**Project Document**

**Project Title:**

**Pattern Sense: Classifying the Fabric Pattern Using Deep Learning**

---

## 1. Introduction

In the realm of the fashion and textile industry, the identification and classification of fabric patterns is traditionally a manual process. This project aims to automate this task using deep learning techniques, specifically Convolutional Neural Networks (CNNs). By leveraging image data of various fabric patterns, this system can accurately classify the type of pattern, such as checked, striped, floral, plain, and others. Automating this process reduces human error, improves efficiency in inventory sorting, and paves the way for scalable applications in e-commerce and smart manufacturing.

---

## 2. Ideation Phase

**Problem Statement:**

Manual classification of fabric patterns is time-consuming, inconsistent, and inefficient, especially when dealing with large inventories.

**Goal:**

To develop a deep learning model capable of automatically classifying fabric patterns into their respective categories using image data.

**Key Features:**

- Use of a labeled dataset containing different types of fabric patterns
- Image preprocessing and augmentation for robust training
- CNN-based model for classification
- Deployment of a basic web application for image upload and real-time prediction

---

## 3. Requirements Analysis

### 3.1 Functional Requirements:

- Load and preprocess dataset images
- Augment the image data to avoid overfitting
- Build and train a CNN for classification
- Evaluate model accuracy and performance
- Build a web interface for users to upload an image and receive prediction

### 3.2 Non-Functional Requirements:

- Model should achieve at least 85% accuracy
- Web application must be responsive and user-friendly
- Real-time prediction (in < 2 seconds)

### 3.3 Dataset Details:

- **Source:** Kaggle Dataset - nguyngiabol/dress-pattern-dataset
- **Categories:** Floral, Dotted, Checked, Striped, Plain
- **Size:** Approximately 1,000+ labeled images
- **Format:** JPEG/PNG images with metadata in CSV

### 3.4 Tools and Technologies:

- **Languages:** Python, HTML, CSS, JavaScript
- **Libraries:** TensorFlow/Keras, NumPy, OpenCV, Matplotlib
- **Frameworks:** Flask (for backend), Bootstrap (for frontend)
- **Environment:** Jupyter Notebook / VS Code

---

## 4. Results

```
print("\n📊 Confidence for all fabric patterns:")
for label, prob in zip(class_labels, prediction):
    print(f"{label:15s}: {prob * 100:.2f}%")

# Print top prediction
predicted_index = int(np.argmax(prediction))
print("\n✅ Most likely class:", class_labels[predicted_index])
print("🔍 Confidence:", round(prediction[predicted_index] * 100, 2), "%")
```

```
1/1 ━━━━━━━━━━━━━━━━━━━━ 1s 1s/step

📊 Confidence for all fabric patterns:
animal          : 1.87%
cartoon         : 2.80%
floral          : 0.31%
geometry        : 6.25%
ikat            : 18.64%
plain           : 13.42%
polka dot       : 21.31%
squares         : 7.08%
stripes         : 7.85%
tribal          : 10.24%

✅ Most likely class: polka dot
🔍 Confidence: 21.31 %
```

```
img_array = image.img_to_array(img) / 255.0
img_array = np.expand_dims(img_array, axis=0)

# Predict
prediction = model.pred

list: class_labels
(11 items) ['animal', 'cartoon', 'data_pattern', 'floral', 'geome

# Print confidence scor
print("\n📊 Confidence
for label, prob in zip(class_labels, prediction):
    print(f"{label:15s}: {prob * 100:.2f}%")

# Print top prediction
predicted_index = int(np.argmax(prediction))
print("\n✅ Most likely class:", class_labels[predicted_index])
print("🔍 Confidence:", round(prediction[predicted_index] * 100, 2), "%")
```

```
1/1 ━━━━━━━━━━━━━━━━━━━━ 2s 2s/step

📊 Confidence for all fabric patterns:
animal          : 3.04%
cartoon         : 7.29%
floral          : 0.28%
geometry        : 3.26%
ikat            : 16.30%
plain           : 4.16%
polka dot       : 18.94%
squares         : 19.90%
stripes         : 15.28%
tribal          : 4.10%

✅ Most likely class: squares
🔍 Confidence: 19.9 %
```

---

## 5. Conclusion

This project demonstrates the effective use of deep learning in the domain of textile image classification. By utilizing CNNs, we achieved high accuracy in classifying fabric patterns. The deployment of this model into a simple web app also proves its practicality and potential use in real-world applications, such as fashion inventory systems or automated e-commerce cataloging. Future enhancements could include using transfer learning with advanced architectures like ResNet or MobileNet and incorporating multi-label classification for complex patterns.

Thankyou

Name : ABHIRAM PAMPANI

EMAIL : pampaniabhiram@gmail.com