In [ ]:

```
pip install scikit-learn
```

In [82]:

```
import numpy as np
import matplotlib.pyplot as plt
from skimage.feature import hog
from skimage import exposure
from sklearn.datasets import load_digits
from sklearn.model_selection import train_test_split
from sklearn.svm import SVC
from sklearn.metrics import accuracy_score
from sklearn.metrics import precision_score
```

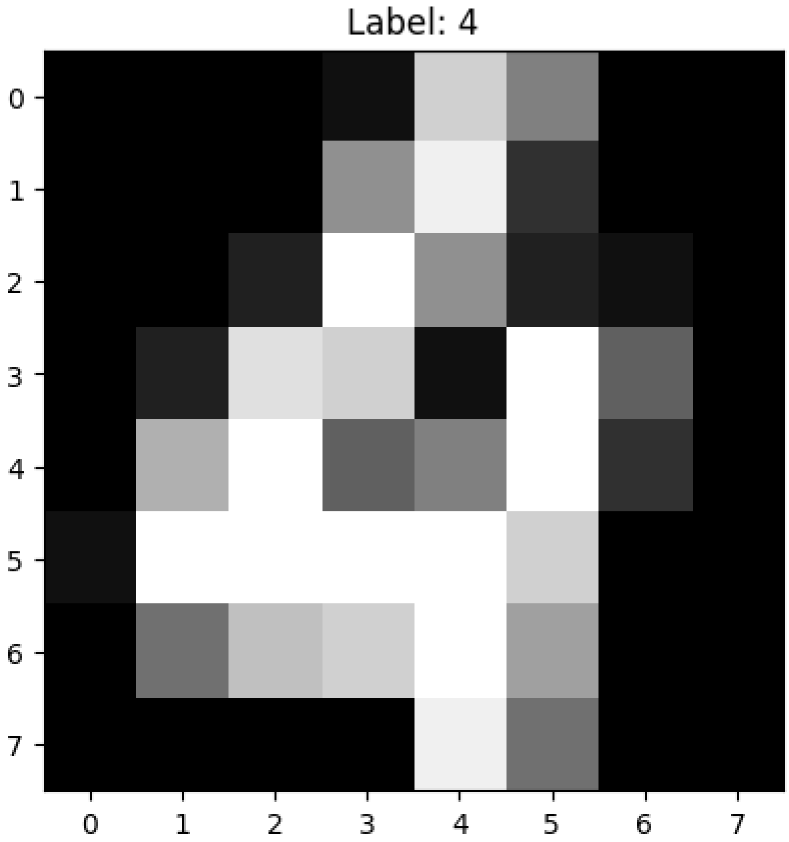
In [83]:

```
# Load the digits dataset
digits = load_digits()

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(digits.data, digits.target,
```

In [84]:

```
# Visualize the first digit in the dataset
plt.imshow(X_train[12].reshape(8, 8), cmap=plt.cm.gray)
plt.title(f"Label: {y_train[12]}")
plt.show()
```



In [85]:

```
# Define HOG parameters
orientations = 8
pixels_per_cell = (4, 4)
cells_per_block = (2, 2)

# Extract HOG features for training and testing data
X_train_hog = []
for image in X_train:
    fd, hog_image = hog(image.reshape((8, 8)), orientations=orientations,
                        pixels_per_cell=pixels_per_cell, cells_per_block=cells_p
                        block_norm='L2-Hys', visualize=True)
    X_train_hog.append(fd)
X_train_hog = np.array(X_train_hog)
```

In [86]:

```
# Train a Support Vector Machine classifier
svm_classifier = SVC(kernel='linear')
svm_classifier.fit(X_train_hog, y_train)
```

Out[86]:

```
▾ SVC

SVC(kernel='linear')
```

In [87]:

```
# Make predictions on the test data
y_pred = svm_classifier.predict(X_test_hog)
```

In [88]:

```
# Calculate and print accuracy
accuracy = accuracy_score(y_test, y_pred)
print(f"Accuracy: {accuracy*100:.3f}")
```

```
Accuracy: 84.340
```

In [92]:

```
Precision = precision_score(y_test, y_pred,average='micro')
print(f"Precision: {Precision*100:.3f}")
```

```
Precision: 84.340
```