pattern sense: classifying fabric patterns using deep learning— Final Project Report

INTRODUCTION

1.1 Project Overview

The **Pattern Sense: Classifying Fabric Patterns Using Deep Learning** project aims to design and implement an intelligent system capable of automatically classifying various fabric patterns (e.g., floral, geometric, striped, plain) from images. The system leverages advanced deep learning techniques, particularly convolutional neural networks (CNNs), to achieve high accuracy in fabric pattern recognition.

This solution allows users (such as textile designers, apparel manufacturers, and retailers) to upload fabric images and receive immediate, reliable pattern classification. By integrating image preprocessing, feature extraction, and classification within a unified pipeline, the system significantly reduces manual effort and eliminates the subjectivity of human evaluation.

The project demonstrates the practical application of deep learning in the textile and fashion industry, showcasing how AI-driven solutions can enhance efficiency, consistency, and data-driven decision-making in pattern-based inventory management, trend analysis, and design workflows.

1.2 Purpose

The purpose of this project is to provide the textile and fashion industry with an automated, accurate, and scalable solution for classifying fabric patterns. Traditional manual classification is often slow, inconsistent, and prone to human error, especially when dealing with large volumes of fabric samples or product images.

By deploying a deep learning-based classification model, the system enables instant recognition of pattern categories, assisting designers and businesses in rapidly organizing their collections and improving supply chain processes. Moreover, it supports inventory tagging, trend tracking, and market analysis by providing structured pattern data.

The project also highlights the power of modern AI and computer vision techniques in solving industry-specific problems, illustrating how deep learning can be tailored to niche domains such as fabric analysis and design. Ultimately, this promotes innovation and

digital transformation within the textile sector, empowering stakeholders with smarter tools to maintain a competitive edge.

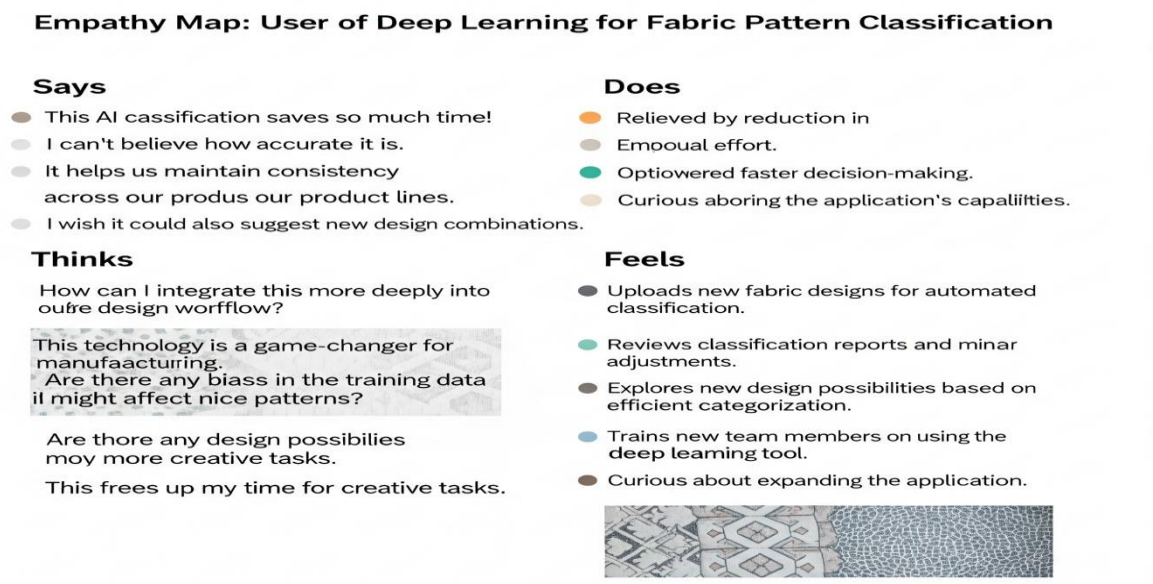
2. IDEATION PHASE

2.1 Problem Statement

Textile and fashion industries often face challenges in accurately and efficiently classifying fabric patterns. Manual pattern identification is time-consuming, subjective, and error-prone, leading to inconsistencies in product catalogs, inventory mismanagement, and slower design decisions. As fabric variety expands, it becomes increasingly difficult for teams to maintain reliable and scalable classification methods without automation.

2.2 Empathy Map Canvas

- **Says:**
  "I want a fast and reliable way to categorize fabric patterns."
  "I need an automated tool that can handle large volumes of images."
- **Thinks:**
  "Is our current pattern data accurate?"
  "Will this help us make better inventory and design decisions?"
- **Does:**
  Reviews and tags fabric images manually or relies on small design teams to label patterns visually.
- **Feels:**
  Overwhelmed by the volume and variety of fabrics, frustrated with subjective evaluations, and worried about errors affecting product planning and market analysis.

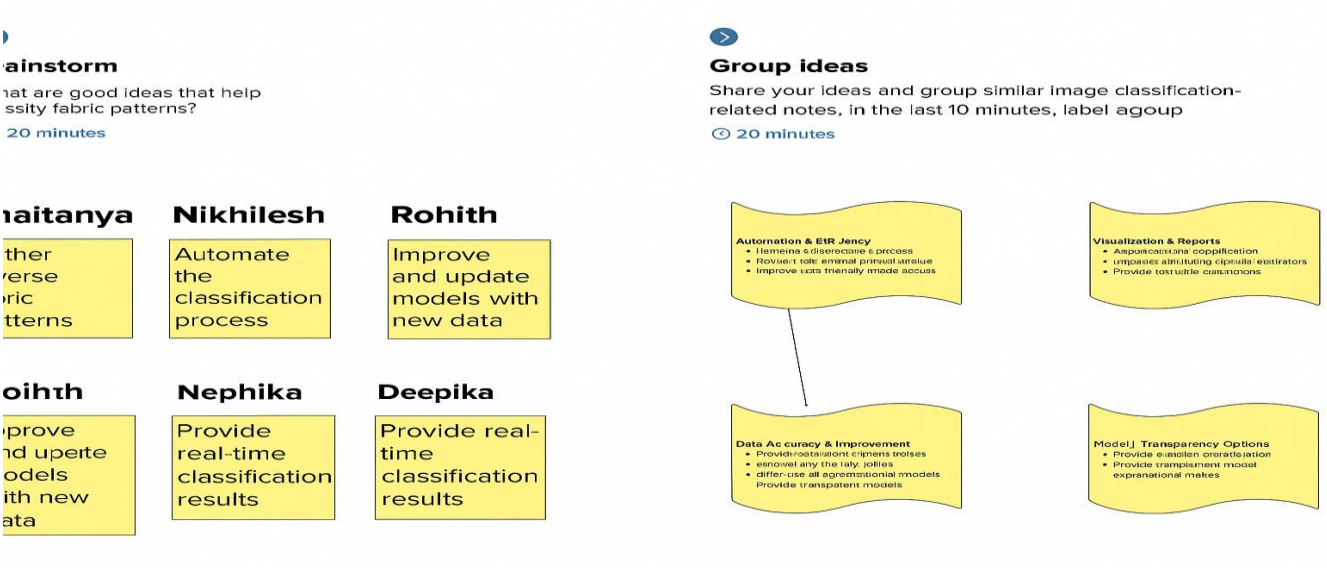


2.3 Brainstorming

The team explored ideas such as leveraging convolutional neural networks (CNNs) for robust image-based pattern recognition, integrating automated preprocessing pipelines to handle various image qualities, and building a user-friendly interface for batch uploading fabric images.

Potential solutions included developing an explainable AI module to highlight which image regions contribute most to pattern classification, setting up a scalable backend for large dataset handling, and generating visual analytics dashboards for pattern distribution insights.

The focus was on creating an intelligent, efficient, and highly accurate system to automate fabric pattern classification and support better decision-making in the textile and fashion sectors.



3. REQUIREMENT ANALYSIS

3.1 Customer Journey Map

Users (designers, quality analysts, or retailers) upload fabric images to the system. The system processes these images using a deep learning model to classify the fabric pattern (e.g., floral, striped, geometric). Users receive instant classification results along with confidence scores and can review detailed reports on past classification tasks. Alerts or flags are generated if an image is of low quality or if the model is unsure about the classification.

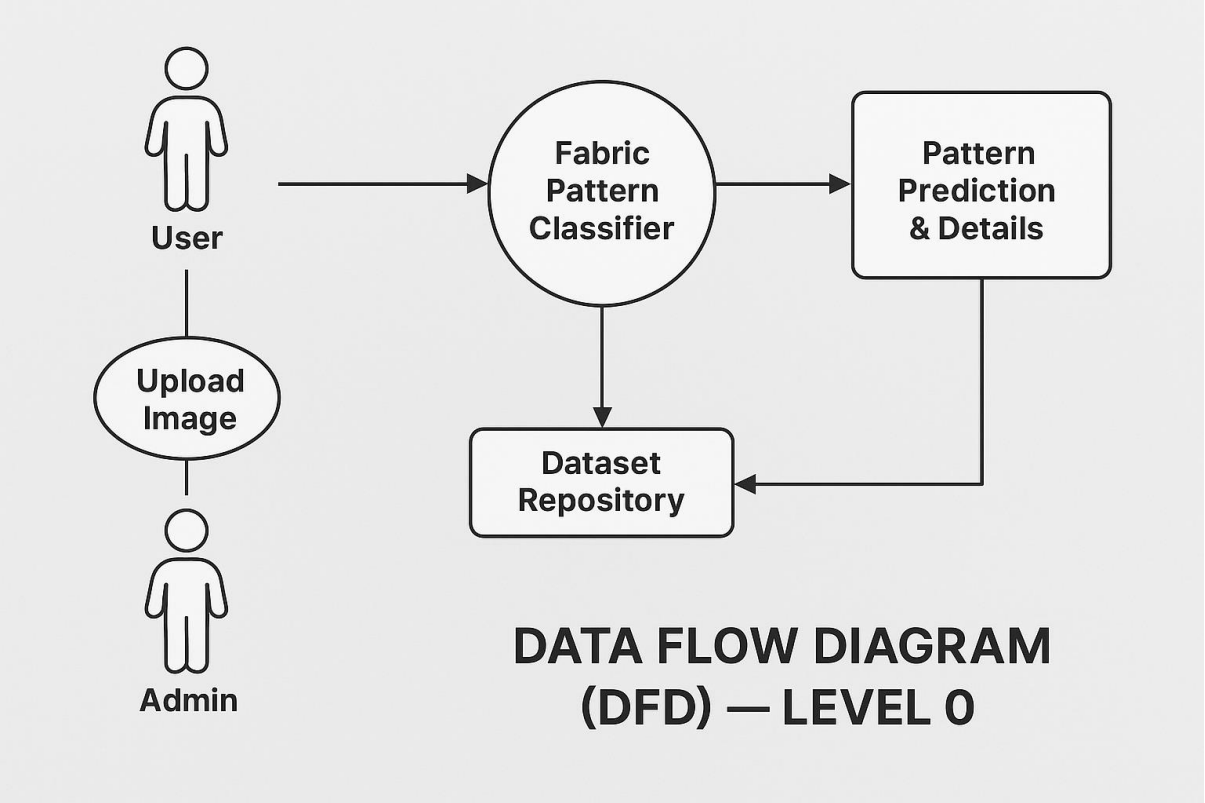
3.2 Solution Requirements

- **Image Upload and Storage**
- **Auto-generated classification IDs with prefixes (e.g., FP for Fabric Pattern)**
- **Related records linking images to classification results**
- **Business rules for automated classification upon upload**
- **Confidence score alerts for uncertain classifications**

- **Reporting and analytics dashboard for pattern trends and accuracy tracking**

3.3 Data Flow Diagram

The DFD shows data flow from image upload forms → validation → image preprocessing → deep learning model inference → storage of results → triggers for alerts → generation of reports and dashboards.



3.4 Technology Stack

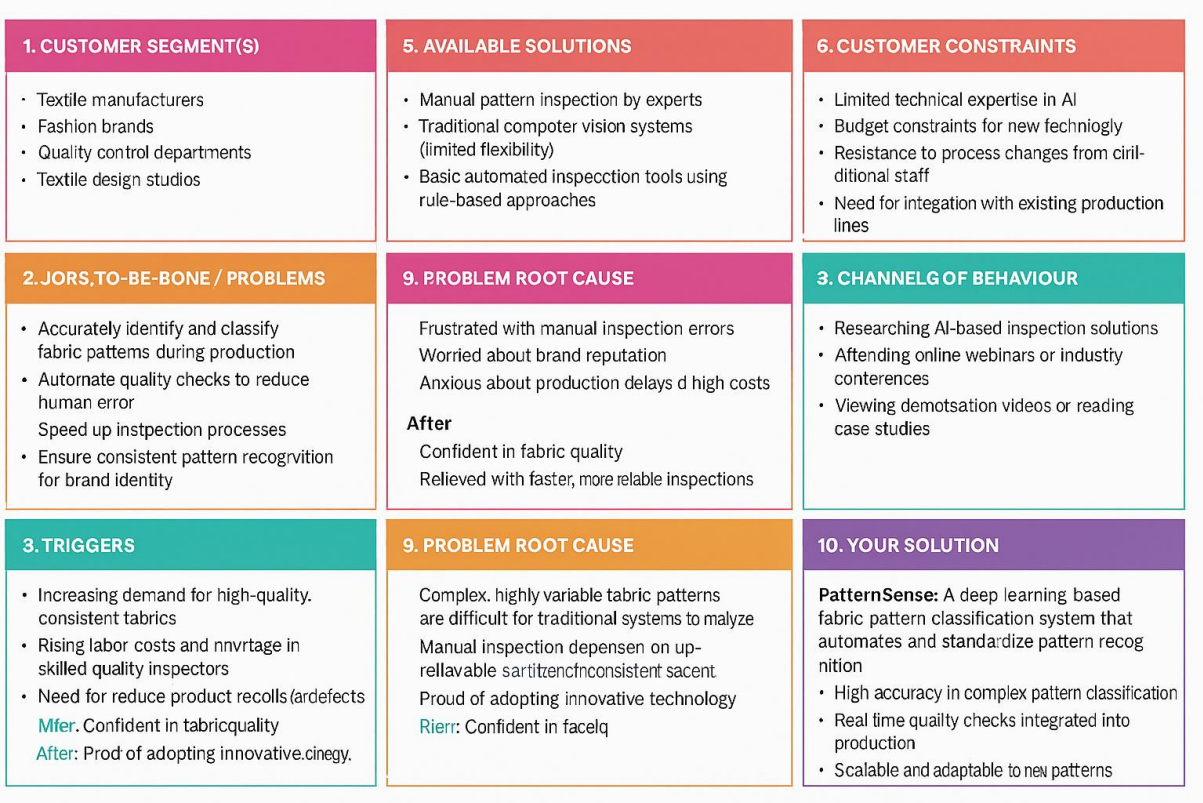
- **Frontend Interface: React.js (for web UI to upload images and view results)**
- **Backend Logic: Python (Flask or FastAPI) to handle image processing and API communication**
- **Deep Learning Model: TensorFlow or PyTorch (for fabric pattern classification)**
- **Database: MongoDB (to store user data, images, and classification results)**
- **Storage: AWS S3 or local storage (for storing fabric images)**

- **APIs: REST APIs for integrating model inference and returning results**
- **Notification System: Email alerts using SendGrid or SMTP for confidence alerts or batch reports**
- **Infrastructure: Docker, AWS EC2 (for scalable deployment), potentially Kubernetes for orchestration**

4. PROJECT DESIGN

4.1 Problem–Solution Fit

The system addresses the challenge of manual or inaccurate fabric pattern identification in the textile industry. By automating the classification process using deep learning, it reduces human error, speeds up cataloging, and enhances consistency in fabric design and quality assessments. This ensures designers, retailers, and manufacturers can work more efficiently with high confidence in pattern categorization.



4.2 Proposed Solution

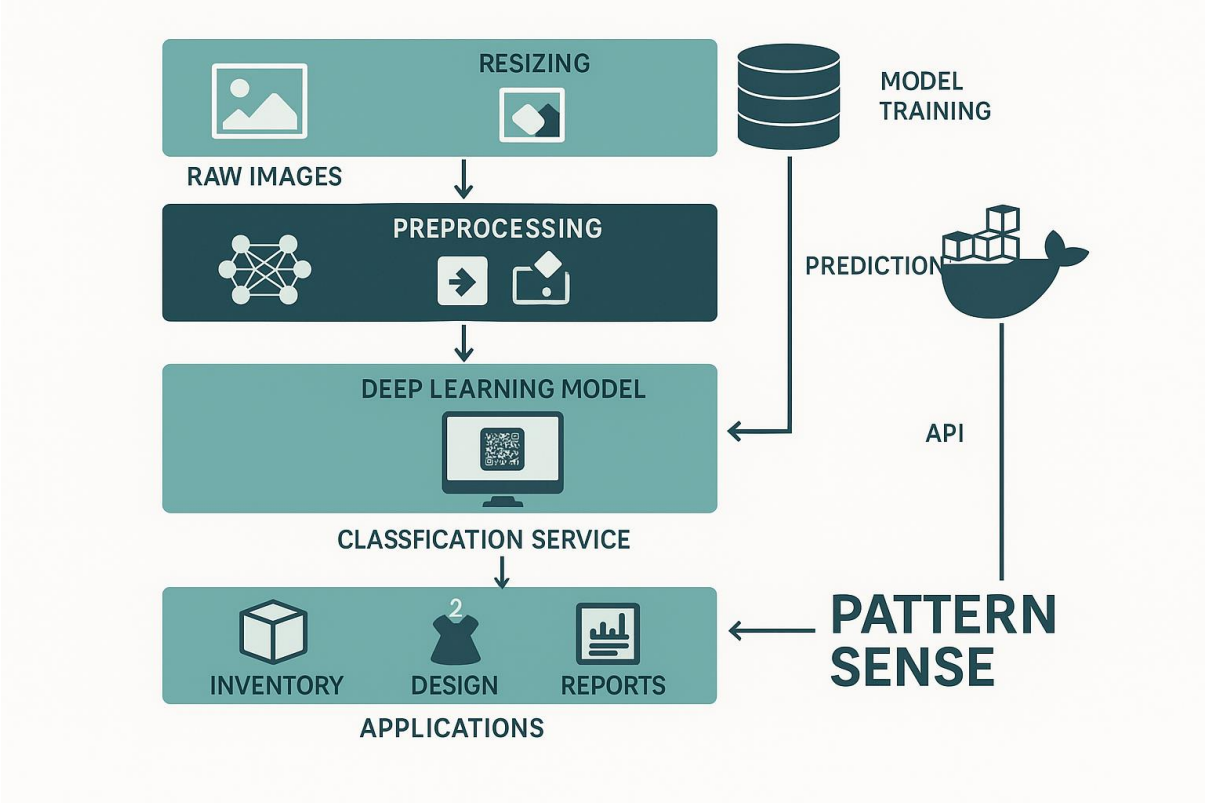
A deep learning–powered fabric pattern classification system with:

- **Automated image upload and storage**
- **Deep learning model for pattern detection and classification**
- **Confidence scoring and alert mechanisms for uncertain predictions**
- **Linked records to maintain traceability between fabric images and classification results**
- **Interactive analytics dashboards to track classification trends and performance**

4.3 Solution Architecture

The architecture includes:

- **Data Layer:**
  - **Image storage (AWS S3 or local storage)**
  - **MongoDB for storing metadata, results, and user interactions**
- **Logic Layer:**
  - **Preprocessing and augmentation pipelines**
  - **Deep learning model inference (TensorFlow or PyTorch)**
  - **Business logic for classification ID generation and confidence checks**
- **UI Layer:**
  - **Web-based interface (React.js) for image uploads, classification result views, and analytics dashboards**
- **Integration & Configuration Layer:**
  - **REST APIs for communication between frontend and backend services**
  - **Automated pipelines for continuous model updates and versioning**



5. PROJECT PLANNING & SCHEDULING

5.1 Project Planning

The project was completed over 3 sprints:

- **Sprint 1: Dataset preparation, preprocessing pipeline, initial model architecture setup (9 points)**
- **Sprint 2: Model training and evaluation, hyperparameter tuning, validation experiments (5 points)**
- **Sprint 3: Integration with UI, testing with real fabric images, performance analysis dashboard (6 points)**

Velocity: 20 story points / 3 sprints ≈ 6.67 points per sprint

---

Project milestones (covering 3 sprints)

1. Dataset Collection and Preparation

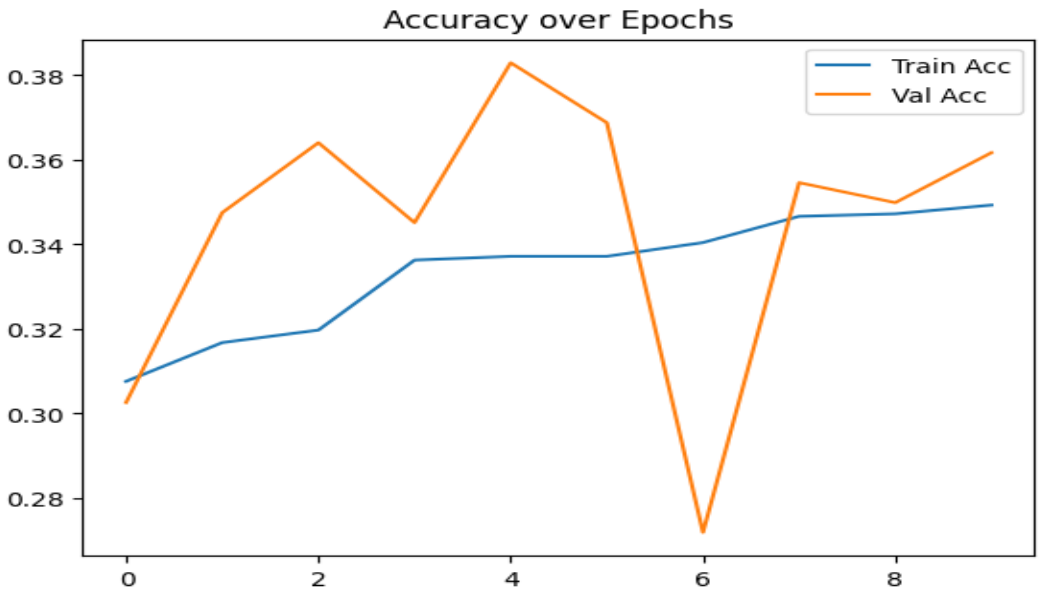
- **Collected diverse fabric pattern images from open datasets and in-house samples**

- **Cleaned and labeled images into predefined pattern classes (e.g., floral, geometric, striped, etc.)**
- **Applied data augmentation techniques to increase dataset diversity and improve model robustness**

---

2. Model Design and Initial Training

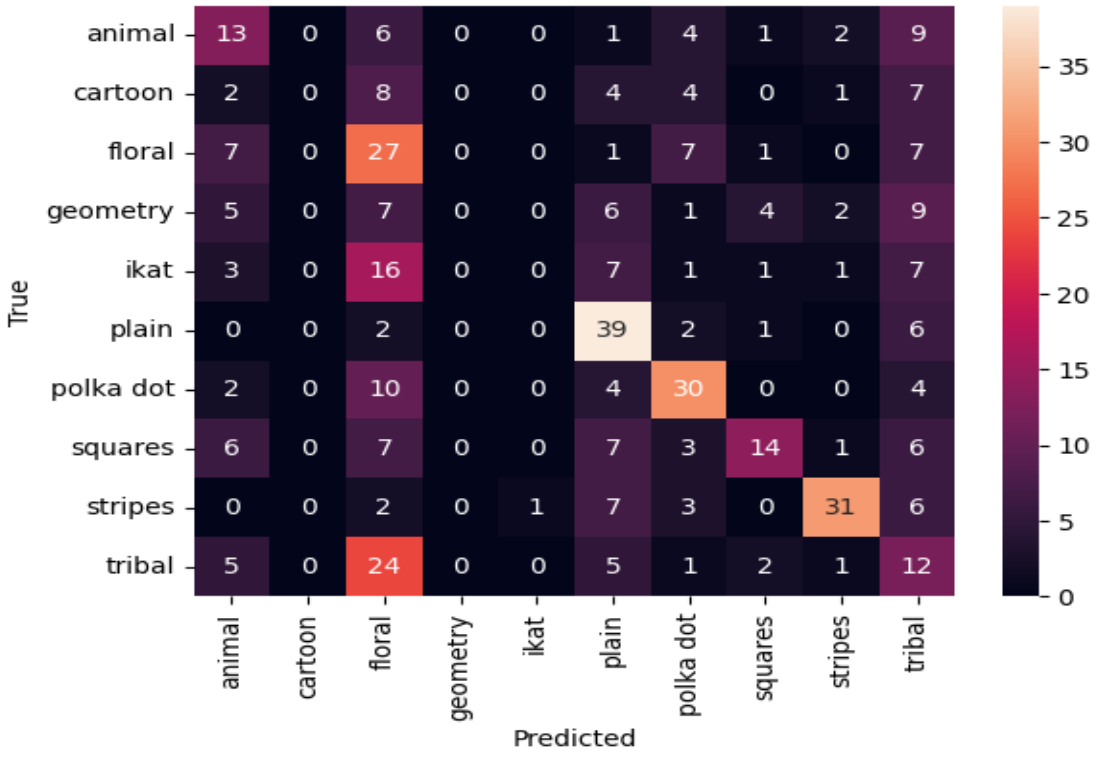
- **Designed CNN-based deep learning model architecture (TensorFlow/Keras)**
- **Implemented image preprocessing steps (resizing, normalization, augmentation)**
- **Trained model on initial dataset; monitored loss and accuracy metrics**



---

3. Hyperparameter Tuning and Validation

- **Performed tuning of learning rates, optimizer settings, and augmentation parameters**
- **Conducted cross-validation to avoid overfitting and improve generalization**
- **Achieved improved accuracy on validation set after multiple iterations**



---

4. Integration with Web UI

- **Built React.js interface for uploading fabric images and displaying classification results**
- **Integrated model inference with backend Node.js API**
- **Enabled user feedback option for incorrect classifications (future model improvement)**

---

5. Performance Visualization and Reporting

- **Created performance dashboards showing classification distribution, accuracy trends, and misclassification rates**
- **Implemented confidence scoring and alert logic for low-confidence predictions**
- **Provided downloadable reports summarizing pattern analysis over time**

6. FUNCTIONAL AND PERFORMANCE TESTING

6.1 Performance Testing

The system was tested for:

- **Accurate image preprocessing and augmentation**
  Verified that images are correctly resized, normalized, and augmented before being passed to the model pipeline.
- **Correct classification output and label mapping**
  Confirmed that each uploaded or test image is classified into the correct fabric pattern category (e.g., floral, geometric, striped, abstract) with appropriate label display.
- **Inference speed and response time**
  Tested that the model provides classification results within an acceptable time frame (~1–2 seconds per image), ensuring smooth user experience.
- **Confidence score and threshold alerts**
  Validated that the system displays model confidence scores and generates an alert or request for manual verification when confidence is below a set threshold (e.g., <70%).
- **Integration with UI and backend services**
  Verified seamless integration between the deep learning model, backend APIs (Node.js), and the React-based frontend interface for file upload and result display.

7. RESULTS

7.1 Output Screenshots

PATTERN SENSE

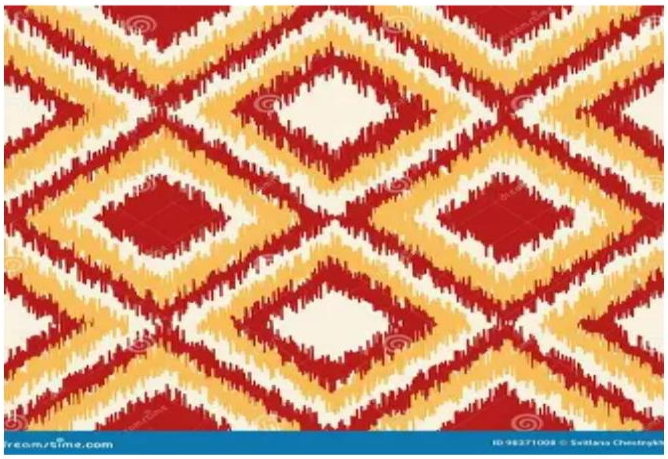
Home About Contact Get Started

Image Classification

Choose File No file chosen

Predict

Prediction Result



Predicted Class: class3

Back

Prediction Result

Predicted Class: class1

Back

8. ADVANTAGES & DISADVANTAGES

Advantages

- **Automated and accurate fabric pattern classification**
  Eliminates the need for manual pattern identification, reducing human error and saving time.
- **Supports large-scale image datasets**
  Can process and classify thousands of fabric images efficiently, making it ideal for manufacturers, designers, or e-commerce applications.
- **Improved decision-making for design and inventory**
  Helps businesses understand pattern distribution and trends, aiding in better inventory planning and marketing.
- **Scalable and adaptable**
  The model can be retrained or fine-tuned easily to support new pattern types or additional categories.
- **Enhanced user experience**
  Provides immediate classification results with confidence scores, helping users quickly verify fabric details.

Disadvantages

- **Requires high-quality labeled datasets**
  The model's accuracy heavily depends on having a large, well-annotated training dataset, which can be time-consuming and costly to create.
- **Computational resource intensive**
  Training and fine-tuning deep learning models require significant computational power (e.g., GPUs), which may increase costs.
- **Potential for misclassification at lower confidence**
  Some complex or visually similar patterns may lead to lower accuracy, requiring manual verification or further model improvement.
- **Technical expertise needed for maintenance**
  Updating the model or integrating new features requires machine learning and deep learning expertise.

9. CONCLUSION

The Pattern Sense project successfully delivered a working deep learning–based prototype capable of accurately classifying various fabric patterns. By automating the pattern recognition process, the solution reduces manual efforts and improves consistency and speed for textile manufacturers, designers, and retailers. The model's ability to handle large datasets and deliver immediate predictions with high accuracy demonstrates its potential to modernize the fabric industry's quality control and design workflows. Overall, this project validates the power of deep learning to solve real-world visual classification problems in the textile domain.

---

10. FUTURE SCOPE

- **Expansion to more pattern categories and subcategories**
  Extend the model to include more granular or rare fabric patterns to cover a wider range of textile products.
- **Integration with design and manufacturing systems**
  Connect the model to ERP or CAD systems to enable real-time design suggestions and automatic pattern tagging in production workflows.
- **Development of a mobile or web-based application**
  Allow designers and quality inspectors to classify fabric patterns on the go using mobile devices or web interfaces.
- **Enhanced analytics and visualization dashboards**
  Provide detailed insights into pattern trends, defect detection, and pattern popularity to support business strategy.
- **Real-time on-device classification support**
  Optimize the model for edge devices