صورة تحتوي على طائر, زهرة

تم إنشاء الوصف تلقائياً

المرشد السياحي الذكي   
Smart Tourist Guide

**فايز العتيق**

**نواف الجبر**

**تركي العتيبي**

Training the model:

model = Sequential()

model.add(Conv2D(32, (5, 5), activation='relu', input\_shape=(32,32,3)))

model.add(MaxPooling2D(pool\_size=(2, 2)))

model.add(Conv2D(64, (5, 5), activation='relu'))

model.add(MaxPooling2D(pool\_size=(2, 2)))

model.add(Flatten())

model.add(Dense(1000, activation='relu'))

model.add(Dropout(0.5))

model.add(Dense(500, activation='relu'))

model.add(Dropout(0.5))

model.add(Dense(250, activation='relu'))

model.add(Dense(10, activation='softmax'))

model.compile(loss='categorical\_crossentropy',

optimizer='adam',

metrics=['accuracy'])

hist = model.fit(x\_train, y\_train\_one\_hot,

batch\_size=256, epochs=10, validation\_split=0.2 )

model.evaluate(x\_test, y\_test\_one\_hot)[1]

\_\_ saving the model.

model.save('path/to/location')

we saved the model, loaded in the app.

------ Testing new Image

from app upload button, user can load image:

uploaded = files.upload() # Use to load feature in the apps

new\_image = plt.imread("neom4015.jpg") #Read in the image (3, 14, 20)

we resize the image to the optimal size. all images have the same size.

resized\_image = resize(new\_image, (32,32,3))

predictions = model.predict(np.array( [resized\_image] ))

--- retrive info about the place from the database

Establish connection with server .

$connection = mysql\_connect("localhost", "root", "");

Selects database.

$db = mysql\_select\_db("landmarks", $connection);

Executes MySQL select query.

$query = mysql\_query("select \* from landmarks", $connection);

Display fetched data

<span>Name:</span> <?php echo $row1['landmarks\_name']; ?>

<span>Info:</span> <?php echo $row1['landmarks\_info']; ?>

Copy

Closing connection with server.

mysql\_close($connection);