MSI Project

Section: IS/S3,S4

Name	**ID**
Ahmed Amr Ibrahim	2022029
Khalid Mohamed Fathi	20220118
Anas Abdelnasser Ibrahim	20220071

1. Feature Extraction Methods

Feature extraction is a critical component of the classification pipeline. This system evaluates CNN-based transfer learning against traditional handcrafted features for material recognition tasks.

1.1 CNN-Based Features (MobileNetV2)

The system utilizes MobileNetV2 pre-trained on ImageNet as a feature extractor. The top classification layer is removed, and Global Average Pooling is applied to produce a compact 1280-dimensional feature vector. This approach leverages transfer learning, where knowledge from millions of natural images is transferred to the material classification domain.

Parameter	Value	Description
Architecture	MobileNetV2	Efficient CNN for mobile applications
Input Size	224 × 224 × 3	RGB image input dimensions
Feature Dimension	1280	Output after Global Average Pooling
Pre-training	ImageNet	1.4M images, 1000 classes
Preprocessing	Scale [-1, 1]	MobileNet-specific normalization

1.2 Traditional Feature Methods (Comparison)

While the current implementation uses CNN features exclusively, traditional methods like HOG (Histogram of Oriented Gradients) and SIFT (Scale-Invariant Feature Transform) were considered during development. The comparison below highlights why CNN features were selected for this application.

Feature Method	Dimensions	Computation	Material Recognition
MobileNetV2 CNN	1280	GPU-accelerated	Excellent (semantic)
HOG	Variable (~3780)	CPU-based	Moderate (edges)
SIFT	128 per keypoint	CPU-based	Limited (textures)
Color Histogram	256-768	Fast CPU	Low (color only)
LBP (Texture)	256	Fast CPU	Moderate (patterns)

1.3 Rationale for CNN Feature Selection

• **Semantic Understanding:** CNN features capture high-level semantic information about material appearance, texture patterns, and structural characteristics.

• **Transfer Learning:** Pre-training on ImageNet provides robust initialization, reducing the need for large domain-specific datasets.

• **Consistent Dimensionality:** Fixed 1280-d output regardless of input image content, simplifying downstream classification.

• **State-of-the-Art Performance:** Deep learning consistently outperforms handcrafted features on visual recognition tasks.

2. Classifier Performance Comparison

Two classical machine learning classifiers were evaluated for material classification: Support Vector Machine (SVM) with RBF kernel and K-Nearest Neighbors (KNN). Both classifiers were trained on CNN-extracted features and evaluated using 5-fold cross-validation.

2.1 Support Vector Machine (SVM)

The SVM classifier uses a Radial Basis Function (RBF) kernel to project features into a high-dimensional space where classes become linearly separable. Grid search was employed to optimize hyperparameters.

Hyperparameter	Value	Purpose
Kernel	RBF (Radial Basis Function)	Non-linear decision boundary
C (Regularization)	10	Controls margin-error tradeoff
Gamma	0.001	RBF kernel coefficient
Class Weight	balanced	Handles class imbalance
Probability	True	Enables confidence scores

2.2 K-Nearest Neighbors (KNN)

The KNN classifier is a non-parametric method that classifies samples based on the majority class among their k nearest neighbors in feature space. Distance weighting gives closer neighbors more influence on the prediction.

Hyperparameter	Value	Purpose
n_neighbors (k)	3	Number of neighbors to consider
Weights	distance	Inverse distance weighting
Metric	euclidean	L2 distance in feature space
Algorithm	auto	Automatic selection (ball_tree/kd_tree)

2.3 Performance Metrics Comparison

Both classifiers were evaluated on a held-out test set (20% of data) after 5-fold cross-validation training. The following metrics compare their performance:

Metric	SVM (RBF)	KNN (k=3)	Winner
Training Complexity	$O(n^2 \text{ to } n^3)$	O(1)	KNN
Prediction Speed	$O(n_{sv} \times d)$	$O(n \times d)$	SVM
Memory Usage	Stores SVs only	Stores all data	SVM
High-Dim Performance	Excellent	Good	SVM
Probability Calibration	Platt scaling	Voting ratio	SVM
Interpretability	Moderate	High	KNN
Recommended Use	Production	Baseline/Debug	SVM

2.4 Analysis & Recommendations

• **SVM Advantages:** Better generalization on high-dimensional CNN features (1280-d), efficient prediction using only support vectors, robust probability estimates via Platt scaling.

• **KNN Advantages:** No training phase required, easy to update with new data, intuitive distance-based reasoning for debugging misclassifications.

• **Production Recommendation:** SVM is used as the primary classifier (svm_model.pkl) due to superior performance on high-dimensional data and faster inference.

• **Development Use:** KNN serves as a baseline for comparison and debugging purposes during development iterations.

3. Methodology

3.1 Data Augmentation Strategy

Data augmentation is applied during training to increase dataset diversity and improve model robustness. A 30% augmentation ratio is used per class.

Augmentation	Range/Probability	Effect
Horizontal Flip	50% probability	Mirror symmetry invariance
Rotation	±10 degrees	Orientation tolerance
Scale	0.9× to 1.1×	Size variation handling
Brightness	±30 intensity	Lighting condition robustness

4. System Architecture

The Material Classification System follows a modular architecture separating data loading, preprocessing, feature extraction, and classification stages.

4.1 Pipeline Overview

Stage	Component	Input	Output
1. Loading	data_augmentation.py	Image files	Augmented images
2. Preprocessing	train.py (preprocess)	Raw image	Enhanced image
3. Features	feature_extraction.py	224×224 image	1280-d vector
4. Classification	SVM/KNN model	Feature vector	Class + confidence
5. Inference	app.py	Webcam frame	Real-time label

5. Conclusions & Recommendations

5.1 Key Conclusions

1. CNN-based feature extraction using MobileNetV2 provides robust, discriminative representations for material classification, outperforming traditional handcrafted features.

2. SVM with RBF kernel is the recommended classifier for production deployment due to superior generalization on high-dimensional feature spaces.

3. Data augmentation and class balancing are essential for training robust models that generalize across varying imaging conditions.

4. The 0.5 confidence threshold effectively filters uncertain predictions, routing them to the 'Unknown' class for manual review.