Computers, Materials & Continua

Tech Science Press

https://doi.org/10.32604/cmc.2026.085198

ARTICLE

DeepMarbleVision: A Texture-Aware Ensemble Deep Learning Model with Energy-Layer-Based Feature Fusion for Marble Classification

Yunis Torun[1,*], **Burak Seckin**[1] **and Rukiye Karakis**[2]

[1]Department of Electrical and Electronic Engineering, Faculty of Engineering, Sivas Cumhuriyet University, Sivas, Türkiye
[2]Department of Software Engineering, Faculty of Engineering, Sivas Cumhuriyet University, Sivas, Türkiye

*Corresponding Author: Yunis Torun. Email: ytorun@cumhuriyet.edu.tr

Received: 07 May 2026; Accepted: 29 July 2026; Published: 15 September 2026

ABSTRACT: Marble classification has traditionally relied on human visual inspection, where operators assess color, texture, and pattern alignment to determine quality. However, this manual process is subjective, inconsistent, and inefficient for large-scale industrial applications. To address these limitations, this study proposes DeepMarbleVision, a texture-aware ensemble deep learning framework with energy-layer-based feature fusion for marble quality classification. A real-world dataset was created using the MarbleVision system, including three marble quality classes acquired from an industrial marble classification environment. The proposed approach integrates energy-layer-based feature fusion into TCNN variants of AlexNet, ResNet, and DenseNet, which were initialized through texture-oriented pre-training and fine-tuned for marble quality classification. To further improve classification robustness, an ensemble learning strategy was applied by averaging the class-probability outputs of individual CNN and TCNN models. The ensemble model combining baseline CNN and energy-enhanced TCNN architectures achieved 99.5% classification accuracy, outperforming the evaluated standalone TCNN models: AlexNet-TCNN, 89.25%; DenseNet-TCNN, 95.16%; and ResNet-TCNN, 92.83%. These findings indicate that energy-layer-enhanced ensemble deep learning models can improve texture-based marble quality classification compared with the evaluated standalone CNN and TCNN models. The proposed model is intended for future integration into the MarbleVision automated marble classification pipeline and provides an adaptable framework for high-precision aesthetic surface inspection in related industrial applications.

KEYWORDS: Deep learning; ensemble learning; energy-layer feature fusion; marble classification; texture-based convolutional neural network (TCNN); transfer learning

1 Introduction

In modern manufacturing systems, image processing has become a pivotal technology for ensuring quality, accuracy, and efficiency, particularly when integrated with intelligent systems and machine learning techniques to support automated inspection and fault detection. In industries where aesthetic quality impacts market value, such as natural stone, ceramics, parquet, and fabric production, classification is traditionally performed through human visual inspection. Among these materials, marble slabs are widely used for large-surface decoration, where color homogeneity, texture type, durability, and surface finish determine quality. However, due to the natural uniqueness of marble, no two slabs are identical, making consistent classification a challenge. To maintain aesthetic integrity, slabs placed side by side must belong to the same homogeneous category [1].

Marble classification is typically performed at the final stage of production, where experts assess texture, pattern consistency, and defects before packaging. However, manual classification suffers from subjectivity, as criteria vary among experts. Additionally, visual fatigue, lighting variations, and inconsistencies lead to misclassification, highlighting the need for automated classification systems that provide objective and reproducible evaluation criteria [2–4].

Early marble classification relied on statistical and histogram-based feature extraction, using mean, variance, skewness, entropy, and correlation attributes. Turan et al. [2] applied Local Binary Patterns (LBP), Scale-Invariant Feature Transform (SIFT), and histogram-based methods, classifying four marble types using machine learning (ML) models such as Extreme Learning Machine (ELM), Decision Trees (DT), Support Vector Machine (SVM), and Multilayer Perceptron (MLP). Similarly, Martinez-Alajarin et al. [1] classified three marble categories (extra, commercial, and low) using Principal Component Analysis (PCA) and MLP. More advanced ML approaches have been explored to improve performance. Selver et al. [5] proposed an electromechanical sorting system with a Hierarchical Radial Basis Function Network (HRBFN), while Doğan and Akay [6] used an Adaboost-based clustering algorithm. López et al. [7] classified 16 granite types using spectrophotometer-measured spectral data instead of images. In another study, MLP models achieved 92.0% and 83.6% accuracy for brightness-based marble classification [8].

Traditional ML-based classification faces critical limitations due to handcrafted feature selection, which restricts scalability and fails to capture complex texture patterns [9]. Deep learning (DL) models, particularly convolutional neural networks (CNNs), overcome these limitations by automatically extracting hierarchical features through convolutional layers, activation functions, and pooling operations [10]. With the integration of deep learning and optimized convolutional neural networks, image processing systems have demonstrated marked improvements in precision, speed, and reliability across a variety of industrial applications. Among DL-based methods, transfer learning (TL) with pre-trained models such as AlexNet, VGG16, ResNet, and LeNet has been widely used. Torun et al. [11] utilized AlexNet and LBP-based features with SVM, achieving high performance. Canayaz and Uludag [12] classified 28 marble types using VGG16, ResNet, and LeNet, while Öktem et al. [13] tested 12 TL models, reporting VGG16 as the most effective. Additionally, Sidiropoulos et al. [14] showed that combining handcrafted texture descriptors with CNN-based feature extraction improves accuracy. These findings confirm that leveraging pre-trained ImageNet models enhances classification performance [12–15]. Beyond TL-based architectures, shallow CNN models have also been explored [16–18]. Karaali and Eminagaoglu [16] applied blurred and filtered augmentations, outperforming traditional ML methods. Tiwari et al. [17] optimized CNN weights and biases using Root Mean Square Propagation (RMSProp), achieving 81% accuracy. Pence and Cesmeli [18] classified marble slabs into two quality grades, obtaining 75% accuracy with the Adam optimizer.

Marble slab classification is inherently a texture analysis problem, as texture represents local structural patterns in an image [2]. Attributes such as roughness, irregularity, and homogeneity are commonly used in object recognition, segmentation, and industrial quality control [19,20]. DL-based texture classification methods can be divided into TL-based architectures and custom models designed specifically for texture analysis. Simon and Uma [20] classified high-level texture features from DenseNet201, ResNet50, ResNet101, and InceptionV3 using SVM, achieving high accuracy. Similarly, Goyal and Sharma [21] used MobileNetV3 and InceptionV3, demonstrating strong performance in multi-texture classification. Yavuz and Türkoğlu [22] performed marble quality classification using nine pre-trained architectures, reporting that ResNet50 with a Medium Gaussian SVM achieved a benchmark accuracy of 95.80%. In their subsequent study [23], the authors integrated Generative Adversarial Network (GAN) based super-resolution as a preprocessing stage to reconstruct low-quality textures, which improved the classification accuracy to 96.4% using the same pipeline. Among these approaches, Texture CNN (TCNN) has emerged as a specialized model for texture feature extraction rather than generic deep features [24]. Andrearczyk and Whelan [19] introduced TCNN, inspired by AlexNet, which distinguishes between low-level edge/shape features, mid-level texture features, and high-level object features. Unlike traditional CNNs, TCNN integrates an energy layer, selectively enhancing texture-specific features for improved classification performance. These studies highlight that DL models significantly enhance texture classification, but their effectiveness depends on training data size. TCNN architectures, particularly those integrating energy layers, provide an advantage in texture classification by emphasizing relevant structural patterns.

Literature studies indicate that manual marble classification remains inefficient, leading to time and financial losses due to visual fatigue, lighting variations, and subjective judgment. While traditional ML approaches have been explored to address these limitations, their dependence on handcrafted feature selection limits scalability and effectiveness in complex texture analysis. DL models, particularly CNN-based architectures, have demonstrated significant improvements in texture classification by automatically extracting discriminative features.

Building on these advancements, this study introduces a texture-aware deep learning framework for automated marble quality classification. Given that Türkiye holds a substantial share of the world's marble reserves [25] and is a major global producer, objective and reproducible marble quality assessment is important for reducing classification errors and supporting industrial automation. Unlike conventional handcrafted approaches that may rely on explicitly designed geometric or statistical descriptors, the proposed approach adopts a texture-oriented deep learning strategy that emphasizes surface appearance, texture continuity, and pattern variations in marble slabs. Although CNN-based models can automatically extract discriminative visual representations, standalone architectures may still suffer from model-specific bias and class ambiguity, particularly when visually similar marble quality grades are considered. To address this issue, this study develops TCNN-enhanced CNN architectures with energy-layer-based feature fusion and combines them through an ensemble learning strategy. The proposed framework is evaluated on a real-world MarbleVision dataset consisting of three marble quality classes acquired in an industrial marble classification environment. MarbleVision refers to the previously developed industrial image acquisition and marble classification platform [26]. In the present study, we do not introduce a new hardware acquisition system. Instead, DeepMarbleVision refers to the proposed DL-based classification framework developed and evaluated using marble images collected by the existing MarbleVision platform. The main contributions of this study are as follows:

- A texture-aware ensemble deep learning framework that integrates energy-layer-based feature fusion for marble quality classification.
- The energy-layer concept originally introduced by Andrearczyk and Whelan [19] is adapted to transfer learning backbones with different feature propagation mechanisms, including residual and dense connectivity-based architectures, and its contribution is systematically evaluated through mid-level and high-level feature fusion for marble texture analysis.
- A texture-oriented deep learning strategy that emphasizes surface appearance and pattern-based representations without relying on handcrafted geometric descriptors.
- A comparative evaluation of baseline CNN models and TCNN-enhanced models, demonstrating the contribution of energy-layer-based feature fusion to texture-sensitive classification performance.
- An industrially motivated framework designed for future integration into the MarbleVision marble classification pipeline and adaptable to other aesthetic surface inspection tasks.

2 Material and Method

2.1 Obtaining Real-World Dataset

The dataset used in this study was obtained from the MarbleVision Marble Classification Automation System, which was developed by Kapsam Electromechanical Industry Co., Ltd., and deployed in an industrial marble classification environment. The system was funded under the TUBITAK-1507 SME R&D Support Program by Kapsam Electromechanical Industry Co., Ltd., [26]. Fig. 1 illustrates the automation system. Marble slab images were acquired using the MarbleVision device and originally had a resolution of 1032 × 532 × 3. For compatibility with DL models, the images were resized to 1024 × 512 × 3. Expert professionals manually labeled the dataset into three quality classes, denoted as C1, C2, and C3, representing first, second, and third-class marble, respectively. To enhance labeling consistency, a ML-based verification step was implemented. The approach, proposed by Seckin et al. [15], used LBP features classified via a voting ensemble classifier (VEC). The probability values for each class were computed and compared with the initial expert labels. In cases of discrepancies, marble slabs were re-examined by experts, and any labeling errors were corrected [15]. As shown in Fig. 2, the dataset includes 1051 marble slab images in total, consisting of 320 C1, 320 C2, and 411 C3 samples. To evaluate model robustness, the dataset was divided into five repeated random subsampling validation (RSV) subsets. RSV was preferred instead of relying on a single random split, allowing the models to be evaluated under different randomly generated train-test partitions. In each RSV repetition, a class-balanced test subset was constructed by randomly selecting 80 samples from each class, resulting in 240 test images in total. The remaining 811 images were used for training, corresponding to an approximately 77/23 train-test split. The same RSV splits were used for all baseline CNN, TCNN-enhanced, and ensemble models to ensure a consistent comparison.

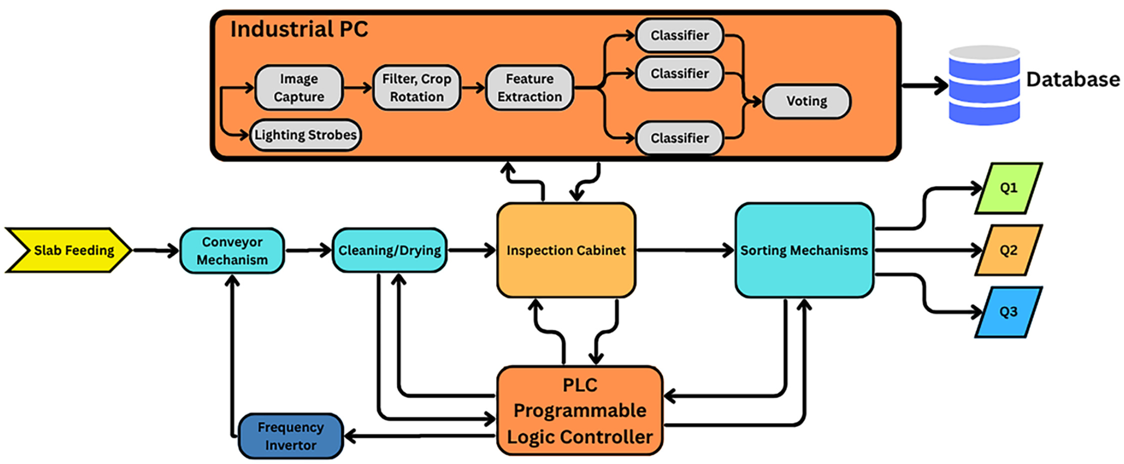

Figure 1: MarbleVision© marble classification system [26].

In this study, feature extraction layers were taken from AlexNet, ResNet50, and DenseNet121, as these models have demonstrated high performance in ImageNet-based texture classification [12–15]. To address the large data requirements of these deep models, a pre-training phase was conducted using a texture dataset extracted from ImageNet, including "stone-wall", "tile-roof", and "velvet" texture classes [19]. The dataset was resized to 1024 × 512 × 3 to ensure consistency with the marble classification task. The three models were trained and evaluated on the texture pretraining dataset extracted from ImageNet using five RSV splits. The trained weights obtained from this three-class texture classification task were then used to initialize the models developed for marble quality classification. This intermediate texture-oriented pretraining step was used to adapt the feature extraction layers toward texture-sensitive representations before fine-tuning on the marble quality dataset. The final marble classification experiments were evaluated separately using five RSV subsets generated from the MarbleVision dataset.

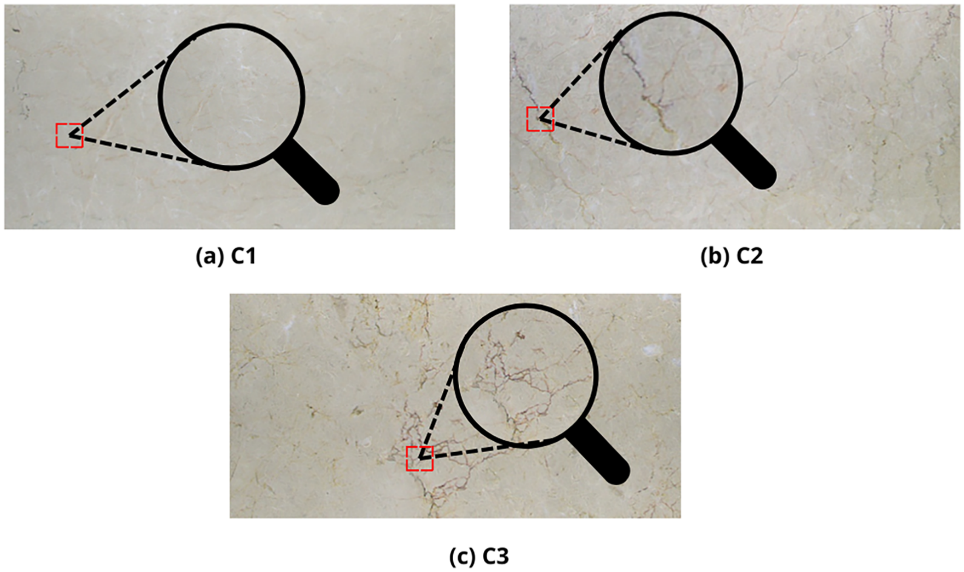

Figure 2: Sample images of the marble dataset obtained from the MarbleVision device: (**a**) C1: first class, (**b**) C2: second class, (**c**) C3: third class.

2.2 Proposed Ensemble-Based Deep Learning Model

In this study, we utilized TL-based models (AlexNet, ResNet50, and DenseNet121) along with their TCNN-enhanced counterparts (AlexNet-TCNN, ResNet50-TCNN, and DenseNet121-TCNN) to classify marble slab quality, which is primarily a texture analysis problem [19]. These models extract mid-level texture features from intermediate layers and combine them with high-level semantic features to enhance classification performance.

As shown in Fig. 3, the proposed approach begins with training six CNN models on a three-class texture dataset from ImageNet to improve feature extraction capabilities [19]. After this pre-training phase, the learned model weights were transferred to initialize the CNN models for marble classification. To further enhance classification robustness, four predefined ensemble configurations were evaluated using the same five RSV splits. For each ensemble, the class-probability outputs of the constituent CNN and TCNN models were averaged with equal weights, and the final class label was assigned according to the maximum averaged probability. The ensemble performances were then reported as the mean values across the five test subsets.

To improve texture representation, a texture energy layer was integrated into the baseline CNN architectures. In the proposed TCNN variants, the energy layer acts as a texture-oriented feature fusion module that combines intermediate feature maps, which retain local texture and pattern information, with deeper feature maps, which encode more abstract and semantic representations. In this implementation, the energy representation is formed by concatenating the selected mid-level and high-level feature maps, followed by global average pooling. Therefore, the energy layer does not operate as an independent classifier; rather, it provides a fused texture representation that enriches the final feature vector before the fully connected classification layers. The model parameters were either initialized from the texture-oriented pre-training stage or learned during fine-tuning on the marble quality classification dataset. To further evaluate the developed TCNN-enhanced models, two additional benchmark datasets were used: the 28-class marble dataset reported by Canayaz and Uludag [12] and the Kylberg texture dataset [27]. The reported results were obtained in this study by re-training and testing all baseline CNN and TCNN-enhanced models using five RSV subsets, rather than being taken directly from the literature.

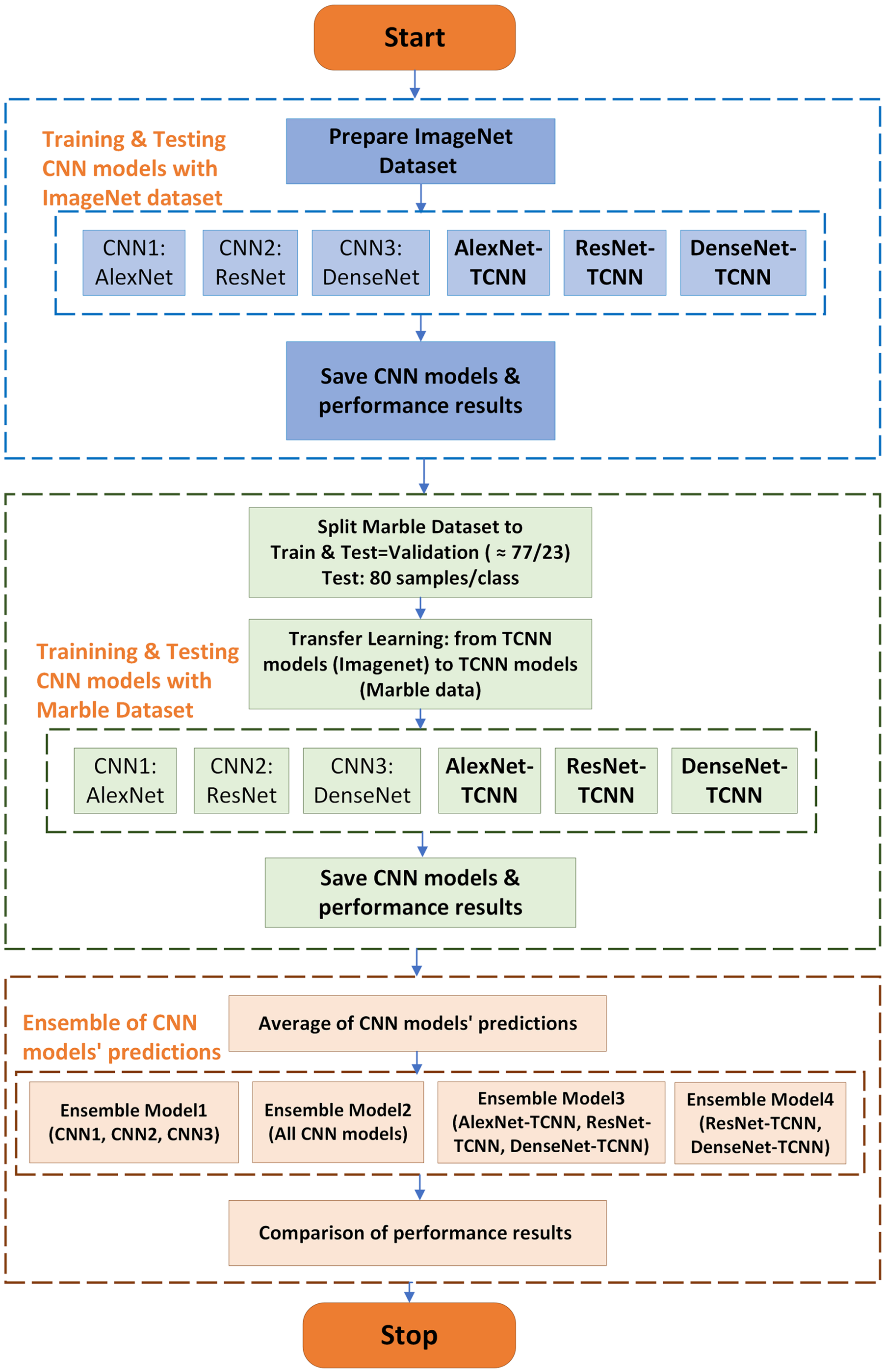

Figure 3: Flow chart of the proposed research.

2.2.1 Texture-Based AlexNet (AlexNet-TCNN) Model

The AlexNet architecture, originally developed for 1000-class ImageNet classification [28] employs ReLU activation to mitigate gradient loss and dropout layers to prevent overfitting. Feature extraction is performed through sequential convolutional layers, where low-level features are captured in the first two layers with 11 × 11 and 5 × 5 kernel sizes, followed by batch normalization and max-pooling.

The proposed AlexNet-TCNN model extends AlexNet's feature extraction layers from 5 to 7 to better adapt to 1024 × 512 input dimensions, allowing for enhanced texture representation. As shown in Fig. 4, the first four convolutional layers apply Convolution (Conv)+Batch Normalization (BN)+ReLU operations to extract low- and mid-level features, while the last three convolutional layers use Conv+ReLU, followed by max-pooling with a 3 × 3 filter kernel and a stride of 2.

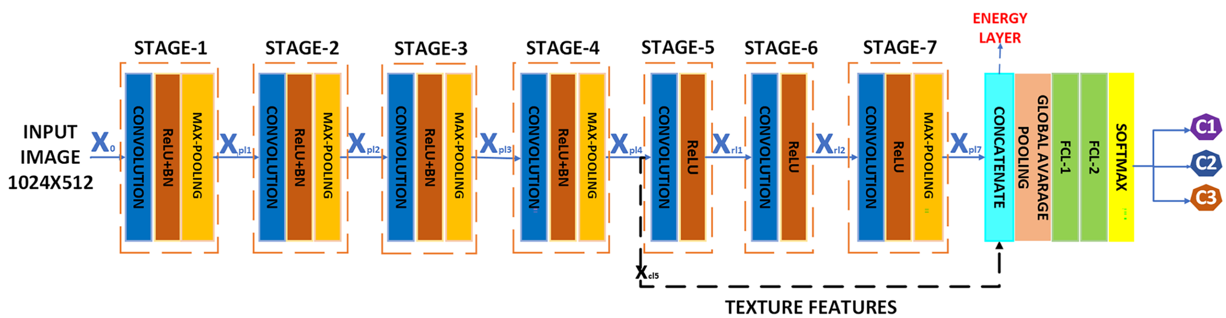

Figure 4: Texture-based AlexNet (AlexNet-TCNN) model.

To capture texture-specific information, mid-level texture features extracted from the 5th convolutional layer (x_{cl}) are combined with high-level feature maps from the final convolutional layer after max-pooling (x_{pl}) in an energy layer, inspired by the TCNN model proposed by Andrearczyk and Whelan [19].

$$Lx_{energy} = \text{Concatenate}(x_{cl}, x_{pl}) \tag{1}$$

here, x_{cl} represents mid-level texture descriptors, while x_{pl} contains high-level semantic features. The output of the energy layer is converted into a compact feature vector using global average pooling (GAP). These extracted feature vectors are classified into three quality classes via two fully connected layers (4096 neurons each), followed by a softmax layer.

Unlike the standard AlexNet model, which only utilizes high-level features for classification, the proposed AlexNet-TCNN architecture (Fig. 4) integrates both texture and semantic information, improving the model's ability to distinguish marble slab quality based on texture characteristics rather than relying solely on object-level representations.

2.2.2 Texture-Based ResNet (ResNet-TCNN) Model

In CNN-based TL models, deeper architectures generally improve performance, but they also introduce the vanishing gradient problem. ResNet [29] and DenseNet [30] are commonly used TL architectures that support more efficient training in deep networks by improving gradient flow through different mechanisms. ResNet employs residual blocks, where identity mappings help preserve gradient flow across layers. In a deep network with L layers, each sub-layer indexed by ℓ applies a nonlinear transformation $H_\ell(\cdot)$, including operations such as convolution, batch normalization (BN), ReLU, and pooling [29–30]. The standard ResNet formulation is given as:

$$\mathrm{l}x_{\ell} = H_{\ell}\left(x_{\ell-1}\right) + x_{\ell-1} \tag{2}$$

where $x_{\ell-1}$ represents the input feature map, and the residual connection allows direct feature propagation across layers.

As shown in Fig. 5, the proposed ResNet50-TCNN model extends ResNet50 by integrating an energy layer to enhance texture feature representation. The ResNet50 architecture, consisting of five stages, applies Conv(1 × 1)+BN+ReLU, Conv(3 × 3)+BN+ReLU, and Conv(1 × 1)+BN operations in convolution blocks. Skip connections in these blocks apply a Conv(1 × 1)+BN operation, while identity blocks directly combine input and output feature maps.

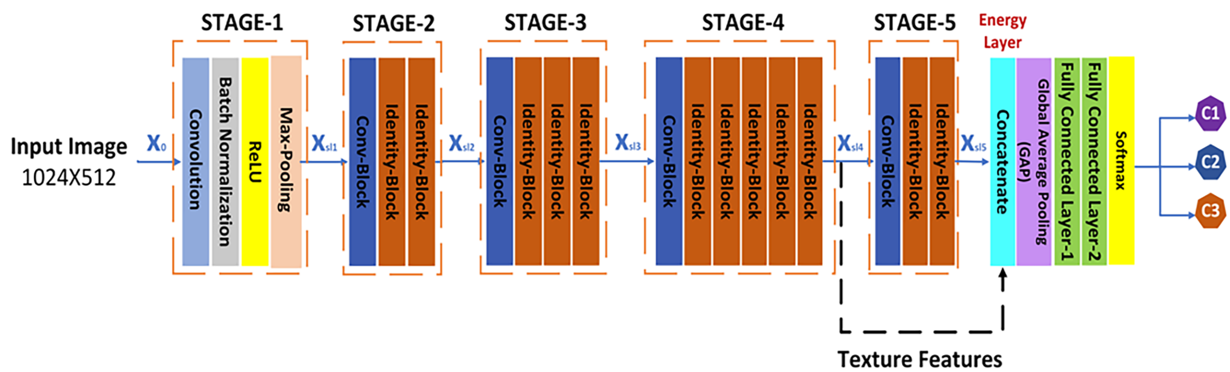

Figure 5: Texture-based ResNet50 (ResNet-TCNN) model.

To capture both mid-level and high-level texture features, the energy layer combines features extracted from the 4th stage (x_{sl-1}) with those from the final layer (x_{sl}), as formulated in Eq. (3):

$$x_{energy} = \mathrm{Concatenate}\left(x_{sl-1}, x_{sl}\right) \tag{3}$$

Experimental evaluations demonstrated that combining the fourth-stage feature maps with the final-layer representations yielded the best results for marble quality classification. This combination allows the model to preserve mid-level texture cues while incorporating deeper semantic information through residual feature propagation.

2.2.3 Texture-Based DenseNet (DenseNet-TCNN) Model

The DenseNet architecture employs a dense connectivity pattern, where each layer in the feature extraction process is directly connected to all subsequent layers in a feed-forward manner. These dense connections mitigate the vanishing gradient problem, facilitate more efficient gradient backpropagation, and enhance feature reuse, resulting in improved learning efficiency [30]. In DenseNet, the output of the ℓ − th layer is computed by concatenating the feature maps from all preceding layers, as formulated in Eq. (4):

$$x_{\ell} = H_{\ell}\left(\left[x_0, x_1, \ldots, x_{\ell-1}\right]\right) \tag{4}$$

In this study, both the DenseNet121 model [30] and the proposed DenseNet121-TCNN model were utilized for marble quality classification, as illustrated in Fig. 6. The DenseNet121 architecture consists of four dense blocks interleaved with three transition blocks.

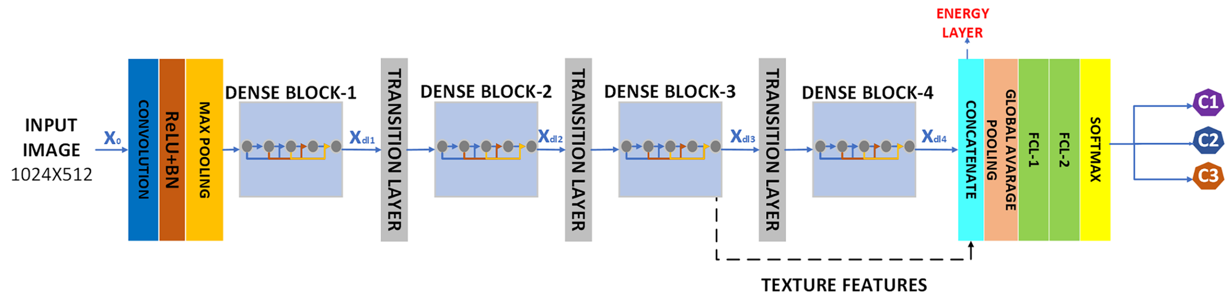

Figure 6: Texture-based DenseNet121 (DenseNet-TCNN) model.

To enhance texture representation, an energy layer was incorporated into the DenseNet121-TCNN model, which integrates mid-level and high-level features. In this structure, the energy layer combines feature maps extracted from the third dense block (x_{dl-1}) with those obtained from the final layer (x_{dl}), following Eq. (5):

$$x_{energy} = \text{Concatenate}(x_{dl-1}, x_{dl}) \tag{5}$$

Experimental evaluations demonstrated that the optimal feature combination for marble quality classification was achieved by merging features from the third dense block with those from the final layer. This integration allows the model to retain detailed texture information from the third dense block while incorporating abstract high-level representations from the final layer, thereby improving classification performance.

2.3 Performance Metrics

In this study, the performance of the CNN, TCNN, and ensemble models was evaluated using accuracy, error rate, sensitivity, specificity, precision, F1-score, and the Matthews correlation coefficient (MCC). Since the marble quality classification task consists of three classes, class-wise true positive (TP), true negative (TN), false positive (FP), and false negative (FN) values were computed using a one-vs-rest strategy for each class. Sensitivity, specificity, precision, and F1-score, and MCC were then macro-averaged across the three classes to obtain the reported multiclass performance values. Accuracy is calculated as the ratio of correctly predicted samples to the total number of samples, as shown in Eq. (6).

$$Accuracy = \frac{TN + TP}{(TN + TP + FP + FN)} \tag{6}$$

where, *TN*, *TP*, *FN*, and *FP* represent true negative, true positive, false negative, and false positive, respectively. Specificity measures the model's success in predicting truly negative samples, as defined in Eq. (7).

$$Specificity = \frac{TN}{(TN + FP)} \tag{7}$$

Sensitivity, also known as recall, indicates the proportion of actual positive samples correctly predicted (Eq. (8)). Precision measures the proportion of true positives among the samples predicted as positive (Eq. (9)).

$$Sensitivity = \frac{TP}{(TP + FN)} \tag{8}$$

$$Precision = \frac{TP}{(TP + FP)} \tag{9}$$

The F1-score is used to balance precision and recall and is calculated using Eq. (10).

$$F1 - score = 2 \times \frac{(Sensitivity * Precision)}{(Sensitivity + Precision)} \tag{10}$$

The MCC was additionally used to measure the agreement between the predicted and true class labels. Since the marble quality classification task consists of three classes, MCC was computed for each class using a one-vs-rest strategy based on TP, TN, FP, and FN values, and the class-wise MCC values were averaged to obtain the reported multiclass MCC score. The MCC formula is given in Eq. (11). Higher MCC values indicate stronger agreement between the predicted and actual labels.

$$MCC = \frac{TPxTN - FPxFN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{11}$$

2.4 Implementation Details

The proposed CNN and TCNN models were implemented in Python using the Keras and TensorFlow deep learning libraries. All experiments were conducted on a workstation equipped with an NVIDIA RTX A6000 GPU with 48 GB memory, an Intel i9-12900KS CPU operating at 3.40 GHz, and 64 GB RAM. All input images were resized to 1024 × 512 × 3 and normalized before being fed into the models. The categorical cross-entropy loss function was used for the three-class classification task. The Adam optimizer was employed to update the model parameters with an initial learning rate of 0.0001, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\varepsilon = 1 \times 10^{-7}$. To mitigate overfitting, dropout regularization and L2 weight regularization were applied. A dropout rate of 0.5 was applied after each of the two fully connected layers in the classification head, and L2 regularization with a coefficient of 0.00001 was applied to the kernel and bias terms of the dense layers. The models were trained for 100 epochs with a batch size of 32. Training was also controlled by reducing the learning rate when the validation performance stopped improving (ReduceLROnPlateau) and by retaining the best model weights based on validation accuracy (ModelCheckpoint). The same training configuration and RSV splits were used for all baseline CNN, TCNN-enhanced, and ensemble models to ensure a consistent comparison.

3 Experimental Results and Discussion

3.1 Results of the Deep Learning Models

Table 1 presents the performance metrics of AlexNet, DenseNet121, and ResNet50, which are commonly used TL models for marble quality classification. These models were first pretrained on a three-class ImageNet texture subset (including stone-wall, tile-roof, and velvet classes) and then fine-tuned for the marble classification task. Since each RSV test subset was class-balanced with 80 samples per class, the macro-averaged sensitivity is mathematically equivalent to overall accuracy in this evaluation setting. DenseNet121 achieved the highest performance among these architectures, while AlexNet exhibited the lowest results. The superior performance of DenseNet121 can be attributed to its dense connectivity pattern, which facilitates gradient flow, enhances feature propagation, and promotes feature reuse. Like residual connections in ResNet50, these dense connections help mitigate information degradation across layers, leading to more effective texture feature extraction. As observed in Table 1, both ResNet50 and DenseNet121 outperformed AlexNet, indicating that architectures incorporating dense or residual connections are more effective in marble texture classification. This improvement can be attributed to their ability to retain relevant features across deeper layers, thereby reducing the vanishing gradient issue and improving feature discrimination among marble classes.

Table 1: Performance results of transfer learning models using five repeated random subsampling validation subsets.

Performance Metric	AlexNet	DenseNet121	ResNet50
Accuracy	0.8858 [0.8667–0.9125]*	0.9342 [0.9167–0.9542]	0.8925 [0.8500–0.9167]
Error	0.1142 [0.0875–0.1333]	0.0658 [0.0458–0.0833]	0.1075 [0.0833–0.1500]
Sensitivity	0.8858 [0.8667–0.9125]	0.9342 [0.9167–0.9542]	0.8925 [0.8500–0.9167]
Specificity	0.9429 [0.9333–0.9562]	0.9671 [0.9583–0.9771]	0.9462 [0.9250–0.9583]
Precision	0.8899 [0.8674–0.9149]	0.9347 [0.9173–0.9547]	0.9058 [0.8764–0.9224]
F1-score	0.8860 [0.8667–0.9130]	0.9342 [0.9169–0.9541]	0.8934 [0.8525–0.9161]
MCC	0.8308 [0.8003–0.8699]	0.9015 [0.8753–0.9316]	0.8458 [0.7888–0.8783]

Note: *Mean [Minimum–Maximum].

Following the evaluation of the baseline TL models, an ablation analysis was conducted to determine the most effective energy-layer location for each TCNN-enhanced architecture. As shown in Fig. 7, different intermediate feature locations were evaluated while keeping the remaining training settings unchanged. For AlexNet-TCNN, Stage-3, Stage-4, and Stage-5 were compared, and Stage-4 provided the best overall performance. For DenseNet-TCNN, Dense Block-2, Transition Layer-2, and Dense Block-3 were evaluated, with Dense Block-3 achieving the highest results. For ResNet-TCNN, Stage-3 and Stage-4 were compared, and Stage-4 produced the best performance. Based on this analysis, these locations were selected as the energy-feature extraction points in the final TCNN configurations.

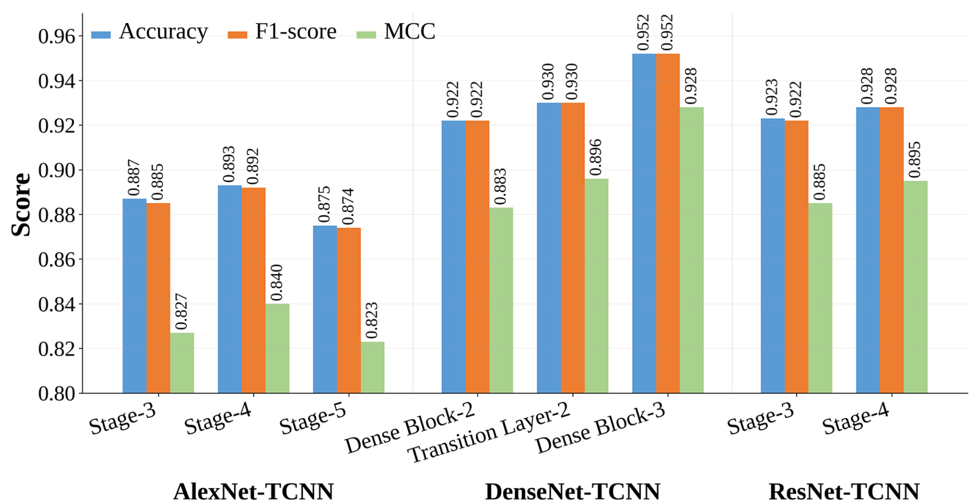

Figure 7: Ablation analysis of different energy-layer locations for AlexNet-TCNN, DenseNet-TCNN, and ResNet-TCNN.

Table 2 presents the performance metrics of the final AlexNet-TCNN, DenseNet121-TCNN, and ResNet50-TCNN configurations selected based on the energy-layer location analysis in Fig. 7. As observed in Tables 1 and 2, the TCNN-enhanced models consistently outperformed their baseline CNN counterparts, demonstrating the effectiveness of integrating mid-level and high-level texture features through an energy layer. Among the TCNN models, DenseNet121-TCNN achieved the highest performance, benefiting from dense connectivity, which facilitates feature reuse, gradient stability, and improved feature propagation across layers. These advantages contribute to higher-quality feature representations in the energy layer, leading to superior classification results. ResNet50-TCNN also performed well, as its residual connections help prevent information loss and preserve texture details, resulting in richer feature representations within the energy layer. In contrast, AlexNet-TCNN, while benefiting from the energy layer, lacks dense or residual connections, leading to comparatively lower performance. The limited depth and feature propagation capabilities of AlexNet may restrict its ability to effectively leverage the texture feature fusion process, impacting its overall classification accuracy.

Table 2: The performance results of TCNN models using five RSV splits.

Performance Metric	AlexNet-TCNN	DenseNet-TCNN	ResNet-TCNN
Accuracy	0.8925 [0.8708–0.9292]*	0.9516 [0.9333–0.9708]	0.9283 [0.9167–0.9458]
Error	0.1075 [0.0708–0.1292]	0.0484 [0.0292–0.0667]	0.0717 [0.0542–0.0833]
Sensitivity	0.8925 [0.8708–0.9292]	0.9516 [0.9333–0.9708]	0.9283 [0.9167–0.9458]
Specificity	0.9462 [0.9354–0.9646]	0.9759 [0.9667–0.9854]	0.9645 [0.9583–0.9729]
Precision	0.8945 [0.8780–0.9318]	0.9526 [0.9361–0.9709]	0.9317 [0.9214–0.9467]
F1-score	0.8920 [0.8695–0.9297]	0.9515 [0.9330–0.9708]	0.9283 [0.9170–0.9457]
MCC	0.8400 [0.8103–0.8950]	0.9280 [0.9016–0.9563]	0.8945 [0.8775–0.9193]

Note: *Mean [Minimum–Maximum].

Table 3 presents the ablation comparison between the plain TL models and the proposed energy-layer-based TCNN models. In the plain TL setting, AlexNet, DenseNet121, and ResNet50 were initialized with weights pretrained on the 1000-class ImageNet-1K dataset and then fine-tuned for the three-class marble classification task. To ensure a fair comparison, the plain TL models were trained using the same experimental protocol used for the TCNN models, including the same input preprocessing, normalization, train-test splits, optimizer settings, and classification head. As shown in Table 3, the energy-layer-based TCNN models achieved higher performance than the plain ImageNet-1K TL models across all three architectures. The improvements were observed in Accuracy, F1-score, and MCC, indicating a consistent gain in classification performance. In particular, the increase in MCC suggests that the proposed energy-layer-based feature fusion improves the consistency of multiclass predictions, not only the overall accuracy. These findings support the contribution of integrating mid-level texture features with high-level representations for marble quality classification.

Table 3: Ablation comparison between plain ImageNet-1K TL models and energy-layer-based TCNN models.

Model	Accuracy	F1-score	MCC
AlexNet-TCNN	0.8925 [0.8708–0.9292]*	0.8920 [0.8695–0.9297]	0.8400 [0.8103–0.8950]
AlexNet+ImageNet-1K**	0.8725 [0.8542–0.8875]	0.8716 [0.8531–0.8870]	0.8097 [0.7814–0.8316]
DenseNet-TCNN	0.9516 [0.9333–0.9708]	0.9515 [0.9330–0.9708]	0.9280 [0.9016–0.9563]
DenseNet+ImageNet-1K	0.9392 [0.9333–0.9608]	0.9390 [0.9330–0.9608]	0.9141 [0.9014–0.9462]
ResNet-TCNN	0.9283 [0.9167–0.9458]	0.9283 [0.9170–0.9457]	0.8945 [0.8775–0.9193]
ResNet+ImageNet-1K	0.9158 [0.8917–0.9375]	0.9162 [0.8905–0.9382]	0.8778 [0.8473–0.9093]

Note: *Mean [Minimum–Maximum], **ImageNet-1K pretrained AlexNet weights were converted from PyTorch to a Keras-compatible format.

Table 4 presents a comparative analysis of the computational complexity of the baseline CNN models and their TCNN-enhanced counterparts in terms of the number of trainable parameters, total prediction time, and average prediction time per image. The inference-time evaluation was performed on the test subset of the corresponding RSV split. Although the TCNN-enhanced models contain more trainable parameters due to the integration of the energy layer, their prediction times remain close to those of the corresponding baseline CNN models. For example, AlexNet and AlexNet-TCNN both required 0.0117 s/image, while DenseNet and DenseNet-TCNN required 0.0282 and 0.0284 s/image, respectively. These results indicate that the energy-layer-based feature fusion improves texture representation without introducing a substantial increase in inference latency.

Table 4: Computational complexity and prediction time comparison between baseline CNN models and TCNN-enhanced models.

DL Model	Number of Trainable Parameters	Total Prediction Time (s)	Average Prediction Time (s/Image)
AlexNet	22,794,307	2.8017	0.0117
AlexNet-TCNN	24,367,939	2.8028	0.0117
DenseNet	27,964,291	6.7600	0.0282
DenseNet-TCNN	32,160,643	6.8181	0.0284
ResNet	48,741,379	6.3031	0.0263
ResNet-TCNN	57,134,083	6.3344	0.0264

To qualitatively examine the feature distributions learned by the baseline CNN and TCNN-enhanced models, t-distributed stochastic neighbor embedding (t-SNE) visualizations were generated from the GAP feature vectors, as shown in Fig. 8. The number of features extracted from the GAP layer was 256 for AlexNet, 2048 for ResNet, 1024 for DenseNet, 640 for AlexNet-TCNN, 3072 for ResNet-TCNN, and 2048 for DenseNet-TCNN. The t-SNE embeddings were generated using the Barnes-Hut algorithm with 50 PCA components, perplexity = 30, exaggeration = 4, learning rate = 500, 1000 iterations, and a fixed random seed. To quantitatively support the visual interpretation, the Silhouette Score was computed on the same two-dimensional t-SNE embeddings. Higher Silhouette Score values indicate better class separation in the projected feature space. The Silhouette Scores were reported as mean [minimum–maximum] across five RSV subsets and were 0.2992 [0.2672–0.3375] for AlexNet, 0.5508 [0.4966–0.6066] for ResNet, 0.4423 [0.3883–0.4876] for DenseNet, 0.3087 [0.2822–0.3323] for AlexNet-TCNN, 0.4191 [0.3449–0.4836] for ResNet-TCNN, and 0.5972 [0.5243–0.6515] for DenseNet-TCNN. Among the examined models, DenseNet-TCNN achieved the highest Silhouette Score, which is consistent with its strong classification performance. AlexNet-TCNN also showed a slight improvement over AlexNet. Although ResNet-TCNN produced a lower Silhouette Score than ResNet, its value still indicated a moderate class separation. Overall, the TCNN-enhanced models achieved comparable or improved class separation in the projected feature space. These findings suggest that the energy-layer-based TCNN structure generally preserves or improves the discriminative feature representation, with the strongest improvement observed in DenseNet-TCNN.

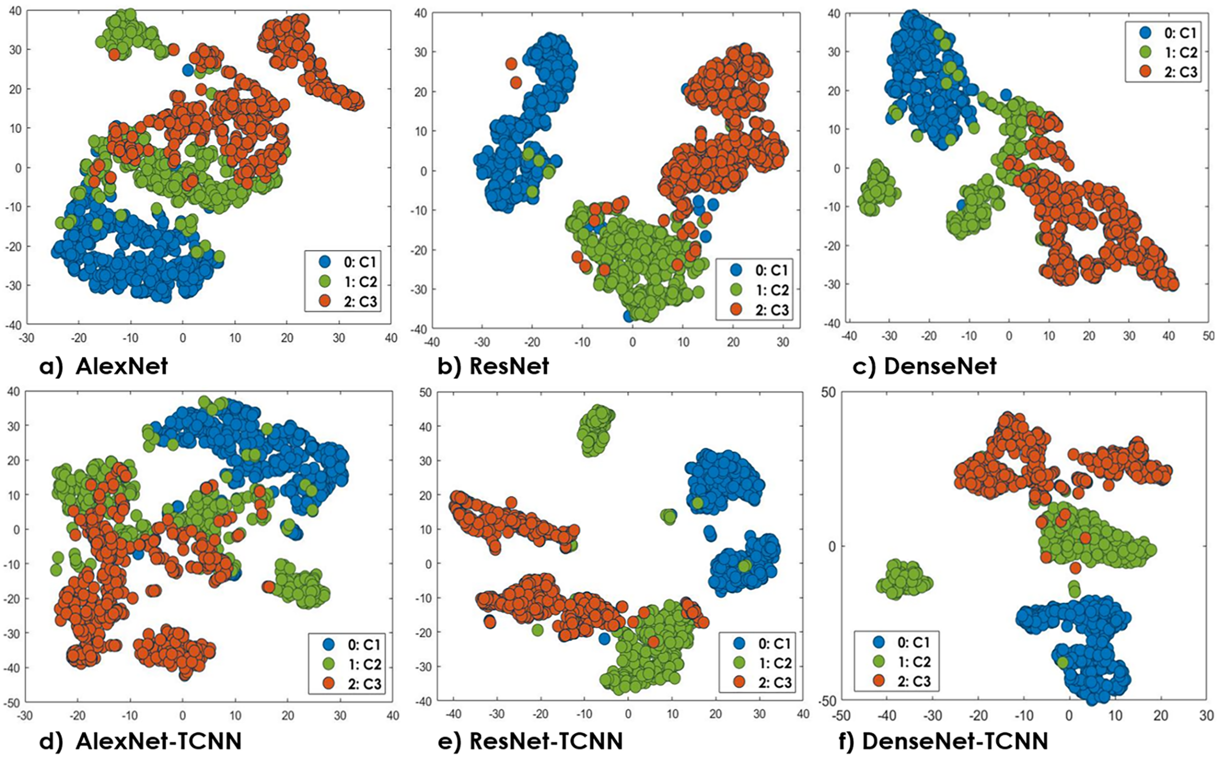

Figure 8: t-SNE visualizations of the GAP feature vectors obtained from the baseline CNN and TCNN-enhanced models: (**a**) AlexNet, (**b**) ResNet, (**c**) DenseNet, (**d**) AlexNet-TCNN, (**e**) ResNet-TCNN, and (**f**) DenseNet-TCNN. The t-SNE embeddings were generated using the Barnes-Hut algorithm with 50 PCA components, perplexity = 30, exaggeration = 4, learning rate = 500, 1000 iterations, and a fixed random seed.

Table 5 presents a comparative performance analysis of baseline CNN models and their TCNN-enhanced counterparts across three datasets: the 28-class marble dataset reported by Canayaz and Uludag [12] the Kylberg texture dataset [27], and the proposed MarbleVision dataset. The results for the Canayaz and Uludag dataset and the Kylberg dataset were obtained in this study by retraining and testing all baseline CNN and TCNN-enhanced models, rather than being taken directly from the literature. The evaluation was conducted using five RSV subsets. Overall, the TCNN-enhanced models improved the performance of their corresponding baseline CNN architectures in most cases, particularly in terms of precision, F1-score, and MCC. This improvement can be attributed to the energy-layer-based feature fusion mechanism, which combines mid-level texture representations with high-level semantic features and therefore provides more discriminative texture-aware representations. However, the magnitude of improvement varies across datasets and backbone architectures, indicating that the effectiveness of TCNN enhancement depends on dataset complexity, intra-class variability, and the representational capacity of the baseline model. The near-perfect performance of DenseNet-TCNN on the Kylberg dataset can be interpreted in relation to the characteristics of this dataset. Since Kylberg contains different texture categories, the classification task is relatively less ambiguous than the quality classification task in the real-world MarbleVision dataset, where the visual differences between classes are more subtle.

Table 5: Mean performance results of baseline CNN and TCNN-enhanced models across different datasets.

Dataset	Model	Accuracy	Sensitivity	Specificity	Precision	F1-score	MCC
Marble-Dataset [12]	AlexNet	0.9663	0.9660	0.9987	0.9669	0.9655	0.9647
	DenseNet	0.9418	0.9469	0.9978	0.9445	0.9436	0.9425
	ResNet	0.9792	0.9787	0.9992	0.9809	0.9789	0.9786
	AlexNet-TCNN	0.9894	0.9901	0.9996	0.9893	0.9893	0.9891
	DenseNet-TCNN	0.9695	0.9685	0.9989	0.9754	0.9708	0.9703
	ResNet-TCNN	0.9713	0.9710	0.9989	0.9732	0.9714	0.9707
Kylberg-Dataset [27]	AlexNet	0.9854	0.9854	0.9995	0.9861	0.9853	0.9850
	DenseNet	0.9981	0.9981	0.9999	0.9982	0.9981	0.9980
	ResNet	0.9708	0.9708	0.9989	0.9720	0.9707	0.9700
	AlexNet-TCNN	0.9896	0.9896	0.9996	0.9903	0.9896	0.9894
	DenseNet-TCNN	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999
	ResNet-TCNN	0.9712	0.9712	0.9989	0.9725	0.9712	0.9705
Our Dataset	AlexNet	0.8858	0.8858	0.9429	0.8899	0.8860	0.8308
	DenseNet121	0.9342	0.9342	0.9671	0.9347	0.9342	0.9015
	ResNet50	0.8925	0.8925	0.9462	0.9058	0.8934	0.8458
	AlexNet-TCNN	0.8925	0.8925	0.9462	0.8945	0.8920	0.8400
	DenseNet-TCNN	0.9516	0.9516	0.9759	0.9526	0.9515	0.9280
	ResNet-TCNN	0.9283	0.9283	0.9645	0.9317	0.9283	0.8945

In this study, marble quality classification was evaluated using five RSV subsets and four predefined ensemble configurations. Equal-weight probability averaging was used as a simple and reproducible ensemble strategy to combine model outputs without introducing additional trainable parameters. More complex fusion strategies, such as weighted averaging and stacking, were not used because they require additional validation-based weight optimization or a trainable meta-classifier, which may increase the risk of overfitting in the limited-dataset setting. Table 6 presents the performance metrics of the ensemble models. Ensemble2 achieved the highest accuracy, F1-score, and MCC values with 0.995, 0.993, and 0.989, respectively. This result shows that combining baseline CNN models with TCNN-enhanced models provides complementary information for marble quality classification. Although Ensemble2 produced the best overall result, Ensemble4 also achieved strong performance with only two TCNN-enhanced models, DenseNet-TCNN and ResNet-TCNN. Its accuracy and MCC values were 0.992 and 0.981, respectively. Therefore, Ensemble4 can be considered a more efficient alternative when model complexity is an important factor.

Table 7 presents the paired t-test results for accuracy and MCC across the five RSV subsets. The ensemble configurations that included TCNN-based models, namely CNNs+TCNNs, TCNNs, and DenseNet-TCNN+ResNet-TCNN, showed statistically significant improvements over the CNN-only ensemble. However, the differences among these TCNN-containing ensemble configurations were not statistically significant. This result indicates that the CNNs+TCNNs ensemble achieved the highest mean performance, while the TCNN-only and DenseNet-TCNN+ResNet-TCNN ensembles provided statistically comparable results with lower model complexity.

Table 6: The performance results of ensemble models using five RSV subsets.

Performance Results	Ensemble1 (CNNs)	Ensemble2 (CNNs+TCNNs)	Ensemble3 (TCNNs)	Ensemble4 (DT&RT)[a]
Accuracy	0.966 (0.947–0.981)[b]	0.995 (0.989–0.997)	0.993 (0.981–0.997)	0.992 (0.978–0.997)
Error	0.034 (0.019–0.053)	0.005 (0.003–0.011)	0.007 (0.003–0.019)	0.008 (0.003–0.022)
Sensitivity	0.948 (0.921–0.971)	0.993 (0.983–0.996)	0.989 (0.971–0.996)	0.988 (0.967–0.996)
Specificity	0.974 (0.960–0.985)	0.996 (0.992–0.998)	0.995 (0.985–0.998)	0.994 (0.983–0.998)
Precision	0.951 (0.926–0.971)	0.993 (0.984–0.996)	0.989 (0.971–0.996)	0.988 (0.967–0.996)
F1-score	0.949 (0.922–0.971)	0.993 (0.983–0.996)	0.989 (0.971–0.996)	0.988 (0.967–0.996)
MCC	0.924 (0.884–0.957)	0.989 (0.975–0.994)	0.984 (0.956–0.994)	0.981 (0.950–0.994)

Note: [a]DT: DenseNet121-TCNN, RT: ResNet50-TCNN, [b]Mean [Minimum–Maximum]

Table 7: Pairwise statistical comparison of the ensemble models using paired t-tests across five RSV subsets.

Comparison*	Accuracy Mean A	Accuracy Mean B	Δ_{Accuracy}	p-value	MCC Mean A	MCC Mean B	Δ_{MCC}	p-value	Significant
E1 vs. E2	0.966	0.995	−0.029	0.002	0.924	0.989	−0.065	0.002	Yes
E1 vs. E3	0.966	0.993	−0.027	<0.001	0.924	0.984	−0.060	<0.001	Yes
E1 vs. E4	0.966	0.992	−0.026	<0.001	0.924	0.981	−0.057	<0.001	Yes
E2 vs. E3	0.995	0.993	0.002	0.294	0.989	0.984	0.005	0.294	No
E2 vs. E4	0.995	0.992	0.003	0.177	0.989	0.981	0.008	0.177	No
E3 vs. E4	0.993	0.992	0.001	0.178	0.984	0.981	0.002	0.178	No

Note: *E1: Ensemble1(CNNs), E2: Ensemble2(CNNs+TCNNs), E3: Ensemble3(TCNNs), E4: Ensemble4(DenseNet-TCNN+ResNet-TCNN). Δindicates the difference between the two mean values, calculated as Mean A − Mean B. Statistical significance was evaluated at $\alpha = 0.05$.

Fig. 9 presents the pooled confusion matrices obtained by summing the confusion matrices across the five RSV test subsets. Since each RSV test subset contained 80 samples per class, the pooled matrices contained 400 prediction instances per class. Ensemble2 correctly classified 398 of 400 C1 prediction instances, 398 of 400 C2 prediction instances, and 395 of 400 C3 prediction instances. For the intermediate C2 class, Ensemble1 misclassified 7 prediction instances as C1 and 7 prediction instances as C3, whereas Ensemble2 reduced these errors to 2 and 0, respectively. These results support the conclusion that the combined CNN and TCNN ensemble reduces confusion between C2 and neighboring quality classes.

3.2 Literature Comparison & Discussion

The fundamental standard for marble slab or quality classification is manual labeling by experts. However, this labeling process can vary according to the expert and be affected by visual fatigue and lighting conditions. Especially in marble quality classification, the process becomes more challenging as the expert classifies the quality of the same type of marble slab as good/bad or low/medium/high quality. For these reasons, ML approaches have been proposed in the literature for objective and consistent marble slab/quality classification criteria. As seen in Table 8, various techniques such as LBP [2,3,11], histogram analyses [1,6,11], SIFT [3,14], PCA [1,5], and scale-invariant local ternary pattern (SILTP) have been used alone or in combination during the feature analysis process feeding the classifiers. For the studies reporting sensitivity, the values generally ranged between 0.971 and 0.993, indicating that handcrafted feature-based methods can achieve strong performance under specific dataset conditions. As is well known, the performance of ML models depends on features, and the processes of selecting and combining features are challenging. Therefore, CNN models, which perform feature analysis within their internal layers, have been proposed for marble slab/quality classification in the literature. The performance of shallow DL models [17,18] which contain a small number of Conv+ReLU+Pooling layers in their feature layers, is 0.75 and 0.81, respectively.

Another shallow CNN model [16] tested without data augmentation had an average accuracy of 0.714 for 6 classes, while its performance increased to 0.922 after data augmentation. These results suggest that shallow CNN models may show limited performance when trained under constrained data conditions or without effective augmentation, whereas their performance can improve substantially when suitable data augmentation strategies are applied.

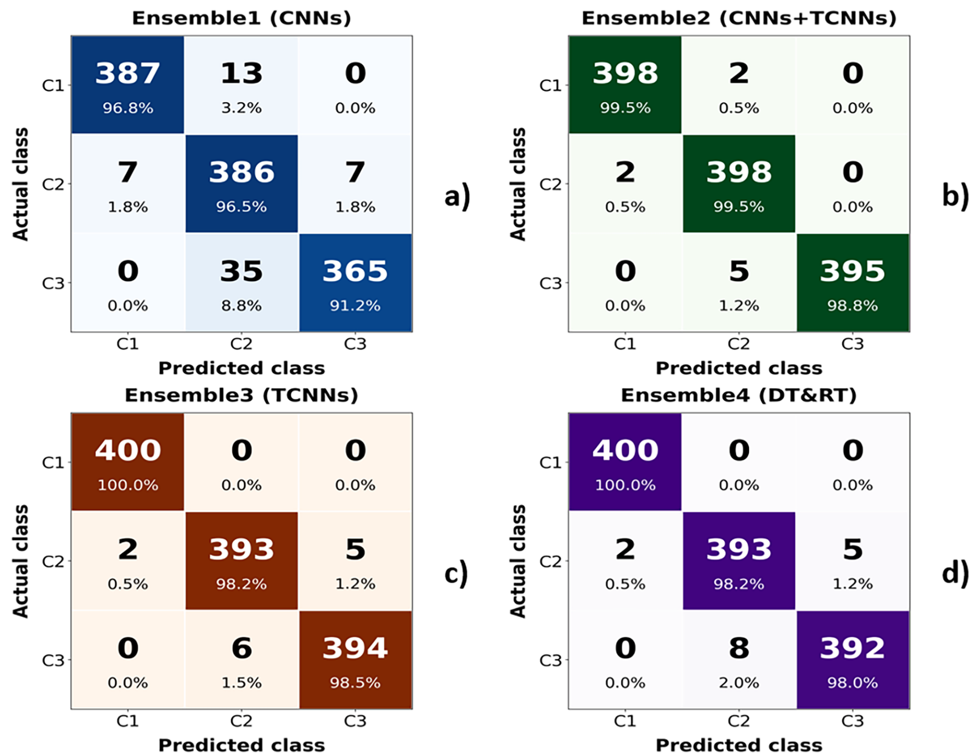

Figure 9: Pooled confusion matrices of the four ensemble models across the five RSV test subsets: (**a**) Ensemble1, (**b**) Ensemble2, (**c**) Ensemble3, and (**d**) Ensemble4. The confusion matrices from the five RSV test subsets were summed for each ensemble model. Since each RSV test subset included 80 samples per class, summing the five RSV confusion matrices resulted in 400 prediction instances for each class. The percentages represent row-normalized class-wise prediction rates.

Table 8: Comparison with previous marble classification studies.

Study	Dataset	Feature Analysis	Model	TL*	Acc[a]	SE[b]
Martínez-Alajarín et al. [1]	Marble Quality (3 classes)	Histogram+PCA	MLP	No	–	0.989
Selver et al. [5]	Marble Quality (4 classes)	Otsu Thresholding+ Statistical Analysis+PCA	HRBFN	No	–	0.971
Doğan and Akay [6]	Marble Quality (4 classes)	Histogram	Adaboost	No		0.975

(Continued)

Table 8 (continued)

Study	Dataset	Feature Analysis	Model	TL*	Acc[a]	SE[b]
Topalova and Tzokev [8]	Marble Quality (10 classes)	Histogram	MLP-1 MLP-2	No	0.920 0.836	–
Karaali and Eminagaoglu [16]	Marble Classification (6 classes)	–	CNN	No	0.922	–
Tiwari et al. [17]	Marble Quality (2 classes)	–	CNN	No	0.810	
Pence and Cesmeli [18]	Marble Quality (2 classes)	–	CNN	No	0.750	–
Torun et al. [11]	Marble Quality (3 classes)	LBP-	SVM AlexNet	No Yes	0.998 0.992	0.993 0.975
Canayaz and Uludag [12]	Marble Classification (28 classes)	–	VGG16	Yes	0.970	–
Öktem et al. [13]	Marble Classification (5 classes)	–	VGG16	Yes	0.967	–
Sidiropoulos et al. [14]	Marble Quality (3 classes)	SILTP SILTP+VGG19	SVM DenseNet201 SVM	- Yes Yes	0.667 0.922 0.950	–
Yavuz and Türkoğlu [23]	Marble Quality (3 classes)	–	GAN+ResNet50+ SVM	Yes	0.964	–
This study	Marble Quality (3 classes)	–	AlexNet-TCNN DenseNet-TCNN ResNet-TCNN Ensemble1[c] Ensemble2[d] Ensemble3[e] Ensemble4[f]	Yes	0.893 0.952 0.928 0.966 0.995 0.993 0.992	0.893 0.952 0.928 0.948 0.993 0.989 0.988

Note: *TL: Transfer learning; [a]Acc: Accuracy; [b]SE: Sensitivity; [c]Ensemble1: AlexNet+DenseNet+ResNet; [d]Ensemble2: AlexNet+DenseNet+ResNet+AlexNet-TCNN+DenseNet-TCNN+ResNet-TCNN; [e]Ensemble3: Alex Net-TCNN+DenseNet-TCNN+ResNet-TCNN; [f]Ensemble4: DenseNet-TCNN+ResNet-TCNN.

As summarized in Table 8, transfer-learning-based CNN models reported in the literature have generally achieved high accuracy values, although direct comparison is limited by differences in datasets, number of classes, acquisition conditions, and evaluation protocols [11–14,18,23]. In this study, three TCNN models, namely AlexNet-TCNN, ResNet-TCNN, and DenseNet-TCNN, were developed by integrating energy-layer-based feature fusion into the corresponding baseline CNN architectures. This design combines intermediate texture-sensitive feature maps with deeper high-level representations to improve marble quality classification. The accuracy performance values calculated by the deep TCNN models for marble quality classification are 0.893, 0.928, and 0.952, respectively. As shown in Table 2 and Fig. 8, the energy layer improved the performance of TL models and enhanced class separation and clustering. Additionally, four ensemble models that combine TL-based CNN and TCNN outputs were evaluated to reduce the misclassification of the intermediate C2 class with neighboring C1 and C3 classes. The pooled confusion matrices in Fig. 9 provide class-level evidence that Ensemble2 achieved the strongest reduction in C2 confusion, while Ensemble3 and Ensemble4 also reduced this confusion compared with the CNN-only ensemble. Ensemble methods effectively combine the strengths of different models to enhance classification accuracy and improve overall performance. In the proposed experimental setting, Ensemble2 achieved the highest performance among the evaluated configurations. In this context, the proposed TCNN and ensemble models are effective for marble quality classification, a fundamentally texture-analysis problem.

It should be noted that the studies summarized in Table 8 were conducted using different datasets, numbers of classes, acquisition systems, feature extraction strategies, and validation protocols. Therefore, the comparison should be interpreted as a literature-level performance overview rather than a strictly controlled head-to-head benchmark. Within the proposed experimental protocol, the TCNN-enhanced models and ensemble configurations demonstrated strong and competitive performance for marble quality classification.

4 Conclusion

Marble classification is traditionally performed through expert visual assessment, which is inherently subjective and prone to inconsistencies. In this study, the proposed DL model demonstrated high accuracy and sensitivity in classifying marble slabs based on a dataset jointly labeled by multiple operators, ensuring greater reliability in the ground truth annotations. The results indicate that the proposed texture-focused DL models, particularly when combined in an ensemble configuration, achieved competitive and high classification performance compared with existing approaches reported in the literature. Since the datasets used in previous studies are not always publicly accessible, the proposed model was primarily validated through comparisons with baseline CNN architectures and their TCNN-enhanced counterparts under the same experimental protocol. A key aspect of future work involves the industrial deployment of the model within MarbleVision, an automated marble classification device previously developed and integrated into real-world applications by the authors.

Although the proposed ensemble model achieved high classification performance, several limitations should be noted. The dataset was collected using a specific industrial acquisition setup and includes three marble quality classes; therefore, additional validation on larger multi-site datasets, different marble types, and varying illumination conditions is required to further assess generalization. Moreover, the ensemble model increases the number of trainable parameters compared with individual models, which should be considered for real-time embedded deployment. Future work will focus on expanding the dataset, evaluating cross-factory robustness, and optimizing the model for edge-based industrial implementation.

Beyond marble classification, the proposed framework offers a scalable and adaptable solution that can be extended to other materials and industries requiring aesthetic-based evaluation and classification, such as ceramics, textiles, and wood surface inspection. By providing an efficient and objective classification approach, this study lays the groundwork for advancing automated quality assessment in industrial applications.

Acknowledgement: The authors would like to thank Kapsam Electromechanical Industry Co., Ltd., for supporting the development and industrial implementation of the MarbleVision system. This study includes research conducted by Burak Seckin as part of his ongoing master's thesis under the supervision of Yunis Torun and Rukiye Karakis. The data used in this article can be obtained electronically by contacting the corresponding author.

Funding Statement: This work was supported by the Scientific and Technological Research Council of Türkiye (TUBITAK) under the 1507 SME R&D Startup Support Programme, Project Code 7218062.

Author Contributions: Conceptualization: Yunis Torun and Rukiye Karakis; methodology: Yunis Torun and Burak Seckin; software: Burak Seckin; validation: Yunis Torun, Burak Seckin and Rukiye Karakis; formal analysis: Yunis Torun and Burak Seckin; investigation: Burak Seckin; resources: Yunis Torun; data curation: Burak Seckin; writing—original draft preparation: Burak Seckin; writing—review and editing: Yunis Torun and Rukiye Karakis; visualization: Burak Seckin; supervision: Yunis Torun and Rukiye Karakis; project administration: Yunis Torun; funding acquisition: Yunis Torun. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: Data available on request from the authors.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study. Yunis Torun is affiliated with Kapsam Electromechanical Industry and Trade Co. Ltd., which supported the development and industrial implementation of the MarbleVision system. The authors confirm that the reported results and conclusions were prepared independently and objectively.

References

1. Martinez-Alajarin J, Luis-Delgado JD, Tomas-Balibrea LM. Automatic system for quality-based classification of marble textures. IEEE Trans Syst Man Cybern Part C Appl Rev. 2005;35(4):488–97. doi:10.1109/TSMCC.2004.843236.
2. Turan E, Ucar F, Dandil B. A novel marble recognition system using extreme learning machine with LBP and histogram features. Concurr Comput. 2021;33(21):e6428. doi:10.1002/cpe.6428.
3. Turan E, Dandil B, Bal C. Grafiksel arayüz tabanlı mermer Sınıflandırma uygulaması geliştirme. Gazi Üniversitesi Fen Bilim Derg Part C Tasarım Ve Teknol. 2020;8(4):1034–50. doi:10.29109/gujsc.818058.
4. González E, Bianconi F, Álvarez MX, Saetta SA. Automatic characterization of the visual appearance of industrial materials through colour and texture analysis: an overview of methods and applications. Adv Opt Technol. 2013;2013(1):503541. doi:10.1155/2013/503541.
5. Selver MA, Akay O, Ardali E, Yavuz AB, Onal O, Ozden G. Cascaded and hierarchical neural networks for classifying surface images of marble slabs. IEEE Trans Syst Man Cybern Part C Appl Rev. 2009;39(4):426–39. doi:10.1109/TSMCC.2009.2013816.
6. Doğan H, Akay O. Using AdaBoost classifiers in a hierarchical framework for classifying surface images of marble slabs. Expert Syst Appl. 2010;37(12):8814–21. doi:10.1016/j.eswa.2010.06.019.
7. López M, Martínez J, Matías JM, Taboada J, Vilán JA. Functional classification of ornamental stone using machine learning techniques. J Comput Appl Math. 2010;234(4):1338–45. doi:10.1016/j.cam.2010.01.054.
8. Topalova I, Tzokev A. Automated texture classification of marble shades with real-time PLC neural network implementation. In: Proceedings of the 2010 International Joint Conference on Neural Networks (IJCNN); 2010 Jul 18–23; Barcelona, Spain. p. 1–8. doi:10.1109/IJCNN.2010.5596494.
9. Talaei Khoei T, Ould Slimane H, Kaabouch N. Deep learning: systematic review, models, challenges, and research directions. Neural Comput Appl. 2023;35(31):23103–24. doi:10.1007/s00521-023-08957-4.
10. Dong S, Wang P, Abbas K. A survey on deep learning and its applications. Comput Sci Rev. 2021;40(1):100379. doi:10.1016/j.cosrev.2021.100379.

11. Torun Y, Akbaş MR, Çelık MA, Kaynar O. Development a machine vision system for marble classification. In: Proceedings of the 2019 27th Signal Processing and Communications Applications Conference (SIU); 2019 Apr 24–26; Sivas, Turkey. p. 1–4. doi:10.1109/SIU.2019.8806419.
12. Canayaz M, Uludağ F. Marble classification using deep neural networks. Eur J Technol. 2020:52–63. doi:10.36222/ejt.671527.
13. Öktem M, Akosman ŞA, Moral ÖT, Kiliç V. Derin öğrenme tabanlı mermer Yüzeylerinin otomatik Sınıflandırılması. Eur J Sci Technol. 2021;26:73–7. doi:10.31590/ejosat.950950.
14. Sidiropoulos GK, Ouzounis AG, Papakostas GA, Lampoglou A, Sarafis IT, Stamkos A, et al. Hand-crafted and learned feature aggregation for visual marble tiles screening. J Imaging. 2022;8(7):191. doi:10.3390/jimaging8070191.
15. Seckin B, Torun Y, Karakis R. Mermer sınıflandırma problemi için güvenilir veri seti oluşturulması. In: Proceedings of the International Cumhuriyet Artificial Intelligence Applications Conference; 2022 Dec 8–9; Online. p. 35–8.
16. Karaalİ İ, Emİnağaoğlu M. Mermer işlemede kalite sınıflandırması için evrişimsel sinir ağı modeli. Gazi Üniversitesi Mühendislik Mimar Fakültesi Derg. 2020;36(1):347–58. doi:10.17341/gazimmfd.720041.
17. Tiwari RG, Yadav DS, Misra A. Performance evaluation of optimizers in the classification of marble surface quality using CNN. In: Proceedings of Third Doctoral Symposium on Computational Intelligence. Singapore: Springer Nature; 2023. p. 181–91. doi:10.1007/978-981-19-3148-2_15.
18. Pence I, Cesmeli M. Deep learning in marble slabs classification. Sci J Mehmet Akif Ersoy Univ. 2019;2(1):21–6.
19. Andrearczyk V, Whelan PF. Using filter banks in convolutional neural networks for texture classification. Pattern Recognit Lett. 2016;84(1):63–9. doi:10.1016/j.patrec.2016.08.016.
20. Simon P, Uma V. Deep learning based feature extraction for texture classification. Procedia Comput Sci. 2020;171(6):1680–7. doi:10.1016/j.procs.2020.04.180.
21. Goyal V, Sharma S. Texture classification for visual data using transfer learning. Multimed Tools Appl. 2023;82(16):24841–64. doi:10.1007/s11042-022-14276-y.
22. Yavuz M, Türkoğlu İ. Determination of the quality classes of Elazig cherry marble with image processing. Ain Shams Eng J. 2025;16(8):103455. doi:10.1016/j.asej.2025.103455.
23. Yavuz M, Türkoğlu İ. Enhancing marble image classification performance via super-resolution-assisted image improvement. Ain Shams Eng J. 2026;17(1):103858. doi:10.1016/j.asej.2025.103858.
24. Priya GS, Padmapriya N. PT-CNN: a non-linear lightweight texture image classifier. Neural Process Lett. 2023;55(6):8483–507. doi:10.1007/s11063-023-11322-0.
25. Efe T, Sallan SG. Türkiye'de Doğal Taş Sanayinin Coğrafi ve Ekonomik Sürdürülebilirliği. Acad J Soc Sci. 2017;3(1):173–81.
26. Kapsam Elektromekanik Ltd., Sti. [Internet]. 2022 [cited 2026 Jan 1]. Available from: https://kapsamelektro.com.tr/Products/MarbleVision.
27. Kylberg G. The Kylberg texture dataset v.1.0 (External Report (Blue Series), Issue 35) [Internet]. 2011 [cited 2026 Jan 1]. Available from: https://kylberg.org/kylberg-texture-dataset-v-1-0/.
28. Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. Commun ACM. 2017;60(6):84–90. doi:10.1145/3065386.
29. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2016 Jun 27–30; Las Vegas, NV, USA. p. 770–8. doi:10.1109/CVPR.2016.90.
30. Huang G, Liu Z, Van Der Maaten L, Weinberger KQ. Densely connected convolutional networks. In: Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2017 Jul 21–26; Honolulu, HI, USA. p. 2261–9. doi:10.1109/CVPR.2017.243.