

Name : SHEIK PAREETH

Deep learning

CNN

In [1]:

```
import tensorflow as tf
(x_train, y_train), (x_test, y_test) = tf.keras.datasets.mnist.load_data()
```

In [2]:

```
x_train.shape
```

Out[2]:

```
(60000, 28, 28)
```

In [3]:

```
x_test.shape
```

Out[3]:

```
(10000, 28, 28)
```

In [9]:

```
x_train=x_train.reshape(x_train.shape[0],28,28,1)
x_test=x_test.reshape(x_test.shape[0],28,28,1)
input_shape = (28, 28, 1)
```

In [12]:

```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Conv2D, Dropout, Flatten, MaxPooling2D
model=Sequential()
model.add(Conv2D(28,kernel_size=(3,3),input_shape=input_shape))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Flatten())
model.add(Dense(128,activation=tf.nn.relu))
model.add(Dropout(0.2))
model.add(Dense(10,activation=tf.nn.softmax))
```

In [19]:

```
model.compile(loss='sparse_categorical_crossentropy',optimizer="adam",metrics = ['accuracy'])
```

In [16]:

```
x_train = x_train.astype('float32')
x_test = x_test.astype('float32')
```

In [17]:

```
x_train /= 255  
x_test /= 255
```

In [20]:

```
model.fit(x=x_train,y=y_train, epochs=10)
```

```
Epoch 1/10  
1875/1875 [=====] - 74s 39ms/step - loss: 0.2222 -  
accuracy: 0.93192s - 1 - ETA: 1s  
Epoch 2/10  
1875/1875 [=====] - 70s 37ms/step - loss: 0.0884 -  
accuracy: 0.9731  
Epoch 3/10  
1875/1875 [=====] - 73s 39ms/step - loss: 0.0626 -  
accuracy: 0.9801  
Epoch 4/10  
1875/1875 [=====] - 72s 38ms/step - loss: 0.0457 -  
accuracy: 0.9854  
Epoch 5/10  
1875/1875 [=====] - 72s 38ms/step - loss: 0.0387 -  
accuracy: 0.9870  
Epoch 6/10  
1875/1875 [=====] - 73s 39ms/step - loss: 0.0319 -  
accuracy: 0.9894  
Epoch 7/10  
1875/1875 [=====] - 72s 38ms/step - loss: 0.0262 -  
accuracy: 0.9916  
Epoch 8/10  
1875/1875 [=====] - 73s 39ms/step - loss: 0.0235 -  
accuracy: 0.9923  
Epoch 9/10  
1875/1875 [=====] - 72s 39ms/step - loss: 0.0205 -  
accuracy: 0.9931  
Epoch 10/10  
1875/1875 [=====] - 78s 42ms/step - loss: 0.0191 -  
accuracy: 0.9937
```

Out[20]:

```
<keras.callbacks.History at 0x1b5613d43c8>
```

In [21]:

```
model.evaluate(x_test, y_test)
```

```
313/313 [=====] - 8s 21ms/step - loss: 0.0582 - acc  
uracy: 0.9863
```

Out[21]:

```
[0.05821174755692482, 0.986299991607666]
```

In [23]:

```
import matplotlib.pyplot as plt
image_index = 6900
plt.imshow(x_test[image_index].reshape(28, 28), cmap='Greys')
predict = x_test[image_index].reshape(28, 28)
pred = model.predict(x_test[image_index].reshape(1, 28, 28, 1))
print(pred.argmax())
```

6

