The Model Used:

A simple **Convolutional Neural Network (CNN)** with three output classes for:

1. Fake Oracle Character
2. Real Oracle Character
3. Roboccon Logo

The network has:

- Two convolution layers to detect patterns
- Pooling layers to reduce image size
- Three fully connected layers for classification

After convolutions and pooling, the images flattens to 21,384 features before classification.

Why This Model

CNNs work well for images because they detect edges and patterns automatically. This custom design:

- Fits exact image size (no resizing issues)
- Is small and fast
- Handles three specific classes perfectly
- Avoids heavy pretrained models that don't matchmdata

How It Was Trained

1. Images organized in folders: train_path/Fake/, train_path/Real/, train_path/Roboccon/
2. Training images get random flips and 5° rotations to handle variations
3. Both train/test images resized to 140x116 and normalized
4. Trained for 70 epochs using SGD optimizer (lr=0.001, momentum=0.9)
5. Batch size 1 to be safe with memory
6. Saved as trained_net3.pth after each epoch

The training loop processes one image at a time, calculates loss, updates weights, and prints average loss per epoch.