ACCSCIENCE PUBLISHING

International Journal of AI for Materials and Design

ORIGINAL RESEARCH ARTICLE

Computer vision and deep learning-based prediction for inkjet-printed electrodes

Gareth Quinn[1*], Achu Titus[1,2,3*], Anesu Nyabadza[1,2,3], Éanna McCarthy[1,2,3], Sithara Sreenilayam[1,2,3], and Dermot Brabazon[1,2,3]

[1]School of Mechanical and Manufacturing Engineering, Dublin City University, Dublin, Ireland

[2]I-Form Advanced Manufacturing Centre Research, Dublin City University, Dublin, Ireland

[3]DCU Institute for Advanced Processing Technology, Dublin City University, Dublin, Ireland

(This article belongs to the *Special Issue: Applications of Deep Learning in Advanced Materials Processing*)

***Corresponding authors:**
Gareth Quinn
(gareth.quinn39@mail.dcu.ie)
Achu Titus
(achu.titus2@mail.dcu.ie)

Citation: Quinn G, Titus A, Nyabadza A, McCarthy E, Sreenilayam S, Brabazon D. Computer vision and deep learning-based prediction for inkjet-printed electrodes. *Int J AI Mater Design*. 2025;2(4):24-36. doi: 10.36922/IJAMD025430040

Received: October 22, 2025

Revised: November 28, 2025

Accepted: December 8, 2025

Abstract

With the development of inkjet-printed electrodes, artificial intelligence-based quality control is essential for classifying inkjet-printed electrodes in a quality control environment. The quality of printed structures can be significantly affected by defects such as cracks, smudging, and misaligned deposits, which can degrade electrical performance and overall device reliability. Traditional quality control methods, including manual inspection and electrical testing, are time-consuming, subjective, and invasive, and they are unsuitable for high-throughput manufacturing environments. This work explores the application of computer vision and deep learning, specifically Convolutional Neural Networks (CNNs) and Feedforward Neural Networks, to automate defect detection and quality classification of inkjet-printed electrodes. To demonstrate the accessibility of deep learning techniques, Neural Architecture Search was implemented, showing the importance of automated model design in achieving high performance without extensive manual tuning or the need for expertise. The CNN models proved to be the most suitable approach for this image classification task, achieving a testing accuracy of 90.9% and a precision of 88.9% for a dataset of 2,406 electrode images containing both high-quality (1,020) and low-quality (1,386) prints.

Keywords: Inkjet printing; Electrodes; Defect detection; Deep learning; Computer vision; Convolutional Neural Networks; Feedforward neural networks; Neural architecture search

1. Introduction

Inkjet printing, also known as non-contact printing, has become a vital fabrication method in the production of flexible electronics, offering key advantages such as high customizability, minimal material waste, low temperature processing, fast deposition process, additive and digital patterning, and compatibility with a broad range of substrates[1] (*e.g.*, polymer, paper, textile, and fabric). It has wide applications in areas such as wearable and personalized healthcare devices,[2-4] electrochemical and biosensor devices,[5,6] flexible electronics (displays and solar cells), energy devices,[7,8] bioengineering,[9] micro-electro-mechanical systems,[10] and microfluidic devices.[11] In an inkjet printer, inks stored in a reservoir are delivered through micrometer-sized nozzles in the print head, which operate either continuously or in a drop-on-demand mode to form electronically designed patterns or images. This process involves droplet generation, additive deposition of micro- and nanometer-scale materials, and ink spreading over the substrate, followed by drying without any physical impact on the substrate. However, this printing technique poses a significant issue known as the coffee ring effect (*i.e.*, formation of a ring-like thicker deposit at the print pattern edge due to the presence of solid particles in a drying ink droplet),[12] which hinders the formation of uniform patterns and negatively impacts the morphological and electrical properties of the electrodes.[13,14] Furthermore, print patterns experience geometric imperfections, such as cracking and discontinuities, due to the properties of the inkjet printable ink material, droplet ejection behavior, substrate surface morphology, and inkjet printing parameters, resulting in defective electrodes.[15] A prolonged sintering process also affects the microstructure of the printed electrode, especially porosity and particle order.[14] Defects are critical in inkjet-printed electrodes because they directly lower conductivity, degrade device performance, and compromise reliability.[16] Studies show that up to 66% of inkjet-printed thin-film transistors are defective due to process-related issues.[15] Issues, including particle agglomeration in the nanomaterials-based inkjet printable inks, air entrapment, nozzle obstruction, droplet misalignment, and environmental contamination, can significantly impair the precision and uniformity of prints. These factors result in unpredictable electrical properties and device unreliability.[17-19] Before the advent of digital microscopy, these defects were managed through quality control techniques, such as manual inspection and electrical testing, which were labor-intensive, subjective, intrusive, and inappropriate for large-scale manufacturing environments. At present, there are numerous methods to check the quality of a completed print, including inspection techniques, data analysis strategies, and quality assurance practices.[20] The quality assurance practices include optical photography, digital imaging, profilometry, and electrical resistance/conductivity measurements. The initial step for conductive inks involves assessing the success of the sintering process by examining the sample for any surface deformation and the presence of cracks. The inkjet-printed sample typically comprises particles at the nano- (10^{-9}) or micro- (10^{-6}) scale, which should be at least 50 times smaller than the nozzle diameter, which helps to avoid issues such as particle accumulation at the nozzle edge, droplet trajectory deviations, or nozzle blockage due to agglomeration.[21] In the absence of cracks, the sample undergoes a series of performance evaluations, which encompass electrical resistivity, adhesion, and mechanical deformation assessments.[22]

Anticipating defective electrodes before printing and modifying printing parameters is a crucial strategy for minimizing the incidence of defective printed electrodes, thereby enhancing the large-scale production of high-quality electrodes. The early identification of defects can be accomplished through a computer vision system, which represents an automated, non-destructive, and high-speed method utilizing advanced imaging and machine learning techniques.[20] Incorporating computer vision into the quality control of inkjet-printed electrodes allows manufacturers to identify and rectify problems, such as absent deposits, misalignment, and variations in line width, before compromising product performance.[23,24] It will enable immediate identification and correction of printing surface defects, minimizing material waste and improving yield rates. This technique offers several significant advantages, including the elimination of manual inspection, reduced human error, and increased efficiency.

Computer vision employs sophisticated imaging methods and machine learning algorithms to detect microscopic defects that may elude human observation. Several significant works have explored the intersection of inkjet printing and machine learning, focusing on improving monitoring, classification, prediction, and optimization of the printing processes.[23] Researchers have applied artificial intelligence (AI) to different aspects of the process, such as ink conductivity, line width, resistance, droplet velocity, jetting behavior, and overall print quality.[25-27] A wide range of methods have been explored, from basic regression models and ensemble techniques like Random Forest and Gradient Boosting to more advanced neural networks, as explained in Table 1. Deep learning models, including Convolutional Neural Networks (CNNs), Feedforward Neural Networks (FNNs), and Neural Architecture Searches (NAS), can analyze complex printed patterns and distinguish between acceptable and defective prints with high precision. In a recent work, a CNN-based algorithm was considered to identify four parameters of inkjet-printed functional structure, such as droplet spacing, line resistance, line quality, and the post-treatment method developed.[27] The model successfully classified printing features from the pictures, proving the use of CNNs for image classification and inkjet-printed structure classification.[28] A residual neural network was employed in another study to classify the three distinct stages of ejected droplets: none, non-spherical, and spherical.[27] The model attained an accuracy of approximately 96% after five epochs, after reducing the input size using a translation-invariant linear transformation, while the uncompressed model reached an accuracy of about 92%.[27] NAS was used

Table 1. Overview of the recent research in computer vision for the evaluation of inkjet-printed electronic structures

No	Predicted quality/defect of inkjet-printed structure	AI method used for prediction	References
1	Ink conductivity of the printed components	Regression models, including DT, XGBoost, LightGBM, and ResNet	30
2	Line width and resistance per unit length of printed traces	NLR, KNN regression, GPR	31
3	Drop pitch, line pitch, and average drop volume	RF, AdaBoost, and GB regressors	32,33
4	Predict drop velocity and formation	SVM, KNN, RFs, XGBoost, and MLP	33
5	Drop velocity and jetting morphology.	SVM, B-Tree, and MLP	34
6	Jetting status and predicting a high-resolution pattern	KNN, CART, RF, LR, gradient boost classifier, and bagging model	35
7	Jetting behavior of novel inks	GB and RF	36
8	Drop velocity and jetting morphology	MLP algorithm	37
9	Space into three regions based on predicted jetting probability: certain jetting, certain non-jetting, and doubt region	SVM, Multilayer Neural Network, and Gaussian Naïve Bayes	38
10	Printing quality optimization and electrical resistivity prediction	CNN architectures	39
11	Categorize the behavior of the droplet (normal, no-droplet, and satellite modes)	BPNN	25
12	Electrical conductivity of organic graphene-based electrodes	DTs, RFs, and KNN	40

Abbreviations: AdaBoost: Adaptive Boosting; BPNN: Backpropagation Neural Network; CART: Classification and Regression Tree; CNN: Convolutional Neural Network; DT: Decision Tree; GB: Gradient Boosting; GPR: Gaussian Process Regression; KNN: K-Nearest Neighbors; LightGBM: Light Gradient Boosting Machine; LR: Logistic Regression; MLP: Multilayer Perceptron; NLR: Non-linear regression; ResNet: Residual Neural Network; RF: Random Forest; SVM: Support Vector Machine; XGBoost: Extreme Gradient Boosting.

to detect and classify defects, such as the coffee ring effect and geometric defects (cracks, discontinuities, and surface impurities) in flexible inkjet-printed sensors. They used the computer vision model YOLOv5 as a baseline; along with NAS, the model achieved a mean average precision of 81.2%, 95.5% accuracy, and a detection time of 4.6 ms, making it suitable for real-time defect monitoring.[5] These two prediction models improved the print success rate to over 80%.[5] A separate study employed machine learning models, such as random forest, multilayer perceptron, and support vector machine, to forecast the printability and drug dosage of inkjet printing before formulation preparation, and it attained an accuracy of 97.22% in predicting the printability of formulations and 97.14% in assessing the quality of the prints.[29]

This work explores the application of deep learning, specifically CNNs and FNNs, to automate defect detection and quality classification of inkjet-printed electrodes. A custom dataset comprising over 400 images of inkjet-printed electrodes was developed, with each image manually classified as either exhibiting good- or poor-quality print based on visual inspection. These two classes were then examined through exploratory data analysis to identify distinguishing features and patterns. The images were subsequently pre-processed using a range of image processing techniques and used to train multiple neural network models. Model performance was assessed using several evaluation metrics, including accuracy, F1-score, and confusion matrices. To examine the accessibility of deep learning techniques, NAS was implemented, highlighting the importance of automated model design in achieving high performance without the need for extensive manual tuning or specialized expertise. This work demonstrates the effectiveness and accessibility of AI-powered visual inspection systems in improving production yield, reducing material waste, and enabling scalable, automated manufacturing of inkjet-printed electrodes.

2. Methods

2.1. Experimental setup

The manganese ink was produced using the pulsed laser ablation technique, and the electrodes were printed using a cartridge with 21-μm nozzle diameter, utilizing a FUJIFILM Dimatix Materials Printer DMP 2850 (FUJIFILM Dimatix, Inc., USA).[41] A 3 × 3 complete factorial Design of Experiments was conducted with the printing parameters to differentiate between good- and bad-quality prints. This study explored three different inkjet printing parameters, namely the jetting frequency (40, 60, 80 kHz), the print bed temperature (28, 30, 35°C), and the number of printed layers (10, 15, 30).[41] Each electrode printing parameter set permutation was printed 16 times with dimensions of 10 × 10 mm on a paper substrate, producing 432 electrodes. All electrode data, including printing and experimental parameters, are provided in Table S1 (in Supplementary File). Inkjet printing mechanism and representative printed electrodes are shown in Figure 1. Some samples did not print correctly for specific printer parameters; therefore, only 401 electrodes were suitable for imaging.

Following the printing process, high-resolution images of individual electrodes and full electrode grids were captured using a Google Pixel 6 smartphone with a 50-megapixel camera and 7× zoom, positioned approximately 100 mm from the substrate. All images were taken parallel to the substrate under fluorescent diffuse white light illumination. This setup mirrors the typical capability of industrial line camera imaging systems. The inkjet-printed electrodes exhibited defects such as smudging, irregular deposition, contamination, and cracks. These conditions resulted in datasets labeled as either "Good Print Quality" or "Bad Print Quality," using binary classification based on visual inspection. Extensive exploratory data analysis techniques were used, such as pixel intensity distribution analysis, t-SNE visualization, Sobel edge detection, and various thresholding methods, in an attempt to highlight defects effectively. Figure 2 illustrates examples of electrodes classified as having "Good Print Quality" and "Bad Print Quality." Defective electrodes typically exhibited significant misprints or structural defects that impaired their ability to conduct electricity.

2.2. Exploring the datasets

Before proceeding with model training, an extensive exploratory data analysis was conducted to investigate the features of the good- and bad-quality printed electrodes. This involved statistical evaluations, intensity histograms, edge detection, thresholding techniques, and feature visualization methods such as t-SNE and Histogram of Oriented Gradients (HOG). The results of this section are described in Figure S1 (in Supplementary File). The pixel intensity distribution analysis showed that the distribution of good prints is more uniform. In contrast, bad prints exhibit greater randomness and a wider range of intensity values. Edge detection and thresholding techniques showed potential in highlighting defects but struggled with variations in the dataset, even when enhanced with automatic methods. Similarly, the HOG effectively captured large-scale defects by detecting sharp intensity changes but had difficulty identifying more minor, subtler anomalies.

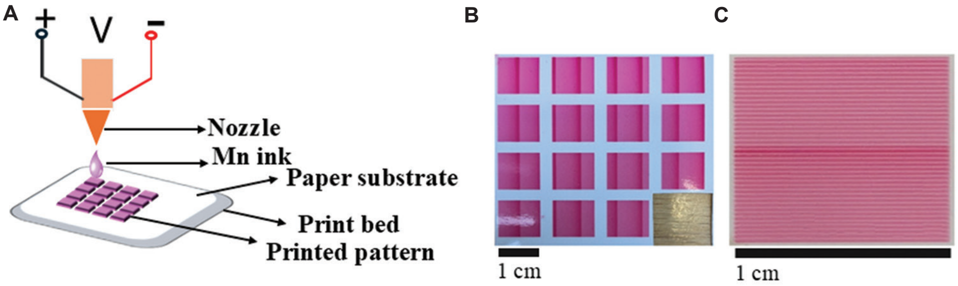

Figure 1. Inkjet printing process and printed electrodes. (A) Schematic view of the inkjet printing of manganese ink; (B) Printed grid (50 mm × 50 mm) with one sample cut out for various characterization; (C) An individual electrode (10 mm × 10 mm). The electrode grid (B) corresponds to sample 9, fabricated with 30 layers at 40 Hz and 30°C, while the electrode (C) was produced with 15 layers under the same frequency and temperature conditions.

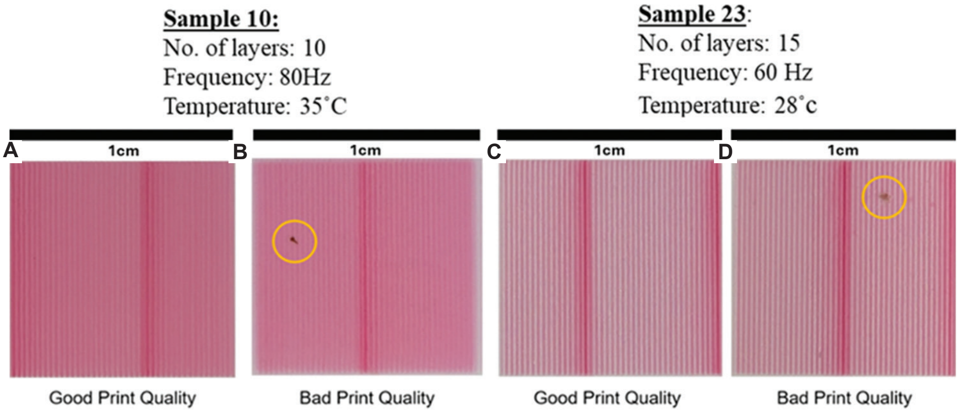

Figure 2. Pictures of electrodes exhibiting good (A and C) and bad (B and D) print quality from sample batches 10 and 23. Sample 10 was printed with 10 layers at 80 Hz and 35°C, while sample 23 was printed with 15 layers at 60 Hz and 28°C. Defects such as localized blotches are highlighted in yellow circles in the bad-quality prints.

2.3. Model preparation, training pipeline, and performance

To ensure optimal model performance and maintain consistency for fair comparison among models, a structured data preprocessing strategy and general pipeline were created. This included data augmentation to enlarge the dataset; a consistent train-test split; class balancing techniques to mitigate bias; callback functions for efficiency and effectiveness; normalization and standardization of the data; and, for evaluation, the performance metric values of accuracy, precision, recall, and F1-score were evaluated for each model. Along with these performance metric values, a confusion matrix was plotted to visualize the predictions made by the model. Finally, training graphs were plotted: loss over epochs and accuracy over epochs. The libraries used to evaluate the model were Tensorflow, Matplotlib, Scikit-Learn, and Pandas. The original datasets contained 401 images, representing a relatively small dataset size for neural networks, especially CNNs. To combat this, the images within the original dataset were augmented with rotation, brightness adjustments, zooming, and flipping to artificially enlarge the dataset while ensuring realistic variations. Each image underwent five augmentations, resulting in a new dataset size of 2,406, with 1,020 images of good print quality and 1,386 of bad print quality.

For most model iterations, except for a few during the optimization stages, all images were resized to a standard 128 × 128 resolution. The OpenCV's resize function was used for this transformation, which provided computational efficiency while preserving essential image details. Then, the dataset was divided into 70% training and 30% testing (15% testing and 15% validation). This train-test split ensured that the model had enough data to train on and that the training data represented the range of defects that could potentially occur. In addition, the testing and validation sets were large enough to provide reliable performance metrics, ensuring accurate and unbiased model assessment. The enlargement of the dataset through data augmentation resulted in a significant class imbalance. The bad print quality images outweighed the good print quality images by over 26%. This class imbalance could cause bias within the models, so class balancing was adopted to mitigate this. For each class, the class weights were calculated using the Scikit-Learn class weight function, which was applied during training by modifying the loss function to apply a higher weight to the minority class and lower weights for the majority class.

Learning rate schedulers were implemented to reduce the set learning rate when the validation loss stopped improving after five epochs, helping the model develop after reaching a plateau. For every model, the same metrics were measured. The metrics that were evaluated for each model were accuracy, precision, recall, and the F1-score. The performance of each neural network model was evaluated using four standard classification metrics: accuracy, precision, recall, and F1-score, each derived from the confusion matrix. In binary classification, predictions can be true positives (TP), true negatives (TN), false positives (FP), or false negatives (FN). Accuracy represents the proportion of correct predictions across all samples, computed using this formula: (TP+TN)/(TP+TN+FP+FN). Precision measures the proportion of correctly identified positive cases out of all positive predictions (TP/(TP+FP)), which is crucial in this project to minimize the number of defective electrodes incorrectly classified as good. Recall (TP/(TP+FN)) reflects the model's ability to correctly detect all good electrodes, while the F1-score provides a harmonic mean between precision and recall, balancing both FP and FN. These metrics were chosen because they provide valuable insights into the model's performance in real-world scenarios. Along with these performance metrics' values, a confusion matrix was plotted to visualize the predictions made by the CNN and FNN models, as shown in Figures S2 and S3, respectively. Finally, training graphs were plotted: loss over epochs and accuracy over epochs (Figures S2 and S3). Figure 3 shows examples of the two training graphs along with the confusion matrix

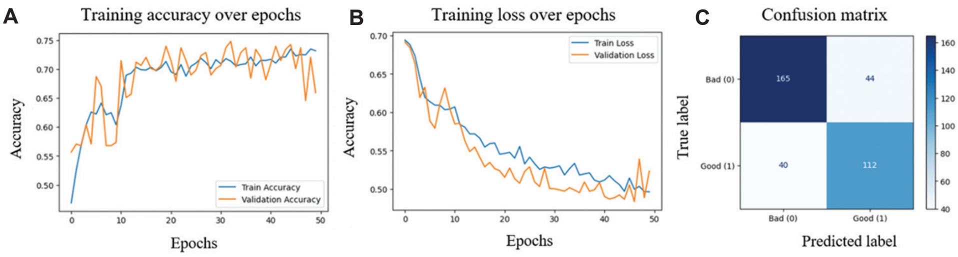

Figure 3. Performance evaluation of the basic CNN model. (A) Training and validation accuracy trends across epochs. (B) Training and validation loss trends across epochs, indicating model convergence. (C) Confusion matrix demonstrating the CNN's classification accuracy for good- and bad-print-quality electrodes.
Abbreviation: CNN: Convolutional Neural Network.

of the basic structured CNN training on the augmented dataset.

2.4. Model developments and optimization

This work modeled and compared CNNs, FNNs, and NAS to detect print quality. All model development and optimization were carried out using Python 3.10.0, primarily within the PyCharm development environment on a local laptop equipped with an Intel Core i7-1065G7 CPU (4 cores, 8 threads, 1.3–3.9 GHz), 8 GB RAM, and a 512 GB SSD. Additional training was performed on an NVIDIA Tesla T4 GPU through Google Colab for faster training and development times. The following section will explain how the previously mentioned neural network models are used to detect defective electrodes.

2.4.1. CNN model optimization

For the development of a fully optimized CNN model, basic and deep models were developed and fine-tuned using hyperparameter tuning and multiple optimization techniques to achieve the highest accuracy. After the initial iterations and improvements on the basic CNN architecture, the decision was made to restructure the network to enhance its ability by incorporating additional layers and nodes along with enhanced regularization techniques to detect more profound relationships within the image data and also to mitigate overfitting. Based on the hyperparameter tuning of the deep CNN model, a fully optimized model was developed to maximize performance and to ensure generalization to the data. The CNN architecture, from its basic structure to the concluding configuration, was modeled and optimized using the augmented datasets, as detailed in Tables S2 and S3 (in Supplementary File). The final architecture was a deep CNN with five convolutional layers, each using the Rectified Linear Unit (ReLU) activation function and L2 regularization ($\lambda = 0.001$) to ensure stable learning and help the model learn complex patterns. Batch normalization was applied after each convolutional layer. Max pooling was implemented after each layer for computational efficiency; however, it was left out in the final convolutional layer to preserve larger feature maps, ensuring the model had sufficient information for effective learning. The extracted features were flattened using the Flatten () function, due to the relatively small feature map size, and passed into a fully connected layer comprising three dense layers (512, 128, and 64 neurons). Each dense layer had a 30% dropout rate, which improved generalization by preventing overfitting and enabling the model to capture complex relationships. The output layer used the sigmoid activation function, producing a probability score between 0 and 1 to classify electrodes as defective or non-defective. The model was trained on RGB images, using the Adam optimizer and a binary cross-entropy loss function for robust optimization. The model had an initial learning rate of 0.0001. All other pipeline components, including callback functions, data splitting, class balancing, image resizing, and *ImageDataGenerator* setup, remained the same. The model was trained for 100 epochs, ensuring full convergence and optimal learning. In addition, early stopping techniques were employed to reduce unnecessary training and prevent overfitting. Figure 4 shows the optimized CNN model architecture.

2.4.2. FNN

The original FNN model featured a basic architecture consisting of three hidden layers and one output layer. The initial hidden layer comprised 12 neurons, succeeded by 64 neurons in the second layer and 16 neurons in the third. For binary classification, the output layer used the sigmoid activation function. To mitigate overfitting, a 20% dropout was introduced after the first hidden layer. The non-linear activation function, ReLU, was used across all the hidden layers to enhance feature learning and improve

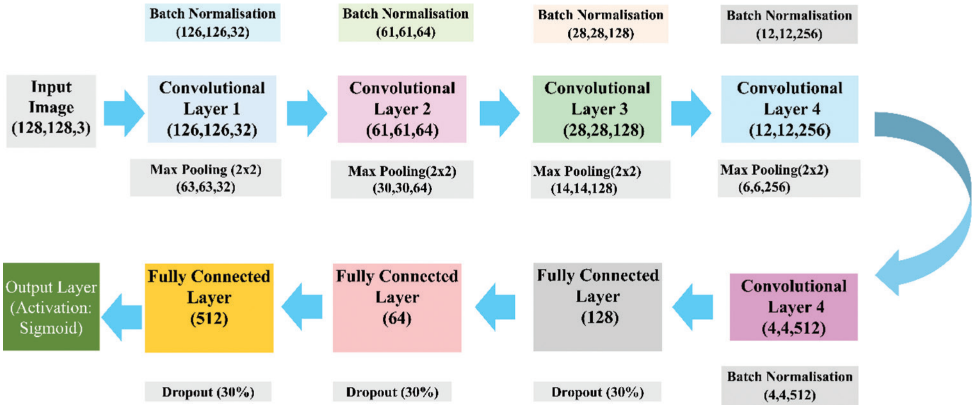

Figure 4. Flow chart illustrating the architecture of the optimized Convolutional Neural Network

generalization. The basic FNN architecture is shown in Table S4 (in Supplementary File). The dataset was further expanded and diversified using the *ImageDataGenerator* function for additional data augmentation. The model was optimized using the Adam optimizer, with binary cross-entropy as the loss function. The initial learning rate was set at 0.0001 during the training process. Following the first iteration's results, the decision was made to eliminate the *ImageDataGenerator* function from the pipeline. Dropouts and batch normalization were implemented following the initial two hidden layers to mitigate overfitting and enhance generalization within the model. In addition, L2-regularization, with λ = 0.001, was implemented on all three fully connected layers. The initial FNN with regularization is detailed in Table S5. Additional layers were incorporated to enhance the model's ability to learn complex patterns, thereby improving generalization and overall performance. Given that FNNs are not intrinsically suited for feature extraction, manual feature extraction was incorporated into the pipeline to assist the model in recognizing features within the images. The manual feature extraction methods implemented were Sauvola's thresholding technique and Sobel edge detection. Sauvola's thresholding was implemented using the libraries OpenCV and Scikit-image. After extensive hyperparameter tuning and iterative design, the final FNN model was composed of eight hidden layers arranged in a decreasing, funnel-shaped structure (256, 128, 64, 64, 32, 16, 1). Each layer employed the ReLU activation function, enabling the model to learn complex patterns and relationships. L2 regularization with a weight of 0.001 was implemented to prevent certain features from overshadowing the output and thereby enhancing generalization. Batch normalization and dropout were implemented after specific layers, as illustrated in the model architecture in Figure 5 and Table S6.

The output layer utilized the sigmoid activation function. The model was optimized using the Adam optimizer with a binary loss function and an initial learning rate of 0.0001 to support steady and stable learning. All other components of the pipeline, including callback functions, data splitting, class balancing, and image resizing, remained unchanged.

2.4.3. NAS

The utilization of NAS facilitated the efficient development of tailored, optimized neural networks for defect detection in inkjet-printed electrodes, demonstrating the accessibility of AI solutions for this issue. This was implemented using the AutoKeras library to automatically design an optimal neural network architecture tailored to the binary classification task. Pre-processed grayscale electrode images were standardized and split into training, validation, and testing sets (70/15/15). Class imbalance was addressed using the computed class weights. Due to computational limitations, only three trials were configured to run. The model was trained on Google Colab using the free T4 GPU, and the best-performing architecture was exported and evaluated.

3. Results and discussion

3.1. CNN results

The development, evaluation, and optimization of the CNN-based image classification model have demonstrated to effectively identify print defects in inkjet-printed electrodes for quality control purposes. The initial CNN model provided a foundational understanding of the network's capability to differentiate between good and bad-printed electrodes. While the base model achieved ~67% accuracy, it lacked robustness and stability, with a low precision score (~61%) indicating a high FP rate (bad-quality electrodes being classified as good), which is unsuitable for real-world quality control systems. However, the slight difference between training and testing performance suggested that the model was not overfitting and had potential for improvement.

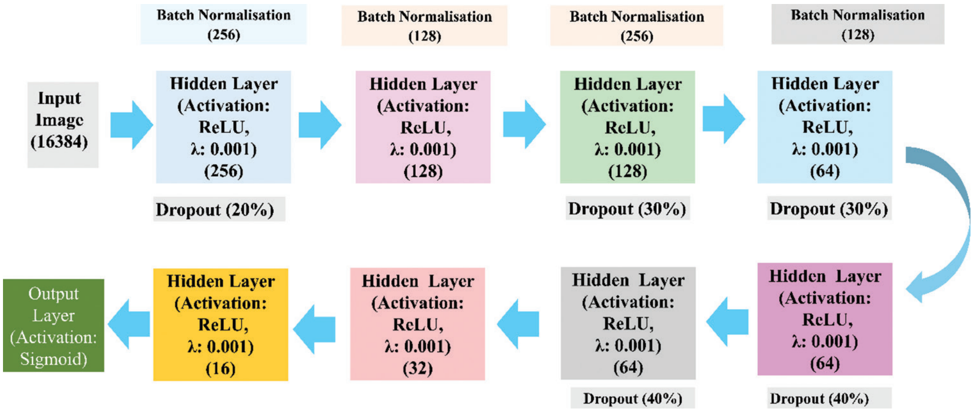

Figure 5. Flowchart illustrating the architecture of the optimized Feedforward Neural Network

After the implementation of data augmentation, there was a significant improvement in the model's performance. The validation accuracy increased from ~67% to ~77% and precision increased from ~61% to ~72%, indicating that there was roughly a 10% decrease in the number of FP (bad electrodes being misclassified). The average difference between training and validation also decreased, from ~5% to ~3%, highlighting the fact that the model generalized better. Standardizing the input data and adding additional augmentation using the *ImageDataGenerator* function, and transitioning to a deeper CNN architecture, significantly boosted performance and stability. Table 2 summarizes the model's accuracy before and after data augmentation. All metrics are derived from the performance on the testing set during training, which consists of the 20% of data withheld during training and not previously seen by the model.

The deeper model achieved an accuracy of up to approximately 75%, and the percentage differences between training and testing results showed improved generalization. While this iteration showed stronger performance, the misclassification of 27% of defective electrodes still posed a risk in a practical quality control implementation. In addition to that, the extensive hyperparameter tuning, which incorporated L2 regularization and dropout, proved effective in reducing overfitting by limiting dominant features and enforcing more general feature learning. *GlobalAveragePooling2D* was found to be a superior alternative to the *Flatten* function for smaller networks, acting as a natural regularizer and improving generalization. Adjustments to batch size and optimizer choice revealed Root Mean Square Propagation (RMSprop) as the most effective, and training on RGB images further increased classification accuracy by providing richer feature information. The summary of performance metrics from the hyperparameter tuning of the deeper CNN model is presented in Table 3.

Table 2. Comparison of the performance of a basic CNN model trained on the original dataset (401 images) versus an augmented dataset (2,406 images)

Dataset	Accuracy	Precision	Recall	F1-Score
Pre-augmentation	0.672131	0.615385	0.827586	0.705882
Post-augmentation	0.767313	0.717949	0.736842	0.727273
% Difference	+14	+17	−11	+3.03

Abbreviation: CNN: Convolutional Neural Network.

The optimized CNN model achieved an impressive ~90% accuracy on the test data, along with an ~88% precision score, resulting in the lowest FP rate among all developed models. In binary classification, predictions can be TP, TN, FP, or FN. Accuracy represents the proportion of correct predictions across all samples, given by (TP+TN)/(TP+TN+FP+FN). Precision measures the proportion of correctly identified positive cases out of all positive predictions (TP/(TP+FP)), which is crucial in this project to minimize the number of defective electrodes incorrectly classified as good. Recall (TP/(TP+FN)) reflects the model's ability to correctly detect all good electrodes, while the F1-score provides a harmonic mean between precision and recall, balancing both FP and FN. Notably, the training and testing metrics were closely aligned, demonstrating strong generalizations on completely unseen data. The training graphs in Figure S2 show that validation accuracy tracked closely with training accuracy until around the 35th epoch, where a slight divergence emerged, indicating minimal overfitting; however, the model continued to learn effectively. The early stopping function stopped training at the 86th epoch, once validation loss had plateaued, ensuring optimal performance without unnecessary computing. Examining the confusion matrix, the model delivered the best performance across all iterations, misclassifying just 18 out of 203 electrodes. This translates to a 92% accuracy

Table 3. Summary of performance metrics from the hyperparameter tuning of deeper CNN

CNN model development stages	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Average % difference to training	% increase from base CNN
Base deep CNN (Batch size: 32; optimizer: Adam)	74.38	68.72	76.88	72.57	2.18	#NA
Adding layers	82.55	76.33	84.87	80.37	5.30	9.74
Batch size 16	80.61	71.81	88.82	79.41	2.97	8.77
Batch size 64	80.33	72.13	86.84	78.81	2.91	8.04
L2 regularization	82.83	75.00	88.82	81.33	3.11	10.80
RGB input	82.37	77.27	85.00	80.95	5.68	10.15
RMSprop optimizer	76.86	73.17	75.00	74.07	3.04	2.20
M.SGD optimizer	72.45	67.65	71.88	69.70	0.88	−3.71
Global average pooling	77.14	74.21	73.75	73.98	2.97	2.19

Abbreviation: CNN: Convolutional Neural Network; M.SGD: Stochastic Gradient Descent with Momentum; RMSprop: Root Mean Square Propagation.

in identifying 121 defective electrodes, demonstrating the model's robustness and reliability for real-world defect detection applications. Overall, the results from this section demonstrate that CNNs, when adequately trained, tuned, and optimized, are highly effective for image-based defect classification, even with very minor defects. Figure 6 presents the results from the optimized CNN in training and testing. Training graphs along with the confusion matrix are shown in Figure S2 (in Supplementary File).

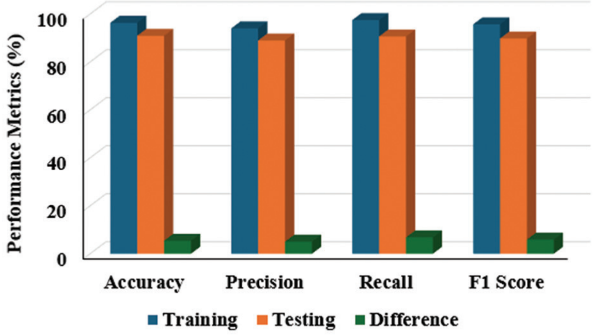

Figure 6. Graph showing the normalized performance metric of accuracy, precision, recall, and F1-score for the Convolutional Neural Network model

3.2. FNN results

Just like CNN, the FNN model's development also followed a structured, iterative approach to detect the defective electrode. The initial baseline model struggled to distinguish between the data due to its simple architecture, which struggled to handle the complexity of the data. Flattening the raw image data into one-dimensional (1D) arrays caused the loss of significant spatial information, making it very difficult for the FNN to distinguish between defective and non-defective electrode images. By removing the *ImageDataGenerator* function, the model's performance and ability to learn patterns in the data significantly increased. The model achieved a test accuracy of ~75% and a precision of ~71%, which are decent metrics. From the testing data, over 88/416 electrodes were classified as bad, giving the model an FP rate of 21%. Within the deeper FNN, employing Sauvola's thresholding yielded a testing accuracy of approximately 72%, while the Sobel edge detection method completely failed to distinguish between the classes. These findings underscore the FNN's sensitivity to noisy or overly complex inputs and its limited capacity to interpret spatial information effectively. The final model, which incorporated an optimized pipeline, regularization, and a deep architecture, achieved ~71% test accuracy and ~67% precision. The model misclassified 95 out of 416 bad electrodes, resulting in a 22% FP rate. Figure 7 presents the results of the optimized FNN. All training graphs, along with the confusion matrix, are illustrated in Figure S3 (in Supplementary File). In addition, receiver operating characteristic curves were plotted for both the CNN and FNN, which are shown in Figures S4 and S5 (in Supplementary File). The area under the curve was found to be 0.95 for optimized CNN and 0.76 for FNN, indicating excellent and acceptable discrimination performance, respectively.

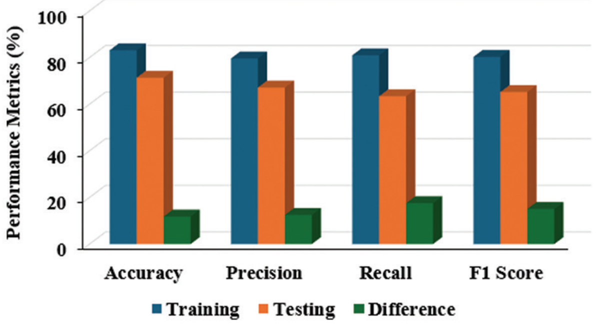

Figure 7. Graph showing the normalized performance metric of accuracy, precision, recall, and F1 score for the Feedforward Neural Network model

3.3. NAS results

From the model summary function, it was understood that the best-performing model obtained through NAS used a pre-trained EfficientNetB7 (a popular CNN architecture) as the backbone, with input shape being scaled grayscale images (224,224,3). The inputs were normalized and augmented (flip, translation). Global average pooling was used to translate the feature maps into a 1D array and the sigmoid activation function was used to classify. This NAS model demonstrated strong generalization, with a testing accuracy of ~91.4% and a training accuracy of ~94.8%. The ~3.67% accuracy difference between training and testing accuracy suggested that the model did not significantly overfit, indicating that it learned generalizable features rather than memorizing the training data. The results from using this automated method are presented in Figure 8 and Figure S6.

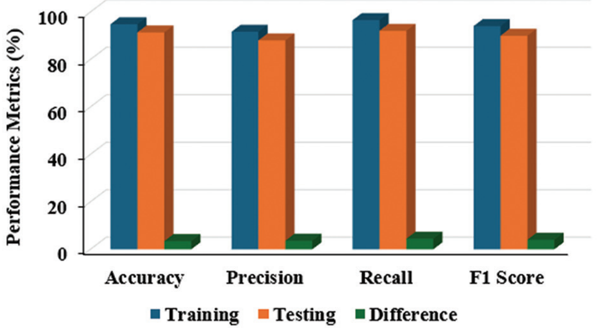

Figure 8. Graph showing the normalized performance metric of accuracy, precision, recall, and F1-score for the Neural Architecture Search-generated model

With a precision score of ~88% on testing, out of 209 printed electrodes, 19 were misclassified, but the model effectively classifies the majority of the remaining bad electrodes. With a minimal FP rate and effective classification, this NAS model effectively meets the strict quality control standards required in the manufacturing industry. These results demonstrate the use of automated deep learning methods to develop and optimize networks for classification tasks, without the need for expertise in machine learning or deep learning.

Overall, CNNs emerged as the most effective approach: the final optimized CNN achieved 90.9% test accuracy and 88.9% precision, aided by data augmentation, batch normalization, L2 regularization, and using RGB inputs. FNNs, by contrast, struggled due to the loss of spatial information when flattening images, with the best model reaching ~71% accuracy. Manual feature extraction (Sauvola thresholding, Sobel detection) offered limited benefit and often introduced noise. NAS with EfficientNetB7 achieved comparable results to the custom-built CNN (91.4% accuracy, 88% precision) while requiring minimal manual tuning. Overall, CNNs demonstrated strong generalization and industrial suitability, while FNNs proved less effective for image-based classification. Figure 9 shows a brief comparison of the three methods used to classify the test images. Figure 10 shows examples of the optimized CNN inference results.

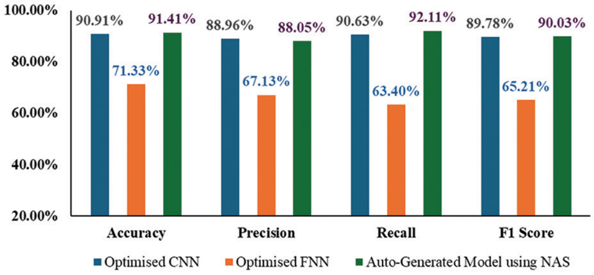

Figure 9. Graph showing accuracy, precision, recall, and F1-score as percentages, representing the proportion of correct predictions across all samples, for the CNN, FNN, and NAS models

Abbreviations: CNN: Convolutional Neural Network; FNN: Feedforward Neural Network; NAS: Neural Architecture Search.

Past literature has presented deep learning work to classify the jetting characteristics and conductivity measurements of inkjet-printed electrodes for assessing the quality of prints.[32,38] In this paper, the quality of inkjet-printed electrodes is classified on the basis of surface defects such as contamination and cracks, caused by irregularities from the deposition process. In particular, for the first time in this manuscript, the classification performance has been assessed and compared of deep learning models, including CNNs, FNNs, and NAS-generated models. CNN models were identified as one of the best classification models, resulting in a testing accuracy and precision of 90.9% and 88.9%, respectively, with improvements such as data augmentation, batch normalization, L2 regularization, global average pooling, and the use of RGB images. This prediction accuracy measured in this work for the CNN was in line with that found from previous work, which examined CNN architectures for extracting representative features of printed lines.[39]

4. Conclusion

This work demonstrates how deep learning, particularly CNNs, can be effectively leveraged to automate visual inspection tasks, offering a scalable, accurate, and efficient solution for industrial quality control. In particular, this work shows how neural networks can be leveraged for the classification of inkjet-printed electrodes in a quality control environment. A comparison between CNNs, FNNs, and NAS models highlights the strengths and limitations of each approach for the classification of inkjet-printed electrodes. The CNN models were the most suitable approach for this image classification task. After iterative improvements, the final optimized CNN model achieved 90.9% testing accuracy and 88.9% precision, demonstrating strong generalization with a low FP rate. Improvements such as data augmentation, batch normalization, L2 regularization, Global Average Pooling, and utilization of RGB images allowed the CNN to capture more complex patterns and stabilize training. These improvements also helped prevent overfitting, making the model reliable and suitable for industrial implementation. RGB input images

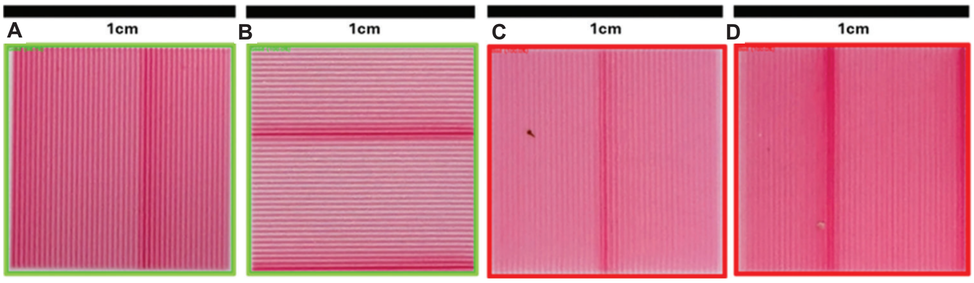

Figure 10. Pictures of electrodes correctly classified by the optimized Convolutional Neural Network. (A and B) Good-print-quality electrodes; (C and D) Bad-print-quality electrodes, which display noticeable blotches and contaminants.

contributed to a ~6% improvement in accuracy over grayscale images but at the cost of increased computational demand and longer training times. The base FNN using *ImageDataGenerator* showed random prediction behavior, with testing accuracy at ~58%. Removing this function improved the performance substantially (~75% testing accuracy and ~71% precision), showing that spatial transformations applied before flattening the data confused the model. Further improvements through applying dropout, batch normalization, and L2 regularization allowed for better generalization, although overfitting was still a recurring issue. The optimized final FNN model achieved ~71% test accuracy and ~67% precision. These results demonstrated the limitations of FNNs in extracting spatial features and reinforced that they are not ideal for this image classification task. It is important to note that, in terms of speed, FNNs are trained significantly faster than CNNs due to their simplistic architectures and reduced computational complexity. Despite the increased training time, CNNs remain the preferred choice because accuracy and reliability are critical for the required task. The NAS was employed to demonstrate automated model building. Using EfficientNetB7 (a popular CNN architecture) as a pre-trained backbone and with augmented grayscale input, this model achieved 91.4% testing accuracy and 88% precision, with only a 3.67% accuracy drop from training to testing, indicating excellent generalization. The model misclassified 19 out of 209 defective electrodes. The NAS AI methodology demonstrates the power of automated model building without requiring deep knowledge in machine learning or deep learning, reinforcing how accessible deep learning is becoming. The predictive capabilities of the established AI system, however, could be improved by collecting more data, including in the form of higher-resolution images. For more accurate defect capture, expanding these datasets with more labeled samples could also help in predicting a wider variety of defect types such as smudges, cracking, lack of connectivity, and contamination. The use of preprocessing methods such as feature extraction techniques could also be considered for creating AI models with more specific defect type predictability.

Acknowledgments

None.

Funding

This publication arises from research supported by Research Ireland under Grant Number 21/RC/10295_P2 and is co-funded by the European Regional Development Fund. This work is supported by I-Form.

Conflict of interest

Dermot Brabazon is one of the Associate Editors of the journal but was not involved in the editorial and peer-review process conducted for this paper, directly or indirectly. Separately, other authors declared that they have no known competing financial interests or personal relationships that could have influenced the work reported in this paper.

Author contributions

Conceptualization: Gareth Quinn, Anesu Nyabadza, Dermot Brabazon
Formal analysis: Gareth Quinn
Funding acquisition: Dermot Brabazon
Investigation: Gareth Quinn, Achu Titus
Methodology: Gareth Quinn
Supervision: Dermot Brabazon
Writing–original draft: Achu Titus
Writing–review & editing: Gareth Quinn, Sithara Sreenilayam, Éanna McCarthy, Anesu Nyabadza, Dermot Brabazon

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Availability of data

The data supporting this article have been included in the Supplementary File.

References

1. Vida J, Solak S, Shao Y, Homola T, List-Kratochvil E, Hermerschmidt F. Sintering of inkjet-printed silver nanoparticles by large-area atmospheric pressure nitrogen plasma. *Appl Phys A*. 2025;131(2):91.

   doi: 10.1007/s00339-024-08206-y

2. Du X, Wankhede SP, Prasad S, *et al*. A review of inkjet printing technology for personalized-healthcare wearable devices. *J Mater Chem C*. 2022;10(38):14091-14115.

   doi: 10.1039/D2TC02511F

3. Carey T, Cacovich S, Divitini G, *et al*. Fully inkjet-printed two-dimensional material field-effect heterojunctions for wearable and textile electronics. *Nat Commun*. 2017;8(1):1202.

   doi: 10.1038/s41467-017-01210-2

4. Rosati G, Cisotto G, Sili D, *et al*. Inkjet-printed fully customizable and low-cost electrodes matrix for gesture

recognition. *Sci Rep.* 2021;11(1):14938.

doi: 10.1038/s41598-021-94526-5

5. Yao C, Wang L, Wang Q, *et al.* Deep-learning-guided evaluation method for the high-volume preparation of flexible sensors based on inkjet printing. *ACS Appl Interfaces Mater*. 2024;16(10):13326-13334.

doi: 10.1021/acsami.4c00322

6. Rossetti M, Srisomwat C, Urban M, *et al.* Unleashing inkjet-printed nanostructured electrodes and battery-free potentiostat for the DNA-based multiplexed detection of *SARS-CoV-2* genes. *Biosens Bioelectron*. 2024;250:116079.

doi: 10.1016/j.bios.2024.116079

7. Mei Y, Cannizzaro C, Park H, *et al.* Cell-compatible, multi-component protein arrays with subcellular feature resolution. *Small.* 2008;4(10):1600.

doi: 10.1002/smll.200800363

8. Sarma Choudhury S, Katiyar N, Saha R, Bhattacharya S. Inkjet-printed flexible planar Zn-MnO_2 battery on paper substrate. *Sci Rep.* 2024;14(1):1597.

doi: 10.1038/s41598-024-51871-5

9. Kumar P, Ebbens S, Zhao X, *et al.* Inkjet printing of mammalian cells-theory and applications. *Bioprinting*. 2021;23:e00157.

doi: 10.1016/j.bprint.2021.e00157

10. Bernasconi R, Invernizzi GP, Gallo Stampino E, Gotti R, Gatti D, Magagnin L. Printing MEMS: Application of inkjet techniques to the manufacturing of inertial accelerometers. *Micromachines (Basel)*. 2023;14(11):2082.

doi: 10.3390/mi14112082

11. Chen S, He Z, Choi S, Novosselov IV. Characterization of inkjet-printed digital microfluidics devices. *Sensors (Basel)*. 2021;21(9):3064.

doi: 10.3390/s21093064

12. Tian L, Liu J, Chen X, Branicio PS, Lei Q. Mechanisms and strategies to achieve stability in inkjet printed 2D materials electronics. *Adv Electron Mater*. 2025;11(3):2400143.

doi: 10.1002/aelm.202400143

13. Yunker PJ, Still T, Lohr MA, Yodh AG. Suppression of the coffee-ring effect by shape-dependent capillary interactions. *Nature*. 2011;476(7360):308-311.

doi: 10.1038/nature10344

14. Soltman D, Subramanian V. Inkjet-printed line morphologies and temperature control of the coffee ring effect. *Langmuir*. 2008;24(5):2224-2231.

doi: 10.1021/la7026847

15. Sowade E, Ramon E, Mitra KY, *et al.* All-inkjet-printed thin-film transistors: Manufacturing process reliability by root cause analysis. *Sci Rep.* 2016;6(1):33490.

doi: 10.1038/srep33490

16. Chen Z, Gengenbach U, Koker L, *et al*. Systematic investigation of novel, controlled low-temperature sintering processes for inkjet printed silver nanoparticle ink. *Small*. 2024;20(21):2306865.

doi: 10.1002/smll.202306865

17. Kwon KS, Jo JY. Towards zero-defect inkjet printing via piezo self-sensing signals. *Sens Actuators A Phys*. 2025;393:116755.

doi: 10.1016/j.sna.2025.116755

18. Kwon KS. Methods for detecting air bubble in Piezo inkjet dispensers. *Sens Actuators A Phys*. 2009;153(1):50-56.

doi: 10.1016/j.sna.2009.04.024

19. Jeurissen R, De Jong J, Reinten H, *et al*. Effect of an entrained air bubble on the acoustics of an ink channel. *J Acoust Soc Am*. 2008;123(5):2496-2505.

doi: 10.1121/1.2835624

20. Law KNC, Yu M, Zhang L, *et al*. *Enhancing Printed Circuit Board Defect Detection through Ensemble Learning*. New Jersey: IEEE; 2024. p. 37-44.

doi: 10.1109/FITYR63263.2024.00013

21. Novikovs A, Tsebriienko T, Trausa A, *et al*. Novel terpineol-based silver nanoparticle ink with high stability for inkjet printing. *Nanomaterials (Basel)*. 2025;15(13):955.

doi: 10.3390/nano15130955

22. Beedasy V, Smith PJ. Printed electronics as prepared by inkjet printing. *Materials (Basel)*. 2020;13(3):704.

doi: 10.3390/ma13030704

23. Amini A, Gan TH. A computer vision-based quality assessment technique for R2R printed silver conductors on flexible plastic substrates. *Appl Sci*. 2023;13(2):1084.

doi: 10.3390/app13021084

24. Kim J, Jung Y, Parajuli S, *et al*. Quantifying printing quality for printed electrodes via deep learning and spatial association: Empowering process optimization. *Adv Intell Syst*. 2025;7:2500178.

doi: 10.1002/aisy.202500178

25. Ogunsanya M, Isichei J, Parupelli SK, Desai S, Cai Y. *In-situ* droplet monitoring of inkjet 3D printing process using image analysis and machine learning models. *Proced Manuf*. 2021;53:427-434.

doi: 10.1016/j.promfg.2021.06.045

26. Huang J, Segura LJ, Wang T, Zhao G, Sun H, Zhou C. Unsupervised learning for the droplet evolution prediction and process dynamics understanding in inkjet printing. *Addit Manuf*. 2020;35:101197.

doi: 10.1016/j.addma.2020.101197

27. Nandipati M, Ogunsanya M, Desai S, *et al*. Predictive

models for 3D inkjet material printer using automated image analysis and machine learning algorithms. *Manuf Lett*. 2024;41:810-821.

doi: 10.1016/j.mfglet.2024.09.101

28. Polomoshnov M, Reichert KM, Rettenberger L, *et al*. Image-based identification of optical quality and functional properties in inkjet-printed electronics using machine learning. *J Intell Manuf*. 2025;36(4):2709-2726.

    doi: 10.1007/s10845-024-02385-4

29. Carou-Senra P, Ong JJ, Castro BM, *et al*. Predicting pharmaceutical inkjet printing outcomes using machine learning. *Int J Pharm X*. 2023;5:100181.

    doi: 10.1016/j.ijpx.2023.100181

30. Sahu A, Aaen PH, Damacharla P, *et al*. *An Automated Machine Learning Approach to Inkjet Printed Component Analysis: A Step Toward Smart Additive Manufacturing*. New York: IEEE; 2024. p. 1-6.

    doi: 10.1109/WMCS62019.2024.10618993

31. Lall P, Soni V, Kulkarni S, Miller S, *et al*. *Comparison of Machine Learning Approaches for Correlating Print Process Parameters to Realized Physical and Electrical Characteristics of Printed Electronics Using Inkjet Platform*. New York: American Society of Mechanical Engineers; 2023. p. V001T03A012.

    doi: 10.1115/IPACK2023-112056

32. Queralto A, Pacheco A, Jimenez N, *et al*. Defining inkjet printing conditions of superconducting cuprate films through machine learning. *J Mater Chem C*. 2022;10(17):6885-6895.

    doi: 10.1039/d1tc05913k

33. Kim S, Cho M, Jung S. The design of an inkjet drive waveform using machine learning. *Sci Rep*. 2022;12(1):4841.

    doi: 10.1038/s41598-022-08784-y

34. Kim S, Wenger R, Bürgy O, *et al*. Predicting inkjet jetting behavior for viscoelastic inks using machine learning. *Flex Print Electron*. 2023;8(3):035007.

    doi: 10.1088/2058-8585/acee94

35. Jiang L, Wolf R, Alharbi K, Qin H, *et al*. *In situ* monitoring and recognition of printing quality in electrohydrodynamic inkjet printing via machine learning. *J Manuf Sci Eng*. 2024;146(11):110901.

36. Brishty FP, Urner R, Grau G, *et al*. Machine learning based data driven inkjet printed electronics: Jetting prediction for novel inks. *Flexible and Printed Electronics*. 2022;7(1):015009.

    doi: 10.1088/2058-8585/ac5a39

37. Kim S, Cho M, Jung S, *et al*. Reinforcement learning-based dynamic optimization of driving waveforms for inkjet printing of viscoelastic fluids. *Langmuir*. 2025;41(17):10831-10840.

    doi: 10.1021/acs.langmuir.4c05141

38. Phung TH, Park SH, Kim I, Lee TM, Kwon KS. Machine learning approach to monitor inkjet jetting status based on the piezo self-sensing. *Sci Rep*. 2023;13(1):18089.

    doi: 10.1038/s41598-023-45445-0

39. Li M, Yin S, Liu Z, Zhang H. Machine learning enables electrical resistivity modeling of printed lines in aerosol jet 3D printing. *Sci Rep*. 2024;14(1):14614.

    doi: 10.1038/s41598-024-65693-y

40. Shirsavar MA, Taghavimehr M, Ouedraogo LJ, *et al*. Machine learning-assisted E-jet printing for manufacturing of organic flexible electronics. *Biosens Bioelectron*. 2022;212:114418.

    doi: 10.1016/j.bios.2022.114418

41. Nyabadza A, Titus A, McCarthy É, *et al*. Fabrication and inkjet printing of manganese oxide electrodes for energy storage. *Chem Eng J Adv*. 2025;22:100761.

    doi: 10.1016/j.ceja.2025.100761