**Project I | Deep Learning: Image Classification with CNN**

Group 1: Daan, Jonh-Baptist and Katy

**PURPOSE**: build a Convolutional Neural Network (CNN) model to classify images from the **CIFAR-10 dataset**, which consists of 60,000 32x32 color images in 10 classes, with 6,000 images per class.

Here are the details on how we address this project:

**I-Data Processing:**

* Class Names: Defined CIFAR-10 class names: ['airplane', 'automobile', 'bird', 'cat', 'deer', 'dog', 'frog', 'horse', 'ship', 'truck']
* Normalization: Converted pixel values to float32 and scaled them to range [0,1]
* One-hot Encoding: Converted labels to categorical format using to\_categorical()

We decided to keep the cleanup done on the previous project

**II – Model Architecture (Sequential Model) & Training**

Sequential Model is the best model for Image classification

Two different optimizers were tested:

* **SGD Optimizers**

**1 – With one layer:**

|  |  |
| --- | --- |
| Accuracy | 0.5741 |
| Loss | 1.2210 |
| Val\_accuracy | 0.5404 |
| Val\_loss | 1.2935 |

**Observations:** The model using SGD with only one layer performs poorly, indicating insufficient model complexity and slower convergence. It suffers from underfitting, unable to represent the data well.

**2 – Adding more layers helped us improve system robustness:**

|  |  |
| --- | --- |
| Accuracy | 0.7249 |
| Loss | 0.7758 |
| Val\_accuracy | 0,7201 |
| Val\_loss | 0.8018 |

**Une image contenant texte, diagramme, ligne, Tracé

Le contenu généré par l’IA peut être incorrect.**

The deeper model works better than the one-layer model. The training was steady, and both the loss and accuracy improved smoothly. The validation results were even a bit better than the training results, which shows the model can generalize well.

* **Adam Optimizers + EarlyStopping:** To stop training if validation accuracy does not improve after 5 epochs:

|  |  |
| --- | --- |
| Accuracy | **0.8970** |
| Loss | **0.2647** |
| Val\_accuracy | **0.8072** |
| Val\_loss | **0.6434** |

**Une image contenant texte, diagramme, Tracé, ligne

Le contenu généré par l’IA peut être incorrect.**

**Observations:**

The **Adam optimizer** significantly **improves both accuracy and convergence speed** compared to the previous SGD model.

Training accuracy reaches **91.4%**, with validation accuracy at **81.6%**, indicating **strong generalization**, even if a gap of ~10% suggests **moderate overfitting**.

The loss and accuracy curves show a **smooth and rapid convergence**, especially in the early epochs, highlighting Adam's effectiveness in adaptive learning.

Validation loss stabilizes after ~10 epochs, but does not improve further

* This model represents the **best overall performance** in our project so far

**III– TESTING THE ADAM MODEL**

We tested the Adam-trained model on 20 randomly selected test images. The model correctly classified 15 out of 20, it gives an idea of how the model performs on unseen data.

Model Evaluation on both Models:

| **Metric** | **Adam Model** | **SGD Model** |
| --- | --- | --- |
| Accuracy | **0.8076** | 0.7201 |
| Precision | **0.8100** | 0.7346 |
| Recall | **0.8076** | 0.7201 |
| F1-score | **0.8076** | 0.7174 |

**IV– TRANSFER LEARNING**

We used VGG16, with our Adam optimizer, because it is a widely used model pretrained on ImageNet. It is easy to integrate into modern deep learning workflows

VGG16 was used with frozen convolutional layers and custom dense layers on top. It achieved **73.27% accuracy.**

Further improvements could be achieved by selectively unfreezing and fine-tuning the top layers of VGG16

**V– APPLICATION**

App under deployment: [https://0f71a478d7a6a6e083.gradio.live](https://0f71a478d7a6a6e083.gradio.live/)

**Conclusion**

We tested several CNN architectures on CIFAR-10 using SGD, Adam, and VGG16. The best model (deep CNN + Adam) achieved 81.6% validation accuracy. Adam significantly improved training speed and generalization vs. SGD. EarlyStopping helped prevent overfitting and stabilized training. VGG16 (transfer learning) reached 73.27% but underperformed due to frozen layers.