JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

ACCURACY ENHANCEMENT OF FLOWER CLASSIFICATION USING MACHINE LEARNING

ANSHUL PAYASI, ASSISTANT PROFESSOR DHIRAJ RAJVAIDHYA

Department of computer science and engineering, Lakshmi Narain College of Technology Indore, Madhya Pradesh

ABSTRACT

Image classification is growing and becoming a trend among the recent year technology developers especially with the growth of data in different organization such as e- commerce, automotive, healthcare, and gaming. Image classification is the one of most critical use case in digital image analysis. The image classification problem is to categorize all the pixels of a digital image into one of the define class. Flower classification is a challenge task due to the wide range of flower species, which have a similar shape, appearance or surrounding objects such as leaves and grass. In this project proposes the classification of flower images using a powerful convolutional neural networks (CNN) algorithm. A flower image database with images is considered for the experimentation. The entire database is categorized into classes a CNN is used to image classification method. The CNN training is initiated in three classes and the validation is carried out on all the datasets and tested with flower image data to obtain better accuracy to flower classification.

KEYWORDS: Flower image classification, CNN (Convolutional neural network, ReLu activation function.

I. INTRODUCTION

Flower Classification is the arrangement of flowers into groups and categories for a clear understanding, proper study and effective organization. According to Radford (1986) classification is the arrangement of groups of plants with particular circumscriptions by rank and position according to artificial criteria, phenetic similarities, or phylogenetic relationships. A plant classification system has predictive value and provides an index to information storage and retrieval on that plant. Plant classification is groundwork in botanical studies, and in agro-forestry production and management. Plant taxonomy is a fundamental research field whose classification criteria rely heavily on the appearances of plants such as leaves, flowers, branches, bark, and fruits.

II. RESEARCH MOTIVATION

There has been much recent success in using ‘bag of features’ or ‘bag of visual words’ models for object and scene classification [1, 2,3]. In such methods the spatial organization of the features is not represented, only their frequency or occurrence is significant. Previous work dealing with object classification has focused on cases where the different object categories in general have little visual similarities (e.g. Caltech, 101), and models have tended to use off-the-shelf features (such as affineHarris [4] detectors with SIFT [5] descriptors). In this thesis we investigate whether a carefully honed visual vocabulary can support object classification for categories that have a

significant visual similarity (whilst still maintaining significant within-class variation). To this end we introduce a new dataset consisting of different flower species. Classifying flowers is a difficult task even for humans – certainly harder than discriminating a car from a bicycle from a human.

The cluttered backgrounds also makes the problem difficult as we risk classifying background content rather than the flower itself. Perhaps the greatest challenge arises from the intra-class vs inter-class variability, i.e. there is a smaller variation between images of different classes than within a class itself, and yet subtle differences between flowers determine their classification.

III. OVERVIEW AND PERFORMANCE MEASURE

In the rest of this thesis we develop a nearest neighbour classifier. The classifier involves a number of stages, starting with representing the three aspects as histograms of occurrences of visual words (a separate vocabulary is developed for each aspect); then combining the histograms into a single vocabulary. SIFT descriptors [6] on a regular grid are used to describe shape, HSV-values to describe colour, and MR-filters [6] to describe texture. Each is vector quantized to provide the visual words for that aspect. Each stage is separately optimized (in the manner of [4]).

CREATING A FLOWER VOCABULARY

Like botanists we need to be able to answer certain questions in order to classify flowers correctly. The more similar the flowers, the more questions that need to be answered. The flowering parts of a flower can be either petals, tepals or sepals. For simplicity we will refer to these as petals. The petals give crucial information about the species of a flower. Some flowers have petals with very distinctive shape, some have very distinctive colour, some have very characteristic texture patterns, and some are characterized by a combination of these properties. We want to create a vocabulary that gives an accurate representation of each of these properties. Flowers in images are often surrounded by greenery in the background. Hence, the background regions in images of two different flowers can be very similar. In order to avoid matching the green background region, rather than the desired foreground region, the image is segmented. The foreground and background RGB colour distributions are determined by labelling pixels in a subset of the training images as foreground (i.e. part of the flower), or background (i.e. part of the greenery).

FEATURE EXTRACTION

Deep learning expands the scope of artificial intelligence and has made great achievements in recent years, among which the deep learning model of convolutional neural network can realize the extraction of relevant features through convolution and achieve great performance improvement in large-scale image classification tasks.

The traditional image classification methods mainly extract the shape features, color features and texture features of the image. Using the manual feature extraction method, there will be difficulties in feature selection and insufficient feature extraction. Convolutional neural network has obvious advantages in the processing of input multidimensional signals, and it can be said that it is the first learning algorithm model of multilayer neural network successfully established and trained [1].

This thesis presents a convolutional neural network model based on different kinds of flower image screening, classification and recognition method. First, four kinds of flower data sets are established, and then the convolutional neural network is built. The convolutional neural network is used to initially extract small features, then select and extract main features, and then summarize the features of each part. Finally, the image classification is realized, and a GUI interface is implemented for using the trained network. In addition, four groups of comparative tests were set from two aspects of learning rate and data enhancement, so as to realize rapid and accurate recognition of different flower images.

IV. PROPOSED METHODOLOGY

PROPOSED METHODOLOGY

Convolutional Neural Networks (CNNs) are a specialized type of neural network primarily used for processing structured grid data like images. Here are the basics of CNNs:

1. Architecture:

Layers in CNNs:

Convolutional Layer: The core building block of a CNN, which applies a convolution operation to the input, passing the result to the next layer. It uses filters (kernels) that slide over the input data to produce feature maps.

Activation Function: Typically, the ReLU (Rectified Linear Unit) function is applied after each convolution operation to introduce non-linearity into the model.

Pooling Layer: Reduces the dimensionality of each feature map while retaining the most important information. Common pooling operations include max pooling and average pooling.

Fully Connected Layer (Dense Layer): Neurons in this layer have connections to all activations in the previous layer. This layer is typically used at the end of the network for classification tasks.

Output Layer: Produces the final output, often with a softmax activation function for classification tasks to output probabilities.

2. Convolution Operation:

Filters/Kernels: Small matrices that slide over the input data, performing element-wise multiplications and summing the results to produce a single output value. Multiple filters can be used to detect different features.

Stride: The step size with which the filter moves across the input matrix. A larger stride reduces the spatial dimensions of the output.

Padding: Adding extra pixels around the border of the input to control the spatial dimensions of the output. Common types include 'valid' (no padding) and 'same' (padding to keep output size same as input).

3. Pooling Operation:

Max Pooling: Takes the maximum value from a region of the feature map.

Average Pooling: Takes the average value from a region of the feature map.

Purpose: Reduces the spatial dimensions, leading to less computational cost and control overfitting by providing an abstracted form of representation.

4. Activation Functions:

ReLU (Rectified Linear Unit): Most commonly used, introduces non-linearity by outputting the input directly if positive, otherwise outputting zero.

Others: Sigmoid, Tanh, etc., but less commonly used in modern CNNs.

5. Training a CNN:

Forward Propagation: Input data is passed through the network layer by layer to obtain an output.

Loss Function: Measures the difference between the predicted output and the actual label.

Backpropagation: Computes gradients of the loss with respect to each parameter (filter weights) and updates them using an optimization algorithm like stochastic gradient descent (SGD) or Adam.

V. RESULT AND SIMULATION

LIBRARIES

The basics behind Python image processing libraries revolves around the manipulation and analysis of digital images. Digital image processing involves various techniques to enhance images, extract useful information, and prepare them for further analysis.

ANALYSIS

```
!pip install -q kaggle
```

```
[ ] from google.colab import files
    uploaded = files.upload()
```

```
Choose Files No file chosen  Upload widget is only available when the cell ha
Saving kaggle.json to kaggle.json
```

```
!mkdir -p ~/.kaggle
!cp kaggle.json ~/.kaggle/
!ls ~/.kaggle
!chmod 600/root/.kaggle/kaggle.json #set permission
```

```
kaggle.json
chmod: missing operand after '600/root/.kaggle/kaggle.json'
Try 'chmod --help' for more information.
```

Fig.5.1 Google colab.

```
# import system libs
import os
import time
import shutil
import pathlib
import itertools
from PIL import Image

# import data handling tools
import cv2
import numpy as np
import pandas as pd
import seaborn as sns
sns.set_style('darkgrid')
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix, classification_report
# import Deep learning Libraries
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras.models import Sequential
```

Fig.5.2 Libraries.

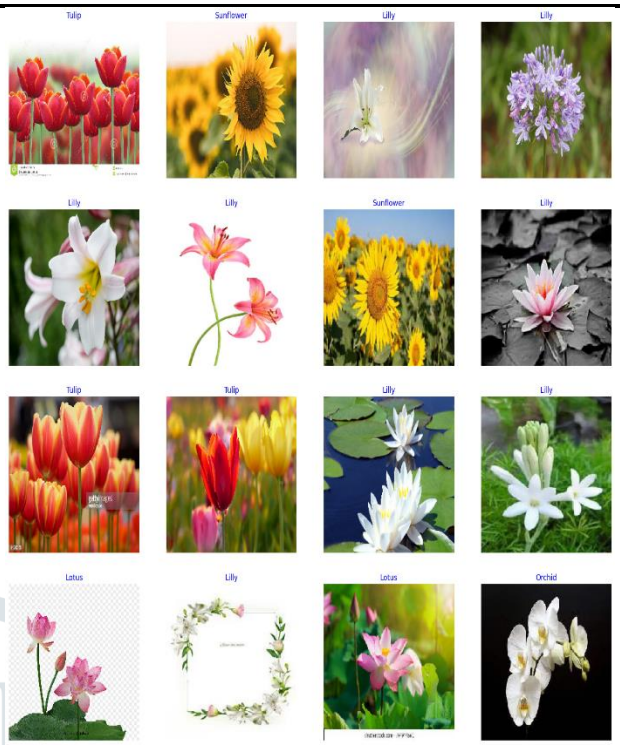

Fig.5.3 Flower image.

```
# Create Model Structure
img_size = (224, 224)
channels = 3
img_shape = (img_size[0], img_size[1], channels)
class_count = len(list(train_gen.class_indices.keys())) # to define number of classes in dense layer

# create pre-trained model (you can built on pretrained model such as :  efficientnet, VGG , Resnet )
# we will use efficientnetb3 from EfficientNet family.
base_model = tf.keras.applications.efficientnet.EfficientNetB3(include_top= False, weights= "imagenet", input_
# base_model.trainable = False
model = Sequential([
    base_model,
    BatchNormalization(axis= -1, momentum= 0.99, epsilon= 0.001),
    Dense(256, kernel_regularizer= regularizers.l2(l= 0.016), activity_regularizer= regularizers.l1(0.006),
              bias_regularizer= regularizers.l1(0.006), activation= 'relu'),
    Dropout(rate= 0.45, seed= 123),
    Dense(class_count, activation= 'softmax')
])

model.compile(Adamax(learning_rate= 0.001), loss= 'categorical_crossentropy', metrics= ['accuracy'])

model.summary()
```

Fig.5.4 Model Generation.

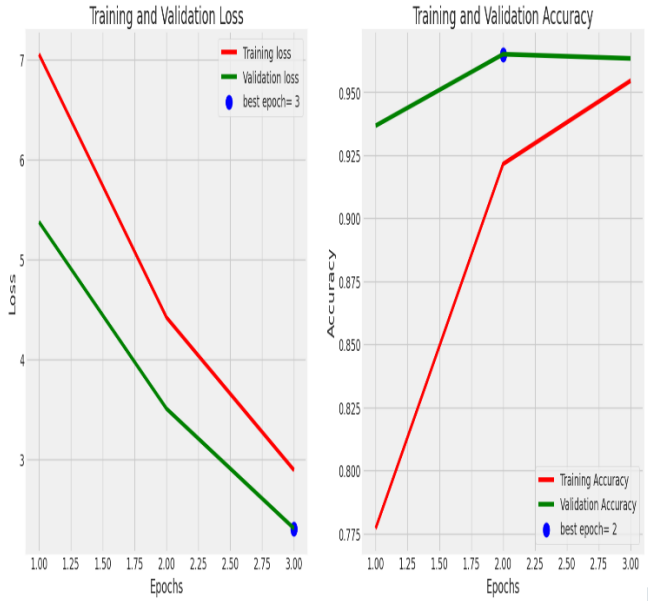

Fig.5.5 Loss and Accuracy Curve.

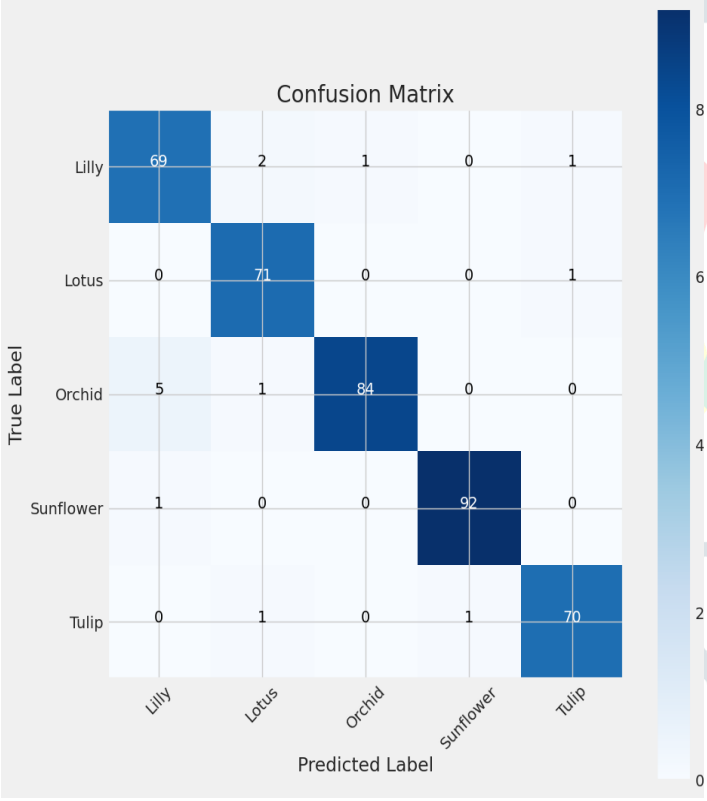

Fig.5.6 Confusion matrix.

COMPARISON BETWEEN PROPOSED AND PREVIOUS METHOD

Proposed Method	SVM Method
99.00%	91.00%

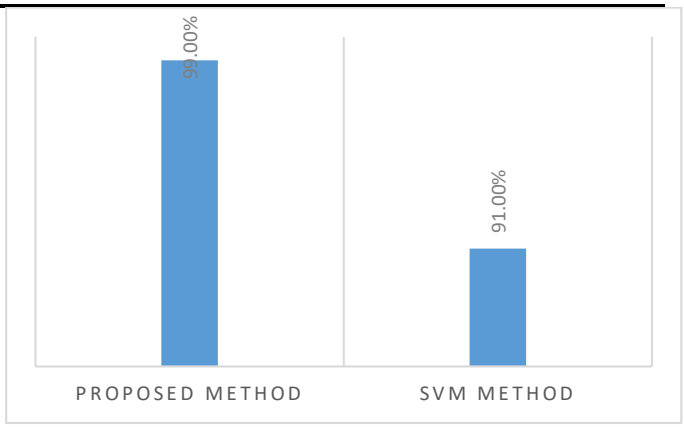

Fig.5.7 Comparison between proposed and previous method.

VI. CONCLUSION AND FUTURE SCOPE

CONCLUSION

The accuracy we obtained for cnn algorithm is more when compared to previous methods . This research endeavors to tackle the intricate task of Flower Species Classification through the application of advanced Machine Learning techniques. The journey commenced with the meticulous selection of the "Flowers Recognition" dataset,a pivotal decision underpinned by its diversity and suitability for the classification objective. Subsequent data preprocessing, involving image resizing, pixel value normalization, and augmentation, set the foundation for a robust and representative dataset. The experimental setup, characterized by a judicious division of data into training, validation, and test sets, laid the groundwork for model training and evaluation. The core of the research lies in the architectural design of a Convolutional Neural Network (CNN), a potenttoolfor image-related tasks.

Automated Species Identification: Improved algorithms can help botanists identify and catalog plant species more efficiently, aiding in biodiversity studies and conservation efforts.

Field Research: Mobile applications that use flower image classification can assist researchers in the field, providing instant identification and data collection.

REFERENCE

Base paper- Alavi-Siney, S. M., Saba, J., Siahpirani, A. F., & Nasiri, J. (2023). Supervised and unsupervised machine learning approaches for prediction and geographical discrimination of

Iranian saffron ecotypes based on flower-related and phytochemical attributes. *Information Processing in Agriculture*.

1. Kaur, R., Jain, A., & Kumar, S. (2022). Optimization classification of sunflower recognition through machine learning. *Materials Today: Proceedings*, *51*, 207-211.
2. Amarathunga, D. C., Ratnayake, M. N., Grundy, J., & Dorin, A. (2022). Fine-grained image classification of microscopic insect pest species: Western flower thrips and plague thrips. *Computers and Electronics in Agriculture*, *203*, 107462.
3. Zheng, Y., Zhang, T., & Fu, Y. (2022). A large-scale hyperspectral dataset for flower classification. *Knowledge-Based Systems*, *236*, 107647.
4. Zhang, C., Mouton, C., Valente, J., Kooistra, L., van Ooteghem, R., de Hoog, D., ... & de Jong, P. F. (2022). Automatic flower cluster estimation in apple orchards using aerial and ground based point clouds. *biosystems engineering*, *221*, 164-180.
5. Bhattarai, U., & Karkee, M. (2022). A weakly-supervised approach for flower/fruit counting in apple orchards. *Computers in Industry*, *138*, 103635.
6. Li, G., Fu, L., Gao, C., Fang, W., Zhao, G., Shi, F., ... & Cui, Y. (2022). Multi-class detection of kiwifruit flower and its distribution identification in orchard based on YOLOv5l and Euclidean distance. *Computers and Electronics in Agriculture*, *201*, 107342.
7. Batchuluun, G., Nam, S. H., & Park, K. R. (2022). Deep learning-based plant classification and crop disease classification by thermal camera. *Journal of King Saud University-Computer and Information Sciences*, *34*(10), 10474-10486.
8. Alavi-Siney, S. M., Saba, J., Siahpirani, A. F., & Nasiri, J. (2023). Supervised and unsupervised machine learning approaches for prediction and geographical discrimination of Iranian saffron ecotypes based on flower-related and phytochemical attributes. *Information Processing in Agriculture*.
9. Bataduwaarachchi, S. D., Sattarzadeh, A. R., Stewart, M., Ashcroft, B., Morrison, A., & North, S. (2023). Towards autonomous cross-pollination: Portable multi-classification system for in situ growth monitoring of tomato flowers. *Smart Agricultural Technology*, *4*, 100205.
10. Albarico, J. P., La Rosa, G. R. F., Santos, R. A. D., Tesorero, A. J. M., Magboo, M. S. A., & Magboo, V. P. C. (2023). Roses Greenhouse Cultivation Classification Using Machine Learning Techniques. *Procedia Computer Science*, *218*, 2163-2171.