9808:675X
HIGHLY CITED JOURNAL
ACCEPTANCE RATION BELOW: 8%

International Machine learning journal and Computer Engineering

Developing a Real-time Flower Detection Application using Deep Learning Techniques

Prof. Michael Davis

Affiliation: Machine Learning Lab, ETH Zurich

Abstract

In this paper, a novel flower detection application anchor-based method is proposed, which is combined with an attention mechanism to detect the flowers in a smart garden in AIoT more accurately and fast. While many researchers have paid much attention to the flower classification in existing studies, the issue of flower detection has been largely overlooked. The problem we have outlined deals largely with the study of a new design and application of flower detection. Firstly, a new end-to-end flower detection anchor-based method is inserted into the architecture of the network to make it more precious and fast and the loss function and attention mechanism are introduced into our model to suppress unimportant features. Secondly, our flower detection algorithms can be integrated into the mobile device. It is revealed that our flower detection method is very considerable through a series of investigations carried out. The detection accuracy of our method is similar to that of the state-of-the-art, and the detection speed is faster at the same time. It makes a major contribution to flower detection in computer vision.

Introduction

1. Introduction

In recent years, flower classification and detection has been of considerable interest to the computer vision community which can be applied in AIoT for the smart garden. The flowers in the smart garden can be automatically designed and recommended to make it more beautiful. The previous work mostly focused on flower classification [1–5] using a traditional detector and method [6, 7]. While it has become a tendency in flower classification and detection based on deep learning anchor-based approaches, flower detection was paid little attention. In most studies of deep learning anchor-based flower detection, the approaches can be divided into two general classes, two-stage object detection [8–10] and one-stage object detection [11–14].

1.1. Flower Classification and Detection Based on Deep Learning Two-Stage Approaches

As we all know, flower detection has become a hot topic in object detection since the convolutional neural network reborns worldwide in 2012. And mainstream flower detection was divided into two categories, one-stage detector approaches and two-stage detector approaches, which were based on anchor approaches. The essence of an anchor is the candidate boxes, which are designed with different scales and proportions and are classified by DNN. The positive anchor can learn how to return it to the right place, and it plays a role which is similar to the sliding window mechanism in traditional detection algorithms. In recent years, many researchers perform flower and fruit classification and detection based on convolutional neural network (CNN) approaches [15–18], which is a kind of feedforward neural network convolution computation contained and a deep structure. And it is one of the representative algorithms commonly used in deep learning. In 2014, Girshick et al. proposed a region with CNN characteristics (RCNN) of the first two-stage object detection (RCNN). And in 2015, they presented Fast RCNN, which enables us to train both the detector and the bounding box regressor in the same network configuration. Based on these models, in 2015, Ren et al. claimed a Faster RCNN detector, which is the first end-to-end and almost in a real-time deep learning detector (Faster RCNN) which proposal detection, feature extraction, bounding box regression, and so on have been gradually integrated into a unified end-to-end learning framework in Faster RCNN. So many researchers have applied them in flower and all kinds of fruit detection including all kinds of fruit like mangoes, almonds, and apples [19–22].

Great progress has been made in flower detection based on two-stage approaches in high accuracy. However, it is concluded that the speed of it needs to be increased.

1.2. Flower Classification and Detection Based on Deep Learning One-Stage Approaches

To overcome the above limitations, considering the speed of the running time, the other one-stage flower detection method was demonstrated. Redmon et al. in 2015 demonstrated the YOLO one-stage detector, which divides the image into several regions and predicts the boundary box and probability of each region. It has fast detection speed which can contribute to processing streaming media video in real time. Compared with other algorithms such as the two-stage detection method, it has half as many false backgrounds as them on account of capturing contextual information effectively and a strong versatility and generalization ability. Then, Redmon et al. made a series of improvements based on YOLO [12], including YOLOv2 [13] and YOLOv3 [11], which has the same accuracy as the two-stage detector at a higher speed. It is faster in YOLOv2 than other detection systems in a variety of monitoring data sets, besides the tradeoff between speed and accuracy can be made. As we all know, improving recall and positioning accuracy was focused on in YOLOv2, while maintaining classification accuracy. However, in YOLOv3, on the premise of maintaining the speed advantage, the prediction accuracy is improved; particularly, the recognition ability of small objects is strengthened. It adjusted the network structure and multiscale feature object detection method used and softmax utilized for object classification that counts. Therefore, many efforts were made to perform flower and fruit detection based on these algorithms of the YOLO detection method [23–25]. While its detection speed has been greatly improved compared with the two-stage detector, the positioning accuracy of it has been reduced, especially for some small objects. To overcome these limitations, Liu et al. in 2015 presented a Single Shot MultiBox Detector (SSD) to raise accuracy, especially small objects. The main contribution of SSD is the introduction of multireference and multiresolution detection technology to perform the detection only on its top layer. So many fruit and flower detection methods were based on SSD [26, 27], which achieved great success.

While great progress has been made in one-stage flower detection and has become a tendency in recent years, however, to obtain abundant image information and enhanced accuracy is still a challenge. Most of the studies were focused on accuracy or speed; however, few studies have considered the accuracy coming up with the speed of the object detection. So it is emphasized that the accuracy should be matched with the speed of the detector in our method. In conclusion, flower detection is based on one-stage approaches working well; the accuracy is still to be enhanced a little.

Although current anchor-based deep learning flower detection methods work well, they still suffer from the following six problems: (1) Due to the irregular shape of the flowers, the bounding box covers a great deal of nonflower regions, which caused a lot of interference. (2) The setting of anchor needs to be designed manually, and different designs are required for different flower datasets, which are quite troublesome and does not conform to the design idea of DNN. (3) The matching mechanism of an anchor makes the frequency of extreme scale (very large and very small object) to match lower than that of moderate size object. It is not easy for DNN to learn these extreme flower samples well when learning. (4) The large number of anchors causes serious imbalance. (5) On account of the unlabelled flower dataset, which only can be trained in the flower classification model, it cannot perform flower detection in a mobile device and AIoT. (6) Labelling the flower dataset consumes a lot of time and power, therefore most of the researchers were not able to pay attention to the flower detection which can be applied in a mobile device and AIoT.

In order to overcome the above limitations, a new object detection method based on new anchor-based approaches and a new-labelled flower dataset is adopted in this paper. To acquire more useful information and more precious object position in the image, attention mechanism SAM is utilized, which the output feature map of the channel attention module is taken as the input feature map of this module. In our paper, we can regard SAM as an attention mechanism to be applied to both channel and spatial dimensions and it can be embedded in most of the current mainstream networks and can improve the feature extraction capability of network models without significantly increasing the amount of computation and parameters. The baseline of our dataset is Oxford 102 Flower dataset, and we have labelled the dataset with all categories and all label and geometry positions to perform flower detection and other plant detection in a smart garden. Our backbone network is the CSPDarnet53 network, and it was designed to solve the previous work in the reasoning process which requires a lot of computation from the point of view of network structure. Also, SSP block is added in CSPDarknet53 and it significantly increases the acceptance field, extracts the most important contextual characteristics, and does little to slow down network operations. Our contributions are summarized as follows:(i)We present a flower detection method, an end-to-end deep convolutional neural network for flower detection applied in a smart garden in AIoT. The backbone network we applied is the CSPDarnet53 network, which can reduce computation when ensuring accuracy by integrating gradient changes into feature maps from end to end. And the running time of the flower detection in the architecture we used is the state-of-the-art fastest model compared with other models, especially when integrated into the mobile device. In order to extract the most important contextual characteristics and also increase the acceptance field, SSP block is added in the backbone network when keeping the network operations. Besides, attention mechanism SAM is utilized to select the information that is more critical to the current task target from the numerous information(ii)We labelled Oxford 102 Flower dataset with annotation containing 102 categories of flowers common in the UK, with each category containing 40 to 258 images, for a total of 8,189 images to perform flower detection. As we all know, the previous work attached much importance to the flower classification with an unannotated dataset. In our work, flower detection was paid much attention to with the annotation flower dataset which can be trained in our architecture. Also, the flower detection we proposed can be integrated into the mobile device to make it convenient to operate in a smart garden which can be applied in flower arrangement and flower horticulture. What is more, it can not only enhance the user's human-computer interaction experience of flower arrangement but save time and cost of horticulturists. It has preliminarily realized the concept of the smart garden in AIoT

The paper is organized as follows. Section 2 describes the materials and methods in our paper including the network architecture and the dataset we labelled. Section 3 describes our results and discussion. We conclude in Section 4.

2. Materials and Methods

The architecture of the flower detection system is shown as in Figure 1, when the flower picture is acquired by the mobile device which is based on 5G networks meaning that Internet of Everything; it is passed through our backbone network, the neck module with SPP and PAN and the module of YOLOv3 head. At last, the result of the flower picture can be obtained which can be applied in the smart garden in AIoT.

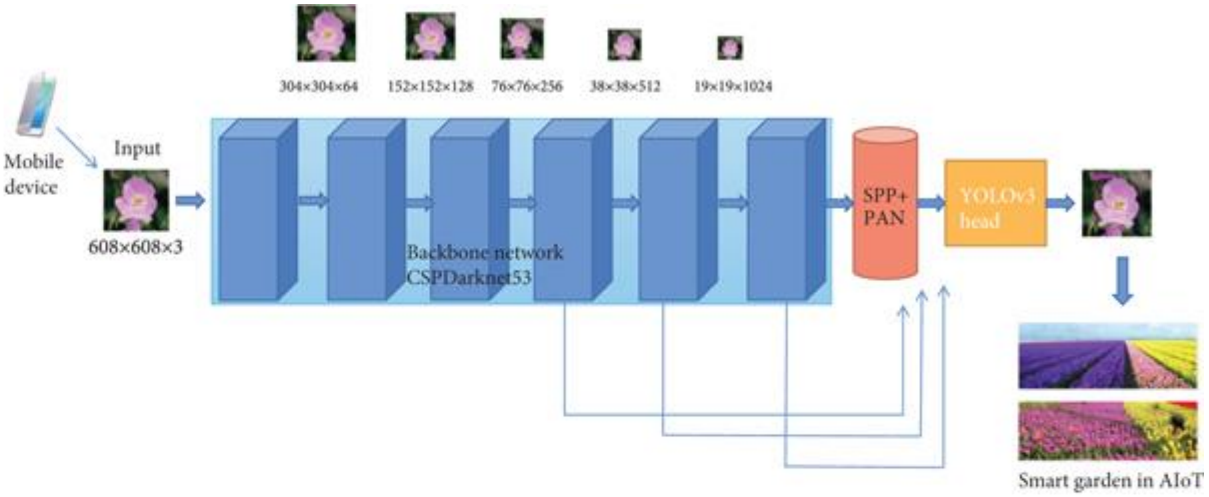


Figure 1

The architecture of our flower detection system.

2.1. Network Architecture

The architecture of our network is shown in Table 1, and the backbone network we utilized is the CSPdarknet53 network, combining CSPNet [28] and darknet53. The CSPNet was designed to solve the previous work in the reasoning process which requires a lot of computation from the point of view of network structure. It is considered that the problem of high inference computation is caused by the gradient information repetition in network optimization while CSPNet can reduce computation when ensuring accuracy by integrating gradient changes into feature maps from end to end. It not only can enhance the learning ability of the CNN but also can reduce computing bottlenecks and memory costs while keeping accuracy in the lightweight. It is a utilized idea ResNET for reference and adding a residue module to the Darknet53 network which can help to solve the deep network gradient problem. Two convolutional layers and one shortcut connection are contained in each residue module, and there are several duplicate residual modules in the layers. The pool layer and full connection layer were not involved in the architecture; the network undersampling is achieved by setting the stride of convolution as 2. After passing through this convolutional layer, the size of the image will be reduced to half. Convolution, BN, and Leaky Relu are contained in each convolutional layer, and a zero padding is added after each residual module.

The architecture of the backbone network CSPDarknet53.

It is studied that CSPDarkNet53 has advantages in object detection, which is better as the backbone of the test model. In CSPDarknet53, the parameters of CSPDarknet53 for image classification are 27.6 M which is bigger than other neural networks, and the receptive field size is also much bigger than neural networks. So it is a very suitable backbone for our proposed flower detection method.

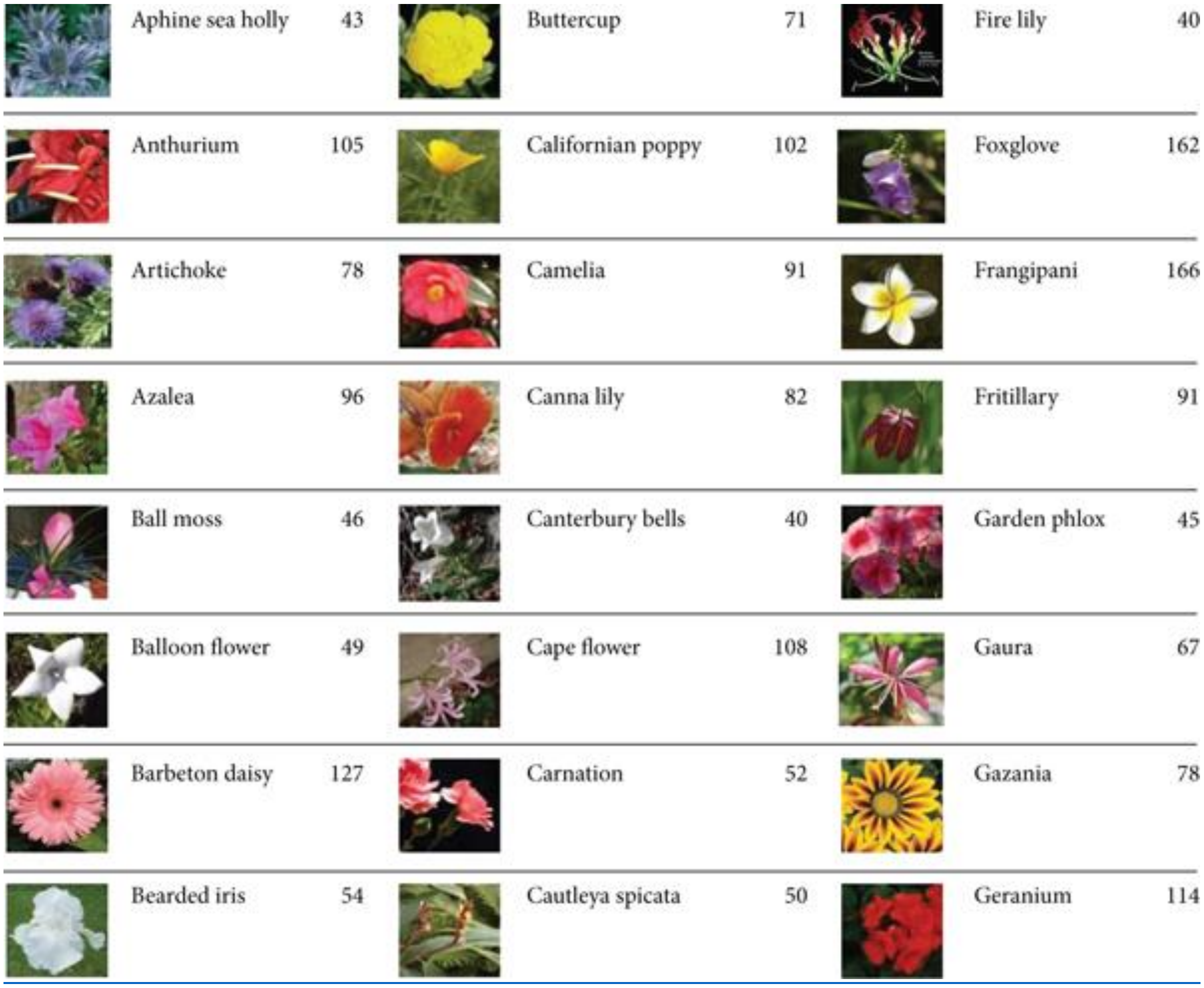
In addition, the SSP block is attached to the backbone network CSPDarknet53, which can produce an output of fixed size regardless of input size and can use different dimensions of the same image as input to get pooling features of the same length. When SPP is placed behind the last convolutional layer, it has no influence on the structure of the network and just replaces the original pooling layer. It can be used not only for image classification but also for object detection. When the SSP block is made use of in CSPDarknet53, it significantly increases the acceptance field, extracts the most important contextual characteristics, and does little to slow down network operations.

Also, PANET [29] was used instead of FPN in YOLOv3 as a parametric polymerization method for different bone levels for different detector levels. It is a bottom-up path enhancement that is aimed at facilitating the flow of information which can contribute to shortening the information path, enhancing the feature pyramid, and accurately locating the signal present at low levels to enhance the whole feature level. PANET developed adaptive feature pools to connect the feature grid to all the feature layers to enable useful information from each feature layer to propagate directly to the proposed subnetwork below.

YOLOv3 [11] is utilized in the head of the architecture, which is an anchor-based detection model. The residual network structure is used for reference to form a deeper network level and multiscale detection in the YOLOv3 model, which can improve mAP and small object detection effect.

2.2. Flower Dataset

As shown in Figure 2, Oxford 102 Flower dataset is a common flower dataset used in research and experiment when it comes to plant field. It contains 102 categories of flowers common in the UK, with each category containing 40 to 258 images, for a total of 8,189 images. Large proportions, gestures, and light changes are involved in the images which can be used for image classification studies. However, it cannot be used for flower detection when it comes to a smart garden in AIoT. Besides, the unlabelled flower dataset cannot be applied in a mobile device and other smart devices.



Aphine sea holly	43	Buttercup	71	Fire lily	40
Anthurium	105	Californian poppy	102	Foxglove	162
Artichoke	78	Camelia	91	Frangipani	166
Azalea	96	Canna lily	82	Fritillary	91
Ball moss	46	Canterbury bells	40	Garden phlox	45
Balloon flower	49	Cape flower	108	Gaura	67
Barbeton daisy	127	Carnation	52	Gazania	78
Bearded iris	54	Cautleya spicata	50	Geranium	114

Figure 2

Oxford 102 Flower dataset, which contains 102 categories of flowers, can be used for image classification studies.

In our work, the Oxford 102 Flower dataset is labelled not only containing flower classification to perform flower detection to be utilized in a smart garden in AIoT. As shown in Figure 3, the labelled flower dataset can be contributed to all flower, plant, and fruit detection studies, especially in a completed natural environment. The dataset was labelled by a professional data annotator to classify the flower dataset which can contribute to the accuracy of the flower detection model and make it easier to perform real-time flower detection.

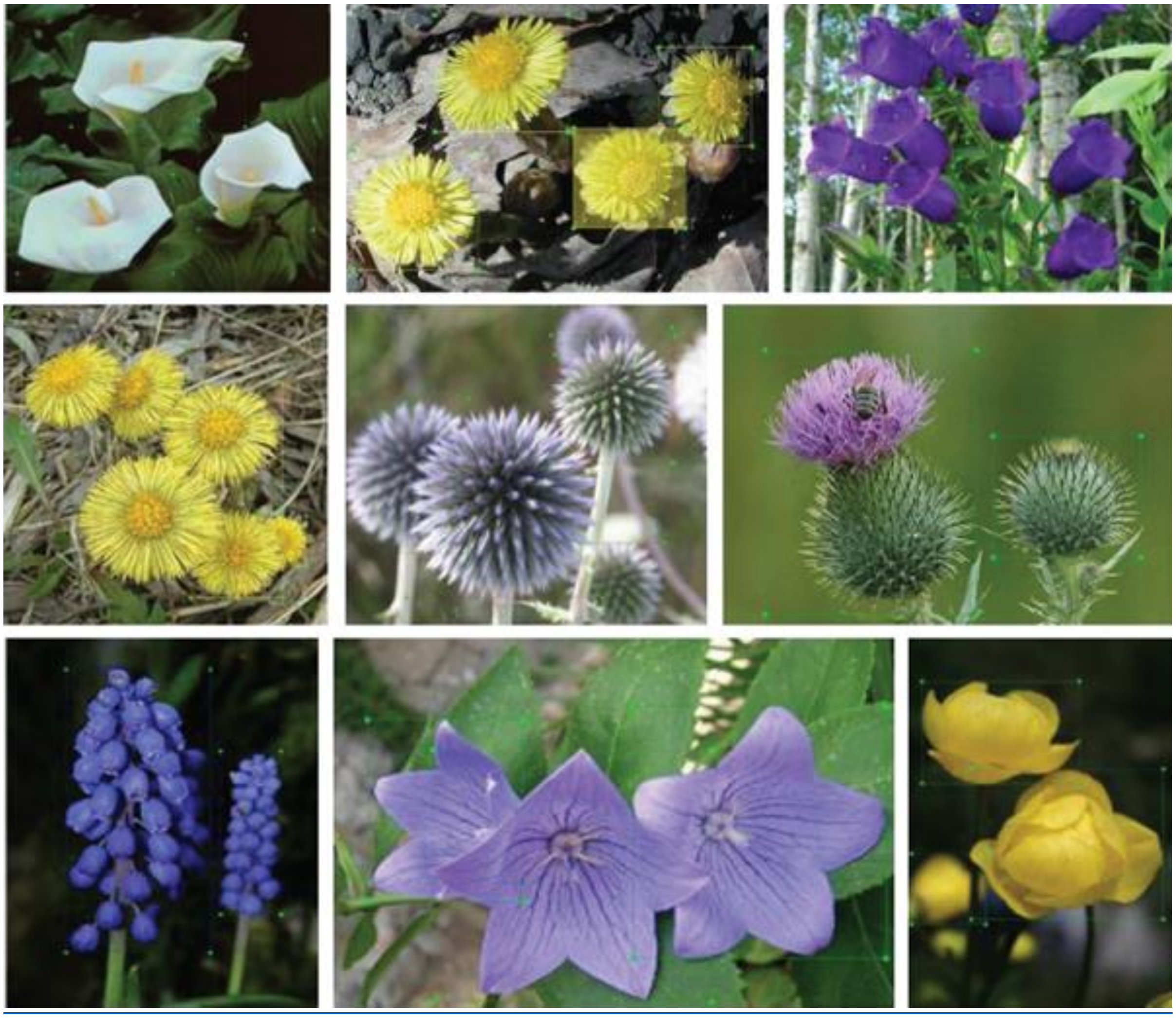


Figure 3

Oxford 102 Flower dataset with annotation, which contains 102 categories of flowers, can be used for flower detection studies.

More training data can lead to a more sound model. If there are limited data volumes, it can make use of data augmentation to increase the diversity of the training sample to enhance model robustness, avoid overfitting, and improve the generalization ability of the model. The common methods include flip, rotation, shift, resize, random crop or pad, color jittering, and noise. In our dataset, data augmentation is also used to enhance our model.

2.3. Loss Function

In our work, the loss function DIoU loss is the loss function we used, which is feasible to directly minimize the normalized distance between the anchor frame and the target frame to achieve faster convergence speed and is more accurate and faster when overlapped or even included with the target box when making regression.

DIoU loss is on the basis of IoU (Intersection over Union), considering the information of the center distance of the bounding box. The definition of it can be defined as follows, formula (1), where is the target box and is the prediction box. And the loss function of the IoU is defined as formula (2); it is shown that it works if the bounding boxes overlap; there is no overlap if the gradient does not change.

Therefore, it is GIoU that can improve the loss of IoU in the case that the gradient does not change without overlapping boundary boxes, which adds a penalty term on the basis of the loss function of IoU. It is defined as the following formula (3). The additional parameter in the formula represents the minimum boundary box that can cover both and . However, if one of or overrides the other box in the case, the penalty term cannot work, which can be regarded as an IoU loss.

To solve the limitations, the DIoU is introduced which can be defined as formula (4),
where , represents the center point of the anchor frame and target frame, respectively, and is the Euclidean distance between two centers. Also, represents the diagonal distance of the minimum rectangle that can cover the anchor and target box simultaneously. What is more, the normalized distance between the anchor box and the target box is modeled in DIoU. The loss function of DIoU can be defined as formula (5).

Besides, CIoU is also proposed to our loss function. There are several advantages in the CIoU loss function: (1) It can increase the overlap area between the ground truth box and the prediction box. (2) It can minimize the distance between the center points. (3) It can keep the frame height ration consistent.

The CIoU can be defined as formula (6) based on formula (4), where represents a positive trade-off and is defined in formula (7) to measure the consistency of the aspect ratio added.

The loss function of CIoU can be defined as formula (8), and the trade-off parameter can be defined in formula (9).

In our work, the DIoU loss function can be applied in NMS (Nonmaximum Suppression) to delete the redundant detection box. Not only the overlapping area is considered but also the distance between the detection box, and the center point of the target box is taken into account, which can effectively avoid the above two loss function mistakes.

2.4. Activation Function

Activation function is a function running on the neuron of the neural network, which is responsible for mapping the input of the neuron to the output. Its function is to increase the nonlinear change of the neural network model. In Figure 4, the Mish activation function and the comparison of different commonly used activation function are shown, in which the original figure can be found in Mish [30].

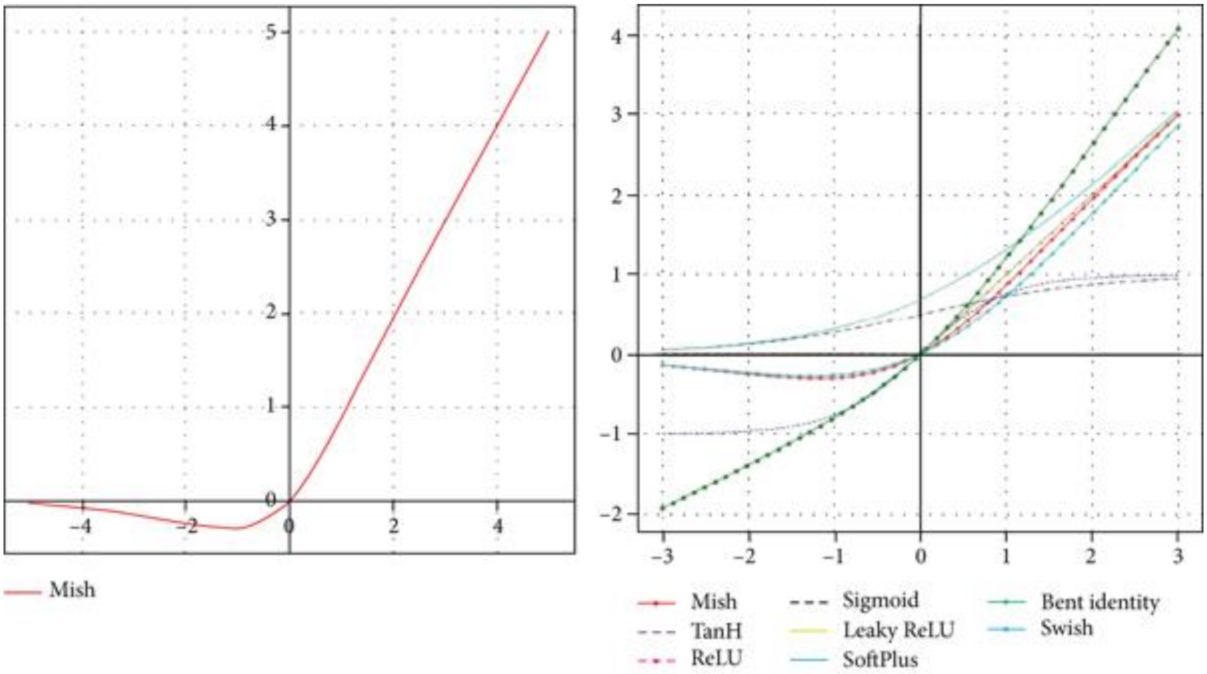


Figure 4

Representation of Mish activation function and its performance: (a) Mish activation function; (b) comparison of the different commonly used activation.

In our architecture, the Mish [30] activation function is the activation function we utilized, which replaces Leaky Relu which is a very small constant leak that contained the improved function of Relu with Mish in YOLOv3 [11]. Leaky Relu is a self-regular nonmonotone neural activation function and a smooth activation function allowing better information into the neural network to obtain better accuracy and generalization. It can be defined as formula (10), where in the formula (10), it is shown that .

It is more accurate and precious than Swish [31] defined as formula (12) and Relu [32] defined as formula (11) when performing on the experiments.

2.5. Attention Mechanism

As we all know, attention mechanism has become an important part of the structure of neural networks and has a large number of applications in fields such as natural language processing, statistical learning, and computer vision in the field of artificial intelligence. The attention mechanism in deep learning is similar to the selective visual attention mechanism of human beings in essence. The core goal is to select the information that is more critical to the current task target from the numerous information.

The SE [33] and SAM [34] model is the common attention mechanism used in deep learning networks. The purpose of the SE model is to reweight to the feature channel and only pay attention to which

layer on the channel level will have stronger feedback ability, but it cannot reflect the meaning of "attention" on the spatial dimension, while SAM which is the output of the feature map of the channel attention module that is taken as the input feature map of this module is more adaptive to our architecture since it saves a lot of computing resources. In our work, the modified SAM has been utilized, which is shown in Figure 5. It is seen that spacewise attention is used instead of pointwise attention in it. Besides, PAN is replaced with concatenation connections.

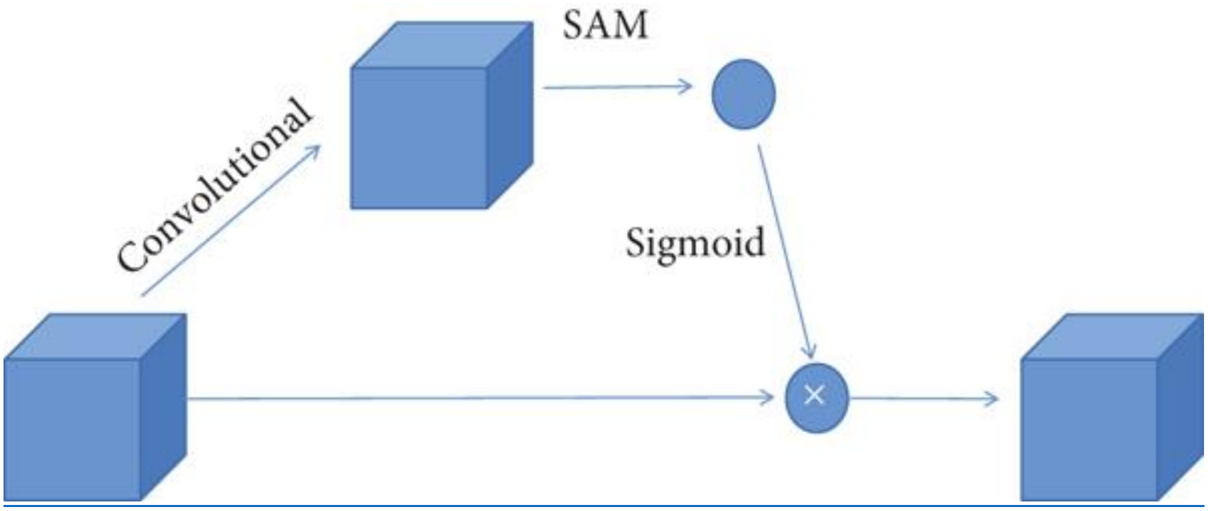


Figure 5

The modified SAM architecture.

3.CODE

```
# DataFlair Iris Classification
# Import Packages
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd
columns = ['Sepal length', 'Sepal width', 'Petal length', 'Petal width', 'Class_labels'] # As per the iris dataset information
# Load the data
df = pd.read_csv('iris.data', names=columns)
```

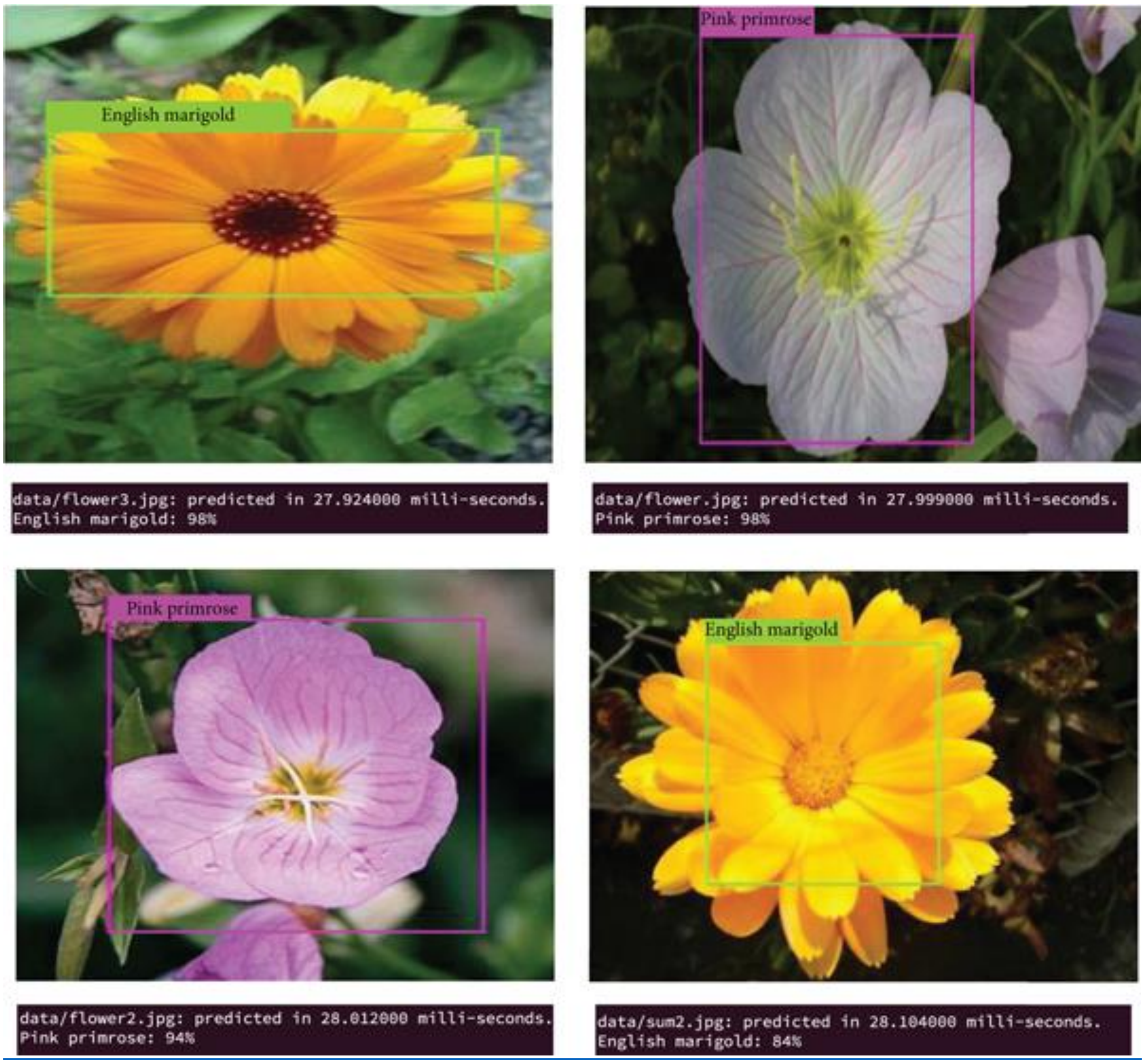
```
df.head()
# Some basic statistical analysis about the data
df.describe()
# Visualize the whole dataset
sns.pairplot(df, hue='Class_labels')
# Seperate features and target
data = df.values
X = data[:,0:4]
Y = data[:,4]
# Calculate avarage of each features for all classes
Y_Data = np.array([np.average(X[:, i][Y==j].astype('float32')) for i in range (X.shape[1]) for j in (np.unique(Y))])
Y_Data_reshaped = Y_Data.reshape(4, 3)
Y_Data_reshaped = np.swapaxes(Y_Data_reshaped, 0, 1)
X_axis = np.arange(len(columns)-1)
width = 0.25
# Plot the avarage
plt.bar(X_axis, Y_Data_reshaped[0], width, label = 'Setosa')
plt.bar(X_axis+width, Y_Data_reshaped[1], width, label = 'Versicolour')
plt.bar(X_axis+width*2, Y_Data_reshaped[2], width, label = 'Virginica')
plt.xticks(X_axis, columns[:4])
plt.xlabel("Features")
plt.ylabel("Value in cm.")
plt.legend(bbox_to_anchor=(1.3,1))
plt.show()
# Split the data to train and test dataset.
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size=0.2)
# Support vector machine algorithm
```

```
from sklearn.svm import SVC

svn = SVC()

svn.fit(X_train, y_train)

# Predict from the test dataset

predictions = svn.predict(X_test)

# Calculate the accuracy

from sklearn.metrics import accuracy_score

accuracy_score(y_test, predictions)

# A detailed classification report

from sklearn.metrics import classification_report

print(classification_report(y_test, predictions))

X_new = np.array([[3, 2, 1, 0.2], [  4.9, 2.2, 3.8, 1.1 ], [  5.3, 2.5, 4.6, 1.9 ]])

#Prediction of the species from the input vector

prediction = svn.predict(X_new)

print("Prediction of Species: {}".format(prediction))

# Save the model

import pickle

with open('SVM.pickle', 'wb') as f:

    pickle.dump(svn, f)

# Load the model

with open('SVM.pickle', 'rb') as f:

    model = pickle.load(f)

model.predict(X_new)
```

4. Results and Discussion

We describe the results of our flower detection in Figure 6, which shows the result of flower detection containing category and degree of confidence added in each category. The confidence levels are 98%, 98%, 94%, and 84%, respectively, in the test results, which achieved the desired effect. It is demonstrated that the flower detection can be achieved at a high standard which the accuracy has matched the speed. The GPU we used are 2 Titan Xp. The basic requirements for a camera on a mobile

device are based on HUAWEI P20 Pro, and the processor is HiSilicon Kirin 970. When applied in a smart garden in AIoT, the basic mobile network relied on a 5G network, which is a high-speed, low-delay, low-power consumption and ubiquitous network.



```
data/flower3.jpg: predicted in 27.924000 milli-seconds.
English marigold: 98%
```

```
data/flower.jpg: predicted in 27.999000 milli-seconds.
Pink primrose: 98%
```

```
data/flower2.jpg: predicted in 28.012000 milli-seconds.
Pink primrose: 94%
```

```
data/sum2.jpg: predicted in 28.104000 milli-seconds.
English marigold: 84%
```

Figure 6

The results of flower detection.

5. Conclusions

In conclusion, it seems that flower detection based on the state-of-the-art method is very considerable. Although widely accepted, it suffers from some limitations due to the flower detection algorithms that have not been integrated into the mobile device completely to perform the final application results of our flower detection to enhance user's human-computer interaction experience in a smart garden combining with virtual reality. It is a tendency for the flower detection proposed in our paper combined with virtual reality. In a smart garden, it can be realized that flowers can be arranged intelligently not depending on people just on the smart phone or other mobile devices. Furthermore, intelligent gardening can come true which can reduce the cost of human and finance and it just relies on the human-computer interaction. In the future work, we will pay much importance to it to accomplish the goal in our future work.

Reference

R.S Khalaf and A. Varol, ”Digital Forensics: Focusing on Image Forensics,” 2019 7th International Symposium on Digital Forensics and Security (ISDFS), 2019, pp. 1-5, doi: 10.1109/ISDFS.2019.8757557

G Maria Jones; S Godfrey Winster, ”An Insight into Digital Forensics: History, Frameworks, Types and Tools,” in Cyber Security and Digital Forensics: Challenges and Future Trends, Wiley, 2022, pp.105-125, doi: 10.1002/9781119795667.ch6

H. Majed, H. N. Noura, and A. Chehab, "Overview of Digital Forensics and Anti-Forensics Techniques," 2020 8th International Symposium on Digital Forensics and Security (IS-DFS), 2020, pp. 1-5, doi: 10.1109/ISDFS49300.2020.9116399

O. M. Adedayo, ”Big data and digital forensics,” 2016 IEEE International Conference on Cybercrime and Computer Forensic (ICCCF), 2016, pp. 1-7, doi: 10.1109/IC-CCF.2016.7740422

Refaces. (2022, January 18). What is Digital Forensics: Process, tools, and types: Computer Forensicsoverview. RecFaces. Retrieved from https://recfaces.com/articles/digital-forensics

K. U. Maheshwari and G. Shobana, "The State of the art tools and techniques for remote digital forensic investigations," 2021 3rd International Conference on Signal Processing and Communication (ICPSC), 2021, pp. 464-468, doi: 10.1109/ICSPC51351.2021.9451718.

L. Chen, L. Xu, X. Yuan and N. Shashidhar, "Digital forensics in social networks and the cloud: Process, approaches, methods, tools, and challenges," 2015 International Conference on Computing, Networking and Communications (ICNC), 2015, pp. 1132-1136, doi: 10.1109/ICCNC.2015.7069509.

K. S. Singh, A. Irfan and N. Dayal, "Cyber Forensics and Comparative Analysis of Digital Forensic Investigation Frameworks," 2019 4th International Conference on Information Systems and Computer Networks (ISCON), 2019, pp. 584-590, doi: 10.1109/ISCON47742.2019.9036214.

K. Ghazinour, D. M. Vakharia, K. C. Kannaji and R. Satyakumar, "A study on digital forensic tools," 2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI), 2017, pp. 3136-3142, doi: 10.1109/ICPCSI.2017.8392304.

A. Al-Sabaawi, "Digital Forensics for Infected Computer Disk and Memory: Acquire, Analyse, and Report," 2020 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE), 2020, pp. 1-7, doi: 10.1109/CSDE50874.2020.9411614.

Chaitanya Krishna Suryadevara, “TOWARDS PERSONALIZED HEALTHCARE - AN INTELLIGENT MEDICATION RECOMMENDATION SYSTEM”, IEJRD - International Multidisciplinary Journal, vol. 5, no. 9, p. 16, Dec. 2020.

Kunduru, A. R. (2023). DATA CONVERSION STRATEGIES FOR ERP IMPLEMENTATION PROJECTS. CENTRAL ASIAN JOURNAL OF MATHEMATICAL THEORY AND COMPUTER SCIENCES, 4(9), 1-6. Retrieved from https://cajmtcs.centralasianstudies.org/index.php/CAJMTCS/article/view/509

Arjun Reddy Kunduru. (2023). Healthcare ERP Project Success: It’s all About Avoiding Missteps. Central Asian Journal of Theoretical and Applied Science, 4(8), 130-134. Retrieved from https://cajotas.centralasianstudies.org/index.php/CAJOTAS/article/view/1268

Kunduru, A. R. (2023). THE PERILS AND DEFENSES OF ENTERPRISE CLOUDCOMPUTING: A COMPREHENSIVE REVIEW. Central Asian Journal of Mathematical Theory and Computer Sciences, 4(9), 29-41.

Kunduru, A. R. (2023). Maximizing Business Value with Integrated IoT and Cloud ERP Systems. International Journal of Innovative Analyses and Emerging Technology, 3(9), 1-8.

Kunduru, A. R. (2023). Blockchain Technology for ERP Systems: A Review. American Journal of Engineering, Mechanics and Architecture, 1(7), 56-63.

Suryadevara, Chaitanya Krishna, Predictive Modeling for Student Performance: Harnessing Machine Learning to Forecast Academic Marks (December 22, 2018). International Journal of Research in Engineering and Applied Sciences (IJREAS), Vol. 8 Issue 12, December-2018, Available at SSRN: https://ssrn.com/abstract=4591990

Suryadevara, Chaitanya Krishna, Unveiling Urban Mobility Patterns: A Comprehensive Analysis of Uber (December 21, 2019). International Journal of Engineering, Science and Mathematics, Vol. 8 Issue 12, December 2019, Available at SSRN: https://ssrn.com/abstract=4591998

Chaitanya Krishna Suryadevara. (2019). A NEW WAY OF PREDICTING THE LOAN APPROVAL PROCESS USING ML TECHNIQUES. International Journal of Innovations in Engineering Research and Technology, 6(12), 38–48. Retrieved from https://repo.ijiert.org/index.php/ijiert/article/view/3654

Chaitanya Krishna Suryadevara. (2020). GENERATING FREE IMAGES WITH OPENAI'S GENERATIVE MODELS. International Journal of Innovations in Engineering Research and Technology, 7(3), 49–56. Retrieved from https://repo.ijiert.org/index.php/ijiert/article/view/3653

Chaitanya Krishna Suryadevara. (2020). REAL-TIME FACE MASK DETECTION WITH COMPUTER VISION AND DEEP LEARNING: English. International Journal of Innovations in Engineering Research and Technology, 7(12), 254–259. Retrieved from https://repo.ijiert.org/index.php/ijiert/article/view/3184

Chaitanya Krishna Suryadevara. (2021). ENHANCING SAFETY: FACE MASK DETECTION USING COMPUTER VISION AND DEEP LEARNING. International Journal of Innovations in Engineering Research and Technology, 8(08), 224–229. Retrieved from https://repo.ijiert.org/index.php/ijiert/article/view/3672

Kunduru, A. R. (2023). Security concerns and solutions for enterprise cloud computing applications. Asian Journal of Research in Computer Science, 15(4), 24–33. https://doi.org/10.9734/ajrcos/2023/v15i4327

Kunduru, A. R. (2023). Industry best practices on implementing oracle cloud ERP security. International Journal of Computer Trends and Technology, 71(6), 1-8. https://doi.org/10.14445/22312803/IJCTT-V71I6P101

Kunduru, A. R. (2023). Cloud Appian BPM (Business Process Management) Usage In health care Industry. IJARCCE International Journal of Advanced Research in Computer and Communication Engineering, 12(6), 339-343. https://doi.org/10.17148/IJARCCE.2023.12658

Kunduru, A. R. (2023). Effective usage of artificial intelligence in enterprise resource planning applications. International Journal of Computer Trends and Technology, 71(4), 73-80. https://doi.org/10.14445/22312803/IJCTT-V71I4P109

Kunduru, A. R. (2023). Recommendations to advance the cloud data analytics and chatbots by using machine learning technology. International Journal of Engineering and Scientific Research, 11(3), 8-20.

Kunduru, A. R., & Kandepu, R. (2023). Data archival methodology in enterprise resource planning applications (Oracle ERP, Peoplesoft). Journal of Advances in Mathematics and Computer Science, 38(9), 115–127. https://doi.org/10.9734/jamcs/2023/v38i91809

Chaitanya Krishna Suryadevara, “DIABETES RISK ASSESSMENT USING MACHINE LEARNING: A COMPARATIVE STUDY OF CLASSIFICATION ALGORITHMS”, IEJRD - International Multidisciplinary Journal, vol. 8, no. 4, p. 10, Aug. 2023.

Chaitanya Krishna Suryadevara. (2023). REVOLUTIONIZING DIETARY MONITORING: A COMPREHENSIVE ANALYSIS OF THE INNOVATIVE MOBILE APP FOR TRACKING DIETARY COMPOSITION. International Journal of Innovations in Engineering Research and Technology, 10(8), 44–50. Retrieved from https://repo.ijiert.org/index.php/ijiert/article/view/3673

Chaitanya krishna Suryadevara. (2023). NOVEL DEVICE TO DETECT FOOD CALORIES USING MACHINE LEARNING. Open Access Repository, 10(9), 52–61. Retrieved from https://oarepo.org/index.php/oa/article/view/3546

Kunduru, A. R. (2023). Artificial intelligence usage in cloud application performance improvement. Central Asian Journal of Mathematical Theory and Computer Sciences, 4(8), 42-47. https://cajmtcs.centralasianstudies.org/index.php/CAJMTCS/article/view/491

Kunduru, A. R. (2023). Artificial intelligence advantages in cloud Fintech application security. Central Asian Journal of Mathematical Theory and Computer Sciences, 4(8), 48-53. https://cajmtcs.centralasianstudies.org/index.php/CAJMTCS/article/view/492

9808:675X
Highly Cited Journal
Acceptance Ration Below: 8%