

# Non-Vibrational Fault Analysis of Turbojet Engine Bearings by using Deep Neural Networks

Juvith Ghosh, Medha Mani

**ABSTRACT-** This paper depicts the implementation of deep neural networks in predicting common faults of the turbojet engine bearings by training the model with images and processing them by designing proper Deep Neural Network model apart from conventional vibration analysis methods, for faster detection of bearing health and reusability. The turbojet engines have higher main-shaft speeds operating at elevated temperature conditions, reducing the bearing estimated life and thus the need of schedule maintenance. This system can identify some of the bearing damages like cracks, dents, fatigue, fretting and smearing conditions prevailing due to thermal effects, high axial and radial loads over the main-shaft, propeller shank and auxiliary systems bearings. It finally assists the aircraft maintenance engineers and technicians to reach to the conclusions of bearing conditions by taking pictures of bearings from any device and fetching them to the system for better results of bearing conditions.

**KEYWORDS-** Bearings, Turbojet Engines, Fault Analysis, Deep Neural Networks

## I. INTRODUCTION

Bearings are one of the principle components in machineries that are used to reduce friction of rotating elements and shafts. They work on the principle of rolling contact friction being lesser than the sliding friction principle. There are various classes of bearings found in the machines depending upon the needs and choices vary on factors of operating conditions like temperature, speed of the shaft or heavy radial and axial loads [1]. Major classification of bearings is done on basis of types of moving elements broadly into ball and roller types. There are three primary components of bearings that are involved in the whole assembly and namely the outer race, Inner race, rolling elements that are either balls or rollers of various dimensions. According to bearing manufacturers,

every bearing had a life of operation and needs replacement once it is out of life. But there are also measures that are to be taken care off to increase the operating life of bearings. Proper lubrication, interference of rolling elements, clearance between moving shaft and races, use of bearing housing and maintaining proper loads predominantly increases bearing life. Schedule maintenance of bearings are thus required as turbojet engines experiences the stresses of operation with sudden variation of shaft speeds at higher temperatures inside engine chambers due to combustion of fuels. It is common to have effect of heat and loads over transmission section coupled with the bearings. Continuing working at such adverse situation causes thermal cracks and fatigue whereas at higher speeds, smearing is observed. Improper lubrication and maintenance causes fretting corrosion in bearing races and rolling elements.

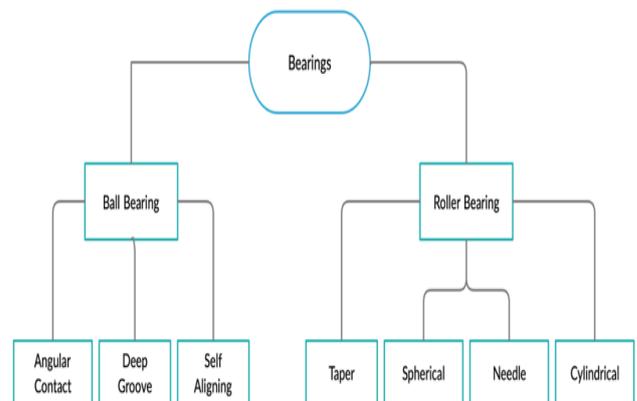


Fig 1: Classification of basic bearing types

- Deep Groove ball bearings can carry smaller to medium loads with high operating speeds and less noise. They can bear moderate radial and thrust loads with lesser shaft misalignments. Commonly used as fan bearings for cooling systems.
- Angular Contact ball bearings have axially asymmetric races. They can bear high axial thrust with moderate radial loads, mostly used in centrifugal pumps and with transmission units.
- Self-Aligning ball bearings could compensate for angular misalignment of  $2.5^\circ$  but they cannot take moment loads. They have better good accuracy at higher speeds and are used in load high speed load carrying shafts.

Manuscript received July 21, 2020

**Juvith Ghosh**, Department of Sensors and Biomedical Technologies, School of Electronics Engineering (SENSE), Vellore Institute of Technology (VIT) – Vellore, Tamil Nadu, India (email: juvithghosh@gmail.com)

**Medha Mani**, Department of Communications Engineering, School of Electronics Engineering (SENSE), Vellore Institute of Technology (VIT) – Vellore, Tamil Nadu, India

- Cylindrical roller bearings have cylindrical type rollers which could carry large radial load but no axial loads. They are applied in high speed operations where there is no moment load. Best suited for free shaft expansion and can tolerate misalignment of  $3.5^\circ$ . These bearings are used in gearboxes and in wheels.
- Spherical roller bearings have barrel shaped rollers, capable of carrying large radial and axial loads with self-alignment of  $2^\circ$  and are generally used in heavy industries.
- Taper roller bearings can carry large loads but have limitations of operating at low speeds. They have tapered type rollers to compensate larger loads at axial directions. These are used in gearboxes and auxiliary systems where shafts operate at low speeds with greater loads.
- Needle roller bearings have ability to carry oscillating loads due to the needle shaped rollers which have less radial space. They are also called as floating end bearings and cannot carry heavy axial loads. Application includes axles of automotive and in steering systems.

## II. CLASSIFICATION OF BEARING DEFECTS AND CAUSES

There are several factors that causes defects in bearings and each of them have specific reasons. Some of them dependent upon exhaustion of working life and some due to operating in adverse situations as in the case of turbojet engine bearings. Being important components in machines, repairing of bearings are generally unacceptable tasks.

### A. Cracks

The most popular cause of cracks is thermal cracking in turbojet bearings due to high temperature, causing the bearings to cross the thermal stress limits and inadequate material expansions. Heating can also be due to excessive lubrication or very high speed of operation. Low quantity of lubricants also causes over heating which may lead to thermal cracks [5] in bearings. It is visible as a part of loss of coloration of moving elements to steel blue and formation of deep crack lines of inner and outer races. Continuing working at such conditions can also lead to annealing of elements of where it is observed as jointed structures due to reduced material hardness.



Fig 2: Images of bearings with thermal cracks

### B. Fretting

It is a surface-initiated corrosion that results in false brinelling of bearings [6]. Visible as a bright marking with brown colorations, they are elliptical wear regions around balls and rollers and point towards the direction of the races. This type of corrosion is caused due to friction and

vibration around shafts due to loosely fittings. Fretting can be prevented by applying anti corrosion wear solutions over the races and moving elements or fitting the bearing with zero clearances.



Fig 3: Fretting corrosion in bearings

### C. Smearing

Occurs due to insufficient lubrication and heavy loads. In this type of corrosion, material from one region gets peeled off and gets deposited at another region. Due to high material temperature resulting from insufficient lubrication [7], the particles get hardened over at deposited surfaces. It is visible as solid patches over the surface and can easily be identified as uneven structures.

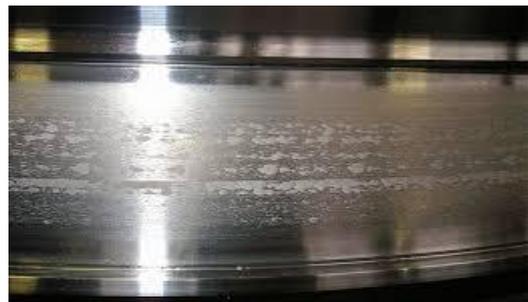


Fig 4: Smearing marks on bearing outer races

### D. Flaking

One of the immediate causes of bearing fracture is high operating loads, crossing the load limit of the bearing material. These are found to occur over housing, inner and outer races or cages supporting the bearing. It can easily be differentiated as flaking is a massive deformation of shape [8] and the only solution is remodeling of bearing by altering the races or other damaged portions.



Fig 5: Damaged bearing race caused by flaking

### E. Dents

They are also known as true brinelling and occurs when there is excess axial and radial loads over the raceways of bearings and crossing the elastic limit of the bearing material. It is observed as dotted indentations which further increases the vibrational noise during operation [9]. They are also caused due to static loads while installation

like hammering or improper handling during mounting. Precautions to be taken while installation to avoid dents.



Fig 6: Dentation marks over bearings

### F. Fatigue

It is also known as Spalling of bearings and occurs over the outer and inner races as well as to the moving elements where there is removal of the material in form of small particles. This results in loss of material from bearing surfaces making the material prone to more friction and vibrations further resulting in increased noise during operation. One of the primary reasons found to cause spalling is large axial or radial load at one direction over for longer period. Fatigue failure can be cured after redesigning the races and damaged sections [10]. Bearings should have calculated fatigue life before implementation to avoid failure.



Fig 7: Material deposition on bearing outer race due to fatigue

## III. METHODOLOGY

A Deep Neural Network or Deep learning is a subdivision of machine learning that comes broadly under Artificial Intelligence. The design of neural networks was inspired from the human brain and the pattern of arrangement of biological neurons [11]. Neurons learn from experience and again train themselves from the gained experience, similar concept was adopted for image classification and regression. The model considers the image datasets of bearings with various faults and trains the hidden layers about the differences to form the basis of classification. Datasets for experimental purpose was divided into training and test along with validation sets in a particularly in ratio of 70:15:15 of total images, the more the number of images the more accurate the training will be due to more experience from learning. Deep learning is widely used for image classification due to their training accuracy and minimum losses due to error. It also provides faster classification output once the training is completed. There are several layers of deep neural networks which train the system to predict the complex input tasks. These layers are based on statistical approach of mathematics and acquires

intelligence by back propagation training of neurons. The layers used for the experimental purposes have been described below with equations to compute the predicted results from each hidden layer and have a better understanding of principle behind deep learning.

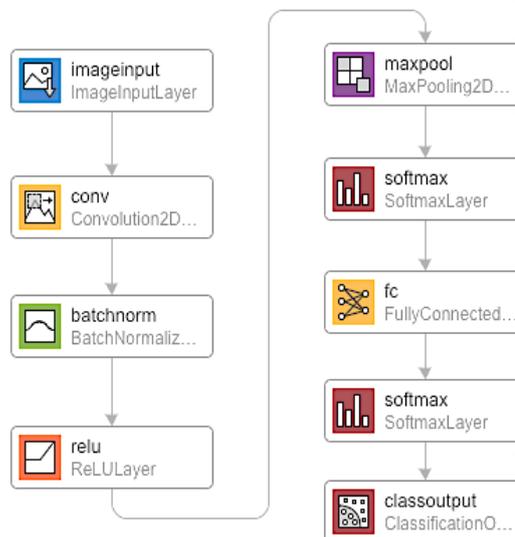


Fig 8: Layers of Deep Neural Network applied for the experimental purpose

### A. Data Pre-processing and Image Input Layer

Before the data is passed through the deep neural networks for processing, we need to overcome preliminary steps. This includes creating datasets [12] from the raw collected images. Random images have varying dimensions that could lead the network to get overfitted or underfitted resulting in delayed training and decreased accuracy. The collection of images is subdivided into folders according to their characteristics and a special function for augmented dataset is created. Augmented datasets are then passed through the image input layer where the image dimensions are specified. The experiment is carried out for the image dimensions of 128x128x3 of color images, so this augmented layer function is specified for 128x128x3 input images and thus adjusts the images to this provided dimensional information.

### B. Convolutional Layer

A convolutional layer is applied once the augmented dataset of images is created. This layer does not contain weights like other neural layers rather it has filters to obtain the important image features [13] like height, width, depth and zero padding along with the filtering characters. For a specific 3x3 image which is to be scanned, this layer sets stride dimension further determining the step with which the layer scans over the specific image. The experimental purpose was set for a stride of (1 1), zero padding with a total of 96 filters performing convolution of 3x3x3 as filter dimensions for the datasets. The output characters of the filters can be given by the following equations:

$$\text{Depth } d_{out} = m, \text{ where } m = \text{number of filters} \quad (1)$$

$$\text{Width } w_{out} = \left\lfloor \frac{w_{in}-f+2p}{s} \right\rfloor + 1 \quad (2)$$

$$\text{Height } h_{out} = \left\lfloor \frac{h_{in}-f+2p}{s} \right\rfloor + 1 \quad (3)$$

$$\text{Number of layer parameters} = md_{in}e^2 \quad (4)$$

$$\text{Total biases} = m \quad (5)$$

Where,

s= Stride dimension  
p=Padding value and  
f=filter dimension

### C. Batch Normalization Layer

After passing the images through convolutional layer, a batch normalization is used before the data is handed over to ReLU layer. One of the issues of deep neural network is that it takes longer time for training especially when the process is done without involving GPUs. This delay is mainly due to Internal Covariance Shift during training. The neurons of the network gains intelligence by learning through Back Propagation [14]. Using this back-propagation method, error estimations are done, and the neural network trains back their weights and biases to correct these errors. While correcting these errors, there exists an internal covariance shift in the network [15] which delays the whole process of training and the reason behind is that the previous layers had to learn about the corrections from beginning to predict the output for forward propagation. Normalization is a term used to specify the zero mean and unity standard deviation, here the normalize of data is in batches and hence the term batch normalization.

The mean ( $\mu$ ) of mini-batches  $B = [y_1, y_2, \dots, y_n]$  is given by:

$$\mu \leftarrow \frac{1}{n} \sum_{i=1}^n y_i \quad (6)$$

Where  $i = 1, 2, 3, \dots, n$ ,

$y_i$  = discrete batches

Batch variance ( $\sigma$ ) can be calculated as:

$$\sigma \leftarrow \frac{1}{n} \sum_{i=1}^n (y_i - \mu)^2 \quad (7)$$

The Normalization after taking out the mean and variance is estimated by the following formula:

$$\hat{y}_i \leftarrow \frac{y_i - \mu}{\sqrt{\sigma^2 + \epsilon}} \quad (8)$$

### D. ReLU Layer

Rectified linear Unit (ReLU) layer is used for thresholding operations. This layer is kept below the batch normalization and after the convolution of images are completed. It is a function which states the output to zero when a value of the input is less than zero. Basically, a ReLU layer improves the performance of the modelled network to some extent in operations involving image datasets. It is a solution to the vanishing gradient issues [16] related with back propagation. Vanishing gradient occurs during back propagation when the hidden layers

behind is unable to adjust the weights and as a result the training about the hidden layers is improper. The ReLU function can be stated as:

$$g(y) = \begin{cases} y, & y \geq 0 \\ 0, & y < 0 \end{cases} \quad (9)$$

### E. Max Pooling 2-D Layer

Pooling operations are used to sample down the image data by taking the height and width of the input images into consideration and finally reduces the image pixel density. This is done by choosing the adequate stride and padding parameters [17]. In the experiment, the padding value is kept at zero padding with stride dimension of [1 1]. The Max Pooling 2D layer benefits translation invariance of deep designed neural networks [18] by reduction the computational time and overfitting of the network as it selects the maximum value of pixel density and eliminating the minimum value pixel data. The formula can be given by:

$$M = \frac{D_x - P}{S} + 1 \quad (10)$$

Where,

M = Floor Value of function

S = Stride value

P = padding Value

$D_x$  = Dimension of image

### F. Fully Connected Layer

A fully connected layer connects every neuron with the previous neurons and thus makes the system to learn more from the connected hidden layers, thereby reducing the errors to the output by increasing the learnable [19]. Application of fully connected layer is general in image classification problems where the output is fetched to the Softmax layer for further processing.

### G. Softmax Layer

The fully connected layer is followed by the Softmax function. It is applicable when there is need to achieve the output in a probabilistic distribution [20] manner for the given input datasets. This is because the probability distribution provides the information about how perfectly the network is trained since the output of this layer provides probability vector where the strong probability gives the vector value close to unity and other weaker entries near to zero. It is mostly used for multiclass probabilities and the function can be given as:

$$f_s(x) = \frac{e^{a_s(x)}}{\sum_{j=1}^k e^{a_j(x)}} \quad (11)$$

Where  $0 \leq f_s \leq 1$  and  $\sum_{j=1}^k y_j = 1$

### H. Classification Layer

Softmax layers brings about Cross-Entropy losses, this loss is estimated by the classification layer and for multiclass situations [21] where the classes are mutually exclusive. Thus, the output of the Softmax layer is finally fed into this classification layer. Cross-Entropy loss can be computed by using the following equations:

$$\Delta = \sum_{i=1}^M \sum_{j=1}^N f_{ij} \ln y_{ij} \quad (12)$$

Where,

M = Number of samples

N = Number of classes

$Y_{ij}$  = output for class  $j^{\text{th}}$  for sample  $i^{\text{th}}$

Table 1: Designed Network Layer Parameters

Sl. No.	Name of Layer	Activations	Learnable
01	Image Input	128x128x3	-
02	Convolution	128x128x96	Weights 3x3x3x96 Bias 1x1x96
03	Batch Normalization	128x128x96	Offset 1x1x96 Scale 1x1x96
04	ReLU	128x128x96	-
05	Max Pooling	124x124x96	-
06	Softmax	124x124x96	-
07	Fully Connected	1x1x10	Weights 10x1476096 Bias 10x1
08	Softmax	1x1x10	-
09	Classification	-	-

### I. Training and Testing

The training of the network begins after deciding the type of the solver. In this experiment, we have used Stochastic Gradient Descent method [21] that divides the data into mini batches and updates the weights shortly after computing the errors. Updated weight is estimated using the following calculations involving the delta rule:

$$\Delta w_{mn} = \alpha e_m x_n \quad (13)$$

Where,

$\alpha$  = rate of learning

$x_n$  = output for the input node n

$e_m$  = error from output node m

The process is passed through 30 epochs with single iteration per epoch where epoch is the number of times the augmented dataset is passed through every designed neural network layer once. For the experimental purpose, the augmented Training datasets was kept at the ratio of 70 % where the Testing and Validation is equally divided with 15% each. This is called as the 70-30 rule and is commonly used in neural network training.

Table 2: Distribution of Datasets

	Fault Type in Bearings	Training Data (70%)	Testing Data (15%)	Validation Data (15%)
01	Cracks	53	08	08
02	Fretting	37	05	05
03	Smearing	51	07	07
04	Flaking	24	04	04
05	Dents	32	05	05
06	Fatigue	25	03	03

### IV. RESULTS AND DISCUSSIONS

The classification of images can be shown once after the completion of training process when all the epochs are iterated. While the training is in process, the observation of losses and accuracy need to be carefully observed. The more the number of datasets, more time it takes to complete the training and more accuracy the system will achieve. One can use a GPU with greater RAM and clock speed for shortening the training period [22].

The elapsed training time for the experiment was 35minutes and 43 seconds with single iteration per epochs. The number of epochs were kept at 30 for a learning rate of 0.01. Final iteration was reached with the validation accuracy of 77.45% and loss encountered was nearby unity on the single CPU hardware. This accuracy can be increased for increased number of datasets for validation and training.

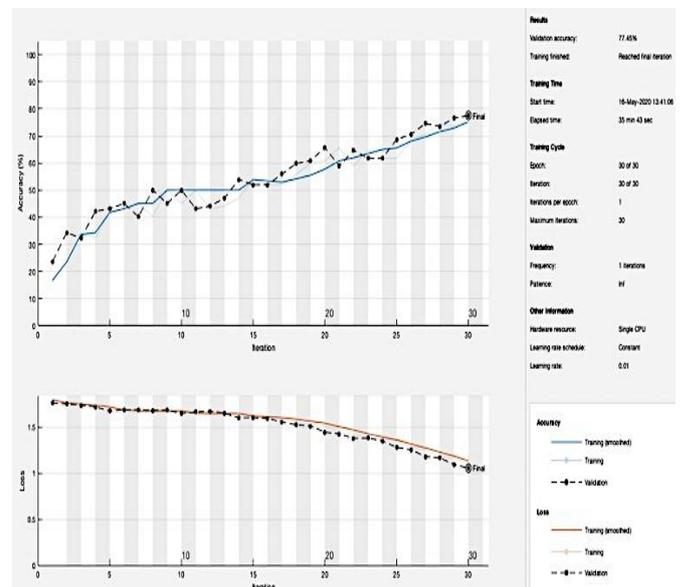


Fig 9: Graph depicting training progress along with losses and accuracy per epoch

The above graph shows the progress during training with the validation data for all the epochs. Accuracy is smoothed along the line of validation line marked by black line. It tells about the fitting of the network and one can predict whether it is overfitted, underfitted or normally fitted network. Losses per epoch is also with at par of validation datasets and decreases as the time elapses for iterations, depicting the learning approach by back propagation technique whereby increasing the error correction rate for the hidden layers.



Fig 10: Images showing the matching of test and training datasets

The above picture was plotted after training the network to show the results and how accurately it can predict the faults in bearing. In left column are the input images that were fetched to the system for testing bearing faults and the output from the system were shown in right column. First row matches dents with the datasets containing dents while training and the second row matches with the crack datasets. Similarly, other faults can also be detected by taking random input of damaged bearings.

$$accuracy = \frac{T}{T+F} \quad (14)$$

Where,

T = True Values and F = False Values

## V. CONCLUSION

From the above results it has been concluded that with the invent of high resolution, portable and low-cost machine vision systems and with the evolutionary field of deep neural networks and artificial intelligence, testing of materials can be done more easily with near accurate results. The more we train the network, the more it becomes intelligent and provides better results. Moreover, these systems are helpful for technicians and engineers for speedy process of fault detection and predicting operating life of materials especially for turbojet engines used in warfare, than the conventional time-consuming methods involving bulky measuring equipment and risky material handling.

## ACKNOWLEDGMENT

I would like to acknowledge Mr. Subhadip Das of Steel Authority of India for briefing me about the common defects in bearings and for his kind motivation to develop a solution to detect the faults which is hassle free from the previous methods.

## REFERENCES

- [1] H. Shen, S. Li, D. Gu, H. Chang, Bearing defect inspection based on machine vision, *Measurement*. 45 (2012)719-733. <https://doi.org/10.1016/j.measurement.2011.12.018.B>.
- [2] L. Averbach, E.N. Bamberger, Analysis of Bearing Incidents in Aircraft Gas Turbine Mainshaft Bearings, *Tribology Transactions*. 34 (1991) 241-247. <https://doi.org/10.1080/1040200910808982032>.
- [3] B. Samanta, K.R. Al-Balushi, S.A. Al-Araimi, Artificial neural networks and support vector machines with genetic algorithm for bearing fault detection, *Engineering Applications of Artificial Intelligence*. 16 (2003)657-665. <https://doi.org/10.1016/j.engappai.2003.09.006>.
- [4] For New Technology Network, CAT. No. 8102 – III/E, NTN Aerospace Bearings. [https://www.ntnamericas.com/en/website/documents/brochures-and-literature/catalogs/aerospace\\_bearings\\_8102\\_III\\_lower.pdf](https://www.ntnamericas.com/en/website/documents/brochures-and-literature/catalogs/aerospace_bearings_8102_III_lower.pdf)
- [5] H. Heshmat, M.J. Tomaszewski, J.F. Walton II, Small Gas Turbine Engine operating with High-Temperature Foil Bearings, in: *Volume 5: Marine; Microturbine and Small Turbomachinery; oil and Gas Applications; Structures and Dynamics, Parts A and B*, ASMEDC, 2006. <https://doi.org/10.1115/gt2006-90791>.
- [6] Bearing Failure: Causes and Cures, 2001. Barden Precision Bearings.
- [7] J.R. Nygaard, M. Rawson, P. Danson, H.K.D.H. Bhadeshia, Bearing steel microstructures after aircraft gas turbine engine service, *Materials Science and Technology*.30(2014)1911-

		Confusion Matrix							
Output Class		Target Class							
		Cracks	Dents	Fatigue	Flaking	Fretting	Smearing		
Cracks	6 5.0%	2 1.7%	2 1.7%	0 0.0%	4 3.3%	2 1.7%	37.5% 62.5%		
Dents	3 2.5%	1 0.8%	1 0.8%	2 1.7%	1 0.8%	2 1.7%	10.0% 90.0%		
Fatigue	4 3.3%	3 2.5%	1 0.8%	1 0.8%	2 1.7%	4 3.3%	6.7% 93.3%		
Flaking	11 9.2%	4 3.3%	1 0.8%	3 2.5%	0 0.0%	5 4.2%	12.5% 87.5%		
Fretting	6 5.0%	5 4.2%	2 1.7%	0 0.0%	11 9.2%	10 8.3%	32.4% 67.6%		
Smearing	6 5.0%	0 0.0%	1 0.8%	1 0.8%	2 1.7%	11 9.2%	52.4% 47.6%		
		16.7% 83.3%	6.7% 93.3%	12.5% 87.5%	42.9% 57.1%	55.0% 45.0%	32.4% 67.6%	27.5% 72.5%	

Fig 11: Plotted Confusion Matrix

Confusion matrix have been plotted after training the network to show about the performance and accuracy of the designed network [23] and it is used for image classification problems. The table is divided into rows and columns where the x-axis shows target class or predicted class and y-axis shows the output class or true class. Errors are identified as True Negative, False Positive, True positive and False Negative [24]. True values are plotted in green whereas the False values in red colors with the percentage of trueness and falseness below digits. The digits show the event values of classes whereas the true event is for True Positive only and others are considered as no event.

- 1918.<https://doi.org/10.1179/1743284714y.0000000548>.
- [7] Bearing faults that can be detected with vibration monitoring. March 2011, Wilcoxon Research Inc., Meggitt.
- [8] K. Deák, I. Kocsis, A. Vámosi, Application of machine vision in manufacturing of bearings using ANN and SVM, in: Proceedings of the 9<sup>th</sup> International Conference on Applied Informatics, Volume 1, Eszterházy Károly College, 2015. <https://doi.org/10.14794/icai.9.2014.1.295>.
- [9] Rolling Bearing Damage, recognition of damage and bearing inspection.2001, Publ. No. WL 82 102/3 EA, FAG – Schaeffler Group Industrial.
- [10] V. Sze, Y.H. Chen, T.J. Yang, J.S. Emer, Efficient Processing of Deep neural Networks: A Tutorial and Survey, Proceedings of the IEEE. 105 (2017) 2295-2329.<https://doi.org/10.1109/jproc.2017.2761740>.
- [11] [dataset] Juvith Ghosh, 2020. Turbojet-Bearing, GitHubRepository.<https://github.com/Juvith/Turbojet-Bearing.git>
- [12] Nikhil B., Nicholas L., 2017. Fundamentals of Deep Learning, first ed. United States of America, O'Reilly.
- [13] Batch normalization in Neural Networks: <https://towardsdatascience.com/batch-normalization-in-neural-networks-1ac91516821c>.
- [14] Batch Normalization – Speed up Neural Network Training, Ilango R – Medium: <https://medium.com/@ilango100/batch-normalization-speed-up-neural-network-training-245e39a62f85>.
- [15] Mark Hudson B., Martin T.H., Howard B.D., Deep Learning Toolbox – User’s Guide, MATLAB 2018b. <https://www.mathworks.com/products/deep-learning.html>
- [16] Oliver. W.L., Neural Networks, lecture – 11, Colby College.<http://cs.colby.edu/courses/F19/cs343/lectures/lecture11/Lecture11Slides.pdf>
- [17] What are Max Pooling, Chris – MACHINECURVE: <https://www.machinecurve.com/index.php/2020/01/30/what-are-pooling-average-pooling-global-max-pooling-and-global-average-pooling/>
- [18] Fully Connected Layer: The brute force layer of a Machine Learning model, Surya Pratap Singh, OpenGenusIQ: <https://iq.opengenus.org/fully-connected-layer/>
- [19] Softmax Layer by DeepAI: <https://deepai.org/machine-learning-glossary-and-terms/softmax-layer>
- [20] Phil K. 2017. MATLAB Deep Learning with Machine Learning, Neural Networks and Artificial Intelligence, APress.
- [21] Vincent V., Andrew S., Mark Z.M., 2011. Improving the speed of neural networks on CPUs. Deep Learning and Unsupervised Feature Learning Workshop, NIPS, California.
- [22] A.M. HAY, The derivation of global estimates from a confusion matrix, International Journal of Remote Sensing.9(1998)1395-1398.<https://doi.org/10.1080/01431168808954945>.
- [23] Confusion Matrix in Machine Learning by GeeksforGeeks.<https://www.geeksforgeeks.org/confusion-matrix-machine-learning/>

## ABOUT THE AUTHORS

**Juvith Ghosh** is a Graduate Biomedical Engineer from Vellore Institute of Technology – Vellore, India. He has previously carried on research over Internet of Things and Servo Actuation mechanisms involving patient monitoring systems where his papers were published in IEEE International Conferences. He worked with KUKA Roboter GmbH as an Intern in India division and had knowledge over Robot Programming – I and Advanced KUKA Robot Programming – II, also while working as Intern at Caparo Engineering (UK) – India, gained experiences over Robotic Spot Welding of white-body automotive components, Formative Manufacturing and coding 6-Axis CNC Turn-Mill Centers with focus over Intelligent Manufacturing Systems.

**Medha Mani** is a Graduate ECE student from Vellore Institute of Technology – Vellore, India. She has previous experience on working with BHEL – Bengaluru as Summer Intern where she had completed projects over microcontroller setup. As a computer science enthusiast, she also worked with CDAC – Noida and participated in several hackathon competitions throughout India. Her active role in research encouraged her to publish papers in IEEE International Conferences.