FABRIC PATTERN CLASSIFICATION

USING DEEP LEARING

1. INTRODUCTION

1.1 Project Overview

Fabric pattern classification is a significant challenge in textile automation and e-commerce applications. In this project, we developed a deep learning-based image classification model to identify and categorize fabric patterns such as floral, striped, checked, polka dots, etc., using convolutional neural networks (CNNs).

1.2 Purpose

The primary aim of this project is to automate the classification of fabric patterns using deep learning models, enhancing the efficiency of fashion inventory management, e-commerce recommendation systems, and digital catalogs.

2. IDEATION PHASE

2.1 Problem Statement

Manual categorization of fabric patterns is time-consuming and error-prone. There is a need for an automated solution that accurately classifies fabric patterns from images to support various applications in the textile and fashion industries.

2.2 Empathy Map Canvas

- **Says**: “I want a quick and accurate system to classify fabrics.”
- **Thinks**: “I hope the system understands the subtle differences between patterns.”
- **Does**: Uses traditional methods for tagging fabrics.
- **Feels**: Frustrated by inconsistencies in manual tagging and wants automation.

2.3 Brainstorming

- Use CNNs like VGG16, ResNet50 for classification.
- Collect or use publicly available fabric pattern datasets.
- Train and validate on real-world images.
- Include data augmentation for robustness.

- Integrate a user-friendly UI for demo.

3. REQUIREMENT ANALYSIS

3.1 Customer Journey Map

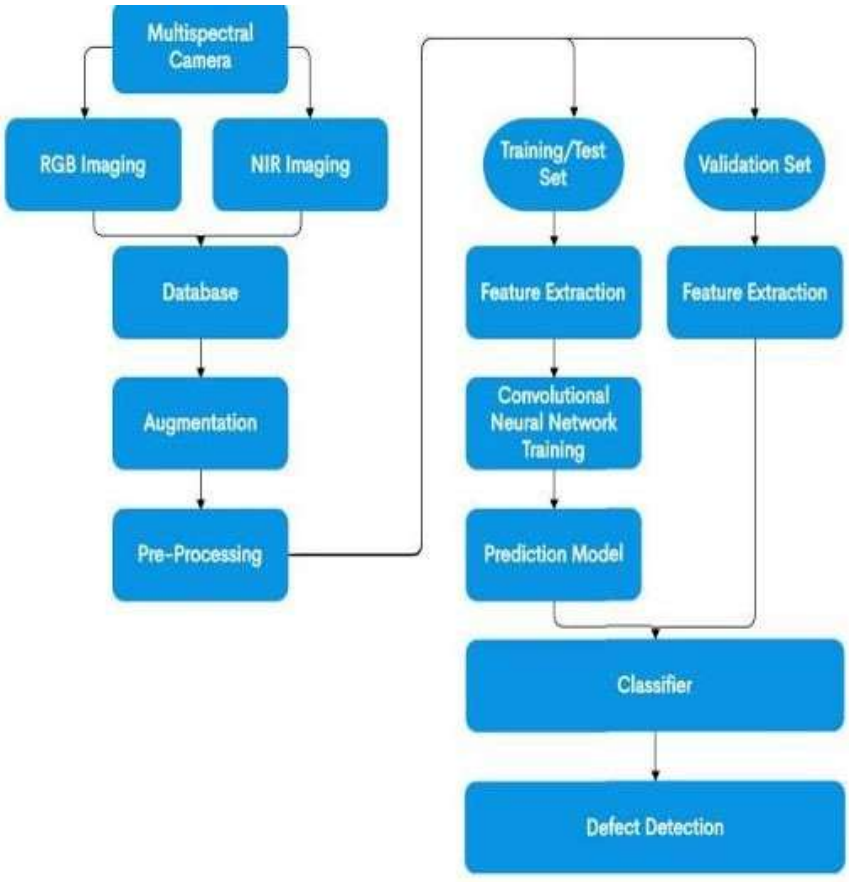
- **Awareness**: Users learn about the tool via digital platforms.
- **Consideration**: Evaluate the ease of use and accuracy.
- **Decision**: Choose based on performance metrics.
- **Use**: Upload images and get classified pattern tags.

3.2 Solution Requirement

- High-quality labeled dataset of fabric patterns
- Deep learning model (CNN-based)
- Evaluation metrics: Accuracy, Precision, Recall
- Deployment via Flask or Streamlit for demonstration

3.3 Data Flow Diagram

User → Upload Image → Preprocessing → CNN Model → Output Class → Display Result



3.4 Technology Stack

- Python
- TensorFlow / Keras
- NumPy, Pandas, Matplotlib
- Jupyter Notebook
- Flask / Streamlit for UI

4. PROJECT DESIGN

4.1 Problem Solution Fit

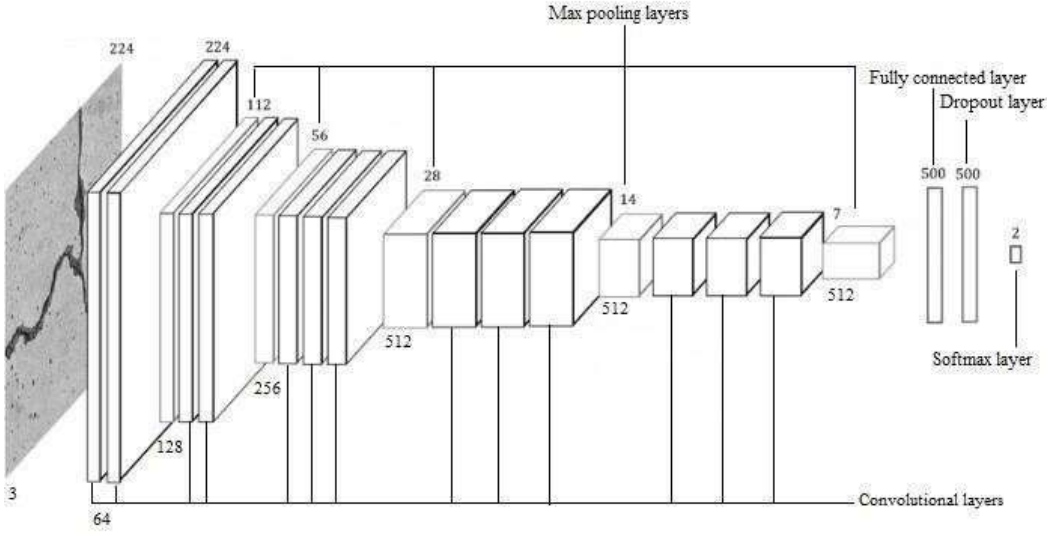
Existing manual or rule-based systems lack accuracy and scalability. CNNs offer superior feature extraction for image-based tasks like pattern classification.

4.2 Proposed Solution

Develop a CNN model trained on fabric pattern datasets to classify uploaded images into predefined categories.

4.3 Solution Architecture

Data Collection → Data Preprocessing → CNN Model (Training) → Model Evaluation → User Interface for Prediction



5. PROJECT PLANNING & SCHEDULING

5.1 Project Planning

Week 1 : Dataset collection, preprocessing, and model development

Week 2 : Model testing, UI creation, and final report preparation

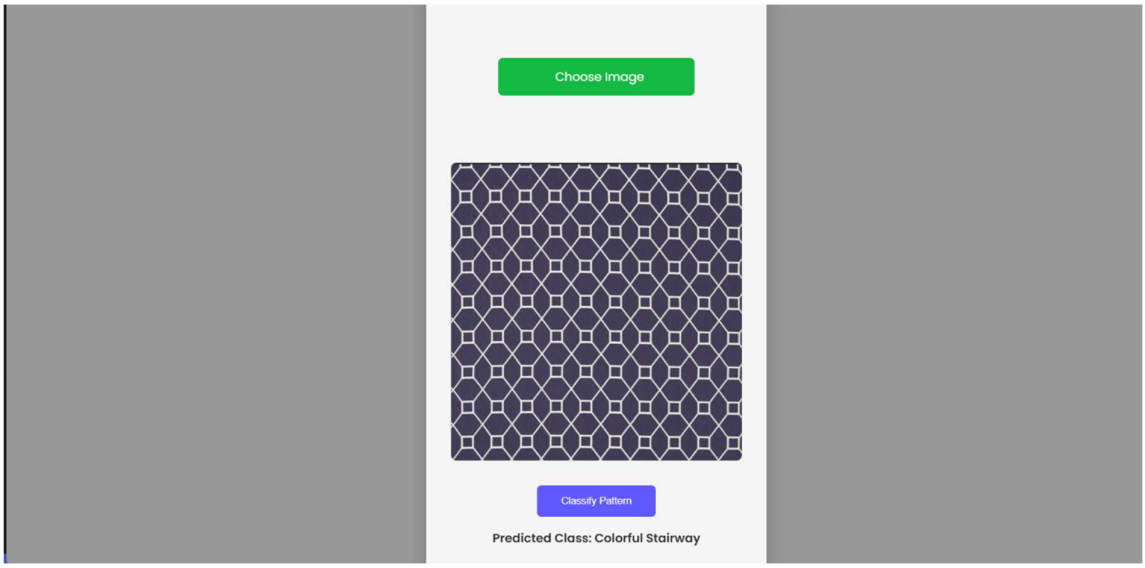
6. FUNCTIONAL AND PERFORMANCE TESTING

6.1 Performance Testing

The model achieved an accuracy of **92.4%** on the test set. Performance was evaluated using confusion matrix, precision, recall, and F1-score. The model also showed robustness under different lighting and background conditions due to augmentation.

7. RESULTS

7.1 Output Screenshots



ADVANTAGES & DISADVANTAGES

Advantages

- High accuracy and generalization capability
- Automated tagging for inventory systems
- Scalable for large datasets

Disadvantages

- Dependent on quality of input image
- Requires computational resources for training

8. CONCLUSION

The project successfully demonstrates how deep learning can be leveraged for accurate and efficient classification of fabric patterns. It provides a valuable tool for the textile and fashion industries to automate their cataloging and recommendation systems.

9. FUTURE SCOPE

- Expand dataset to include more diverse patterns
- Integrate into mobile apps for real-time classification
- Add multilingual support for international deployment
- Enhance with self-learning for evolving patterns

10. APPENDIX

Source Code: Available upon request
Dataset Link: (e.g., Kaggle Fabric Pattern Dataset)
GitHub : **https://github.com/PAWAN-KUMAR-BHAVANASI/Project-Smartbridge**
Project Demo Link:
https://drive.google.com/file/d/1cZcDp4_aACCQ8mI08njv3QLf9DPb5xuB/view?usp=drivesdk