Rafik Ahmad, et. al. International Journal of Engineering Research and Applications
www.ijera.com
ISSN: 2248-9622, Vol. 11, Issue 7, (Series-I) July 2021, pp. 01-05

RESEARCH ARTICLE OPEN ACCESS

Fault Detection using Image Classifier

Rafik Ahmad* Nasrullah** Vivek Kumar***
Marut Nandan Tripathi**** Shrikant Singh*****
Babu Banarsi Das Institute of Technology And Management
Lucknow, Uttar Pradesh, India

ABSTRACT:
The image classifier is a traditional medium of image operation, computer eyesight and machine training fields. Image classification is playing a very paramount role in automation in different sector. In this writing paper, we study the image recognition of faulty electrical elements(RLC) using deep learning. We apply TensorFlow teachable machine architecture with convolutional neural networks for this purpose. For testing of faults which are chosen from the Image dataset for the stratification purpose. The generated dataset of images of various test samples and standard normal samples of resistors, capacitors and inductors help in differentiating standard normal samples. The result reveal the potency of deep learning-based image classifier using TensorFlow teachable machines.

Keywords: CNN, AI, Image classifier, Object detection, Deep neural networks, TensorFlow, Teachable Machine, Dart, Flutter, Android Studio, Visual Studio, Model Training, Webpage, User Interface (UI), Image Processing, RLC Fault Detection, Visual Fault Detection.

Date of Submission: 20-06-2021 Date of Acceptance: 05-07-2021

I. INTRODUCTION:

Image classification organizes the dataset into groups and categories based on its feature characteristic. Image classification come to light to minimize the space between the computer eyesight and human perception by training the computer with the created dataset [3]. The regular practice used for image classifying is basic segment of the field of artificial intelligence (AI) formally called machine learning. Machine learning comprises feature extraction module that extracts the desired features from dataset such as design, pattern, shape, size and color etc[4]. The main limitation of machine learning is, while extracting features it can only extract a certain set of features from images and unable to extract other important differentiating desired features from the trained dataset.

Deep learning is used to explore the study of image classification. Some years back, deep learning techniques represented by convolutional neural networks (CNNs) [1], have been used for fault detection problems on image classification data with magnificent outcomes. As is accurate for most supervised learning techniques, the performance of a Convolutional neural network fault detector mostly rely on trained dataset, and post-classification regularization may increase performance of the fault detector. Mainly, an unpolluted CNN-type fault detector that perform well on artificial data may unable to perform well on training dataset. In this paper, we explore a resistor, inductor and capacitor fault detection workflow using CNN [12] and applying on a realistic fault detection model based on the image classification model.

In deep learning, we consider the neural networks that recognize the image based on its features. This is achieved for the building of a complete feature extraction model which is capable of solving the difficulties faced due to the conventional methods [3]. The feature extraction process of the blended model should be able to master how to extract the differentiating features from the training dataset of images precisely.

Object Detection is broadly used in several models along with detecting traffic mild, picture annotation, pastime recognition, face detection, face reputation, video object co-segmentation face detection, independent vehicles and pedestrians on streets [9].

TensorFlow's Object Detection API is a effective tool that can quickly permit anybody to construct and installation effective photograph reputation software. Object detection not solely consists of classifying and spotting gadgets in an image but additionally localizes those objects and attracts bounding packing containers around them[10]. This paper usually makes a speciality of detecting dangerous items like threatening gadgets. To ease item detection for threatening gadgets, we've got TensorFlow object detection API to train models and we have used the Faster R-CNN algorithm for implementation. The model is built on two classes of threatening Objects. The model is evaluated on check dataset for the six training of detecting threatening items.

1. Image classifier: Image classifier is a technique used for the systematic arrangement in bulk and categories based totally on its feature characteristics. Image class got here into play for decreasing the space among the computer imaginative and prescient and human imaginative by using trained the computer with the dataset[11]. The traditional strategies used for image classifying is part and piece of the field of artificial intelligence (AI) formally called machine learning. In image classification, we use feature extraction like RGB pixel value, Histogram of Oriented Gradient(HOG) and Supported Vector machine(SVM) to detect the fault in the resistor, inductor and capacitor[3]. Features extraction with the help of an image classifier identifies the change in geometry and chroma of the test samples in comparison with the standard samples. The below block diagram fig.no:1 shows the basic steps for the image classification process.

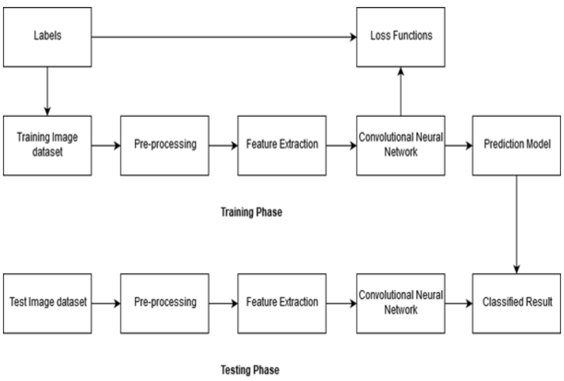

Fig.1: Block diagram of Image classifier

2. Object detection: Object detection is a feature extraction approach that works to perceive and point out items inside an photo or video. Mainly, object detection attracts square bounding boxes round these detected objects, which permit us to find wherein stated objects are in a given scene. Our model uses camera access to capture the images to find fault, the feature extraction process initiates and compares the standard sample with the percentage feature extracted from test samples[9]. The fig.2 shows the block diagram of object detection.

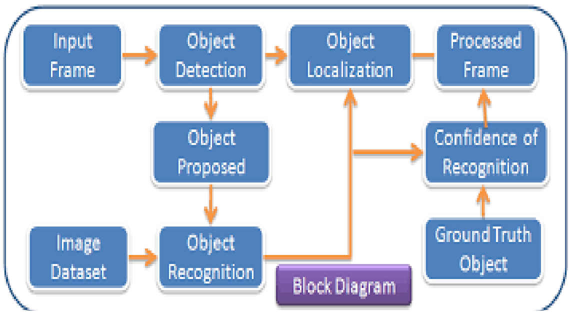

Fig.2: Block diagram of Object detection

3. Artificial Neural Network: Artificial Neural Networks-ANNs , which contemporarily are called "Deep Learning" ; are essentially computer packages loosely stimulated by means of the structure of the biological mind [5]. The mind is made from billions of cells known as neurons linked through synapses. New observations and stories not simplest regulate the power of those synaptic connections, however their accumulation drives convergence of the connections power ensuing in "learning "[6]. The fig.3 showing that how artificial neural network learn to give the desired output.

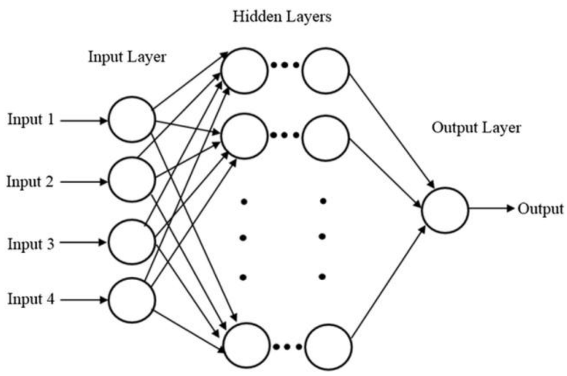

Fig.3: Showing the Artificial neural network logics.

4. Convolutional Neural Network: Convolutional Neural Network (*CNN*) is a class of deep neural networks[7] which is practically used to do image identification, object spotting, etc[8].. The fig.4 shows the Convolutional neural network block diagram. A convolution is essentially sliding a filter over the input. CNN identifies the faulty samples of the electrical element images uploaded by the user with its ability to find the difference. The processing of the image is done with the help of the dataset uploaded in the system. At the end of all processing, it give the output as normal or abnormal.

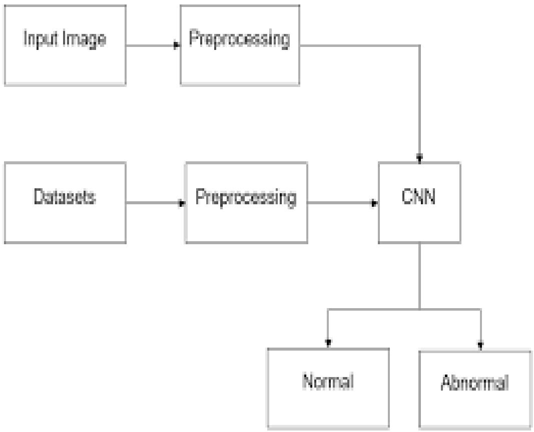

Fig.4: Block diagram of CNN

5. The software required for the model:

5.1. TensorFlow: TensorFlow is a non-proprietary software library for machine learning used around a range of tasks but has particular focus on training and inference of deep neural networks Teachable machine is a web-type tool that helps machine learning models to learn fast, easy and accessible to everyone [9]. It uses, gathers and assembles the examples into classes, or categories, that computer model needs to learn [10]. TensorFlow trains the model then directly tests it out to spot whether it had flawlessly classified latest examples and then we can export our model for our projects.

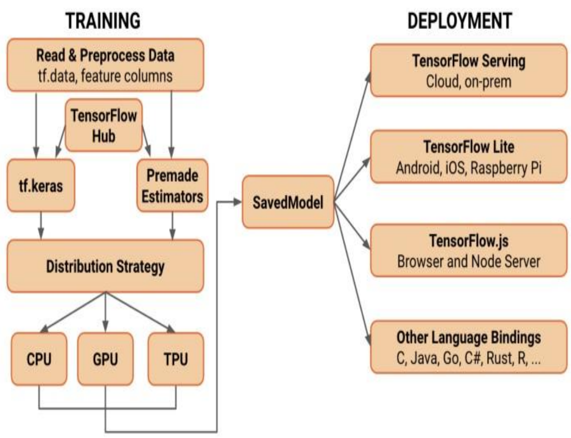

Fig.5: Training and development through TensorFlow

The above-mentioned fig.5 shows the classified image faults in the test dataset samples with the help of the TensorFlow platform.

5.2. Dart: Dart is a client optimized programming language for apps on numerous platforms. It used to build mobile, desktop, server, and web applications. It is an object oriented, class based, garbage collected language with C style syntax. Dart can compile to either native code or JavaScript.

5.3. Flutter: Flutter is a SDK(Software Development Kit) and portable UI(User Interface) tool kit for crafting appealing, natively compiled applications for mobile web and desktop from a single codebase. Flutter works with existing codes, is used by developers and institutes around the world, and is free and open source.

5.4. Android studio: Android studio is an official (IDE) Integrated Development Environment for android development. It is based on the IntelliJ IDEA, a kotlin or Java integrated development environment for software, and incorporates its code editing and developer tools.

5.5. Visual Studio: Visual Studio Code is a freeware source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for development operations like debugging, task running, and version control. It aims to provide just the tools a developer needs for a quick code-built debug cycle and leaves more complex workflows to feature IDEs, such as visual studio IDE.

5.6.Implementation: In this [9], Tensorflowlite gives us a pre-trained and optimized model to identify hundreds of classes of objects, including faults like resistors, inductors, capacitors etc.

The various steps for the implementation of the fault detection using Image classifier:

Step 1: Collecting Images

- Collecting images of faulty test samples and standard samples to train the dataset.

Step 2: Uploading the dataset for training

- Open the TensorFlow teachable machine website to create a class,ve the class a label, click on **upload>choose images from your files>select the images** that you have previously downloaded, repeat this step to detect the objects that you want [7].
- Now model can be trained by just clicking on the **Train Model** button.

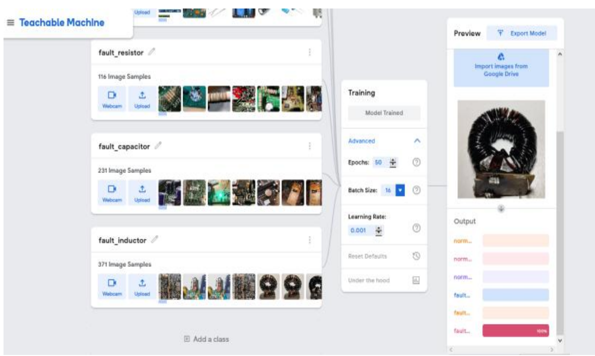

Fig.6: Uploading the dataset for training

Step 3: Exporting the trained model

- The trained model is tested by uploading an image and the result is displayed below.
- If everything is working well, now you can export the model. Click on the Export Model button and this popup will open, now select the TensorflowLite tab and click on Download my model. It will download a zip file for you.

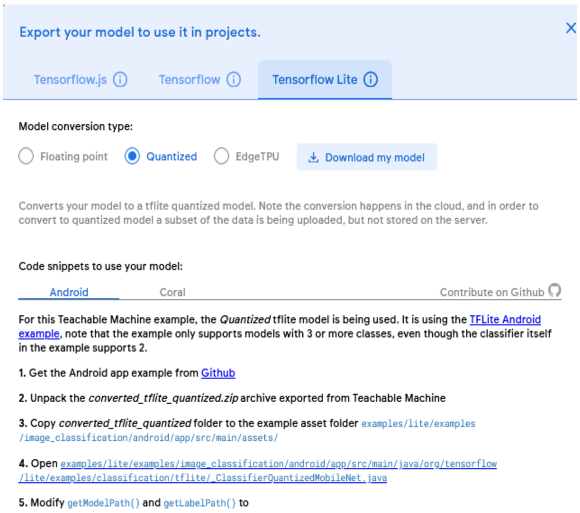

Fig.7: Exporting the trained model

Step 4: Creating the flutter project

- Clone this repository and install the dependencies by running the command in the terminal.

Step 5: Run the app and test it

Fig.8: The final result shown by the TensorFlow model

- Now simply you just have to run the app and the interface looks like this, obviously, this is a demo UI you can make your own tweaks and changes to make it shine.

II. CONCLUSION:

In this paper, a deep learning based fault detection using image classifier model has been discussed to detect electrical faults mainly in resistors, inductors, and capacitors. The model give real time possible cause of fault and possible method to improve or prevent that fault. It has user friendly User Interface (UI) to understand and detect the fault and with low specification and minimum use of RAM.

We have created a precised dataset to detect the fault by training and testing with the test samples. In this we can increase the dataset of images to amplify the accuracy and performance of the model.

REFERENCES:

[1]. **Santisudha, Panigrahi, Anuja Nanda, Tripti Swarnkar, (2018).** Deep Learning Approach for Image Classification. 2018 2nd International Conference on Data Science and Business Analytics (ICDSBA), IEEE 10.1109/ICDSBA.2018.00101, https://ieeexplore.ieee.org/document/8588970/authors.

[2]. **Chao Ma, Shuo Xu, Xianyong Yi, Linyi Li, Chenglong Yu (2020).** Research on Image Classification Method Based on DCNN. 2020 International Conference on Computer Engineering and Application (ICCEA). IEEE 10.1109/ICCEA50009.2020.00192, https://ieeexplore.ieee.org/document/9103855/authors#authors.

[3]. **Xin Jia, (2017).** Image recognition method based on deep learning. 2017 29th Chinese Control And Decision Conference (CCDC).IEEE 10.1109/CCDC.2017.7979332, https://ieeexplore.ieee.org/document/7979332/authors#authors.

[4]. **Mathieu Hatt, Chintan Parmar, Jinyi Qi, Issam El Naqa (2019).** Machine (Deep) Learning Methods for Image Processing and Radiomics. IEEE Transactions on Radiation and Plasma Medical Sciences (Volume: 3, Issue: 2, March 2019). IEEE, 10.1109/TRPMS.2019.2899538, https://ieeexplore.ieee.org/document/8657645/authors#authors.

[5]. **Mangmang Geng, Xiaojun Huang (2015).** The research and implementation of artificial intelligence in mobile applications. 2015 6th IEEE International Conference on Software

Engineering and Service Science (ICSESS). IEEE, 10.1109/ICSESS.2015.7339119, https://ieeexplore.ieee.org/document/7339119/authors#authors.

[6]. **Manish Mishra, Monika Srivastava (2014).** A view of Artificial Neural Network. 2014 International Conference on Advances in Engineering & Technology Research (ICAETR - 2014). IEEE, 10.1109/ICAETR.2014.7012785, https://ieeexplore.ieee.org/document/7012785/authors#authors.

[7]. **Hoo-Chang Shin; Holger R. Roth; Mingchen Gao; Le Lu; Ziyue Xu; Isabella Nogues; Jianhua Yao; Daniel Mollura; Ronald M. Summers(2016).** Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning. IEEE Transactions on Medical Imaging (Volume: 35, Issue: 5, May 2016).IEEE, 10.1109/TMI.2016.2528162, https://ieeexplore.ieee.org/document/7404017/authors#authors.

[8]. **Saad Albawi; Tareq Abed Mohammed; Saad Al-Zawi (2017).** Understanding of a convolutional neural network. International Conference on Engineering and Technology (ICET), 10.1109/ICEngTechnol.2017.8308186, IEEEhttps://ieeexplore.ieee.org/document/8308186.

[9]. **Rasika Phadnis; Jaya Mishra; Shruti Bendale(2018).** Objects Talk - Object Detection and Pattern Tracking Using TensorFlow. 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT). IEEE, 10.1109/ICICCT.2018.8473331, https://ieeexplore.ieee.org/document/8473331/authors#authors.

[10]. **Liping Yuan; Zhiyi Qu; Yufeng Zhao; Hongshuai Zhang; Qing Nian(2017).** A convolutional neural network based on TensorFlow for face recognition. IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC). 10.1109/IAEAC.2017.8054070, IEEE, https://ieeexplore.ieee.org/document/8054070

[11]. **Toto Haryanto; Ito Wasito; Heru Suhartanto(2017).** Convolutional Neural Network (CNN) for gland images classification. 2017 11th International Conference on Information & Communication Technology and System (ICTS). IEEE, 10.1109/ICTS.2017.8265646, https://ieeexplore.ieee.org/document/8265646.

[12]. **Jiudong Yang; Jianping Li (2017).** Application of deep convolution neural network. 2017 14th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP). IEEE, 10.1109/ICCWAMTIP.2017.8301485, https://ieeexplore.ieee.org/document/8301485.