Identification of Edible and Non-Edible Mushroom Through Convolution Neural Network

Devika G1,* Asha Gowda Karegowda2

1Government Engineering College, K R Pet, Karnataka, India, sgdevika@gmail.com
2Siddaganga Institute of Technology, Tumakuru, Karnataka, India,
*Corresponding author. Email: sgdevika@gmail.com

ABSTRACT
Mushroom is one among the most popular consumed food in India. In India people are cultivating mushroom as viable income source for their livelihood. Now-a-days deep learning is being applied to process big data and vision related applications. Recent smart devices can be utilized for automated edibility diagnosis of mushroom using deep convolution neural network (CNN) it has revealed a remarkable performance capability in all its sphere of research activities. DCNN works on static dataset. The models on which it applies will pose as well determine its requirement for training. This paper presents a classification tool for edibility detection of mushroom through deep CNN. Better performance is obtained by tuning the hyper-parameters and through adjustments in pooling combinations in order to obtain real time inference suitably. DCNN has been trained with a data set of segmentation as train and test sets. Performance is analyzed on sNet, Lenet, AlxNet, cNET network architectures. DCNN results are comparatively better in its performance.

Keywords: Bigdata, CNN, classification, DCNN, Mushroom,

1. INTRODUCTION
Mushrooms have origin from fungi family. A viewable sized fruit body produced by fungi is known as mushroom. They are not autotrophs with the absence of chlorophyll but through their enzymes can degrade complex substrate to get nutrition required for growth [1-3]. They are capable to flourish in all wide range of habitats from tropics to polar and from below to above soil. There is no typical method for classification of mushrooms. In general, they are categorized as edible or medicinal or poisonous mushrooms. Based on habitat mushrooms are categorized as temperate, tropical or sub-tropical mushrooms. On morphological features grouped as grilled or others (Cyphelloids, Gasteroids, Clavarioid, Polypore, stinkhorn, Morels, jelly) [4-5].

The very first crop detection algorithm was proposed in 1996 [6-7]. Infrared images were considered to segment crops and weeds using hysteresis umbral and the Min Neighbouring algorithm for crops planted row wise. New possibilities of solutions started to be adoptable for these similar problems with innovative advancements of ML. CNN based feature detection DBSCAN method applied using Harris corner detection technique in [8] and obtained an accuracy of 98%. Machine learning and deep learning methods are successfully adopted in recognition and classification of plant spices [9]. Indian government is also funding on artificial Intelligence related projects with projected annual GDP growth of 7-7.5% by 2022. The improvement’s in these technologies has eliminated traditional manual investigation and analysis of feature maps. The deep learning methods will automatically learn and adopt multi-layer processing approach for spatial and temporal data set. This paper is simple means to provide solution for examining mushroom edibility with CNN.

The specific objective of proposed work is to develop an automated image categorization as edible or non-edible mushroom applying deep CNN technique. Currently, CNN is at stage of state-of-art in image
recognition. CNN involves classification through learning for this reason large set of images are required to carry on experiment. The obtained images require preprocessing to yield with normalized image set having same size and shape as CNN expect pre specified input sized images. As images are provided as input location and odor cannot be used to classify mushroom. The model is capable of covertly extract features influential to activation function required from the input image through convolution layer. The problem is considered as simple binary class resulting in two feature mining as edible or non-edible.

This research studies on mushroom classification for CNN architectures like sNet, LeNet, AlexNet and cNet. sNet is scalable CNN includes three convolution followed by fully connected layer[10]. LeNet was proposed by Yann Lecun in 1998, it is a simple convolution neural network [11]. LeNet architecture consist of two sets of convolution and pooling layers, flattening convolutional layer, two fully connected layers and softmax classifier at end. AlexNet contains eight layers initial five are convolution and later three are fully connected layer [12]. CNet includes two convolutions and three fully connected layer [13].

This paper organized as follows; in section II a brief review on deep learning, CNN and its application for mushroom is discussed. The dataset and model description of CNN is included in section III. Proposed work is briefed in section IV followed by experiment discussion and conclusion in section V and VI respectively.

2. RELATED STUDIES

Deep learning (DL) is descendant of artificial neural network but includes deeper neural nets [14]. DL is an emerging new modern approach finding success in all fields where is being used through its potential applications [15]. The DL model includes more capabilities of real-life system which helps to design better trained model for classification accuracy [16]. The survey reveals that most of works apply convolution neural network with ANN with different levels of convolution layers for image related applications to resolve problem of classification, prediction, or identification [14, 17].

Leaf disease detection of thirteen different types of plants is predicted using CNN in [17]. CNN is widely applied in agricultural field for many purposes. GoogleNet applied for plant disease detection for 14 crops species in [18] and specifically for banana in [19]. For land cover classification is carried on in [14]. Crop type classification among wheat, maize, soybean, sunflower and sugar beet in [20], similar concept is applied for plant identification considering seven plants in [21] and 44 different plants in [22] for Alex net framework. Few works of CNN evident are corn crop yield estimation in [23], fruit counting of tomatoes [24], apples and oranges [Chen], mangoes, almonds and apples [25]. Cattle race classification is carried on using CNN among five races in [26].

Mushroom is having commercial value because of its nutritional and commercial benefits [1, 7]. Mushroom accounts to 0.14 million worldwide among these 7000 are edible but remaining 14000 are not easily recognizable as edible or poisonous [3, 6]. The machine learning supervised, semi-supervised or unsupervised methods are applied for classification problems in several research works. In [31] classification of mushroom is compared applying multi-layer preceptor (MLP) and base radical network (BRF) among them MLP results are better than BRF. MLP based forecasting tool is presented in [32]. Harvesting of mushroom using robot vision system applied to check damage through support vector machine (SVM). UCI repository for clustering with K-modes is applied in [30] to analyse different combination of attributes to group mushrooms as poisonous or edible. The Kaggle dataset is used but attributes considered is confusing and less so the accuracy is less. With usage of images in classification accuracies will increase as segmentation will get increased scale and minimize the negative error impacts significantly at large scales [33].

The mushroom Diagnosis Assistance system (MDAS) presented in [31-35], inculcates web application, unified database and mobile phone application. The comparative study is made between naive Bayes and decision tree (DT) algorithm for mushroom classification. SVM and Navie bayes algorithms are compared for classification of mushroom in [34] among techniques simulated, SVM ends in better results. Three algorithms decision tree algorithm, SVM, and ANN with clustering are compared in classification of mushroom in [36-38] among them ANN performs better. MLP model is used to create, predict and classify by conducting experiment on dataset of 8124 for training and testing [36], sigmoidal function is used as activation function. K-means, a traditional clustering algorithm is used to categorize dataset of mushroom [39-42]. DT and Navie Bayes are applied on application system named Mushroom Diagnosis Assistance System for classification in [42-46]. The same experiments of [1] are conducted on same dataset in [47-49] using Weka mining tool for interactive identification of
mushrooms. Multi-nominal problem of classifying problem to multiple categorizes with Naive Bayes, RIDOR and SMO algorithms are proposed in [50-53] amongst Naive Bayes gives best result. Zero, Naïve Bayes and Bayes net methods are applied to classify mushroom in [54,55] as poisonous or non-poisonous and Naive Bayes accuracy is good in simulated results.

3. MATERIALS AND METHODS

This section briefs on data preprocess and CNN algorithm which is used in classification in the proposed work.

3.1. Data retrieval and transformation

The images used for experimentation is collected from sources http://www.mushroom.world and http://archive.ics.uci.edu/ml/datasets/Mushroom. It consists of large data base including mushroom images with information as status of edibility. The data available is web scrapped because not directly accessible. In data set edibility is classified into seven classes as edible, edible and good, edible and excellent, not edible, edible when cooked, poisonous, lethally poisonous. Data is narrowed into binary task classification problem as edible and non-edible. The images are then disemboveled through procedure applied in Python scripts. Comparison was with metadata includes name, size and date. Few images were resized; all images are converted to .jpg and compressed. Considered images are colored and represented in 280X280 matrices with values between 0-255 but for our processing converted to values between 0-1. The images were then guaranteed with human verification multiple times after automated preprocessing. Dataset of certain images of mushroom is shown in figure 1.

![Figure 1 Sample Dataset of mushroom](image)

3.2. Convolution Neural Network

The basic layers of the CNN include the convolution, max-pooling and fully-connected (dense) layer [23, 24]. Typically, the network tends to learn better as the network gets deeper [24]. However, this may affect the computational time. Hence, we have carefully designed the network architecture which requires shorter computational time. The highest classification performance is obtained with parameters which are finely tuned during the training phase.

The convolutional layer convolves with the input signal using a kernel (window) [25, 26]. A feature map for the next layer is generated after the convolution. After which, the batch normalization layer is applied to normalize the input training data to flow between the intermediate layers. The purpose is to enable faster learning and boosting. Then, the rectified linear unit is applied to threshold the input data and reduced the redundancies in the data. To reduce the size of the feature map, the max-pooling layer is used. Finally, every neuron of the max-pooling layer is connected to every neuron in the fully-connected layer where the output predicts the outcome (normal or PD) of the input signal [27, 28, 29].

4. PROPOSED WORK

CNN architecture is preferably suitable for implementation of model which includes train and test among given dataset. In the proposed work classification of mushrooms into edible or non-edible is performed using CNN architecture and named as DCNN model as in previous study [13, 26]. Image dataset of mushrooms are classified as edible or non-edible applying machine learning algorithms such as sNet, LeNet, AlexNet, cNet...
and DCNN. Classification is performed using different CNN network architectures and evaluated for performance accuracy. The samples consist of 280X280 size, preprocessing and normalization for images will be performed. Later, the implementation of DNN model to predict is carried using open source tool Tensorflow. The detailed network architecture of DCNN model is presented in next sub-section.

4.1. Network Structure

![Preprocessed Dataset](image)

Table 1. sNet Network structure

<table>
<thead>
<tr>
<th>Feature</th>
<th>Convolution Layer 1</th>
<th>Convolution Layer 2</th>
<th>Convolution Layer 3</th>
<th>Fully connected layer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input Size</td>
<td>64X64</td>
<td>32X32</td>
<td>16X16</td>
<td>1080X1</td>
</tr>
<tr>
<td>Filter Size</td>
<td>8X8</td>
<td>4X4</td>
<td>4X4</td>
<td>1024</td>
</tr>
<tr>
<td>Activation Function</td>
<td>relu</td>
<td>relu</td>
<td>relu</td>
<td>softmax</td>
</tr>
</tbody>
</table>

Table 2. LeNet Network structure

<table>
<thead>
<tr>
<th>Feature</th>
<th>Convolution Layer 1</th>
<th>Convolution Layer 2</th>
<th>Fully connected layer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input Size</td>
<td>32X32</td>
<td>10X10</td>
<td>10284X1</td>
</tr>
<tr>
<td>Filter Size</td>
<td>5X5</td>
<td>2X2</td>
<td>1024</td>
</tr>
<tr>
<td>Activation Function</td>
<td>Tanh</td>
<td>Tanh</td>
<td>softmax</td>
</tr>
</tbody>
</table>
Table 3. cNet Network structure

<table>
<thead>
<tr>
<th>Feature</th>
<th>cNet</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Convolution Layer 1</td>
</tr>
<tr>
<td>Input Size</td>
<td>55X55</td>
</tr>
<tr>
<td>Filter Size</td>
<td>7X7</td>
</tr>
<tr>
<td>Activation Function</td>
<td>relu</td>
</tr>
</tbody>
</table>

Table 4. AlexNet Network structure

<table>
<thead>
<tr>
<th>Feature</th>
<th>AlexNet</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Convolution Layer 1</td>
</tr>
<tr>
<td>Input Size</td>
<td>64X64</td>
</tr>
<tr>
<td>Filter Size</td>
<td>11X11</td>
</tr>
<tr>
<td>Activation Fun.</td>
<td>relu</td>
</tr>
</tbody>
</table>

Table 5. DCNN Network structure

<table>
<thead>
<tr>
<th>Feature</th>
<th>DCNN</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Convolution layer 1</td>
</tr>
<tr>
<td>Input Size</td>
<td>280X280</td>
</tr>
<tr>
<td>Filter Size</td>
<td>3X3</td>
</tr>
<tr>
<td>Activation Function</td>
<td>relu</td>
</tr>
</tbody>
</table>

4.2. Design and Implementation

The model input is matrix form images values preprocessed as input range between 0 and 1. Input is assumed to be inputted as in eq 1. In the eq. 1 s is the sample index. Total number of samples is S in size (15400). N and M is total number of sweeps in one sample (280) and resolution channels captured bands (280) respectively. Features are extracted through CNN with combination of convolution and pooling layers. In model pooling convolution combination layer works as in eq. 2 for convolution layer 1. In eq. 2 W is activation function, c is number of convolution filters, b is weight and bias. Activation function is included with the aim to manage, transform, scale and output data range to train model. ReLu function is operated as in eq 3 for the activation function. It includes complex and non-linear functions. Similarly, if framework consists of nth convolution layer then output will be generated using eq 4. In the eq .4 total number of proposed CNN architecture is indicated with T. Extraction of feature in CNN is performed at two stages on using combination of convolution and pooling. Next feature extraction using dense fully functional layer. The dense fully functional layer is computed as in eq 5. In the eq 5 indicates leveling function to learn features extracted in convolution layer. The extraction of required attributes of proposed model is obtained using total fully functional layer of eq 6.In the eq 6, weight and bias parameters are taken into considered W and b respectively. The Es fully functional layer generates on predicted output from the S samples.

\[
X_s = [x_{i,j}, \ldots, \ldots, x_{N,M}] \quad i \in [1,N], j \in [1,M], s \in [1,S] \\
\text{cp}_1 = \text{pool} \left( \sigma(W_1^f \ast X_s + b_1^f) \right) \quad f \in [1,c_1] \\
\sigma(x) = \begin{cases} 
  x & \text{if } x > 0 \\
  0 & \text{Otherwise} 
\end{cases} \\
o_n = \text{pool} \left( \sigma(W_n^f \ast O_{n-1} + b_n^f) \right), \quad f \in [1,c_n], n \in [1,T - 1]
\]
\[ O^l = f(O_{n-1}) \]
\[ E_{fs} = W_N * \left( f\left( \text{pool}\left( \sigma(W^l_n * O_{n-1} + b^l_n) \right) \right) \right) + b_N \]

5. RESULT AND DISCUSSION

The classification of mushroom dataset is performed using CNN as mentioned. Proposed CNN model is developed using Keras in python. The classification is evaluated using accuracy, sensitivity and specificity evaluation parameters. Performance of classification is analysed through comparison in two stages. In first stage deep CNN, SVM, Navie Bayes, random forest, decision tree and KNN tested for classification of mushrooms. In second stage test, deep CNN, LeNet, cNet, sNet and AlexNet are compared. The result obtained is represented in figure 3.

The CNN is also known for its multiple network architectures. The different CNN framework DCNN, LeNet, cNet, sNet and AlexNet used in performance analysis. Classification process of these frameworks is tabulated in table 6. among the network architecture DCNN perform better than others.

The machine learning algorithms were compared for classification with dataset split of 70% test and 30% train and results are plotted as in figure 4. The comparison result is tabulated in table 7. In order to find better performance in each method best parameters are set accordingly. Deep CNN method result is better in terms of accuracy, sensitivity and specificity. Results of SVM, Naive Bayes, random forest and KNN are getting results next to DCNN. SVM and Naive Bayes are better than random forest, decision tree and KNN in results and training time required might be because of large dataset. Among all methods DNN and SVM took more training and computational time.

![Figure 3 Comparison results for edibility](image-url)

**Table 6. Network models Framework of Classification**

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Input size</th>
<th>Layer numbers</th>
<th>Number of parameters</th>
<th>Iterations</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>sNet</td>
<td>64x64</td>
<td>4</td>
<td>135872</td>
<td>100</td>
<td>80.4</td>
</tr>
<tr>
<td>LeNet</td>
<td>32X32</td>
<td>9</td>
<td>652500</td>
<td>100</td>
<td>86.48</td>
</tr>
<tr>
<td>AlexNet</td>
<td>64X64</td>
<td>11</td>
<td>20166688</td>
<td>100</td>
<td>93.86</td>
</tr>
<tr>
<td>cNet</td>
<td>64X64</td>
<td>8</td>
<td>6421568</td>
<td>100</td>
<td>96.86</td>
</tr>
<tr>
<td>DCNN</td>
<td>280x280</td>
<td>16</td>
<td>1659376</td>
<td>100</td>
<td>97.03</td>
</tr>
</tbody>
</table>
Figure 4 Classification results obtained in different models at each epoch.

Table 7: Comparison of classification results

<table>
<thead>
<tr>
<th>Method</th>
<th>DCNN</th>
<th>SVM</th>
<th>Naïve Bayes</th>
<th>Random forest</th>
<th>Decision tree</th>
<th>KNN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>0.93</td>
<td>0.66</td>
<td>0.57</td>
<td>0.34</td>
<td>0.27</td>
<td>0.14</td>
</tr>
<tr>
<td>Sensitivity</td>
<td>0.91</td>
<td>0.57</td>
<td>0.57</td>
<td>0.37</td>
<td>0.28</td>
<td>0.14</td>
</tr>
<tr>
<td>Specificity</td>
<td>0.90</td>
<td>0.66</td>
<td>0.6</td>
<td>0.4</td>
<td>0.27</td>
<td>0.14</td>
</tr>
</tbody>
</table>

6. CONCLUSION

In proposed work CNN model classifies mushroom either edible or not. The proposed method of DCNN proves to be better in classifying mushroom with an accuracy of 0.93. Along with this conclusion can be drawn that the depth of network and performance of classification not correlated so in that case with increase in complexity will not yield any better results. Further investigate on accuracy improvements can be tested through improved deep learning models. The optimal depth and size of filters can be researched.

REFERENCES


