Assignment 1

Objective

To build a CNN model that can classify fluid flow as laminar or turbulent provided the velocity distribution.

Break down

1. Importing libraries

pandas(pd) : data manipulation library.
numpy(np) : library for numerical operations.
tensorflow(tf) : for building and training neural networks
scikit-learn : a machine learning library that provides tools for data preprocessing, modeling, and evaluation.
matplotlib : helpful for plotting graphs

2. Preprocessing

The data was cleaned and made ready for the neural network model.
All images were resized to make uniform inputs, and then flattened to form a vector.
The pixel values in this vector were normalised.
Target variable was label encoded.
Categorical data was converted to numerical data.
The data was split into train, test and validation data for further processing.

3. Building and compiling

Model 1
The model has three convolutional layers :
Layer 1 : 32 filters of size 3x3 with ReLu activation
Layer 2 : 64 filters of size 3x3 with ReLu activation
Layer 3 : 128 filters of size 3x3 with ReLu activation
These layers are implemented with max-pooling over 2x2 grids after each layer to simplify computation.
We then flatten the output generated from these layers and pass it into the dense layer (fully connected layer)
This layer has 512 neurones and ReLu activation.
The output layer has two neurons since it is a binary classification problem.

Model 2
The model has five convolutional layers :
Layer 1 : 96 filters of size 11x11 with ReLu activation
Layer 2 : 256 filters of size 5x5 with ReLu activation
Layer 3 : 256 filters of size 3x3 with ReLu activation
Layer 4 : 384 filters of size 3x3 with ReLu activation
Layer 5 : 384 filters of size 3x3 with ReLu activation
Maxpooling over 3x3 grids is applied to simplify data after two convolutional layers. This also controls overfitting.
These variations in kernel sizes help the model focus on smaller local patterns as well as larger global patterns thus improving the accuracy of the model.
Then the data is flattened to enable the fully connected neural network to easily learn it.
There are two Dense layers with 4096 neurons each and ReLU activation, with a dropout of 50%.

The output layer has two fully connected neurons enabling the binary classification using softmax activation which feeds probabilities of each class to the units.

4. Training and evaluating

The model was trained on the dataset provided and accuracy was determined to be 95% for the first model and 80% for the second model.
The simpler CNN model performed much better for this classification task.