```
import numpy as np
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
model = load_model("Gesture.h5")
path = r'C:\Users\mrith\Desktop\dataset\test\0\0.jpg'
train_datagen = ImageDataGenerator(rescale=1./255,
                                   shear_range=0.2,
                                   zoom_range=0.2,
                                   horizontal_flip=True)
test_datagen=ImageDataGenerator(rescale=1./255)
x_train = 
train_datagen.flow_from_directory(r'C:\Users\mrith\Desktop\dataset\train',
                                            target_size=(64, 64),
                                            batch_size=3,
                                            color_mode='grayscale',
                                            class_mode='categorical')
x_test = 
test_datagen.flow_from_directory(r'C:\Users\mrith\Desktop\dataset\test',
                                          target_size=(64, 64),
                                          batch_size=3,
                                          color_mode='grayscale',
                                          class_mode='categorical')
```

```
Found 594 images belonging to 6 classes.
Found 30 images belonging to 6 classes.
```

In [89]:

```
%pylab inline
import matplotlib.pyplot as plt
import matplotlib.image as mpimg
imgs = mpimg.imread(path)
imgplot = plt.imshow(imgs)
plt.show()
```

```
Populating the interactive namespace from numpy and matplotlib
```

In [90]:

```
#loading of the image
img = image.load_img(path,
                     color_mode='grayscale',
                     target_size= (64,64))
x = image.img_to_array(img)#image to array
x.shape
```

Out[90]:

```
(64, 64, 1)
```

In [91]:

```
type(x)
```

Out[91]:

```
numpy.ndarray
```

In [92]:

```
#changing the shape
x = np.expand_dims(x,axis = 0)
```

In [93]:

```
x.shape
```

Out[93]:

```
(1, 64, 64, 1)
```

In [95]:

```
pred = model.predict(x)
pred
```

```
1/1 [==============================] - 0s 84ms/step
```

Out[95]:

```
array([[1., 0., 0., 0., 0., 0.]], dtype=float32)
```

In [96]:

```
x_test.class_indices
```

Out[96]:

```
{'0': 0, '1': 1, '2': 2, '3': 3, '4': 4, '5': 5}
```

In [97]:

```
index=['0','1','2','3','4','5']
```

In [98]:

```
index[np.argmax(pred)]
```

Out[98]:

```
'0'
```