Classification on Describable Textures Dataset (DTD)

Harshita Mandalika
Dept. of CSCE
Texas A&M University
College Station, USA
harshita.mandalika@tamu.edu

Vishwam Raval
Dept. of ECEN
Texas A&M University
College Station, USA
vishwam@tamu.edu

Chetan Sai Borra
Dept. of ECEN
Texas A&M University
College Station, USA
chetansai2003@tamu.edu

Xuanyu Liu
Dept. of CSCE
Texas A&M University
College Station, USA
lxugrad2025_usa@tamu.edu

***Abstract*—The Describable Textures Dataset (DTD) is a widely used benchmark for fine-grained texture recognition with 47 visually diverse texture categories. This work presents a comparative study of a custom convolutional neural network (SimpleCNN), a deeper residual architecture (ResNet-18), and InceptionV3 trained from scratch on DTD. We first analyze the dataset through descriptive statistics and exploratory data analysis, then describe the model architectures, training strategies, and hyperparameter search procedures. Experimental results show that ResNet-18 achieves an overall test accuracy of 45.53%, a precision of 45.43%, recall of 45.53%, and F1-score of 45.10%, outperforming both the SimpleCNN baseline and InceptionV3. The confusion matrix analysis further highlights systematic confusions between visually similar texture categories. We further discuss interpretability and potential application-oriented insights derived from the model behavior.**

***Index Terms*—Texture classification, Describable Textures Dataset (DTD), Convolutional Neural Networks, ResNet-18, Deep learning.**

I. Introduction

Texture understanding plays a significant role in computer vision, supporting material recognition, industrial inspection, scene understanding, and content-based image retrieval [1], [2]. Early work in texture analysis relied on curated photographic collections such as the Brodatz album [2], followed by the development of hand-crafted descriptors like Local Binary Patterns (LBP), which improved robustness to illumination and rotation [3]. The Describable Textures Dataset (DTD) introduced a more challenging benchmark containing natural, in-the-wild textures annotated with 47 human-interpretable attributes [1]. These advances provide the foundation for modern supervised texture classification approaches used in deep learning research.

In this paper, we investigate supervised texture classification on DTD using three convolutional architectures: a custom SimpleCNN baseline, a deeper residual network, ResNet-18 and InceptionV3. Our goals are to (i) characterize the dataset through descriptive statistics and exploratory analysis, (ii) compare the performance of a lightweight CNN, a residual model trained from scratch and InceptionV3, and (iii) study model behavior through class-wise metrics and confusion patterns to derive interpretability and practical insights.

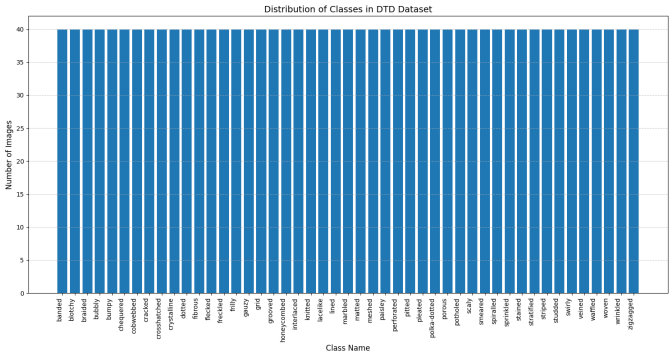

Fig. 1. Class distribution for the DTD dataset. The dataset contains 47 classes, each uniformly balanced.

II. Method

A. Data Preparation

The Describable Textures Dataset (DTD) contains 5,640 images across 47 texture categories, with 1,880 images each in the training, validation, and test splits provided by the dataset authors. Since DTD is already clean and balanced, no manual data cleansing was required. All images were loaded using the PyTorch DTD dataset class, and their original file structure was preserved. Prior to training, images are resized and cropped according to the model-specific preprocessing pipelines described in the following subsections.

B. Exploratory Data Analysis

The Describable Textures Dataset (DTD) contains 5,640 images grouped into 47 texture classes, with 1,880 images assigned to each of the training, validation, and test sets. Each class contributes exactly 120 images, making the dataset fully balanced. Image resolutions range from 271×231 pixels to 900×778 pixels, with an average size of about 496×451 pixels. The mean RGB values across all images are 0.528, 0.471, and 0.423, and the corresponding standard deviations are 0.180, 0.182, and 0.178. These statistics provide a clear overview of the dataset's scale, structure, and color characteristics.

As shown in Figure 1, the DTD dataset is uniformly balanced. Example samples from a subset of categories are

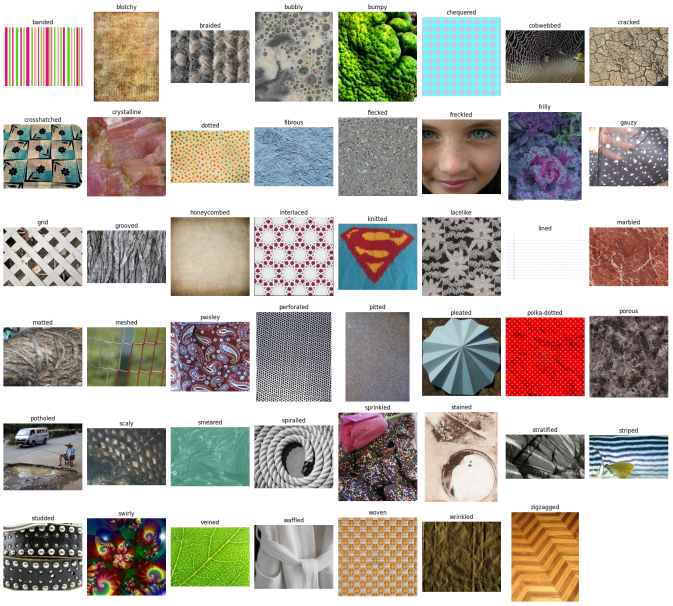

Fig. 2. Example images for a subset of texture classes in the DTD dataset.

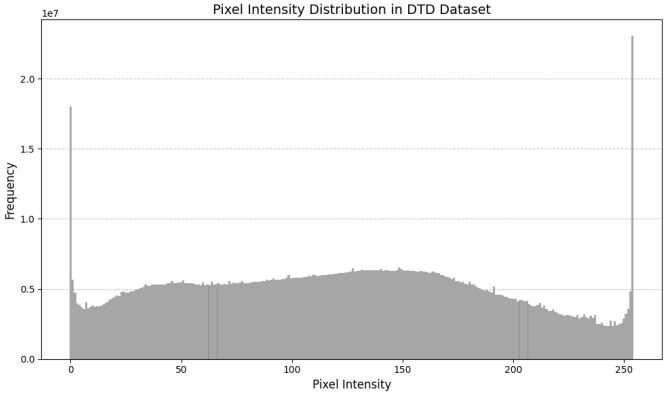

Fig. 3. Pixel Intensity Distribution in DTD Dataset: high frequency of pixels at the extreme ends of the spectrum: pure black (0) and pure white (255)

presented in Figure 2, highlighting the range of visual textures. The overall pixel intensity distribution (Figure 3) exhibits prominent peaks at intensity values 0 and 255, indicating a large presence of very dark and very bright regions in the dataset. Figure 4 summarizes the distribution of image widths, heights, and aspect ratios. Heights vary between 231 and 701 pixels and widths between 300 and 800 pixels. The aspect ratio ranges from 0.49 to 2.13, with mean height and width of 453.35 and 500.01 pixels, respectively. This variability supports the use of scale and crop-based data augmentation during training.

C. SimpleCNN Baseline

1) Architecture: The SimpleCNN model serves as a lightweight baseline for texture classification. Input images are resized to 224×224 pixels and passed through a sequence of convolutional, activation, and pooling layers, followed by fully

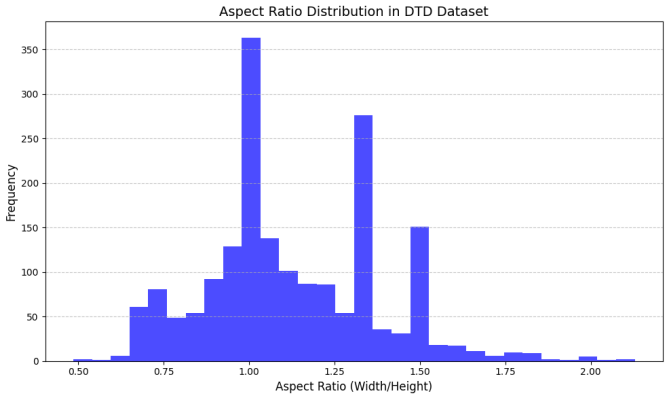

Fig. 4. Aspect ratio and resolution distribution for the DTD dataset (min/max height and width, and aspect-ratio range).

connected layers for classification into 47 texture categories. A dropout rate of 0.5 is applied before the final fully connected layer to reduce overfitting and encourage better generalization. Figure 5 illustrates the overall architecture of the SimpleCNN model.

2) Training Configuration: The SimpleCNN model is trained using the following Hyperparameters:

- **Epochs:** 200
- **Batch size:** 256
- **Optimizer:** Adam
- **Learning rate:** 0.005
- **Weight decay:** 0.0005
- **Loss function:** CrossEntropyLoss

3) Data Augmentation: To improve robustness and mitigate overfitting, the following data augmentation pipeline is applied to the training set:

- Resize to 256×256
- Random crop to 224×224 (scale in [0.7, 1.0])
- Random horizontal flip
- Color jitter with brightness = 0.2, contrast = 0.2, saturation = 0.2, hue = 0.1
- Normalization using standard ImageNet mean and standard deviation

D. ResNet-18 Model

1) Architecture and Residual Learning: ResNet-18 is selected as a second, deeper model due to its efficient residual learning framework. Standard deep neural networks often suffer from vanishing gradients as depth increases, which can slow or prevent convergence. ResNet-18 addresses this issue through residual blocks that learn a function $F(x)$ and add it to the input x, producing the output

$$y = F(x) + x \tag{1}$$

This skip-connection mechanism stabilizes gradient propagation and enables deeper feature extraction without significant performance degradation. ResNet-18 offers a favorable balance

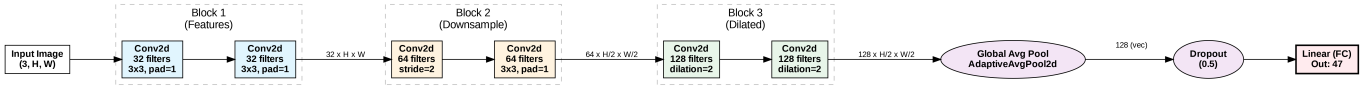

Fig. 5. SimpleCNN architecture used as a baseline model for DTD texture classification.

between model complexity and computational efficiency, making it suitable for training from scratch on the Describable Textures Dataset. Its hierarchical feature-learning capability aligns well with the needs of texture classification, where both local and global visual patterns contribute to class discrimination.

2) Model Building and Hyperparameter Tuning: ResNet-18 is initialized without pretrained weights so that all feature representations are learned directly from the DTD dataset. The original fully connected layer is replaced with a custom classification head consisting of a dropout layer, an activation function, and a linear layer that outputs logits for the 47 texture categories. The model is trained using cross-entropy loss and a batch size of 64.

A structured hyperparameter search is conducted to identify an effective training configuration. The search explores three optimizers (Adam, SGD, AdamW), two learning rates (1×10^{-3} and 3×10^{-4}), two dropout rates (0.0 and 0.3), and four activation functions (ReLU, LeakyReLU, Tanh, Sigmoid). The complete search space is summarized in Table I. Each combination is trained for five epochs, and the configuration achieving the highest validation accuracy is selected for full training. The best-performing setup uses the Adam optimizer with a learning rate of 3×10^{-4}, dropout of 0.0, and the ReLU activation function.

For the final model, training is extended to 50 epochs using the combined training and validation splits, again with a batch size of 64. To improve convergence during extended training, a step-based learning rate scheduler is applied. The learning rate is reduced by a factor of 0.1 every 20 epochs, following

$$\eta_t = \eta_0 \cdot \gamma^{\lfloor t/s \rfloor}, \tag{2}$$

where η_0 is the initial learning rate, γ is the decay factor, and s is the update interval in epochs.

3) Data Augmentation: Several data augmentation techniques are applied to the training images to improve generalization and reduce overfitting. Each image is randomly cropped to a resolution of 224×224 using a resized crop with a scale range of (0.7, 1.0), followed by random horizontal flipping. A color jitter transformation adjusts brightness, contrast, and saturation by 0.3 and hue by 0.1. After augmentation, all images are converted to tensor format.

E. InceptionV3 Model

1) Architecture: InceptionV3 is incorporated as a higher-capacity model featuring multi-branch convolutional blocks designed to capture features at multiple spatial scales. Input images are resized to 224×224 to match the network's requirements. The original classification head is replaced with a custom output layer for 47 texture categories, preceded by a dropout layer to reduce overfitting. Compared to SimpleCNN and ResNet-18, InceptionV3 introduces factorized convolutions and parallel filter paths, increasing representational power but also computational demand.

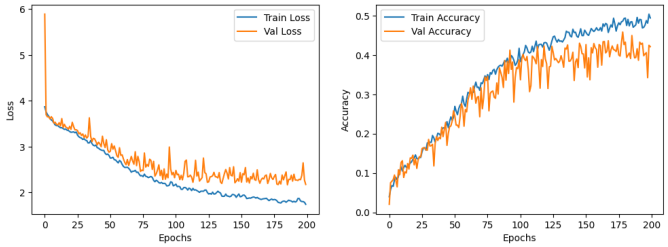

Fig. 6. Training and validation losses (left) and accuracies (right) for Simple CNN

2) Hyperparameter Configuration: The InceptionV3 model is trained using a lightweight optimization setup. Training is performed for 100 epochs with a batch size of 32, using the Adam optimizer and a minimal weight decay of 1×10^{-4} to introduce mild regularization. A step based learning rate scheduler is applied, starting from an initial learning rate of 1×10^{-3} and reducing it by a factor of 0.1 every 30 epochs, eventually reaching values close to 1×10^{-6}. Early stopping with a patience of 10 epochs is used to prevent unnecessary training once validation accuracy plateaus.

TABLE I
HYPERPARAMETER TUNING FOR RESNET-18

Hyperparameter	Values Tested
Optimizer	Adam, SGD, AdamW
Learning rate	$1e^{-3}$, $3e^{-4}$
Dropout	0.0, 0.3
Activation	ReLU, LeakyReLU, Tanh, Sigmoid

F. Evaluation Metrics

Model performance is evaluated using accuracy, precision, recall, and F1-score. Since the dataset is balanced, accuracy provides a reliable overall measure of correct predictions. Precision and recall help reveal how well each class is identified, while the F1-score balances the two for a clearer view of model behavior. Confusion matrices are also used to examine class-wise errors and identify which texture categories are frequently confused, offering useful insight into the strengths and limitations of each model.

III. EXPERIMENTAL RESULTS AND DISCUSSION

A. SimpleCNN Results

The SimpleCNN baseline achieves moderate performance on the DTD test set. While it captures some of the more

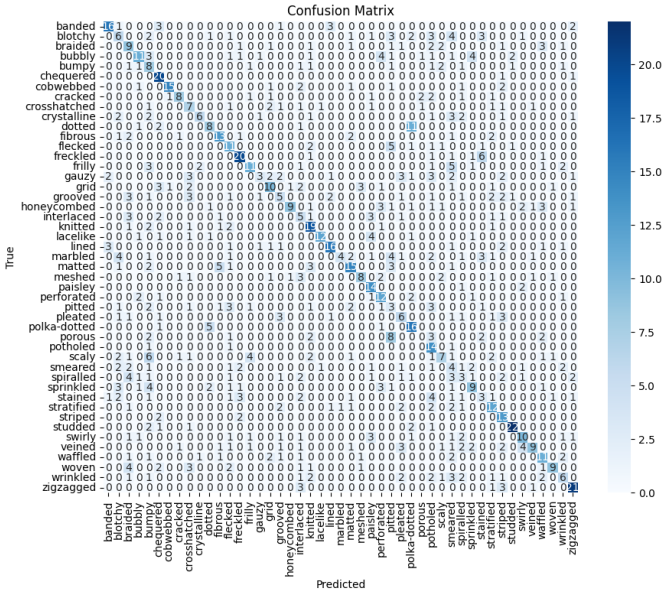

Fig. 7. Confusion matrix for the SimpleCNN baseline on the DTD test set.

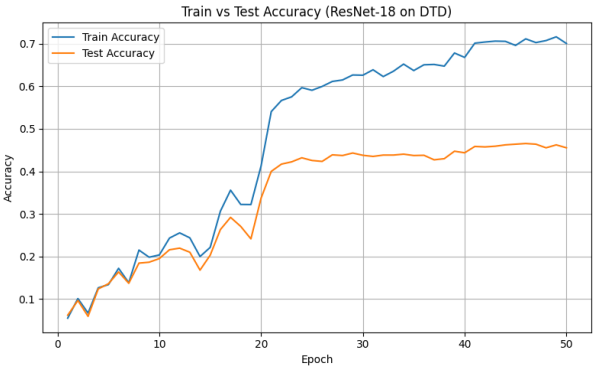

Fig. 8. Train versus test accuracy for ResNet-18 over the training epochs.

distinctive texture categories, its capacity is limited for modeling the full intra-class variability present in the dataset. The training and validation accuracy/loss curves for SimpleCNN are shown in Fig. 6.

As shown in Figure 7, misclassifications are common among visually similar textures. The model struggles in particular with classes whose distinguishing characteristics rely on more subtle, higher-level structures rather than simple local patterns.

B. ResNet-18 Results

ResNet-18 achieves higher test accuracy than SimpleCNN, and InceptionV3 and exhibits better generalization behavior, as illustrated in Figure 8. The gap between training and test accuracy remains moderate, indicating that the model is able to leverage its higher capacity without severe overfitting.

The confusion matrix in Figure 9 shows fewer severe misclassification clusters compared to the SimpleCNN baseline, particularly for classes with strong local texture patterns. However, some confusion persists between visually related

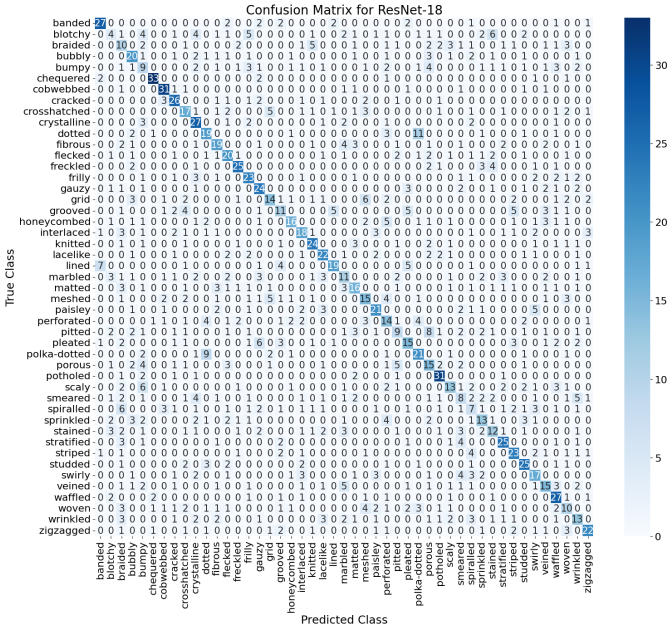

Fig. 9. Confusion matrix for ResNet-18 on the DTD test set.

categories, suggesting that even deep residual networks can find certain fine-grained distinctions challenging.

C. InceptionV3 Results

InceptionV3 underperforms compared to both the SimpleCNN and ResNet-18 baselines on the DTD test set. Despite its substantially higher capacity and architectural sophistication, the model achieves a lower test accuracy and shows weaker generalization behavior.

This performance drop can be attributed to several factors. First, InceptionV3 is optimized for large-scale natural image datasets such as ImageNet, and its deep multi-branch design is prone to overfitting when trained on a relatively small dataset like DTD. Second, the fine-grained texture variations in data do not necessarily benefit from the large receptive fields and complex feature hierarchies that Inception architectures provide. As a result, the model fails to consistently capture the subtle intra class patterns required for robust discrimination.

D. Model Comparison

Table II summarizes the test performance of all three models. ResNet-18 outperforms both the SimpleCNN baseline, and InceptionV3 across all metrics, indicating that deeper residual architectures are better suited for capturing the complex texture patterns present in DTD.

InceptionV3 performs worse than both SimpleCNN and ResNet-18. Despite its higher capacity, the model exhibits lower accuracy and substantially reduced recall and F1-score, suggesting difficulty in consistently recognizing fine-grained texture cues. This result indicates that increasing architectural complexity does not necessarily translate to improved performance on smaller, high-variation datasets such as DTD, and

that compact residual networks may offer a more favorable balance between expressive power and generalization.

TABLE II
COMPARISON OF TEST PERFORMANCE ACROSS MODELS

Model	Accuracy	Precision	Recall	F1-score
SimpleCNN	0.4296	0.4446	0.4296	0.4086
ResNet-18 (Best Model)	0.4553	0.4543	0.4553	0.4510
InceptionV3	0.3972	0.4032	0.3679	0.3710

Although the absolute improvement in accuracy is modest, the gains in precision, recall, and F1-score suggest that ResNet-18 produces more reliable predictions across classes, with fewer extreme failure cases. This aligns with the expectation that residual connections help the model learn richer, more discriminative texture representations.

IV. INTERPRETABILITY

A. Model Interpretability

ResNet-18 model's per-class scores show that it performs well on textures with clear and repetitive patterns, such as *chequered*, *cobwebbed*, *knitted*, and *potholed*, since these are easier for the network's filters to recognize. Classes like *blotchy*, *braided*, and *smeared* have much lower scores because their textures are more irregular and visually similar to other categories, which leads to confusion.

The confusion matrix in Figure 9 shows clusters of mistakes between look-alike textures, suggesting the model relies mainly on local patterns rather than long-range structure. Overall, the results indicate that ResNet-18 learns meaningful texture features, but struggles when different classes share similar visual characteristics. The SimpleCNN baseline exhibits similar, but more pronounced, patterns of confusion, consistent with its lower capacity and less expressive feature hierarchy.

B. Business Insights

1) Automated Defect Detection in Manufacturing: In industrial settings where consistent surface patterns are critical (e.g., textiles, ceramics, precision machining), automated defect detection relies heavily on identifying disruptions in texture. The superior performance of ResNet-18 on highly structured classes suggests that deeper residual networks are well suited for detecting periodic or regular texture anomalies such as stitching errors, scratches, or uniformity defects. However, the models struggle with irregular or visually ambiguous textures, indicating that safety-critical inspection pipelines may require higher-capacity architectures, attention mechanisms, or additional domain-specific fine-tuning to reduce false alarms and missed defects.

2) Content Tagging and Visual Search in Media Platforms: Texture classification also plays an important role in media and e-commerce applications, enabling fine-grained tagging and improved visual search. More stable performance from ResNet-18 across diverse texture categories can support automated metadata generation for large image repositories, enhancing user search relevance and personalization. Remaining confusion between visually similar classes suggests that integrating texture-based features with semantic labels, user interaction signals, or multimodal embeddings can further improve retrieval quality.

V. CONCLUSION

This work presented a comparative study of three convolutional architectures a SimpleCNN baseline, ResNet-18, and InceptionV3 trained from scratch on the Describable Textures Dataset (DTD). After characterizing the dataset through descriptive statistics and exploratory analysis, we detailed the architectural choices and training configurations for each model, including a structured hyperparameter search for ResNet-18.

Experimental results showed that ResNet-18 consistently outperforms both SimpleCNN and InceptionV3 across accuracy, precision, recall, and F1-score. While SimpleCNN provides a reasonable baseline, its limited capacity restricts its ability to capture the high intra-class variability present in DTD. InceptionV3, despite its significantly higher complexity, underperforms relative to both models, suggesting that large-scale architectures optimized for natural image classification do not directly translate to improved texture recognition under limited data conditions.

Confusion matrix analysis further revealed that all models perform best on categories with distinctive repetitive patterns, while struggling with irregular or visually similar textures. These findings highlight the challenges of fine-grained texture classification and indicate that increased architectural depth alone is insufficient for robust generalization.

SUPPLEMENTARY MATERIAL

The complete code implementation of this project is available at: GitHub Link.

The accompanying blog post provides a walkthrough of the project: Medium Link

ACKNOWLEDGMENT

The authors would like to thank Dr. Peeples for guidance throughout the course, as well as the course staff and peers for valuable feedback and discussions.

REFERENCES

[1] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, and A. Vedaldi, "Describing Textures in the Wild," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 3606–3613. Available: https://www.robots.ox.ac.uk/~vgg/data/dtd/

[2] P. Brodatz, *Textures: A Photographic Album for Artists and Designers*. New York: Dover Publications, 1966.

[3] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, 2002.

[4] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.