LABORATORY REPORT

Application Development Lab
(CS33002)

B.Tech Program in CSE

Submitted By

Name:-Madhurima Ghosh

Roll No: 2305941

Kalinga Institute of Industrial Technology
(Deemed to be University)
Bhubaneswar, India

Spring 2025-2026

Experiment Number	4
Experiment Title	Machine Learning for Cat and Dog Classification
Date of Experiment	14-01-2026
Date of Submission	21-01-2026

1. Objective:-To classify images as cats or dogs using machine learning models.

2. Code:-

a. train.py

```python
import cv2
import os
import numpy as np
import joblib
from skimage.feature import hog
from sklearn.svm import SVC
from sklearn.preprocessing import StandardScaler

data = []
labels = []

dataset_path = "dataset"
label_map = {"cat": 0, "dog": 1}

for label_name, label in label_map.items():
    folder = os.path.join(dataset_path, label_name)
    for img in os.listdir(folder):
        img_path = os.path.join(folder, img)
        image = cv2.imread(img_path, cv2.IMREAD_GRAYSCALE)
        image = cv2.resize(image, (64, 64))

        features = hog(image, pixels_per_cell=(8, 8),
                cells_per_block=(2, 2), block_norm="L2-Hys")

        data.append(features)
        labels.append(label)

X = np.array(data)
y = np.array(labels)

scaler = StandardScaler()
X = scaler.fit_transform(X)

model = SVC(kernel="linear", probability=True)
model.fit(X, y)

os.makedirs("models", exist_ok=True)
joblib.dump(model, "models/svm.pkl")
```

```python
joblib.dump(scaler, "models/scaler.pkl")
joblib.dump(label_map, "models/label_map.pkl")

print("✅ Model trained correctly")
```

b. app.py

```python
from flask import Flask, render_template, request
import cv2
import numpy as np
import joblib
from skimage.feature import hog

app = Flask(__name__)

model = joblib.load("models/svm.pkl")
scaler = joblib.load("models/scaler.pkl")
label_map = joblib.load("models/label_map.pkl")
inv_label_map = {v: k for k, v in label_map.items()}

@app.route("/", methods=["GET", "POST"])
def index():
    result = None

    if request.method == "POST":
        file = request.files["image"]

        img = cv2.imdecode(np.frombuffer(file.read(), np.uint8),
                    cv2.IMREAD_GRAYSCALE)
        img = cv2.resize(img, (64, 64))

        features = hog(img, pixels_per_cell=(8, 8),
                    cells_per_block=(2, 2), block_norm="L2-Hys")

        features = scaler.transform([features])
        pred = model.predict(features)[0]

        result = inv_label_map[pred].upper()

    return render_template("index.html", result=result)

if __name__ == "__main__":
    app.run(debug=True)
```

c. templates/index.html

```html
<!DOCTYPE html>
<html>
<head>
    <title>Cat vs Dog Classification</title>
    <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">
</head>
<body>
<div class="container">
    <h1>🐱 Cat vs Dog Classifier 🐶</h1>
```

```html
<form method="POST" enctype="multipart/form-data">
    <input type="file" name="image" required>

    <select name="model">
        <option value="svm">SVM</option>
        <option value="rf">Random Forest</option>
        <option value="lr">Logistic Regression</option>
        <option value="kmeans">K-Means</option>
    </select>

    <button type="submit">Predict</button>
</form>

{% if result %}
    <h2>Prediction: {{ result }}</h2>
{% endif %}
</div>
</body>
</html>
```

d. static/style.css

```css
body {
    font-family: Arial, sans-serif;
    background: #1e272e;
    color: white;
    display: flex;
    justify-content: center;
    align-items: center;
    height: 100vh;
}
.container {
    background: #2f3640;
    padding: 30px;
    border-radius: 10px;
    text-align: center;
}
input, select, button {
    margin: 10px;
    padding: 10px;
}
button {
    background: #44bd32;
    color: white;
    border: none;
    cursor: pointer;
}
```
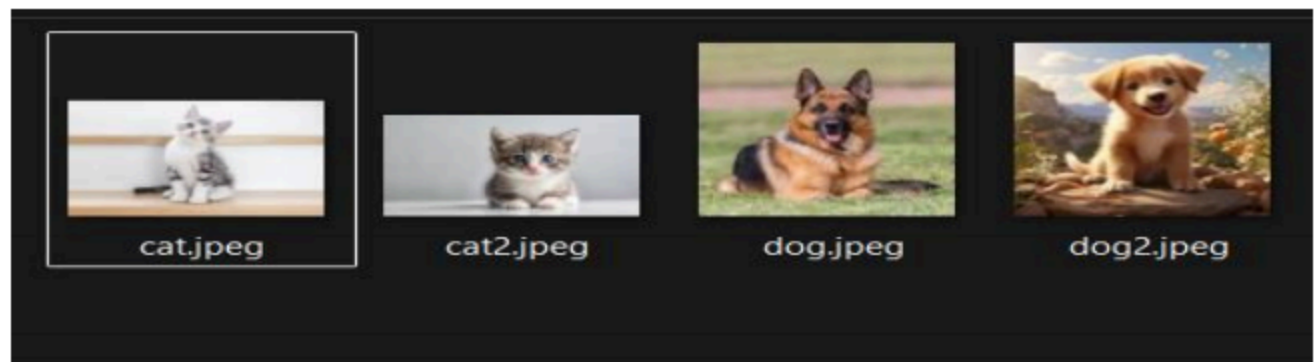
```
button:hover {
    background: #4cd137;
}
```

e. **Folder dataset: cat & dog**

f. **Folder models: label_map.pkl, scaler.pkl, svm.pkl**

3. Results/Output:- Classification result on the webpage

🐱 **Cat vs Dog Classifier** 🐶

Choose file | cat.jpeg   SVM ▾ | Predict

Prediction: CAT

🐱 **Cat vs Dog Classifier** 🐶

Choose file | cat2.jpeg   Random Forest ▾ | Predict

Prediction: CAT

🐱 **Cat vs Dog Classifier** 🐶

Choose file | dog.jpeg   Logistic Regression ▾ | Predict

Prediction: DOG

🐱 **Cat vs Dog Classifier** 🐶

Choose file | dog2.jpeg   K-Means ▾ | Predict

Prediction: DOG

Roll number: 2305941

Signature of the Student   Signature of the Lab Coordinator

_____   _____

(Name of the Student)   (Name of the Coordinator)