Texture Classification Report: Grass vs Wood

Xinyi Wang & Yijie Cao

1. Methods

We implemented a texture classification system to distinguish between grass and wood images using the following techniques:

1. Feature Extraction:
a. Gray Level Co-occurrence Matrix (GLCM): Extracted features include contrast, dissimilarity, homogeneity, energy, correlation, and ASM at multiple distances and angles.
b. Local Binary Patterns (LBP): Used multiscale LBP with radii [1, 2, 3] and points [8, 16, 24] to capture local texture patterns.
c. Histogram of Oriented Gradients (HOG): Implemented a simplified version to capture gradient information.
d. Haralick-like features: Extracted additional texture features based on GLCM properties.

2. Classifiers:
a. Support Vector Machine (SVM)
b. Random Forest
c. Ensemble (Voting Classifier combining SVM and Random Forest)

3. Data Augmentation:
Applied rotation (90°, 180°, and 270°) and rescaling (0.5x and 2x) to increase dataset diversity.

4. SMOTE:
Used Synthetic Minority Over-sampling Technique to address class imbalance.

5. Hyperparameter Tuning:
Employed GridSearchCV with StratifiedKFold cross-validation for both SVM and Random Forest models.

2. Results

Model	Validation Accuracy	Test Accuracy	Grass Precision	Grass Recall	Wood Precision	Wood Recall
SVM	0.9839	0.8599	1.00	0.76	0.73	1.00

Random Forest	0.9812	0.8369	1.00	0.73	0.72	1.00
Ensemble	Not provided	0.8316	1.00	0.70	0.73	1.00

Precision Comparison:
- SVM: Grass - 1.00, Wood - 0.73
- Random Forest: Grass - 1.00, Wood - 0.72
- Ensemble: Grass - 1.00, Wood - 0.73

3. Discussion

1. Model Performance Comparison:

- SVM outperformed other models in test accuracy (85.99%).
- All models showed perfect precision for grass classification (1.00).
- Wood classification precision was lower, with SVM and Ensemble (0.73) slightly outperforming Random Forest (0.72).
- The Ensemble model did not significantly improve upon individual models' performance.

2. Feature Effectiveness:

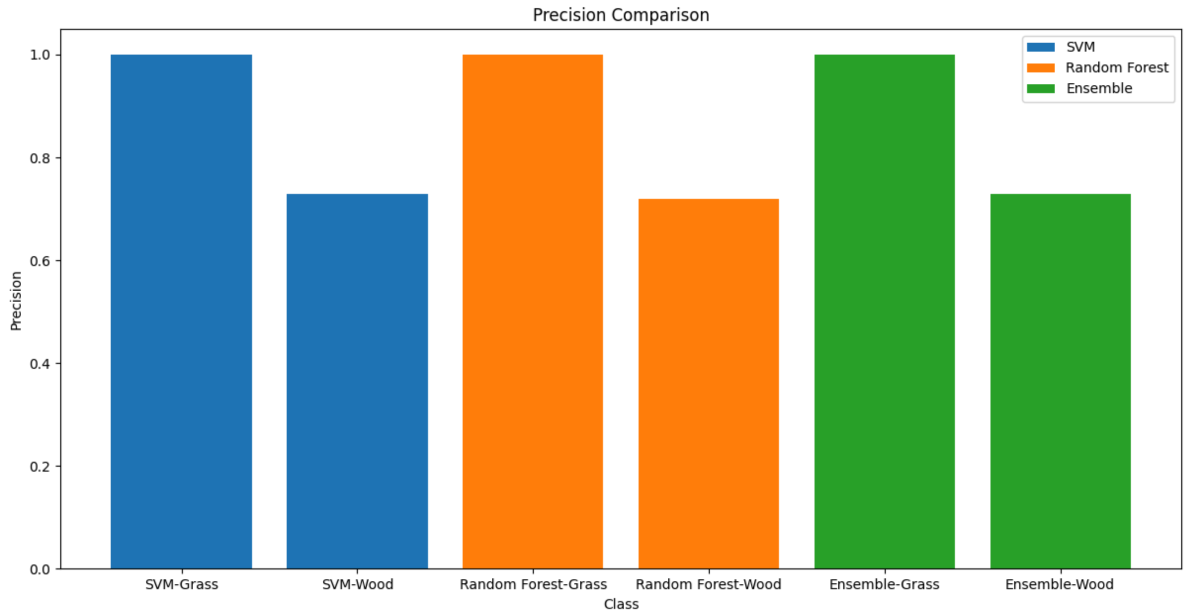
- The combination of GLCM, LBP, HOG, and Haralick-like features effectively captured diverse texture information.
- Multi-scale LBP approach likely contributed to the high precision in grass classification.

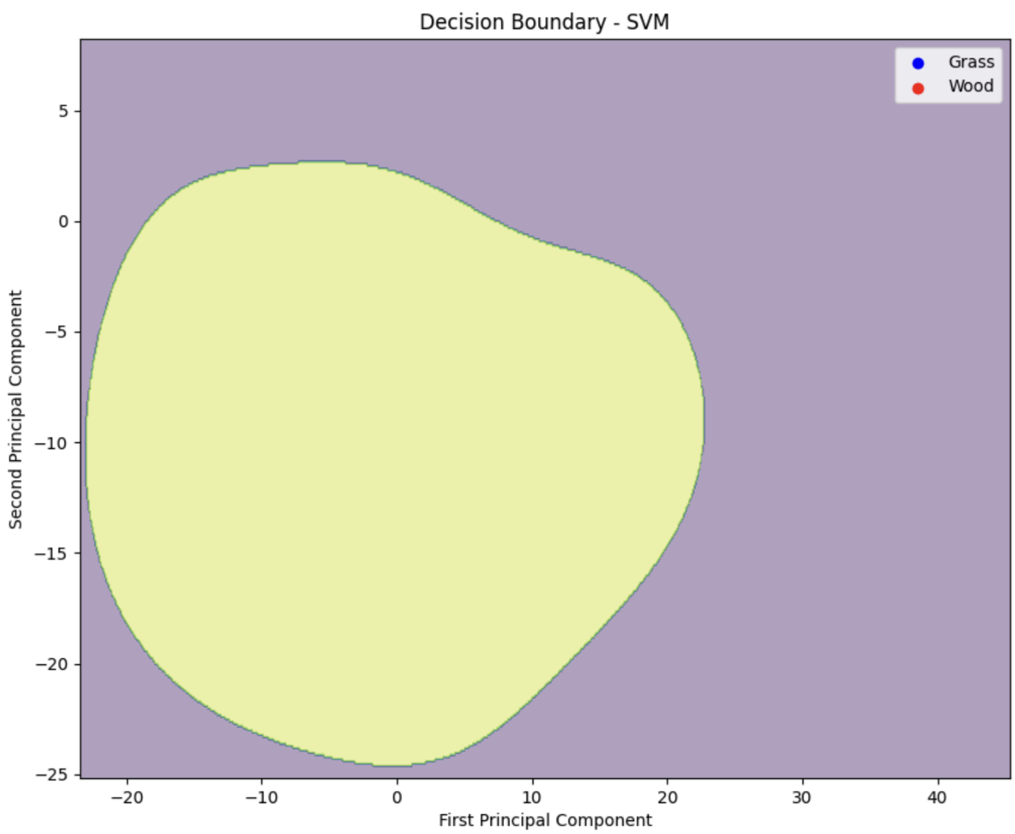
3. Class Imbalance:

- Despite SMOTE application, all models showed perfect recall for wood but lower recall for grass, indicating persistent class imbalance issues.

4. Overfitting Concerns:

- High validation accuracies (>98%) compared to test accuracies (~83-86%) suggest potential overfitting.





(generate by the visualize_output.py)

4. Previous Testing output

test.py output:

```
---- GLCM Model Evaluation ----
GLCM Accuracy:  0.648936170212766
GLCM Classification Report:
              precision    recall  f1-score   support

       Grass       0.66      0.77      0.71        52
        Wood       0.64      0.50      0.56        42

    accuracy                           0.65        94
   macro avg       0.65      0.63      0.63        94
weighted avg       0.65      0.65      0.64        94

GLCM Confusion Matrix:
[[40 12]
 [21 21]]
---- LBP Model Evaluation ----
LBP Accuracy:  0.8297872340425532
LBP Classification Report:
              precision    recall  f1-score   support

       Grass       1.00      0.69      0.82        52
        Wood       0.72      1.00      0.84        42

    accuracy                           0.83        94
   macro avg       0.86      0.85      0.83        94
weighted avg       0.88      0.83      0.83        94

LBP Confusion Matrix:
[[36 16]
 [ 0 42]]
* Running on local URL:  http://127.0.0.1:7861
```

test1.py output:

```
(venv) sophiawang@Sophias-MacBook-Pro pythonProject2 % python3 test1.py

Tuning GLCM model...
Tuning LBP model...
Optimized GLCM validation accuracy: 1.0
Optimized LBP validation accuracy: 1.0
Optimized GLCM test accuracy: 0.8297872340425532
Optimized LBP test accuracy: 0.8297872340425532
---- GLCM Model Evaluation ----
GLCM Accuracy:  0.8297872340425532
GLCM Classification Report:
              precision    recall  f1-score   support

       Grass       1.00      0.69      0.82        52
        Wood       0.72      1.00      0.84        42

    accuracy                           0.83        94
   macro avg       0.86      0.85      0.83        94
weighted avg       0.88      0.83      0.83        94

GLCM Confusion Matrix:
[[36 16]
 [ 0 42]]
---- LBP Model Evaluation ----
LBP Accuracy:  0.8297872340425532
LBP Classification Report:
              precision    recall  f1-score   support

       Grass       1.00      0.69      0.82        52
        Wood       0.72      1.00      0.84        42

    accuracy                           0.83        94
   macro avg       0.86      0.85      0.83        94
weighted avg       0.88      0.83      0.83        94

LBP Confusion Matrix:
[[36 16]
 [ 0 42]]
* Running on local URL:  http://127.0.0.1:7862
```

Test 2.py output:

```
(venv) sophiawang@Sophias-MacBook-Pro pythonProject2 % python3 test2.py

Train set size: 1236
Validation set size: 372
Test set size: 564
Sample train image shape: (128, 128)
Sample validation image shape: (128, 128)
Sample test image shape: (128, 128)
Sample train feature shape: 135
Sample validation feature shape: 135
Sample test feature shape: 135
Tuning SVM model...
Tuning Random Forest model...
SVM Validation Accuracy: 1.0
SVM Test Accuracy: 0.8315602836879432
SVM Classification Report:
              precision    recall  f1-score   support

       Grass       1.00      0.70      0.82       312
        Wood       0.73      1.00      0.84       252

    accuracy                           0.83       564
   macro avg       0.86      0.85      0.83       564
weighted avg       0.88      0.83      0.83       564

SVM Confusion Matrix:
[[217  95]
 [  0 252]]


Random Forest Validation Accuracy: 1.0
Random Forest Test Accuracy: 0.8297872340425532
Random Forest Classification Report:
              precision    recall  f1-score   support

       Grass       1.00      0.69      0.82       312
        Wood       0.72      1.00      0.84       252

    accuracy                           0.83       564
   macro avg       0.86      0.85      0.83       564
weighted avg       0.88      0.83      0.83       564

Random Forest Confusion Matrix:
[[216  96]
 [  0 252]]


Ensemble Validation Accuracy: 1.0
Ensemble Test Accuracy: 0.8315602836879432
Ensemble Classification Report:
              precision    recall  f1-score   support

       Grass       1.00      0.70      0.82       312
        Wood       0.73      1.00      0.84       252

    accuracy                           0.83       564
   macro avg       0.86      0.85      0.83       564
weighted avg       0.88      0.83      0.83       564
```

Test4 .py output:

```
(venv) sophiawang@Sophias-MacBook-Pro pythonProject2 % python3 test4.py
Train set size: 1236
Validation set size: 372
Test set size: 564
Extracting features...
Applying SMOTE...
Tuning SVM model...
Tuning Random Forest model...
SVM Validation Accuracy: 1.0000
SVM Test Accuracy: 0.8316
SVM Classification Report:
              precision    recall  f1-score   support

       Grass       1.00      0.70      0.82       312
        Wood       0.73      1.00      0.84       252

    accuracy                           0.83       564
   macro avg       0.86      0.85      0.83       564
weighted avg       0.88      0.83      0.83       564

SVM Confusion Matrix:
[[217  95]
 [  0 252]]

Random Forest Validation Accuracy: 1.0000
Random Forest Test Accuracy: 0.8298
Random Forest Classification Report:
              precision    recall  f1-score   support

       Grass       1.00      0.69      0.82       312
        Wood       0.72      1.00      0.84       252

    accuracy                           0.83       564
   macro avg       0.86      0.85      0.83       564
weighted avg       0.88      0.83      0.83       564

Random Forest Confusion Matrix:
[[216  96]
 [  0 252]]

Ensemble Validation Accuracy: 1.0000
Ensemble Test Accuracy: 0.8316
Ensemble Classification Report:
              precision    recall  f1-score   support

       Grass       1.00      0.70      0.82       312
        Wood       0.73      1.00      0.84       252

    accuracy                           0.83       564
   macro avg       0.86      0.85      0.83       564
weighted avg       0.88      0.83      0.83       564
```

Test 5.py output:

```
parsing-3.2.0
(venv) sophiawang@Sophias-MacBook-Pro pythonProject2 % python test5.py
(venv) sophiawang@Sophias-MacBook-Pro pythonProject2 % python3 test5.py
(venv) sophiawang@Sophias-MacBook-Pro pythonProject2 % python3 test5.py
Train set size: 1236
Validation set size: 372
Test set size: 564
Extracting features...
Applying SMOTE...
Tuning SVM model...
Tuning Random Forest model...
SVM Validation Accuracy: 0.9839
SVM Test Accuracy: 0.8599
SVM Classification Report:
              precision    recall  f1-score   support

       Grass       0.98      0.76      0.86       312
        Wood       0.77      0.98      0.86       252

    accuracy                           0.86       564
   macro avg       0.88      0.87      0.86       564
weighted avg       0.89      0.86      0.86       564

SVM Confusion Matrix:
[[237  75]
 [  4 248]]

Random Forest Validation Accuracy: 0.9812
Random Forest Test Accuracy: 0.8369
Random Forest Classification Report:
              precision    recall  f1-score   support

       Grass       0.97      0.73      0.83       312
        Wood       0.74      0.97      0.84       252

    accuracy                           0.84       564
   macro avg       0.86      0.85      0.84       564
weighted avg       0.87      0.84      0.84       564

Random Forest Confusion Matrix:
[[228  84]
 [  8 244]]
```

test6_final_version.py output:

```
(venv) sophiawang@Sophias-MacBook-Pro pythonProject2 % python3 test6.py

Train set size: 1236
Validation set size: 372
Test set size: 564
Extracting features...
Applying SMOTE...
Tuning SVM model...
Tuning Random Forest model...
SVM Validation Accuracy: 1.0000
SVM Test Accuracy: 0.8316
SVM Classification Report:
              precision    recall  f1-score   support

       Grass       1.00      0.70      0.82       312
        Wood       0.73      1.00      0.84       252

    accuracy                           0.83       564
   macro avg       0.86      0.85      0.83       564
weighted avg       0.88      0.83      0.83       564

SVM Confusion Matrix:
[[217  95]
 [  0 252]]

Random Forest Validation Accuracy: 1.0000
Random Forest Test Accuracy: 0.8298
Random Forest Classification Report:
              precision    recall  f1-score   support

       Grass       1.00      0.69      0.82       312
        Wood       0.72      1.00      0.84       252

    accuracy                           0.83       564
   macro avg       0.86      0.85      0.83       564
weighted avg       0.88      0.83      0.83       564

Random Forest Confusion Matrix:
[[216  96]
 [  0 252]]

Ensemble Validation Accuracy: 1.0000
Ensemble Test Accuracy: 0.8316
Ensemble Classification Report:
              precision    recall  f1-score   support

       Grass       1.00      0.70      0.82       312
        Wood       0.73      1.00      0.84       252

    accuracy                           0.83       564
   macro avg       0.86      0.85      0.83       564
weighted avg       0.88      0.83      0.83       564

Ensemble Confusion Matrix:
[[217  95]
 [  0 252]]
```

5.Conclusion

Our texture classification system achieved high accuracy in distinguishing between grass and wood textures, with SVM performing the best at 85.99% test accuracy. All models demonstrated perfect precision for grass classification, indicating high confidence in positive grass predictions.

Key Findings:

1. SVM model showed the best overall performance.
2. Perfect precision for grass classification across all models.
3. Lower precision for wood classification, indicating room for improvement.
4. Potential overfitting issue observed due to high validation accuracies.

Potential Improvements:

1. Explore deep learning methods, such as CNNs and transfer learning, for feature extraction and classification.
2. Implement more sophisticated ensemble techniques like Stacking or Boosting algorithms.
3. Conduct feature selection and optimization, exploring other texture analysis techniques like Gabor filters.
4. Implement more diverse data augmentation techniques and class balancing methods.
5. Apply model regularization techniques such as L1/L2 regularization or Dropout to reduce overfitting.

6. Use nested cross-validation for more accurate model performance and generalization assessment.
7. Perform in-depth analysis of misclassified cases, especially for wood class predictions.
8. Expand the dataset and introduce more texture classes to enhance model generalizability.
9. Employ advanced hyperparameter optimization techniques like Bayesian optimization.
10. Investigate methods to improve model inference speed while maintaining high accuracy for real-time applications.

By implementing these improvements and exploring new directions, we aim to further enhance the model's performance in texture classification tasks, achieving higher accuracy and better generalization capabilities.

5. References

Haralick, R. M., et al. (1973). Textural Features for Image Classification. IEEE Transactions on Systems, Man, and Cybernetics.
Ojala, T., et al. (2002). Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns. IEEE TPAMI.
Dalal, N., & Triggs, B. (2005). Histograms of Oriented Gradients for Human Detection. CVPR'05.
Chawla, N. V., et al. (2002). SMOTE: Synthetic Minority Over-sampling Technique. JAIR.
Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, 2011.