APPLICATION OF DEEP NEURAL NETWORKS IN AUTOMATIC VISUAL INSPECTION OF UO_2 PELLETS


DAVID VERDEJO GARRIDO
Ingeniero de Ingeniería Mecánica e Industrial
ENUSA INDUSTRIAS AVANZADAS


PABLO RAMÍREZ HEREZA
Estudiante de doctorado y personal investigador del Grupo AUDIAS
GRUPO AUDIAS, UNIVERSIDAD AUTÓNOMA DE MADRID


ÁNGEL RAMOS GALLARDO
Ingeniero de Ingeniería Mecánica e Industrial
ENUSA INDUSTRIAS AVANZADAS


SERGIO ÁLVAREZ BALANYA
Estudiante de doctorado y personal investigador del Grupo AUDIAS
GRUPO AUDIAS, UNIVERSIDAD AUTÓNOMA DE MADRID


DOROTEO TORRE TOLEDANO
Catedrático de Universidad. Coordinador del Grupo AUDIAS
GRUPO AUDIAS, UNIVERSIDAD AUTÓNOMA DE MADRID

INTRODUCTION

All the parts of nuclear fuel assembly shall comply with high quality standards to ensure optimal and safe performance in the reactor. The last stage of UO_2 pellet manufacturing process is a visual inspection of all the pellets to discard those ones with defects on its surface.

Prior to the automatic pellet inspection equipment, pellets were visually inspected by a qualified inspector. In 2000, ENUSA developed its first Automatic Pellet Inspection Equipment (API); and in 2005 a second equipment (2nd generation) was developed. The main difference is that first generation equipment manages the pellets in trays (multiple pellets columns), while on the second generation inspects the pellets in line, stopping the column of pellets momentarily for its inspection.

In 2019, ENUSA developed a new API equipment for Jianzhong Nuclear Fuel (CJNF), nuclear fuel assembly manufacturer in Yibin (Sichuan, China), based on the first generation (pellets inspected in trays), but upgraded both hardware (control system, visual system, computer, etc.) and software (new inspection application developed by ENUSA).

ENUSA decided to investigate deep neural networks (DNN) and its implementation in APIs, within a collaboration project with AUDIAS (Audio, Data Intelligence and Speech) research group of *Universidad Autónoma de Madrid*. This group has considerable experience in neural networks and signal processing for real applications.

An outcome of this collaboration has been development of a defect detection and classification system based on deep neural networks. This system is capable to detect the defects on pellets surface and classify them using the pellets images acquired by the API.

API EQUIPMENT VISUAL SYSTEM AND INSPECTION PROCESS

The visual inspection process consists in observing the lateral surface of the pellets while they are rotating on some spinning rollers to detect damages or defects on them. On the Automatic Pellet Inspection equipment the pellet column image is acquired by a camera, together with a led illumination system. So API equipment performs a visual inspection similar to than performed by qualified inspectors, but more robust as it avoids the variability due to human factors (subjectivity, fatigue and similar). It should be noted that this system doesn't allow the direct observation of pellet end faces, but it has been proved that those defects which affect to end faces, extend and are revealed on the lateral surface (Figure 1).

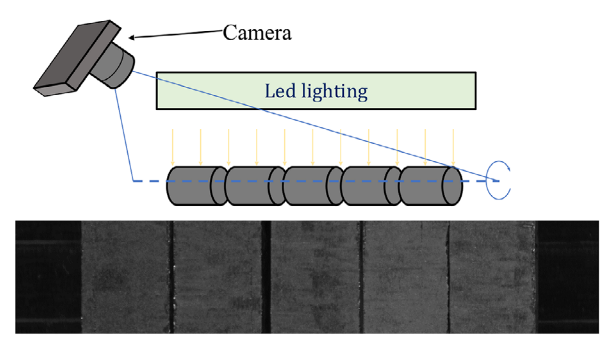

Figure 1. Basic scheme of visual system components disposition and example of acquired image.

Once the pellet column image is acquired, the software analyses the image of each pellet. Current system uses traditional image processing techniques for the image analysis. It is based on a combination of image adaptive thresholding, extraction of region of interest (ROI's) and their processing for defect detection.

Depending on the morphology and other characteristics of extracted areas, they are identified as defect (or not) and classified in different types of defects. Extracted defects characteristics are compared with the corresponding limits (depending on the defect type) to consider the pellet as accepted or rejected.

PELLET IMAGE ANALYSIS BY DEEP NEURAL NETWORKS

As previously indicated, current image analysis is based on traditional image process techniques, using algorithms with certain adaptation capability to image variations. It consists on the extraction of measurable image characteristics (numeric values that correspond to image properties such as brightness, contrast, variability, etc.). More characteristics than the ones used in current algorithms could be extracted from the images and ROI areas. Most probably, a detailed study of these characteristics and their combination will allow for a better defect detection and classification. However, due to the large amount of potential parameters and their multiple combinations, it would require to invest significant time, thus being impractical.

Image analysis by deep neural networks [1] provides a more efficient image analysis as it allows an automated learning of the most representative image and ROI characteristics, and a more efficient combination of them for defects on pellets detection and classification.

In particular, convolutional neural networks (CNNs) [2] are very well suited for image processing. They allow to extract characteristics that are progressively more complex and abstract, and their combination with different weighting factors. Provided that a large number of pellet images which contains defects are available (and properly labelled), an image analysis system based on DNN shall identify and classify them in a more reliable way.

Figure 2 shows a simplified scheme of a deep neural network structure for defect classification from a sub-image extracted from the complete image of the pellet. It consists of two parts: a convolutional neural network followed by a deep neural network. The convolutional layers extract the more representative image characteristics applying convolutional kernel masks. It is followed by maxpooling operations (reduction of the dimensionality of convolutional operation results by extracting the maximum value from n x n blocks). Once all these characteristics are extracted, they are redistributed in a big array (flattening) and inputted in a fully connected neural network that determines the defect presence and its type.

For the training of the defect classification based on neural networks, it's necessary a big database with the patterns to be identified by the network in some way; in other words, a bunch of data with known results. The determination of the desired result is performed by an expert, indicating the result of each image.

The process of identification, location and classification of defects in the pellet images is called "labelling". It's a highly time-consuming process but it shall be done in a rigorous way, since the network training result and the good network behavior depends on the quality of the database.

The training process is an iterative process where a set of data (training set) from a database is tested repeatedly. On each iteration, the neural network parameters (weights) are adjusted in order to minimize the error between the obtained result and the desired result provided by the labels of the data set. The results of each training iteration are compared to the results obtained in another data set (validation set). The training ends when the error on validation set reaches a minimum.

Finally, the global result of the training process (evaluation of obtained network) is determined by the result using a third set of data (test set).

Training process is performed with different network configurations (different number of layers, number of neurons

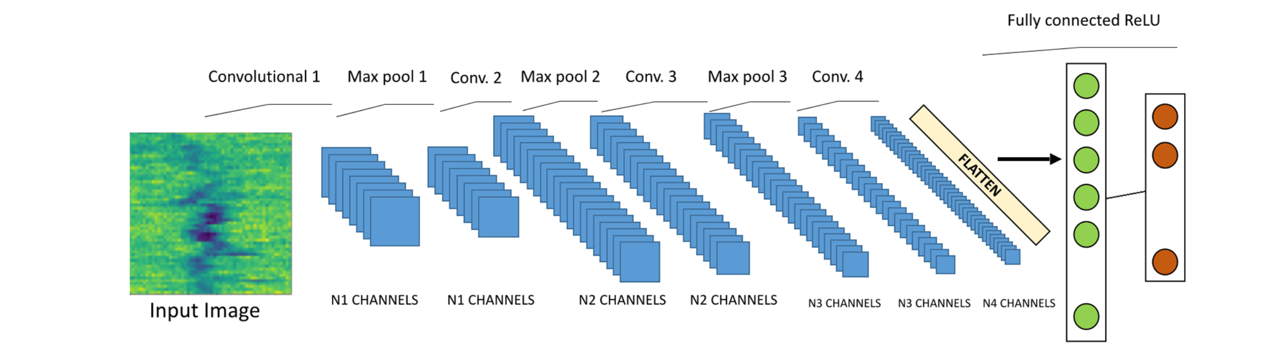

Figure 2. Basic structure of image classification system based on neural network.

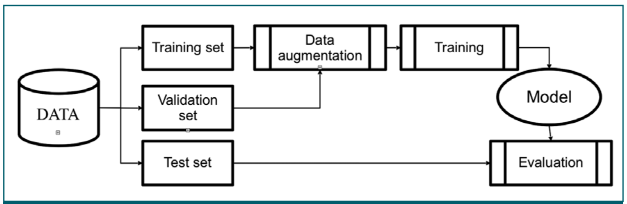

Figure 3. Training process and information division from data base.

per layer, interconnections, etc.) in order to get the optimal network structure (Figure 3).

As previously indicated, the more available data, the better the result will be obtained by the neural network because a big quantity of data provides the network the ability to generalize.

When it is not possible to collect a big quantity of data to create a good database, some additional data can be created in an artificial way. This is called "data augmentation" (explained below).

DEVELOPMENT OF DEFECT CLASSIFICATION SYSTEM BASED ON DEEP NEURAL NETWORKS

Initially, as a base, it was considered the use of an existing system of object detection and classification in images, exemplified by COCO evaluations [3]. YOLO [4] and RetinaNet [5] are two high-profile system considered. Both were studied and slightly tested, but these systems needs a large amount of images (thousands) to achieve a satisfactory result. This made their use not viable.

Due to the limited quantity of images of pellets and the need to obtain results in a relatively short period of time, the decision was to use a simpler system using less complex networks with less number of parameters and easy to train with less quantity of data. Existing neural networks for MNIST handwritten digits recognition [6] seemed to be similar to the desired goal. These type of neural networks served as an inspiration and were taken as starting point.

Image analysis system has been developed in Python 3.7 using packages for image processing (OpenCV) and neural networks and machine learning (Keras and TensorFlow).

The image analysis process is divided in two well differentiated parts:

- **Detection:** extraction of those parts (sub-images) of the pellet image were irregularities are detected and are potential defects.
- **Classification:** analysis of the areas extracted by the detection process to determine if they content a defect and, if so, classify it.

Detection process is performed using traditional segmentation techniques, in particular, OTSU global image thresholding algorithm. The project focused on the usage of neural network in defect determination and classification.

Multiple configurations of the deep neural network for defect determination and classification were tested in order to find a good network structure that provides reliable results.

The following network characteristics were changed to get different network configurations:

- Size for rescaling detected areas as input for the network.
- Number of convolutional layers.
- Size of the convolutional and max pooling kernel.
- Number of fully connected layers.
- Quantity of neurons on each fully connected layer.

The size and location of the sub-image is a significant information for defect classification. As it is made up of numerical values, it only can be added in fully connected layers. After several tests, the best option was to introduce it in the first fully connected layer.

For each network configuration, several training processes were performed using different training sets: using only available pellets, discarding some types of defects, increasing the database by creating defects in an artificial way (data augmentation). The most significant result are included and explained in the result section.

To provide more robustness and avoid an excessive adaptation of the network to the training set (known as "overfitting"), during the training some neurons are randomly deactivated. This technique, known as "dropout", provides to the network a greater generalization capacity because, to reach a good result although some neurons are disabled, it learns to rely on the general values of previous layers instead of on particular values of a single or a few units.

This system is set to detect the following defect types:

- Simple crack (end-crack, longitudinal, oblique, circumferential).
- Branch crack.
- Multiple crack (nail type, multiple cracks).
- Material losses (end-chamfer and lateral side).
- Pits.
- Surface defect (grinding defects, surface contamination).
- End-capping.
- End-chamfer damage.
- No defect.

Table 1 shows the details of the optimal network configuration which obtains the best results after several tests.

IMAGES AND DEFECTS DATABASE

During the development of the image analysis system, the database with labelled pellet images was created using images from API equipment. The objective was to obtain as much quantity of images as possible while trying to get a

	Layer	Output size
	Input image	56 x 56 x 1
Convolutional layers	Convolutional + Max pooling	28 x 28 x 32
	Convolutional + Max pooling	14 x 14 x 64
	Convolutional + Max pooling	7 x 7 x 64
	Flatten + size and location	3140 x 1
	Concatenation (flatten)	3140 x 1
Fully connected neural layers	Dense 1	128 x 1
	Dense 2	64 x 1
	Dense 3	9 x 1

Table 1. Architecture of convolutional block and fully connected block for defect classification.

Defect type	#
Simple crack	2443
End-capping	1932
Multiple crack	2335
End-chamfer damage	898
Material losses	976
Pits	583
Grinding/Cleaning	671

Table 2. Type and number of defects on labelling and current labelled number.

minimum number of images of each type of defect. It should be stressed that, although many images were available from the API equipment, until they are not labelled, they cannot be used to train the system.

Table 2 shows the number of labelled defects.

For an acceptable result, it would be desirable to have at least a minimum of about 500 samples per type.

In addition to the type of defects that the network is capable to recognize, the "no defect" type was added. Otherwise, those sub-images extracted by detection process that doesn't contain a defect (or contains small marks that are not defects) would be classified as the defect type whose features are more similar to the extracted area features, which would be a false detection. In order to rule out false detections, some samples without defects are introduced into the data set, labelled as "No Defect".

DATA AUGMENTATION SYSTEM

As for some type of defects the number of available samples was small, it was necessary to create a piece of software to generate more defects in an artificial way. This technique is known as "data augmentation". This tool allows the generation of large amounts of sample images (hundreds or thousands) in minutes.

An application was developed to generate new pellet images with defects in two different ways:

- Modifying the complete image: the pellet image is modified by changing its brightness, contrast, flipping it, rotating it, cropping it, and others.
- Generating artificial defects: one image of a pellet without defects is taken as a base. Some defects from other images are added on it. This can be easily done because using the labels, sub-images of the defects are already located and can be directly extracted.

In both cases, the original labels are modified for the generated image to correspond with the location of the defects on the new image.

RESULTS

This section includes the results obtained with the network architecture that gets the better defect classification success rate. It means what percentage of defects have been properly classified by the network comparing with defect label.
For all cases, the set of labels used to evaluate the network consisted of:

- **Training set:** depends on the test. Specified on each case.
- **Test set:** 50 real defects (not artificially generated) randomly chosen. Same set for all tests.
- **Validation set:** 100 real defects (not artificially generated) randomly chosen. Same set for all tests.

To obtain the better network architecture, multiple tests were performed for different network architectures, using input sizes of 28x28 px, 56 x 56 px and 112 x 112 px combined with 3, 4 and 5 convolutional layers. The results outlined below uses an input image size of 56 x 56 px and four convolutional layers.

First results were obtained **without using the data augmentation**, just using the available labelled images from the data base. To obtain a homogeneous number (about 2000 samples) for each type of defect, for those types where number of samples are lower, they are replicated up to reach the same quantity. This method allows to give the same level of importance to each defect type, but does not add particularly useful information for training the networks. A success level of **75.6 %** is achieved.

A second step was to **use data augmentation to homogenize the database** and obtain 4000 defects per type for training. Comparing the obtained results with the previous test one, the success rate improves as expected because the more samples are available for the network training process, the better result should be obtained. This tests achieved a success rate of **81.3 %**.

A third stage consisted on using data augmentation to obtain up to 10.000 defects per type. The success ratio improves up to **85.9 %**.

For final tests, the network architecture was improved a bit more by adding the defect information (location and size), and using dropout in training process. With this final network configuration, some tests were performed generating higher number of samples using data augmentation and get the final success rate of the image analysis system.

By generating 30000 samples per defect type, a **success rate of 90.08 %** is achieved (Table 3).

Figure 4 represents a confusion matrix that represents graphically the classification accuracy per defect type for the test dataset.

Generated samples per type	Validation success rate (%)	Test success rate (%)
10,000	87.32	88.40
20,000	87.14	89.56
30,000	**87.79**	**90.08**

Table 3. Final results with high number of generated samples by data augmentation.

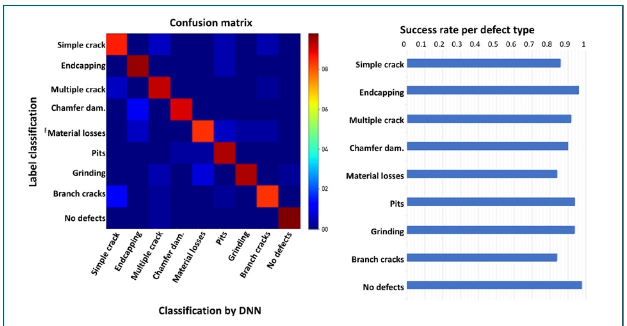

Figure 4. Confusion Matrix obtained by the final network on the test dataset.

It might be thought that, by generating more artificial samples, the success rate would improve up to almost 100%. However, it is not so for two main reasons. First, due to the inherent limits of the neural network system. And second, because a bunch of artificial data is generated from real data, and after a certain point it does not provide extra information to the training process.

Finally, it should be noted that the classification of regions without defect ("No Defect" type) is carried out with an accuracy of approximately a 96%, which makes possible to discard possible false detections obtained in the detection stage.

CONCLUSIONS

Deep neural networks have shown a big potential for image analysis, since they are able to extract more characteristics from the images and find more complex relations than using traditional techniques for image analysis.

This project proves that it is feasible to inspect UO_2 pellets using a system based on deep neural networks with positive results. The main advantages of a pellet image analysis system based on deep neural networks would be a greater reliability of detection and classification and less sensibility to variations on image conditions.

More types of defects could be included on this system without modifying it. In order to do so, it would be necessary to include images (and its labels) that contains a new type of defect, and to train the network with it. The same would be applicable for adding a new type of pellet, where its defects (or the pellet itself) might have a different aspect.

Currently ENUSA is integrating the system based on DNNs on API's inspection software to check its behaviour on real conditions. The coexistence of both systems will allow to quantify the benefits of using DNN system and continue improving it. The final stage will be to use DNN system as image analysis on API equipment.

Until the DNN system is not integrated, it won't be possible a comparison between them in terms of detection and classification accuracy because their detection and classification processes are not direct comparable. This final comparison can only be done once both system are working under the same conditions and previously adjusted to reach certain maximum values of false rejection and false acceptation.

REFERENCES

[1]. LeCun, Y. et al. Deep Learning. Nature 521, 28 May 2015, pp. 436-444 (2015).

[2]. Rawat, Waseem, and Zenghui Wang. "Deep convolutional neural networks for image classification: A comprehensive review." Neural computation 29.9 (2017): 2352-2449.

[3]. Chen, Xinlei, et al. "Microsoft coco captions: Data collection and evaluation server." arXiv preprint arXiv: 1504.00325 (2015).

[4]. Redmon, Joseph, and Ali Farhadi. "Yolov3: An incremental improvement." arXiv preprint arXiv: 1804.02767 (2018).

[5]. Lin, Tsung-Yi, et al. "Focal loss for dense object detection." Proceedings of the IEEE international conference on computer vision. 2017.

[6]. Deng, Li. "The mnist database of handwritten digit images for machine learning research [best of the web]." IEEE Signal Processing Magazine 29.6 (2012): 141-142.■