In [2]:

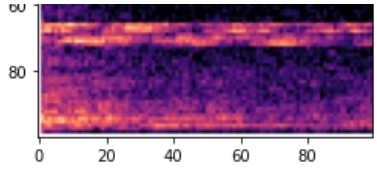
```
import tensorflow as tf #Derin Ağ için tensorflow u import ettik
from tensorflow.keras.preprocessing.image import ImageDataGenerator #rescale edip oranlamak ve test değerlerini a

base_dir = 'C:/Users/s7522/Desktop/UrbanSound8K/spectograms/' #spectogramların olduğu path i aldık.

train_datagen = ImageDataGenerator(rescale = 1./255, validation_split = 0.1) #Her bir görseli 255 e bölerek numpy
test_datagen = ImageDataGenerator(rescale = 1./255, validation_split = 0.1)

train_datagen = train_datagen.flow_from_directory(base_dir, target_size = (100,100), subset = "training", batch_s
test_datagen = test_datagen.flow_from_directory(base_dir, target_size = (100,100), subset = "validation", batch_s
```

```
Found 7861 images belonging to 10 classes.
Found 871 images belonging to 10 classes.
```

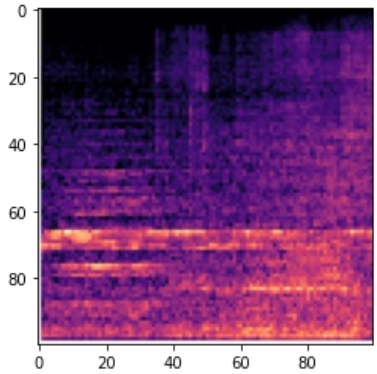
In [3]:

```
import matplotlib.pyplot as plt #5 tane spectogram üzerinden test

for _ in range(5):
    img, label = test_datagen.next() #test_datagen de iki tane çıktı vardır image ve label
    print(img.shape) #Gelen resimlerin shape ini gördük.
    plt.imshow(img[0])
    print(img[0]) #img ve label gördük
    print(label[0])
    plt.show()
```

```
(2, 100, 100, 3)
[[[1.         1.         1.        ]
  [0.         0.         0.01176471]
  [0.         0.         0.01176471]
  ...
  [0.         0.         0.01176471]
  [0.         0.         0.01176471]
  [0.         0.         0.01176471]]

 [[1.         1.         1.        ]
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]
  ...
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]]

 [[1.         1.         1.        ]
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]
  ...
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]]

 ...

 [[1.         1.         1.        ]
  [0.80392164 0.2509804  0.4431373 ]
  [0.82745105 0.2627451  0.43137258]
  ...
  [0.19215688 0.06666667 0.39607847]
  [0.5137255  0.14901961 0.5058824 ]
  [0.5568628  0.16470589 0.5058824 ]]

 [[1.         1.         1.        ]
  [0.70980394 0.21176472 0.43529415]
  [0.20000002 0.05882353 0.39607847]
  ...
  [0.02745098 0.02352941 0.10980393]
  [0.06666667 0.04705883 0.18039216]
  [0.12156864 0.0627451  0.28627452]]

 [[1.         1.         1.        ]
  [1.         1.         1.        ]
  [1.         1.         1.        ]
  ...
  [1.         1.         1.        ]
  [1.         1.         1.        ]
  [1.         1.         1.        ]]]
[0. 0. 0. 0. 0. 0. 0. 0. 1. 0.]
```

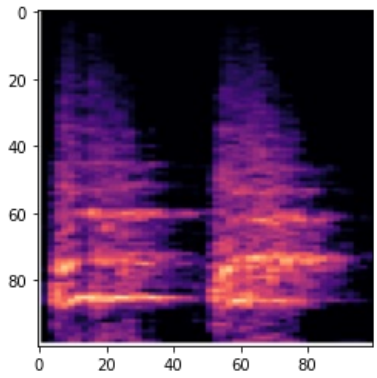



```
(2, 100, 100, 3)
[[[1.         1.         1.        ]
  [0.         0.         0.01176471]
  [0.         0.         0.01176471]
  ...
  [0.         0.         0.01176471]
  [0.         0.         0.01176471]
  [0.         0.         0.01176471]]

 [[1.         1.         1.        ]
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]
  ...
  [0.         0.         0.01568628]
  [0.03137255 0.02745098 0.11764707]
  [0.00784314 0.00784314 0.0509804 ]]

 [[1.         1.         1.        ]
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]
  ...
  [0.02352941 0.01960784 0.09411766]
  [0.02745098 0.02352941 0.10980393]
  [0.03137255 0.02745098 0.11764707]]

 ...

 [[1.         1.         1.        ]
  [0.8588236  0.2784314  0.4156863 ]
  [0.9176471  0.3372549  0.3803922 ]
  ...
  [0.8980393  0.3137255  0.3921569 ]
  [0.8352942  0.26666668 0.427451  ]
  [0.6        0.1764706  0.5019608 ]]

 [[1.         1.         1.        ]
  [0.53333336 0.15294118 0.47450984]
  [0.3921569  0.10196079 0.47450984]
  ...
  [0.49411768 0.14117648 0.47450984]
  [0.47450984 0.13333334 0.47450984]
  [0.5137255  0.14901961 0.47450984]]

 [[1.         1.         1.        ]
  [1.         1.         1.        ]
  [1.         1.         1.        ]
  ...
  [1.         1.         1.        ]
  [1.         1.         1.        ]
  [1.         1.         1.        ]]]
[0. 0. 0. 0. 0. 0. 0. 1. 0.]
```



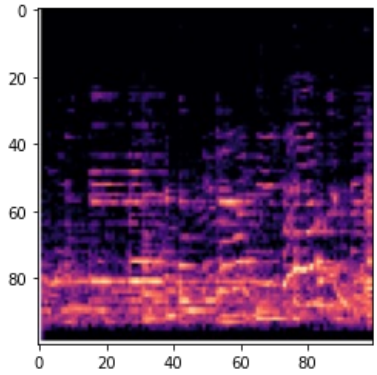

```
(2, 100, 100, 3)
[[[1.         1.         1.        ]
  [0.         0.         0.01176471]
  [0.         0.         0.01176471]
  ...
  [0.         0.         0.01176471]
  [0.         0.         0.01176471]
  [0.         0.         0.01176471]]

 [[1.         1.         1.        ]
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]
  ...
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]
```

```
  [0.         0.         0.01568628]]

 [[1.         1.         1.        ]
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]
  ...
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]]

 ...

 [[1.         1.         1.        ]
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]
  ...
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]]

 [[1.         1.         1.        ]
  [0.         0.         0.01176471]
  [0.         0.         0.01176471]
  ...
  [0.03529412 0.02745098 0.1254902 ]
  [0.         0.         0.01176471]
  [0.         0.         0.01176471]]

 [[1.         1.         1.        ]
  [1.         1.         1.        ]
  [1.         1.         1.        ]
  ...
  [1.         1.         1.        ]
  [1.         1.         1.        ]
  [1.         1.         1.        ]]]
[0. 0. 0. 1. 0. 0. 0. 0. 0. 0.]
```

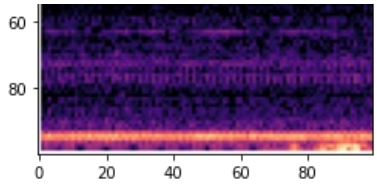



```
(2, 100, 100, 3)
[[[1.         1.         1.        ]
  [0.         0.         0.01176471]
  [0.         0.         0.01176471]
  ...
  [0.         0.         0.01176471]
  [0.         0.         0.01176471]
  [0.         0.         0.01176471]]

 [[1.         1.         1.        ]
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]
  ...
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]]

 [[1.         1.         1.        ]
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]
  ...
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]]

 ...

 [[1.         1.         1.        ]
  [0.26666668 0.05882353 0.46274513]
  [0.         0.         0.01568628]
  ...
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]]

 [[1.         1.         1.        ]
  [0.22352943 0.05490196 0.4156863 ]
```

```
  [0.         0.         0.01176471]
  ...
  [0.         0.         0.01176471]
  [0.         0.         0.01176471]
  [0.         0.         0.01176471]]

 [[1.         1.         1.        ]
  [1.         1.         1.        ]
  [1.         1.         1.        ]
  ...
  [1.         1.         1.        ]
  [1.         1.         1.        ]
  [1.         1.         1.        ]]]
[0. 0. 0. 0. 0. 0. 0. 0. 1.]
```




```
(2, 100, 100, 3)
[[[1.         1.         1.        ]
  [0.         0.         0.01176471]
  [0.         0.         0.01176471]
  ...
  [0.         0.         0.01176471]
  [0.         0.         0.01176471]
  [0.         0.         0.01176471]]

 [[1.         1.         1.        ]
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]
  ...
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]]

 [[1.         1.         1.        ]
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]
  ...
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]
  [0.         0.         0.01568628]]

 ...

 [[1.         1.         1.        ]
  [0.39607847 0.10196079 0.5019608 ]
  [0.7490196  0.227451   0.4666667 ]
  ...
  [0.97647065 0.47450984 0.3647059 ]
  [0.79215693 0.24313727 0.44705886]
  [0.65882355 0.19607845 0.4901961 ]]

 [[1.         1.         1.        ]
  [0.31764707 0.07058824 0.46274513]
  [0.6039216  0.1764706  0.4666667 ]
  ...
  [0.86274517 0.30980393 0.36078432]
  [0.9058824  0.39607847 0.3372549 ]
  [0.8235295  0.27058825 0.38431376]]

 [[1.         1.         1.        ]
  [1.         1.         1.        ]
  [1.         1.         1.        ]
  ...
  [1.         1.         1.        ]
  [1.         1.         1.        ]
  [1.         1.         1.        ]]]
[0. 0. 0. 0. 0. 1. 0. 0. 0. 0.]
```



In [4]:

```
import numpy as np
import pylab as pl
from keras import backend as K
import matplotlib.pyplot as plt
from keras.utils import np_utils
from keras.models import Sequential
from keras.layers.convolutional import Conv2D, MaxPooling2D
from keras.layers.core import Dense,Dropout,Activation,Flatten
from tensorflow.keras import layers,activations
```

In [5]:

```
model = Sequential() #Sequential modeli oluşturduk.(Boş)

model.add(layers.Conv2D(filters = 4, activation = 'relu', kernel_size = (5,5), input_shape = (100,100,3))) #Oluşa
model.add(layers.MaxPooling2D(2,2)) #Üstteki işlemden sonra pooling yaptık.Filtreleri arttırarak bağlantıları güç

model.add(layers.Conv2D(filters = 8, activation = 'elu', kernel_size = (3,3)))
model.add(layers.MaxPooling2D(2,2))

model.add(layers.Conv2D(filters = 16, activation = 'elu', kernel_size = (2,2)))
model.add(layers.MaxPooling2D(2,2))

model.add(layers.Conv2D(filters = 32, activation = 'elu', kernel_size = (2,2)))

model.add(layers.Flatten()) #CNN in son aşaması olarak düzleştirme yaptık.

model.add(layers.Dense(50, activation="relu")) #Oluşan ağları dense yaptık ve son outputta 10 adet sınıf istediği
model.add(layers.Dense(100, activation="relu"))
model.add(layers.Dense(100, activation="relu"))
model.add(layers.Dense(50, activation="relu"))
model.add(layers.Dense(10, activation="softmax"))
```

In [6]:

```
model.summary() #Oluşan modelin özetini çıkardık.
```

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 96, 96, 4)         304

 max_pooling2d (MaxPooling2D  (None, 48, 48, 4)        0
 )

 conv2d_1 (Conv2D)           (None, 46, 46, 8)         296

 max_pooling2d_1 (MaxPooling  (None, 23, 23, 8)        0
 2D)

 conv2d_2 (Conv2D)           (None, 22, 22, 16)        528

 max_pooling2d_2 (MaxPooling  (None, 11, 11, 16)       0
 2D)

 conv2d_3 (Conv2D)           (None, 10, 10, 32)        2080

 flatten (Flatten)           (None, 3200)              0

 dense (Dense)               (None, 50)                160050

 dense_1 (Dense)             (None, 100)               5100

 dense_2 (Dense)             (None, 100)               10100
```

```
 dense_3 (Dense)             (None, 50)                5050

 dense_4 (Dense)             (None, 10)                510

=================================================================
Total params: 184,018
Trainable params: 184,018
Non-trainable params: 0
_________________________________________________________________
```

In [7]:
```
#Model Eğitimi
```

In [8]:
```
import tensorflow as tf
optimizer = tf.keras.optimizers.Adamax(learning_rate = 0.001) #Optimizer seçtik ve learning_rate değeri belirledi
loss = tf.keras.losses.CategoricalCrossentropy()
```

In [9]:
```
model.compile(optimizer = optimizer, loss = loss, metrics = [ "mse","accuracy"], run_eagerly=True) #compiler etti
```

In [10]:
```
result = model.fit(train_datagen, epochs= 5, verbose = 1, validation_data = test_datagen) #Modeli test ettik
```

```
Epoch 1/5
3931/3931 [==============================] - 142s 36ms/step - loss: 1.4909 - mse: 0.0658 - accuracy: 0.4633 - val
_loss: 1.2512 - val_mse: 0.0587 - val_accuracy: 0.5476
Epoch 2/5
3931/3931 [==============================] - 133s 34ms/step - loss: 0.9482 - mse: 0.0439 - accuracy: 0.6776 - val
_loss: 1.2788 - val_mse: 0.0596 - val_accuracy: 0.5522
Epoch 3/5
3931/3931 [==============================] - 134s 34ms/step - loss: 0.7010 - mse: 0.0325 - accuracy: 0.7701 - val
_loss: 1.1812 - val_mse: 0.0515 - val_accuracy: 0.6039
Epoch 4/5
3931/3931 [==============================] - 134s 34ms/step - loss: 0.5422 - mse: 0.0253 - accuracy: 0.8223 - val
_loss: 1.4011 - val_mse: 0.0568 - val_accuracy: 0.5913
Epoch 5/5
3931/3931 [==============================] - 134s 34ms/step - loss: 0.4293 - mse: 0.0204 - accuracy: 0.8601 - val
_loss: 1.4353 - val_mse: 0.0576 - val_accuracy: 0.5924
```

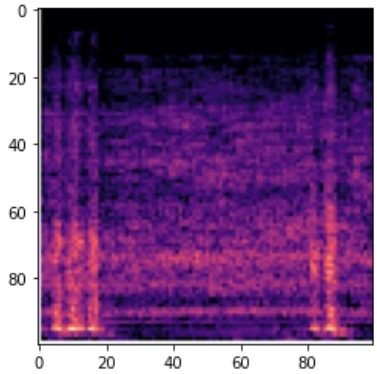
In [11]:
```
model.evaluate(test_datagen) # Loss mse ve accuracy değerlerini kısa bir testte gördük
```

```
436/436 [==============================] - 8s 19ms/step - loss: 1.4353 - mse: 0.0576 - accuracy: 0.5924
```

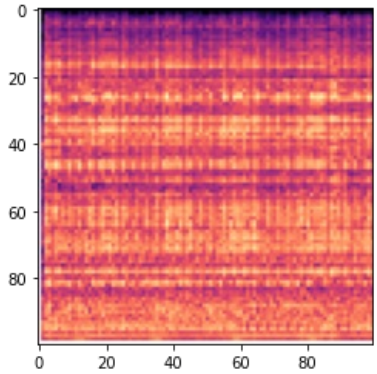
Out[11]:
```
[1.435295820236206, 0.0576358400285244, 0.5924224853515625]
```

In [12]:
```
print(test_datagen.class_indices) #Baştaki paththten 100 tane resim seçip sınıf değerlerine göre sonuç vermesini i
for i in range(100):
    img,label = test_datagen.next()
    a = model.predict(img)
    np.argmax(a[0])
    plt.imshow(img[0])
    if np.argmax(a[0]) == 0:
        print("air_conditioner")
    if np.argmax(a[0]) == 1:
        print("car_horn")
    if np.argmax(a[0]) == 2:
        print("children_playing")
    if np.argmax(a[0]) == 3:
        print("dog_bark")
    if np.argmax(a[0]) == 4:
        print('drilling')
    if np.argmax(a[0]) == 5:
        print("engine_idling")
    if np.argmax(a[0]) == 6:
        print("gun_shot")
    if np.argmax(a[0]) == 7:
        print("jackhammer")
    if np.argmax(a[0]) == 8:
        print("siren")
    if np.argmax(a[0]) == 9:
        print("street_music")
    plt.show()
```

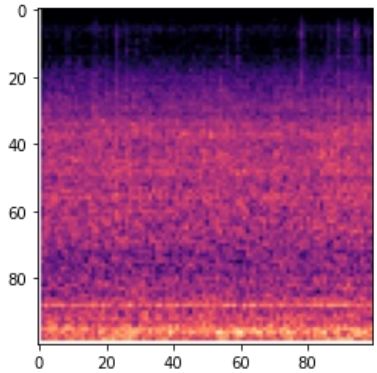
```
{'0': 0, '1': 1, '2': 2, '3': 3, '4': 4, '5': 5, '6': 6, '7': 7, '8': 8, '9': 9}
1/1 [==============================] - 0s 48ms/step
drilling
```



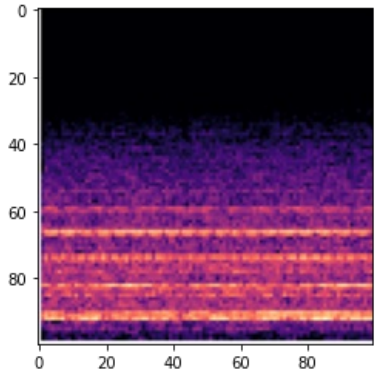

```
1/1 [==============================] - 0s 31ms/step
jackhammer
```



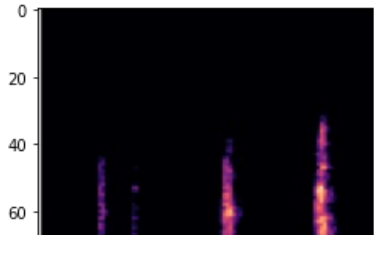

```
1/1 [==============================] - 0s 31ms/step
engine_idling
```

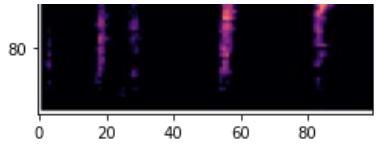



```
1/1 [==============================] - 0s 31ms/step
car_horn
```

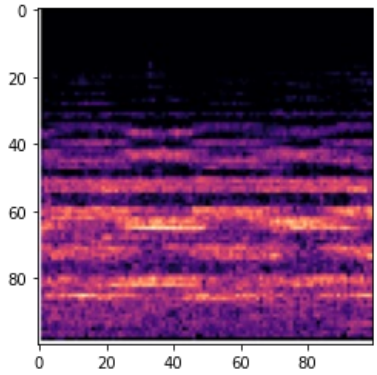



```
1/1 [==============================] - 0s 29ms/step
dog_bark
```

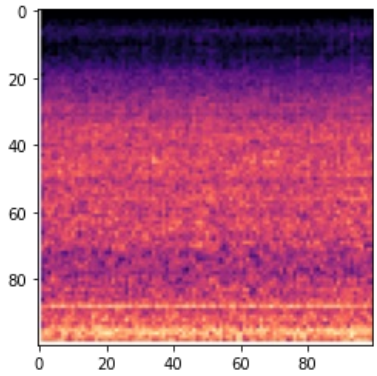




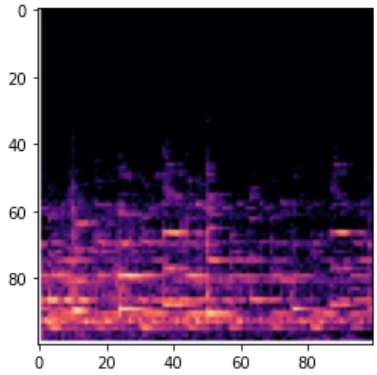

```
1/1 [==============================] - 0s 29ms/step
street_music
```



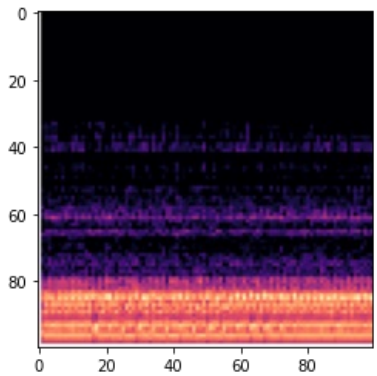

```
1/1 [==============================] - 0s 31ms/step
engine_idling
```



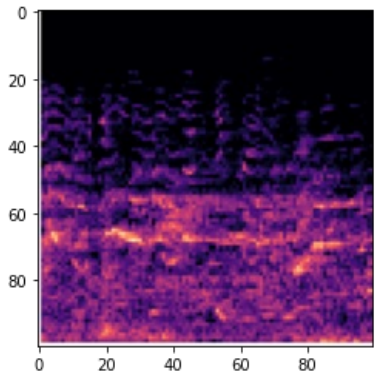

```
1/1 [==============================] - 0s 30ms/step
street_music
```



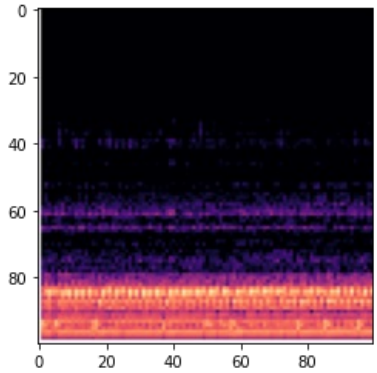

```
1/1 [==============================] - 0s 30ms/step
street_music
```



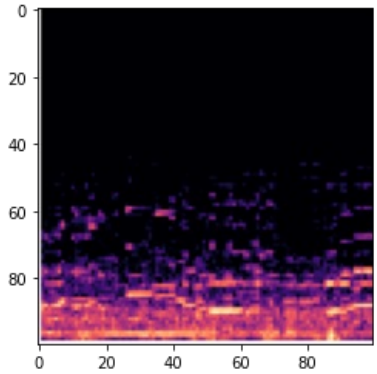

```
1/1 [==============================] - 0s 30ms/step
siren
```



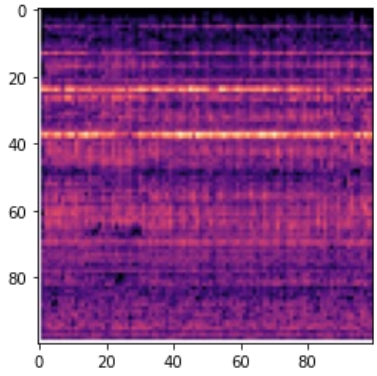

```
1/1 [==============================] - 0s 30ms/step
street_music
```



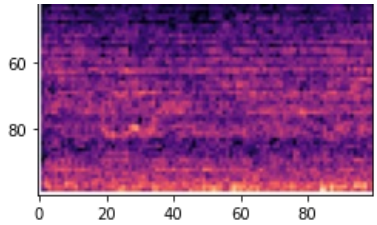

```
1/1 [==============================] - 0s 33ms/step
street_music
```



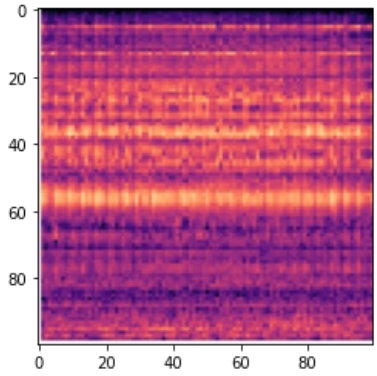

```
1/1 [==============================] - 0s 29ms/step
jackhammer
```



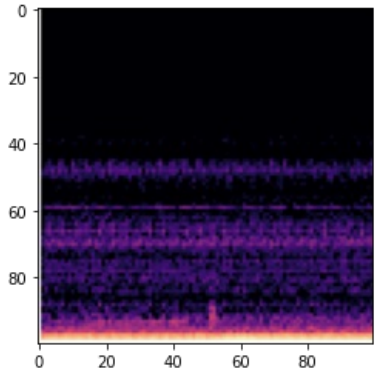

```
1/1 [==============================] - 0s 29ms/step
siren
```



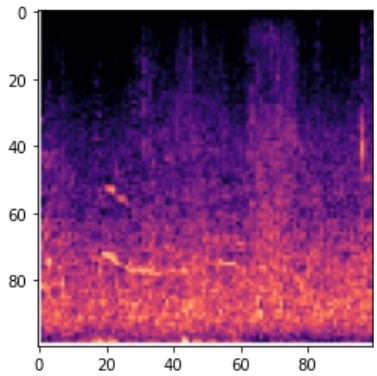

```
1/1 [==============================] - 0s 31ms/step
jackhammer
```



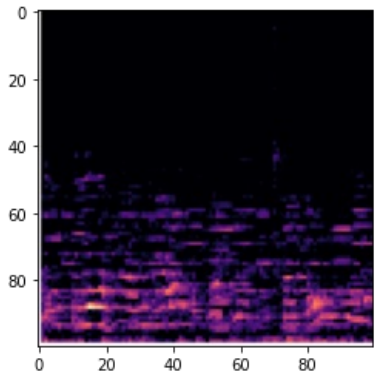

```
1/1 [==============================] - 0s 30ms/step
engine_idling
```



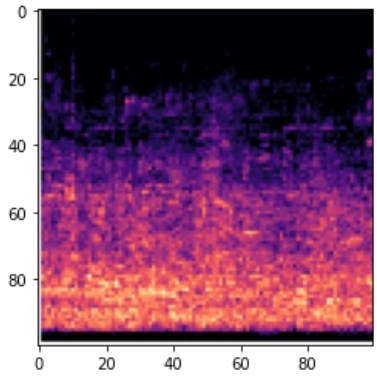

```
1/1 [==============================] - 0s 29ms/step
drilling
```



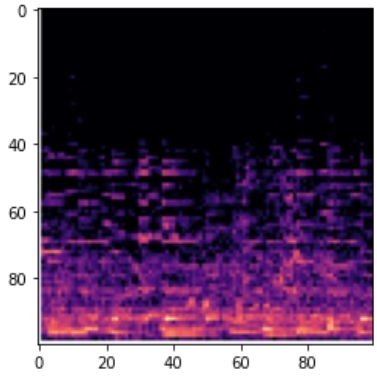

```
1/1 [==============================] - 0s 29ms/step
street_music
```



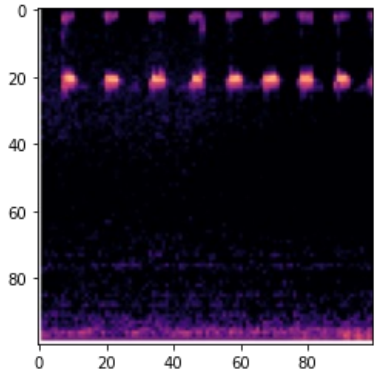
```
1/1 [==============================] - 0s 32ms/step
street_music
```



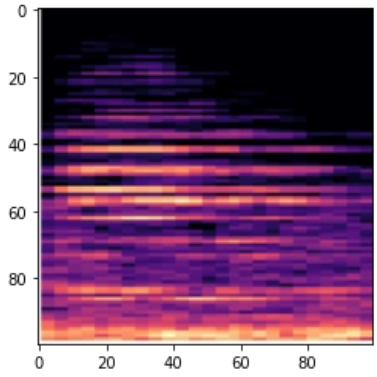

```
1/1 [==============================] - 0s 29ms/step
street_music
```



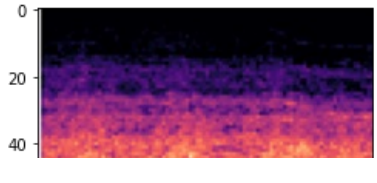

```
1/1 [==============================] - 0s 32ms/step
siren
```

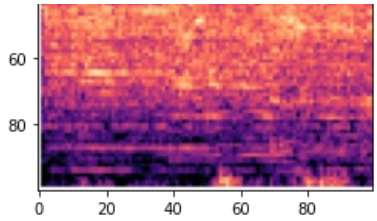



```
1/1 [==============================] - 0s 30ms/step
car_horn
```

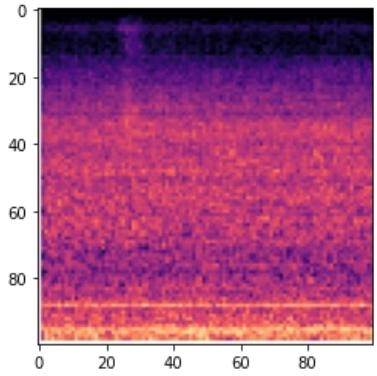



```
1/1 [==============================] - 0s 30ms/step
drilling
```

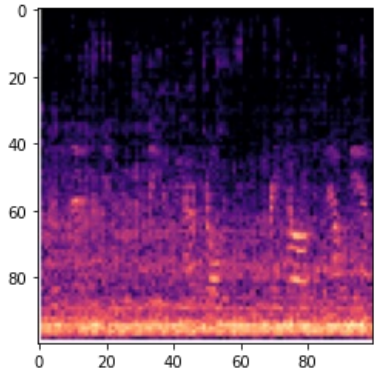




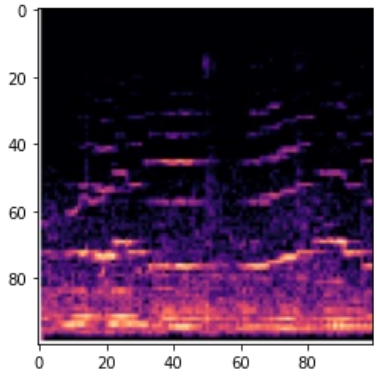

```
1/1 [==============================] - 0s 31ms/step
engine_idling
```



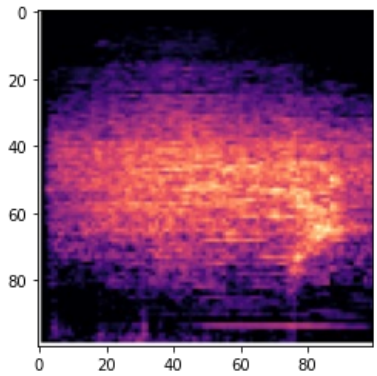

```
1/1 [==============================] - 0s 33ms/step
siren
```



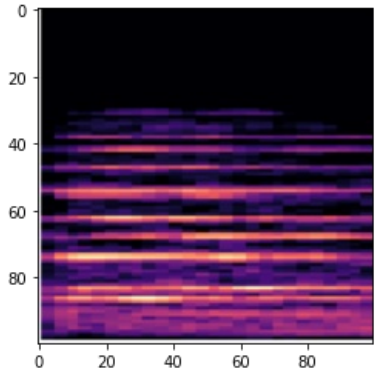

```
1/1 [==============================] - 0s 34ms/step
street_music
```



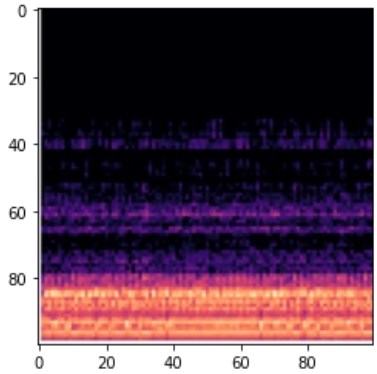

```
1/1 [==============================] - 0s 32ms/step
drilling
```



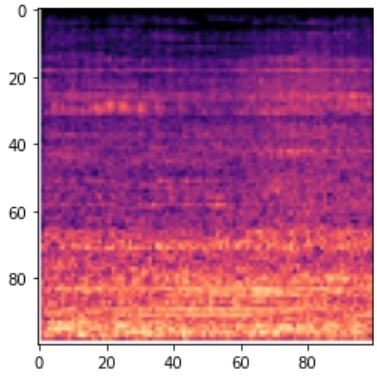
```
1/1 [==============================] - 0s 29ms/step
car_horn
```



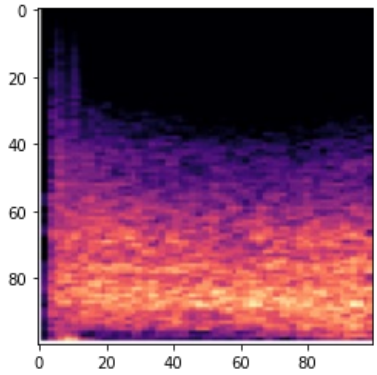

```
1/1 [==============================] - 0s 31ms/step
engine_idling
```



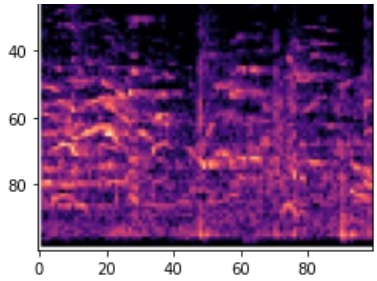

```
1/1 [==============================] - 0s 30ms/step
jackhammer
```



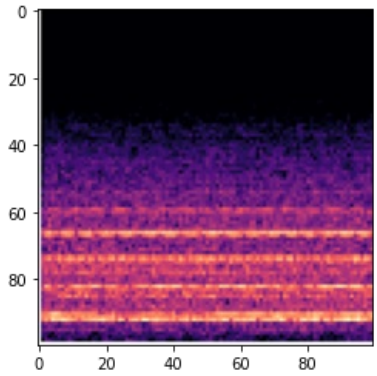

```
1/1 [==============================] - 0s 33ms/step
gun_shot
```



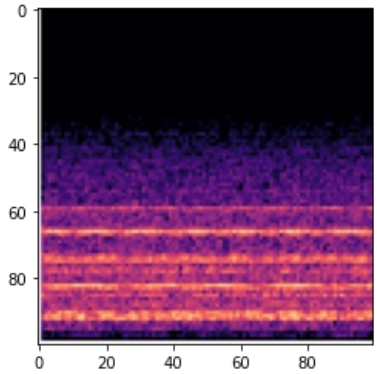

```
1/1 [==============================] - 0s 32ms/step
children_playing
```



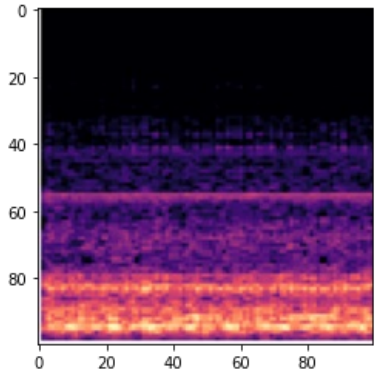

```
1/1 [==============================] - 0s 31ms/step
car_horn
```



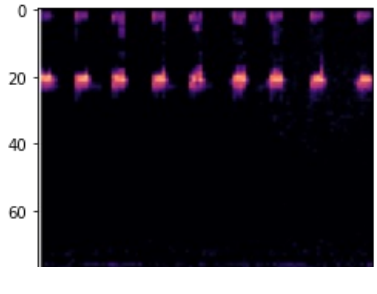

```
1/1 [==============================] - 0s 30ms/step
car_horn
```

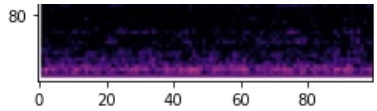



```
1/1 [==============================] - 0s 32ms/step
street_music
```

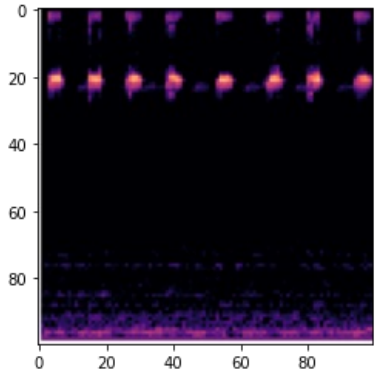



```
1/1 [==============================] - 0s 29ms/step
siren
```

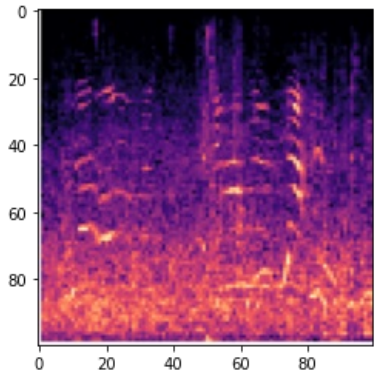




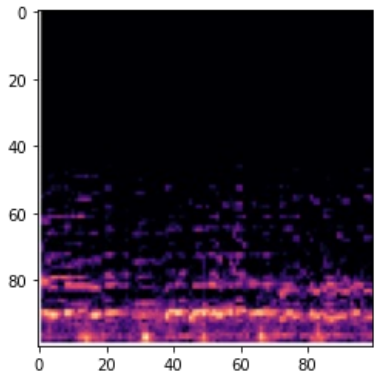

```
1/1 [==============================] - 0s 31ms/step
siren
```



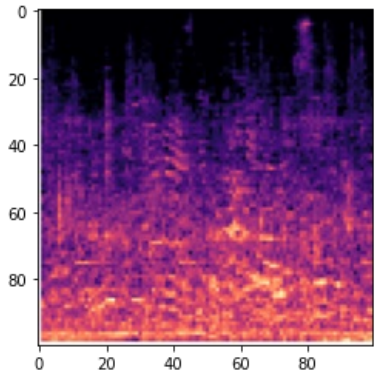

```
1/1 [==============================] - 0s 30ms/step
children_playing
```




```
1/1 [==============================] - 0s 32ms/step
street_music
```

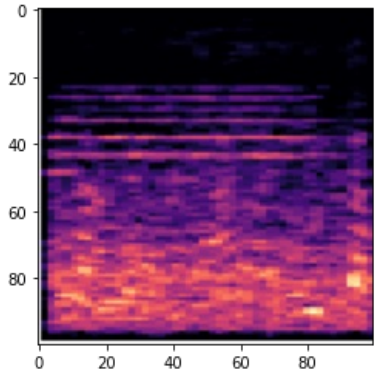



```
1/1 [==============================] - 0s 29ms/step
street_music
```

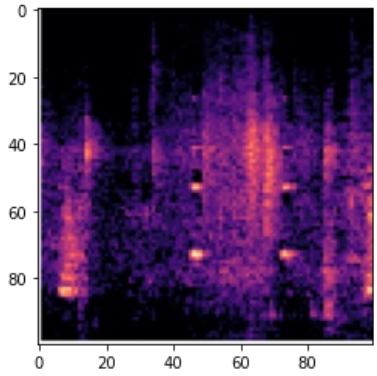



```
1/1 [==============================] - 0s 26ms/step
```

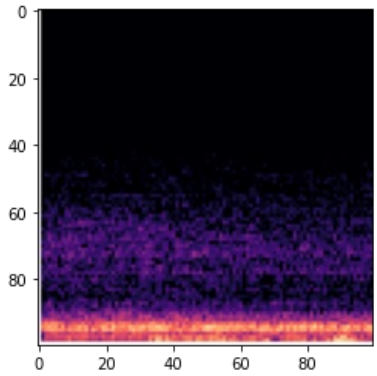
```
street_music
```



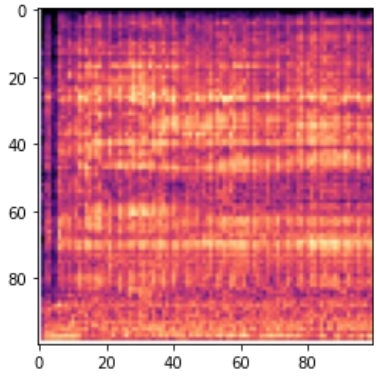

```
1/1 [==============================] - 0s 25ms/step
dog_bark
```



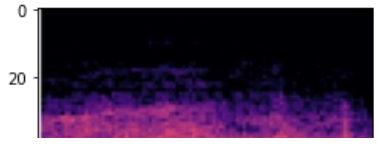

```
1/1 [==============================] - 0s 22ms/step
siren
```

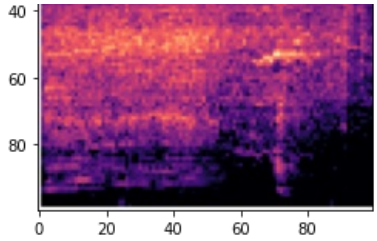



```
1/1 [==============================] - 0s 23ms/step
drilling
```

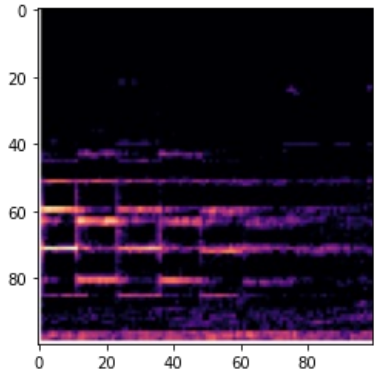



```
1/1 [==============================] - 0s 24ms/step
drilling
```

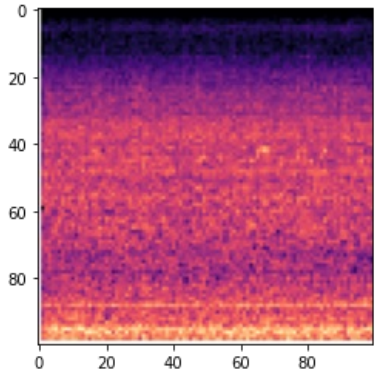




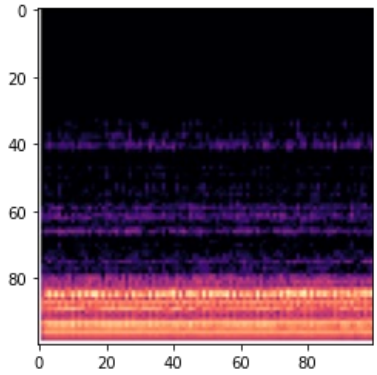

```
1/1 [==============================] - 0s 22ms/step
siren
```



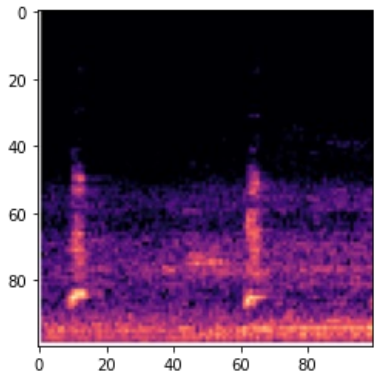

```
1/1 [==============================] - 0s 22ms/step
engine_idling
```



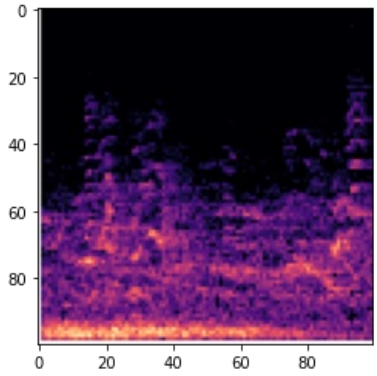

```
1/1 [==============================] - 0s 25ms/step
car_horn
```



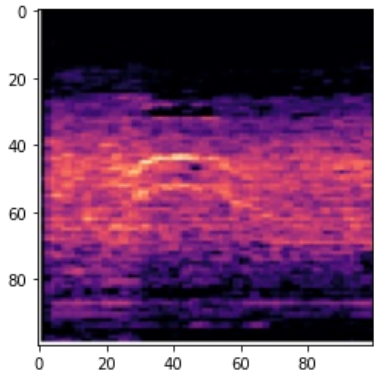

```
1/1 [==============================] - 0s 23ms/step
dog_bark
```



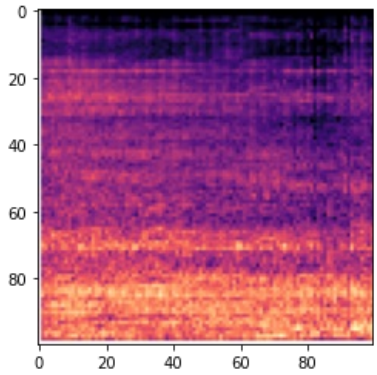
```
1/1 [==============================] - 0s 25ms/step
children_playing
```



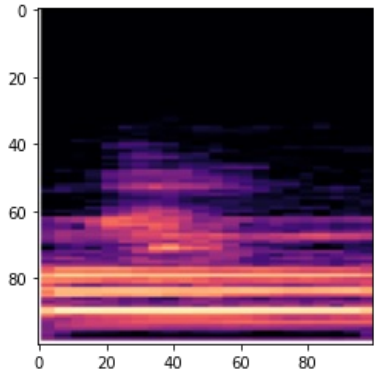

```
1/1 [==============================] - 0s 24ms/step
children_playing
```



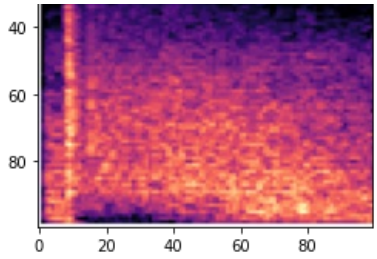

```
1/1 [==============================] - 0s 24ms/step
air_conditioner
```



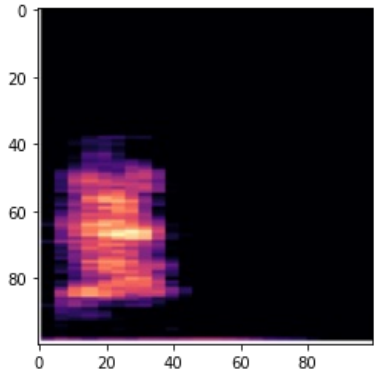

```
1/1 [==============================] - 0s 26ms/step
jackhammer
```



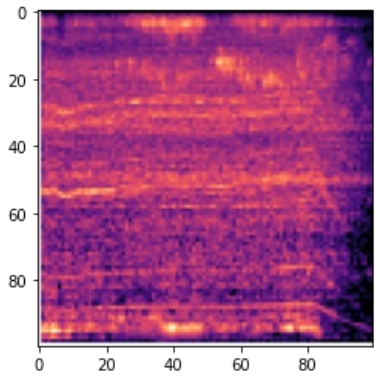

```
1/1 [==============================] - 0s 25ms/step
gun_shot
```



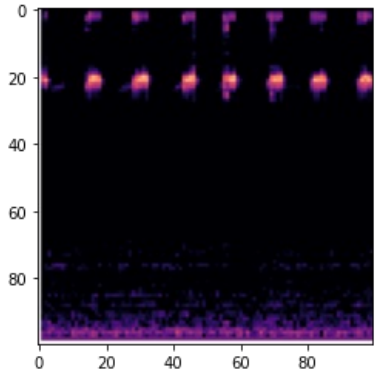

```
1/1 [==============================] - 0s 22ms/step
dog_bark
```



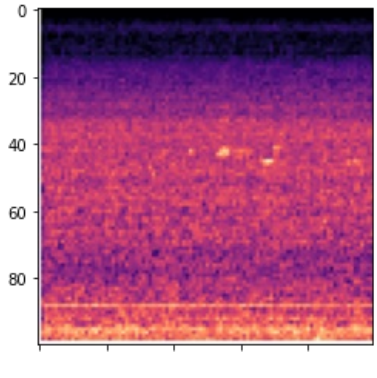

```
1/1 [==============================] - 0s 24ms/step
drilling
```



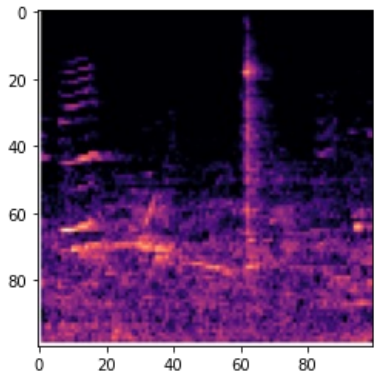

```
1/1 [==============================] - 0s 23ms/step
dog_bark
```



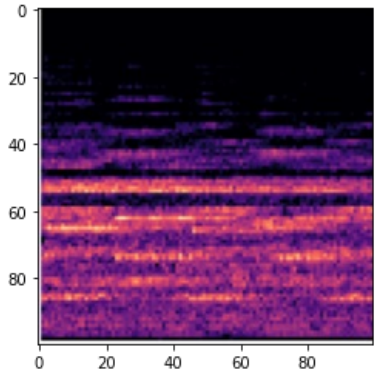

```
1/1 [==============================] - 0s 23ms/step
engine_idling
```



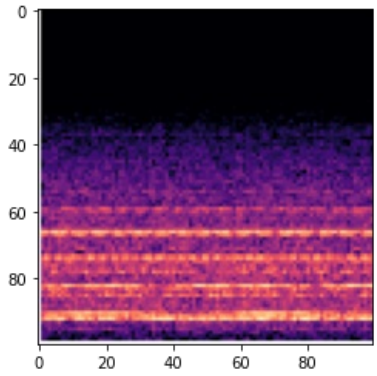
```
1/1 [==============================] - 0s 24ms/step
dog_bark
```



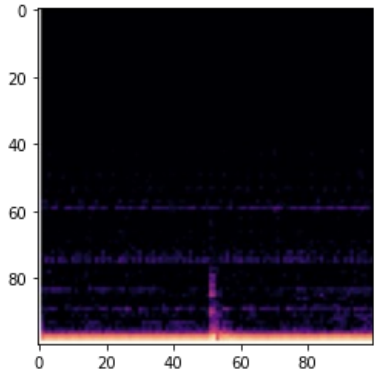

```
1/1 [==============================] - 0s 22ms/step
siren
```



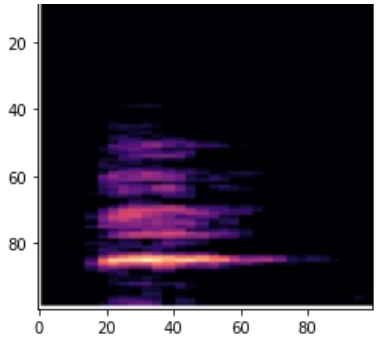

```
1/1 [==============================] - 0s 24ms/step
car_horn
```



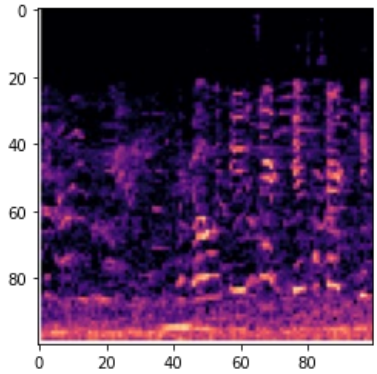

```
1/1 [==============================] - 0s 26ms/step
children_playing
```



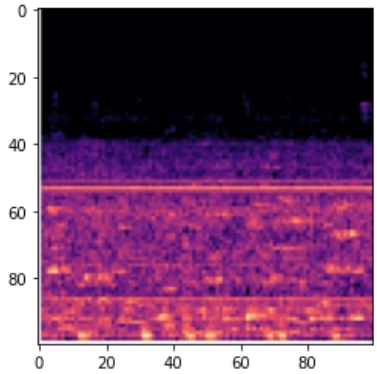

```
1/1 [==============================] - 0s 22ms/step
dog_bark
```



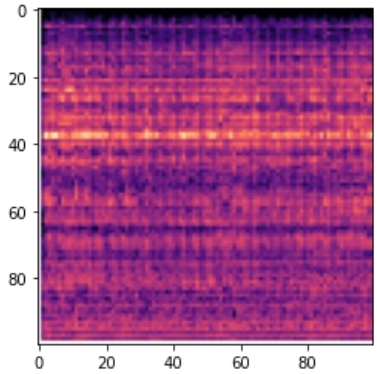

```
1/1 [==============================] - 0s 22ms/step
children_playing
```



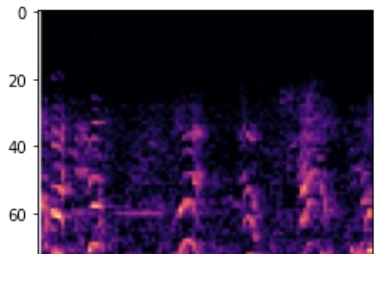

```
1/1 [==============================] - 0s 23ms/step
air_conditioner
```

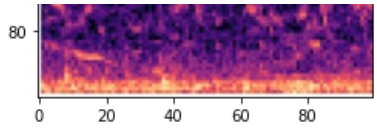



```
1/1 [==============================] - 0s 22ms/step
drilling
```

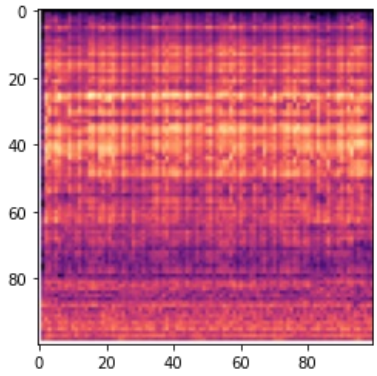



```
1/1 [==============================] - 0s 24ms/step
children_playing
```

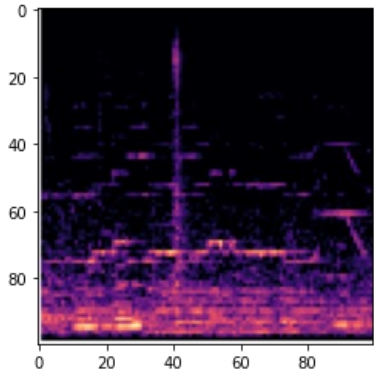




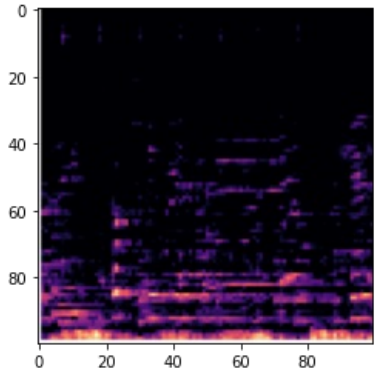

```
1/1 [==============================] - 0s 23ms/step
jackhammer
```



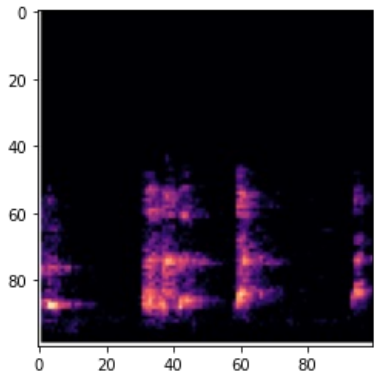

```
1/1 [==============================] - 0s 22ms/step
children_playing
```




```
1/1 [==============================] - 0s 25ms/step
street_music
```

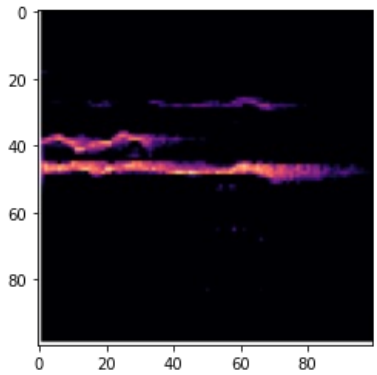



```
1/1 [==============================] - 0s 23ms/step
dog_bark
```

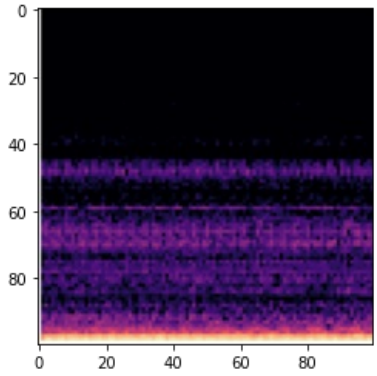



```
1/1 [==============================] - 0s 24ms/step
```

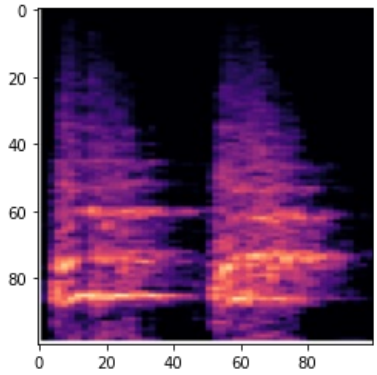
```
children_playing
```



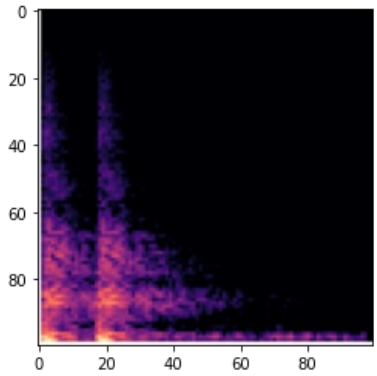

```
1/1 [==============================] - 0s 25ms/step
children_playing
```



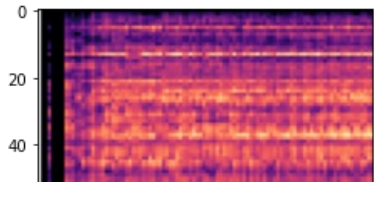

```
1/1 [==============================] - 0s 22ms/step
dog_bark
```

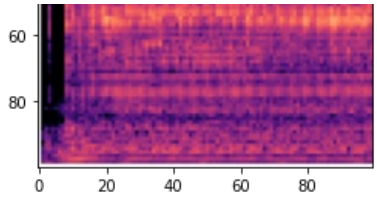



```
1/1 [==============================] - 0s 22ms/step
gun_shot
```

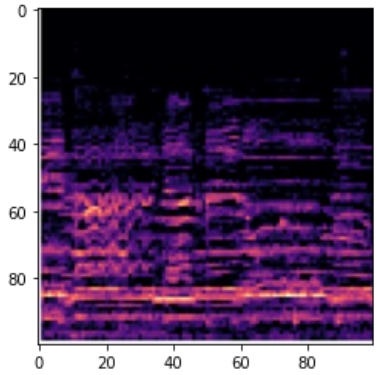



```
1/1 [==============================] - 0s 24ms/step
jackhammer
```

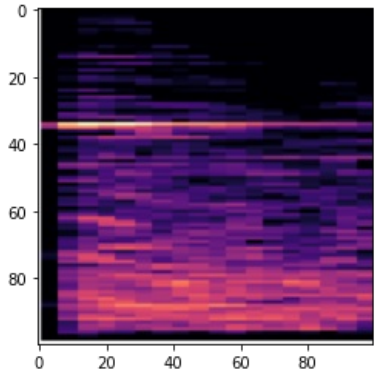




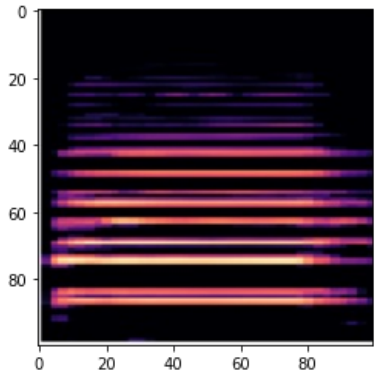

```
1/1 [==============================] - 0s 23ms/step
street_music
```



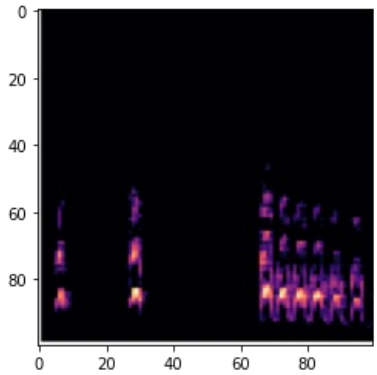

```
1/1 [==============================] - 0s 24ms/step
car_horn
```



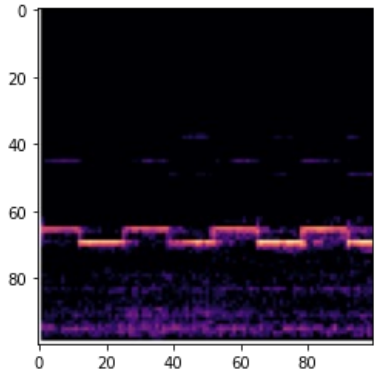

```
1/1 [==============================] - 0s 22ms/step
car_horn
```



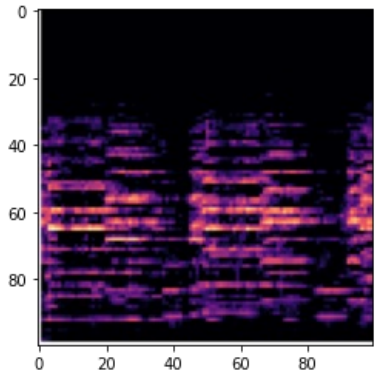

```
1/1 [==============================] - 0s 23ms/step
dog_bark
```



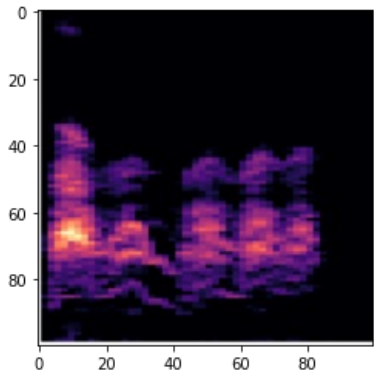
```
1/1 [==============================] - 0s 23ms/step
siren
```



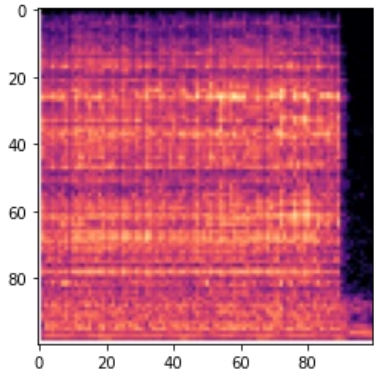

```
1/1 [==============================] - 0s 22ms/step
siren
```



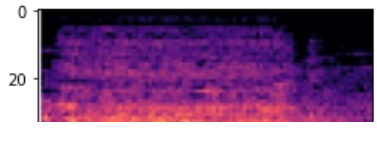

```
1/1 [==============================] - 0s 23ms/step
dog_bark
```

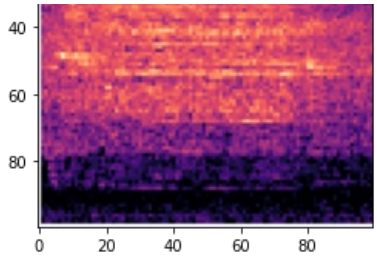



```
1/1 [==============================] - 0s 22ms/step
jackhammer
```

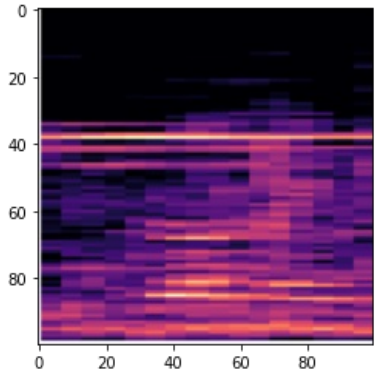



```
1/1 [==============================] - 0s 23ms/step
drilling
```

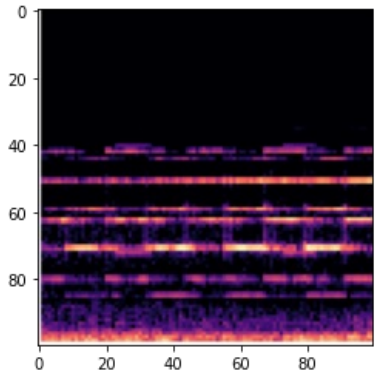




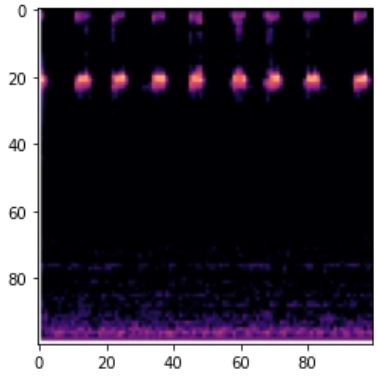

```
1/1 [==============================] - 0s 23ms/step
car_horn
```



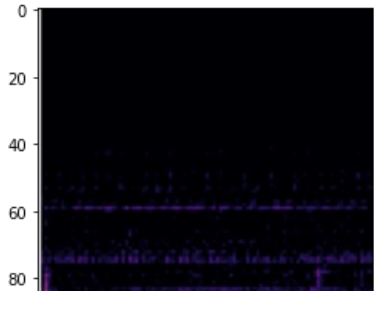

```
1/1 [==============================] - 0s 22ms/step
siren
```

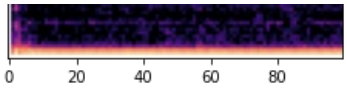



```
1/1 [==============================] - 0s 29ms/step
siren
```

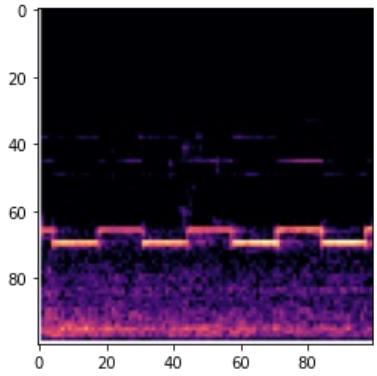



```
1/1 [==============================] - 0s 28ms/step
air_conditioner
```

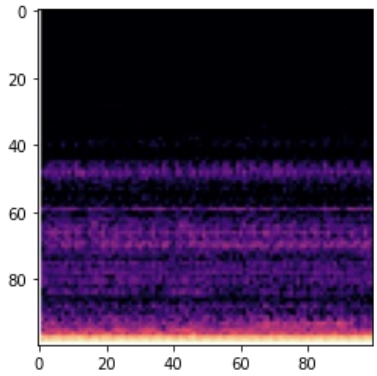




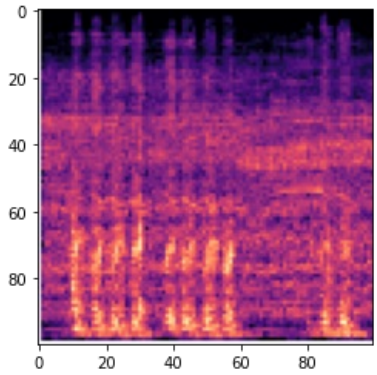

```
1/1 [==============================] - 0s 26ms/step
siren
```



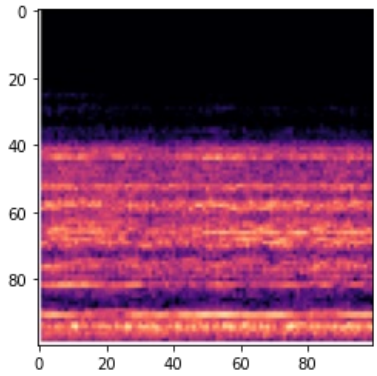

```
1/1 [==============================] - 0s 28ms/step
children_playing
```



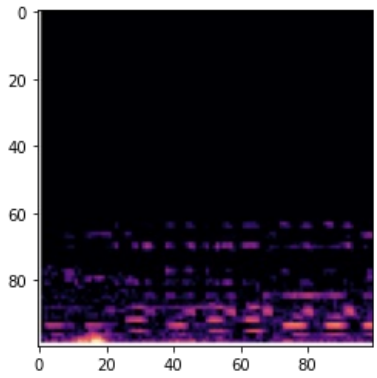

```
1/1 [==============================] - 0s 29ms/step
drilling
```



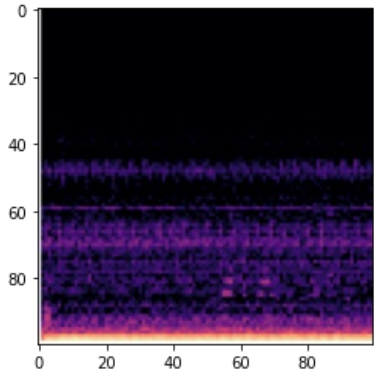

```
1/1 [==============================] - 0s 22ms/step
siren
```



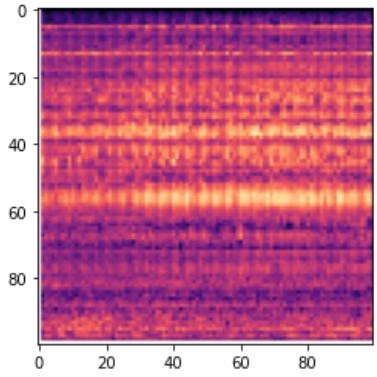

```
1/1 [==============================] - 0s 23ms/step
street_music
```



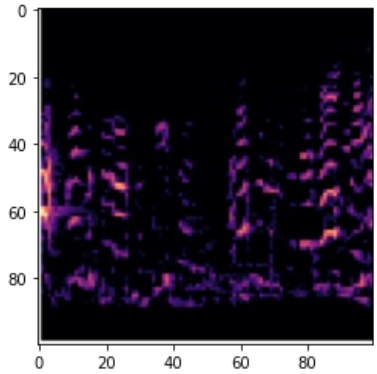

```
1/1 [==============================] - 0s 25ms/step
engine_idling
```



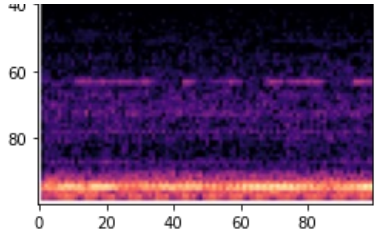

```
1/1 [==============================] - 0s 21ms/step
jackhammer
```



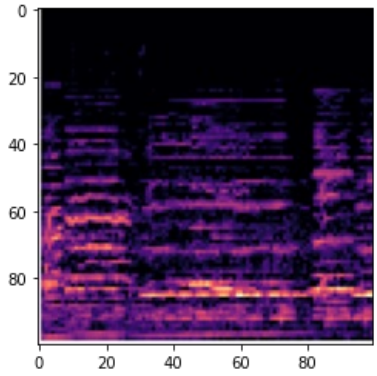

```
1/1 [==============================] - 0s 24ms/step
children_playing
```




```
1/1 [==============================] - 0s 24ms/step
siren
```




```
1/1 [==============================] - 0s 22ms/step
street_music
```




```
1/1 [==============================] - 0s 26ms/step
gun_shot
```