Fruits Recognition Using Deep Learning With Data Augmentation

Jana Ghoniem

Misr International University

CSC344 - Advanced Artificial Intelligence

Dr. Alaa Hamdy

1. Abstract

1.1 Problem Statement

Accurate classification of fruits from images is essential for various real-world applications such as automated grocery checkout systems, quality control, and food sorting. The main challenge lies in handling variations in lighting, background, occlusions, and similar-looking fruit classes. The objective of this project is to train an image classification model that can effectively recognize different types of fruits from the Fruits-360 dataset.

1.2 Proposed Solution

To address the challenges of fruit classification in diverse conditions, a deep learning-based image classification pipeline was implemented, integrating advanced data augmentation techniques and efficient model architectures. By augmenting the dataset with techniques such as background replacement, blur, and noise addition, the model's generalization capability was improved.

Three convolutional neural network (CNN) models were trained and evaluated:

- **EfficientNetB0**
- **ResNet50**
- **MobileNetV2**

Additionally, the use of Squeeze-and-Excitation (SE) blocks was explored to further enhance feature learning. The final comparison included the three models with background augmentation and SE blocks.

2. Dataset Sample

2.1 Dataset Description

The dataset used was the fruits-360 by Mihai Oltean which consists of 137,104 images of 201 fruits, vegetables, nuts and seeds. The dataset was reduced to 53,345 images belonging to 25 fruit classes for the training set, and 17,809 images for the test set.

The issue with the original dataset is that all class images contain a white-only background with a single fruit in the foreground, leading to the models' inability to generalize when it comes to noisy backgrounds or images with multiple fruits.

2.2 Dataset Sample

3. Data Preprocessing & Augmentation

The following preprocessing steps were performed:

- Data augmentation using ***ImageDataGenerator*** with horizontal/vertical flips, rotation, shearing, zooming, etc.
- Custom augmentation functions based on generation of random probabilities:
    - ***replace_background(img)*** – replaces the uniform background with random textures to simulate real-world scenarios.
    - ***add_blur(img)*** – applies Gaussian blur to introduce variation in focus.
    - ***add_noise(img)*** – adds Gaussian noise to simulate sensor irregularities.

These augmentations were combined in the ***custom_preprocess(img, model_name)*** function which also contains the actual ***preprocess_funcs*** of each model architecture.

4. Model Architecture

Pre Trained CNN architectures from TensorFlow's Keras Applications were used to leverage transfer learning. Transfer learning allows a model trained on a large-scale dataset like ImageNet to extract general features applicable to other tasks, thus reducing training time and improving performance with limited data.

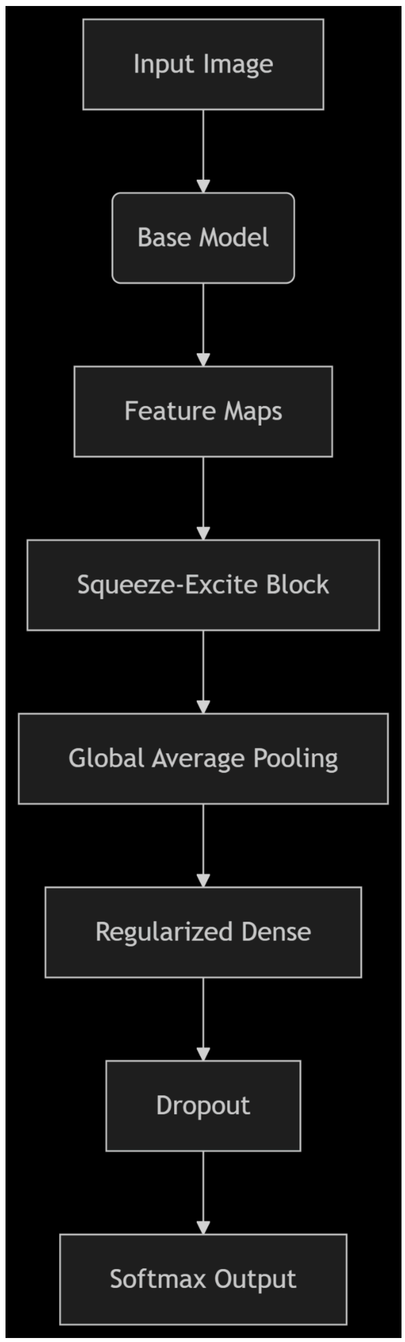
- **EfficientNetB0:** Utilizes a compound scaling method that balances network depth, width, and resolution for optimized performance with fewer parameters.
- **ResNet50:** Introduces residual connections which help propagate gradients through deep networks by adding shortcut connections, thereby preventing vanishing gradient issues.
- **MobileNetV2:** Uses depthwise separable convolutions to reduce parameters and FLOPs while maintaining performance, and employs expansion layers to increase feature dimensionality before applying lightweight depthwise convolutions.

Each model was modified by:

- Freezing the pretrained base layers to preserve learned low-level features.
- Adding a GlobalAveragePooling2D layer to reduce feature maps to a single vector.
- Adding one or more Dense layers with ReLU activation to learn task-specific representations. ReLU (Rectified Linear Unit) introduces non-linearity and helps mitigate the vanishing gradient problem by outputting zero for negative inputs and a linear identity for positive values.
- Inserting a **Squeeze-and-Excitation (SE) block**, which enhances representational power by recalibrating feature channels. The SE block introduces an attention mechanism that helps the model prioritize the most relevant features. It uses global average pooling

followed by fully connected layers and a sigmoid activation to recalibrate the importance of each channel.

- Using a Dropout layer for regularization to prevent overfitting.
- Applying a final Dense layer with softmax activation to perform multi-class classification.



5. Training & Evaluation

The models were trained on two variations of the same dataset: one with the original class distribution which was heavily unbalanced and biased towards class “Apple”, and the other with class weights assigned to balance the data during the training process, enforcing penalties on the misclassification of the minority classes, and forcing the models to focus on learning underrepresented patterns.

5.1 Training Strategy

- Loss function: Categorical Crossentropy, suitable for multi-class classification.
- Optimizer: ***Adam***, which combines the advantages of AdaGrad and RMSProp for efficient training.
- Callbacks:
  - ***EarlyStopping*** monitors validation accuracy and stops training when it stops improving.
  - ***ModelCheckpoint*** saves the model with the best validation performance.
- Training was conducted for up to 15 epochs, with early stopping based on validation accuracy.

5.2 Evaluation Metrics

- **Accuracy:** Proportion of correctly classified instances.
- **Precision:** Proportion of correct positive predictions.
- **Recall:** Proportion of actual positives correctly identified.

6. Evaluation Results

6.1 Imbalanced Dataset Models' Evaluation Results

Evaluation for ResNet50

	precision	recall	f1-score	support
apple	1.00	0.96	0.98	5348
apricot	1.00	1.00	1.00	164
banana	1.00	1.00	1.00	561
blackberrie	1.00	1.00	1.00	600
blueberry	1.00	1.00	1.00	154
cantaloupe	1.00	1.00	1.00	328
cherry	0.94	1.00	0.97	1373
dates	1.00	1.00	1.00	166
fig	1.00	1.00	1.00	234
grape	1.00	0.99	1.00	818
guava	1.00	1.00	1.00	166
kiwi	1.00	1.00	1.00	156
mango	0.99	1.00	1.00	308
orange	1.00	1.00	1.00	160
papaya	1.00	0.90	0.95	164
passion_fruit	1.00	0.89	0.94	166
peach	0.96	0.86	0.91	574
pear	0.85	1.00	0.92	1761
pineapple	1.00	1.00	1.00	329
plum	0.96	1.00	0.98	597
pomegranate	1.00	1.00	1.00	164
raspberry	1.00	1.00	1.00	166
strawberry	1.00	1.00	1.00	410
tomato	1.00	0.97	0.98	2785
watermelon	1.00	1.00	1.00	157
accuracy			0.97	17809
macro avg	0.99	0.98	0.98	17809
weighted avg	0.98	0.97	0.98	17809

```
Evaluation for MobileNetV2
               precision    recall  f1-score   support

        apple       0.89      0.94      0.92      5348
      apricot       0.62      0.70      0.66       164
       banana       1.00      1.00      1.00       561
  blackberrie       0.99      1.00      0.99       600
    blueberry       0.95      1.00      0.97       154
   cantaloupe       1.00      0.99      1.00       328
       cherry       0.99      0.78      0.88      1373
        dates       1.00      1.00      1.00       166
          fig       0.97      0.97      0.97       234
        grape       1.00      0.88      0.93       818
        guava       0.96      0.98      0.97       166
         kiwi       0.95      0.99      0.97       156
        mango       0.64      0.70      0.67       308
       orange       1.00      1.00      1.00       160
       papaya       0.92      0.87      0.89       164
passion_fruit       1.00      0.43      0.60       166
        peach       0.82      0.90      0.86       574
         pear       0.85      0.96      0.90      1761
    pineapple       1.00      1.00      1.00       329
         plum       0.96      0.96      0.96       597
...
     accuracy                           0.93     17809
    macro avg       0.94      0.91      0.92     17809
 weighted avg       0.93      0.93      0.93     17809
```

```
Evaluation for EfficientNetB0
               precision    recall  f1-score   support

        apple       0.93      1.00      0.96      5348
      apricot       1.00      0.90      0.95       164
       banana       1.00      1.00      1.00       561
  blackberrie       1.00      1.00      1.00       600
    blueberry       1.00      1.00      1.00       154
   cantaloupe       1.00      0.94      0.97       328
       cherry       1.00      0.89      0.94      1373
        dates       0.98      1.00      0.99       166
          fig       1.00      1.00      1.00       234
        grape       1.00      1.00      1.00       818
        guava       1.00      1.00      1.00       166
         kiwi       0.90      1.00      0.95       156
        mango       1.00      0.96      0.98       308
       orange       1.00      1.00      1.00       160
       papaya       0.99      1.00      0.99       164
passion_fruit       1.00      0.30      0.46       166
        peach       1.00      0.81      0.90       574
         pear       1.00      0.92      0.96      1761
    pineapple       1.00      1.00      1.00       329
         plum       0.83      0.99      0.90       597
...
     accuracy                           0.97     17809
    macro avg       0.98      0.95      0.96     17809
 weighted avg       0.97      0.97      0.97     17809
```

6.2 Balanced Dataset Models' Evaluation Results

Evaluation for ResNet50

	precision	recall	f1-score	support
0	0.99	0.90	0.95	3307
1	1.00	1.00	1.00	164
2	1.00	1.00	1.00	561
3	1.00	1.00	1.00	600
4	1.00	1.00	1.00	154
5	1.00	1.00	1.00	328
6	0.88	1.00	0.93	1373
7	1.00	1.00	1.00	166
8	1.00	1.00	1.00	234
9	0.96	1.00	0.98	818
10	1.00	1.00	1.00	166
11	1.00	1.00	1.00	156
12	0.80	1.00	0.89	308
13	1.00	1.00	1.00	160
14	1.00	0.98	0.99	164
15	0.83	1.00	0.91	574
16	0.98	0.98	0.98	1761
17	1.00	1.00	1.00	329
18	0.99	1.00	1.00	597
19	0.91	1.00	0.95	164
20	1.00	1.00	1.00	166
21	1.00	1.00	1.00	410
22	1.00	0.94	0.97	2632
23	1.00	1.00	1.00	157
accuracy			0.97	15449
macro avg	0.97	0.99	0.98	15449
weighted avg	0.97	0.97	0.97	15449

Evaluation for MobileNetV2

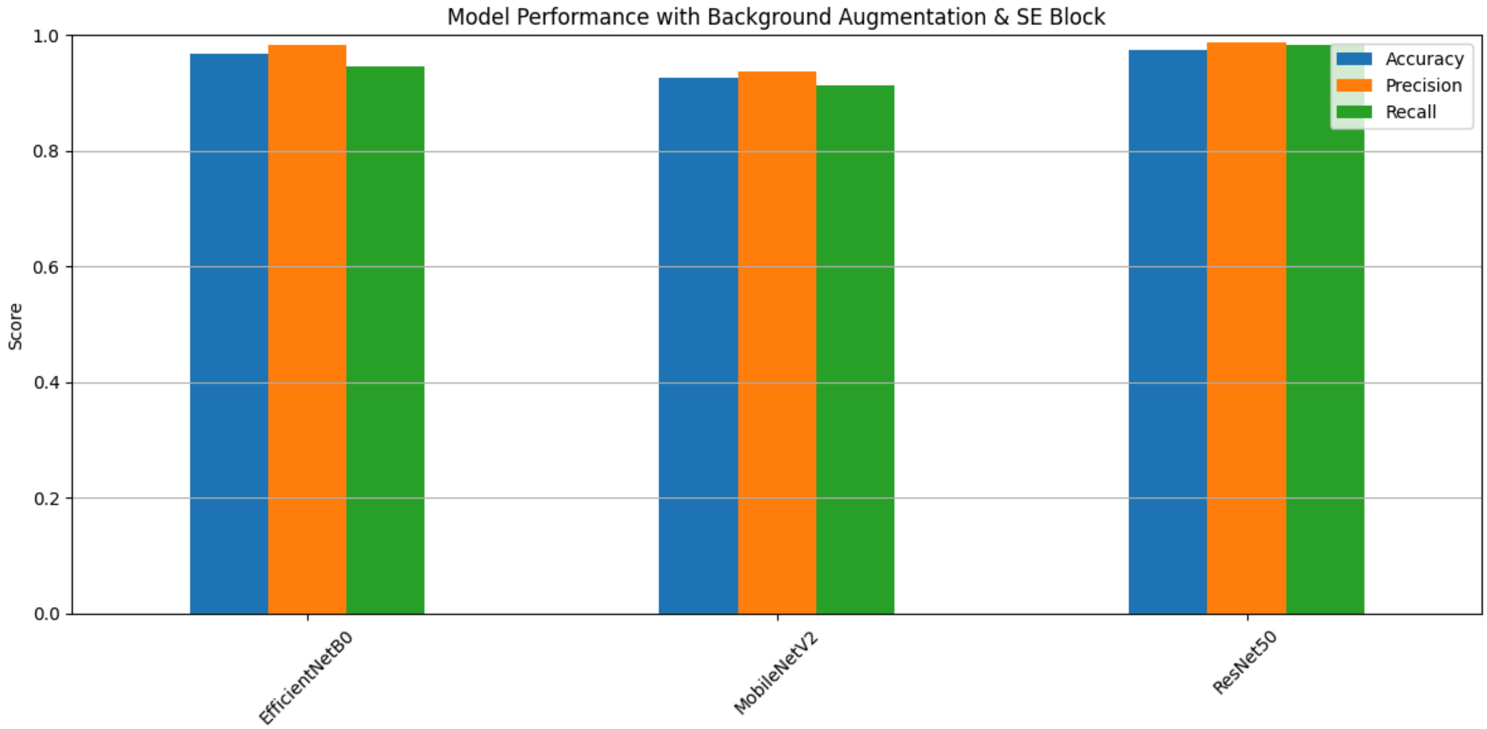
	precision	recall	f1-score	support
0	0.91	0.71	0.80	3307
1	0.57	0.95	0.72	164
2	0.97	1.00	0.98	561
3	0.99	1.00	1.00	600
4	0.78	1.00	0.88	154
5	1.00	0.99	0.99	328
6	0.81	0.81	0.81	1373
7	1.00	0.90	0.95	166
8	0.72	0.99	0.83	234
9	0.92	0.97	0.94	818
10	0.92	1.00	0.96	166
11	0.74	1.00	0.85	156
12	0.54	0.83	0.65	308
13	0.65	1.00	0.78	160
14	0.40	0.93	0.56	164
15	0.78	0.87	0.83	574
16	0.87	0.86	0.87	1761
17	0.98	1.00	0.99	329
18	0.92	0.85	0.88	597
19	0.52	1.00	0.68	164
20	1.00	1.00	1.00	166
21	0.97	1.00	0.99	410
22	0.98	0.86	0.92	2632
23	0.88	1.00	0.93	157
accuracy			0.86	15449
macro avg	0.83	0.94	0.87	15449
weighted avg	0.89	0.86	0.87	15449

Evaluation for EfficientNetB0

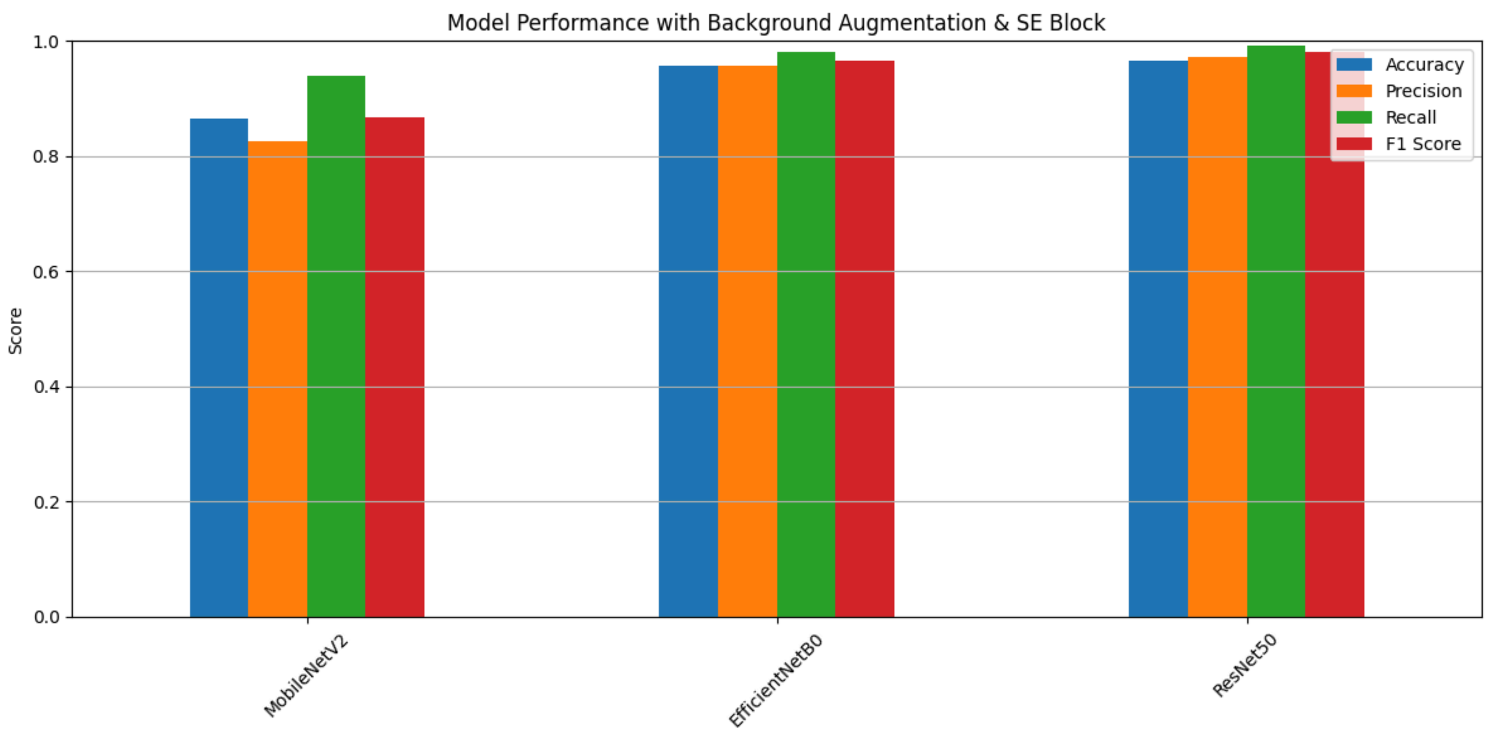
	precision	recall	f1-score	support
0	0.91	0.93	0.92	3307
1	1.00	1.00	1.00	164
2	1.00	1.00	1.00	561
3	1.00	1.00	1.00	600
4	1.00	1.00	1.00	154
5	1.00	0.95	0.98	328
6	0.96	0.90	0.93	1373
7	0.99	1.00	1.00	166
8	0.97	1.00	0.99	234
9	1.00	1.00	1.00	818
10	1.00	1.00	1.00	166
11	0.93	1.00	0.96	156
12	0.92	1.00	0.96	308
13	1.00	1.00	1.00	160
14	0.76	1.00	0.86	164
15	0.87	0.93	0.89	574
16	1.00	0.87	0.93	1761
17	1.00	1.00	1.00	329
18	0.94	0.99	0.96	597
19	0.72	1.00	0.83	164
20	1.00	0.99	1.00	166
21	1.00	1.00	1.00	410
22	0.99	1.00	0.99	2632
23	1.00	1.00	1.00	157
accuracy			0.96	15449
macro avg	0.96	0.98	0.97	15449
weighted avg	0.96	0.96	0.96	15449

7. Models Comparison

7.1 Imbalanced Dataset Model Performance Comparison



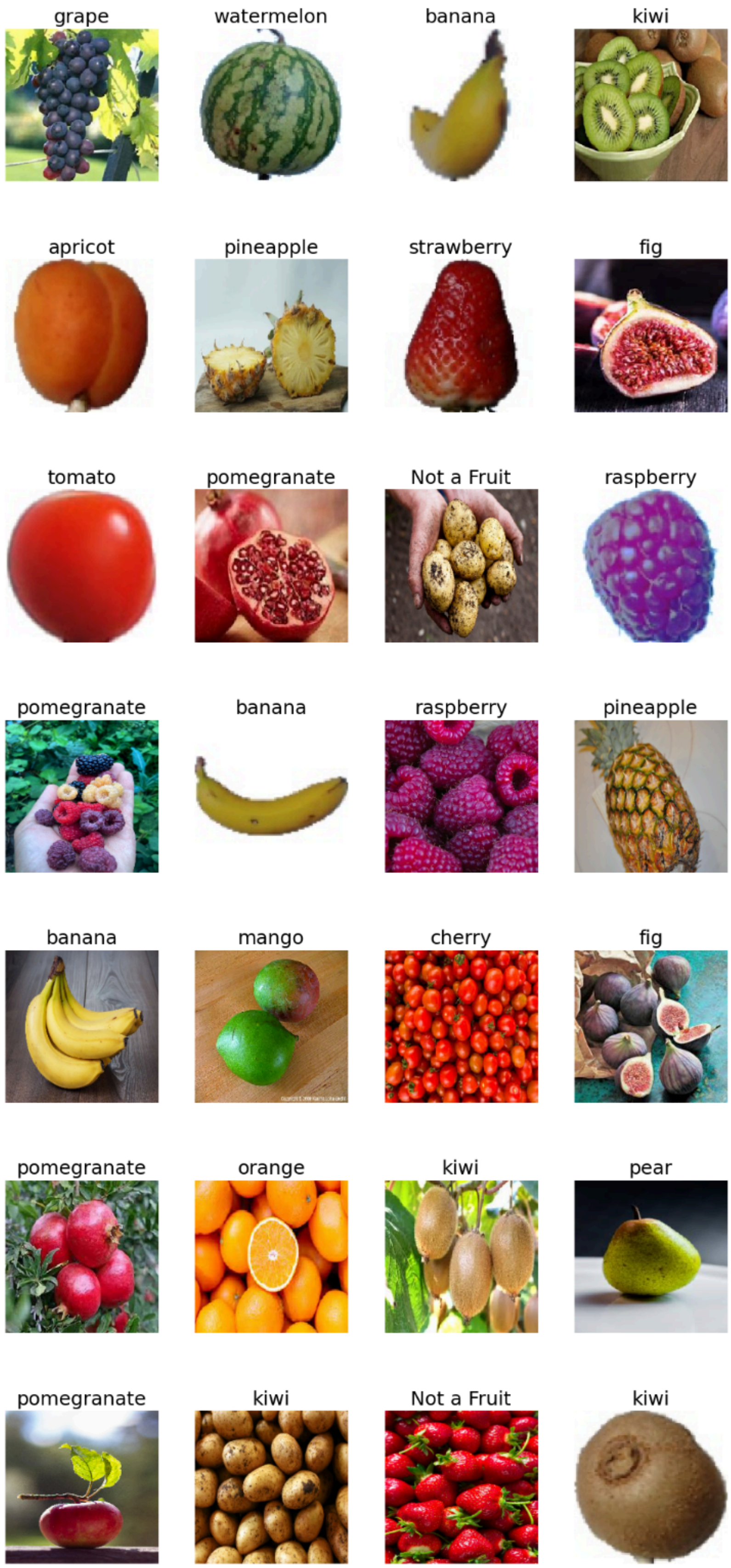
7.2 Balanced Dataset Model Performance Comparison



8. Best Model (EfficientNetB0) Testing

8.1 Unbalanced Dataset Model Testing

8.2 Balanced Dataset Model Testing



9. Conclusion

The project demonstrated that integrating background augmentation and using pre-trained models like EfficientNetB0 significantly improves classification accuracy. The addition of SE blocks further enhanced the model’s ability to focus on important features. These techniques, when combined, form a robust and generalizable fruit classification pipeline.