on the food industry and its advancement. In: Food Tech Transitions: Reconnecting Agri-Food, Technology and Society. Springer, pp. 23–47.

O'Mahony, N., Campbell, S., Carvalho, A., Harapanahalli, S., Hernandez, G.V., Krpalkova, L., Riordan, D., Walsh, J., 2020. Deep learning vs. traditional computer vision. In: Advances in Computer Vision: Proceedings of the 2019 Computer Vision Conference (CVC), Volume 1 1. Springer, pp. 128–144.

Phuhongsung, P., Zhang, M., Devahastin, S., Mujumdar, A.S., 2022. Defects in 3D/4D food printing and their possible solutions: A comprehensive review. Compr. Rev. Food Sci. Food Saf. 21 (4), 3455–3479.

Portanguen, S., Tournayre, P., Sicard, J., Astruc, T., Mirade, P.-S., 2019. Toward the design of functional foods and biobased products by 3D printing: A review. Trends Food Sci. Technol. 86, 188–198.

Rachmawati, S.M., Putra, M.A.P., Lee, J.M., Kim, D.S., 2023. Digital twin-enabled 3D printer fault detection for smart additive manufacturing. Eng. Appl. Artif. Intell. 124, 106430.

Rettenberger, L., Beyer, N., Sieber, I., Reischl, M., 2024. Fault detection in 3D-printing with deep learning. In: 2024 IEEE International Conference on Consumer Electronics. ICCE, IEEE, pp. 1–4.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation. In: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18. Springer, pp. 234–241.

Sani, A.R., Zolfagharian, A., Kouzani, A.Z., 2024. Artificial intelligence-augmented additive manufacturing: insights on closed-loop 3D printing. Adv. Intell. Syst. 6 (10), 2400102.

Seo, Y., Kim, J., Park, U., 2024. Swish-T: Enhancing swish activation with tanh bias for improved neural network performance. arXiv preprint arXiv:2407.01012.

Seol, J., Kim, J., Hong, Y., Cha, M., Park, S., Jang, K.-J., Kim, S.-J., Son, H.I., 2025. Toward intelligent 3D food printing: a review on the perspective of materials, fabrication, monitoring, and control. Crit. Rev. Food Sci. Nutr. 1–32.

Shin, H.-C., Lu, L., Summers, R.M., 2017. Natural language processing for large-scale medical image analysis using deep learning. Deep. Learn. Med. Image Anal. 405–421.

Sun, J., Peng, Z., Zhou, W., Fuh, J.Y., Hong, G.S., Chiu, A., 2015. A review on 3D printing for customized food fabrication. Procedia Manuf. 1, 308–319.

Sun, J., Zhou, W., Yan, L., Huang, D., Lin, L.-y., 2018. Extrusion-based food printing for digitalized food design and nutrition control. J. Food Eng. 220, 1–11.

Van der Velden, B.H., Kuijf, H.J., Gilhuijs, K.G., Viergever, M.A., 2022. Explainable artificial intelligence (XAI) in deep learning-based medical image analysis. Med. Image Anal. 79, 102470.

Wan, H., Liu, D., Yu, X., Sun, H., Li, Y., 2015. A Caco-2 cell-based quantitative antioxidant activity assay for antioxidants. Food Chem. 175, 601–608.

Windhab, E.J., 2006. What makes for smooth, creamy chocolate? Phys. Today 59 (6), 82–83.

Yin, L., Tao, W., Zhao, D., Ito, T., Osa, K., Kato, M., Chen, T.-W., 2024. UNet–: Memory-efficient and feature-enhanced network architecture based on U-Net with reduced skip-connections. In: Proceedings of the Asian Conference on Computer Vision. pp. 4085–4099.

Zhang, C., Lu, W., Wu, J., Ni, C., Wang, H., 2024. SegNet network architecture for deep learning image segmentation and its integrated applications and prospects. Acad. J. Sci. Technol. 9 (2), 224–229.

Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J., 2017. Pyramid scene parsing network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2881–2890.