

**CM23019**

## **Digit Recognition (MNIST Dataset)**

**Output:**

Digit Recognition CNN

Train CNN on MNIST digits and predict via console.

```
Loading MNIST dataset...  
Training started...  
Epoch 1: loss=2.3277, val_loss=2.2780,  
acc=0.1000, val_acc=0.1500  
Epoch 2: loss=2.2856, val_loss=2.2899,  
acc=0.1300, val_acc=0.1500  
Epoch 3: loss=2.2748, val_loss=2.2870,  
acc=0.1200, val_acc=0.1500  
Training completed.  
Use predictX(index) in console for predictions.  
> predictX(8)  
Predicted: 3, True Label: 9  
< undefined  
> predictX(4)  
Predicted: 3, True Label: 1  
< undefined  
> |
```

### Lab 3: MNIST Digit Recognition

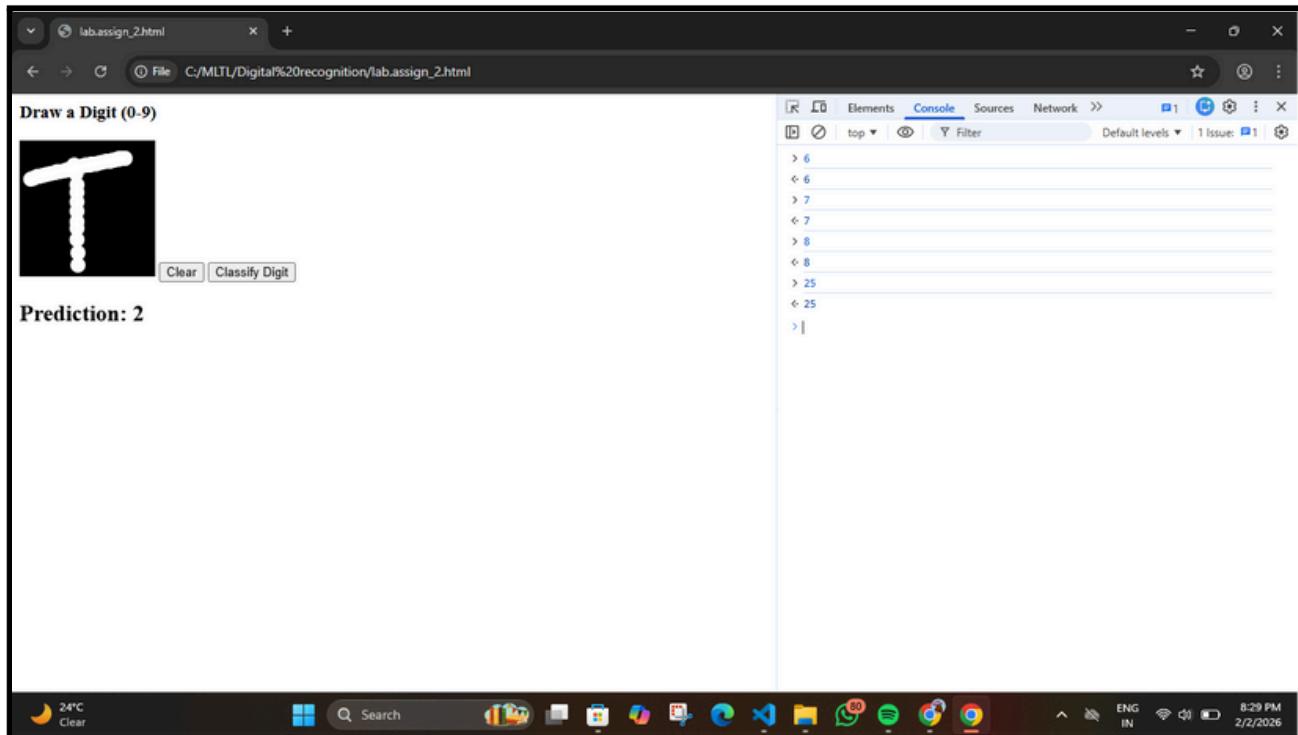
**CNN Model**

Final Accuracy: 98.2%  
Epochs: 5

**Dense Model**

Final Accuracy: 92.5%  
Reason: CNNs capture spatial patterns better!

```
> 5  
< 5  
> 9  
< 9  
> 8  
< 8  
> |
```



A screenshot of a web browser window titled "MINIST: CNN vs Dense Comparison". The page features a section titled "Task 3: Performance Comparison" with the sub-instruction: "Compare a Convolutional Neural Network (CNN) against a Simple Dense Network after 5 epochs." Below this, there are two cards: "Convolutional Network (CNN)" and "Simple Dense Network". Both cards show an architecture summary and a status indicator. A "Run Comparison Test" button is located above the cards. To the right of the cards is a table comparing the two models. The table has three columns: "Feature", "Simple Dense Network", and "CNN (Convolutional)". The first row compares "Spatial Awareness" with "None (treats image as a flat)" for the Dense Network and "High (detects edges, shapes, and" for the CNN. The browser's developer tools are open, showing the same log entries as the previous screenshot. The operating system taskbar at the bottom shows various pinned icons and the date/time as 2/2/2026.

Feature	Simple Dense Network	CNN (Convolutional)
Spatial Awareness	None (treats image as a flat)	High (detects edges, shapes, and)