International Journal of Management, IT & Engineering
Vol. 11 Issue 08, August 2021
ISSN: 2249-0558 Impact Factor: 7.119
Journal Homepage: http://www.ijmra.us, Email: editorijmie@gmail.com
Double-Blind Peer Reviewed Refereed Open Access International Journal - Included in the International Serial Directories Indexed & Listed at: Ulrich's Periodicals Directory ©, U.S.A., Open J-Gate as well as in Cabell's Directories of Publishing Opportunities, U.S.A

Artificial Intelligence Based Technique to Recognize and Separate Washers and Locating Pins

Kamran Rashid
M.Tech,Mechanical Engineering,Universal Institutions of Engineering and Technology, Bollapur, Lalru.

Rahul Kumar Mittal
Assistant Professor, Mechanical Engineering Department, Universal Institutions of Engineering and Technology, Bollapur, Lalru.

Abstract:
When the data and graphs of the London Metal Exchange are studied, stainless steel prices are obviously increasing. There are, therefore, significant problems faced by steel producers with pricing competitiveness and growing expenses. Waste reduction and improved productivity methods in the production and factory washing plugs are therefore necessary. It involves the adoption of automatic processing and classification of items and replacement of people with machines. The machines incorporated need to be smart to take decisions like humans and thus artificial intelligence makes iteasy. In this paper, we have proposed our study related to easy classification of items like washers and locating pins in the industries in order to save money and time. The system uses Convolutional Neural Network. The dataset is downloaded from the internet and same is used for training the model. Many models are made with different training parameters and the highest accuracy found was 96%. The testing phase involved the classification of image into washer and locating pin. Spyder software was used for simulation and language used is python.

Keywords:*Dataset, OpenCV, Convolutional Neural Network, Epoch, Batch Size.*

1. Introduction

Thanks to the rapid technological progress, mechanical engineering is often regarded as the fundamental field of regular living. However, the mechanical engineering-based technology has some drawbacks, such as the system inconsistency, since the mechanical system is defective. The AI can swap the input quickly and process it likewise promptly, so that the output is not impacted. It can also decide effectively on this flaw. The information system has to face more challenges throughout the whole process of changing the input into the output of mechanical engineering technology if the input information is more sophisticated.The ordinary system of information may err more than the manual system. Automated processing solutions based on AI were needed increasingly to overcome these shortcomings. When coupled with mechanical engineering, then mechanical engineering defects may easily be solved. Figure 1 shows the fundamental design and functionality of the AI-based system. At first, the data supplied by an end user were taken from the machine through a human-machine interface. Secondly, a reasoning module based on a machine initializes the rules for achieving the outcomes based on what is referred to as a positive reasoning.This approach provides the expert's opinion on the results, which is then redeemed in a database by a specified procedure or algorithm. Eventually, this technique is done to achieve virtually the same outcome. The similarity is then calculated on the basis of prior findings and calculations and then the mechanical defects are diagnosed more precisely.Convolutional neural network is one of the primary image recognition and classification categories in neural networks.Object detections, reconnaissance faces, etc. are certain areas of widespread usage of CNNs. CNN image classification analyses and classifies the picture into specified categories.

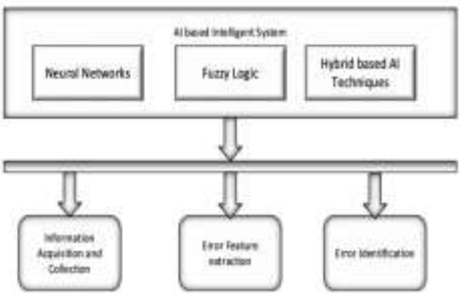

Fig 1: Artificial Intelligence architecture

2. Problem Statement

While packing the items in manufacturing industries, lot of expenses is wasted on the classification of the items and packing them in specific packets. Persons involved in this work are to be paid. Moreover, there is always an error or mistake probability related to humans. Thus, a novel system is proposed for classification of locating pins and washer accurately and with high reliability.

3. Literature Review

In [1], authors describe, analyse and define the prerequisites for fast, economic and sustained agricultural expansion. Existing technologies for creating ground-based sensors to monitor health and illness in plant have been proven in areas such as spectroscopy, pictures and volatile plant disease detection procedures.
Authors have analysed photos of the identification methods used for the diagnosis of plant illness, such as riber-stricken double-stranded acid, atomic acid and microscopy[2]. Various techniques of identifying computer-viewed plant diseases are being used.One of these is the diagnosis of diseases via color extraction shown in [3]. For diagnosing diseases from a wide range of sources, including camera flashes, which were unaffected by the sound, Color model YcbCr, HSI and CIELB were employed.
Further, formal removal of features might be used for plant disease detection. This technique has been utilized by Patil and Bodhe to calculate the leaf area and lesion zone triangle threshold with an average 98.60 percent for their final experiments[10].In addition, the texture extraction feature may be utilized to detect the plant condition. Patil and Kumar suggested the diagnosis model for plant diseases based on structured characteristics such as inertia, homogeneity, and the correlation of the grey matrix in the image [4]. The maize leaves were tested for pathogen in combination with colour extraction. Increased image quality and better classification are all robust. In [5], the authors gave a survey of the well-known conventional feature extraction approaches.The authors proposed this model, comprising input, output and one hidden layer[6], for the purpose of determining the illness of leaf species, pest species or disease in the propagation of the feedback of the neural network. They have developed a sort of pesticide or disease correcting software in farms.The author[7] also uses the characteristics of the Particle Swarm Optimization (PSO)[8] and forward neural networks to detect areas of the injured cotton floor and increase systems with accuracy by the ultimate overall accuracy of 95%.

4. Objectives

The objective of this research is to develop a Convolutional Neural Network based model that can accurately distinguish between locating pins and washers.

5. Hardware and Software Requirements

Hardware: Laptop for training and testing the CNN model.
Software:Anaconda, Spyder, Tensorflow, Keras, Numpy, Pytorch.

6. Convolutional Neural Network Model Design

6.1 Import dataset and load the libraries

In order to train our model, we import all the components required. For training the import modules utilized are:

ImageDataGenerator: ImageDataGenerator produces batches of image data tensors in real time. The generator output images are the same output as the input photos. Increased image data is a technology that is employed by introducing updated image copies in the data set to artificially enlarge the data size. Deep model training for neural networks with additional data can lead to improved models. The Keras neural network's profound learning library can tailor models to the ImageDataGenerator class.

Fig 2: Washer Images

Fig 3: Locating Pins Images

Conv2D: This layer generates a kernel that transforms a layer input into an output tensor. The image processing kernel contains the cooling matrix or mask which is used to blur a kernel or image via sharpening, taping, edge detection and so on.
MaxPooling2D: Max 2D space data pooling procedure. Procedure Take the maximum value for each dimension axis and download the input value through the pond size window. With steps in each dimension, the window is customizable.
Dropout: Every stage of exercise with overfit frequency is changed by the drop-off layer. The non-0 input is electrically boosted by 1/(1) such that the total remains unchanged throughout all the inputs. When the practice is set to True, only then does the drop-off layer apply, and hence no values are discarded. Training is automated if you use model.fit. We utilized the drop-off value 0.25 in our training model.
Dense: The normal neural network layer is dense. It is the layer that is most commonly utilized.
Flatten: There is a flattening layer between the convolutional layer and the fully linked layer. An interconnected neural network classifier is converted into a vector. An integrated categorization of neural networks is offered.
Early Stopping: The amount of time that can be used for neural networks training is a problem. The training data set might too frequently be over-adjusted, while few may result in a terrible model. Early stopping is a technique to set several training intervals and end the training when model performance is accomplished with a complete data package for validation. We utilized patience=10 in our train model (Number of epochs with no improvement after which training will be stopped).
Sequential: Deep learning models with the Keras Python module are easy and fast to construct. In the majority of scenarios, the sequential API can build layer by layer model. It is limited to prevent the creation of models for layer sharing or numerous inputs or outputs.
Batch Normalization: The load standardization of a deep neural learning network means that entries on a single layer are standardized. The batch standardization approach greatly speeds up the neural network training process and enhances the model by way of minor regulatory impact under specified scenarios. The inputs are converted to standard inputs, which means a zero and standard deviation.

```
from keras.preprocessing.image import ImageDataGenerator
from keras.layers import Conv2D
from keras.layers import MaxPooling2D
from keras.layers import Dropout
from keras.layers import Dense
from keras.layers import Flatten

from keras.callbacks import EarlyStopping, ModelCheckpoint
from keras.models import Sequential

from keras.layers import BatchNormalization
```

Fig 4: Code snippet of import modules

6.2 Data Preprocessing
When the picture data is pre-processed, it becomes meaningful floating point tensors, which are supplied into neural networks. The tensors serve as multidimensional arrays for storing data. The dimensions of a tensor with a 64x64 image with three channels (64, 64, 3)is now saved as JPEG files on a disc and permits the techniques to be followed.
6.3 Model Creation
The CNN model mainly includes concentration layers, concentration and decomposition and is fully integrated. It worked better for grid data, hence the picture classification problem was well dealt with by CNN. The drop layer might deactivate neural components and decrease the fitness of the model during the exercise. The Adam Optimizer is compiled. To add layers in our model, we utilise the "add()" method.
Conv2D at the first two levels. Input pictures are condensed layers in two-dimensional matrices. In the first and second layers there are sixty-four nodes. Depending on the extent of the data collection, the number may be changed to more or less. We work really well with 64 and 32, thus for now we will remain to it.

The size of the kernel is our filter matrix. A kernel size 3 hence proposes a 3x3 matrix filter. See the first picture and the update introduction. Activating the layer is a layer function. We use the ReLU or Rectified Linear Activation for our first two layers. This function is successful in neural networks. The input form is likewise included in our initial layer. This is the form of each picture that you enter. There is a flattened layer between the Conv2D layers and the thick layer. Flatten is used as a link between thick and twisted layers. This type of layer is the thick layer that we employ for our output layer. Dense is a typical layer for neural networks in many instances. In our output layer, we have 10 nodes, one for every potential output (0-9). The 'Softmax' is enabled. Softmax adds up to 1 to interpret the result as probability. The model then predicts the options most commonly used.

6.4 Compiling the model

Our model must be built next. Three components are needed for model development: optimizer, loss and metrics. The optimizer manages the rate of learning. As our optmizer, we shall use 'adam'. In many circumstances, Adam is frequently an efficient optimizer. During training the adam optimizer modifies the learning rate. How quickly the best weights can be achieved influences the learning ratio. A lesser learning rate can increase weight accuracy, but calculating weights takes longer.For our loss function, we use categorical crossentropy. This is the usual categorization choice.

6.5 Fitting the model

Use the fit generator technology, which is identical to the data generator, to fit the model into data. First point is to create a Python generator that creates limitless lots of inputs and targets, as the data is created continually. Before saving the model, the Keras model must know how many samples the generator may draw. The steps for each epoch parameter will be responsible.

7. Methodology

The whole process, beginning with the collecting of photographs to testing the trained CNN model for categorization, has been split down into multiple parts.

7.1 Dataset

Appropriate dataset is required at all phases of object identification research from the training period through the performance evaluation of recognition algorithms. All photographs gathered for the dataset were taken using the DSLR camera. The photos were separated into two categories in the dataset. A class was shown as 'diseased.' A second class has been termed as "healthy."

After collecting the photographs, a python script with a comparative methodology was developed that removed all duplicated photograph. The script removes duplicates while comparing image info, names, sizes, and date. A database including 2000 training pictures and 589 validation pictures was created.

7.2 Data Pre-processing

In general, before providing photographs to the network, two preprocessing processes are necessary. First, the photographs will be resized to the CNN input layer. Sizes of 227-227 are very usual for AlexNet, 224-224 for DenseNet, ResNet and VGG. Second, photos must be standardized so that the model can converge unseen data more quickly and more generally. Further pre-treatment was taken into account. The colored images in the dataset are gray scaled because Convolutional Neural Network works better with grayscale images and time for training is also reduced.

7.3 Training Phase

Several times throughout the training stage, the intrinsic weights of the model are automatically modified. External variables like as training strategy, architecture, regularization techniques impact this training process.

There are two CNN training techniques, with scratch or transmission. A network of pre-trained images provides transmission learning (e.g. ImageNet and its 1.2 million photos in 1000 classes). This form of learning is made feasible by the fact that the earliest layers of CNNs are generically low-level qualities separate from classes. The network weights of previous workouts are utilized to change this in practice. Transfer learning allows us to use CNNs even when minimum training data are available, as is generally the case for agricultural disease detection.

This method helps to make the network more widely available, since millions were educated. In terms of computing, it also saves time and capacity. Transfer learning takes place in two ways: extracting and adjusting characteristics. The extraction technique involves keeping the weights of a pre-trained model intact. The integrations provided are also used to build a new classifier in the target group. The ideal adjustment involves using the weights of the pre-trained model to initiate and train in the target data set all or part of these weights. The decision between the one methodology and the other depends, in particular, on the proximity of both the source and objective datasets, but also on the magnitude of the target data. A large number of layers may increase the risk of excessive data.

Scratch training occurs when network weights are not inherited from a previous model but randomly launched. It requires a larger training set and there is no previous network training experience and hence the input information is needed to determine the network weight.

Nevertheless, this technology allows us to develop a problem-specific network design that can boost performance. For instance, architectures can be created to overcome challenges in order to manage more than 256 multi-scale colored channels or to merge many separate models. In our work, we are designing a CNN Model from scratch.

7.4 Architecture

Three important components for CNN are built: convergence layers, pooling layers and active functions (ReLUs). The number of layers used, their organisation and the creation of additional treatment units vary from one design to another and alter their features.

7.5 Training the Model

I trained four models on different Epochs, Batch size, Validation steps and steps per epoch. Each time the model formed had different accuracy value and other output values. Summary of the models is given below:

Model 1			
Parameters		**Resulted Model**	
Batch size	8	Loss	0.1678
Steps per epoch	128	Accuracy	0.9473
Epochs	216	Validation Loss	0.0024
Validation steps	16	Validation Accuracy	0.9922

Model 2			
Parameters		**Resulted Model**	
Batch size	8	Loss	0.2373
Steps per epoch	140	Accuracy	0.9223
Epochs	236	Validation Loss	0.0941
Validation steps	32	Validation Accuracy	0.9844

Model 3			
Parameters		**Resulted Model**	
Batch size	16	Loss	1.7922
Steps per epoch	156	Accuracy	0.1627
Epochs	230	Validation Loss	1.7881
Validation steps	20	Validation Accuracy	0.1844

Model 4			
Parameters		**Resulted Model**	
Batch size	8	Loss	1.7916
Steps per epoch	220	Accuracy	0.1676
Epochs	270	Validation Loss	1.7861
Validation steps	28	Validation Accuracy	0.1607

Fig 5: Trained Models

8. Results and Observations

After training many models, each trained with different parameters, testing was done. It was done by keeping some washers and locating pins images in a folder. The path of the folder was given in the "Testing" code and the images were fed to the model for classification. It was seen that out of four models, one had the accuracy of 96%.

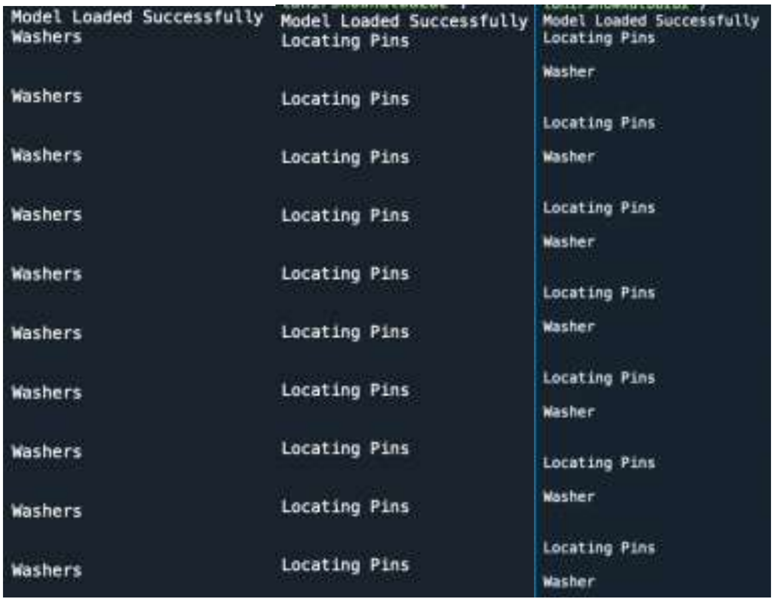

Fig 6: Testing Results

9. Conclusion

A precise diagnosis and classification of washers and locating pins is crucial for packaging as well as saving costs on labor. There is a list of ways to identify different items and in our classification study, we used CNN algorithms and image analytics. We trained four models with different epoch, batch size and other parameters. On testing the models we found the best one with accuracy 96%.

REFRENCES

[1] https:llwww.health24.comlMedicallAcne!About-acne!Acne-20120721, Accessed: 30 May, 2019 .
[2] Suva, M., "A Brief Review on Acne Vulgaris: Pathogenesis,Diagnosis, and Treatment", Research &Reviews: Journal ofPharmacology. Vol. 4, Issue 3,2020.
[3] https:/Iwww.medicalnewstoday.comlarticlesIl07l46.php.Accessed: 31 May, 2019.
[4] Taylor, M., Gonzalez, M. and Porter, R., "Pathways toInflammation: Acne Pathophysiology", Eur J Dermatol. 21(3),323-33p,2020.
[5] Vos, T. F. Years Lived with Disability (YLDs) for 1160 Sequelaeof 289 Diseases and Injuries 1990-2020: A Systematic Analysisfor the Global Burden of Disease Study 2020, Lancet, 380(9859):2l63-96p,20l2.208
[6] Razavian, N., Sontag, D., "Temporal Convolutional NeuralNetworks for Diagnosis from Lab Tests", Computer ScienceDepartment, New York University, 2020.
[7] Maroni , G., Ermidoro, M. and Previdi, F., "Automated Detection,Extraction and Counting of Acne Lesions for AutomaticEvaluation and Tracking of Acne Severity", Department ofManagement, Information and Production Engineering , 2017.
[8] Shen, X., Zhang, J., Yan, c., &Zhou, H. An AutomaticDiagnosis Method of Facial Acne Vulgaris Based onConvolutional Neural Network . Scientific reports, 8(1), 2021.
[9] Chantharaphaichit, T., Uyyanonvara, B., Sinthanayothin , c., andNishihara, A Automatic acne detection with featured Bayesianclassifier for medical treatment. Proceedings of the 3rd International Conference on Robotics, Informatics, andIntelligence Control Technology (RIIT20l5), 10-16,2020.
[10] DigitalePhotographie GmbH, "VISIA Complexion Analysis",http://www.visia-complexion-an alysis.comlvisia-complex ion-analysis.asp, 2020.
[11] Fujii, H., Yanagisawa, T., Mitsui, M., et al., "Extraction of acnelesion in acne patients from Multispectral Images," 30th AnnualInternational IEEE EMBS Conference, pp. 4078-4081, 2019.
[12] Alamdari, N., Alhashim, M., and Fazel-Rezai, R. "Detection andclassification of acne lesions in acne patients: a mobileapplication", 2016 IEEE International Conference on ElectroInformation Technology (EIT), 739-743, 2021.