LAB No. 9

Convolution Neural Network

Open Ended LAB

Build a Convolutional Neural Network (CNN) using Python and TensorFlow/Keras to classify images of Cats and Dogs.

Instructions:

1. Collect or download a dataset containing:
    - **500 images of Cats**
    - **500 images of Dogs**
2. Organize the dataset as:

    dataset/

        cats/

        dogs/
3. Write Python code to:
    - Load and preprocess images (resize to 150×150)
    - Split into training (80%) and testing (20%)
    - Build a CNN model
    - Train the model for 10–20 epochs
    - Plot training & validation accuracy and loss
    - Evaluate model performance using a confusion matrix
    - Predict whether a new input image is Cat or Dog
4. At the end of the program, show the prediction result for a test image:
    - "This image is a CAT"
    - or
      "This image is a DOG"
5. Submit your Python code, dataset, graphs, and output screenshots.

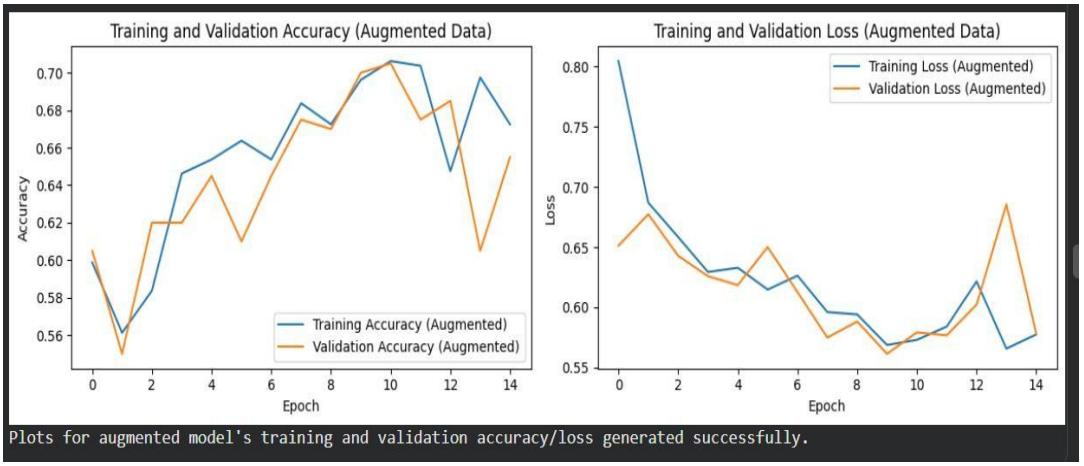
Code:

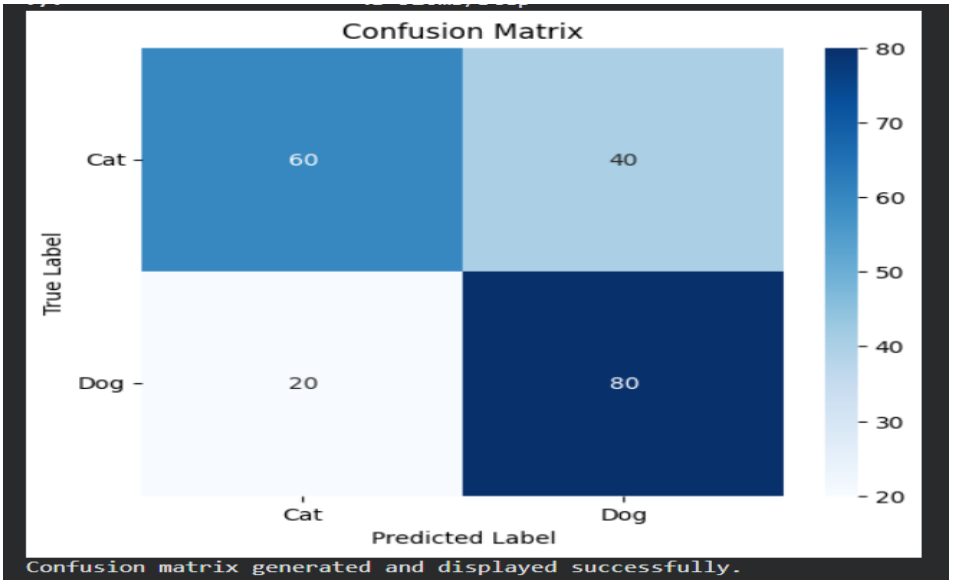
```
import numpy as np
import matplotlib.pyplot as plt
import kagglehub
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix, ConfusionMatrixDisplay
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Flatten, Dense, Dropout
from tensorflow.keras.utils import to_categorical
path = kagglehub.dataset_download("karakaggle/kaggle-cat-vs-dog-dataset")
print("Path to dataset files:", path)
DATASET_PATH = None
for root, dirs, files in
  os.walk(path): if "PetImages" in
  dirs:
    DATASET_PATH = os.path.join(root, "PetImages") break
if DATASET_PATH is None:
  raise Exception("PetImages folder not found!")
print("Final Dataset Path:", DATASET_PATH)
print("Classes:", os.listdir(DATASET_PATH))
IMG_SIZE = 150
EPOCHS = 15
data = []
labels = []
categories = ["Cat", "Dog"]
for label, category in enumerate(categories):
  folder_path = os.path.join(DATASET_PATH, category)
  for img_name in os.listdir(folder_path):
img_path = os.path.join(folder_path, img_name
```

```
Img=cv2.imread(img_pat)
if img is None:   continue
    img = cv2.resize(img, (IMG_SIZE,
    IMG_SIZE)) img = img / 255.0
    data.append(img)
    labels.append(label)
data = np.array(data)
labels = to_categorical(labels, num_classes=2)
X_train, X_test, y_train, y_test = train_test_split(
data, labels, test_size=0.2, random_state=42)
model=Sequential([Conv2D(32, (3,3), activation='relu',input_shape=(IMG_SIZE,
IMG_SIZE, 3)),
  MaxPooling2D(2,2),
  Conv2D(64, (3,3), activation='relu'),
  MaxPooling2D(2,2),
  Conv2D(128, (3,3), activation='relu'),
  MaxPooling2D(2,2),
  Flatten(),
  Dense(128,
  activation='relu'),
  Dropout(0.5),
  Dense(2,
activation='softmax')])
model.compile(
  optimizer='adam',
  loss='categorical_crossentrop
  y', metrics=['accuracy'])
model.summary()
     history =
    model.fit(
X_train, y_train,
 validation_data=(X_test, y_test),
```

```
plt.figure(figsize=(12,5))
plt.subplot(1,2,1)
plt.plot(history.history['accuracy'], label='Train Accuracy')
plt.plot(history.history['val_accuracy'], label='Validation Accuracy')
plt.legend()
plt.title("Accuracy")
plt.subplot(1,2,2)
plt.plot(history.history['loss'], label='Train Loss')
plt.plot(history.history['val_loss'], label='Validation Loss') plt.legend()
plt.title("Loss")   plt.show()
y_pred =  model.predict(X_test)
y_pred_classes = np.argmax(y_pred, axis=1) y_true
= np.argmax(y_test, axis=1)
cm  =  confusion_matrix(y_true,  y_pred_classes)
disp = ConfusionMatrixDisplay(cm, display_labels=["Cat", "Dog"])
disp.plot(cmap=plt.cm.Blues) plt.show()
def predict_image(img_path): img =
   cv2.imread(img_path)
   img = cv2.resize(img, (IMG_SIZE, IMG_SIZE))
   img = img / 255.0
   img = img.reshape(1, IMG_SIZE, IMG_SIZE, 3)
   prediction = model.predict(img)
   if np.argmax(prediction) == 0: print("This
     image is a CAT else:
predict_image(os.path.join(DATASET_PATH, "Cat",
os.listdir(os.path.join(DATASET_PATH,   "Cat"))[0]))
```

Output:






```
Prediction probability: 0.0970
True label for this image: Dog
This image is a Cat
```

LAB Assessment

Student Name		**LAB Rubrics**	CLO3 , P5, PLO5
		Total Marks	10
Registration No		**Obtained Marks**	
		Teacher Name	Dr. Syed M Hamedoon
Date		**Signature**	

Laboratory Work Assessment Rubrics

Sr. No.	Performance Indicator	Excellent (5)	Good (4)	Average (3)	Fair (2)	Poor (1)
1	**Theoretical knowledge 10%**	Student knows all the related concepts about the theoretical background of the experiment and rephrase those concepts in written and oral assessments	Student knows most of the related concepts about the theoretical background of the experiment and partially rephrase those concepts in written and oral assessments	Student knows few of the related concepts about the theoretical background of the experiment and partially rephrase those concepts in written and oral assessments	Student knows very little about the related concepts about the theoretical background of the experiment and poorly rephrase those concepts in written and oral assessments	Student has poor understanding of the related concepts about the theoretical background of the experiment and unable to rephrase those concepts in written and oral assessments
2	**Application Functionality 10%**	Application runs smoothly and operation of the application runs efficiently	Application compiles with no warnings. Robust operation of the application, with good recovery.	Application compiles with few or no warnings. Consideration given to unusual conditions with reasonable	Application compiles and runs without crashing. Some attempt at detecting and correcting errors.	Application does not compile or compiles but crashes. Confusing. Little or no error detection or correction.
3	**Specifications 10%**	The program works very efficiently and meets all of the required specifications.	The program works and meets some of the specifications.	The program works and produces the correct results and displays them correctly. It also meets most of the other specifications.	The program produces correct results but does not display them correctly.	The program is producing incorrect results.
4	**Level of understanding of the learned skill 10%**	Provide complete and logical answers based upon accurate technical content to the questions asked by examiner	Provide complete and logical answers based upon accurate technical content to the questions asked by examiner with few errors	Provide partially correct and logical answers based upon minimum technical content to the questions asked by examiner	Provide very few and illogical answers to the questions asked by examiner.	Provide no answer to the questions asked by examiner.
5	**Readability and Reusability 10%**	The code is exceptionally well organized and very easy to follow and reused	The code is fairly easy to read. The code could be reused as a whole or each class could be reused.	Most of the code could be reused in other programs.	Some parts of the code require change before they could be reused in other programs.	The code is poorly organized and very difficult to read and not organized for reusability.

6	**AI System Design 10%**	Well-designed AI models. Code is highly maintainable	Good designed AI models and Little code duplications	Some attempt to make AI models. Code can be maintained with significant effort	Little attempt to design AI models and less understanding of code	Very poor attempt to design AI models and its code
7	**Responsiveness to Questions/ Accuracy 10%**	1. Responds well, quick and very accurate all the time. 2. Effectively uses eye contact, speaks clearly, effectively and confidently using suitable volume	1. Generally Responsive and accurate most of the times. 2. Maintains eye contact, speaks clearly with suitable volume and pace.	1. Generally Responsive and accurate few times. 2. Some eye contact, speaks clearly and unclearly in different portions.	1. Not much Responsive and accurate most of the times. 2. Uses eye contact ineffectively and fails to speak clearly and audibly	. 1. Non Responsive and inaccurate all the times. 2. No eye contact and unable to speak 3. Dresses inappropriatel y
8	**Efficiency 10%**	The code is extremely efficient without sacrificing readability and understanding	The code is fairly efficient without sacrificing readability and understanding	Some part of the code is efficient and other part of the code is not understandable and work properly	The code is brute force and unnecessarily long	The code is huge and appears to be patched together
9	**Delivery** **10%**	The program was delivered in time during lab.	The program was delivered in Lab before the end time.	The program was delivered within the due date.	The code was delivered within a day after the due date.	The code was delivered more than 2 days overdue.
10	**Awareness of Safety Guidelines 10%**	Student has sufficient knowledge of the laboratory safety SOPs and protocol and is fully compliant to the guidelines	Student has sufficient knowledge of the laboratory safety SOPs and protocol and is Partially compliant to the guidelines	Student has little knowledge of the laboratory safety SOPs and protocol and is Partially compliant to the guidelines	Student has little knowledge of the laboratory safety SOPs and protocol and is non-compliant to the guidelines	Student has no knowledge of the laboratory safety SOPs and protocol and is non-compliant to the guidelines