LEX LOCALIS-JOURNAL OF LOCAL SELF-GOVERNMENT
ISSN:1581-5374 E-ISSN:1855-363X
VOL. 24, NO. 2(2026)

CNN WITH OPTIMIZED FULLY CONNECTED LAYER BASED INTELLIGENT DEFECT DETECTION SYSTEM FOR FOOD PACKAGE

[1*]Dr. S. Ananth

[1*]Associate Professor, Department of Information Technology, Mahendra Engineering College (Autonomous), Namakkal, Tamil Nadu, India.

Corresponding author email: sananthphd@gmail.com[1]

Abstract
The quality of food packaging plays a significant role in determining the shelf life of consumer products. This is a good preservative that extends the shelf life of food. In some cases, food packaging may contain some defects, such as blow holes, holes, burrs, shrink defects, mold material defects, metal defects, metal defects, etc., which may affect the purity of the food. To avoid these issues, in this paper, an efficient automatic detection system using a deep learning (DL) algorithm is proposed. The proposed approach consists of two stages namely, pre-processing and defect detection. Initially, the images are collected from the dataset. Median filters are used to remove noise from the images. Once the pre-processing has been completed, the pre-processed images are given to the classifier. Optimised fully connected CNNs are proposed in this paper for classification. With adaptive reptile search optimization algorithm (ARSA), full-connected layer weight parameters are optimized for enhanced performance. The proposed algorithm effectively detects the problem and correctly segments the affected region. The performance of the proposed approach is analysed based on accuracy, sensitivity, specificity, and precision.

Keywords: Food package, Defect detection, deep learning, CNN, adaptive reptile search optimization.

1. Introduction

It is becoming increasingly important for the quality control of products in industrial production to perform quality visual inspections. By detecting defects visually, this process ensures the quality of the product. In the production process, poor product quality does more harm than good to all participants [1]. Food quality affects productivity. Poor quality food products can reduce sales and result in loss of cash and raw materials.Reliability of the products and the market share of the factory determine the quality of the product [2]. Long-term market occupation can only be achieved by high quality products. Although many factories perform quality visual inspections using artificial methods, which require a substantial amount of manpower and consume a large amount of financial resources, many factories use manual methods to detect defects. Since people's energy levels are limited and human inspection work is insignificant, workers working continuously for long periods may become fatigued, resulting in significant economic losses. Automated defect detection must be integrated into the manufacturing process in order to achieve this goal[3].

Increased requirements for accuracy, performance, objectivity, and reliability necessitated the introduction of computer-based image processing techniques, including magnetic resonance imaging, charge-coupled device camera, electrical tomography, ultrasound, and computed tomography for image acquisition; pixel and local pre-processing approaches to image pre-processing; threshold-based, region-based, gradient-based and classification-based methods for image segmentation; features colour, Size, shape and texture for object measurement and fuzzy logic, statistical and neural network methods for classification [4]. In some cases, image processing can be a good alternative to human visual decision-making.

The field of computer vision has made great progress in recent years due to deep learning algorithms. Convolution neural networks (CNNs) represent deep learning algorithms that have improved accuracy and detection speed compared to traditional methods when it comes to recognizing targets [5]. Currently, there are mainly two types of target recognition algorithms: one is a two-level algorithm, such as the R-CNN series algorithm, which is high in accuracy, but also has a rich network structure [6]. Real-time detection is not possible when target recognition occurs at slow speeds. It is also possible to use regression-based first-order algorithms, such as SDD, YOLO sequence algorithms, etc. Due to its fast inference speed and high practicality, it is capable of detecting and recognizing objects in real-time [7]. The following contribution is made in this paper to detect food package defect.

- This paper presents an efficient automatic detection system based on the deep learning algorithm for defect detection.
- Initially, the images are collected from the dataset. Then, noises present in the images are removed using a median filter.
- For the classification purpose, CNN with optimized fully connected layer is used.
- To enhance the performance of fully connected layer, weight parameters of it are optimized using the adaptive reptile search algorithm (ARSA).
- The proposed approach is analysed based on accuracy, sensitivity, specificity, and precision.

In this paper, Section 2 discusses the existing methodology regarding the objective. Section 3 explains the methodology used to develop a package defect detection system using DL. Section 4 shows the experimental results of the approach and the conclusion is given in Section 5.

2. Literature survey

In 2017 Wang, et al., [8] proposed a deep convolution neural networks (CNN) system that automatically extracts powerful features with minimal prior knowledge of images for defect detection. It is resilient to noise. An experimental evaluation of this CNN model was conducted on a benchmark dataset. Models based on this model were shown to classify images into the appropriate image class and to detect defects within them. In addition to achieving a high accuracy rate, this model also maintains high processing speed in order to guarantee real-time defect detection on the DAGM dataset.

In 2017 Chen, et al., [9] proposed a fast auto-clean convolution neural network (CNN) model for online prediction of food items, which addresses the issues of complexity of food items, uniformity of illumination, focus of displacement, and complexity characteristics of food images. The goal of this paper was to propose a technique for automatic image cleaning and classification, which predicts multi-class clean images based on self-cleaning CNNs. After the convolution layers have been fed forward into the CNNs, they are fed into the loss layers in order to generate the composite features. In order to optimize image datasets, it uses multi-class classification to intelligently classify and classify images. Additionally, an algorithm for online prediction is proposed for improving image recognition. Experiments have shown the proposed algorithm and model to be accurate and perform well.

In 2021 Hu, Liu and Cui [10] proposed an intrusion detection system based on convolution neural networks. Training and testing were done using NSL-KDD. The Fruit Fly Optimization Algorithm was used to perform the pre-training process to obtain the best training data and resample the weights to solve the class imbalance problem. Pre-training was conducted using

FOA to address class imbalance. In training, blocks are resampled according to resampling weights. Tests were conducted on the NSL-KDD dataset. With this method, accuracy was higher compared to CNN without data balancing.

In 2022 Gao, et al., [11] proposed a method for detecting gear face defects based on deep convolution generative adversarial networks (TCGANs) and lightweight convolution neural networks (CNNs). DCGAN and traditional data augmentation methods were initially used to augment the training data. To classify defects, a lightweight CNN model based on state-of-the-art VGG11 was used. Dropout layers and leaky ReLU implementation functions are introduced in the proposed CNN. A high score of 98.40% was achieved by the proposed framework compared to the classic Vgg11 network model.

In 2018 Yangping, et al., [12] proposed a method to develop the detection rate of printing defects by combining gray scale and gradient differences. According to the gray scale difference threshold of the unweighted neighbourhood, the gray scale difference between the template image and the probed image was performed to determine the defect in the nonedge region. After that, the slope difference between the template and probed images was used to determine the edge defect using the gray scale difference threshold. With two different image fusions, contrast artifacts were effectively eliminated while real defects were retained. Results showed that this method have a higher detection rate than traditional differential methods due to the effective removal of artifacts.

In 2017 Laila, et al., [13] proposed the use of Convolution Neural Networks (CNN), a deep learning technology for image processing. In terms of classifying images, CNN proved to be very effective. For the purpose of checking the accuracy of data, this CNN method was implemented using a four-fold validation cross. It has been found that initializing the parameter configuration of the CNN architecture model accelerated the process of network training. Based on the results of the tests, it was able to detect defects in mangosteen fruit with an efficiency of 97%.

3. Proposed methodology

3.1 Overview

Fig.1 shows the architectural diagram of a defect detection system for food packaging. Initially images are collected from the dataset. A median filter is used to remove unwanted noise from the images. It removes noise by replacing the center value of the window with the average value of the center neighbourhood. The pre-processed images are then fed as input to the classifier. Here, convolution neural network (CNN) with optimized fully connected layer (FCL) is used for the classification process. For the improvement of a fully connected layer, the weight parameters are optimized using the Adaptive Reptile Searchoptimization Algorithm (ARSA). This method effectively diagnoses the problem and correctly segments the affected area.

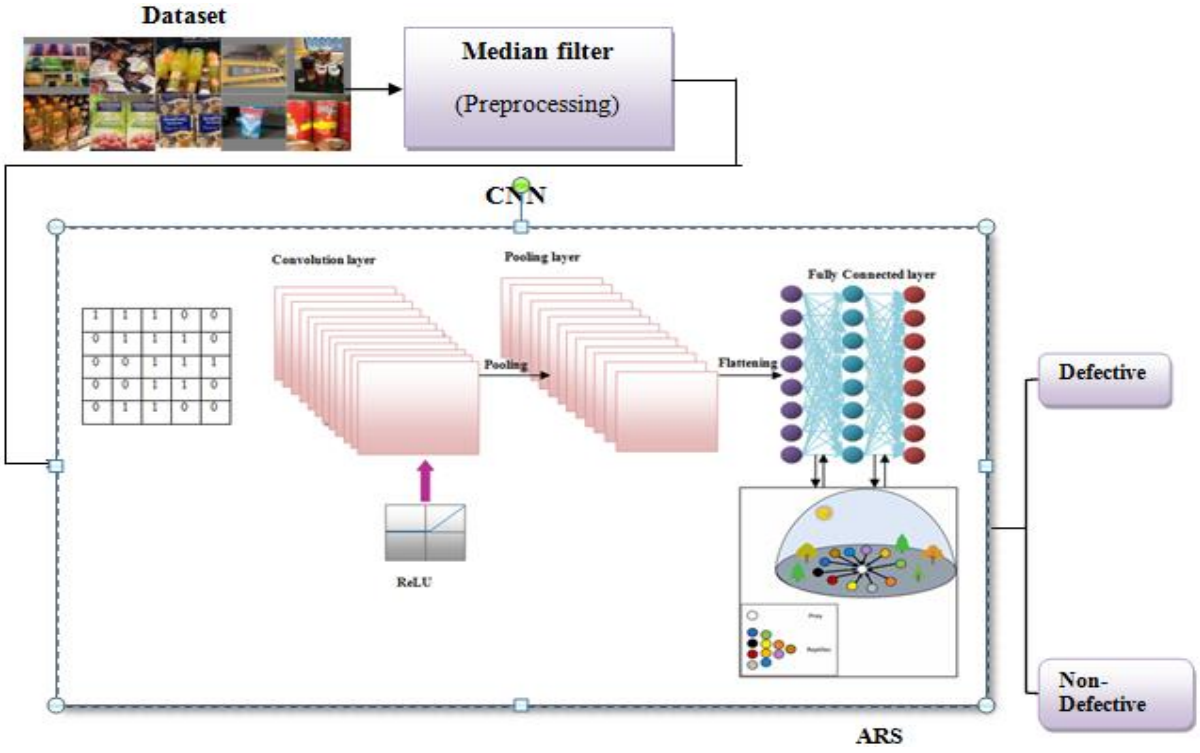


Figure 1: Overall architectural diagram for defect detection system for food packaging

3.2Median filter

Non-linear filter for removing salt-and-pepper noise is known as median filters. It removes noise by replacing the center value of the window with the median value of the center neighbourhood. According to statistics, the median of a given sorted list of numbers is equal to its center value. Signal processing began with the use of the median by J. W. Tukey [14]. It is possible for there to be more than one center value if the number of items in the list is even. It is convenient to use odd list sizes when searching for a median if the count of the list is odd. The Median Filter sorts vector magnitudes within a mask by magnitude as part of its operation. By replacing the studied pixel with the median magnitude, the pixel with the median magnitude represents the median magnitude. By relying on the median of the data instead of the mean, the Simple Median Filter is an advantage over the Mean Filter. The mean of a set of images can be significantly skewed by a single noisy pixel in an image. Simple Median Filter can produce the same result as the original pixel in some cases. It is known as the root of the mask if a pixel does not change as a result of filtering. Each signal converges to a root signal after sufficiently many iterations of median filtering.

Considering $\{V_1, V_2, V_3, \cdots, V_k\}$ as k random variables, in ascending order, the observations are as follows:

$$V_{(1)} \le V_{(2)} \le V_{(3)} \le \cdots \le V_{(K)}. \tag{1}$$

As a result of multiple random variables, $V_{(j)}, j = 1,2,\cdots,k$ represents the j^{th}order statistics for each random variable. In order to simplify things, let us assume that $k = 2l-1$ is an odd

number. We can consider $V^k = \{V_1, V_2, V_3, \cdots, V_k\}$ being a collection of these random variables. Based on the median filter, we obtain the following output:

$$MedFi(V^k) = V_{(l)} \tag{2}$$

In the following section CNN with fully connected layer is proposed to classify the output signal.

3.3Convolution Neural network (CNN)

Convolution neural networks (CNNs) are one type of artificial neural networks (ANNs). Unlike handcrafted features extracted by elaborate algorithms, CNNs are capable of learning hierarchies of features from input image matrices automatically. Several revolutionary breakthroughs have been achieved by CNN models in computer vision recently, including classification, segmentation, and object tracking. With fewer connections and fewer parameters, CNN architecture incorporates weight parameter sharing and pooling operations.

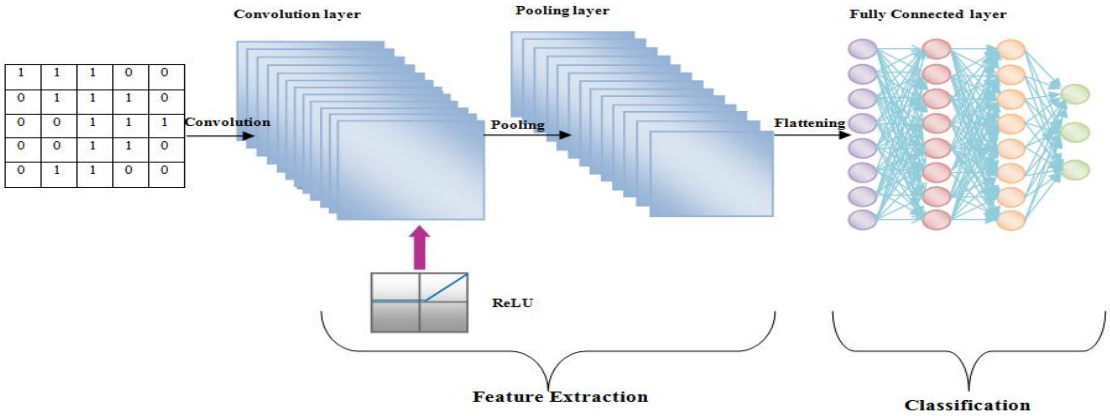


Figure 2:architecture of CNN

Fig.1 represents the structure of CNN. There are several nested convolution layers followed by fully connected layers in a typical CNN architecture. It is possible to present this type of network in a simplified way as follows:

Input: The input of a CNN typically consists of matrices of 3-channel color or 1-channel gray images containing intensity values at each position.

Conv (Convolution layer): Each of the convolution layers applies a set of filters to a small region of the output of the last layer. In addition, the filters are usually learnable matrices of small size, such as 3 x 3 or 5 x 5. To extract one feature from an image, parameter sharing is used to convolve one filter across all spatial dimensions.

ReLU (Rectified Linear Units): In convolution and fully connected layers, ReLU are usually used as activation functions for adding non-linear transformations. This function has the formula f (y) = max (0, y). In recent years, deep networks have been so successful because ReLU prevents expedites training convergence, gradient saturation, while maintaining as much original value as possible, which has been demonstrated experimentally to be superior to conventional sigmoid-like activation functions.

Pool: The pooling layer reduces the spatial size of the output by downsampling both spatial dimensions in a nonlinear manner. The network parameters are reduced to reduce computation

costs and network parameters. The maximum value of the input feature map can be generated by max pooling if it is placed between two successive convolution layers.

FCL (Fully connected layer): An artificial neural network ends with its FCLs. In the last layer, each neuron in the FCL is connected to every other neuron. In this network, the last FCL generates the output from all the input labels; it has N neurons. Each label's probability is represented by its value in the N-dimensional output using the softmax function.

$$P(x_j) = \frac{\exp(x_j)}{\sum_{j=1}^{N} \exp(x_j)}, \quad 3)$$

$P(x_j)$ represents the probability of predicting the j^{th} value calculated by the second last layer. When all layers are stacked together, they form a CNN, which then feeds back the input to make decisions.

The FCL layer has separate weights for all input and output units. As well as this, this layer has the bias for each output node, so the weight parameter for FCL is

$$Y = (w_1 x_1 + w_2 x_2 + \cdots + w_j x_j) + b \quad (4)$$

Where, w represents the weight parameter, w_j represents number of weight parameters and b represents bias value. The performance of fully connected layer will improved by optimizes the weight parameters using ARSA algorithm.

3.4Adaptive Reptile Search Optimization (ARSO)

Crocodile hunting behaviour is used to develop a meta-heuristic optimizer called Reptile Search optimization Algorithm (RSA). In a recent study, Abualigah et al [15] developed the RSA metaheuristic algorithm. RSA replicates crocodile hunting behaviour in its natural habitat. It is generally considered that crocodiles belong to the family Crocoodylinae, and they prefer a climate with plenty of water and food. Crocodiles belong to the amphibian family and can hunt in the water. They also live outside of water.

i. **Vision**: Crocodiles are one of the few animals with night vision. Because of their poor night vision, they hunt at night unlike other animals.
ii. **Eating**: They eat fish, cows, deer, baby elephants, zebras and small crocodiles as part of their diet in the environment around their habitat. Also, large crocodiles will eat cats and sharks in addition to other predators. Without food sources in the environment, it is capable of surviving long periods without food. Sources indicate that some of them consume fruits.
iii. **Physical action**: Swimming, walking and running are all abilities of crocodiles. Swimmers steer with their tails, ignoring their legs. Their legs facilitate movement and carry their bodies when they walk, while their tails act as a balance and steering mechanism.The tail transmits energy to the body for forward acceleration, allowing crocodiles to run out of water to attack prey.
iv. **Cognition:** Crocodiles are able to identify their prey to determine which animals frequent the water.
v. **Hunting:** In order to catch animals that dive or swim up from the shore to drink, snucking into the water is a common practice. Once their prey approaches the water, crocodiles attack stealthily. Once crocodiles catch their prey, they drag them into the

water and drown them. A crocodile eventually eats its prey whole after cutting it up into large pieces. Prey is often shared between crocodiles through fighting.

vi. It is common for crocodiles to live in groups. In preparation for predation ambushes, crocodiles use this pattern. Each member of the group plays a part in predatory behaviour. Crocodiles attack animals that drink from riverbanks in order to push them to the water, where they attack the prey from beneath the water.

As illustrated in figure 1, the RSA procedure steps are described in detail.

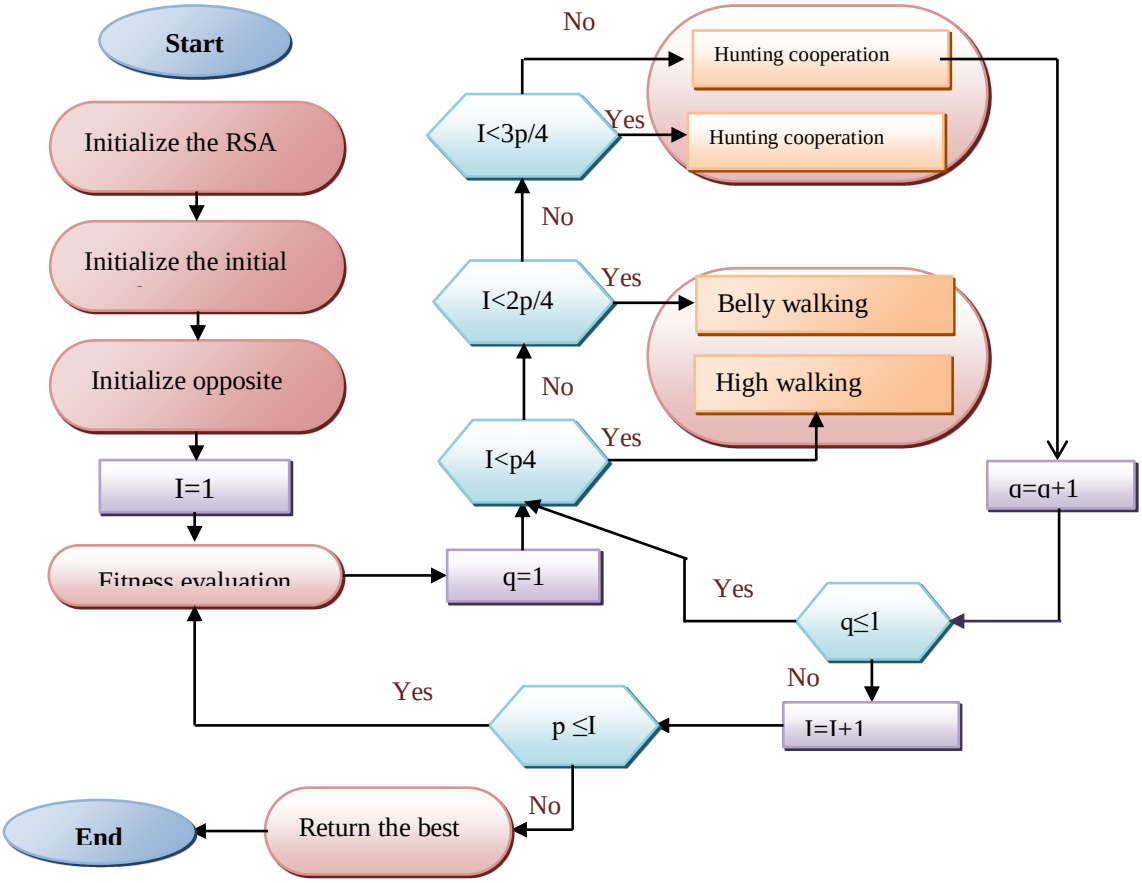


Figure 3: Steps for ARSA

i. **Initialization:** ARSA should be executed with both weight parameters and algorithmic parameters initialized. Weight parameters are composed of two parameters, C and I. C represents the number of crocodiles, and I represent the number of iterations. Furthermore, ARSA uses two algorithmic parameters, α and β. In order to achieve the right balance between exploration and exploitation abilities, these two algorithmic parameters control the two abilities, respectively.

ii. **Population Initialization:** Using the following equation, an initial set of solutions is generated randomly

$$Y_{p,q} = Y_q^{\min} + rc\,D \times (Y_q^{\max} - Y_q^{\min}), \forall p = 1,2,\cdots,C, \quad \forall q = 1,2,\cdots,D, \tag{5}$$

Here, Y represents the weight parameter. The decision variable $Y_{p,q}$ represents the p[th] solution at the q[th] position. There are two upper and lower bounds to the decision variable at the q[th] position: $Y_q^{\max}$ *and* $Y_q^{\min}$. Rand indicates the random number between 0 and 1, while D indicates the total number of decision variables at each solution. In Y, a set of C solutions is generated and is stored as follows:

$$Y = \begin{bmatrix} Y_{1,1} & Y_{1,2} & \cdots & Y_{1,D-1} & Y_{1,D} \\ Y_{2,1} & Y_{2,2} & \cdots & Y_{2,D-1} & Y_{2,D} \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ Y_{C,1} & Y_{C,2} & \cdots & Y_{C,D-1} & Y_{C,D} \end{bmatrix} \tag{6}$$

$Y_{1,D} = \{w_{1,1}, w_{1,2}, \cdots, w_{1,j}\}_{1,D}$, where $Y_{1,D}$ indicates the weight parameter.In each row $Y_p = (Y_{p,1}, Y_{p,2}, \cdots, Y_{p,D-1}, Y_{p,D})$ represents the solution of position *p*.

Also,

$$Y = \begin{bmatrix} Y_{1,1} & Y_{1,2} & \cdots & Y_{1,C-1} & Y_{1,C} \\ Y_{2,1} & Y_{2,2} & \cdots & Y_{2,C-1} & Y_{2,C} \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ Y_{D,1} & Y_{D,2} & \cdots & Y_{D,C-1} & Y_{D,C} \end{bmatrix}$$

The solution of position q is represented by $Y_q = (Y_{q,1}, Y_{q,2}, \cdots, Y_{q,C-1}, Y_{q,C})$ in each row.

iii. **Opposite solution:** In order to create candidate solutions (the initial position of the crocodile) without prior knowledge, the random initialization method is applied using Equation (5). The distance between the starting point and the optimum solution determines convergence speed and performance. Generally, the approach performs better when the randomly generated solutions have a lower target function value. An adaptive version of the classic RSA approach is presented in this study in order to increase convergence and the likelihood of finding global optima. In contrast to the previous ARSA, the new method creates the opposite positions for each solution after first generating the initial random solutions (i.e., the positions of the rats) based on the opposite number principle. It is necessary to define the opposite number in order to describe the initialization of the new population. A vector Y with a dimension of C is shown below:

$$Y = (Y_1, Y_2, \cdots, Y_c) \tag{7}$$

Where, Y_q is $Y_q^{\min}, Y_q^{\max}$. As a result, the opposite point of Y_q, represented by $\overline{Y_q}$

$$\overline{Y_q} = (Y_q^{\min} - Y_q^{\max}) - Y_q, q = 1,2,\cdots,c \quad (8)$$

By assuming that X_p is a random generated solution in C-dimensional space, the ARSA population is initialized. As a result of solving inverse equation (8), $\overline{Y_q}$ represents a stochastic solution. As the objective function of the evaluation, fn(.) is used to evaluate both solutions (i.e., Y_p and $\overline{Y_q}$. If $f_c(Y_q)$ is superior to $f_c(\overline{Y_q})$ (i.e $f_c(\bar{y}_q) < f_c(y_q)$, then replaces agent $\overline{Y_q}$; otherwise, continue with agent q. The best initial solution is selected in the first iteration. When dealing with tasks that are more complex, the ARSA method can become trapped in local optima. Although it is more efficient than traditional methods, it tends to trap in local optima. The search process may require much iteration for certain agents to remain stationary. It will address these shortcomings and improve exploration and search capabilities by replacing the poorest solution with the best fitness value.

iv. **Fitness Function:**In this stage, the fitness of each solution is calculated along with the opposite solution. Each node along the routing path has an energy and confidence value, which are used to calculate a fitness function. By summing all intermediates along the routing path, the fitness of the weights is calculated. A loss function is a function used to predict an error in the neural network that is detected during training. For this study, the fitness function is the minimum loss value. Based on the loss function, the fitness function of each solution is calculated as follow:

$$f(Y_p) \quad \forall_p = \min[loss] \quad (9)$$

v. **Encircling phase:** In the RSA, crocodiles explore the water. According to equation (10), we introduce two strategies for locating new regions in the search space of a problem: high walking and belly walking. Using equation (10) we can see that high walking is controlled by i ≤ I/4, belly walking is controlled by I/4 <i ≤ 2I/4.

$$Y_{p,q}(i+1) = \begin{cases} Y_q^{best}(i) - \eta_{p,q}(i) \times \beta - r_{p,q}(i) \times rc\,D, i \le \frac{I}{4}, \\ Y_q^{best}(i) \times Y_{R1,q}(i) \times ES(i) \times rcD, \quad \frac{I}{4} < i \le \frac{2I}{4}, \end{cases} \quad (10)$$

The decision variable $Y_{p,q}$ corresponds to the solution at position q. The best solution obtained at i iterations is $Y_q^{best}(i)$. Iteration i+1 represents the new iteration, while iteration i represents the previous iteration. Using (11), we calculate the hunting operator for the qth position in the ith solution. In the high-walking strategy, β controls its exploration capability. The value of β is set to 0.1. A random value between zero and one is generated by rcD. The decision variable $Y_{r1,q}(i)$ is at the qth position in the r1th solution, where $r1 \in [1,C]$. It is calculated that $\eta_{p,q}(i), F_{p,q}$ and $Avg(Y_p)$ are, respectively,

$$\eta_{p,q} = Y_q^{best} \times F_{p,q}, \quad (11)$$

$$F_{p,q} = \alpha + \frac{Y_{p,q} - Avg(Y_p)}{Y_q^{best} \times (Y_q^{\max} - Y_q^{\min}) + \varepsilon}, \quad (12)$$

$$Avg(y_p) = \frac{1}{D}\sum_{q=1}^{D} Y_{p,q}, \tag{13}$$

Y^{best}at the q^{th}positionand$Y_p$at its corresponding position in this example constitute a percentage difference between the decision variables. α controls RSA exploration ability during cooperative hunting by setting it to 0.1. The value should be random between 0 and 2. In the current solution, Y_p, all decision variables have a combined average value of $Avg(Y_p)$. P[th] solution positions q are reduced by $R_{p,q}(i)$, and ES(i) factors assign their values.

$$R_{p,q} = \frac{Y_q^{best} - Y_{r2,q}}{Y_q^{best} + \varepsilon} \tag{14}$$

$$ES(i) = 2 \times r3 \times (1 - \frac{1}{I}), \tag{15}$$

In a population of solutions, r2 represents the probability of selecting one solution among them at random. The value of r3 can be represented as an integer between 0 and 1.

vi. **Hunting phase:** This is how crocodiles exploit their habitat in the ARSA. To find optimal solutions based on the existing research regions, ARSA adopts two strategies in this phase: coordination and cooperation. Coordinating hunting is controlled by $i \le 3I/4$, while coordinating hunting is controlled by [1].

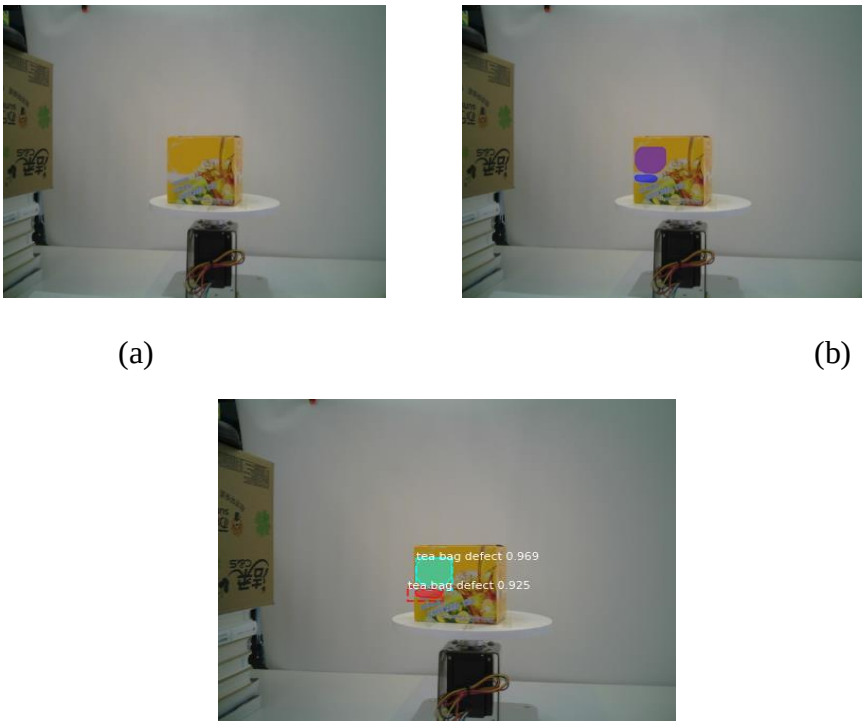
$$Y_{p,q}(i+1) = \begin{cases} Y_q^{best}(i) \times F_{p,q}(i) \times rcD, & \frac{2I}{4} < i \le \frac{3I}{4}, \\ Y_q^{best}(i) - \eta_{p,q}(i) \times \varepsilon - R_{p,q}(i) \times rcD, & \frac{3I}{4} < i \le I. \end{cases} \tag{16}$$

vii. **Stop Criterion:** The process is repeated from Step 3 to Step 5 until $I$is reached.

4. Results and discussion

In this work, we applied deep learning method and classification to detect imperfect food package using proposed ARSA defect detection. Images of food packages were obtained from the collected dataset. In this system, human intervention is not required to extract features from input images. Using a computer with 8GB of memory running Windows 10 operating system and an Intel Core i7 processor, we simulate the proposed defect detection model using Python. A pre-processing technique, i.e. median filtering, was used to enhance the input images. Standard performance measures such as precision, sensitivity, specificity, recall, score, and precision were determined and used to evaluate the proposed system. An optimal weight is the objective of the proposed deep learning model. In this work there are three kinds of images tested by this proposed technique. They are a lemon flavour tea pack, a food cane and a water bottle. Table 1 shows the Metrics evaluation for Lemon flavoured tea pack image

Our proposed work compares three techniques; Namely, CNN- PSO (Particle Swarm Optimization), CNN and SVM (Support Vector Machine). Fig.4 shows the defect detection process of lemon tea flavour tea packet from the dataset. There are two affected regions are segmented, the first segmented region have the defect of 96.9% and the second region have 92.5%.



(a) (b)

(c)

Figure 4: a) Uploaded input image- lemon flavoured tea pack, b) Image testing and c) Prediction

Table 1: Metrics evaluation for Lemon flavoured tea pack image

	Accuracy	**Sensitivity**	**Specificity**	**Precision**	**Recall**	**FScore**
CNN-ARSA	99.19	95.68	97.67	98.67	98.69	98.75
CNN-PSO	97.05	93.28	95.67	92. 51	91.52	93.16
CNN	96.72	93.10	94.54	92.21	91.25	92.06
SVM	95.53	92.05	93.42	91.11	90.99	91.87

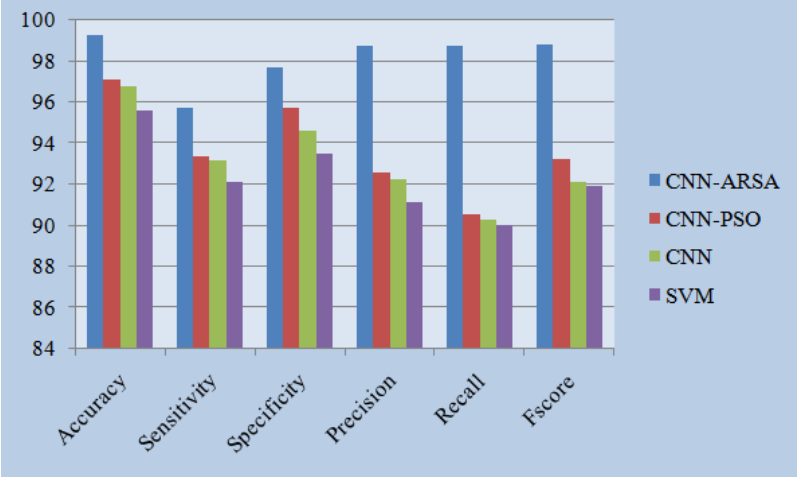


Figure 5: Metrics evaluation for the Lemon flavour tea package image

Fig.5 shows the quantitative evaluation for the lemon flavoured tea package. From the figure, our proposed task achieves a high accuracy of 99.19%, which is 2.14% higher than CNN-PSO, 2.47% higher than CNN and 3.66% higher than SVM. The sensitivity of the proposed work reaches 95.68%, which is 2.4% higher than CNN-PSO, 2.58% higher than CNN and 3.63% higher than SVM. The specificity of the proposed work is 97.67%, which is 2% higher than CNN-PSO, 3.13% higher than CNN and 4.25% higher than SVM. The precision of the proposed work is 98.67%, which is 6.16% higher than CNN-PSO, 6.46% higher than CNN and 7.56% higher than SVM. The recall of the proposed task is 98.69%, which is 7.17% higher than CNN-PSO, 7.34% higher than CNN, and 7.7% higher than SVM. The proposed work achieves an Fscore of 98.75%, which is 5.59% higher than CNN-PSO, 6.69% higher than CNN and 6.88% higher than SVM. The ROC AUC score for the lemon flavoured tea pack is 0.79; which is shown in the figure.6. Table 2 shows the Metrics evaluation for the food- can image.

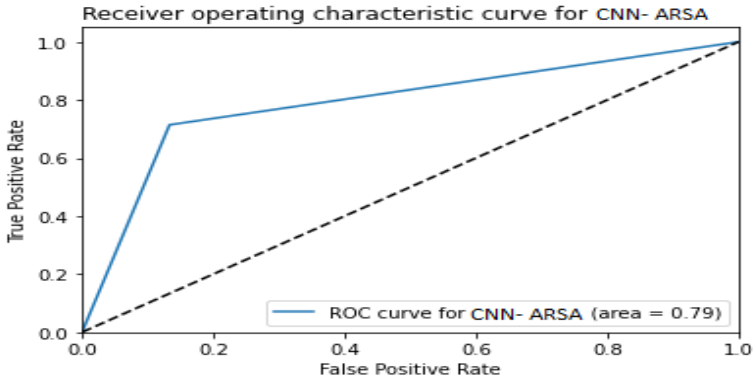


Figure 6: ROC curve CNN- ARSA (Lemon flavour tea package image)

(a)

(b)

(c)

Figure 7: (a) Uploaded input image- Food can, b) Image testing and c) Prediction

Table 2: Metrics evaluation for the food- can image

	Accuracy	**Sensitivity**	**Specificity**	**Precision**	**Recall**	**FScore**
CNN-ARSA	99.99	97.65	95.67	98.64	98.88	97.64
CNN-PSO	98.65	94.78	94.76	93. 15	90.52	93.96
CNN	97.15	93.82	93. 45	92.84	90.11	92.45
SVM	95.06	92.94	92.87	91.78	89.54	91.67

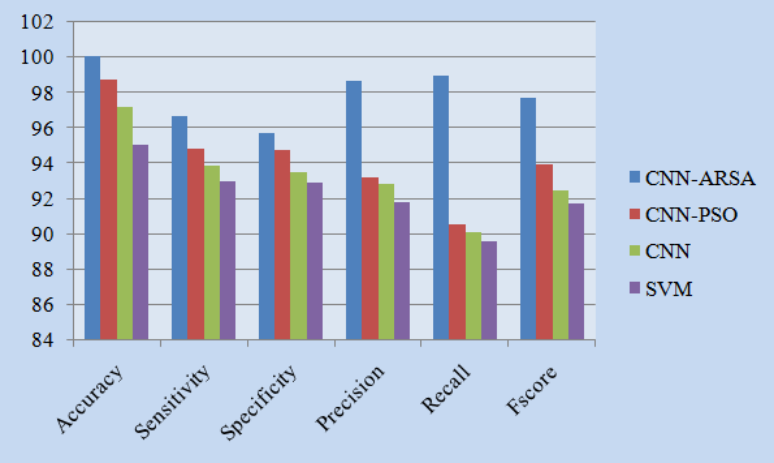


Figure 8: Metrics evaluation for the food- can image

Fig.7 shows the process of lemon tea flavoured tea packet from the dataset. From this figure there are two affected regions segmented. The first region has the probability of 99% defects and the second region have 99.8%. Fig.8 shows the measurement evaluation for the food can. From the figure, our proposed work achieves a high accuracy of 99.99%, which is 1.34% higher than CNN-PSO, 2.84% higher than CNN and 4.93% higher than SVM. The sensitivity of the proposed work reaches 97.65%, which is 2.87% higher than CNN-PSO, 3.83% higher than CNN and 4.71% higher than SVM. The specificity of the proposed work is 95.67%, which is 0.91% higher than CNN-PSO, 2.22% higher than CNN and 2.6% higher than SVM. The precision of the proposed work is 98.64%, which is 5.49% higher than CNN-PSO, 5.8% higher than CNN and 6.84% higher than SVM. The recall of the proposed task is 98.88%, which is 8.36% higher than CNN-PSO, 8.77% higher than CNN, and 9.34% higher than SVM. The proposed work achieves an Fscore of 97.64%, which is 3.68% higher than CNN-PSO, 5.19% higher than CNN and 5.97% higher than SVM. The ROC AUC score for the lemon flavoured tea pack is 0.76; which is shown in the figure.9. Metrics evaluation for the water bottle image shows in Table.3.

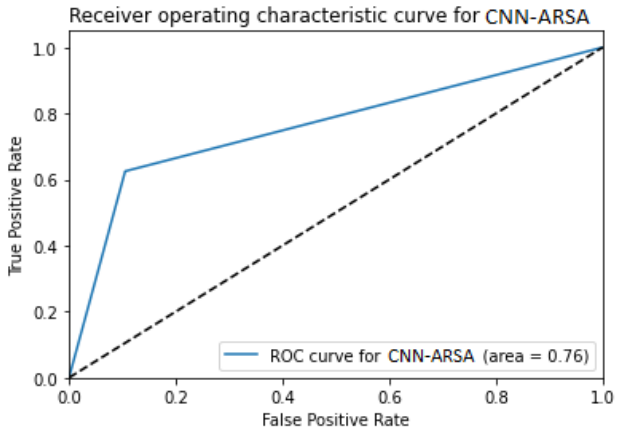
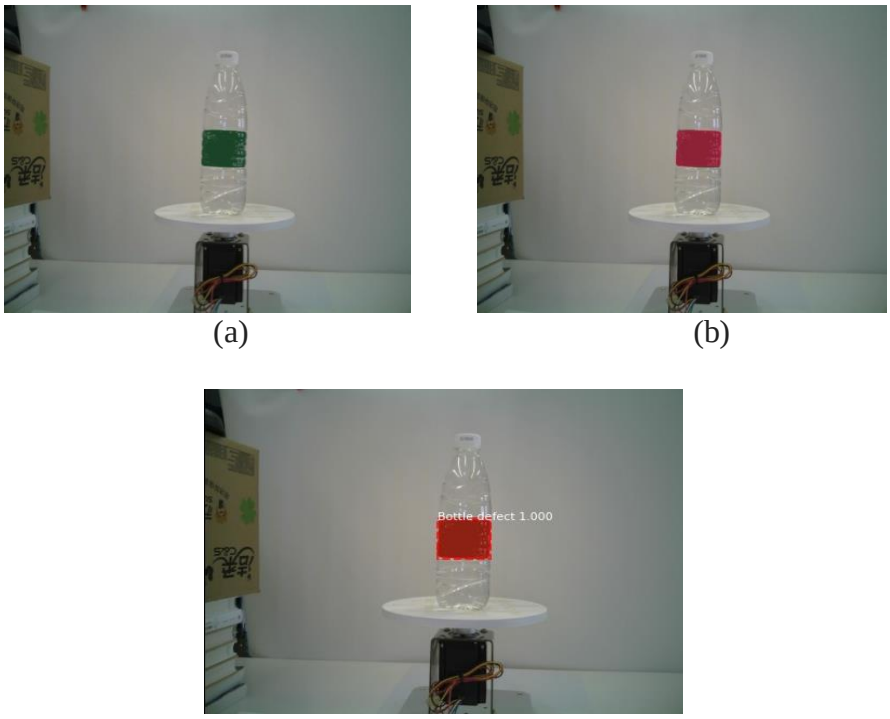


Figure 9: ROC curve for CNN-ARSA (food-can image)



(a) (b)

(c)

Figure 10: (a) Uploaded input image- Water bottle, b) Image testing and c) Prediction

Table.3: Metrics evaluation for the water bottle image

	Accuracy	Sensitivity	Specificity	Precision	Recall	FScore
CNN-ARSA	99.94	97.75	93.78	97.75	98.88	96.87
CNN-PSO	97.15	95.54	92.35	93. 5	96.76	95.83
CNN	96.32	93.63	91. 65	92.65	94.03	93.57
SVM	92.86	92.20	89.95	91.89	93.32	92.76

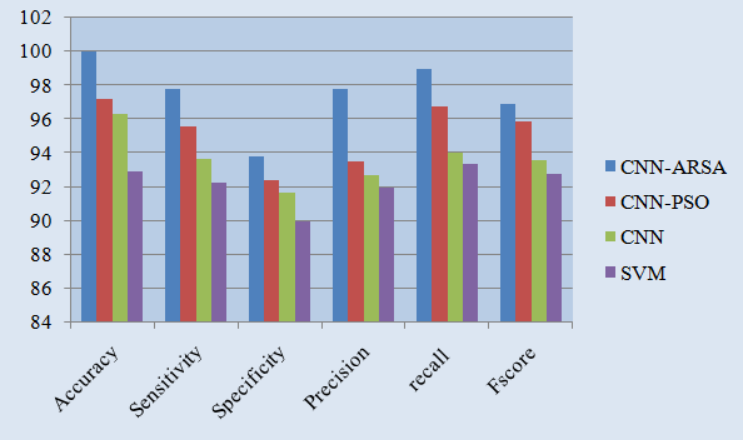


Figure 11: Metrics evaluation for the water bottle image

Fig.10 shows the process of lemon tea flavoured tea packet from the dataset. Our proposed algorithm detects 100% defects in the segmented region. Fig.11 shows the measurement evaluation for the food can. From the figure, our proposed work achieves a high accuracy of 99.94%, which is 2.94% higher than CNN-PSO, 3.62% higher than CNN and 7.08% higher than SVM. The sensitivity of the proposed work reaches 97.75%, which is 2.21% higher than CNN-PSO, 4.12% higher than CNN and 45.55% higher than SVM. The specificity of the proposed work is 93.78%, which is 1.43% higher than CNN-PSO, 2.13% higher than CNN and 3.83% higher than SVM. The precision of the proposed work is 97.75%, which is 4.25% higher than CNN-PSO, 5.1% higher than CNN and 5.86% higher than SVM. The recall of the proposed task is 98.88%, which is 2.12% higher than CNN-PSO, 4.85% higher than CNN, and 5.56% higher than SVM. The proposed work achieves an Fscore of 96.87%, which is 1.04% higher than CNN-PSO, 3.3% higher than CNN and 4.11% higher than SVM. The ROC AUC score for the lemon flavoured tea pack is 0.798; which is shown in the figure.12.

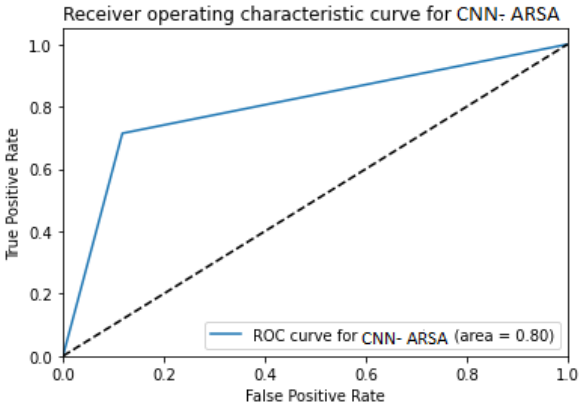


Figure 12: ROC curve for CNN-ARSA (water bottle image)

4.1 Comparative analysis

Table.4 compares the detection rate, accuracy, precision, recall, F-measure, and G-mean of CNN-ARSA with those of previous works. Wang et al. [8] proposed a deep convolution neural network (CNN) feature extraction system that automatically detects defects based on minimal prior knowledge. The accuracy of the proposed work improved 0.1% from this work. Hu, Liu and Cui [10] were proposed a CNN based intrusion detection system using Fruit fly optimization algorithm. The Recall, Fscore of the proposed work improved from 11.71% and 31.9% respectively. And the precision of the proposed work is reduced by 0.06%. Gao, et al., [11] was proposed a method for detecting gear face defects based on deep convolution generative adversarial networks (DCGANs) and lightweight convolution. The accuracy, recall and fscore of the proposed work improved from 2.23%, 0.41 and 0.35 respectively, the precision of the proposed work reduced by 0.06%. Laila, et al., [13] proposes the use of Convolution Neural Networks (CNN), a deep learning technology for image processing. From this work our proposed work improved the accuracy and defect detection rate by 2.4% and 0.96% respectively.

Table 4: Comparative analysis with existing works

	Accuracy	Sensitivity	Specificity	Precision	Recall	FScore	Defect detection rate
DCNN [8]	99.8%	-	-	-	-	-	-
CNN-FOA [10]	-	-	-	98.46%	87.1%	65.85%	-
DCGAN [11]	97.67%	-	-	98.46%	98.40 %	98.40%	-
CNN [13]	97.5%	-	-	-	-	-	97%
Proposed ARSA	99.9%	97.02%	95.70%	98.4%	98.81 %	97.75	97.96%

5. Conclusion

A deep learning algorithm is used in this paper to detect defects in food packages automatically. Images are first collected from the dataset. A median filter is used to remove noise from the images. Next, images that have been pre-processed are fed into a classifier. CNNs with fully connected layers are used for classification.An Adaptive Reptile Search Optimization algorithm is used to optimize weight parameters for the fully connected layer (ARSO). Using this method, defects are effectively detected and the affected area is correctly segmented. The performance of the presented technique is analyzed based on different prediction metrics and our presented technique achieved 99.9% best accuracy, 97.96% defect detection rate, 98.4% precision, 98.81% recall, 97.75% F-score, 97.02% sensitivity and 95.70% . These results show that our proposed model is useful for detecting food packaging deficiency.

Declarations

Funding

The authors declare that they have no funding

Competing Interests

On behalf of all authors, the corresponding author states that there is no competing Interests.

Author's contributions

All authors read and approved the final manuscript.

Availability of data and material

Data sharing is not applicable to this article because of proprietary nature.

Code Availability

Code sharing is not applicable to this article because of proprietary nature.

Reference

[1] Porter, M.E. and Kramer, M.R., 2019. *Creating shared value: How to reinvent capitalism—and unleash a wave of innovation and growth* (pp. 323-346). Springer Netherlands.

[2] Atiyah, Latif. "Product'S Quality And Its Impact On Customer Satisfactiona Field Study In Diwaniyah Dairy Factory." In *Proceedings of The International Management Conference I, Faculty of Management, Academy of Economic Studies, Bucharest, Romania*, vol. 10, no. 1, pp. 57-65. 2016.

[3] Li, Y., Huang, H., Xie, Q., Yao, L. and Chen, Q., 2018. Research on a surface defect detection algorithm based on MobileNet-SSD. *Applied Sciences*, *8*(9), p.1678.

[4] Santarelli, M.F., Giovannetti, G., Hartwig, V., Celi, S., Positano, V. and Landini, L., 2021. The core of medical imaging: State of the art and perspectives on the detectors. *Electronics*, *10*(14), p.1642.

[5] Wan, S. and Goudos, S., 2020. Faster R-CNN for multi-class fruit detection using a robotic vision system. *Computer Networks*, *168*, p.107036.

[6] Meng, Z., Zhang, M. and Wang, H., 2020. CNN with pose segmentation for suspicious object detection in MMW security images. *Sensors*, *20*(17), p.4974.

[7] Bhatti, F., Shah, M.A., Maple, C. and Islam, S.U., 2019. A novel internet of things-enabled accident detection and reporting system for smart city environments. *sensors*, *19*(9), p.2071.

[8] Wang, T., Chen, Y., Qiao, M. and Snoussi, H., 2018. A fast and robust convolutional neural network-based defect detection model in product quality control. *The International Journal of Advanced Manufacturing Technology*, *94*, pp.3465-3471.

[9] Chen, H., Xu, J., Xiao, G., Wu, Q. and Zhang, S., 2018. Fast auto-clean CNN model for online prediction of food materials. *Journal of Parallel and Distributed Computing*, *117*, pp.218-227..

[10] Hu, J., Liu, C. and Cui, Y., 2021. An improved CNN approach for network intrusion detection system. *Int. J. Netw. Secur*, *23*(4), pp.569-575.

[11] Gao, H., Zhang, Y., Lv, W., Yin, J., Qasim, T. and Wang, D., 2022. A deep convolutional generative adversarial networks-based method for defect detection in small sample industrial parts images. *Applied Sciences*, *12*(13), p.6569.

[12] Yangping, W., Shaowei, X., Zhengping, Z., Yue, S. and Zhenghai, Z., 2018. Real-time Defect Detection Method for Printed Images Based on Grayscale and Gradient Differences. *Journal of Engineering Science & Technology Review*, *11*(1).

[13] Azizah, L.M.R., Umayah, S.F., Riyadi, S., Damarjati, C. and Utama, N.A., 2017, November. Deep learning implementation using convolutional neural network in mangosteen surface defect detection. In *2017 7th IEEE international conference on control system, computing and engineering (ICCSCE)* (pp. 242-246). IEEE.

[14] Randall, R.B., 2017. A history of cepstrum analysis and its application to mechanical problems. *Mechanical Systems and Signal Processing*, *97*, pp.3-19.

[15] Abualigah, L., Abd Elaziz, M., Sumari, P., Geem, Z.W. and Gandomi, A.H., 2022. Reptile Search Algorithm (RSA): A nature-inspired meta-heuristic optimizer. *Expert Systems with Applications*, *191*, p.116158.