Implement VGG on MNIST dataset

In [1]:
```
import numpy as np
import matplotlib.pyplot as plt
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras import layers
from tensorflow.keras.layers import Embedding,Flatten,Dense,Dropout
from tensorflow.image import grayscale_to_rgb,resize
from tensorflow.keras.applications import VGG19
from tensorflow.keras.models import Model,Sequential
from tensorflow.keras.utils import to_categorical
from tensorflow.keras.datasets import mnist
```

```
WARNING:tensorflow:From D:\JUPYTER FOLDER\Lib\site-packages\keras\src\losses.py:2976: The name tf.losses.sparse_softmax_cross
_entropy is deprecated. Please use tf.compat.v1.losses.sparse_softmax_cross_entropy instead.
```

In [2]:
```
(train_images,train_labels),(test_images,test_labels)=mnist.load_data()
train_images=(train_images.astype('float32')/255.0).reshape(-1,28,28,1)
test_images=(test_images.astype('float32')/255.0).reshape(-1,28,28,1)
train_labels,test_labels=to_categorical(train_labels),to_categorical(test_labels)
```

In [3]:
```
base_model=VGG19(weights='imagenet',include_top=False,input_shape=(48,48,3))
for layer in base_model.layers:
    layer.trainable=False
```

```
WARNING:tensorflow:From D:\JUPYTER FOLDER\Lib\site-packages\keras\src\backend.py:1398: The name tf.executing_eagerly_outside_
functions is deprecated. Please use tf.compat.v1.executing_eagerly_outside_functions instead.

WARNING:tensorflow:From D:\JUPYTER FOLDER\Lib\site-packages\keras\src\layers\pooling\max_pooling2d.py:161: The name tf.nn.max
_pool is deprecated. Please use tf.nn.max_pool2d instead.
```

In [4]:
```
odel(inputs=base_model.input,outputs=Dense(10,activation='softmax')(Dense(1024,activation='relu')(Flatten()(base_model.output))
```

In [5]:
```
train_images_vgg=grayscale_to_rgb(resize(train_images,(48,48),method='bicubic'))
test_images_vgg=grayscale_to_rgb(resize(test_images,(48,48),method='bicubic'))
```

In [6]:
```
model.compile(metrics=['accuracy'],loss='categorical_crossentropy',optimizer='adam')
```

```
WARNING:tensorflow:From D:\JUPYTER FOLDER\Lib\site-packages\keras\src\optimizers\__init__.py:309: The name tf.train.Optimizer
is deprecated. Please use tf.compat.v1.train.Optimizer instead.
```

In [ ]:
```

```

In [7]:
```
history=model.fit(train_images_vgg,train_labels,epochs=1,validation_data=(test_images_vgg,test_labels))
```

```
WARNING:tensorflow:From D:\JUPYTER FOLDER\Lib\site-packages\keras\src\utils\tf_utils.py:492: The name tf.ragged.RaggedTensorV
alue is deprecated. Please use tf.compat.v1.ragged.RaggedTensorValue instead.

WARNING:tensorflow:From D:\JUPYTER FOLDER\Lib\site-packages\keras\src\engine\base_layer_utils.py:384: The name tf.executing_e
agerly_outside_functions is deprecated. Please use tf.compat.v1.executing_eagerly_outside_functions instead.

1875/1875 [==============================] - 476s 253ms/step - loss: 0.2075 - accuracy: 0.9376 - val_loss: 0.1381 - val_accur
acy: 0.9551
```

In [8]:
```
model.save('D:/codes/vgg.h5')
```

```
D:\JUPYTER FOLDER\Lib\site-packages\keras\src\engine\training.py:3103: UserWarning: You are saving your model as an HDF5 file
via `model.save()`. This file format is considered legacy. We recommend using instead the native Keras format, e.g. `model.sa
ve('my_model.keras')`.
  saving_api.save_model(
```

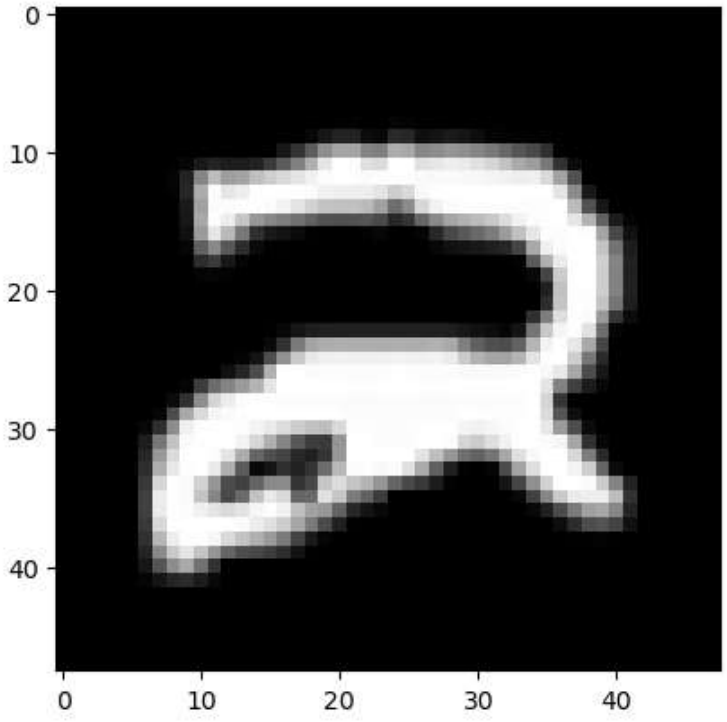
In [ ]:
```

```

In [ ]:
```

```

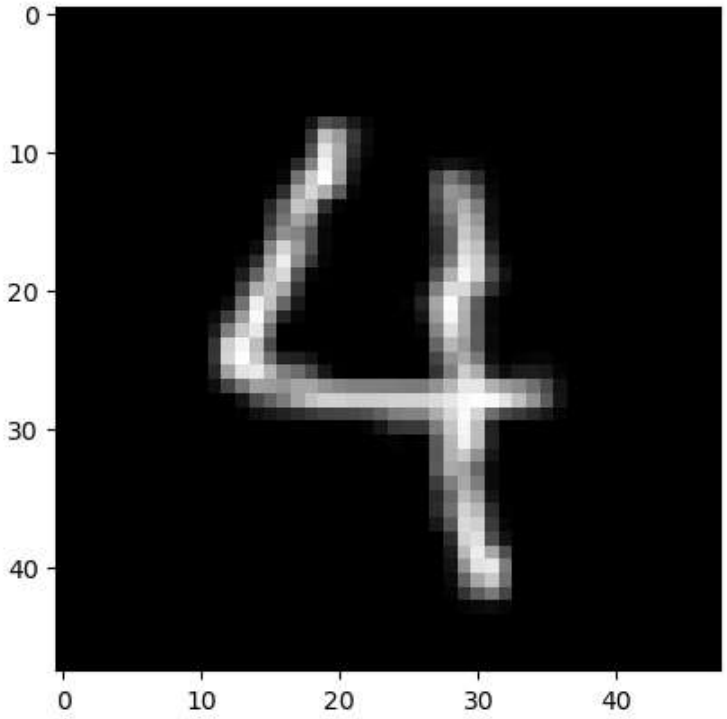
In [ ]:
```

```

In [13]:
```
for _ in range(3):
    index = np.random.randint(0, len(test_images_vgg))
    input_image = test_images_vgg[index]  # Use the resized RGB image
    input_image = np.expand_dims(input_image, axis=0)  # Add batch dimension
    pred = model.predict(input_image)
    predicted = np.argmax(pred)
    print(predicted)
    plt.imshow(test_images_vgg[index])
    plt.show()
```

```
1/1 [==============================] - 0s 60ms/step

Clipping input data to the valid range for imshow with RGB data ([0..1] for floats or [0..255] for integers).

2
```



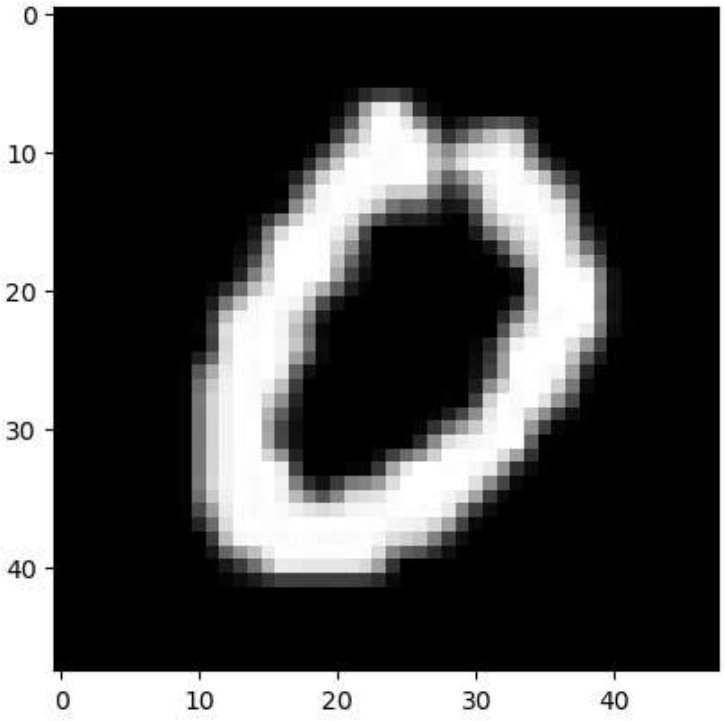

```
1/1 [==============================] - 0s 60ms/step

Clipping input data to the valid range for imshow with RGB data ([0..1] for floats or [0..255] for integers).

4
```




```
1/1 [==============================] - 0s 59ms/step
```

```
Clipping input data to the valid range for imshow with RGB data ([0..1] for floats or [0..255] for integers).
```

```
0
```



In [ ]: