Detection of Steel Surface Defect Based on Machine Learning Using Deep Auto-encoder Network

Ehab A. Kholief
Industrial Engineering
Faculty of Engineering, Alexandria University
Alexandria, 21544, Egypt
ehabkholief@gmail.com

Samy H. Darwish
Electrical Engineering Department
Faculty of Engineering, Pharos University
Alexandria, 21648, Egypt
salord1960@yahoo.com

M. Nashat Fors
Industrial Engineering
Faculty of Engineering, Alexandria University
Alexandria, 21544, Egypt
nashatfors@gmail.com

Abstract

The non-contact inspection of surface defects has become more and more important in the manufacturing industrial systems because of the great demands on the quality of high surface finishes. The machine learning achieved impressive recognition rates in image classification tasks. In order to exploit those capabilities, this paper represents a detection and classification of surface defects on hot rolled steel strip by means of differently captured digital intensity images of that process samples. The feed-forward artificial neural networks and deep auto-encoder network as classifiers are trained for detecting six popular classes of steel defects, i.e., crazing, patches, pitted surface, inclusion, rolled-in scale, and scratches. A comparative study between the two classifiers is done with respect to the cross validation and the confusion matrix. The results demonstrate excellent defect detection outcome with very low false rates.

Keywords
Machine Learning, Surface Defects Inspection, Classifier, Artificial Neural Networks (ANN), Deep Auto-encoder Network (DAN).

1. Introduction

The hot-rolled steel strip has an essential role in many industrial fields, such as aerospace, aircraft, shipbuilding, automobiles, and machinery manufacturing (Zhao and Luo, 2008; Shao et al, 2015). There are some unavoidable defects in hot-rolled steel strip; hole, scratch, rolled in scale, crack, pits/scab, edge defect/coil break, shell, lamination, sliver (Neogi et al, 2014). These defects do not only impact the manifestation of the product but also reduce the features of the product such as corrosion resistance, abrasion resistance, and fatigue strength which make huge economic losses (Song and Yan, 2013). The manual inspection process is not appropriate to warranty defect-free surface of steel strip with a reasonable degree of dependability and naturally. Therefore, many automatic surface defect inspection techniques were generated to meet the customer demands and minimize the economic losses: for instance, strobe light, eddy currents, magnetic leakage flux, infrared ray, X-ray, and machine vision technology (Neogi et al, 2014; Luo and He, 2016; Zhao et al, 2016).

Many researchers introduced machine vision steel surface inspection systems. Caleb and Steuer (2000) used artificial neural networks (ANN) to detected defects in hot rolled steel strip, and overall accuracy achieved 80%. Pakkanen et al (2004) applied edge histogram, color structure, and homogeneous texture as features extractor, and K-Nearest Neighbor as classifier on hot-rolled steel strip surface to detect the defected images with overall accuracy is 76%. Hongbin et al (2004) and Keesug et al (2006) applied support vector machine (SVM) as classifier of the inspection system, SVM give better performance than ANN for their samples on hot rolling steel, and achieved accuracy 90%. Smriti and Bhandari (2008) considered edge detection with modified scheme based on heuristics used by human inspectors for identifying surface imperfections to compute the features then applied SVM as classifier for classifying surface images into two classes defective and defect-free; the system was applied on surface texture database and achieved 94.19% correct classification.

Sharifzadeh et al (2008) used image processing algorithms for detecting four popular classes of steel defects and the result of test 90%. Liu et al (2010) used relevance vector machine as classifier to detect four kinds of defects on the steel surface and the correct rate of the defect detection 89%. Luiz et al (2010) adopted Principal Component Analysis as features extractor, and Self-Organizing Maps as classifier to classify six classes of the hot-rolled steel surface defects with classification accuracy 87%. Song, and Yan (2013) proposed a new method Adjacent Evaluation Completed Local Binary Patterns as feature extractor and employed SVM as classifier on Northeastern University (NEU) hot-rolled steel strip surface defect database, the resulted accuracy 97.8%. Song et al (2014) adopted scattering convolution network as feature extractor and employed SVM as classifier on NEU database and the overall accuracy is 98.6%.

Wang et al (2016) proposed fault diagnosis based on a continuous sparse auto-encoder and illustrated the effectiveness of the presented approach by IEC TC 10 dataset of transformers faults and the accuracies from 93.3%:100%. Mao et al (2016) proposed an intelligent fault-diagnosis by auto-encoder algorithm the effectiveness of the presented approach is verified by rolling element bearings data set the accuracies from 98.5%:100%. Lu et al (2016) used stacked denoising auto-encoder as fault diagnosis method and rotating machinery datasets are employed to demonstrate the effectiveness of the proposed method the accuracies from 91.6%:99.8%.

Based on machine vision idea, this paper presents a machine learning processing technique to identify and classify strip surface defects without contact based on extracted features from digital intensity images of hot-rolled steel. The proposed technique consists of two classifiers, the first one is ANN as a supervised classifier that trained by two extracted features as histogram and edge detection. The second classifier is deep auto-encoder network (DAN) as an unsupervised classifier where the typical image was used as an input. The two classifiers are used to verify whether healthy or faulty surfaces and to identify the defects.

2. Methodology

2.1 Workflow

The overall framework as shown in figure 1 consists of three stages; the first stage is the input preparation, the second stage is feature extraction that used to extract features as the intensity histogram and edge detection both are used as an input to the ANN classifier. The third stage is the classifiers where the two proposed classifiers are trained and tested.

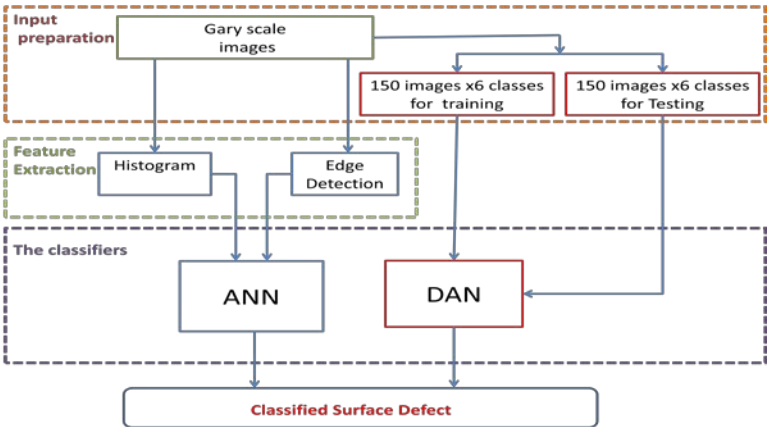
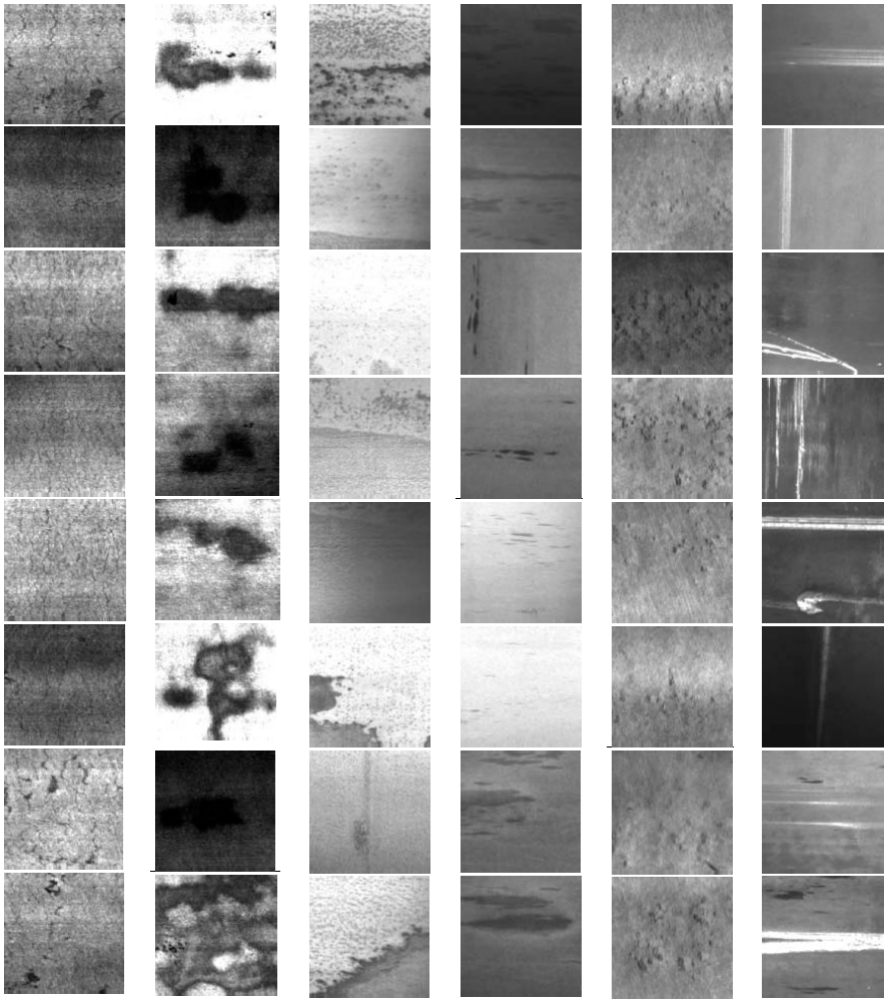


Figure 1. Inspection block diagram of the proposed method

2.2 Input preparation

In order to demonstrate the effectiveness of the proposed technique to classify hot-rolled steel strip surface defects, a NEU surface defect database is used. According to defect catalogues published by Verlag Stahleisen GmbH, Germany, the famous defects in hot-rolled steel strip are; crazing (Cr), patches (Pa), pitted surface (PS), inclusion (In), rolled-in scale (RS), and scratches (Sc) (Lee et al, 2001; Ghorai et al, 2013; Neoga et al, 2014; Frank, 2016). In this database, six classes of typical surface defects of the hot-rolled steel strip are collected. The used datasets includes 1800 grayscale images: 300 samples each of six different classes of typical surface defects. Figure 2 shows sample images of six classes of typical surface defects; the original resolution of each image is 200 × 200 and physical size 5.29×5.29 centimeters which verifies measurement length of hot rolling (hot rolling has cut - off length = 8mm, measurement length = 6×8mm = 4.8 centimeters) (Anand and Vinay, 2009). The NEU surface defect database includes two difficult challenges; first one the same class of defects existing large differences in appearance, for instance, the scratches may be horizontal scratch vertical scratch, and slanting scratch, etc, while the different classes of defects have similar aspects, e.g., rolled-in scale, crazing, and pit-ted surface. Second challenge is the defect images suffer from the influence of illumination and material changes (Song et al, 2014).
All images are processed by histogram and edge detection to extract features which are used for training and test ANN classifier. Typical images are used as input for DAN classifier, where the samples are divided into two sets: set-1 (150) random samples of each class were used for training and set-2 another 150 samples of each class are used for testing.



Crazing Patches Pitted Surface Inclusion Rolled-in scale Scratch

Figure 2. Samples of six classes of typical surface defects on NEU surface defect database

2.3 Feature Extraction

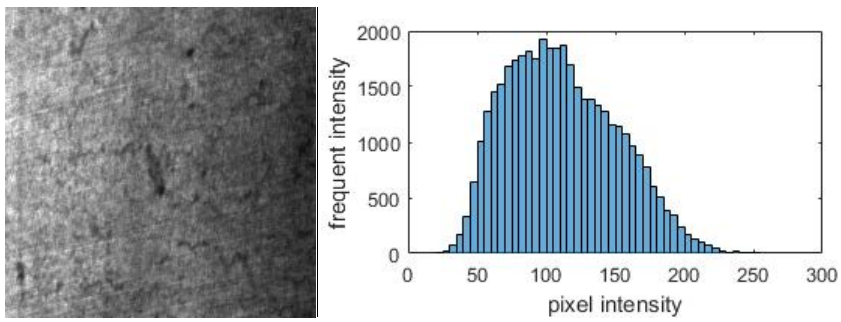
Many researchers were used different features to classify images. Smriti and Bhandari (2008) adopted edge detection to compute the features to classify test surface images. Sharifzadeh et al (2008) used edge detection as feature extractor for detecting four popular classes of steel defects. Kobayashi (2013) used histogram as feature extractor to image classification. Mary et al (2015) adopted four different feature sets and histogram had the best performance. Ashour et al (2015) were adopted histogram and edge detection as feature extractors for classifying surfaces.
In this paper two feature extracted methods are used for fault classification and identification they are histogram and edge detection.

2.3.1 Histogram

The histogram shows the distribution of data values, it distributes the bins along the x-axis between the minimum and maximum intensity values. For an image z with pixel intensity values ranging from 0 to L-1, and a normalized histogram p with a bin for each possible intensity level, the probability of the occurrence of intensity value k can be written as (Wajid and Hussain, 2015).

$$p(k) = \frac{\text{number of pixels with intensity } k}{\text{total number of pixels in the image}} = \frac{n_k}{n} \quad k : \text{ 0 to L} - 1 \tag{1}$$

Histogram shows the pixel distribution of grayscale values, ranging from 0 to 255. Figure 3 gives an example of a 256-bin histogram for one hot-rolled steel strip surface image.



(a) (b)

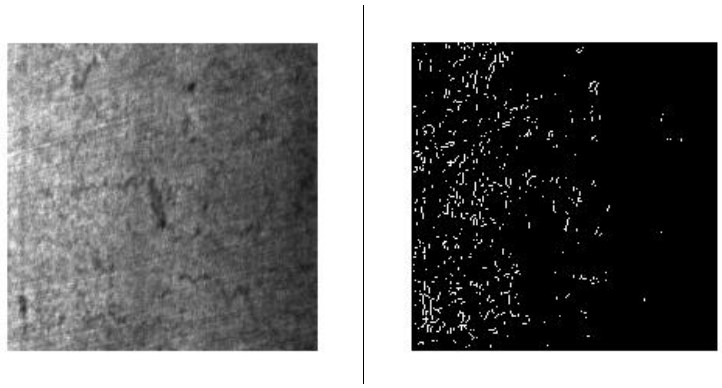
Figure 3. (a) The Grayscale Image of a hot-rolled steel strip surface
(b) The resultant 256-bin histogram

2.3.2 Edge detection

Edges exist when rapid intensity changes occur in an image. These edges can be used to determine (and analyze) the texture of a particular surface; in this paper, the Sobel edge detection algorithm is used (Smriti and Bhandari, 2008), where an image is convolved with a vertical (H_x) and horizontal (H_y) filter. The results of these convolutions are then combined into a single Sobel edge-intensity map s_{mag} as in equation (2).

$$s_{mag} = \sqrt{(1 \times H_x)^2 + (1 \times H_y)^2} \tag{2}$$

A threshold is then empirically determined and applied to this map to produce a final edge map (Smriti and Bhandari, 2008). An example for applying the Sobel edge detector on one hot-rolled steel strip surface image (I) is shown in figure 4.



(a) (b)

Figure 4. (a) The Grayscale Image of a hot-rolled steel strip surface I
(b) Detected edges of I image using the Sobel edge detector

2.4 The classifiers

Two classifiers were proposed, the first one is supervised feed-forward back propagation (BP) ANN and the second is DAN.

2.4.1 ANN classifier

The idea behind neural networks is that many neurons can be linked together by communication links to execute complex computations. It is common to depict the structure of a neural network as a graph whose nodes are the neurons and each edge in the graph links the output of some neuron to the input of another neuron.
A feed-forward neural network is described by a directed acyclic graph, $G = (V, E)$ where V are the layers, and E are the edges.
Each single neuron is modeled as a simple scalar function, $\sigma: R \rightarrow R$. There are three possible functions for the activation function of the neuron σ:

a. The sign function, $\sigma\,(a) = sign\,(a)$,
b. The threshold function, $\sigma\,(a) = 1_{\,(a>0)}$, and
c. The sigmoid function, $\sigma\,(a) = 1/\,(1 + exp\,(-a))$ which is a smooth approximation to the threshold function.

Each edge in the graph links the output of some neuron to the input of another neuron. The input of a neuron is gained by taking a weighted sum of the outputs of all the neurons connected to it, where the weighting is according to ω.
The network is organized in layers. That is, the set of nodes can be put into a union of (nonempty) disjoint subsets V, such that every edge in E connects some node in V_{t-1} to some node in V_t, for some $t \in [T]$ where T is the number of layers in the network (excluding V_0). V_0 is called the input layer, it contains $n + 1$ neurons, where n is the dimensionality of the input space.
For each $i \in [n]$, the output of neuron $i\ in\ V_0$ is simply x_i. The last neuron in V_0 is the constant neuron, which always outputs 1. Where $v_{t,i}$ denotes the *i*th neuron of the *t*th layer and $o_{t,i}$(x) denotes the output of $v_{t,i}$ when the network is fed with the input vector x. Therefore, $i \in [n]$, $o_{0,i}(\mathrm{x}) = x_i$ and for $i = n + 1$, $o_{0,i}(\mathrm{x}) = 1$. To calculate the outputs of the neurons at layer $t + 1$ let $v_{t+1,j} \in V_{t+1}$ and $a_{t+1,j}$(x) denote the input to $v_{t+1,j}$ when the network is fed with the input vector x equation (3) and (4) are used; (Shalev-Shwartz and Ben-David, 2014)

$$a_{t+1,j}(x) = \sum_{r:\,(v_{t,r}, v_{t+1,j})\,\in\,E} \omega\,((v_{t,r}, v_{t+1,j}))\; o_{t,r}(x) \tag{3}$$

And

$$o_{t+1,j}(x) = \sigma\,(a_{t+1,j}(x)) \tag{4}$$

The ANN learning includes the following steps:

1. Collect the data and determine input/output relationship.
2. Pre-process the data and divide them into training and testing data sets.

3. Train the network with the associated data whereby the target output at each output neuron is compared with the actual network output whereby the difference or error is minimized by adjusting the weights and biases through some training algorithms.

The BP algorithm uses a chain rule to compute the derivatives of the error function with respect to the weights and biases in the hidden layers. In order to minimize the error function, optimization method Conjugate Gradient (CG) is used (Ghaisari et al, 2012).

2.4.2 DAN classifier

An auto-encoder is one type of unsupervised neural networks with three layers and the output target of the auto-encoder is the input data. As depicted in figure 5, the auto-encoder comprises two parts, i.e., encoder network and decoder network. The encoder network transforms the input data from a high-dimensional space into codes in a low-dimensional space and the decoder network reconstructs the inputs from the corresponding codes. The encoder network is explicitly defined as an encoding function denoted by f_θ. This function is called the encoder. For each image x^m from a dataset the encode vector H^m obtained by equation (5):

$$H^m = f_\theta\,(x^m) \tag{5}$$

The decoder network is defined as a reconstruction function denoted by $g_{\grave{\theta}}$, namely the decoder. It maps H^m from the low-dimensional space back into the high-dimensional space, producing a reconstruction equation (6):

$$\acute{x}^m = g_{\grave{\theta}}\,(H^m) \tag{6}$$

The parameter sets of the encoder and decoder are learned simultaneously on the task of reconstructing as well as possible the original input, attempting to incur the lowest possible reconstruction error $L(x, \acute{x})$ over the M training examples, where $L(x, \acute{x})$ is a loss function that measures the discrepancy between x and $\acute{x}$. In summary, the auto-encoder training aims to find the parameter sets θ and θ' minimizing reconstruction error as in equation (7):

$$\emptyset_{AE}\left(\theta, \grave{\theta}\right) = \frac{1}{M}\sum_{m=1}^{M} L(\,x^m, g_{\grave{\theta}}\,(\,f_\theta\,(\,x^m))) \tag{7}$$

The commonly used forms for the encoder and decoder are affine mappings, optionally followed by a non-linearity as in the equations (8), (9), and (10):

$$f_\theta\,(x) = s_f\,(W_x + b) \tag{8}$$
$$g_{\theta'}(x) = s_g(W^T x + d) \tag{9}$$
$$L(x, \acute{x}) = \|x - \acute{x}\|^2 \tag{10}$$

Where s_f and s_g are the encoder and decoder activation functions, respectively. Thus, the parameter sets of the auto-encoder are $\theta = (W, b)$ and $\grave{\theta} = (W^T, d)$ where **b** and **d** are bias vectors, and $\boldsymbol{W}$ and $\boldsymbol{W^T}$ are the weight matrices (Baldi, 2012; Goodfellow et al, 2015).

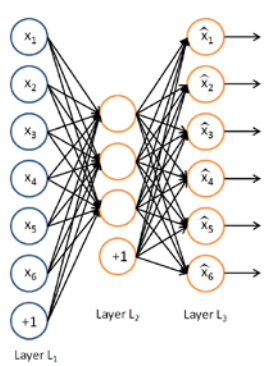


Figure 5. Architectural graph of an auto-encoder

N auto-encoders could be stacked to pre-train an N-hidden-layer of DAN. When given input x^m, the input layer and the first hidden layer of the DAN are regarded as the encoder network of the first auto-encoder. After the first auto-encoder is trained through minimizing the reconstruction error, the trained parameter set θ_1 of the encoder network is used to initialize the first hidden layer of the DAN. And the first encode vector H_1^m of the x^m is calculated as in equation (11):

$$H_1^m = f_{\theta_1}(x^m) \tag{11}$$

Then the encode vector □$_1^m$ is the input data, the first hidden layer and the second hidden layer of the DAN are regarded as the encoder network of the second auto-encoder. Correspondingly, the second hidden layer of the DAN is initialized by the second trained auto-encoder. The process is conducted in the sequence until the Nth auto-en coder is trained for initializing the final hidden layer of the DAN. And the Nth encode vector H_N^m of the x^m is calculated as in equation (12):

$$H_N^m = f_{\theta_N}(H_{N-1}^m) \tag{12}$$

Where θ_N is the parameter set of the Nth auto-encoder.
In this way, through training N stacked auto-encoders, all the hidden layers of the DAN are pre-trained. This pre-training process is proven to yield significantly better local minima than random initialization of the DAN and helps achieve better generalization in classification tasks.
After the DAN is pre-trained, fine-tuning process is utilized in next step of the DAN training. The output layer of the DAN is employed to contain the output targets for classification tasks .The output of the DAN calculated from the input x^m as in equation (13) (Baldi, 2012; Goodfellow et al, 2015):

$$y^m = f_{\theta_{N+1}}(H_N^m) \tag{13}$$

Where θ_{N+1} is the parameter set of output layer.
DAN with two auto-encoders (hidden layers) was constructed to classify NEU surface defects of the hot-rolled steel strip images. First the two hidden layers were trained individually in an unsupervised method using auto-encoders. Then a final softmax layer was trained in a supervised method, and the layers were joined together to form the proposed DAN as in figure 6.

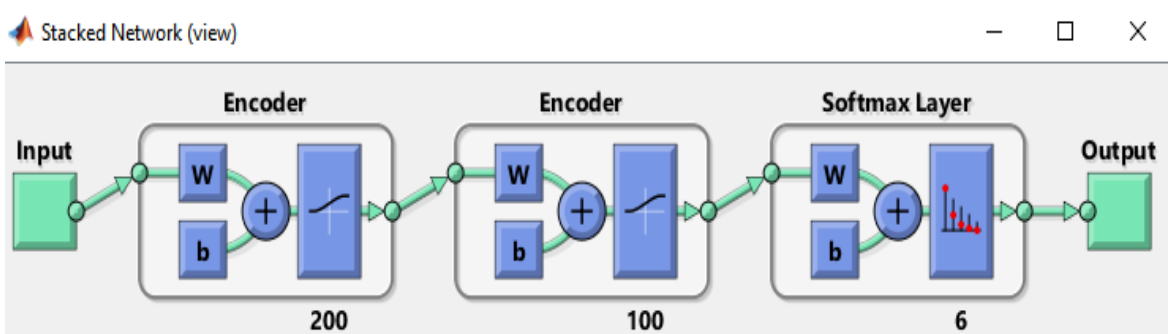


Figure 6.Proposed DAN block diagram

3. Results and Discussion

ANN classifier was trained and tested in cross validation by 10% to 90% of all samples, while DAN was trained and tested; the output is a confusion matrix which gives the overall accuracy.

3.1 The result for training- cross validation concept in ANN

Related to the two feature extracted methods that applied to input NEU images as histogram and edges; the next tables can represent the results to satisfy the training cross validation concept.

3.1.1 The result related to histogram technique

The total datasets (300 NEU images) of each class were divided into two parts; one part contains (270) samples (90%) for training and the second part takes (30) images (10%) for cross validation. The ANN training results for one group of histogram is shown in figure 7, while table.1 represents the O/P results of the ANN performance as a classifier using the intensity histogram NEU image.

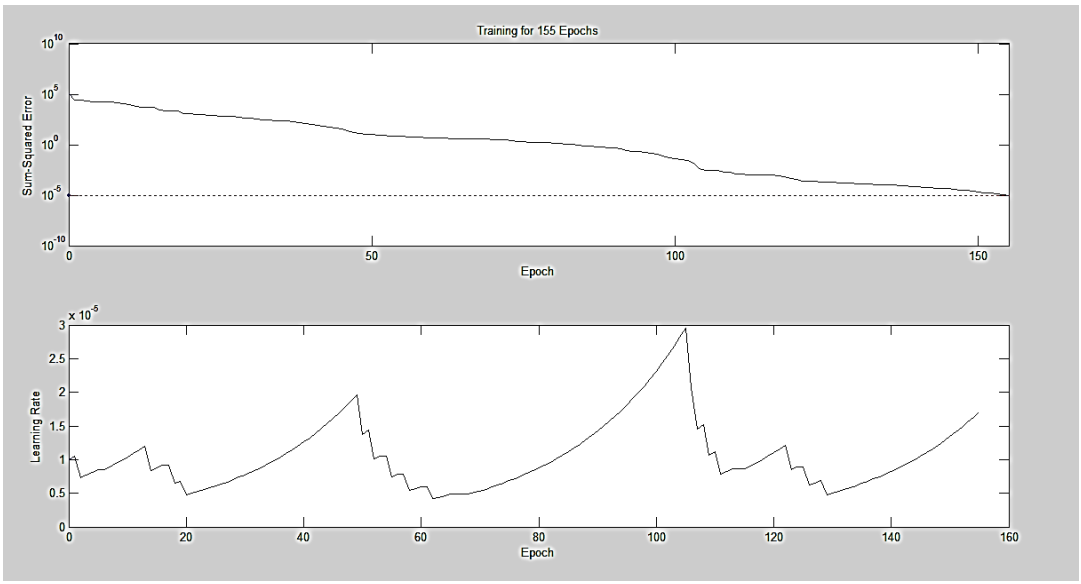


Figure. 7 The sum squared error and learning rate to training epochs for one group of histogram at ANN

Table 1. ANN performance using histogram

Defect class name	CR	LN	SC	RS	PS	PA
Training (undefined/total) samples	0/270	0/270	0/270	0/270	0/270	0/270
Cross validation (undefined/total) samples	16/30	21/30	25/30	13/30	8/30	26/30
Training Accuracy %	100	100	100	100	100	100
Cross validation accuracy %	46.66	30.00	16.66	56.66	73.33	13.33
Training epochs	2936	2874	2911	2895	2950	2866
No of I/P neurons	256					
No of hidden neurons	50					

From figure 7. It's clear to notice that the training can be done with low number of epochs with respect to suitable sum squared error and learning rate. From table 1 one can see that the number of hidden nodes of the ANN for the histogram feature was 50, the size of the input datasets are 256×300, and the total number of training epochs 17432. ANN was capable of classifying the accuracies 100% for training and 13.33 to 73.33 for cross validation.

3.1.2 The result related to edge detection technique

The ANN performance related to the morphological operation as edge detection that used in the second feature extraction analysis is illustrated in table 2, while ANN training results for one group of edge detection is shown in figure 8. Where total 300 defect images of each class were divided into two parts; one part contains (270) samples (90%) for training and the second part takes (30) images (10%) for cross validation.

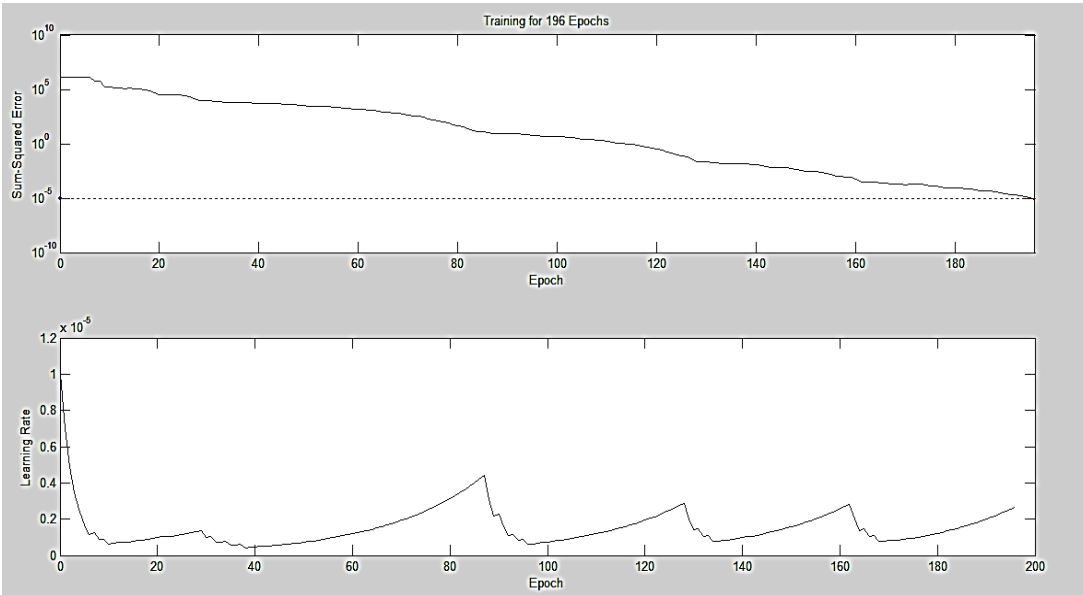


Figure 8. The sum squared error and learning rate to training epochs for one group of edge detection at ANN

Table 2. ANN performance using edge detection

Defect class name	CR	LN	SC	RS	PS	PA
Training (undefined/total) samples	0/270	0/270	0/270	0/270	0/270	0/270
Cross validation (undefined/total) samples	16/30	21/30	25/30	13/30	8/30	26/30
Training Accuracy %	100	100	100	100	100	100
Cross validation accuracy %	53.64	39.33	26.22	66.54	83.44	23.22
Training epochs	3004	3011	2954	2892	3020	3066
No of I/P neurons	950					
No of hidden neurons	50					

From figure 8. It's clear to notice that the training can be done with low number of epochs with respect to suitable sum squared error and learning rate. From table 2 one can see that the number of hidden nodes of the ANN for the edge feature was 50, the size of the input datasets are 950×300, and the total number of training epochs 17947. ANN was capable of classifying the accuracies 100% for training and 23.22% to 83.44% for cross validation.

3.2 DAN performance

Table.3 represents the O/P results of the DAN performance as a classifier by the confusion matrix.

Table 3. The confusion matrix of DAN on the NEU surface defect database

Output class							
Cr	150 16.7%	0 0%	0 0%	0 0%	0 0%	0 0%	100% 0%
In	0 0%	150 16.7%	0 0%	0 0%	0 0%	0 0%	100% 0%
Pa	0 0%	0 0%	150 16.7%	0 0%	0 0%	0 0%	100% 0%
PS	0 0%	0 0%	0 0%	150 16.7%	0 0%	0 0%	100% 0%
RS	0 0%	0 0%	0 0%	0 0%	150 16.7%	0 0%	100% 0%
Sc	0 0%	0 0%	0 0%	0 0%	0 0%	150 16.7%	100% 0%
	100% 0%	100% 0%	100% 0%	100% 0%	100% 0%	100% 0%	100% 0%
	Cr	In	Pa	PS	RS	Sc	

Target class

In table 3the green diagonal cells show the number and percentage of correct classifications of each class of defects by the trained network, in the first cell the all 150 images are correctly classified as Cr, this corresponds to 16.7% of all 900 images is tested. At the red cells at the first column can see that Cr images weren't classified as another class and 100% are correctly classified as Cr. At the first row can see that zero of other classes of defects were classified as Cr defect, and out of 150 Cr images, 100% are correct and 0% are wrong. Similarly for all columns and rows. Overall, 100% of the classified surface defects are correct and 0% are wrong classifications at blue cell.

4. Conclusion

This paper proposes a novel intelligent non-contact inspection technique as DAN, where it processes the massive raw data and automatically provides accurate results without use extracted features. Furthermore, a supervised ANN with two statistical features as histogram and edge detection to processes the same data was presented.

The effectiveness of the proposed DAN and ANN is verified using 1800-grayscale images for six popular classes of steel defects that prepared by NEU as crazing (Cr), patches (Pa), pitted surface (PS), inclusion (In), rolled-in scale (RS) and scratches (Sc).

For Cross-Validation of ANN classifier, the database is divided into two sets, first one is (90%) for training and second one is (10%) for test. When using the histogram technique, the ANN was capable an identifying accuracies up to 100% for training and from 13.33% to 73.33% for cross validation; while the accuracies reaches to100% for training and, from 23.22% to 83.44% when edge detection is used.

Two group of database (50% for training and 50% for testing) are used when satisfied the DAN classifier. The output validation for DAN is tabulated in a confusion matrix which present that overall accuracy is 100%. This work demonstrated that the DAN gives an efficient good performance in classification instead of it process a huge raw data without any reduction, also its activation process is done with untrained 50% of database for validation.

References

Anand, K., and Vinay A., *Metrology & Measurement*, 1st edition, Tata McGraw-Hill Education, 2009.

Ashour, M., Halin, A., Khalid F., Abdullah L., and Darwish SH., Texture-based classification of workpiece surface images using the Support Vector Machine, *International Journal of Software Engineering and Its Applications*, vol. 9, no. 10, pp. 147-160, 2015.

Baldi, P., Autoencoders Unsupervised Learning and Deep Architectures, *ICML Unsupervised and Transfer Learning*, vol. 1, pp.37-50, 2012.

Caleb, P., and Steuer, M., Classification of surface defects on hot rolled steel using adaptive learning methods, *Fourth International Conference on Knowledge-Based Intelligent Engineering Systems and Allied Technologies*, Bright, UK, August 30 – September 1, 2000.

Frank, T., Surface Defects in Hot Rolled Flat Steel Products, Verlag Stahleisen GmbH, Available: http://en.stahl-online.de/index.php/about-us/verlag-stahleisen-gmbh/, October 10, 2016.

Ghaisari, J., Jannesari, H., and Vatani, M., Artificial neural network predictors for mechanical properties of cold rolling products, *Advances in Engineering Software*, vol. 45, no. 1, pp.91-99, 2012.

Ghorai, S, Mukherjee, A., Gangadaran, M., and Dutta, P., Automatic defect detection on hot-rolled flat steel products, *IEEE .Transactions on Instrumentation and Measurement*, vol. 62, no. 3, pp.612-621, 2013.

Goodfellow, I., Bengio, Y., and Courville, A., *Deep Learning*, MIT Press book, www.deeplearningbook.org, 2015.

Hongbin, J., Yi, L., and Jianjun, S., An Intelligent Real-time Vision System for Surface Defect Detection, *17th International Conference on Pattern Recognition*, Cambridge, UK, August 23-26, 2004.

Keesug, C., Kyungmo, K., Lee, J., Development of defect classification algorithm for POSCO rolling strip surface inspection system, *SICE-ICASE International Joint Conference*, Busan, Korea, Oct. 18-21, 2006.

Kobayashi, T., BFO meets HOG: Feature extraction based on histograms of oriented p.d.f. gradients for image classification, *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Portland, Oregon, USA, 23-28 June 2013.

Lee, J., Park, C., Jung, J., Mark, R., Mark, D., The Development of Surface Inspection System Using the Real-time Image Processing, *3rd WSES International Conference on Mathematics and Computers in Mechanical Engineering MCME*, Piraeus, Greece. 2001.

Liu, Y., Hsu, Y., and Sun, Y., A Computer Vision System for Automatic Steel Surface Inspection, *5th IEEE Conference on Industrial Electronics and Applications*, USA, June 15-17, 2010.

Lu, C., Zhen-Ya, W., Wei-Li, Q., and Jian, M., Fault diagnosis of rotary machinery components using a stacked denoising auto-encoder based health state identification, *Signal Processing*, vol. 130, pp. 377-388, 2016.

Luiz, A., Flavio, L., and Paulo, E., Automatic detection of surface defects on rolled steel using Computer Vision and Artificial Neural Networks, *36th Annual Conference on IEEE Industrial Electronics Society*, Glendale, USA, November 7-10, 2010.

Luo, Q., and He, Y., A cost-effective and automatic surface defect inspection system for hot-rolled flat steel, *Robotics and Computer-Integrated Manufacturing*, vol. 38, pp. 16-30, 2016.

Mao, W., He, J., Li, Y., and Yan, Y., Bearing fault diagnosis with auto-encoder extreme learning machine: A comparative study, *Journal of Mechanical Engineering Sc*ience, vol. 1, no. 1, pp. 1–19, 2016.

Mary, A. ,Chacko, M.,and Dhanya, P., A Comparative Study of Different Feature Extraction Techniques for Offline Malayalam Character Recognition, *Computational Intelligence in Data Mining*, Kochi, India 2015.

Neogi, N., Mohanta, D., and Dutta, P., Review of vision-based steel surface inspection systems, *Eurasip Journal on Image and Video Processing*, vol. 50, pp.1-19, 2014.

Pakkanen, J., Iivarinen, J., Rautkorpi, R., and Rauhamaa, J., Content-based retrieval of surface defect images with PicSOM, *International Journal of Fuzzy Systems*, vol. 6, no. 3, pp.160-167, 2004.

Shalev-Shwartz, S., Ben-David, S., *Understanding Machine Learning: From Theory to Algorithms*, 1st Edition, Cambridge University Press, 2014.

Shao, C., Xu, G., Xu, H., et al., Research of Recognition Method for Surface Defects of Hot-rolled Round Steel Based on Image Processing, *Advanced Materials Research*, vol. 1090, pp.84-89, 2015.

Sharifzadeh, M., Alirezaee, S., Amirfattahi, R., and Sadri, S., Detection of steel defect using the image processing algorithms, *12th IEEE International Multitopic Conference*, Karachi, Pakistan, December 23-24, 2008.

Smriti, H. and Bhandari, S., A Simple Approach to Surface Defect Detection, *the Third International Conference on Industrial and Information Systems*, Kharagpur, India, December 8-10, 2008.

Song, K., and Yan, Y., A noise robust method based on completed local binary patterns for hot-rolled steel strip surface defects, *Applied Surface Science*, vol. 285, pp. 858-864, 2013.

Song, K., Hu, S, Yan, Y., Automatic Recognition of Surface Defects on Hot-rolled Steel Strip using Scattering Convolution Network, *Journal of Computational Information Systems*, vol. 7, pp. 3049-3055, 2014.

Wajid, S., and Hussain, A., Local energy-based shape histogram feature extraction technique for breast cancer diagnosis, *Expert Systems with Applications*, vol. 42, no. 20, pp. 6990-6999, 2015.

Wang, L., Xiaoying, Z., Jiangnan, P., and Gongyou, T., Transformer fault diagnosis using continuous sparse autoencoder, *Springer Plus* ,vol. 5, pp. 1-13, 2016.

Zhao, X., and Luo, S., Research on System of Quality Assessing and Defects ' Information Managing of Strip Steel Surface, *Advanced Materials Research* ,vol. 46, pp. 173-180, 2008.

Zhao, Y., Yan, Y., and Song, K., Vision-based automatic detection of steel surface defects in the cold rolling process : considering the influence of industrial liquids and surface textures, *The International Journal of Advanced Manufacturing Technology*, vol. 86, pp. 3511-3524, 2016.

Biography

Ehab A. Kholief obtained his BSc in Communication from the MTC in 1994 and MSc from Production Engineering Department, Faculty of Engineering, Alexandria University, Egypt in 2011. He is currently a PhD student in the same university. His research interests focus on the application of computer vision to solve manufacturing and production related issues.

Samy H. Darwish, is an affiliate instructor at the Faculty of Engineering, Pharos University in Alexandria, Egypt. He obtained his BSc in Communications and Electro-Physics in 1983 from Alexandria University. He then obtained both his MSc and PhD in Electrical Engineering from the same university in 2000 and 2007, respectively. Dr Samy's research interests include, but are not limited to, signal/image processing applications and artificial intelligence.

M. Nashat Fors is Professor Emeritus of Industrial Engineering, Production Engineering Department, Alexandria University, Egypt. He earned his PhD in Industrial Engineering 1981 from Production Engineering Department, Faculty of Engineering, Alexandria University. He has written or co-authored research papers and articles on Industrial Applications of Mathematical Programming, Supply Chain Management, Maintenance planning, Water Management, Scheduling & Distribution. Dr. Fors has many joint projects, consultation, and training programs with industry in the area of Operations planning & scheduling, Maintenance planning, and Project management.