Journal of Intelligent Manufacturing (2023) 34:2989–3000
https://doi.org/10.1007/s10845-022-01980-7

Foreign objects detection using deep learning techniques for graphic card assembly line

R. J. Kuo[1] · **Faisal Fuad Nursyahid**[1]

Received: 29 June 2021 / Accepted: 4 June 2022 / Published online: 27 June 2022

Abstract
An assembly is a process in which operators and machines manufacture products from semi-finished components into finished goods. It is important to conduct quality control at the end of the assembly line and ensure that no foreign object is put on the conveyor. This study uses a case of foreign object detection in graphics card assembly line to create models which is capable of detecting and marking foreign objects using convolutional neural network (CNN) models. This study uses Inception Resnet v2 to conduct the foreign object classification and Attention Residual U-net++ for the foreign object segmentation. Both benchmark datasets and case study dataset are employed for model evaluation. The result shows that the proposed models can have more promising result than some existing models.

Keywords Foreign object detection · Attention · CNN · U-net

Introduction

An assembly line is a manufacturing process in which the production of products is divided into stages of inserting semi-finished components before finished goods are made. Due to the long and manual assembly processes, it is possible for foreign objects to be carried on the conveyor so that it can interfere with the production process and damage production machines. Therefore, quality control is important to ensure that no foreign object is put on the conveyor.

This phenomenon always occurs in graphic card manufacturing. Graphic card manufacturing is an assembly-oriented and labor-intensive industry which needs a lot of manpower and machines in the production line. The first stage of graphic card manufacturing is the mainboard production. At this stage, various types of machines are used to produce each important component in the graphic card mainboard. After the production of all components is complete, then those components are soldered together on the graphic card mainboard by operator. The finished graphic card mainboard is then inspected using X-rays detector to determine its quality. Then, the good quality mainboard which passes the quality check will be moved to the final assembly stage. Finally, the mainboard graphic card is assembled together with the fan, casing, and the other parts. The process of graphic card manufacturing are illustrated in Fig. 1.

In the graphic card manufacturing process, most of the assembly processes are carried out manually by labors. Furthermore, the finished graphic card must undergo a final testing process to ensure that the graphic card meets quality requirements and that no foreign object is brought to the final packaging materials. This manual process, however, is time consuming and erroneous. Thus, making this process automatically reduces processing time, energy used, and error. Therefore, this study aims to build models that can automate the inspection process, particularly the task of detecting foreign objects. One way to overcome this problem is by using computer vision method. Computer vision has been proven to be capable of performing foreign object detection tasks in manufacturing with satisfactory results (Rong et al., 2019). Thus, this study combines two models of CNN to identify and localize foreign objects by using a case of foreign object identification in the manufacturing of graphic cards. First, this study proposes Inception Resnet v2 with attention mechanism to foreign object classification task and modifies the activation function of the Inception Resnet v2 using the Mish activation function. In addition, this study also proposes a novel U-net architecture called attention residual U-net++ by

✉ R. J. Kuo
rjkuo@mail.ntust.edu.tw

[1] Department of Industrial Management, National Taiwan University of Science and Technology, No. 43, Section 4, Kee-Lung Road, Taipei 106, Taiwan

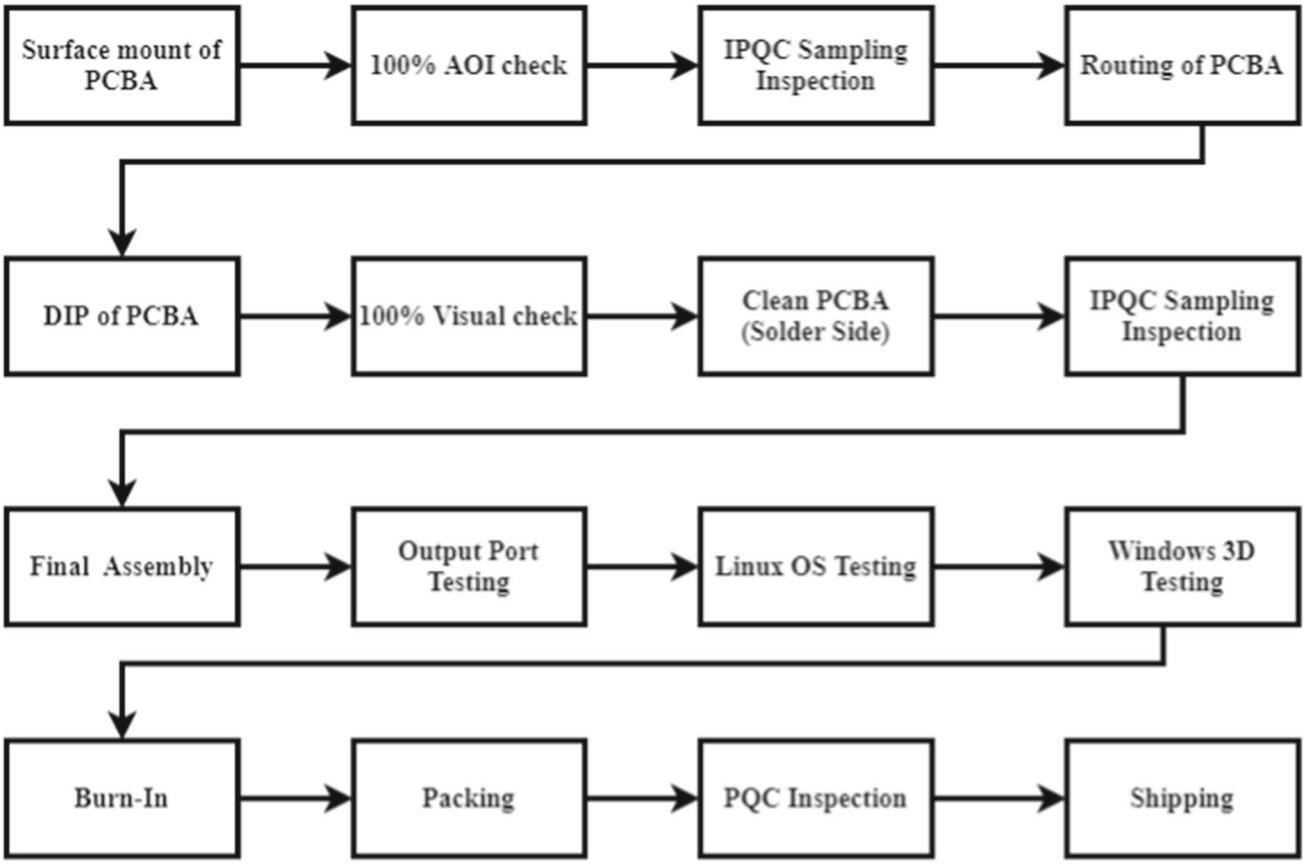

Fig. 1 Graphic card manufacturing process

combining the residual module with every layer in the contracting and expansive part and modifies the skip connection with attention module and dense module as well.

The remainder of this study is organized as follows. "Literature review" section presents some general backgrounds regarding the current study, while the proposed models are provided in "Methodology" section. "Model evaluation" and "Case study" sections illustrate the computational results for both model evaluation and case study. Finally, the concluding remarks are made in "Conclusions" section.

Literature review

Foreign objects detection

In a manufacturing process, foreign object identification is an attempt to identify objects that are useless and can interrupt the development process. This foreign object may come from debris or other objects that in the manufacturing process are unintentionally carried over. Foreign object detection is important to prevent foreign objects from being carried away in the production process so as not to affect the next process of the production.

Kwon et al. (2008) investigated foreign object detection on X-ray images of dry food. In this analysis, items accidentally mixed during the manufacture of dry food are described, such as: stainless steel, Teflon, aluminum, rubber, glass, ceramics, each of which has 6 different sizes. Before detecting the X-ray image that has been obtained, image pre-processing is performed, such as: removing the background and the mean zero gray value. After that, the positive response of zero image is calculated for feature extraction purpose by using Sobel mask within variance classified using the Gaussian. The detection result for high density materials such as stainless steel, aluminum, glass, and aluminum balls has a high detection rate of 98% without false positives. While for Teflon and rubber balls the detection rate is low due to the low gray intensity of the X-ray image compared to higher density objects.

The detection of foreign objects for food was studied by researchers at the National Meteorology Institute of Japan in 2018 using the principle of shifting phase and amplitude using a microwave (Kon et al., 2018). In this research, microwave is used due to its ability to detect non-metal foreign objects, whereas other techniques such as X-ray scanners, metal detectors and image recognition cannot detect. Aluminum oxide and zirconium dioxide of three different

sizes were used as foreign object samples in this test. In this study, they placed salad and hidden foreign objects in Microstrip Transmission Line (MSTL). In salads without foreign objects and those containing foreign products, changes in amplitude and phase shift are calculated. The ratio of amplitude and phase shift shifts will be proportional to the ratio of the actual and imaginary part of the material dielectric constant in MSTL.

In order to minimize the effect of hazards on patients, Moghadas and Rabbani (2010) attempted to assess the existence or absence of foreign objects in medical vials. They built a machine capable of taking 30 pictures in 1 s. They then develop a system to recognize foreign objects and categorize these foreign objects. Foreign objects that usually found on the medical vials are glass particles, rubber chips, calcium carbonate, chemical fiber, hair, and dust particles.

The developed device can perform tasks such as: recognize failures, distinguish bottle surface defects and foreign objects, distinguish bubbles from foreign objects, and classify foreign objects. To perform these tasks, this study first performs feature extraction by manually calculating: object's mean gray value, number of holes in object, object eccentricity, width of bounding box, height of bounding box, height to width ratio, object's area, ratio between object's area and bounding box area, object's circularity, object's perimeter straightness. After these features are obtained, support vector machines (SVM) and the multi-layer perceptron (MLP) are used to perform the tasks that have been designed. By using SVM they were able to get a misdetection rate and misclassification rate of 2.75% and 7.96%, respectively. Meanwhile, by using MLP with a configuration of 3 hidden layers with the number of neurons of 15, 8, and 4 respectively, and a SoftMax classifier they managed to get a misdetection rate and misclassification rate of 2.25% and 6.13% respectively.

Principal component analysis (PCA) and flat neural network (FNN) was used by Zhao et al. (2019) for feature extraction for a coal mine conveyor belt foreign object classifier. They collected video from the conveyor belt areas of the yanking group company's in Xinglong Zhuang coal mine and labeled it in three groups, including: no foreign entities, small foreign entities and large foreign entities. The algorithm's model training is fully calculated using feed forward calculation, so the complexity is low. They used two-layer filter PCA with 12 filter number for L1 and 16 filter number for L2 for feature extraction and then utilize these features in FNN. The FNN consist of feature mapping layers and enhancement layers. The feature mapping layer is a layer consisting of several nodes that take the input from the output PCA parameters. This parameter output will then be calculated using dot multiplication with a weight matrix and non-linearized with an activation function. The output from the feature map layer then becomes input for the enhancement layer with the same approach. In comparison with PCA net- SVM and Le-net5, their model was able to outperform them with 89.2% accuracy compared with 81.6% and 87.8% respectively. Also, their model has 8.37% and 18.8% faster training times when compared to PCANet-SVM and Le-Net 5 respectively.

In 2019, Rong et al. (2019) proposed a model which was a modification of U-net (Ronneberger et al., 2015) for detecting foreign object in walnut. They use modified U-net model for segmented and a CNN model for classified the foreign object. They used 7 convolutional layers in the down-sampling path. In each of the convolution layer followed with batch normalization and ReLu layer. The up-sampling path used mirrored architecture of the down-sampling path. At the end of the model, they use sigmoid layer to reconstruct binary image of the foreign object. The model's result can segment 99.5% of the foreign object. The classification task using 9 convolutional layers, with each layer followed by ReLu, max-pooling and dropout layer. Using SoftMax classifier at the end of the model, they can obtain 95% accuracy.

CNN architectures

The CNN is a subset of neural networks consisting of many layers that are usually intended for pattern recognition and image classification (Dieleman et al., 2015). In general, CNN consists of three main layer types, namely: convolutional layer, sub-sampling layer (pooling layer), and fully connected layer (Wang et al., 2019). So far, there have been different CNN models proposed including ZF net (Zeiler & Fergus, 2014), VGG 19 (Simonyan & Zisserman, 2014), and Google net.

All of these techniques combined into a model named Inception module. Inception module is a module that combine 1×1, 3×3, and 5×5 convolutional layers along with global average pooling arranged in parallel configuration to handle objects at different scales. There are two Inception modules that are introduced. Since it was first introduced, Google net (Inception v1) has been developed several times into other models such as: Inception v2, Inception v3, Inception Resnet v1, and Inception Resnet v2 (Szegedy et al., 2015, 2016, 2017). Basically, the above mentioned approaches still need some improvements in order to achieve better classification accuracy. In addition, more advanced CNN should be considered to apply to foreign object detection.

U-net architectures

U-net

U-net is an end-to-end fully CNN proposed and developed by Ronneberger et al. (2015) for segmentation tasks in biomedical such as segmenting neural structures in electron microscope stacks. Since its appearance, U-net is very well known and has begun to be applied in other fields includ-

ing industry. In the industrial sector, U-net can be used for the segmentation of defective textured-surfaces (Mittal et al., 2019; Sarakon et al., 2019) and foreign objects segmentation (Rong et al., 2019).

U-net consists of two main parts, namely the contracting part (left) and the expansive path (right) as shown in Fig. 4. The contracting part uses two 3×3 convolution layers, each followed by ReLu and 2×2 max Pooling (2 strides). Meanwhile, each layer in the expansive part consists of 2×2 up-sampling, concatenation with a cropped feature map from the contracting path, and two 3×3 convolution followed by ReLu. In the last layer, U-net uses 1×1 convolution with sigmoid activation function to reconstruct the feature map. U-net is managed to get an Intersection over Union (IoU) value of 77.5% compared with 46% IoU for second-best algorithm for DIC-HeLa dataset. Meanwhile, for the PhC-U737 dataset, U-net is also managed to beat the second-best algorithm with IoU value of 92.03% compared to 83%.

R2U-net

In 2018, Alom et al. (2018) proposed two new models of U-net called recurrent U-net (RU-net) and recurrent residual U-net (R2U-net). In recurrent U-net model, they combine U-net with three-fold recurrent mechanism from recurrent CNN to ensure better feature representation for segmentation tasks. Using three-fold recurrent convolution for every layer in U-net makes their model more complex, so they combined the recurrent U-net model with residual mechanism to help train deep architecture by avoiding vanishing gradient problems. They tested their model on DRIVE, STARE, and CHASE_DB1 datasets. On every datasets R2U-net successfully achieve best F1 score with 81.71%, 84.75% and 79.28% respectively.

Attention U-net

Attention U-net is a model that was proposed in a paper by Oktay et al. (2018). Attention U-net uses an additive soft attention that can be trained using back propagation without need for Monte Carlo sampling. The aim of applying attention mechanism in U-net is for highlighting salient feature maps that are passed to the skip connection.

Attention mechanisms acted as a gate to reduce disambiguates and noisy response in skip connection feature map right before concatenation operation to merge only with relevant activation from next lowest layer. In attention mechanism, feature map from next lowest layer of network (g) convoluted using 1×1 convolution layer with dimension of D_g and 1. Meanwhile, feature maps from the skip connection (x) get convoluted with 1×1 convolutional layer with dimension of D_x with number of strides of ratio between $H_g x W_g$ and $H_x x W_x$ to match the dimension of g. These two features map then concatenate together and activate using ReLu activation function. The result of this activation convoluted using $1 \times 1 \times 1$ convolutional layer to reduce the dimensionality of the feature map. Finally, sigmoid is used as a sampler or logistic probabilistic function. Output from this sigmoid function (α) acted as a resample for the feature map from skip connection (x), as the aligning feature map will become larger while the unaligned feature map becomes smaller. Using attention U-net for TCIA Pancreas-CT dataset can obtain 1.1% higher IoU compared with U-net model.

U-net++

Zhou et al. (2018) proposed a new U-net model called U-net ++. The difference between this U-net++ and original U-net is in the skip connection. U-net++ uses a new skip connection using dense module principal and a new skip pathway compared to direct concatenation skip connection in U-net model. The new skip pathway uses dense convolutional blocks whose number of convolutional blocks depends on the pyramid level. Every convolutional block in the skip pathway gets input from the same level of contracting part and next lower convolution block (after up-sampling). All of the convolutional blocks in skip pathway sum together with the skip connection from contracting part in the same level. They tested this model on 4 datasets including cell nuclei, colon polyp, liver, and lung nodule. They obtain 1.86%, 3,37%, 6.28%, and 5.74% higher IoU respectively compared to the original U-net on each dataset. However, U-net++ still needs improvement in order to provide better performance.

Methodology

There are two main tasks for the proposed method including classification task and segmentation task. The detailed discussion for each task is illustrated as follows.

Classification task

For the classification task, Inception Resnet v2, which is a variant of the Inception model combined with the residual mechanism found in Resnet, is employed. Inception Resnet v2 is a type of CNN model consisting of a stem block, three Inception Resnet blocks and two reduction blocks which each block located between Inception Resnet blocks.

At the first block, stem block takes an image as input then extracted into finer feature map for Inception Resnet A input. Inception Resnet A processed its input feature map using combination of 1×1 and 3×3 filter size convolution configuration then produces feature map for the next module. This process occurs repeatedly and follows the configuration pattern in Figs. 2 and 3. The final feature map from Incep-

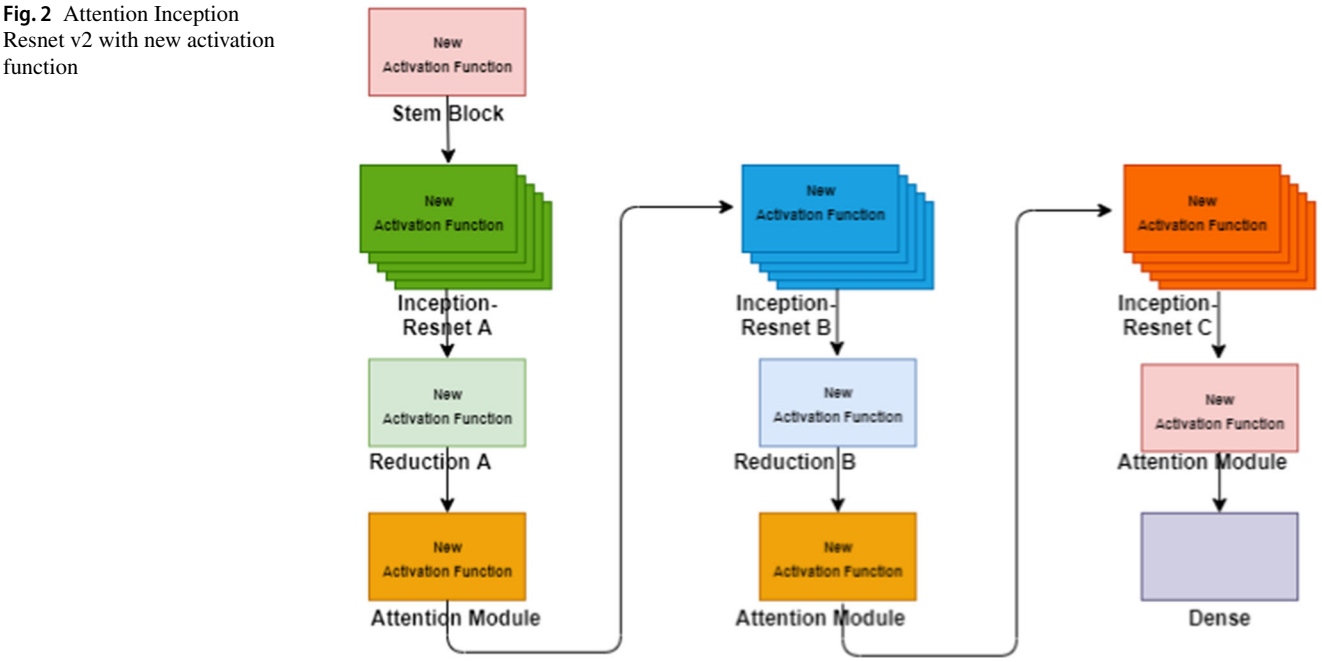

Fig. 2 Attention Inception Resnet v2 with new activation function

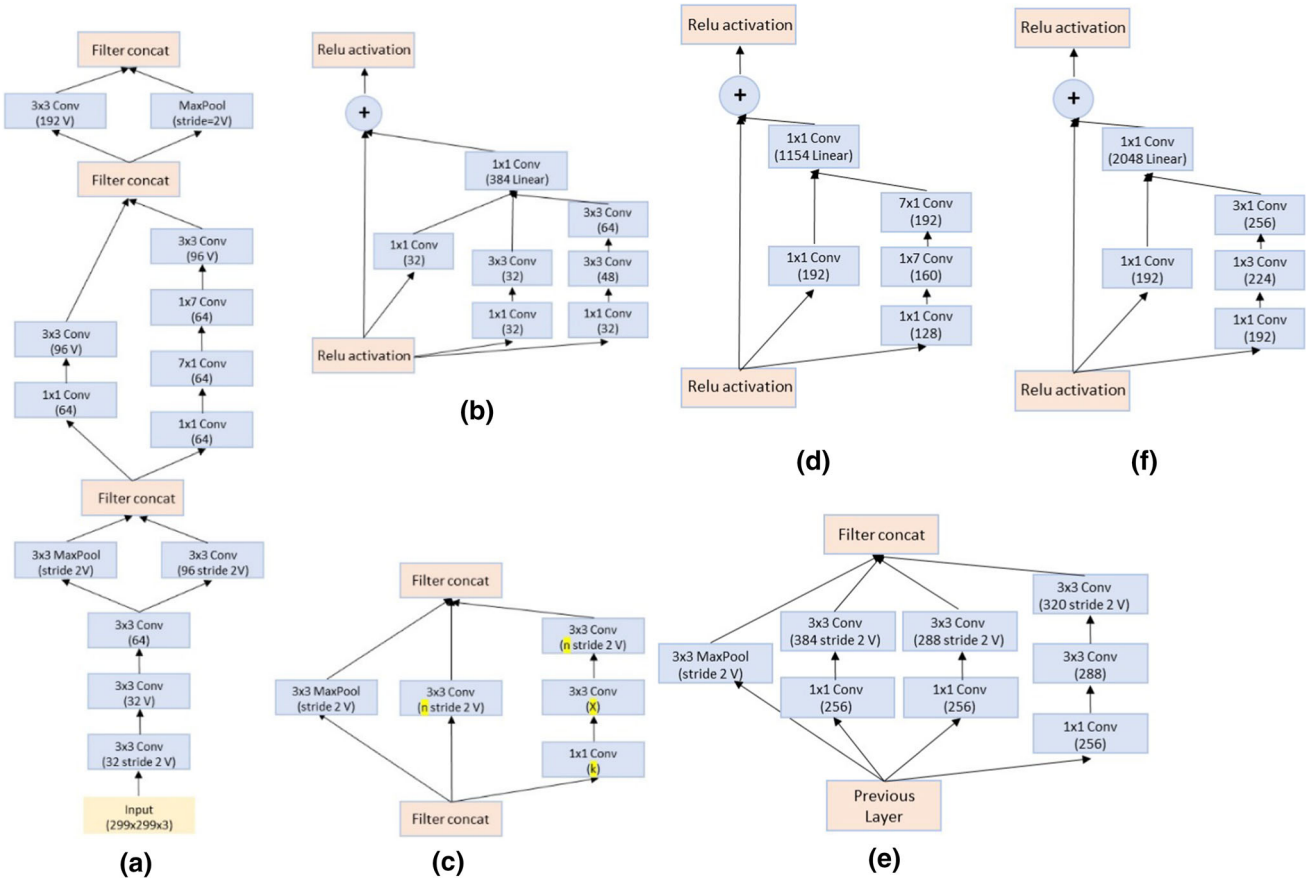

Fig. 3 Inception Resnet v2 configuration: **a** stem block, **b** Inception Resnet-A, **c** reduction A, **d** Inception Resnet-B, **e** reduction B, and **f** Inception Resnet-C (Szegedy et al., 2017)

tion Resnet C is then used in the SoftMax fully connected classifier to perform classification.

This model is further combined with the attention module mechanism, namely: Squeeze and excited network (SE net) and convolutional block attention module (CBAM). SE net has been proven to be able to post accuracy Inception Resnet v2 around 2.79% at top-1 error rate (Hu et al., 2018) by using dynamic channel wise calibration. Meanwhile, CBAM is claimed to be able to compete with the accuracy of the SE net with an improvement of 2.07% by combining spatial attention and channel attention.

The SE net consists of two mechanisms, namely squeeze and excited mechanisms. The squeeze mechanism is used to extract spatial information into channel descriptors. This mechanism is carried out by using the global average pooling $z \in R^C$ along the HxW spatial information. The following is the global average pooling formula on the c^{th} element of z elements for an u_c feature map:

$$z_c = F_{sq}(u_c) = \frac{1}{HxW}\sum_{i=1}^{H}\sum_{j=1}^{W} u_c(i, j) \tag{1}$$

where z_c is an output feature map from squeeze mechanism, F_{sq} is global average pooling, u_c is the previous feature map, and (i, j) is spatial information through HxW.

The excited mechanism uses the z vector as input to produce a vector s which can be used as a scale to re-calibrate the model we are using two fully connected networks of ReLu and sigmoid that have a bottleneck which can be represented using the size C/r, which can be calculated as follows:

$$s = F_{ex}(z, W) = \sigma(W_2 ReLu(W_1 z)) \tag{2}$$

where $W_1 \in R^{\frac{C}{r}xC}$ and $W_2 \in R^{Cx\frac{C}{r}}$.

CBAM uses average pooling and maximum pooling of a feature map to produce two different spatial descriptors, F_c^{avg} and F_c^{max}. These two spatial descriptors feed into a shared MLP layer to produce the $M_c \in R^{Cxlxl}$ feature map. To reduce the number of parameters a bottleneck $R^{\frac{C}{r}xlxl}$ is used, where r is the reduction ratio. CBAM consists of two modules, namely the spatial module and channel module. In the channel module, a shared ReLu layer MLP is used. It uses inputs from F^{avg} and F^{max}. The sum of the MLP output is then fed to a sigmoid activation function. The following is the complete channel wise module formula:

$$\begin{aligned} M_c(F) &= \sigma(MLP(AvgPool(F) + MLP(MaxPool(F))) \\ &= \sigma(W_1\left(W_0 F_c^{avg}\right) + W_1\left(W_0 F_c^{max}\right), \end{aligned} \tag{3}$$

where σ is sigmoid function, $W_0 \in R^{\frac{C}{r}xc}$ and $W_1 \in R^{Cx\frac{C}{r}}$.

For spatial attention, the convolutional layer is used to produce 2D spatial attention feature maps. The spatial attention module formula is as follows:

$$\begin{aligned} M_s(F) &= \sigma\left(f^{7\times7}([AvgPool(F); MaxPool(F)])\right) \\ &= \sigma\left(f^{7\times7}\left(\left[F_s^{avg}; F_s^{max}\right]\right)\right) \end{aligned} \tag{4}$$

where $f^{7\times7}$ is a 7×7 convolutional layer, F_s^{avg} is an average pooling layer, and F_s^{max} is a maximum pooling layer.

In addition, the activation function of Inception Resnet v2 is modified as well. The original Inception Resnet v2 use ReLu activation function. However, ReLu has a well-known problem for it dying out gradient at learning a large negative bias or negative error feedback from its backpropagation. The study from Lu et al. (2019) proved that neural network (NN) with deeper and wider network has higher probability of ReLu dying out gradient problem. They mention that neural network with composition of 10 ReLu layers and 3 width has 60% probability of ReLu dying out gradient problem. They propose a new weigh initialization named random asymmetric initialization (RAI) to resolve this problem. In this study we will try another approach by using different activation functions such as swish and mish to overcome this problem. Swish is an activation function that is capable of outperforming ReLu of 0.6–0.9% (Ramachandran et al., 2017) and is able to overcome the ReLu dying problem, which happens when the output model has a value of 0.

The swish activation function is shown as follows:

$$y = \frac{x}{\left(1 + e^{-x}\right)} \tag{5}$$

On the other hand, Mish activation function can increase the accuracy of Swish and ReLu by 0.494% and 1.671%, respectively (Misra, 2019). Mish's unbounded above, bounded below, smooth and non-monotonic nature is claimed to be the reason for this good performance.

The Mish activation function is represented as follows:

$$y = x * \tanh\left(\ln\left(1 + e^{x}\right)\right) \tag{6}$$

In this study, a combination of the two attention mechanisms and the two activation functions mentioned above will be implemented to determine which combination has the better accuracy for the current application.

Segmentation task

This study also modifies U-net++ architecture with addition of attention module and residual module. The skip connection of U-net + + , which are combination of pyramid convolution and dense module are then combined with the

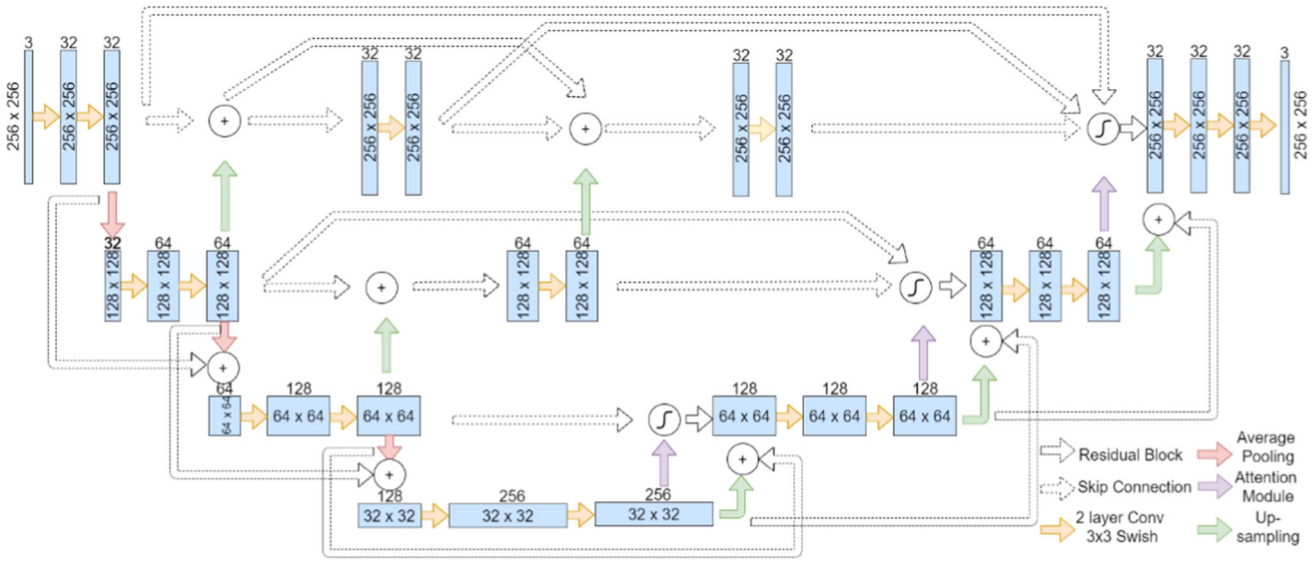

Fig. 4 Attention U-net++ with residual block

skip connection, as seen in Fig. 4. The purpose of the attention gate in U-net++ skip connection is to gate/filter low level spatial features from skip connection between contracting and expansive part. It is expected that the resulting feature map from the new skip connection will reduce the semantic gap between the contracting and expansive part and easier to be optimized.

On the other hand, the residual block (consisting of input and output from previous layer) is used as input for the next layer in each of layer in contracting and expansive part. Residual block is used to solve vanishing gradient problems on networks that have a large number of layers. This study does not use all the previous layer feature maps in the residual block. Instead, only part of it is used by pre-defining weight or scaler (0.1–0.5) for the previous layer feature map. The reason is that we want our model to learn only from important features from the previous layer. Feature map from the upper layer contains a high features map, but still there are many features which are unimportant. That is why they are valued less important than features from the lower layer. This mechanism also can be found in the residual module in Inception Resnet v2 model in Keras applications.

Model evaluation

This section will employ some benchmark datasets to evaluate the proposed models' performance for classification and segmentation, respectively. All the proposed algorithms were coded by Python 3.7.9 programming language and run on a PC with an Intel Core i5-9600 processor and 16 GB RAM.

Classification task

For model evaluation, two benchmark datasets, Cifar-10 and cifar-100 datasets, are employed. The Cifar-10 and cifar-100 datasets are created by the computer science department of the University of Toronto, Canada. Cifar-10 consists of 10 classes of images. It consists of 60,000 32×32-pixel color images which are divided into 50,000 training images and 10,000 testing images. While cifar-100, consist of 100 classes of images with the same size and number of samples as cifar-10. The state-of-the-art for cifar-10 is ViT-H/14 which achieved 99.5% correct accuracy and cifar-100 is BiT-L (Big Transfer- Large) model which achieved 93.5% accuracy.

For Cifar-10 dataset, according to the results as shown in Table 1, it can be concluded that Inception Resnet v2 has the best testing accuracy of 79.22% followed by ZF net with 78.99% accuracy and VGG19 with 75.74% accuracy. This means that Inception Resnet v2 can extract and select meaningful feature map from the image used for learning and optimize the model to help the classifier (SoftMax classifier) learn and produce better testing accuracy.

Further, the effect of adding attention mechanism and changing activation function is examined. According performance comparison in Table 2, it shows that adding squeeze and excited network (SE net) and changing activation function could help the model increase its testing accuracy by around 4. While using CBAM and changing its activation function does not help increase the testing accuracy. This reveals that using channel attention from SE net is preferable rather than using channel and spatial attention from CBAM.

It can be concluded that Mish function is slightly superior to Swish function as testing accuracy of SE Inception Resnet v2 with Mish function is superior when compared to

Table 1 Model comparison of ZF net, VGG19, and Inception Resnet v2 for Cifar-10 dataset

Model	Training accuracy (%)	Training loss	Testing accuracy (%)	Testing loss	Computational time (s)
ZF net	98.08	0.06	78.99	1.25	420
VGG19	98.43	0.05	75.74	0.75	3706
Inception Resnet v2	98.58	0.04	79.22	0.79	7864

Table 2 Performance comparison of different activation functions and attention module for Inception Resnet v2

Model	Training accuracy (%)	Training loss	Testing accuracy (%)	Testing loss	Computational time (s)
Inception Resnet v2	98.58	0.04	79.22	0.79	7864
SE Inception Resnet v2 Swish	99.05	0.03	83.19	0.87	8974
SE Inception Resnet v2 Mish	99.08	0.03	83.81	0.86	9270
CBAM Inception Resnet v2 Swish	98.49	0.04	76.09	1.38	9250
CBAM Inception Resnet v2 Mish	98.78	0.04	76.6	1.30	9621

SE Inception Resnet v2 with Swish by 0.62%. According to (Misra, 2019), Mish function is superior to Swish function because it has a self-regularizing conditioner which helps make optimizations of deep networks much easier.

Regarding Cifar-100 dataset, following the similar result from cifar-10 dataset, from the experimental results as shown in Table 3, Inception Resnet v2 has the best testing accuracy, 49.4%, followed by ZF net's 48.8% and VGG19's 24.57% although it also has the longer computational time, 8,611 s. Thus, it is concluded that from Cifar-10 and Cifar-100 performance comparison of three models, Inception Resnet v2 can outperform ZF net and VGG19 in terms of testing accuracy and stability.

For the effect of adding attention mechanism and changing activation function, from the Table 4, it showed that for the addition of the attention module to the Inception Resnet v2, the testing accuracy of the SE Net is always superior compared to CBAM. As for the effect of changing activation function in Inception Resnet v2, Mish and Swish activation functions always outperformed ReLu. This can happen because the problem of ReLu, dying out, can be handled well by Swish and Mish functions. Both activation functions allow a slight change if the activation function value is below zero. Furthermore, the performance of Mish function is slightly better than that of Swish function because of the nature of Mish activation function in which it has a self-regularized characteristic in the first derivative of this activation function which is able to help training in deep networks.

Segmentation task

For segmentation task, similarly, there are two datasets used including Carvana and Oxford Pet IIIT datasets. Carvana dataset is an open competition dataset from Kaggle website that is held by a car company named Carvana. In this dataset, the sample image is already separated into the training folder and the testing folder. In our study, we only use the training folder images for efficiency. We use 5088 images from the training folder along with its ground truth images from the train mask folder. We did not perform pre-processing for this dataset, except modify the train mask file extension from gif to png. Regarding Oxford Pet IIIT dataset, it is a dataset created by a team from Visual Geometry Group (VGG). This dataset contains 7349 images of 37 different classes of cat breed and dog breed. Each class consists of an average of 200 images that have large variations in scale, pose, and lighting.

For Carvana dataset, the summarized performance result of each U-net model can be seen in Table 5. From the table below, it reveals that the proposed model, attention residual U-net++ , has the highest mean IoU value compared to other methods. The second best model for this dataset is U-net++ , with a mean IoU of 93.97% followed by attention U-net, residual U-net and U-net. In addition, the proposed our model is also the most stable one as the variance of 0.13% compared to the U-net++ whose variance is 0.27%.

Regarding Oxford PET IIIT dataset, from Table 6, it can be seen that the performance of our proposed model, attention residual U-net++ , is better than those of other models except for Attention U-net for this dataset. The attention U-net model has the same mean IoU as attention residual U-net++ . It seems that attention residual U-net++ is more stable than attention U-net because it has lower variance value. Moreover, in terms of computational time, the proposed models require longer computation time than attention U-net. Further comparison in statistical hypothesis testing will be conducted in the next section.

Table 3 Model Comparison of ZF net, VGG19, and Inception Resnet v2 for Cifar-100 dataset

Model	Training accuracy (%)	Training loss	Testing accuracy (%)	Testing loss	Computational time (s)
ZF net	94.75	0.16	48.88	3.20	689
VGG19	55.50	1.56	24.57	4.18	2368
Inception Resnet v2	96.12	0.12	49.4	8.33	8611

Table 4 Performance comparison of different activation functions and attention module for Inception Resnet v2

Model	Training accuracy (%)	Training loss	Testing accuracy (%)	Testing loss	Computational time (s)
Inception Resnet v2	96.12	0.12	49.4	8.33	8611
SE Inception Resnet v2 Swish	96.99	0.09	53.67	2.67	9203
SE Inception Resnet v2 Mish	97.01	0.09	54.46	2.89	9390
CBAM Inception Resnet v2 Swish	96.06	0.12	47.3	3.93	9511
CBAM Inception Resnet v2 Mish	96.30	0.11	47.64	3.47	9734

Table 5 Performance comparison of various U-net models for Carvana dataset

Model	Training loss	Testing loss	IoU (%)	Computational time (s)
U-net	0.01	0.016	93.22 ± 0.27	726
Residual U-net	0.009	0.016	93.23 ± 0.26	799
Attention U-net	0.0067	0.007	93.69 ± 1.14	849
U-net++	0.007	0.007	93.97 ± 0.27	1561
Attention Residual U-net++	0.0069	0.007	94.13 ± 0.13	2841

Case study

The dataset used for the real-world application was collected in the Intelligent Operation Center laboratory at National Taiwan University of Science and Technology as shown in Fig. 5. The lab is financially supported by the case company. Basically, if the system detects something wrong, then the conveyor will be stopped and the alarm will be triggered. This dataset consists of 7 classes, which are: graphic card, graphic card with ID card, graphic card with glasses, graphic card with mobile phone, graphic card with mask (due to COVID-19), graphic card with pen, and graphic card with screwdriver. These pictures are taken using a mobile phone camera by recording video of the objects from various angle, such as: from above, right side, left side, upside, and down side. Then, from every frame in the video will be extracted into an image. For classification task, the pre-processing techniques used are compressing and resizing to reduce the computational memory and reduce computational time.

The raw image for segmentation task uses the same raw images from classification task. Every image consists of six classes. However, they are pre-processed using different pre-processing techniques. The pre-processed images used are the images after removing unwanted objects, compressing and resizing, implementing K-means algorithm segmentation, and pixel-thresholding. Thus, one example of the final image produced can refer to Fig. 6. For classification, the number of training samples is 6844 and the number of testing samples is 2281. For segmentation, the number of training samples is 3757 and the number of testing samples is 900.

Table 6 Performance comparison of various U-net models for Oxford PET IIIT dataset

Model	Training loss	Testing loss	IoU (%)	Computational time (s)
U-net	0.21	0.37	86.38 ± 0.44	977
Residual U-net	0.36	0.41	86.24 ± 0.48	1171
Attention U-Net	0.29	0.31	87.83 ± 0.65	1590
U-net++	0.34	0.36	87.64 ± 0.48	3183
Attention Residual U-net++	0.28	0.32	87.82 ± 0.53	5790

Table 7 Model comparison of ZF net, VGG19, and Inception Resnet v2 for Cifar-10 dataset

Model	Training accuracy (%)	Training loss	Testing accuracy (%)	Testing loss	Computational time (s)
ZF net	99.93	0.003	99.74	0.012	382
VGG19	99.70	0.002	96.93	0.095	418
Inception Resnet v2	100	0.002	99.78	0.012	1112

Fig. 5 Production line for assembling GPU card at National Taiwan University of Science and Technology

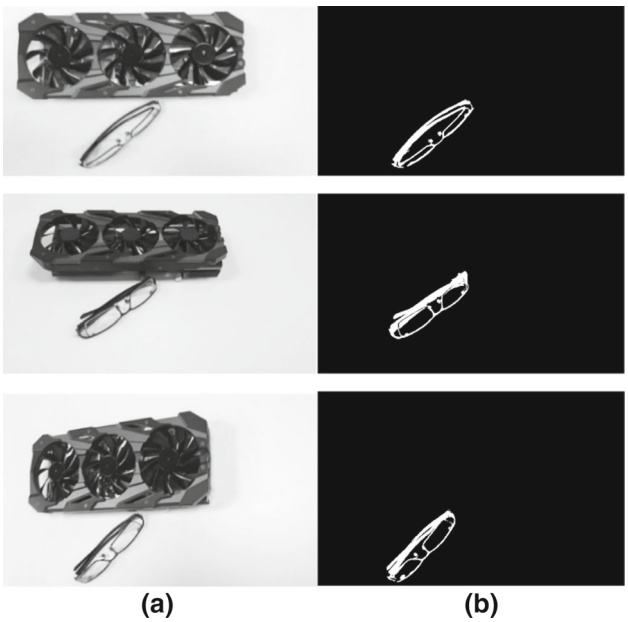

Fig. 6 Images for segmentation: **a** raw image and **b** ground truth

Classification task

According to the performance comparison of the three model in Table 7, Inception Resnet v2 has the highest testing accuracy with 99.78%. Inception Resnet v2 is developed by considering the effects of dying deep networks, while ZF net and VGG19 Net are not. In addition, Inception Resnet is built using multiple size filters that operate parallelly at the same level. Thus, the model can capture various sizes of salient objects of the images, while the ZF net and VGG19 are developed using one filter size at the same level. Even so, the impressive Inception Resnet v2 performance is followed by a longer computational time when compared to VGG19 and ZF net.

The experimental results are presented in Table 8 for changing activation function. The results reveal that Swish and Mish activation functions have superior performances compared to ReLu activation functions. Both are able to improve the testing accuracy performance of Inception Resnet v2 as well as reduce the testing errors. As aforementioned, the reason is that ReLu has the dying problem and Swish and Mish are able to solve this problem well. Nonetheless, Swish and Mish have very close performances, because both have very similar curves. They all have characteristics of smooth, unbounded above and allow slight allowance at negative values. However, Table 5.4 shows that Mish is slightly better than Swish. Mish has 0.13% higher testing accuracy when compared to Swish. In addition, both the training errors of Mish are lower than those of Swish. According to (Misra, 2019), Mish has characteristic of self-regularizing which help training of deep network easier as the first derivation of Mish can be write as:

$$f(x) = x\tanh((x)) = x\tanh\left(\ln\left(1+e^{x}\right)\right)$$

$$f'(x) = sech^{2}\left(\ln\left(1+e^{x}\right)\right)xsigmoid(x) + \frac{f(x)}{x}$$

$$f'(x) = \Delta xswish(x) + \frac{f(x)}{x} \tag{7}$$

From the Eq. 13 above, Δx is acts the self-regularizing characteristic from first derivative of Mish. This implies that Mish is easier to optimize when compared to Swish.

Regarding the effect of adding attention mechanism, from Table 9, we can conclude that adding attention module to Inception Resnet v2 does not help to increase the testing accuracy. As the SE Inception Resnet v2 has the same testing accuracy as Inception Resnet v2, even CBAM Inception Resnet v2 has an accuracy that is slightly below Inception

Table 8 Performance comparison of various activation functions for Inception Resnet v2

Model	Training accuracy (%)	Training loss	Testing accuracy (%)	Testing loss	Computational time (s)
Inception Resnet v2 (Relu)	100	0.002	99.78	0.012	1112
Inception Resnet v2 (Swish)	99.90	0.004	99.82	0.004	1230
Inception Resnet v2 (Mish)	99.94	0.002	99.95	0.002	1257

Table 9 Performance comparison of attention mechanism for Inception Resnet v2

Model	Training accuracy (%)	Training loss	Testing accuracy (%)	Testing loss	Computational time (s)
Inception Resnet v2 (Mish)	99.94	0.002	99.95	0.002	1257
SE-Inception Resnet v2	100	0.0001	99.95	0.002	1268
CBAM-Inception Resnet v2	100	0.0007	99.91	0.002	1351

Resnet v2. This could be because the feature selection of Inception Resnet v2 is already very good, so weighting the feature map using the attention module is not very influential thus does not help to increase the testing accuracy.

Segmentation task

From Table 10, it reveals that the performance of the proposed model, namely U-net++ attention residual, is the best one in performing segmentation tasks for foreign objects with a mean Iou (mIoU) of 77.89%, although it needs longer computation time. In the proposed model, it applies two modifications to the original U-net model. The first one is to add a residual module to each contracting and expansion layer. This improvement is made to prevent the model from having deep network dying problems as the feature map in the proposed model will continue to be updated so it can avoid the vanishing gradient problem. The second improvement is to combine two skip connection schemes from U-net++ and attention module to reduce segmentation gap between the contracting part and the expansion part, which also help to reduce the training error.

Conclusions

This study has performed a comparative study of the performance of CNN Models and U-Net Models. In the CNN model comparison, the performances of 3 CNN Models, namely ZF net, VGG19, and Inception Resnet v2, FOR Cifar-10 and Cifar-100 datasets were examined. In addition, this study also conducted comparisons of adding attention modules and modifying activation functions in Inception Resnet v2. The experimental results obtained from the comparison showed that Inception Resnet v2 performs well on both Cifar-10 and Cifar-100 datasets compared with ZF net and VGG19. The hybrid of SE Net can improve the performance of Inception Resnet v2 compared with CBAM. The adoption of the Mish activation function on the SE Net can improve the performance of Inception Resnet v2 and beat the performance of ReLu and Swish Activation Functions. In addition, SE-Inception Resnet v2 with Mish activation is the best model for the case study.

Regarding the 5 existing U-net models including U-net, Residual U-net, Attention U-net, U-net++ and the proposed U-net model, Attention Residual U-net++, for Oxford Pet IIIT dataset, Carvana dataset, and case study dataset, Attention Residual U-net++ is the best model for Carvana and case study datasets, while Attention U-net is the best model for the Oxford Pet IIIT dataset. The hybrid of Attention Module, Residual Module, and Dense Module with U-net++ is proven to be able to improve the performance of U-net++ .

According to the experiences during data collection, data pre-processing and model building, there are still some future research directions. For instance, it is possible to make other comparisons of CNN models by adding Attention Module and changing activation function or apply random asymmetric initializer method for weight initializer to addressing ReLu dying out gradient problem. In addition, applying the data augmentation on the U-net model to improve its performance can be considered.

Table 10 Performance comparison of various U-net models for case study data

Model	Training loss (%)	Testing loss (%)	IoU (%)	Computational time (s)
U-net	0.83	0.98	75.88 ± 1.75	540
Residual U-net	0.6	0.87	77.4 ± 1.13	660
Attention U-net	0.48	0.48	77.34 ± 0.76	945
U-net++	0.53	0.44	77.39 ± 0.6	2024
Attention Residual U-net++	0.4	0.47	77.89 ± 0.59	2595

Declarations

Conflict of interest The authors declare that they have no conflict of interest that could have appeared to influence the work reported in this paper.

References

Alom, M. Z., Hasan, M., Yakopcic, C., Taha, T. M. & Asari, V. K. (2018). Recurrent residual convolutional neural network based on u-net (r2u-net) for medical image segmentation. arXiv preprint arXiv:1802.06955.

Dieleman, S., Willett, K. W., & Dambre, J. (2015). Rotation-invariant convolutional neural networks for galaxy morphology prediction. *Monthly Notices of the Royal Astronomical Society, 450*(2), 1441–1459.

Hu, J., Shen, L. & Sun, G. (2018). Squeeze-and-excitation networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 18–22, Salt Lake City, USA. pp. 7132–7141.

Kon, S., Watabe, K. & Horibe, M. (2018). Nondestructive method using transmission line for detection of foreign objects in food. *Proceedings of the 2018 IEEE Sensors Applications Symposium (SAS)*, March 12–14, Seoul, South Korea, pp. 1–4.

Kwon, J.-S., Lee, J.-M. & Kim, W.-Y. (2008). Real-time detection of foreign objects using X-ray imaging for dry food manufacturing line. *Proceedings of the 2008 IEEE International Symposium on Consumer Electronics*, April 14–16, Vilamoura, Portugal, 1–4.

Lu, L., Shin, Y., Su, Y. & Karniadakis, G. E. (2019). Dying relu and initialization: Theory and numerical examples. arXiv preprint arXiv:1903.06733.

Misra, D. (2019). Mish: A self regularized non-monotonic neural activation function. arXiv preprint arXiv:1908.08681.

Mittal, S., Chopra, C., Trivedi, A. & Chanak, P. (2019). Defect segmentation in surfaces using deep learning. *Proceedings of the 2019 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT)*, Sep. 27–28, Ghaziabad, India, pp. 1–6.

Moghadas, S. M. & Rabbani, N. (2010). Detection and classification of foreign substances in medical vials using MLP neural network and SVM. *Proceedings of the 2010 6th Iranian Conference on Machine Vision and Image Processing*, Oct. 27–28, Isfahan, Iran, pp. 1–5.

Oktay, O., Schlemper, J., Folgoc, L. L., Lee, M., Heinrich, M., Misawa, K. & Kainz, B. (2018). Attention u-net: Learning where to look for the pancreas. arXiv preprint arXiv:1804.03999.

Ramachandran, P., Zoph, B. & Le, Q. V. (2017). Searching for activation functions. arXiv preprint arXiv:1710.05941.

Rong, D., Xie, L., & Ying, Y. (2019). Computer vision detection of foreign objects in walnuts using deep learning. *Computers and Electronics in Agriculture, 162*, 1001–1010.

Ronneberger, O., Fischer, P. & Brox, T. (2015). U-net: convolutional networks for biomedical image segmentation. *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention*, Nov. 18, Shenzhen, China. pp. 234–241.

Sarakon, P., Kawano, H. & Serikawa, S. (2019). Surface-defect segmentation using U-shaped inverted residuals. *Proceedings of the 2019 12th Biomedical Engineering International Conference (BMEiCON)*, Nov 19–22, Ubon Ratchathani, Thailand, pp. 1–4.

Simonyan, K. & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

Szegedy, C., Ioffe, S., Vanhoucke, V. & Alemi, A. (2017). Inception-v4, inception-resnet and the impact of residual connections on learning. *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, Feb. 4–10, San Francisco, California, USA, pp. 4278–4284.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Rabinovich, A. (2015). Going deeper with convolutions. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 7–12, Boston, USA, pp. 1–9.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J. & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 27–30, Las Vegas, NV, USA, pp. 2818–2826.

Wang, W., Yang, Y., Wang, X., Wang, W., & Li, J. (2019). Development of convolutional neural network and its application in image classification: A survey. *Optical Engineering, 58*(4), 040901.

Zeiler, M. D. & Fergus, R. (2014). Visualizing and understanding convolutional networks. *Proceedings of European Conference on Computer Vision*, Sep 8–14, Munich, Germany, pp. 818–833.

Zhao, X., Li, X., Yin, L., Feng, W., Zhang, N. & Zhang, X. (2019). Foreign body recognition for coal mine conveyor based on improved PCANSet. *Proceedings of the 2019 11th International Conference on Wireless Communications and Signal Processing* (WCSP), Oct 13–25, Xi'an, China, pp. 1–6.

Zhou, Z., Siddiquee, M. M. R., Tajbakhsh, N., & Liang, J. (2018). Unet++: A nested u-net architecture for medical image segmentation. *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support* (pp. 3–11). Cham: Springer.