



# HW1: IMAGE CLASSIFICATION WITH DISTRIBUTION SHIFT

Student:  
Bitā Davoodi  
1000039449

# HW1

## HW1 description:

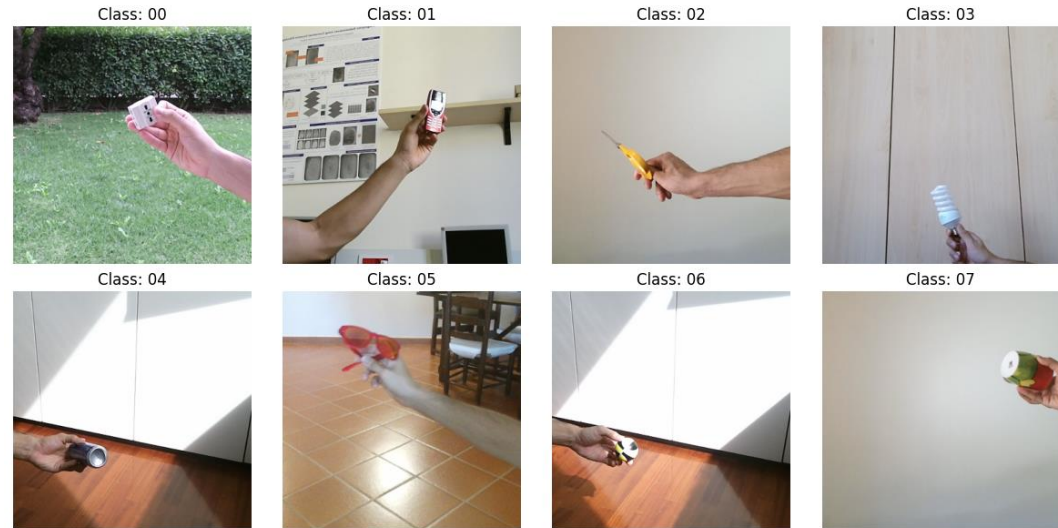
The image:

- **train:** 1600 images with different backgrounds, 8 different classes.
- **test:** 800 images with the same backgrounds, with unknown classes.
- **train.csv, test.csv:** related bounding boxes

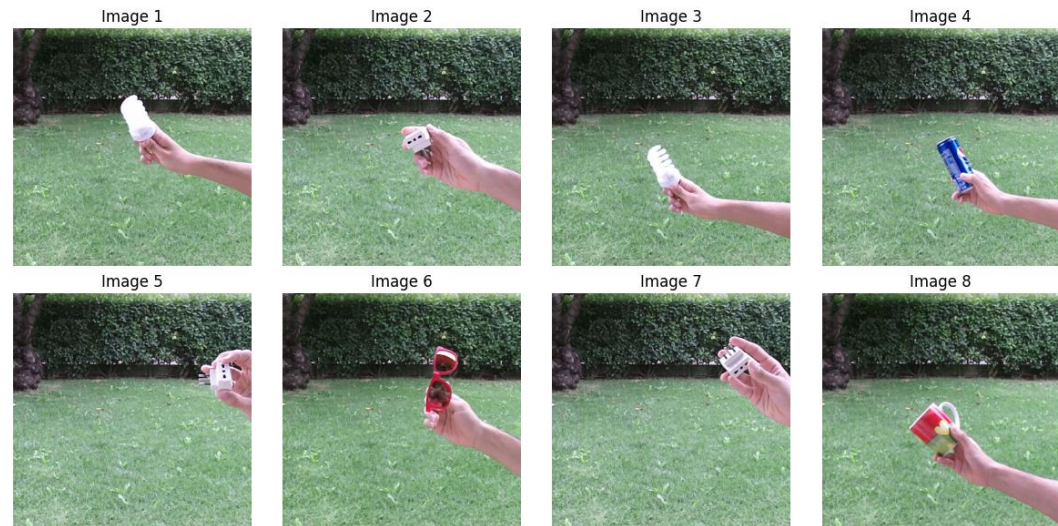
## Difficulty:

Images belonging to class "0" all have the same background, represented as grass and trees. This uniform background causes a challenge for the model because it starts associating the characteristics of class "0" with the grass and trees. As a result, during the initial attempt, the model predicts the entire test set as class "0" due to this association.

## train



## test



+

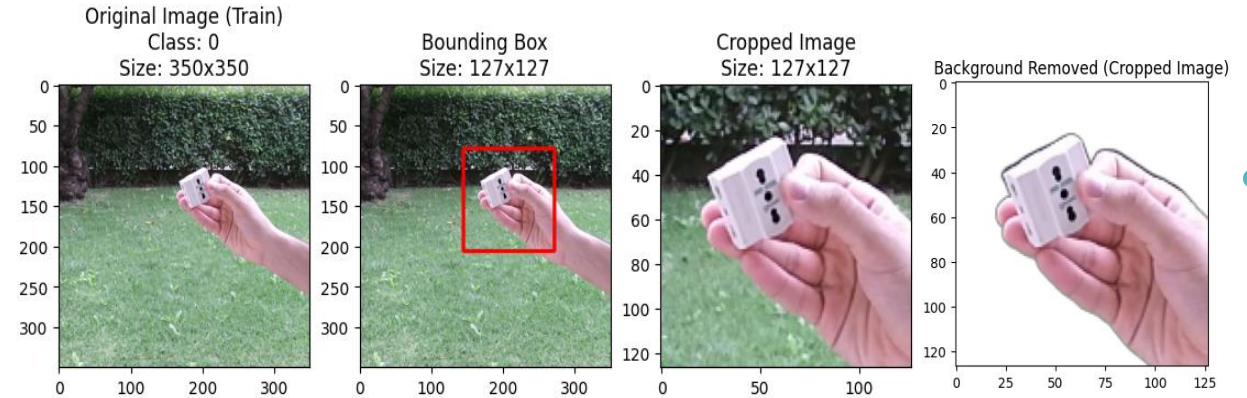
●

# HW1

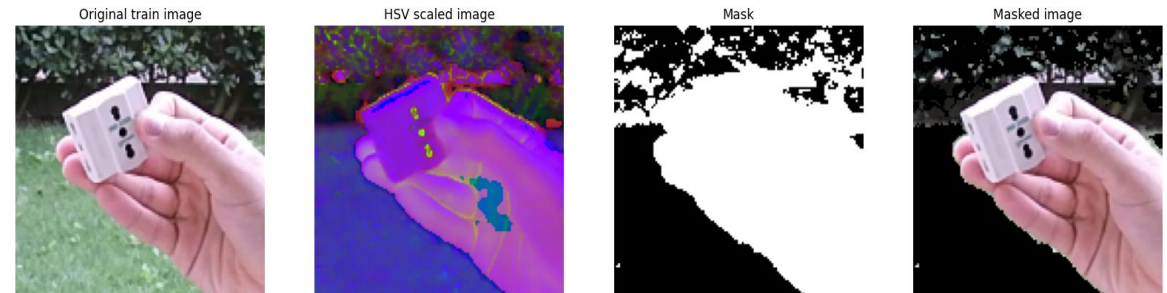


## Inspiring principles for solutions:

- ❑ Cropping the objects using Bounding Boxes and Removing the Background



- ❑ Cropping the objects using Bounding Box and Removing Background and Image Segmentation (HSV Scale, Background Selection and Binary Mask)

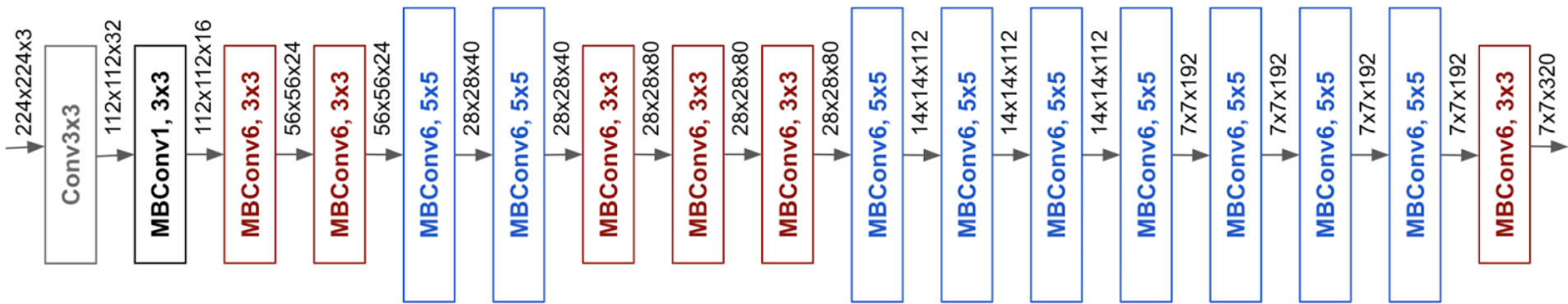


- ❑ Using pre-trained EfficientNetB0
- ❑ Using pre-trained Resnet18

# HW1 – Model Architecture



+



□ Figure of the architecture

# HW1 – Model Architecture



+

●

**Table 1. EfficientNet-B0 baseline network** – Each row describes a stage  $i$  with  $\hat{L}_i$  layers, with input resolution  $\langle \hat{H}_i, \hat{W}_i \rangle$  and output channels  $\hat{C}_i$ . Notations are adopted from equation 2.

| Stage<br>$i$ | Operator<br>$\hat{\mathcal{F}}_i$ | Resolution<br>$\hat{H}_i \times \hat{W}_i$ | #Channels<br>$\hat{C}_i$ | #Layers<br>$\hat{L}_i$ |
|--------------|-----------------------------------|--|--------------------------|------------------------|
| 1            | Conv3x3                           | $224 \times 224$                           | 32                       | 1                      |
| 2            | MBConv1, k3x3                     | $112 \times 112$                           | 16                       | 1                      |
| 3            | MBConv6, k3x3                     | $112 \times 112$                           | 24                       | 2                      |
| 4            | MBConv6, k5x5                     | $56 \times 56$                             | 40                       | 2                      |
| 5            | MBConv6, k3x3                     | $28 \times 28$                             | 80                       | 3                      |
| 6            | MBConv6, k5x5                     | $14 \times 14$                             | 112                      | 3                      |
| 7            | MBConv6, k5x5                     | $14 \times 14$                             | 192                      | 4                      |
| 8            | MBConv6, k3x3                     | $7 \times 7$                               | 320                      | 1                      |
| 9            | Conv1x1 & Pooling & FC            | $7 \times 7$                               | 1280                     | 1                      |



# HW1 – Training procedure



## train:

- ❖ **Resizes** the image to a fixed size of 224x224 pixels.
- ❖ **Randomly flips** with  $p=0.5$ .
- ❖ **GridDistortion**, with limit of distortion between -0.2 and 0.2,  $p=0.3$
- ❖ **Randomly rotates** 10 degrees
- ❖ **Converts** the image from PIL Image format to a **PyTorch tensor**.
- ❖ **Normalizes the input** in the range  $((0.485, 0.456, 0.406), (0.229, 0.224, 0.225))$

## test:

- ❖ **Resizes** the image to a fixed size of 224x224 pixels.
- ❖ **Converts** the image from PIL Image format to a **PyTorch tensor**.
- ❖ **Normalizes the input** in the range  $((0.485, 0.456, 0.406), (0.229, 0.224, 0.225))$

## ▣ Hyparameters:

- ▣ Optimizer: Adam
- ▣ Learning rate: 0.0001
- ▣ Regularization L2, Weight Decay  $5e-3$
- ▣ Batch size: 16

# HW1 – Dataset overview

---



+



## ■ train, val and test splits

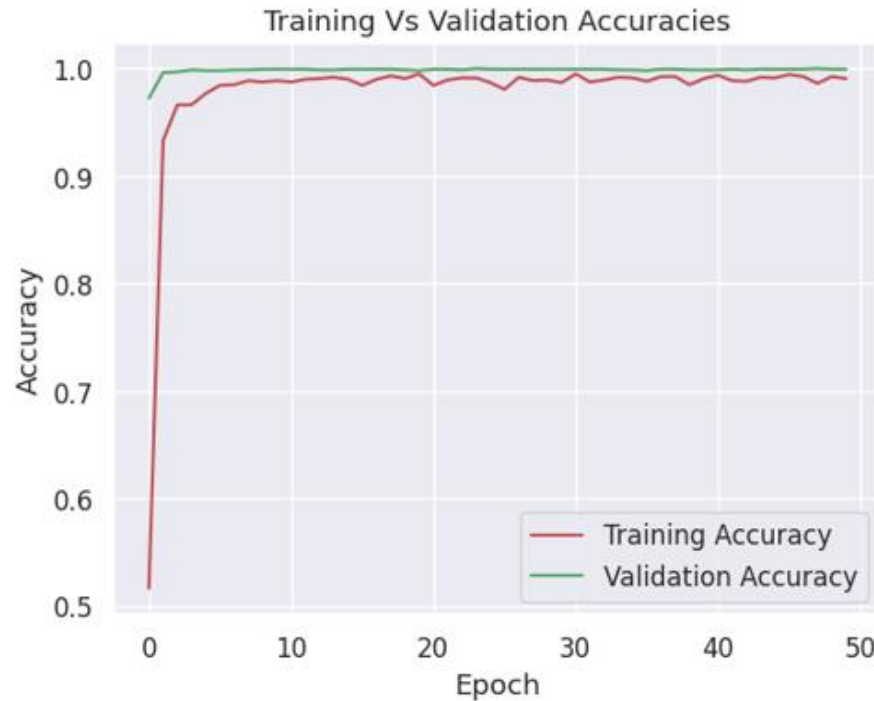
**train: 80 %**    1280   Images

**valid: 10 %**    160   Images

**test: 10 %**    160   Images

However, the split ratio may vary depending on the size and complexity of the dataset.

# HW1 – Experimental Results



Loss and Accuracy functions for EfficientNetB0 with Epoch=10, which has been applied on the train crop masked dataset

Train Loss: 0.0004,  
Train Acc: 0.9984,  
Val Loss: 0.0002,  
Val Acc: 1.0000

loss and accuracy functions for train and test datasets and they are as follows at Epoch=10 and we can observe from Epoch=10 they almost converge:



# HW1 – Experimental Results



|    |    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|----|
| 00 | 20 | 0  | 0  | 0  | 0  | 0  | 0  | 0  |
| 01 | 0  | 18 | 1  | 1  | 0  | 0  | 0  | 0  |
| 02 | 0  | 0  | 20 | 0  | 0  | 0  | 0  | 0  |
| 03 | 0  | 0  | 0  | 19 | 0  | 0  | 0  | 1  |
| 04 | 0  | 0  | 0  | 0  | 20 | 0  | 0  | 0  |
| 05 | 0  | 1  | 1  | 0  | 0  | 18 | 0  | 0  |
| 06 | 0  | 0  | 0  | 0  | 0  | 0  | 20 | 0  |
| 07 | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 20 |
|    | 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 |



Accuracy of the network on the valid images:  
96 %

Accuracy of 00 : 100 %  
Accuracy of 01 : 90 %  
Accuracy of 02 : 100 %  
Accuracy of 03 : 95 %  
Accuracy of 04 : 100 %  
Accuracy of 05 : 90 %  
Accuracy of 06 : 100 %  
Accuracy of 07 : 100 %

Confusion matrix and Grad-CAM with Epoch=50 on the validation set of train\_crop\_np (cropped train images whose background has been removed)

Although the accuracy in this model seems to be high, but the final Accuracy obtained as 52 percent.

# HW1 – Experimental Results

---



+



Table 2: Summary of Model Training Results

| Train model    | Input Data                     | Epoch | Final Accuracy |
|----------------|--------------------------------|-------|----------------|
| EfficientNetB0 | train_crop_nb <sup>1</sup>     | 50    | 52%            |
| EfficientNetB0 | train_crop_masked <sup>2</sup> | 10    | 88.5%          |
| Resnet18       | train_crop_masked              | 10    | 76.5%          |

---

1. Train cropped image with the background removed: No Background

2. Train cropped image with a mask applied

# HW1 – results



## ■ Kaggle challenge results (screenshot of the leaderboard)












+



|   |   |              |                          |
|---|---|--------------|--------------------------|
|    | <b>1000039449_efficientnetb0.csv</b><br>Complete · now    | <b>0.885</b> | <input type="checkbox"/> |
|    | <b>Bitā Davoodi- 1000039449.csv</b><br>Complete · 18m ago | <b>0.765</b> | <input type="checkbox"/> |
|    | <b>submission.csv</b><br>Complete · 7d ago                | <b>0.52</b>  | <input type="checkbox"/> |
|  | <b>Bitā_Davoodi_1000039449.csv</b><br>Complete · 1mo ago  | <b>0.405</b> | <input type="checkbox"/> |

# HW1 – results



| #  | Team                | Members   | Score   | Entries | Last | Join |
|----|---------------------|---|---------|---------|------|------|
| 1  | Turex               |    | 0.99000 | 25      | 1mo  |      |
| 2  | Orazio Pontorno     |    | 0.99000 | 65      | 1mo  |      |
| 3  | Alessandro Platania |    | 0.98250 | 32      | 1mo  |      |
| 4  | Giuseppe Pulino     |    | 0.97000 | 48      | 1mo  |      |
| 5  | Amir Rezadoost      |    | 0.97000 | 9       | 7h   |      |
| 6  | Melika Kalbasi      |    | 0.97000 | 9       | 1d   |      |
| 7  | Fayad               |    | 0.95250 | 6       | 1mo  |      |
| 8  | Gaia Schillaci      |    | 0.95000 | 4       | 1mo  |      |
| 9  | Raza Arain          |  | 0.91500 | 7       | 10d  |      |
| 10 | Manuela Lo Giudice  |  | 0.89500 | 26      | 1d   |      |
| 11 | <b>bita davoodi</b> |  | 0.88500 | 6       |      |      |



Thanks for your attention!