# Image Recognition AI

Since I started the portfolio with experimenting with Teachable Machine – so Image Recognition – I think it's a nice wrap up to end the first period with the implementation of an Image Recognition AI.

My original plans to base this on a friend’s research project didn't work out, due to miscommunication, so instead I followed along a Youtube Video. The tutorial is using a dataset from Kaggle for training and testing.

## Introduction

Deep learning is a subset of machine learning.
Learning large amount of data, using multiple layers of neural networks.

Feature learning: Deep learning models, especially CNN can automatically learn hierarchical features from images. Low layers learn edges and textures, while high layers learn more complex structures like shapes and objects.

Common deep learning models for image recognition are:

CNN (Convolutional Neural Network):

- Most commonly used for image recognition
- Convolution layer, pooling layer, fully connected layers

Pre-trained Models (what we’re using)

- Models like ResNet, VGG, Inception
- Have been pre-trained on large data sets

Transfer Learning

- Using retrained models and fine-tuning them on specific data

Follow Tutorial -> Notebook

When I first followed the tutorial, I trained my model with 4 epochs. An epoch is one pass through the training. A common amount for example projects like this is 100, but I did this at Zernike (because I lose focus easily at home), so I didn't really want to wait that long. I later trained it with 50 epochs.

These are the immediate differences I saw:

| | **Model 1** | **Model 2** |
|---|---|---|
| **Epochs** | 4 | 50 |
| **Test Accuracy:** | 48,57% | 79,05% |
| Accuracy | 0.4857 | 0.7905 |
| Loss | 1.7974 | 0.6719 |
| **Jowar Example:** | `predict_img( 'output/test/jowar/image (3).jpeg', model )` | |
| Result | Pearl_millet(bajra) | Jowar |
| Duration | 2000ms | 238ms |
| **Tomato Example:** | `predict_img( 'output/test/tomato/image (6).jpeg', model )` | |
| Result | Cherry | Tomato |
| Duration | 199ms | 247ms |

So, in theory more epochs also mean, that the model gets smarter.

This was indeed the case for my two models. The first model had less accuracy, got both examples wrong and took for the first one two whole seconds. For the second example it was faster, but still wrong.

This was a very nice project and I will try to incorporate more like this in the second period.