AUT Journal of Electrical Engineering
10.22060/EEJ.2026.24823.5773

SaveDoc: A Robust Technique for Degradation Classification on Ancient Document Images

Khairun Saddami[1,*], Rizka Ramadhana[1], Fathurrahman Fathurrahman[1,2], Khairul Munadi[1], Fitri Arnia[1]

[1]Department of Electrical and Computer Engineering, Faculty of Engineering, Universitas Syiah Kuala, Banda Aceh, Indonesia
[2]Electrical Engineering Department, College of Engineering and Physics, King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia

*Corresponding Author: khairun.saddami@usk.ac.id

Abstract:

The analysis of degraded ancient documents remains challenging because multiple degradation patterns may appear within a single document image and reduce document quality. In this paper, we propose a combination of convolutional neural networks and extreme learning machine for patch-level degradation classification in ancient document images. The dataset was constructed from 100 source document images and cropped into 700 labeled patches. Four pre-trained architectures, namely VGG19, ResNet101, MobileNetV2, and EfficientNetB0, were used as feature extractors. The number of hidden nodes in the ELM classifier was tuned, and two activation functions, namely radial basis function and sigmoid, were evaluated. The experimental results show that EfficientNetB0 features combined with the extreme learning machine classifier achieved the best performance, with a testing accuracy of 98.21% ± 2.08%. This combination also produced faster inference time compared with the other evaluated models. These results indicate that the EfficientNetB0-extreme learning machine combination is effective for patch-level degradation classification in ancient document images.

Keywords:
Degraded Ancient Documents, Degradation Classification, Extreme Learning Machine, Pre-trained CNN features, EfficientNet.

1. Introduction

Document image analysis is an important field that enables computers to extract the content of documents, which is important for various applications, including historical research, cultural preservation, and information retrieval. However, analyzing ancient documents is more challenging due to the severe degradation inside the documents. To overcome these challenges, accurate enhancement and restoration are important steps that must be taken before analysis can occur, as they can significantly improve the readability and recognizability of the document. Effective enhancement can lead to a more accurate extraction of information, making it easier to restore and analyze the content of the document [1] .

Current methods for document analysis are limited in their ability to address the various types of degradation that occur, such as ink bleed, paper degradation, and digitalization noise. Mostly, these methods focus on a narrow range of issues and are not ready to handle the complexities of real-world datasets, which can suffer from multiple types of degradation simultaneously. Furthermore, the digitalization process can introduce additional noise, particularly when using devices with low sensor sensitivity or under poor lighting conditions, which can fail simple image analysis techniques. Ancient documents often suffer from

During the preparation of this work, the author(s) used [chatgpt] for translation assistance from Bahasa Indonesia to English and for language polishing. After using this tool, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the publication.

multiple types of degradation, making it difficult to enhance and analyze them and requiring more advanced techniques or approaches to address these challenges [2].

The main idea to address these challenges is to divide the documents into smaller sections, making it easier to detect and classify the types of degradation present. Recognizing the types of degradation in ancient documents simplifies the restoration and information-extraction processes, as it allows for more targeted and effective enhancement techniques to be applied. However, manual recognition is not feasible due to the vast number and variety of degraded documents worldwide, making it important to develop automatic degradation classification methods [3].

Despite its significance, automatic degradation classification has received limited attention, with few studies addressing this challenge. Traditional machine learning methods, such as random forest and support vector machines, have been employed, achieving moderate accuracy (up to 96.77% for worn-holes degradation) [4-6]. However, these approaches have limitations, and deep learning-based methods have recently been proposed to improve performance. For example, a study using CNN models achieved 100% accuracy on an unblind dataset but the CNN model has a large number of learning parameters, resulting in long training and inference times. This highlights a gap in existing research: the need for robust and efficient methods that balance accuracy and computational complexity [7].

One of the techniques to accelerate the training and inference process is Extreme Learning Machines (ELM). Compared to traditional artificial neural networks (ANN) and convolutional neural networks (CNN), the ELM resulted in a significantly faster training process [8], which is primarily due to the fact that ELM does not require iterative updates during training, thereby enabling the fast and accurate estimation of weights between input and hidden layers. This advantage is particularly pronounced when dealing with small datasets, where ANN and CNN may be prone to overfitting or require extensive tuning to achieve satisfactory performance [9, 10] . Moreover, ELM is built by a simpler math computation but leads to its higher generalization performance, lower computational cost, and reduced susceptibility to hyperparameter tuning. These characteristics leads ELM an attractive option for a variety of machine-learning applications, particularly those with limited time and computing resources [11-13].

Based on the aforementioned, this study aims to develop a model to classify degradation on ancient documents by combining the strengths of CNN as a feature extractor and ELM as a classifier. Beyond applying CNN–ELM to a new domain, this study contributes the following insights. First, it reveals the efficiency trade-offs among different CNN backbones, showing that EfficientNetB0 combined with ELM achieves accuracy comparable to deeper architectures such as ResNet101 while requiring significantly fewer parameters and faster inference, making it more suitable for resource-constrained scenarios. Second, it characterizes the architectural behavior of pre-trained CNNs when paired with ELM, demonstrating that RBF and Sigmoid activation functions exhibit opposite sensitivity to the number of hidden nodes, and that ResNet101 and EfficientNetB0 are more robust to such variations than MobileNetv2 and VGG19. Third, it provides insights into degradation-type separability, showing that faint text and low contrast share visual features with smear/spot/stain and uniform classes, leading to occasional misclassification, while the other degradation types are clearly distinguishable.

The rest of this paper is organized as follows: Section 2 provides related works, highlighting the existing research in the field. Section 3 describes the material and method employed in this study, including the dataset, the CNN architecture, the ELM approach, and implementation details. Section 4 presents the results and discussion. Finally, Section 5 concludes the paper, summarizing the key contributions.

2. Related Works

Previous research has been conducted on degradation classification in ancient document images. Shahkolaei introduced the VDIQA metric, which is designed to assess the quality of degraded document images [4].

This metric works by segmenting input images into four layers: text, degradation regions close to the text, degradation regions farther from the text, and non-degraded pixels. The method then applies a log-Gabor filter for segmentation and uses the MSCN coefficient for feature extraction. The extracted features are converted to Mean Opinion Score (MOS) values using Support Vector Regression (SVR), and the VDIQA metric is compared with other blind image quality assessment (BIQA) metrics, showing superior performance. Although the VDIQA metric presents an innovative approach to assessing degraded document image quality, it faces several challenges. For example, while the segmentation strategy and MSCN coefficient work well for mild degradation, they may not be sufficiently robust for more complex or severe distortions. Another limitation is the use of SVR for mapping features to MOS values, which may not effectively generalize to new or unseen degradation types.

Multi-Distortion Document Quality Measure (MDQM) has been proposed for degraded document image classification [5]. This framework aims to evaluate the quality of degraded document images by segmenting the images into text and non-text regions. The MDQM proposed the Mean Subtracted Contrast Normalized (MSCN) coefficient for feature extraction and also modifies the locally weighted mean phase angle (LWMPA) for extracting features from the RGB channels of the images. The framework then uses an SVM classifier to categorize the degraded images into four classes: paper translucency, reader annotations, stains, and worn holes. The method achieved accuracy of 85.07%. Several limitations can be identified in this research—first, the proposed framework performed on handcrafted features. Furthermore, while the MSCN coefficient and LWMPA for feature extraction work well for the degradations addressed in the study, these methods may not generalize to all possible degradation scenarios. The integration of deep learning-based feature extraction methods might offer more flexibility in handling diverse and more complex degradation types.

Another limitation of the MDQM framework is that its evaluation is based on relatively controlled or synthetic datasets. The true performance of the method on real-world data, where document degradation might be more varied and unpredictable, remains uncertain. Further evaluation on diverse datasets with varying levels of degradation is required to test its robustness and generalizability. Additionally, the study does not address the scalability of the method, particularly in terms of its ability to handle large datasets or high-resolution images.

Wang introduced a novel deep learning module for classifying low-quality image datasets, with a particular focus on images affected by common degradation types such as fog, low contrast, and brightness distortion [14]. The proposed method begins by extracting low-level features through convolutional layers from pre-trained models like VGG16 and AlexNet. These layers, termed Shallow Pretrained Layers (SPL), help differentiate degraded features from clear ones. Furthermore, the study proposes a Feature De-drifting Module (FDM) designed to address degraded features. The FDM works by enhancing degraded features and comparing them to clear features, using Mean Squared Error (MSE) loss to guide model parameter updates. This framework reduced the impact of degradation on image quality, aiming to improve classification accuracy.

Experimental results indicate that the method performs well for images with low degradation levels, achieving classification accuracy between 60–70%. However, the model's performance significantly declines when faced with higher levels of degradation, indicating a limitation in handling more severe distortions. Moreover, the study only considers relatively simple degradation types like fog and low contrast without exploring more complex distortions, such as bleed-through and faint text.

Recently, deep learning-based methods have been proposed for degradation classification in ancient documents [7, 15]. Saddami evaluated three CNN models for degradation classification in four categories: Bleed-through/show-through, Faint-text and low-contrast, Smear-stain-spot, and uniform [15]. This research fine-tuned ResNet101, MobileNetV2, and ShuffleNet for degradation classification. ShuffleNet obtained the best result with 100% accuracy on the unblind dataset and 85% accuracy on the blind dataset. Furthermore, Arnia proposed an improved CNN architecture based on ShuffleNet and MobileNetv2, namely DCNet [7]. They proposed a transition layer after applying a Swish activation to the concatenated features of ShuffleNet and MobileNetv2. DCNet demonstrates exceptional performance; however, its large number of learning parameters results in extended training and inference time.

This study aims to accelerate training and testing processes while maintaining deep learning classification accuracy. Extreme Learning Machine (ELM) is one of the methods that can accelerate both training and inference processes. Several studies have successfully implemented ELM by utilizing features extracted from pre-trained deep learning benchmark models, including [16-18]. These studies have shown that the combination of ELM and CNN can improve testing and inference speeds while maintaining high accuracy.

3. Material and Method

In this section, firstly we explained the material we used in the experiments, including the dataset. Furthermore, we described the Convolutional Neural Networks (CNN) architecture used in the experiments, including VGG19, ResNet101, MobileNetv2, and EfficientNetB0. Then, we also discussed the Extreme Learning Machine (ELM) as a classifier. Finally, we presented the implementation details of the experiments.

3-1- Convolutional Neural Networks

Convolutional Neural Networks have been successfully applied to various challenging tasks, such as face detection, autonomous vehicles, intelligent services, biomedical intelligence support, and others [19]. Various CNN architectures have been developed to improve feature representation and classification performance across different domains. One of the widely used CNN architectures is VGG19 [20]. VGG19 was proposed in 2014 as a CNN consisting of 19 layers. The architecture consists of 16 convolutional layers and 3 fully connected layers. This model was originally designed to classify images into up to 1000 categories. The architecture of VGG19 is shown in Fig. 1.

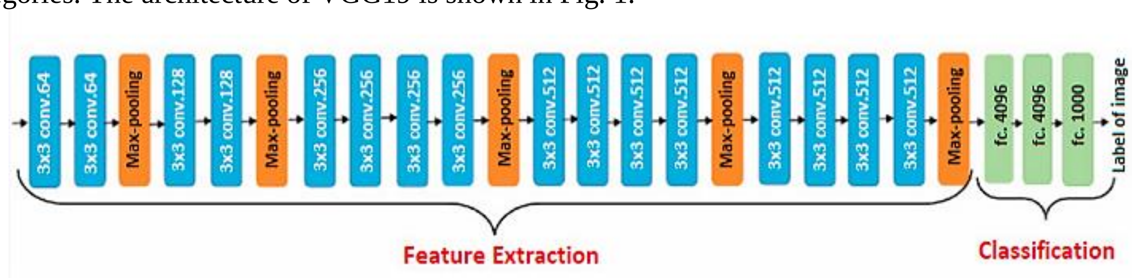


Fig. 1. VGG19 architecture

In VGG19, the convolutional layers act as feature extractors, while the fully connected layers perform classification. Due to its strong feature extraction capability, VGG19 has also been used in historical document image analysis. VGG19 showed good performance when combined with other models, such as the Siamese model [21]. For example, a VGG19-based Siamese model with a Sigmoid similarity function achieved 97% accuracy for historical Arabic document image classification, although the training process was relatively slow [22].

One of the most popular models is ResNet, which uses the deep residual network concept [23]. ResNet101 is a variant of ResNet that has 101 layers and is designed to learn deep feature representations through residual connections. Fig. 2 shows a building block of ResNet, which has a weighted layer followed by a ReLU activation function. The term $F(x) + x$ represents a shortcut connection that enables identity mapping across stacked layers. ResNet has been widely used to classify document images, such as in [24]. In the study, ResNet101 was used as the backbone network for Mask R-CNN. In Shui manuscript character

AUT Journal of Electrical Engineering
10.22060/EEJ.2026.24823.5773

classification, ResNet101 achieved a higher mAP score while compared with other networks, such as CSP-Darknet, and VGG16 [25].

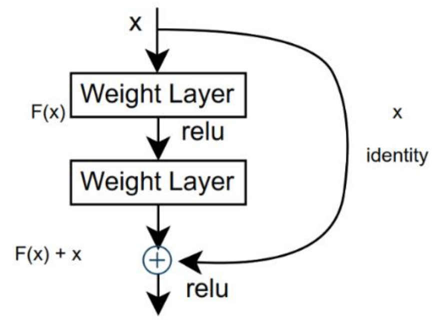


Fig. 2. Basic block architecture of ResNet101

MobileNetV2 is a lightweight CNN architecture based on inverted residual blocks, designed to improve performance in mobile and resource-constrained environments [26]. MobileNetV2 has fewer parameters while maintaining better accuracy than MobileNetV1. Fig. 3 shows the building block of MobileNetV2. The block consists of convolutional and depthwise convolutional layers with ReLU activation. MobileNetV2 has been used to develop optical character recognition in Arabic manuscripts [27]. It has also been applied to invoice image classification, where it achieved better accuracy than VGG19 and LeNet5 with moderate training time [28].

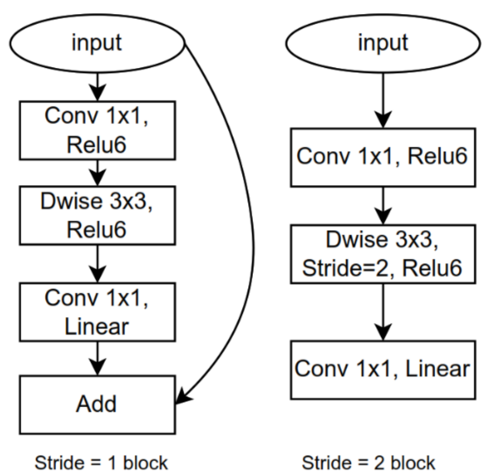


Fig. 3. MobileNetv2 baseline blocks.

EfficientNet was developed by scaling network depth, width, and input resolution using a fixed compound scaling coefficient, allowing it to reduce parameters while maintaining high accuracy [29]. The illustration of EfficientNet is shown in Fig. 4. The compound scaling strategy adjusts the depth, width, and resolution of the network in a balanced manner. EfficientNetB0 is the smallest architecture in the EfficientNet family. It provides competitive accuracy with a smaller model size than ResNet50 and DenseNet169. Previously, EfficientNetB0 has been used to classify document images [30]. This research showed that standalone

EfficientNetB0 resulted in an accuracy score of about 94% for the image stream. This indicates that EfficientNetB0 can provide an effective balance between classification performance and computational efficiency for document image classification.

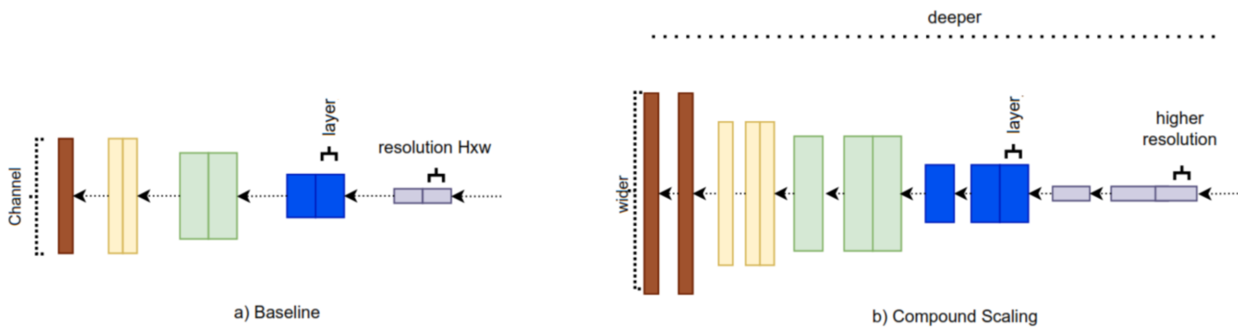


Fig. 4: EfficientNet baseline and compound-scaling

3-2-Extreme Learning Machine

The Extreme Learning Machine (ELM) is a type of single-hidden-layer feedforward neural network that reforms the traditional training process by eliminating the need for backpropagation. Unlike conventional neural networks, ELM updates its weights using the Moore-Penrose generalized inverse method, which significantly reduces the training time. This approach is faster because it avoids iterative training in the model. This innovative approach bypasses the time-consuming process of repeated parameter tuning, making ELM a highly efficient and scalable solution. By avoiding the backpropagation step, ELM resulted in faster model deployment [8]. Fig. 5 shows an example of ELM architecture.

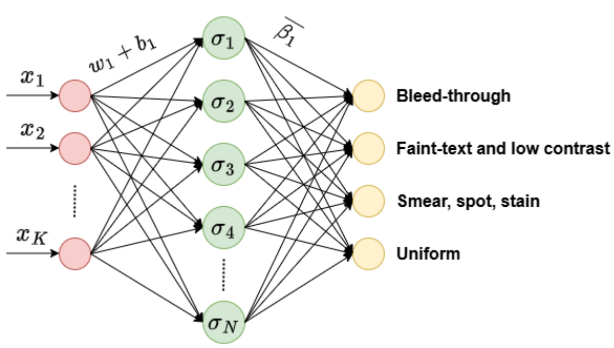


Fig. 5: Extreme learning machine

Given a training dataset $\{(x_i,t_i) \mid x_i \in \mathbb{R}^n, t_i \in \mathbb{R}^m, i = 1, \ldots, N\}t$ with N input–target pairs, the output of an SLFN with L hidden nodes can be expressed as:

$$f(x) = \sum_{j=1}^{L} \beta_j \sigma(w_j \cdot x + b_j) \quad (1)$$

where $\sigma(.)$ is the activation function, w_j is the random weight, x is the input, and b_j is the random bias. To calculate the ELM model, the network in Eq. (1) can be written as Eq. (2):

$$T = H.\beta \tag{2}$$

where **T** is the target label, **β** is a special matrix which is model of ELM, and **H** is the activation of weight, bias and input data. **H** is formulated as Eq. (3):

$$H = \begin{pmatrix} \sigma(w_1.x_1 + b_1) & \cdots & \sigma(w_L.x_1 + b_L) \\ \vdots & \ddots & \vdots \\ \sigma(w_1.x_N + b_1) & \cdots & \sigma(w_L.x_N + b_L) \end{pmatrix} \tag{3}$$

where σ is the activation function, w is weighted value of ELM model, b is bias value, x is the input value. To compute the output weights, ELM solves the following least-squares problem using Eq. (4):

$$\min_{\beta} \|H\beta - T\|_2^2 \tag{4}$$

The solution is obtained via the Moore–Penrose pseudoinverse that is formulated as Eq. (5):

$$\beta = H'.T \tag{5}$$

where **H'** is pseudo-inverse or Moore-Penrose inverse of **H**. if $\mathbf{H^T H}$ is nonsingular, then the pseudo-inverse can be written as Eq. (6):

$$H' = (H^T H)^{-1} H^T \tag{6}$$

In practice, to avoid overfitting and improve stability, a regularized form (ridge regression) is often applied as in Eq. (7):

$$\beta = (H^T H + \lambda I)^{-1} H^T T \tag{7}$$

with $\lambda>0$ as the regularization coefficient.

Once the output weights β are determined, the prediction for a new input x is computed directly using Eq. (8):

$$T = H(x).\beta \tag{8}$$

The ELM model offers faster convergence during training and eliminates the need for iterative weight adjustment [16, 31, 32]. ELM resulted in better performance in classification tasks. In [16], the proposed method is a combination of CNN and ELM. They built CNN for feature extraction and ELM for fast learning on extracted features. In [32] they used AlexNet and combined it with the ELM model in case to propose a lighter network. The reported training time showed that AlexNet + ELM reached a lower time than conventional AlexNet.

3-3-Dataset

The dataset that was conducted in this research comprises the DIBCO, PHIBD, and Private Jawi Dataset [7, 15, 33-42]. Prior to training, the dataset was pre-processed by cropping it into patches of 224x224 pixels, with the cropped areas specifically selected to represent degradation. Four types of degradation were utilized, namely: Bleed and show through; Faint text and low contrast; smear, spot, and stain; and uniform. These degradation types are predominantly found in ancient documents [7]. The dataset used in this study was constructed from 100 document images collected from DIBCO, PHIBD, and a private Jawi manuscript dataset. Each document image has a resolution up to 2048 × 2048 pixels, and may contain multiple degradation patterns within the same document image. Thus, instead of using each document image as a single image, regions containing specific degradation patterns were manually selected and cropped into 224 × 224 image patches. Several patches may originate from the same source document due to combined degradation within a single document. Nevertheless, these patches were cropped from different non-overlapping spatial regions, and no duplicated or identical patches were included in the dataset. This cropping process produced 700 labeled degradation patches. These patches were categorized into four degradation classes: bleed-through or show-through, faint text and low contrast, smear or spot or stain, and uniform. The dataset was divided into two subsets: 80% for training and 20% for testing purposes. Fig. 6 shows example images in the dataset.

(a) Bleed-through

(b) Faint text and low contrast

(c) Smear, spot or stain

(d) Uniform

Fig. 6. Example of ancient document degradation images

3-4- Implementation

As shown in Fig. 7, the features were extracted from the final feature layer of CNN models. Next, the feature set was classified using ELM. In this study, we conducted a transfer learning approach by using pre-trained Convolutional Neural Network (CNN) models to extract relevant features from images. Specifically, we selected four benchmarking CNN architectures, namely VGG19, ResNet101, MobileNetv2, and EfficientNetB0, which have been pre-trained on ImageNet dataset. To ensure

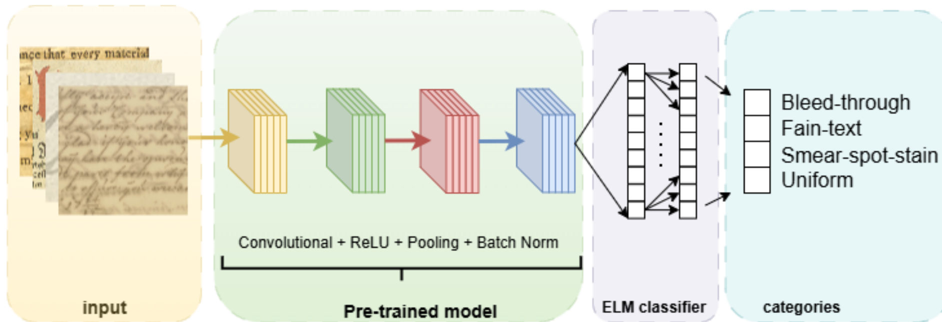


Fig. 7. The proposed frameworks

compatibility, we resized the input images into 224×224 pixels because this input size is required for these pre-trained models. Subsequently, we modified the pre-trained models by removing their classification layers and replacing them with an Extreme Learning Machine (ELM). The dataset was then split into training and testing sets in an 80:20 ratio, allowing us to evaluate the performance of the ELM.

In the simulation, we performed two stages: the training stage and the testing stage. In the training stage, we investigated and evaluated different numbers of hidden nodes and activation functions in the ELM structure. Then, the testing stage was performed, and the best model performance was investigated, which was deployed in the training model. During the training process, we conducted a systematic investigation of the impact of varying the number of hidden neurons in the ELM, ranging from 10^1 to 10^5, on the accuracy of the model. We investigated the optimal number of hidden nodes, treating it as a hyperparameter to be tuned. Specifically, we examined the performance of our model with varying numbers of hidden nodes, including 1, 10, 100, 1000, 2000, 5000, and 10000. This wide search was motivated by the need to identify the most suitable architecture for our specific problem. It is worth noting that the ELM approach differs from traditional deep learning methods in that it relies on pre-extracted features as input rather than learning features from the dataset during training.

Notably, previous research revealed that the activation function of the radial basis function (RBF) yielded the most promising results on ELM. Therefore, we used this activation function in the study and compared it with the sigmoid activation function. The RBF and sigmoid activation functions have been widely recognized for their ability to capture complex patterns and relationships in data effectively. Moreover, we investigated the impact of varying the number of hidden nodes to gain a deeper understanding of the interplay between model complexity and performance. This rigorous approach enabled us to identify the optimal configuration for our model, thereby ensuring that our results were reliable and generalizable.

For comparison purposes, we compared the proposed method with DCNet. This model classified degradation on ancient documents based on CNN models, SVM with RBF kernel and one-against-one approach, and Random Forest with 1000 trees inside it. We used VDQAM as a feature extractor for the SVM and RF. Furthermore, all experiments were conducted using an Intel Core i7 computer with 16 GB RAM and a GPU with 6 GB VRAM. To investigate the stability of ELM models against variations in initial random weights, we conducted ten independent runs using different random seeds and reported the mean and standard deviation for each evaluation metrics. Furthermore, the reported inference time represents the total elapsed time required to process the entire test set of 140 image patches. The test set was processed in a single run, corresponding to a full-batch inference setting. This timing includes the testing-stage processing of the proposed framework, namely CNN feature extraction and ELM classification. Therefore, the reported value reflects the practical processing time needed to classify all test images. The corresponding average per-image processing time can be obtained by dividing the reported total time by the number of test images.

3-5- Evaluation Measures

We used a confusion matrix to evaluate the proposed method. We measured the confusion matrix using accuracy, recall and average recall, precision and average precision, and f-measure and average f-measure. Accuracy is defined as Eq. (9) [7]:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{9}$$

while *TP* is the true positive, which represents a correct prediction of the class; *TN* is the true negative, which represents a correct prediction of outside the class; *FN* is the false negative, which represents an incorrect prediction of the class; *FP* is the false positive, which represents an incorrect prediction of outside the class. Recall and average recall are defined as Eq. (10):

$$Recall_i = \frac{TP_i}{TP_i + FN_i}$$
$$Average\ Recall = \frac{1}{N}\sum_{i=1}^{N} Recall_i \tag{10}$$

Precision and average precision are defined as Eq. (11):

$$Precision_i = \frac{TP_i}{TP_i + FP_i}$$
$$Average\ Precision = \frac{1}{N}\sum_{i=1}^{N} Precision_i \tag{11}$$

F-measure and average *F-measure* are defined as Eq. (12):

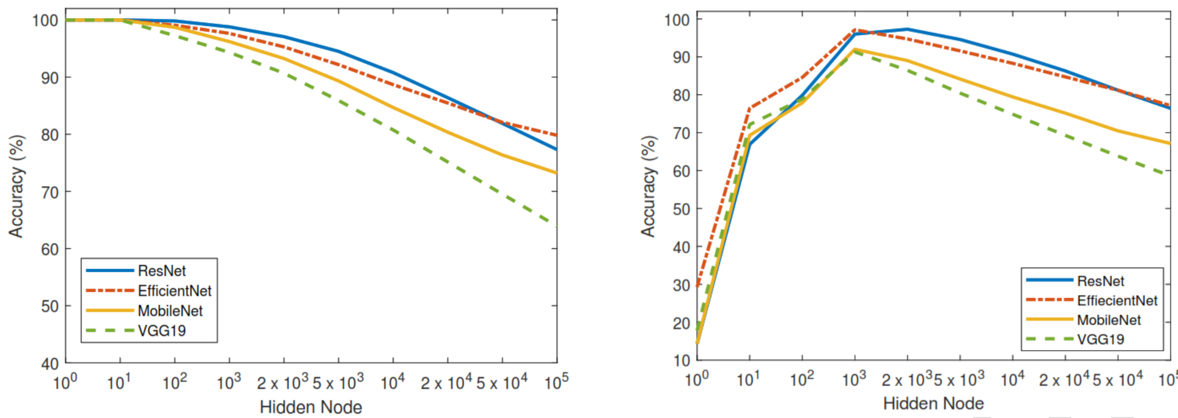
$$F-measure_i = 2 \times \frac{Recall_i \times Precision_i}{Recall_i + Precision_i}$$
$$Average\ F-measure = \frac{1}{N}\sum_{i=1}^{N} F-measure_i \tag{12}$$

Furthermore, we evaluated the CNN models' learning parameters. Learning parameters are the number of parameters used in a CNN architecture to produce classification results.

4. Result and Discussion

In this section, we present and discuss the simulation results, which provide valuable insights into the experiment's performance. The training and testing results are presented based on the outcomes obtained during the experiment. Fig. 8 illustrates the training results using features from VGG19, ResNet101, MobileNetv2, and EfficientNetB0. Notably, the number of nodes in the ELM was the primary parameter tuned during the training process.

10.22060/EEJ.2026.24823.5773



(a) RBF (b) Sigmoid

Fig. 8. Training results, where: (a) RBF activation function; (b) Sigmoid activation function

Based on Fig. 8, the graphs present a different trend on RBF and sigmoid activation function. Applying the RBF activation function led to a decline in training accuracy while an increasing number of hidden nodes across four CNN architectures: ResNet101, EfficientNetB0, MobileNetv2, and VGG19. When the number of hidden nodes is less than 10^2, several architectures achieved 100% accuracy. This performance indicates that with a small number of hidden nodes (10 to 10^2), the models are efficient and generalize well to the given task. At this stage, the network complexity is minimal, and overfitting is not a concern, allowing the models to focus on extracting meaningful features from the data. Fig. 8b presents a different trend of training results. As the number of hidden nodes increased, the training accuracy also increased until the number of nodes is between 10^3 and $2x10^3$. The results showed that ResNet101 showed the most robustness attributed to its architectural design, particularly the skip connections that mitigate the vanishing gradient problem and allow the network to maintain efficient learning even in deeper configurations. Similarly, EfficientNetB0 performs almost as well, demonstrating its ability to handle increased network complexity. EfficientNetB0's design philosophy, which balances depth, width, and resolution, enables it to retain accuracy over a wide range of hidden nodes, making it highly scalable.

MobileNetv2 and VGG19 show similar trend in both training graphs. MobileNetv2, optimized for lightweight applications, is inherently designed for computational efficiency rather than handling large-scale network configurations. As a result, it loses accuracy faster when confronted with increasing hidden node counts, underscoring its limitations in more complex scenarios. A similar result was also obtained by VGG19. The VGG19 as an older architecture lacks modern design feature enhancements such as skip connections or advanced attention mechanisms.

The best-accuracy models from ten independent testing runs, each conducted using a different random seed, are presented as confusion matrices in Figs. 9 and 10. The model selected for testing was based on the most optimum number of hidden nodes obtained during training. The model selected for testing is based on the most optimum number of hidden nodes during training. Fig. 9 presents the confusion matrix of the performance of a degradation classification model for ancient documents using the RBF activation function. Figs. 9a, 9b, 9c, and 9d represent the performance of VGG19, ResNet101, MobileNetv2, and EfficientNetB0, respectively.

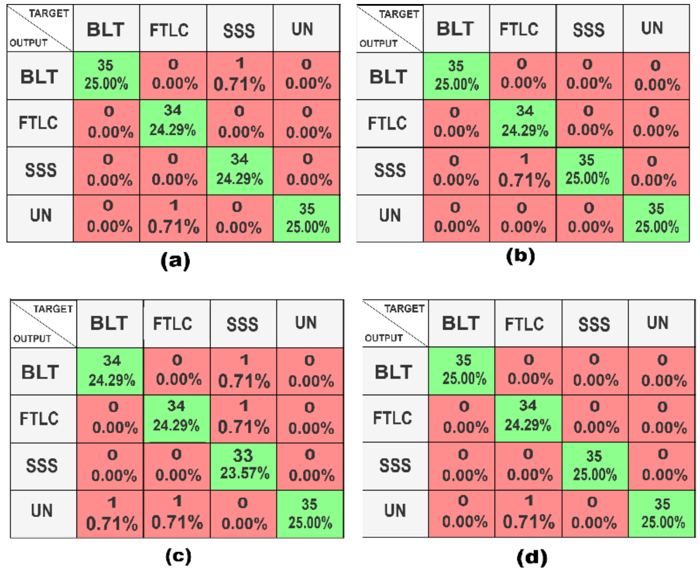


Fig. 9. Confusion matrix of RBF activation, where: (a) VGG19 features, (b) ResNet101 features, (c) MobileNetv2 features, (d) EfficientNetB0 features

Notably, Figs. 9b and 9d show that ResNet101 and EfficientNetB0 achieve 99.29% accuracy, with most types of degradation being correctly identified. However, one image from each model was misclassified. Specifically, an image that should have been identified as FTLC was instead classified as a smear spot or stain degradation using ResNet101 and classified as a uniform degradation by EfficientNetB0. This error may have occurred because these two degradation types share similar features in document images whereas other degradations have distinct, easily separable features that the model can capture effectively.

Fig. 10 shows the confusion matrix of testing results for the models trained using the sigmoid activation function. VGG19, ResNet101, and EfficientNetB0 achieved better classification results with an accuracy of 98.57%. VGG19 demonstrated consistent performance, whether using RBF or sigmoid activation. On the other hand, ResNet101 and EfficientNetB0 experienced a decrease in performance compared to when using the RBF activation function. However, ResNet101 and EfficientNetB0 remained the best performers.

10.22060/EEJ.2026.24823.5773

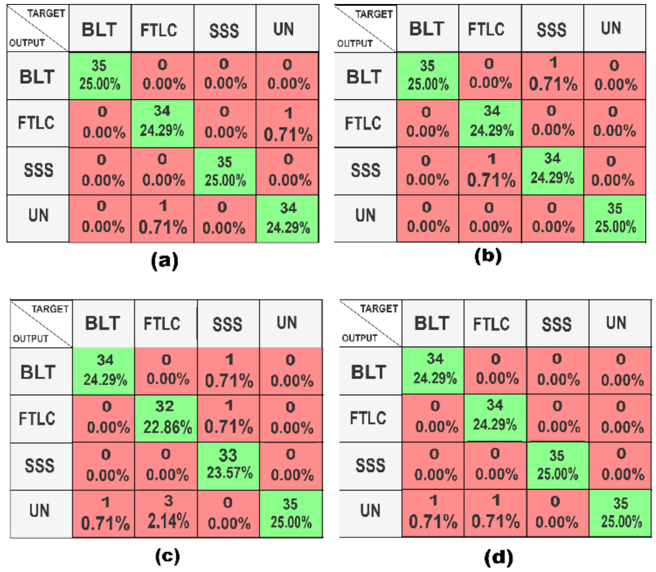


Fig. 10. Confusion matrix of Sigmoid activation, while: (a) VGG19 features, (b) ResNet101 features, (c) MobileNetv2 features, (d) EfficientNetB0 features

Although their performance was not as strong as that of ResNet101 and EfficientNetB0, VGG19 and MobileNetv2 still demonstrated satisfactory results. The high accuracy achieved by these models holds practical significance in the field of document preservation and restoration. Reliable identification of document degradation types can streamline the restoration process by providing accurate diagnoses of defects. In cases such as spot, smear, stain and bleed-through, the classification ensures that targeted restoration techniques can be applied, preserving the authenticity and quality of ancient manuscripts.

The results indicate that the ResNet101 and EfficientNetB0 models have successfully learned to differentiate between the four distinct types of defects. The models are showing promising performance in the patch-level test scenario. This performance has practical implications, particularly for industries that rely on accurate degradation identification, such as printing, document analysis, and packaging.

Table 1 presents the performance comparison of CNN pre-trained models as feature extractors and two different activation functions: RBF and Sigmoid. The results are focused on key performance metrics such as accuracy, average recall, average precision, average F-measure (FM), and the number of learning parameters that is represent the number of learning parameter from CNN-backbone.

Table 1. Degradation classification performance on different CNN pre-trained model. The reported parameter count refers to the CNN backbone used for feature extraction. The CNN weights were not updated during ELM training.

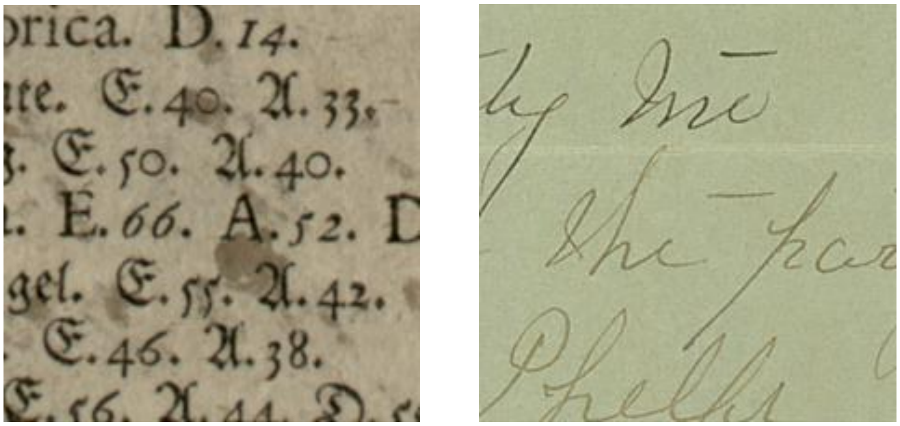
			RBF		
Feature Extractor	**Accuracy**	**Average Recall**	**Average Precision**	**Average FM**	**CNN-backbone Parameters**
VGG19	96.14% ± 4.80%	96.14% ± 4.80%	96.25% ± 4.72%	96.39% ± 4.76%	144 M
ResNet101	98.21% ± 2.43%	98.21% ± 2.43%	98.19% ± 2.39%	98.17% ± 2.41%	44.6 M
MobileNetv2	94.71% ± 5.50%	94.71% ± 5.50%	94.80% ± 5.51%	94.75% ± 5.51%	3.5 M
EfficientNetB0	98.21% ± 2.08%	98.21% ± 2.08%	98.26% ± 2.04%	98.24% ± 2.06%	**5.3 M**
			Sigmoid		
Feature Extractor	**Accuracy**	**Average Recall**	**Average Precision**	**Average FM**	**CNN-backbone Parameters**
VGG19	95.36% ± 6.04%	95.36% ± 6.04%	95.49% ± 5.97%	95.42% ± 6.01%	144 M
ResNet101	96.14%± 3.97%	96.14%± 3.97%	96.22 %± 3.92%	96.18 %± 3.94%	44.6 M
MobileNetv2	94.21%± 5.01%	94.21%± 5.01%	94.35 %± 5.05%	94.28 %± 5.03%	3.5 M
EfficientNetB0	96.14% ± 3.92%	96.14% ± 3.92%	96.24 %± 3.84%	96.19 %± 3.88%	5.3 M

The models trained with the RBF activation function obtained impressive performance, particularly EfficientNetB0 and ResNet101. The highest accuracy of 98.21% was achieved by both EfficientNetB0 and ResNet101, with EfficientNetB0 slightly ahead on the remaining metrics. EfficientNetB0 reached 98.21% recall, 98.26% precision, and 98.24% F-measure, while ResNet101 reached 98.21% recall, 98.19% precision, and 98.17% F-measure. The lower standard deviation obtained by EfficientNetB0 also indicates a more stable performance across the cross-validation folds. This indicates that these models show a good ability to identify and classify different types of document degradation correctly. VGG19 performed slightly lower with 96.14% accuracy, showing values of 96.14% recall, 96.25% precision, and 96.39% F-measure. This consistent performance across different evaluation measures highlights the reliability of VGG19 in classifying degradation types despite having a significantly larger number of parameters (144 million), although such complexity does not translate into a clear advantage over the lighter models. On the other hand, MobileNetv2, which is designed for efficiency with only 3.5 million parameters, achieved 94.71% accuracy with recall, precision, and F-measure of 94.71%, 94.80%, and 94.75%, respectively. While MobileNetv2 may not perform as well as the other models in terms of accuracy and classification metrics, its low number of parameters makes it a highly efficient model, ideal for deployment in resource-constrained environments.

Using the Sigmoid activation function, the performance of the models slightly changes. The highest accuracy of 96.14% was obtained by both ResNet101 and EfficientNetB0, with EfficientNetB0 reaching 96.14% recall, 96.24% precision, and 96.19% F-measure, and ResNet101 reaching 96.14% recall, 96.22% precision, and 96.18% F-measure. This suggests that switching from RBF to Sigmoid leads to a noticeable decline in performance for both models. Similarly, VGG19 also decreased to 95.36% accuracy, with 95.36% recall, 95.49% precision, and 95.42% F-measure. However, MobileNetv2 showed the lowest performance with the Sigmoid function, with accuracy dropping to 94.21% and recall, precision, and F-measure values of 94.21%, 94.35%, and 94.28%, respectively.

Beyond the mean values, the standard deviation provides further insight into the stability of each model. Under the RBF activation function, the standard deviations of the feature extractors remained relatively low, ranging from ± 2.04% to ± 2.43% for EfficientNetB0 and ResNet101, which indicates that their high accuracy was obtained consistently rather than being driven by a few favorable tests. In contrast, the Sigmoid activation function was generally accompanied by larger standard deviations, with values rising to ± 3.92% for EfficientNetB0, ± 3.97% for ResNet101, and ± 6.04% for VGG19. This widening of the spread suggests that the feature representations are separated less reliably under the Sigmoid activation, so that performance becomes more dependent on the particular data partition. Therefore, the RBF activation function is shown to offer not only higher average accuracy but also better consistency.

10.22060/EEJ.2026.24823.5773



(a) (b)

Fig. 11. Misclassification examples on document degradation type using EfficientNetB0–ELM: (a) smear-spot-stain classified as uniform under sigmoid activation; (b) faint-text low-contrast classified as uniform under both sigmoid and RBF activations.

In terms of computational efficiency, MobileNetv2 stands out due to its low number of learnable parameters (3.5 million), making it highly efficient and suitable for environments where computational resources are limited, such as edge devices or mobile platforms. Despite its lower performance in comparison to the more complex models, MobileNetv2 still offers a favorable trade-off between efficiency and accuracy. On the other hand, VGG19 has the highest number of learnable parameters (144 million), which represents a more complex model with higher computational demands. Although this does not impede its performance, it may limit its suitability in resource-constrained environments. ResNet101 and EfficientNetB0, with 44.6 million and 5.3 million parameters, respectively, offer a balance between performance and computational efficiency. Particularly, EfficientNetB0 provides a notable advantage, achieving similar performance to ResNet101 with fewer parameters, making it a highly efficient model without sacrificing much accuracy.

Fig. 11 presents representative misclassification cases using by the EfficientNetB0–ELM model. With the sigmoid activation function, one image from the smear-spot-stain class and one image from the faint-text low-contrast class were incorrectly classified as uniform. In contrast, with the RBF activation function, only the sample in Fig. 11b was misclassified, while the sample in Fig. 11a was correctly classified. These results caused by the visual similarity between the degraded samples and the uniform class, particularly in Fig. 11b. In this case, the text appears faint and is weakly distinguishable from the background, resulting in very low image contrast. Consequently, the model tends to interpret the image as having a uniform degradation pattern.

Table 2. Comparison with previous methods. ResNet101+ELM and EfficientNetB0+ELM were selected as the best-performing models obtained across the ten testing simulations.

Feature Extractor	Accuracy	Inference times	CNN-backbone Parameters
DCNet [7]	100%	2.8 s	18.6 M
Random forest [4]	70.17%	1.62 s	-
VDQAM [5]	78.00%	1.06 s	-
ResNet101+ELM	99.29%	1.27 s	44.6 M
EfficientNetB0 + ELM	99.29%	0.95 s	5.3 M

Table 2 presents performance metrics for several degradation classification models. The evaluation includes key parameters such as accuracy, inference time, and the number of learning parameters. Models compared include DCNet, Random Forest, VDQAM, ResNet101+ELM, and EfficientNetB0+ELM. These models represent a range of machine learning approaches, from traditional algorithms like Random Forest to deep learning-based models enhanced with techniques such as Extreme Learning Machine (ELM). Based on the results, DCNet achieved 100% accuracy, which is an impressive result. However, this performance comes with an inference time of 2.8 seconds, which is relatively slow compared to the other comparison models. DCNet has 18.6 million learning parameters, which reflect its complexity while resulting in a high accuracy but increased computational cost. This trade-off between accuracy and inference speed is a crucial consideration in real-time applications, where faster responses may be required.

In comparison, Random Forest and VDQAM show lower accuracy rates, with 70.17% and 78.00%, respectively. Despite their lower accuracy, these methods offer the advantage of faster inference times, with Random Forest achieving a relatively fast 1.62 seconds and VDQAM performing in 1.06 seconds. However, the absence of learning parameters for Random Forest suggests that it is less complex than the deep learning models, which may explain its lower accuracy. VDQAM, which used SVM as a classifier, achieves better accuracy compared with Random Forest but is still behind the deep learning models, indicating that while it offers reasonable performance, it may not be as effective for complex tasks like document degradation classification.

ResNet101+ELM and EfficientNetB0+ELM offer a good balance between accuracy, inference time, and learning parameters. ResNet101+ELM achieves an inference time of 1.27 seconds with 44.6 million learning parameters, making it a robust model, though it has a relatively higher number of parameters, indicating more complexity. On the other hand, EfficientNetB0+ELM resulted in a highly efficient model, achieving the same best accuracy of 99.29% as ResNet101+ELM, obtained as the highest value across the ten independent runs, but with a significantly lower learning parameter count of 5.3 million. Furthermore, EfficientNetB0+ELM achieves the fastest inference time of just 0.95 seconds, making it the most computationally efficient deep learning model in the comparison. Based on the result, EfficientNetB0+ELM, with its 99.29% accuracy, 5.3 million parameters, and 0.95-second inference time, represents an optimal choice for degradation classification tasks, balancing the need for high accuracy with fast, efficient processing. A limitation of this evaluation is that the reported accuracy reflects patch-level rather than document-level classification. As the 700 patches were cropped from only 100 source documents, patches from the same document may share similar background, writing style, or scan characteristics, which should be considered when interpreting the results in the context of document-level generalization.

5. Conclusion

In this paper, we studied a combination of Convolutional Neural Networks (CNN) and Extreme Learning Machines (ELM) for degradation classification on ancient document images. CNN was used as a feature extractor, and ELM was used as a classifier. We used four CNN benchmark models to extract the feature,

10.22060/EEJ.2026.24823.5773

including VGG19, ResNet101, MobileNetv2, and EfficientNetB0. The combination of the EfficientNetB0 features combined with the ELM classifier achieved the best performance based on accuracy, the number of learning parameters, and inference time. The results have important implications for the design of feature extractors for degradation classification and suggest that a smaller number of learnable parameters can be sufficient to achieve high accuracy in degradation classification. Future work should consider document-level splitting and evaluate the method on larger, more diverse manuscript collections to ensure its robustness in real-world applications. Exploring additional image defects or environmental factors (e.g., lighting conditions, image noise) could provide further insights into the model's adaptability. Additionally, testing the model's performance across different classification architectures would reveal whether this accuracy is tied to a particular model structure. Explore how different lighting conditions, image resolution, and noise affect classification performance in real-world scenarios.

6. Acknowledgement

This research is funded by Universitas Syiah Kuala under Penelitian Lektor grant with No. 326/UN.11/PNBP/2024.

References

[1] W. Xiong, X. Jia, J. Xu, Z. Xiong, M. Liu, J. Wang, Historical document image binarization using background estimation and energy minimization, in: 2018 24th International Conference on Pattern Recognition (ICPR), IEEE, 2018, pp. 3716–3721.
[2] Z. Yang, S. Zuo, Y. Zhou, J. He, J. Shi, A review of document binarization: main techniques, new challenges, and trends, Electronics, 13(7) (2024) 1394.
[3] K. Endo, M. Tanaka, M. Okutomi, CNN-based classification of degraded images with awareness of degradation levels, IEEE Transactions on Circuits and Systems for Video Technology, 31(10) (2020) 4046–4057.
[4] A. Shahkolaei, H.Z. Nafchi, S. Al-Maadeed, M. Cheriet, Subjective and objective quality assessment of degraded document images, Journal of Cultural Heritage, 30 (2018) 199–209.
[5] A. Shahkolaei, A. Beghdadi, M. Cheriet, Blind quality assessment metric and degradation classification for degraded document images, Signal Processing: Image Communication, 76 (2019) 11–21.
[6] R.D. Lins, S. Banergee, M. Thielo, Automatically detecting and classifying noises in document images, in: proceedings of the 2010 ACM symposium on applied computing, 2010, pp. 33–39.
[7] F. Arnia, K. Saddami, K. Munadi, Dcnet: Noise-robust convolutional neural networks for degradation classification on ancient documents, Journal of Imaging, 7(7) (2021) 114.
[8] J. Tang, C. Deng, G.-B. Huang, Extreme learning machine for multilayer perceptron, IEEE transactions on neural networks and learning systems, 27(4) (2015) 809–821.
[9] J. Wang, S. Lu, S.-H. Wang, Y.-D. Zhang, A review on extreme learning machine, Multimedia Tools and Applications, 81(29) (2022) 41611–41660.
[10] G. Huang, G.-B. Huang, S. Song, K. You, Trends in extreme learning machines: A review, Neural Networks, 61 (2015) 32–48.
[11] Y. Qing, Y. Zeng, Y. Li, G.-B. Huang, Deep and wide feature based extreme learning machine for image classification, Neurocomputing, 412 (2020) 426–436.
[12] R. Kaur, R.K. Roul, S. Batra, A hybrid deep learning CNN-ELM approach for parking space detection in Smart Cities, Neural Computing and Applications, 35(18) (2023) 13665–13683.
[13] M. Jain, W. Andreopoulos, M. Stamp, Convolutional neural networks and extreme learning machines for malware classification, Journal of Computer Virology and Hacking Techniques, 16(3) (2020) 229–244.

[14] Y. Wang, Y. Cao, Z.-J. Zha, J. Zhang, Z. Xiong, Deep degradation prior for low-quality image classification, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 11049–11058.
[15] K. Saddami, K. Munadi, F. Arnia, Degradation classification on ancient document image based on deep neural networks, in: 2020 3rd international conference on information and communications technology (ICOIACT), IEEE, 2020, pp. 405–410.
[16] S. Zhou, B. Tan, Electrocardiogram soft computing using hybrid deep learning CNN-ELM, Applied soft computing, 86 (2020) 105778.
[17] S. Ali, J. Li, Y. Pei, M.S. Aslam, Z. Shaukat, M. Azeem, An effective and improved CNN-ELM classifier for handwritten digits recognition and classification, Symmetry, 12(10) (2020) 1742.
[18] F. Arnia, K. Saddami, R. Roslidar, R. Muharar, K. Munadi, Towards accurate Diabetic Foot Ulcer image classification: Leveraging CNN pre-trained features and extreme learning machine, Smart Health, 33 (2024) 100502.
[19] L. Alzubaidi, J. Zhang, A.J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaría, M.A. Fadhel, M. Al-Amidie, L. Farhan, Review of deep learning: concepts, CNN architectures, challenges, applications, future directions, Journal of big Data, 8(1) (2021) 53.
[20] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv:1409.1556, (2014).
[21] A. Naseer, K. Zafar, Meta-feature based few-shot Siamese learning for Urdu optical character recognition, Computational Intelligence, 38(5) (2022) 1707–1727.
[22] M.M. Khayyat, L.A. Elrefaei, M.M. Khayyat, Historical Arabic Images Classification and Retrieval Using Siamese Deep Learning Model, Computers, Materials & Continua, 72(1) (2022).
[23] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.
[24] S.A. Siddiqui, A. Dengel, S. Ahmed, Self-supervised representation learning for document image classification, IEEE Access, 9 (2021) 164358–164367.
[25] M. Tang, S. Xie, X. Liu, Ancient character recognition: a novel image dataset of Shui manuscript characters and classification model, Chinese journal of electronics, 32(1) (2023) 64–75.
[26] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L.-C. Chen, Mobilenetv2: Inverted residuals and linear bottlenecks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 4510–4520.
[27] M. Almanea, Deep learning in written arabic linguistic studies: A comprehensive survey, IEEE Access, 12 (2024) 172196–172233.
[28] Ö. Arslan, S.A. Uymaz, Classification of invoice images by using convolutional neural networks, Journal of Advanced Research in Natural and Applied Sciences, 8(1) (2022) 8–25.
[29] M. Tan, Q. Le, Efficientnet: Rethinking model scaling for convolutional neural networks, in: International conference on machine learning, PMLR, 2019, pp. 6105–6114.
[30] S. Kanchi, A. Pagani, H. Mokayed, M. Liwicki, D. Stricker, M.Z. Afzal, EmmDocClassifier: Efficient multimodal document image classifier for scarce data, Applied Sciences, 12(3) (2022) 1457.
[31] J. Liu, X. Liu, C. Liu, B.T. Le, D. Xiao, Pruning multilayered elm using cholesky factorization and givens rotation transformation, Mathematical Problems in Engineering, 2021(1) (2021) 5588426.
[32] C. Junyue, D.Q. Zeebaree, C. Qingfeng, D.A. Zebari, Breast cancer diagnosis using hybrid AlexNet-ELM and chimp optimization algorithm evolved by Nelder-mead simplex approach, Biomedical Signal Processing and Control, 85 (2023) 105053.
[33] B. Gatos, K. Ntirogiannis, I. Pratikakis, ICDAR 2009 document image binarization contest (DIBCO 2009), in: 2009 10th International conference on document analysis and recognition, IEEE, 2009, pp. 1375–1382.
[34] I. Pratikakis, B. Gatos, K. Ntirogiannis, H-DIBCO 2010-handwritten document image binarization competition, in: 2010 12th international conference on frontiers in handwriting recognition, IEEE, 2010, pp. 727–732.

[35] I. Pratikakis, B. Gatos, K. Ntirogiannis, ICDAR 2011 Document Image Binarization Contest (DIBCO 2011), in: 2011 International Conference on Document Analysis and Recognition (ICDAR), IEEE, Beijing, China, 2011, pp. 1506–1510.
[36] I. Pratikakis, B. Gatos, K. Ntirogiannis, ICFHR 2012 competition on handwritten document image binarization (H-DIBCO 2012), in: 2012 international conference on frontiers in handwriting recognition, IEEE, 2012, pp. 817–822.
[37] S.M. Ayatollahi, H.Z. Nafchi, Persian heritage image binarization competition (PHIBC 2012), in: 2013 first Iranian conference on pattern recognition and image analysis (PRIA), IEEE, 2013, pp. 1–4.
[38] I. Pratikakis, B. Gatos, K. Ntirogiannis, ICDAR 2013 document image binarization contest (DIBCO 2013), in: 2013 12th international conference on document analysis and recognition, IEEE, 2013, pp. 1471–1476.
[39] K. Ntirogiannis, B. Gatos, I. Pratikakis, ICFHR2014 competition on handwritten document image binarization (H-DIBCO 2014), in: 2014 14th International conference on frontiers in handwriting recognition, IEEE, 2014, pp. 809–813.
[40] I. Pratikakis, K. Zagoris, G. Barlas, B. Gatos, ICFHR2016 handwritten document image binarization contest (H-DIBCO 2016), in: 2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR), IEEE, 2016, pp. 619–623.
[41] I. Pratikakis, K. Zagoris, G. Barlas, B. Gatos, ICDAR2017 competition on document image binarization (DIBCO 2017), in: 2017 14Th IAPR international conference on document analysis and recognition (ICDAR), IEEE, 2017, pp. 1395–1403.
[42] K. Saddami, K. Munadi, S. Muchallil, F. Arnia, Improved thresholding method for enhancing Jawi binarization performance, in: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), IEEE, 2017, pp. 1108–1113.