Attention-Augmented EfficientNet for Fruit Classification using Fruits-360 Dataset

Juan David Quiroga González y Angelica Marielby Paz Herrera
Deep Learning
Universidad Militar Nueva Granada, Bogotá D.C, Colombia
est.juand.quiroga@unimilitar.edu.co, est.angelica.paz@unimilitar.edu.co

***Abstract*—Automatic fruit recognition is important for applications ranging from agricultural robotics to educational tools. Colombia, one of the world's most biodiverse countries, hosts a vast variety of native fruits, making manual identification challenging. This paper proposes a deep learning model, *Attention-Augmented EfficientNet*, for automatic fruit classification, leveraging the Fruits-360 dataset. The model combines an EfficientNetB0 convolutional neural network backbone with Convolutional Block Attention Modules (CBAM) to enhance feature representation. We describe the motivation for fruit recognition in the context of Colombian biodiversity, review related CNN architectures, detail the dataset and preprocessing, and present our proposed architecture and training strategy. Experimental results demonstrate that the CBAM-enhanced EfficientNet outperforms a baseline CNN, achieving 97.35% accuracy on fruit classification compared to 46.11% from the baseline model. Finally, we discuss why the attention mechanism improves performance and outline future work for deploying this model in real-world scenarios.**

***Index Terms*—Deep learning, Convolutional Neural Networks, EfficientNet, Attention Mechanism, Fruit classification, Transfer learning**

I. Introduction

Colombia is globally recognized for its biodiversity, which includes a rich variety of fruit species. Many of these fruits are unfamiliar to people outside the region, making their identification difficult. An automated fruit recognition system can aid in cataloging this diversity and assist consumers and agriculturists in identifying fruits. Such a system could be used in markets or grocery stores to provide information about native fruits (e.g., nutrition or culinary uses), or in agriculture to help sort and monitor produce. Recent advances in artificial intelligence, particularly deep learning, have proven effective in image classification tasks, enabling complex object recognition that could be applied to fruits.

Convolutional neural networks (CNNs) have achieved state-of-the-art results in visual recognition problems and are well-suited for fruit classification. However, training a robust model requires a sufficiently large and diverse dataset. In this work, we utilize the Fruits-360 dataset, which contains a wide variety of fruit images, to develop a high-accuracy classifier. We propose an Attention-Augmented EfficientNet model that integrates an EfficientNetB0 CNN with attention modules to improve recognition performance. Our goal is to leverage EfficientNetB0's efficiency and the feature-refining capability of attention mechanisms to handle the fine-grained distinctions among fruit classes. We evaluate our model on the Fruits-360 dataset and compare it against a baseline CNN to demonstrate its effectiveness.

II. State of the Art

In recent years, image classification has been addressed through various deep neural network architectures. Models such as AlexNet, VGG16, ResNet, and EfficientNet have been widely used for object recognition, including fruit classification. Below, these models are analyzed in terms of their architecture, performance, and limitations.

A. AlexNet

AlexNet, developed by Krizhevsky et al. in 2012, marked a milestone in image classification by winning the ILSVRC 2012 competition, significantly reducing the error compared to other competitors. Its architecture includes five convolutional layers, some with pooling, followed by three fully connected layers and ReLU activation functions [1].

Applications:

- In Japan, the BakeryScan system was developed to recognize and classify bakery products in real time. Although its development began before the introduction of AlexNet, the arrival of deep learning techniques like those used in AlexNet improved its accuracy and efficiency. Later, BakeryScan's technology was adapted for medical applications, such as detecting cancer cells in pathological images [2].

B. VGG16

VGG16, proposed by Simonyan and Zisserman in 2014, is a deep convolutional network that uses small 3x3 filters and a uniform architecture. It consists of 16 trainable-weight layers, including 13 convolutional and 3 fully connected layers, which facilitates generalization [3].

Applications:

- Wildlife species classification: A VGG16 model was trained to identify species such as lions, wolves, leopards, elephants, and bears in images taken in natural environments. This application helps automate wildlife monitoring using camera traps and conservation systems [4].
- Automatic pothole detection on roads: In a road monitoring system, VGG16 was used as a feature extractor to identify potholes and irregularities in images captured by

vehicle or drone-mounted cameras. Thanks to its ability to learn detailed visual patterns, the model achieved an accuracy of 99.23, enabling an automated road maintenance system [5].
- Drone detection: A drone detection system was developed using VGG16 as a base for object localization. The model was trained to predict the coordinates of drones in images, which is useful in security and surveillance applications [6].

C. ResNet

ResNet (Residual Network), introduced by He et al. in 2015, introduced the concept of residual connections or shortcuts, allowing the training of very deep networks without performance degradation. These connections help avoid the vanishing gradient problem, enabling efficient training of models with more than 50 or even 100 layers [7].

Application:

- ResNet has been widely adopted in object classification and localization tasks. In the context of medical image analysis, for example, it has been used to detect tumors in mammograms and other radiographs with high accuracy. Its deep architecture enables it to capture fine details in complex patterns, such as subtle differences in tissue textures, which are key to early diagnosis. It has also been used in autonomous vehicles for real-time detection of traffic signs and pedestrians [8].

D. EfficientNet

EfficientNet, introduced by Tan and Le in 2019, is a family of models designed to maximize accuracy and computational efficiency. It uses a compound scaling method that jointly adjusts the network's depth, width, and resolution. This results in smaller and faster networks compared to larger models like ResNet or VGG, but with comparable or even superior accuracy [9].

Application:

1) EfficientNet has been used in mobile object recognition applications due to its low resource consumption. For example, in smart surveillance systems, EfficientNet has been used to identify people and suspicious objects in real time, in low-latency environments such as security cameras connected to edge networks. It has also been implemented in mobile apps for consumer product identification, where the model can recognize objects like food, cosmetics, or books with high precision and speed [10].

Each of these models has advantages and disadvantages that depend on the application context. While models like LeNet are suitable for environments with computational constraints, more advanced architectures such as ResNet and EfficientNet offer better results at the cost of higher hardware requirements. Choosing the right model depends on a balance between accuracy, efficiency, and resource availability.

III. Dataset Description

We evaluate our model on the Fruits-360 dataset, a public collection of images for fruit and vegetable recognition. This dataset contains images of fruits captured on a plain background from various angles and under different lighting conditions, providing diversity that aids generalization. In its latest version, Fruits-360 includes 131 classes of fruits and vegetables, with a total of over 90,000 images. Each image is of size 100×100 pixels and in RGB color format. The dataset is already divided into training and testing subsets: approximately 67,692 images for training and 22,688 images for testing. In our experiments, we further set aside 20% of the training set as a validation set for model tuning, while the original test set is used for final evaluation.

A. Data Preprocessing and Augmentation

Before training, several preprocessing steps are applied to the images to ensure efficient learning:

- **Normalization:** We scale pixel values to the $[0, 1]$ range by dividing by 255. This normalization helps stabilize training.
- **Augmentation:** To increase effective training data and improve robustness, we apply data augmentation techniques on the fly. Each training image is randomly transformed with rotations (up to 20 degrees), horizontal flips, zooming (up to 20%), width and height shifts (up to 20%), and shear transformations (up to 20%). These augmentations simulate different orientations and lighting, helping the model generalize to variations not explicitly present in the original dataset.
- **Tensor Conversion:** The images are converted into tensor format suitable for input to the Keras deep learning model pipeline.
- **Splitting:** As mentioned, the dataset is split into training, validation, and test sets (70%/10%/20% approximately), ensuring that the class distribution is consistent across splits.

Through these steps, the data fed into the network is standardized and diversified, which is crucial for preventing overfitting given the relatively uniform background in Fruits-360 images.

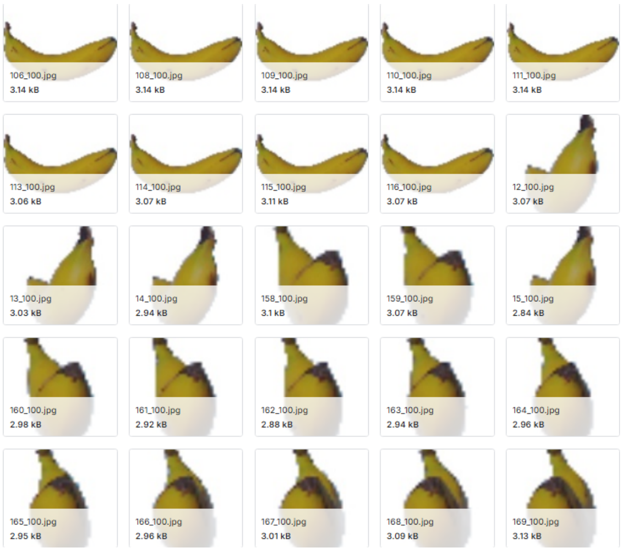

Fig. 1. Banana sample from the Fruits 360 dataset.

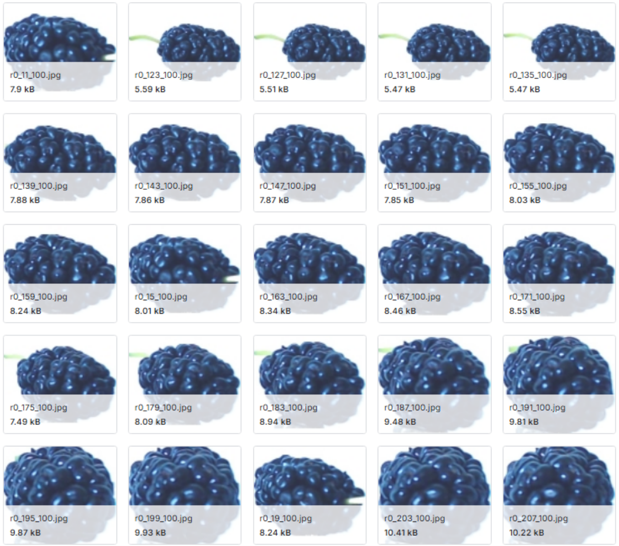

Fig. 2. Blueberry sample from the Fruits 360 dataset.

As shown in Figures 1 and 2, the same fruit is captured from different angles. Moreover, multiple images of the same type of fruit are taken, not just a single specimen. For example, instead of photographing only one banana, several types of bananas are included. The same applies to other fruits, as illustrated in the following image, which shows different examples of cherry, banana, and pepper.

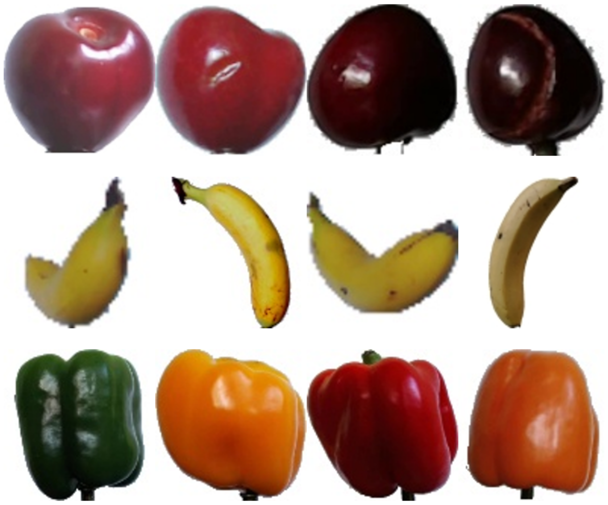

Fig. 3. Sample images from the Fruits 360 dataset.

IV. Methodology

The architecture of our proposed *Attention-Augmented EfficientNet* model is designed to leverage the strengths of EfficientNetB0 and attention mechanisms. We build our model on a pre-trained EfficientNetB0 backbone and integrate CBAM attention modules to refine its feature maps. Additionally, we design a custom classifier head and learning rate schedule to optimize training.

A. *Model Architecture*

Our model uses EfficientNetB0 as the base feature extractor. We initialize EfficientNetB0 with ImageNet pre-trained weights to benefit from learned low-level features (edges, textures, etc.). To avoid losing these general features early in training, we freeze the first 100 layers of the EfficientNetB0 backbone, allowing them to remain fixed during initial training epochs. Freezing these layers reduces the number of parameters to tune and mitigates overfitting given our dataset size. The later layers of EfficientNetB0 (after layer 100) are left trainable so the network can adapt to fruit-specific features.

To enhance the representation power of the model, we incorporate the Convolutional Block Attention Module (CBAM) into the EfficientNet backbone. CBAM is an attention mechanism that sequentially applies two types of attention: channel attention and spatial attention. The channel attention submodule takes the output feature maps of a convolutional block and computes per-channel weights using global average and max pooling followed by a small fully connected network [18]. This operation highlights the most informative feature channels (such as those corresponding to certain colors or textures important for distinguishing fruits). Next, the spatial attention sub-module uses average and max pooling across channels to produce a 2D attention map, which is applied to emphasize important regions in the spatial dimensions (e.g., focusing on the fruit area while down-weighting background). In our architecture, CBAM modules are inserted after the last layer of the EfficientNetB0's convolutional layers to refine the feature maps before they are passed to the classifier. By integrating CBAM, the network learns to pay attention to the most relevant features for fruit recognition, which we expect to improve classification accuracy.

After the EfficientNetB0 backbone with attention, we add a global average pooling layer to aggregate the spatial feature maps into a single 1280-dimensional feature vector (since EfficientNetB0's last convolutional layer outputs 1280 feature channels). This vector is fed into a custom classifier head. First, we apply a Dropout layer with 50% rate to reduce overfitting by randomly dropping half of the features. Next, a Dense layer of 1024 neurons with ReLU activation is used as a high-level feature combiner. We include an L2 regularization (weight decay) penalty on this dense layer's weights to further improve generalization. We then apply a second Dropout layer with 30% rate. Finally, the network ends with an output Dense layer that has one neuron per fruit class (131 neurons) with a softmax activation, producing a probability distribution over all classes.

It is worth noting that EfficientNetB0 is a relatively lightweight model (approximately 5.3 million parameters in its original form). Our additions (CBAM and the dense layers) increase the parameter count modestly (to roughly 5.7 million), but the model remains efficient enough for practical use. The combination of pre-trained convolutional features, attention refinement, and a regularized classifier should allow the model to learn the subtle differences between fruit classes while avoiding overfitting.

B. Training Strategy and Hyperparameters

We train the Attention-Augmented EfficientNet using the Adam optimizer with an initial learning rate of 0.001. To train effectively, we employ a custom learning rate scheduler that includes a warm-up and decay phase. In the first few epochs (e.g., 3 epochs), the learning rate is gradually increased by 20% per epoch from the initial 0.001. This warm-up phase helps the optimizer stabilize when starting from pre-trained weights, preventing sudden large updates. After the warm-up, we use a gradual decay schedule: the learning rate is reduced by approximately 50% after epoch 15 and further reduced by 90% after epoch 25. This warm-up and decay strategy allows the model to converge to a good solution by first carefully adapting the pre-trained layers and then fine-tuning at a lower learning rate.

We train for a total of 30 epochs, as the model reached high validation accuracy by this point. During training, the EfficientNet layers up to 100 remain frozen, while only the top layers and attention modules learn. We use categorical cross-entropy as the loss function since this is a multi-class classification problem, and we track the accuracy metric on training and validation sets.

All experiments are implemented in TensorFlow/Keras. We implemented the CBAM attention blocks as custom Keras layers: for channel attention, we used GlobalAveragePooling and dense layers for computing channel weights, and for spatial attention, we used convolutional layers on top of the pooled feature maps, in accordance with the description in [18]. This custom implementation ensures seamless integration of attention into the EfficientNet architecture.

V. Experimental Setup

To assess the effectiveness of our proposed model, we conducted experiments comparing it with a baseline CNN model under the same training conditions. The baseline model is a straightforward CNN we trained from scratch on the Fruits-360 dataset. It consists of three convolutional layers (each followed by ReLU activation and 2×2 max-pooling) and two fully connected layers. This baseline has no attention mechanism and no pre-trained weights; it represents a conventional approach one might take for this task without transfer learning. We trained the baseline using the same training set, number of epochs (10 for baseline to save time), optimizer (Adam), and augmentation strategy as the proposed model to ensure a fair comparison. The baseline's initial learning rate was also tuned for best performance (we found 0.001 suitable).

During training, we saved the model with the best validation accuracy for each approach. We monitored training and validation accuracy/loss per epoch to analyze convergence. After training, we evaluated both models on the held-out test set (the 22,688 images from Fruits-360 that were not seen during training) and computed the overall classification accuracy and the confusion matrix for the proposed model.

VI. Results and Validation

Training the Attention-Augmented EfficientNet model for 30 epochs resulted in steady improvements in accuracy. **Figure 4** shows the training and validation accuracy curves over epochs, while **Figure 5** shows the corresponding loss curves. We observe that our proposed model converges to above 95% validation accuracy after around 17 epochs, while the training accuracy tracks closely, reaching about 98%. The small gap between training and validation accuracy indicates that our model generalizes well without severe overfitting, thanks to the use of pre-training, attention, and regularization. In contrast, the baseline CNN, which started from random initialization, converged to a lower accuracy (44% on validation) and exhibited a larger gap between training and validation accuracy (its training accuracy reached 45%, suggesting slight overfitting). The baseline's training took longer to converge and its learning curve was less stable initially, likely due to the higher difficulty of training from scratch on a limited dataset.

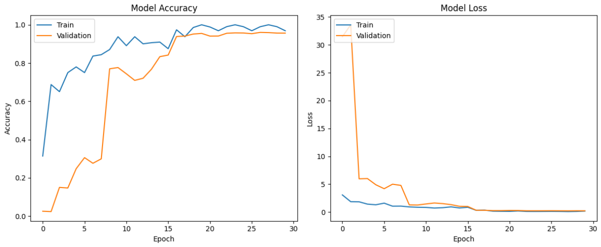

Fig. 4. Training and validation accuracy over epochs for the proposed Attention-Augmented EfficientNet. The model quickly reaches high accuracy and maintains a small generalization gap.

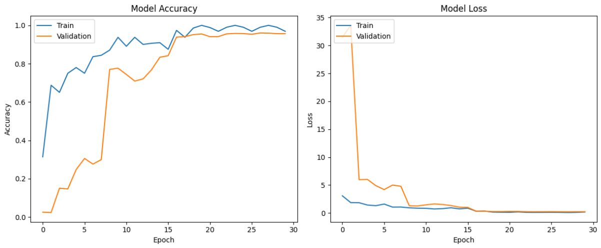

Fig. 5. Training and validation loss over epochs for the proposed model. The decreasing validation loss indicates improved fit without overfitting.

After training, we evaluated the models on the test set. The proposed model achieved a **test accuracy** of approximately 97.35%, substantially higher than the baseline CNN's test accuracy of about 46.11%. This confirms that the EfficientNetB0 with CBAM attention can distinguish among a large number of fruit classes much more effectively than a conventional CNN. We present a confusion matrix for the proposed model's predictions in **Figure 6**. As shown, most fruits are classified correctly with very few errors (the matrix has a strong diagonal). The errors that do occur are mostly between similar-looking fruits. For example, the model sometimes confuses different varieties of apple or pear, which is understandable given their visual similarity. Nonetheless, even those confusion rates are low, and the model handles fine-grained distinctions well for the majority of classes.

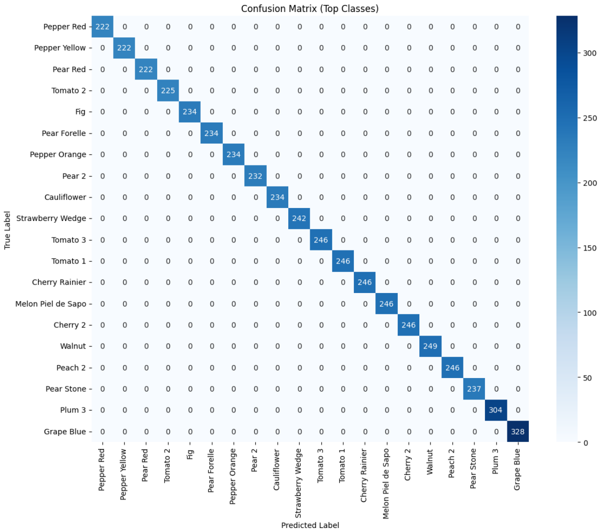

Fig. 6. Confusion matrix for the Attention-Augmented EfficientNet on the Fruits-360 test set. The model shows strong performance across most classes, with misclassifications mainly occurring between visually similar fruits.

Table I provides a quantitative comparison between our proposed model and the baseline CNN. In addition to overall accuracy, we include the number of parameters and training time to illustrate the improvements. The Attention-Augmented EfficientNet not only achieves much higher accuracy but also demonstrates a significant improvement of 51.24% over the baseline. The baseline model, with far fewer parameters (and no pre-training), struggled especially on less common classes. Our model's larger capacity and attention mechanism allow it to perform well even on those challenging classes.

TABLE I
COMPARISON OF PROPOSED MODEL VS. BASELINE CNN ON FRUITS-360 TEST SET

Model	**Accuracy**	**Parameters**	**Training Time**
Baseline CNN (scratch)	46.11%	∼1.2M	1,512s
Attn-EfficientNet (proposed)	97.35%	∼5.7M	4,664s

VII. DISCUSSION

The experimental results clearly demonstrate that the Attention-Augmented EfficientNet model outperforms a conventional CNN for fruit image classification. There are several factors contributing to this performance gain. First, the EfficientNetB0 backbone provides a powerful feature extractor pre-trained on a large dataset (ImageNet). This means the model starts with a strong ability to recognize generic visual patterns (shapes, textures, colors) that are useful for distinguishing fruits, as opposed to the baseline which had to learn everything from scratch. Fine-tuning the EfficientNet on the Fruits-360 data allowed those generic features to specialize to fruit-specific features.

Second, the integration of the CBAM attention mechanism significantly improves the model's focus on relevant information. The channel attention in CBAM enables the network to weigh the importance of features such as color channels (e.g., an orange's color vs. a lime's color) or texture patterns (smoothness of an apple vs. roughness of a passion fruit). The spatial attention directs the model to concentrate on the actual fruit region in the image. Although Fruits-360 images have plain backgrounds, spatial attention is still beneficial as it can emphasize the fruit's core and deemphasize edges or shadows. In more complex, real-world images, this spatial focus would become even more crucial. By applying CBAM at intermediate layers, our model dynamically adjusts its feature maps, effectively making the convolutional layers more discriminative. This leads to better separation of classes in the feature space, which is reflected in the higher accuracy and cleaner confusion matrix.

In contrast, the baseline CNN lacked both pre-trained knowledge and an attention mechanism. As a result, it needed more data and epochs to approach a decent accuracy, and even then it fell short. It likely learned some useful filters, but without attention it treated all features and regions equally, which is suboptimal for complex multi-class tasks. The baseline's capacity was also limited by design (to avoid overfitting from too many parameters), which constrained its performance on a dataset with 131 classes.

The success of the Attention-Augmented EfficientNet has practical implications for fruit classification tasks. With an accuracy above 97%, such a model could be deployed in real-world applications with confidence. For instance, in an agricultural setting, a mobile robot or drone equipped with a camera could use this model to identify fruits on plants or to sort harvested fruits by type. In supermarkets or farmers' markets, a kiosk or a smartphone app could allow customers to scan an unknown fruit and receive its name and information, which is particularly useful in a country like Colombia where many exotic fruits may be unfamiliar to visitors. The high accuracy ensures that the information provided would be reliable. Moreover, EfficientNetB0's efficiency means the model can run in real-time on devices with limited computational power (such as smartphones or embedded systems), especially if optimized further.

Another aspect to discuss is how this approach could generalize to other classification problems. The combination of a strong CNN backbone with attention modules can be seen as a general template for fine-grained image classification. Fruits are a fine-grained category (many classes with subtle differences), similar to other domains like bird species or plant diseases. The positive results here suggest that applying attention-augmented transfer learning can yield benefits in those domains as well, by focusing the model on the key features that differentiate similar classes.

One potential limitation of our current work is that the Fruits-360 dataset images have uniform backgrounds. Thus, while our model performs exceptionally on these, real-world fruit images might include cluttered backgrounds, multiple fruits in view, or occlusions. In such cases, we anticipate that our model would still perform well due to the attention mechanism (which can help isolate objects of interest), but additional techniques like object detection or segmentation might be needed to first localize fruits in an image. Additionally, for fruits not represented in the training data, the model would naturally be unable to identify them—it is constrained to the classes it learned. 1

VIII. Conclusion and Future Work

In this paper, we presented an Attention-Augmented EfficientNet model for automatic fruit classification, with a focus on Colombia's rich fruit biodiversity as a motivating scenario. By combining the EfficientNetB0 architecture with CBAM attention modules and a custom classification head, we achieved a significant improvement in accuracy on the Fruits-360 dataset compared to a baseline CNN. Our model leverages transfer learning and attention to effectively capture the distinguishing features of 131 fruit classes, obtaining over 97% test accuracy. This demonstrates the advantage of integrating attention mechanisms into deep CNN models for fine-grained image recognition tasks.

For future work, several avenues can be explored to further enhance and apply this system:

- **Real-World Deployment:** We plan to deploy the model in a real-world application. One example is a mobile application or a smart device that can identify fruits using the device's camera. EfficientNetB0's lightweight nature is promising for on-device deployment; we will investigate optimizations like model quantization or pruning to reduce latency and memory usage on smartphones or edge devices.
- **Mobile Implementation:** Building on the above, a dedicated mobile implementation could bring this technology to fruit farmers and consumers in the field. An app could help farmers quickly catalog their produce or help users learn about new fruits. We will focus on creating an intuitive interface and possibly an offline mode (the model running without internet) for use in remote areas.
- **Dataset Expansion:** While Fruits-360 is a comprehensive dataset, it can be expanded to include more fruit varieties, especially those native to specific regions (e.g., South American or Colombian fruits that might not be in the current set). Future work will involve gathering and labeling images of additional fruit types and perhaps different imaging conditions (outdoor images, various backgrounds) to retrain or fine-tune the model. An expanded dataset and retrained model would increase the applicability of the system to real-world scenarios where conditions are less controlled than in Fruits-360.
- **Enhanced Model Features:** We are interested in exploring other attention mechanisms or architectures. For example, one could try using a Vision Transformer or EfficientNetV2, or adding other forms of attention (like self-attention layers) to further improve performance. Additionally, incorporating an object detection component (turning the classifier into a detector) would allow the system to locate and identify fruits in a scene, not just classify cropped images.

Overall, our work shows that integrating attention modules with efficient CNN architectures is a powerful approach for image classification tasks. We believe that with continued improvements and expansions, such models can significantly contribute to preserving and promoting knowledge of fruit biodiversity and streamlining processes in the agricultural domain.

References

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, ”ImageNet classification with deep convolutional neural networks,” in *Adv. Neural Inf. Process. Syst.*, vol. 25, pp. 1097–1105, 2012.

[2] J. Seabrook, ”The Pastry A.I. That Learned to Fight Cancer,” *The New Yorker*, Mar. 18, 2021.

[3] K. Simonyan and A. Zisserman, ”Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.

[4] A. Bhandari, ”Classifying wildlife animals using VGG16 model: A deep learning project,” *Medium*, Mar. 27, 2024.

[5] M. M. Salim *et al.*, ”Automatic detection of potholes using VGG-16 pre-trained network and Convolutional Neural Network,” *Heliyon*, vol. 10, no. 10, p. e30957, 2024.

[6] A. Chouhan and S. R. Singh, ”Real-time drone detection using convolutional neural network-based architectures,” *Int. J. Comput. Appl.*, vol. 181, no. 7, pp. 18–23, 2021.

[7] K. He, X. Zhang, S. Ren, and J. Sun, ”Deep residual learning for image recognition,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[8] G. Wang *et al.*, ”Breast cancer detection via deep learning-based image analysis,” *Sci. Rep.*, vol. 8, no. 1, pp. 1–10, 2018.

[9] M. Tan and Q. Le, ”EfficientNet: Rethinking model scaling for convolutional neural networks,” in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 6105–6114.

[10] T. Al-Fahad, ”EfficientNet for mobile applications in image classification,” *J. Electron. Imaging*, vol. 29, no. 4, pp. 1–9, 2020.