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

Neural Network Based Image Classifier for Nuts and Bolts

Farman Yousuf Sheikh
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:

Especially in the mechanical automobile industry where nuts and bolts are produced, a system that recognises the nuts and bolts has to be designed. The major objective of this work is to establish a method for recognising the difference in the separation of nuts and bolts. The data collection of 1000 pictures of nuts and bolts is collected for processing this system and the model is trained using the Convolution Neural Network. Many models are trained at different times with different batch size and have been documented with a proportional loss of validation. The trained template is stored and a picture is then supplied to the model to be classified into a nut or a bolt.In the classification of the nuts and bolts, our model was 95% correct. The language used was python for coding and the simulation was done with anaconda software.

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

1. Introduction

Through human perception of identification, people may recognise any item by the natural logical thinking process. However, robots do not have the perception of identification in order to recognise an item as compared to human identification. Our research is therefore to create a technique to enhance the machine's detection efficiency. Classification of images is a task in which a label is assigned to a picture from a predetermined set of categories. This is one of the basic issues of Computer Vision, with a wide range of practical applications despite its simplicity. Our system receives a picture as an input when we conduct picture categorization.The system will now be aware of a number of categories and its objective is to assign a picture category. It may appear straightforward or simple, but for the computer to solve it is a really hard task. The computer sees a numerical grid, and not the image that we see or the way we perceive it. As we know. The pictures are a 3D array with integer sizes of Width x Height x 3 from 0 to 255. We use the Convolutional Neural Network to properly classify and reliably. Neural networks or CNNs are a family of deep neural education networks that represent an enormous leap in the recognition and categorization of images.

2. Problem Statement

Nuts and bolts account for almost 25 percent of the total sales in the heavy manufacturing machinery industry. One of their greatest issues is to classify these goods from photographs only, particularly if they are contradictory with the categories supplied by the brands. This problem can be overcome by computer vision that has attracted the gaze of a number of profound students.

3. Literature Review

The CNN is utilised in several tasks with remarkable performance in diverse applications. The neural networks are developed for various purposes. A initial application for CNN architecture was the acquaintance of hand-written digits[17]. The networks with the innovation of additional layers and the participation of different computer-vision techniques have constantly improved ever since the inception of CNN[18]. The ImageNet Challenge mainly uses revolutionary neural networks with different mixes of drawing datasets[19]. Few of the scientists demonstrated a comparison between the individual and the detection capabilities of a trained network using picture data sets.The comparative findings indicate a Human Rate of 73.1% on the dataset, whereas trained network results show a Rate of Precision of 64%[21]. Similarly, while applying the same dataset to Convolutional Neural Networks, 74.9 percent were accurate and therefore superior to human accuracy[21].
The strategies employed are mostly used to get a significantly improved precision rate by the strokes. In many scenarios, investigations are underway to understand the behaviour of the Deep Neural Network [20].These experiments show how the findings may be dramatically altered by simple modifications to a picture. The study also contains pictures that are completely unknown to people, yet categorised by trained networks with great accuracy rates[20]. In the field of function detectors and descriptors much advancements had place and several artefacts and approaches were created for categorization of objects and scenes. The

resemblance between detectors of objects, texture filters and filters is often praised. In object detection and scene categorization literature, there is a plethora of work[3].
The existing Felzenszwalb descriptions and context classifiers for Hoeim are primarily used by researchers[4]. The notion of creating different object detectors to understand pictures fundamentally resembles the work carried out inside a multi-media community in which several "semantic" ideas are used for annotation of photos and video and semanticisedindexation[22]. Every semanticised notion is taught in literature related to our job through either the picture or video frames. Therefore, with a number of crowded items in the scene, the method is tough to utilise and comprehend.The preceding approaches were detected and classified by a single item based on a human feature set. These approaches examine the relationship of items in the categorization scene[3].
Sanjeev S, Sannakkiet.a. (2013) uses neural networks to identify and categorise grape leaf diseases. In the first phase, authors have taken the step to obtain grape leaf photos and have taken the background using the threshold technique. Anitropic diffusion is used for picture pre-processing to preserve information on the affected section of the clustering technique for K-medium image segmentation.Meunkaewjinda et.al are known for the grape plant disease (2008). Colors of grapes were recognised to analyse concurrently the backbone neural network and the auto-organizing feature map. Segmented self-organize maps and genetic algorithms have been utilised for segmentation.
For categorisation, the supported vector machine is utilised. Following segmentation, the Gabor wavelet filter is used for preprocessing. SVM is again employed after segmentation [8].The cluster method for k-medoids was improved by Danyang Cao et al. (2010). All information was maintained in this CF-tree investigation and k-medoids have been used for clustering the clustering characteristics of CF-leaf nodes. From the root of the CF tree, K-clusters were found. Clusters quality and scalability are enhanced using the k-medoid method. [9].
Patil.S.P et.al, (2015)obtained images from the data set using the approach of imaging and extracts from the image many attributes, including texture, colour, shape and other aspects. They diagnosed the illness using neural background methods.The technique to automated leaf disease detection is created and the fertiliser necessary to treat this disease is produced by Neethu K.S, P. Vijay Ganesh et al. (2017). The authors used two lemon diseases and mango leaves. In order to compose the artificial neural network[11], K-Median method is applied.Ashwini Awate et al suggested a technique to the detection of grape, grape and pomegranate illnesses (2015). The authors employ image processing and neural networks for categorisingdiseases[13].

4. Hardware and Software Requirements

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

5. Proposed Methodology

The entire procedure is divided into many phases: from the generation of a new training dataset to the construction of a new CNN model, deep extraction of features for the training of a model and ultimately the classification of nut and bolts.

5.1 Dataset

A total of 1000 original RGB colored images of nuts and boltsarecollected from the open-source platform. A sample of dataset images isshowninFig.1. In all the work conducted in this paper, different sizesof images are used to evaluate the performance.The sizes of the images are 128 ×128, 256 ×256 and 512 ×512.

Fig1: Dataset with Nut and Bolt Images

5.2 Dataset Augmentation

Less training data is a main obstacle for constructing effective models of deep learning, especially CNN models for identification of nuts and bolts. In order to verify that the neural network model is resilient, we need greater data in order to enhance the model's functional diversity. We use the addition of the picture data to improve the dataset by a little distortion for this purpose. This increase in data makes the model more widespread. We employ 500 RGB originals and 500 healthy photographs to prepare the increased data. In addition, we rotate the pictures by −90◦,90◦, 180◦and 270◦, respectively in order to increase the dataset content. We also use flipping (up and down), shifting (horizontal and vertical), and scaling to enhance the data (by 0.6, 0.75 and 0.90). We employ OpenCV library to accomplish the aforesaid data enhancement approaches.

5.3 CNN-Based Nut and Bolt Recognition Model

We suggest a tailored CNN model for nut and bolt recognition. The model is conceived in ten levels of depth. This includes a layer of inputs, a layer of convolutions (Conv1), a max pooling layer (Pooling1), a layer of convolution (Conv2), a max layer of pools of layer 2 (Polling2), a layer of convolution layer (Conv3), a layer of pooling (Pooling3) and a layer of output (softmax) as illustrated in Fig. 2.

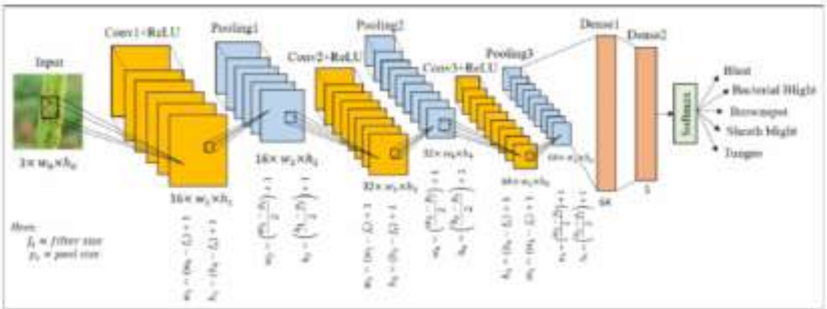

Fig 2: Novel CNN Model

5.3.1 Input Layer

Our model'sinput is fed by an RGB image of sizew0×h0, where w0is the width and h0is the height of the image, respectively.

5.3.2 Convolution Layer(s)

The main objective of a convolution layer is to recognise and transfer the local conjunctions of functions of the preceding layer to a feature map. We employ three convolution layers in our model, including many filters, to provide output maps. These maps therefore store information about where and how much the feature is installed inside the picture. Therefore each filter is educated spatially to identify specific characteristics of the picture of the nut/bolt related to its position in the volume to which it has been applied.We employ nonlinear ReLU activation function foreasy learning of complicated data connections. We employ three layers, namely Conv1, Conv2 and Conv3, in our model. In Conv1, Conv2 and Conv3, we utilise 16, 32 and 64 filters.

5.3.3 Pooling Layer(s)

Pooling plays a key role in our model by minimising variance and difficult calculations, resulting in fewer learning parameters. A downgrade process along with the spatial dimensions is performed and the dimensions of the characteristic map are reduced. In addition, it highlights the functionality that appears in a section of the convolution-layer map. The rest of the procedures are thus carried out using summarised features which make the model stronger in relation to changes in the position of the pictures of nuts and bolts. We employ 3 levels in our model, namely Pooling1, Pooling2 and Pooling3.Table 1illustrates the model summary.

Layers	Function	Filter/Pool	#Filters	Output	#parameters
Input	–	–	–	256 × 256	0
Conv1	Convolution	3 × 3	16	16 × 254 × 254	448
Pooling1	Max pooling	2 × 2	–	16 × 127 × 127	0
Conv2	Convolution	3 × 3	32	32 × 125 × 125	4640
Pooling2	Max pooling	2 × 2	–	32 × 62 × 62	0
Conv 3	Convolution	3 × 3	64	64 × 60 × 60	18496
Pooling3	Max pooling	2 × 2	–	64 × 30 × 30	0
Dense1	–	–	–	1×1 × 64	3686464
Dense2	–	–	–	1 × 1 × 5	325
Output	Softmax	–	–	1 × 1 × 5	0

Table 1: Proposed Model Summary

5.3.4 Dense Layer(s)

The output of the last max Pooling layer is flattened intoa uniform vector and then is fed into a fully connected dense layer. This layeroutputs a one-dimensional vector Mof size 64 which is input into second fullyconnected dense layer to produce a one-dimensional vector N of size 5.

5.3.5 Output (Softmax) Layer

The output layer uses the softmax activation function, which exponentially normalises the dense layer(s) output N and results in a probability distribution across the nuts and bolt classes.

5.4 Training the Model

In the model for our CNN identification of rice leaf disorders, our bespoke CNN models extract deep characteristics of rice leaf disorders. Activations on every layer of our CNN model turn details of the nut/bolt image into a more abstract image as the picture reaches the deeper levels of the model and summarises the essential aspects. The information is subsequently employed as functionality and classified by the Softmax layer of our model as deeper and more precision. In order to train our model, we pass pictures into batches to learn and optimise network confusion, pooling and dense layers in order to summarise these attributes into one vector of 1 to 64.These characteristics are then transferred to another thick layer to create a vector of 1×5. Finally, this vector is sent into the softmax layer to categorise the picture of a rice leaf disease into the appropriate class. We pass the training photographs a number of times called epochs and use the validation photos to confirm the model and related parameters. As a loss function for our model, we employ "Categorical Cross-Entropy." Our models are enhanced by having two classes of training pictures and constraining the output of the softmax layer into two labels to conform to the binary classification problem.

Model 1			
Parameters		**Resulted Model**	
Batch size	16	Loss	0.8776
Steps per epoch	50	Accuracy	0.7903
Epochs	2	Validation Loss	1.3283
Validation steps	100	Validation Accuracy	0.4985

Model 2			
Parameters		**Resulted Model**	
Batch size	16	Loss	0.2932
Steps per epoch	60	Accuracy	0.8912
Epochs	4	Validation Loss	0.9422
Validation steps	100	Validation Accuracy	0.5007

Model 3			
Parameters		**Resulted Model**	
Batch size	64	Loss	0.0231
Steps per epoch	110	Accuracy	0.9917
Epochs	20	Validation Loss	0.0587
Validation steps	150	Validation Accuracy	0.975

Model 4			
Parameters		**Resulted Model**	
Batch size	256	Loss	0.0221
Steps per epoch	200	Accuracy	0.9918
Epochs	80	Validation Loss	0.0115
Validation steps	150	Validation Accuracy	1

Table 1: Model Training

5.4.1 Effect of Epochs

The suggested CNN-based model is trained by a number of epochs up to 200. The optimum tuned period is 80, as no more improvements have been observed for our model in the training and validation accuracy.In general, greater the number of epochs, better the trained model.

6. Results and Observations

6.1 Experimental Settings

A confusion matrix is a table that is typically employed in a series of test data for which the real values of a classification model (or classifier) are known.

The assessment indexes may change based on the study's topic. Precision, recall and harmonic mean F1 score based on accuracy and recall are common assessment metrics.

$$Precision = \frac{TP}{TP + FP} \cdot 100\%,$$

$$Recall = \frac{TP}{TP + FN} \cdot 100\%.$$

In order to quantify model correctness, an F1 score is also added. The F1 score both takes the exactness and the reminder of the model into consideration. The formula is

$$\text{F1} = \frac{2 Precision \cdot Recall}{Precision + Recall} \cdot 100\%.$$

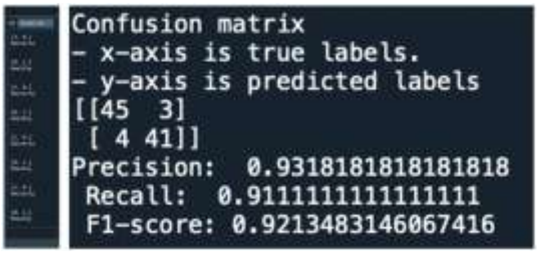

Fig 3: Model Testing

7. Conclusion

We have suggested in this study a customised CNN model that could be classified into a nut and bolt. On independent test photos our model achieves an accuracy of 92%. In addition, because of its decreased number of network parameters, our approach is effective for memory storage. We seek to increase the reliability and robustness of our model on various datasets from other places We may also work to classify nuts and bolts when background is not clear or in varied illumination.

REFRENCES

[1] T. Wang, “Ph.D. forum: Real-time lane-vehicle detection for advanced driver assistance on mobile devices,” in Proc. IEEE International Conference on Smart Computing, 2019.

[2] J. Arróspide and L. Salgado, “Video based vehicle detection and tracking for driver assistance systems,” Securitas Vialis, vol. 7, pp. 1-9, 2020.

[3] X. Chen, L. Liu, Y. Deng et al., “Vehicle detection based on visual attention mechanism and AdaBoost cascade classifier in intelligent transportation systems,” Optical and Quantum Electronics, 2019, vol. 51, no. 8, pp. 263.

[4] Y. Mao and P. Shi, “Noise reduction algorithm of vehicle detection in intelligent transportation system,” Intelligent Transportation Systems, 2018.

[5] C. Lv and T. Shan, “Multi-dimensional image edge localization method based on edge symmetry algorithm,” Multimedia Tools and Applications, pp. 1-15. , 2020

[6] S. S. Pillai and B. Radhakrishnan, “Night-Time vehicle detection using tail lights: A survey,” International Journal of Engineering Research and General Science, vol. 4, no. 2, 2020.

[7] Y. Xu, G. Yu, Y. Wang et al., “A hybrid vehicle detection method based on viola-jones and HOG+ SVM from UAV images,” Sensors, vol. 16, no. 8, p. 1325, 2019.

[8] A. Kurniawan, R. Saputra, M. Marzuki et al., “The implementation of object recognition using Deformable Part Model (DPM) with Latent SVM on lumen robot friend,” in Proc. International Conference on Engineering and Technology Development (ICETD), 2019.

[9] J. Redmon, S. Divvala, R. Girshick et al., “You only look once: Unified, real-time object detection,” in Proc. the IEEE Conference on Computer Vision and pattern Recognition, 2016, pp. 779-788.

[10] J. Redmon and A. Farhadi, “YOLO9000: Better, faster, stronger,” arXiv preprint, 2019.

[11] S. Ren, K. He, R. Girshick et al., “Faster r-CNN: Towards real-time object detection with region proposal networks,” Advances in Neural Information Processing Systems, pp. 91-99, 2020.

[12] Z. Dong, Y. Wu, M. Pei et al., “Vehicle type classification using a semi-supervised convolutional neural network,” IEEE Transactions on Intelligent Transportation Systems, vol. 16, no. 4, pp. 2247-2256, 2020.

[13] L. Wen, D. Du, Z. Cai et al., “UA-DETRAC: A new benchmark and protocol for multi-object detection and tracking,” ArXiv preprint arXiv:1511.04136, 2019.

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” Communications of the ACM, vol. 60, no. 2, 2021.

[15] K. He, X. Zhang, S. Ren et al., “Spatial pyramid pooling in deep convolutional networks for visual recognition,” in Proc. European Conference on Computer Vision, Springer, Cham, 2019, pp. 346-361.