

```
In [13]: from tensorflow.keras.datasets import mnist
from tensorflow.keras.utils import to_categorical

(x_train, y_train), (x_test, y_test) = mnist.load_data()

x_train = x_train.astype('float32') / 255.0
x_test = x_test.astype('float32') / 255.0

x_train = x_train.reshape(-1, 28, 28, 1)
x_test = x_test.reshape(-1, 28, 28, 1)

y_train_cat = to_categorical(y_train, 10)
y_test_cat = to_categorical(y_test, 10)
```

Downloading data from <https://storage.googleapis.com/tensorflow/tf-keras-datasets/mnist.npz>  
11490434/11490434 ————— 7s 1us/step

```
In [14]: from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Flatten, Dense, Dropout

model = Sequential([
    Conv2D(32, (3, 3), activation='relu', input_shape=(28, 28, 1)),
    MaxPooling2D(pool_size=(2, 2)),
    Conv2D(64, (3, 3), activation='relu'),
    MaxPooling2D(pool_size=(2, 2)),
    Flatten(),
    Dense(128, activation='relu'),
    Dropout(0.5),
    Dense(10, activation='softmax')
])

model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])

history = model.fit(x_train, y_train_cat, epochs=5, batch_size=128, validation_split=0.1)
```

C:\Users\Admin\anaconda3\Lib\site-packages\keras\src\layers\convolutional\base\_conv.py:107: UserWarning: Do not pass an `input\_shape`/`input\_dim` argument to a layer. When using Sequential models, prefer using an `Input(shape)` object as the first layer in the model instead.  
super().\_\_init\_\_(activity\_regularizer=activity\_regularizer, \*\*kwargs)

Epoch 1/5

422/422 ————— 23s 43ms/step - accuracy: 0.7917 - loss: 0.6518 - val\_accuracy: 0.9833 - val\_loss: 0.0620

Epoch 2/5

422/422 ————— 20s 48ms/step - accuracy: 0.9664 - loss: 0.1150 - val\_accuracy: 0.9877 - val\_loss: 0.0423

Epoch 3/5

422/422 ————— 18s 41ms/step - accuracy: 0.9771 - loss: 0.0776 - val\_accuracy: 0.9882 - val\_loss: 0.0426

Epoch 4/5

422/422 ————— 18s 42ms/step - accuracy: 0.9820 - loss: 0.0608 - val\_accuracy: 0.9883 - val\_loss: 0.0379

Epoch 5/5

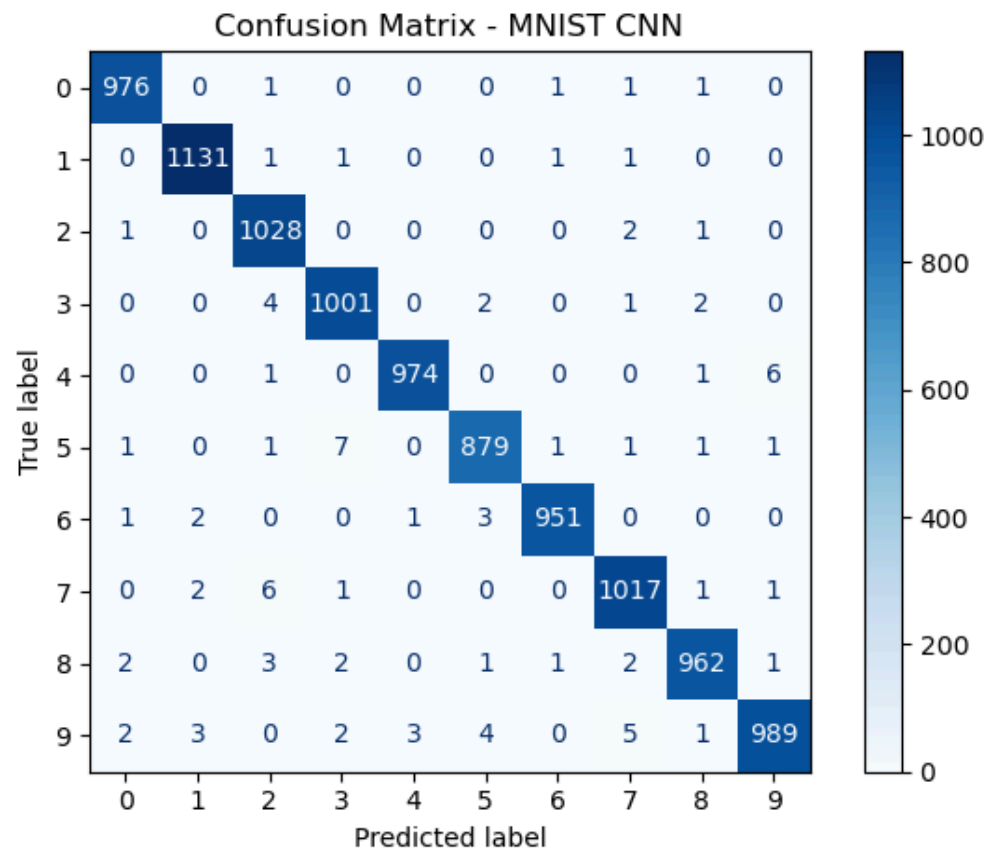
422/422 ————— 18s 42ms/step - accuracy: 0.9847 - loss: 0.0503 - val\_accuracy: 0.9903 - val\_loss: 0.0332

```
In [15]: import numpy as np
from sklearn.metrics import confusion_matrix, ConfusionMatrixDisplay
import matplotlib.pyplot as plt

y_pred_probs = model.predict(x_test)
y_pred_classes = np.argmax(y_pred_probs, axis=1)

cm = confusion_matrix(y_test, y_pred_classes)
disp = ConfusionMatrixDisplay(confusion_matrix=cm, display_labels=range(10))
disp.plot(cmap='Blues')
plt.title("Confusion Matrix - MNIST CNN")
plt.tight_layout()
plt.show()
```

313/313 ————— 3s 9ms/step



```
In [16]: test_loss, test_acc = model.evaluate(x_test, y_test_cat)
         print(f"Test Accuracy: {test_acc:.4f}")

313/313 ————— 3s 10ms/step - accuracy: 0.9888 - loss: 0.0332
Test Accuracy: 0.9908

In [ ]:
```