

A LIGHTWEIGHT CNN MODEL FOR BINARY CLASSIFICATION OF FINGERPRINT IMAGES

Jelena Dencic
Marija Zahar Djordjevic¹
Tijana Petrovic

Received 20.04.2026.
Revised 15.07.2026.
Accepted 03.08.2026.

Keywords:

Convolutional Neural Networks,
Deep Learning, Biometric Systems,
Image Classification, Fingerprint
Classification, SOCOFing

ABSTRACT

Fingerprint recognition is one of the most widely used biometric identification techniques due to its uniqueness and long-term stability. However, intentional fingerprint alterations can significantly reduce the reliability of automatic recognition systems, creating a need for robust classification methods capable of distinguishing between genuine and altered fingerprint images. This paper presents a lightweight Convolutional Neural Network model for binary fingerprint classification. The proposed model consists of four convolutional layers with progressively increasing numbers of filters, combined with batch normalization, ReLU activation, max-pooling, global average pooling, and fully connected layers, providing a balance between classification performance and computational complexity. The study also examines the influence of class weighting and input image resolution on model performance. Experimental results demonstrate that the proposed model achieved a classification accuracy of 94.3%, with high precision, recall, and F1-score values. In addition, the results indicate that the class weighting technique significantly improves the classification of minority-class samples, whereas increasing the input image resolution does not substantially affect the overall classification performance. These findings suggest that the proposed lightweight CNN architecture can provide reliable performance for binary fingerprint classification while maintaining relatively low computational requirements. This indicates the potential of the proposed model for biometric applications and possible applicability within industrial systems, particularly in secure access control and authentication scenarios where computational efficiency is an important consideration.



© 2026 Published by Faculty of Engineering

1. INTRODUCTION

Biometric systems represent an essential component of modern security and identification systems. Unlike traditional authentication methods, such as passwords or identification cards, biometric methods rely on unique physiological or behavioral characteristics of an

individual. Among the most commonly used biometric traits are fingerprints, facial recognition, iris recognition, and voice recognition. Fingerprints are considered one of the oldest and most reliable biometric modalities due to their high degree of uniqueness and long-term stability (Cherrat et al., 2020).

¹ Corresponding author: Marija Zahar Djordjevic
Email: maja_199@yahoo.com

With the rapid development of digital technologies and automatic identification systems, fingerprint analysis and classification have become important research areas in machine learning. Traditional fingerprint recognition approaches are primarily based on the extraction and analysis of distinctive fingerprint features. However, these methods are often sensitive to variations in image quality, and intentional fingerprint alterations. Consequently, deep learning techniques have gained increasing attention in recent years because they enable the automatic extraction of relevant features directly from images (Gu et al., 2018; Kanth Gudala et al., 2025; Kumar et al., 2024).

Convolutional Neural Networks (CNNs) are among the most significant deep learning techniques used for image analysis and classification tasks. These networks are especially well suited for processing visual data, as they can efficiently identify spatial patterns and local image features. Owing to their architecture, which consists of convolutional layers, pooling layers, and fully connected layers, CNN models have achieved outstanding performance in various computer vision applications, including object recognition (Gu et al., 2018; Manda et al., 2021), image classification (Zhang et al., 2013), and medical image analysis (Yamashita et al., 2018). According to recent survey studies, CNNs represent one of the most successful deep learning techniques for visual data analysis (Li et al., 2021).

In recent years, CNN-based models have also been increasingly adopted in biometric systems, including fingerprint and face recognition applications (Singhal et al., 2025). Their primary advantage, as suggested by Loganathan and Latha (2025), lies in the ability to automatically learn discriminative features directly from data without requiring manually designed descriptors. This approach provides greater robustness to image noise, variations in image quality, and different types of fingerprint deformations. Furthermore, some deep learning architectures such as AlexNet, VGG, ZFNet, GoogLeNet, and Xception, have significantly improved the performance of image classification systems (Yamashita et al., 2018; Saleem et al., 2022; Swapna et al., 2020; Knapcikova et al., 2024; Lata et al., 2025).

Although numerous CNN-based approaches have been proposed for fingerprint alteration detection, many of them rely on deep and computationally expensive architectures. Consequently, there is still a need for lightweight CNN models capable of achieving competitive performance while reducing computational complexity. This is particularly relevant in industrial environments, where biometric authentication can be used to control access to Industrial IoT devices and critical infrastructure (Bedari et al., 2022; Kalunga et al., 2020; Saipidinov et al., 2025). In such applications, a lightweight model capable of detecting altered fingerprints could serve as an additional security layer

before the subsequent identity-matching and authorization stages.

The objective of this study is to develop and evaluate a lightweight CNN-based model for the binary classification of fingerprints into Real and Altered categories using the publicly available SOCOFing dataset. Also, this paper investigates the influence of class weighting on imbalanced fingerprint datasets and evaluates the effect of different input image resolutions. Furthermore, this paper presents and discusses a comparative evaluation of widely adopted CNN architectures, data preprocessing methodology, the proposed network architecture, training strategy, and classification performance achieved.

2. LITERATURE REVIEW

Traditional fingerprint recognition systems primarily relied on minutiae-based matching algorithms, in which fingerprints were compared using the spatial locations and orientations of detected minutiae (Maltoni et al., 2009). As fingerprint alteration became an increasing challenge in biometric systems, Yoon et al. (2012) proposed automatic methods for the detection of altered fingerprints. Their approach analyzed ridge orientation discontinuities and minutiae distribution to identify common alteration types, including obliteration, distortion, and imitation. Although the proposed method demonstrated promising performance, it exhibited several important limitations. Detection accuracy decreased when only small fingerprint regions were altered or when fingerprints contained imitation-type alterations, as the ridge orientation and minutiae distribution remained similar to those of natural fingerprints. In addition, poor image quality frequently resulted in incorrect feature extraction and an increased number of false positive detections (Ali et al., 2025).

To address the shortage of publicly available altered fingerprint databases, Papi et al. (2016) proposed the STRANGE (SynThetic fingerRprint AlteratioNs GEnerator) tool for generating synthetic fingerprint alterations. The generated synthetic samples facilitated the development and evaluation of fingerprint alteration detection algorithms, although the reliance on synthetic data highlighted the need for publicly available datasets containing real altered fingerprint images.

Building upon this idea, Shehu et al. (2018) presented the publicly available SOCOFing dataset, which contains real and synthetically altered fingerprint images and provides a valuable benchmark for developing and evaluating modern fingerprint classification and alteration detection methods. Hameed et al. (2023) used this dataset for the Real/Altered classification problem from a conventional machine-learning perspective, combining image enhancement and handcrafted HOG and SFTA features with discriminant-analysis classifiers, and achieving classification accuracy of up to 99%.

In recent years, fingerprint analysis has increasingly shifted toward CNN-based approaches, which automatically learn hierarchical feature representations directly from image data instead of relying on manually engineered descriptors (Yamashita et al., 2018).

2.1 Convolutional Neural Networks (CNNs)

CNNs have become one of the most representative deep learning architectures due to their remarkable success in computer vision applications (Li et al., 2021; Krichen, 2023). Their applications include image classification (Krizhevsky et al., 2012), object detection (Dhillon & Verma, 2020) and medical image analysis (Abdou, 2022). Image classification aims to assign input images to predefined categories and has been significantly advanced by deep CNN architectures (Krizhevsky et al., 2012). Object detection extends this task by identifying objects within an image while simultaneously determining their spatial locations (Dhillon & Verma, 2020). In medical image analysis, CNNs have been successfully applied to image classification, detection, localization, segmentation, and computer-aided diagnosis, improving the accuracy and efficiency of medical image interpretation (Abdou, 2022). Their ability to automatically extract relevant features from input data enables efficient pattern recognition in complex data structures (Perišić & Jovanović, 2023).

CNNs were originally designed for processing two-dimensional data, such as digital images. However, recent studies have demonstrated that these networks can also be successfully applied to three-dimensional data, further extending their range of applications (Wang, 2023).

A typical CNN architecture consists of one or more convolutional layers responsible for feature extraction, followed by one or more fully connected (FC) layers that perform the final classification (Lopez Pinaya et al., 2020; Ghosh et al., 2019). Although certain components of CNNs are similar to those of traditional multilayer neural networks, Gu et al. (2018) noted that the key difference lies in the use of convolution operations, which enable efficient processing of spatial data.

Over the past decade, numerous CNN architectures have been proposed, introducing architectural improvements aimed at increasing classification accuracy, improving feature extraction, and enhancing computational efficiency (Li et al., 2021). The development of Convolutional Neural Networks accelerated significantly after deep learning models achieved remarkable success in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC), a large-scale computer vision competition (Zhao et al., 2024).

One of the most influential architectures in the early development of CNNs is AlexNet, introduced in 2012 by Krizhevsky et al., achieving remarkable success in

the ILSVRC competition and demonstrating that convolutional architectures could significantly outperform previous approaches in image classification tasks. The AlexNet architecture consists of five convolutional layers followed by max pooling operations, fully connected layers, and a softmax output layer. It employs the ReLU activation function and the dropout technique to reduce overfitting, while containing approximately 60 million trainable parameters (Krizhevsky et al., 2012). Despite its groundbreaking performance, AlexNet contains a large number of trainable parameters, resulting in high computational complexity and substantial memory requirements during training (Li et al., 2021).

Following the introduction of AlexNet, Zeiler and Fergus (2014) proposed the architecture commonly known as ZFNet, which introduced several modifications to the AlexNet design. While the overall network structure remained largely unchanged, several hyperparameters were modified to enhance feature extraction. The most significant improvements included reducing the filter size in the first convolutional layer and decreasing the filter stride. These modifications enabled better preservation of spatial information in the early layers of the network, resulting in improved image classification performance (Zeiler & Fergus, 2014; Liu et al., 2026). Although these modifications improved feature extraction and classification accuracy, the architecture still relied on a relatively limited number of convolutional layers, leaving room for further improvements through deeper network designs.

Another significant milestone in the evolution of CNN architectures is VGGNet, introduced in 2014 by Simonyan and Zisserman. The defining characteristic of this architecture is its considerable network depth combined with a simple and homogeneous design. Instead of using large convolutional filters, VGGNet employs exclusively 3×3 convolutional kernels arranged in multiple consecutive layers. This design enables more effective extraction of complex features while maintaining a relatively simple network architecture (Simonyan & Zisserman, 2015).

Several variants of this model have been proposed, following the original model, including VGG-11, VGG-16, and VGG-19, which differ primarily in the number of layers. Despite its excellent classification performance, VGGNet is characterized by a very large number of trainable parameters, resulting in high computational cost, increased memory requirements, and long training times (Li et al., 2021).

The mentioned architectures significantly advanced image classification, but their high computational complexity motivated the development of other CNN models tailored to specific applications, such as fingerprint classification.

Cherrat et al. (2020) proposed that CNNs are specifically suitable for biometric image analysis because they can automatically extract discriminative features such as edges, lines, and texture patterns. In fingerprint images, these features correspond to ridge patterns, which provide essential information for recognition and identification. Due to the hierarchical nature of CNNs, shallow layers learn simple features, whereas deeper layers capture increasingly complex patterns that are more relevant for classification tasks (Kim, 2017). Consequently, CNNs have become widely adopted for fingerprint classification and alteration detection due to their ability to learn discriminative image features and improve robustness to variations in fingerprint appearance (Peralta et al., 2017; Shehu et al., 2018a; Militello et al., 2021; Efrizoni et al., 2024; Mahajan et al., 2026).

Giudice et al. (2020) proposed a CNN Inception-v3-based multi-task approach for altered fingerprint analysis using the SOCOFing dataset. Their model distinguished authentic from altered fingerprints with an accuracy of 98.2% and additionally classified alteration type, gender, hand, and finger with 98.4%, 92.5%, 97.5% and 92.1% accuracies, respectively. Ibitayo et al. (2022) used CNN and SOCOFing dataset for another biometric classification task. Their approach focused on classifying fingerprint images according to gender and achieved an accuracy of 96.5%.

Although CNN-based approaches have demonstrated high performance in fingerprint analysis, previous studies have primarily focused on optimizing network architectures, feature learning, and training strategies to improve classification performance. The available literature provides limited systematic analysis of the effects of class weighting and input image resolution on lightweight CNN-based Real/Altered classification. This gap motivated the presented study to systematically investigate the effects of class weighting and input image resolution within a lightweight CNN framework, with the aim of achieving reliable Real/Altered classification while maintaining a relatively simple model architecture.

3. METHODOLOGY

The proposed methodology consists of dataset preparation, image preprocessing, model development, and performance evaluation.

3.1 Dataset

The experiments presented in this study were conducted using the SOCOFing (Sokoto Coventry Fingerprint Dataset) (Shehu et al., 2018), a publicly available dataset, downloaded from Kaggle (Ruiz-Garcia, n.d.), that is widely used in biometric identification and machine learning research (Singh et al., 2023; Mohammed et al., 2025).

The SOCOFing dataset consists of real and synthetically altered fingerprint images collected from 600 subjects, with ten fingerprint images available for each subject. In addition to the original fingerprint images, the dataset provides synthetic alterations generated using the STRANGE toolbox with three levels of alteration difficulty (easy, medium, and hard). The altered fingerprints are categorized into three alteration types: Central Rotation (CR), Obliteration (Obl), and Z-cut (Zcut), providing a comprehensive benchmark for fingerprint alteration detection and classification.

In this study, the problem was formulated as a binary classification task, where the fingerprint images were divided into two classes:

- Real - original fingerprint images
- Altered - intentionally modified fingerprint images

The dataset used in this study comprises a total of 55270 fingerprint images, including 6000 images from the Real class and 49270 images from the Altered class. This distribution indicates a significant class imbalance, which is a common challenge in machine learning tasks and may adversely affect model performance.

The images in the SOCOFing dataset are stored in BMP format and represented as grayscale images. In addition to the image itself, each filename contains metadata describing the subject's gender, hand orientation, and finger type, facilitating dataset organization and analysis.

Besides its large number of samples, the SOCOFing dataset is primarily suitable for biometric research because it includes several types of fingerprint alterations. These modifications simulate different ways in which fingerprints may be intentionally or unintentionally altered, allowing the robustness of recognition algorithms to be thoroughly evaluated. Consequently, the dataset provides a realistic benchmark for assessing model performance under conditions that closely resemble real-world biometric identification systems.

3.2 Data preprocessing

Before training the proposed model, the dataset underwent a preprocessing stage to ensure a consistent input format, reduce computational complexity, and improve training stability.

The fingerprint images were loaded from the SOCOFing dataset directory, which is organized into separate Real and Altered subdirectories. This directory structure enabled the automatic assignment of class labels based on the file path. After loading the dataset, a data structure was created to store the file path of each image together with its corresponding class label.

In the next step, the dataset was divided into training and validation subsets using an 80:20 split. Specifically, 80% of the data were allocated to the training set, while the remaining 20% were reserved for model validation during training. This approach enables the evaluation of the model's ability to generalize to previously unseen data.

Since the images in the dataset had different dimensions, they were standardized by resizing all images to 96×96 pixels. This ensured a consistent input size, which is required for the proper operation of CNN.

After resizing, the pixel values were normalized to the range $[0, 1]$ by dividing each pixel value by the maximum possible intensity value (255), providing a more stable training process and faster convergence of the optimization algorithm.

To improve data processing efficiency during training, the `tf.data` API provided by the TensorFlow framework was utilized. This approach enables efficient data loading and preprocessing, which is particularly important when working with large datasets such as SOCOFing.

Within the implementation, a function named `make_ds()` was defined to convert the tabular data structure into a TensorFlow Dataset object. First, the image file paths and their corresponding class labels were extracted from the dataset.

These data were then converted into a TensorFlow dataset using the `from_tensor_slices()` function, allowing them to be processed within the `tf.data` pipeline.

For the training dataset, the `shuffle()` operation was applied to randomly reorder the samples. This step prevents the model from being trained on data presented in the same sequence during every epoch, thereby improving its ability to generalize.

Finally, the images were grouped into fixed-size batches and fed to the model during the training process.

3.3 Proposed Architecture

In this study, a lightweight CNN model was employed to classify fingerprint images into two categories: Real and Altered. A custom CNN architecture was developed to achieve a balance between classification accuracy and computational efficiency.

The proposed architecture consists of multiple successive convolutional blocks. The first three blocks comprise a convolutional layer, a max-pooling layer, and a batch normalization layer, while the final convolutional layer is followed by global average pooling. This structure enables the gradual extraction of image features while providing a more stable training process. The convolutional layers are responsible for extracting relevant features from the input images, such as lines, edges, and texture patterns, whereas the pooling layers reduce the dimensionality of the data and the computational complexity of the model.

The application of batch normalization contributes to training stability and accelerates the convergence of the model during training. The combination of these layers enables the network to progressively learn increasingly complex features from the input data, which is especially important in image classification tasks such as fingerprint analysis.

The input to the network is a 96×96 grayscale image with a single channel. The first part of the architecture consists of multiple convolutional layers using 3×3 filters and the ReLU activation function. In the deeper part of the network, the number of filters is gradually increased from 32 to 64, then 128, and finally 256, enabling the extraction of increasingly complex features from the input images. Following the convolutional layers, Global Average Pooling is applied, after which the extracted features are passed to fully connected layers that perform the final prediction. A detailed description of the proposed model architecture is presented in Figure 1 and Table 1.

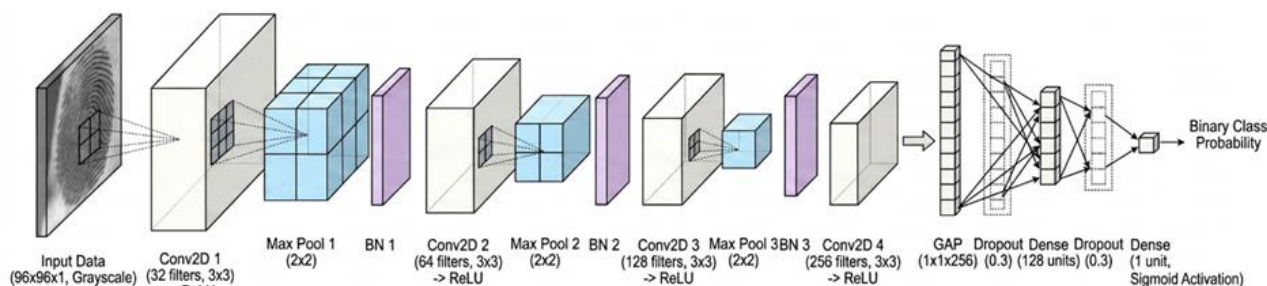


Figure 1. Proposed model architecture

Note: Digitally rendered with the assistance of generative AI, based on technical specifications prepared by the authors

Table 1. Proposed model's layers

Layer	Layer type	Number of filters / neurons	Filter dimension	Output dimension
1	Input Layer			$96 \times 96 \times 1$
2	Conv2D	32	3×3	$96 \times 96 \times 32$
3	MaxPooling2D		2×2	$48 \times 48 \times 32$
4	BatchNormalization			$48 \times 48 \times 32$
5	Conv2D	64	3×3	$48 \times 48 \times 64$
6	MaxPooling2D		2×2	$24 \times 24 \times 64$
7	BatchNormalization			$24 \times 24 \times 64$
8	Conv2D	128	3×3	$24 \times 24 \times 128$
9	MaxPooling2D		2×2	$12 \times 12 \times 128$
10	BatchNormalization			$12 \times 12 \times 128$
11	Conv2D	256	3×3	$12 \times 12 \times 256$
12	GlobalAveragePooling2D			256
13	Dropout			256
14	Dense	128		128
15	Dropout			128
16	Dense (Sigmoid)	1		1

The final part of the network consists of fully connected (Dense) layers that perform the final classification. Between these layers, the Dropout technique is applied, randomly deactivating a portion of the neurons during training to reduce the risk of model overfitting.

The output layer consists of a single neuron with a sigmoid activation function, which produces the probability of an input image belonging to one of the two classes.

The proposed network contains 421,313 trainable parameters, supporting its characterization as a lightweight CNN architecture.

3.4 Training strategy

The proposed fingerprint classification system was implemented in Python using the TensorFlow deep learning framework (Ketkar & Moolayil, 2021). The Keras API was employed to simplify the implementation, training, and evaluation of the neural network.

All experiments were conducted using Google Colab, which provides access to Graphics Processing Units (GPUs) for accelerated deep learning computations. Since CNNs require a large number of mathematical operations, mainly within convolutional layers, GPU acceleration significantly reduced the model training time compared with conventional CPU-based execution.

During training, the model was optimized using the training dataset, while its performance was continuously monitored on the validation dataset. This strategy enabled the assessment of the model's generalization capability while reducing the risk of overfitting.

During model training, several hyperparameters were employed to influence the performance and stability of the learning process. One of the most important hyperparameters is the batch size, which determines the number of samples processed simultaneously during a single training step. In this study, a batch size of 128 was used, representing a compromise between training speed and optimization stability. The hyperparameter configuration is summarized in Table 2.

Table 2. Hyperparameters

Hyperparameters	Value
Input image dimensions	96×96
Number of channels	1
Batch size	128
Number of epochs	12
Optimizer	Adam
Learning rate	0.001
Loss function	Binary Cross-Entropy
Activation function	ReLU
Output activation	Sigmoid
Dropout	0.30
Early Stopping	patience = 3
Dataset split	80 / 20

Model parameters were optimized using the Adam optimizer, which is widely adopted in deep learning applications because of its stability and fast convergence. The Binary Cross-Entropy loss function was employed, as it is the standard choice for binary classification problems.

The network was trained over multiple epochs, where each epoch corresponds to one complete pass through the training dataset. To prevent overfitting and improve generalization, the Early Stopping technique was applied, automatically terminating the training process if the validation performance failed to improve over a predefined number of consecutive epochs.

After training, the model was evaluated on the validation dataset using several performance metrics, including accuracy, precision, recall, and F1-score. In addition, a confusion matrix was analyzed to provide a more detailed assessment of the classification performance for each class.

4. RESULTS AND DISCUSSION

This section presents the experimental results obtained using the proposed lightweight CNN architecture on the SOCOFing dataset. The objective of the experiments was to evaluate the model's ability to accurately classify fingerprint images into two categories: Real and Altered.

4.1 Overall model performance and effect of class weighting

According to the proposed methodology the model was initially trained using the training dataset, while its performance was monitored on the validation dataset. Table 3 presents the classification results obtained without applying any class balancing strategy.

Table 3. Initial classification results

Metric	Value
Accuracy	0.827
Precision	0.976
Recall	0.826
F1-score	0.895

The initial experiment achieved satisfactory overall performance. However, the relatively lower recall

indicated that the class imbalance present in the SOCOFing dataset negatively affected the model's ability to correctly identify samples from the minority class. Since the dataset contains substantially more Altered than Real fingerprint images, an additional experiment was conducted using the class weighting technique during model training (Fernando & Tsokos, 2021; Fabricius et al., 2023). Higher weights were assigned to the minority class, encouraging the model to correctly classify underrepresented samples during training. The classification results obtained after applying the class weighting are presented in Table 4.

Table 4. Classification results after applying class weighting

Metric	Value
Accuracy	0.943
Precision	0.987
Recall	0.949
F1-score	0.967

Using the class weighting technique, the model achieved a classification accuracy of 0.943, compared with 0.827 without class weighting, representing an improvement of 11.6 percentage points. While precision increased from 0.976 to 0.987, Recall increased from 0.826 to 0.949, showing that the new model successfully identified more of the samples belonging to the target class. The F1-score increased from 0.895 to 0.967, demonstrating a substantial improvement in classification performance after applying class weighting. The comparison between the obtained results is shown in Figure 2.

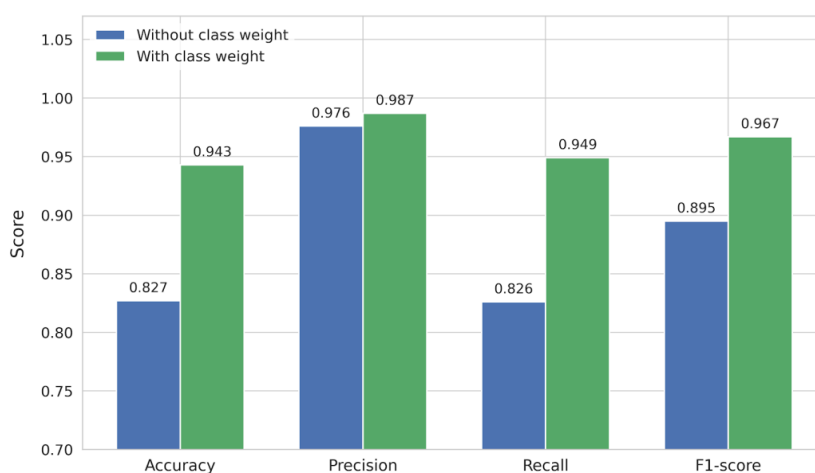


Figure 2. Performance comparison: with vs. without class weight

Based on the obtained results, it can be observed that the model trained using class weighting achieved significantly better performance than the model trained without this technique. The improvement is particularly evident in the recall and F1-score metrics, indicating a better ability of the model to correctly identify samples from both Real and Altered classes.

This result was expected because the SOCOFing dataset exhibits a substantial class imbalance, with the number of Altered fingerprint images considerably exceeding the number of Real samples. Under such conditions, a model trained without class weighting tends to favor the majority class, resulting in a decline in overall classification performance.

Based on this analysis, it can be concluded that the application of the class weighting technique contributes to a better-balanced training process and enables the model to achieve improved performance in the binary classification of fingerprint images.

4.2 Learning dynamics and classification performance

Following the selection of the class-weighted configuration presented in the previous section, the learning dynamics and classification performance of the resulting model were further analyzed. During model training, both the loss function and the AUC metric were monitored on the training and validation datasets.

Figure 3 illustrates the training and validation loss curves, showing how the loss values changed throughout the training process.

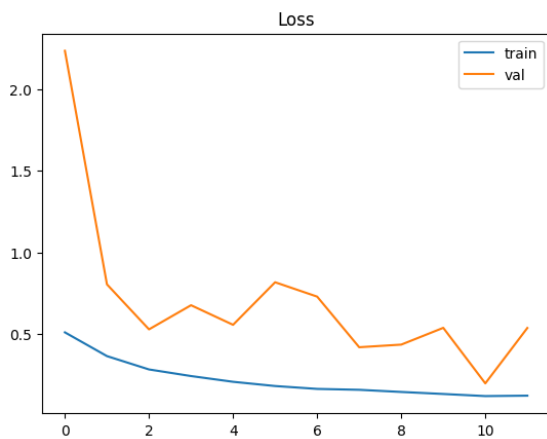


Figure 3. Loss curves

As shown in Figure 3, both training and validation loss decreased steadily and converged without significant divergence, indicating stable learning and good generalization ability.

Figure 4 presents the training and validation AUC curves, which measure the model's ability to distinguish between the two classes (Muschelli, 2020).

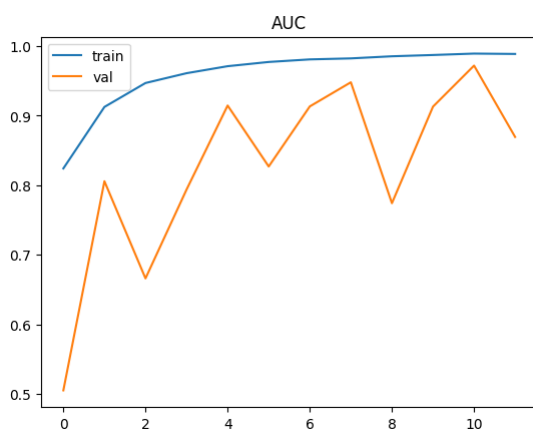


Figure 4. AUC curves

As shown in Figure 4, the AUC values increased during model training and stabilized after a certain number of epochs, indicating that the model gradually learns relevant features from the input data. The high AUC scores further confirm the model's ability to accurately classify Real and Altered fingerprint images.

To gain a more detailed insight into the classification results, a confusion matrix (Heydarian et al., 2022) obtained on the validation dataset, was also analyzed, as shown in Figure 5.

		Predicted	
		Real	Altered
Actual	Real	1075	125
	Altered	506	9348

Figure 5. Confusion matrix

This matrix enables a more detailed analysis of the classification results than accuracy alone, as it shows how the model performs for each individual class. The confusion matrix makes it possible to identify cases in which samples were incorrectly classified, thereby providing a better understanding of the model's limitations.

In addition, it reveals the relationship between correctly and incorrectly classified samples, which is markedly important when working with imbalanced datasets. The analysis of the confusion matrix also provides insight into whether the model tends to favor one class over the other.

The confusion matrix shows that 9348 samples were correctly classified as Altered, and 1075 correctly classified as Real. On the other side, 125 samples were incorrectly classified as Altered, while 506 were incorrectly classified as Real.

The obtained results indicate that the proposed CNN architecture achieves strong performance in the binary classification of fingerprint images. The high values of accuracy, precision, and F1-score (Table 4) demonstrate that the model successfully extracts relevant features from the input images and effectively distinguishes between the two target classes. These findings confirm that CNNs represent an effective approach for biometric data analysis, particularly in tasks involving image processing and classification.

Furthermore, the results indicate that the proposed architecture achieves stable performance despite its relatively simple network design and moderate number of trainable parameters. This demonstrates that competitive classification performance can be achieved without relying on very deep and computationally expensive architectures, which is specifically valuable in practical applications where computational resources are limited.

4.3 Effect of input image resolution

To further investigate the influence of model parameters on classification performance, an additional experiment was conducted to evaluate the effect of input image resolution on the model's performance.

In this experiment, the same network architecture and hyperparameter configuration were retained, including the use of class weighting, while the only modification was increasing the input image resolution from 96×96 pixels to 128×128 pixels.

A comparison of the results obtained from the two experiments is presented in Table 5.

Table 5. Effect of input Image Resolution on model performance

Configuration	Accuracy	Precision	Recall	F1-score
CNN (96×96)	0.943	0.987	0.949	0.967
CNN (128×128)	0.943	0.993	0.942	0.967

Based on the obtained results, it can be observed that increasing the input image resolution did not lead to a significant improvement in the overall performance of the model. The accuracy and F1-score values remained

nearly unchanged, indicating that the proposed CNN architecture was already able to extract relevant features effectively from 96×96 input images.

Although a slightly higher precision value was achieved with the higher image resolution, the recall value decreased marginally, resulting in identical F1-score. These findings suggest that increasing the input image resolution does not necessarily lead to improved model performance, primarily when the network is already capable of extracting discriminative features at a lower resolution.

Furthermore, a higher input resolution increases computational complexity and prolongs the model training time, which may represent a limitation in practical applications. Therefore, it can be concluded that an input resolution of 96×96 pixels was sufficient for successfully performing the binary classification of fingerprint images in this study.

4.4 Architectural comparison with established CNN models

To provide a broader perspective on the proposed architecture, a comparative analysis was conducted with three well-established CNN models, namely AlexNet, ZFNet, and VGGNet (Table 6). These architectures were selected because they represent important milestones in the evolution of CNNs and are widely used as benchmark models in image classification tasks. The analysis shown in Table 6 highlights the architectural differences that motivated the design of the proposed lightweight CNN model.

Table 6. Architectural comparison of the proposed CNN and established CNN models

Characteristic	AlexNet	ZFNet	VGGNet	Proposed CNN
Primary purpose	General image classification	General image classification	General image classification	Binary fingerprint classification
Architecture	5 convolutional + 3 fully connected layers	5 convolutional + 3 fully connected layers	13-16 convolutional + 3 fully connected layers	4 convolutional layers with activation functions, followed by fully connected layers and a sigmoid output
Network depth	Deep	Deep	Very deep	Moderate
Model complexity	High	High	Very high	Low
Computational requirements	Moderate-High	Moderate-High	Very high	Low
Risk of overfitting on small datasets	Moderate	Moderate	High	Low
Feature extraction for fingerprint patterns	General-purpose visual features	General-purpose visual features	Highly generic hierarchical features	Designed to capture fingerprint ridge and texture patterns
Suitability for binary fingerprint classification	Moderate	Moderate	Low	High

As summarized in Table 6, the proposed architecture differs from the compared CNN models in several important aspects. While AlexNet, ZFNet, and VGGNet were originally developed for large-scale image classification tasks, the proposed CNN was specifically

designed for binary fingerprint classification. Consequently, it employs a lightweight architecture with fewer convolutional layers and lower computational complexity than the established models. Rather than increasing network depth, the proposed

architecture gradually increases the number of convolutional filters, enabling the extraction of increasingly discriminative fingerprint features while maintaining a relatively small number of trainable parameters.

As a result, the proposed architecture provides a balanced trade-off between classification performance and computational efficiency, making it a promising candidate for practical biometric applications.

5. CONCLUSION

This paper presented the application of CNNs to the task of binary fingerprint classification. The primary objective of the study was to develop and evaluate a lightweight CNN-based model capable of distinguishing between Real and Altered fingerprint images.

The theoretical part of the paper provided an overview of the fundamental concepts of CNNs. In addition, several influential CNN architectures, such as AlexNet, ZFNet, and VGGNet, were reviewed and compared to the proposed architecture. In the experimental part, the proposed lightweight CNN model was developed, trained, and evaluated using the publicly available SOCOFing dataset. Also, experimental analyses were conducted to investigate the effect of class weighting under the substantial class imbalance present in the dataset and to assess the effect of input image resolution on classification performance.

The experimental results showed that the application of class weighting technique increased classification accuracy from 82.7% to 94.3%, demonstrating the substantial impact of class weighting on classification performance under the highly imbalanced SOCOFing class distribution. The proposed model achieved high performance in the binary fingerprint classification task, while the obtained precision, recall, and F1-score values (98.7%, 94.9% and 96.7% respectively) also indicated a

strong ability of the model to correctly classify the two target classes.

In contrast, increasing the input image resolution from 96×96 to 128×128 pixels did not lead to a substantial improvement in the obtained results, indicating that the proposed architecture was already capable of effectively extracting relevant features at the lower resolution.

Although the obtained results are encouraging, this study has several limitations that provide opportunities for future research. First, the proposed model was evaluated only on the SOCOFing dataset, so its capability could be further validated using additional publicly available fingerprint datasets. Second, the study was limited to binary classification of Real and Altered fingerprints, providing opportunities to extend the proposed approach to more complex tasks, such as identifying specific types of fingerprint alterations or recognizing individual identities based on fingerprint images. Future research may also investigate the application of transfer learning by leveraging pretrained deep learning models, which could reduce training time and potentially improve classification performance.

Despite these limitations, the results indicate that the proposed lightweight CNN provides a reasonable balance between classification performance and architectural simplicity. In particular, the substantial improvement achieved through class weighting highlights the importance of appropriately addressing class imbalance in fingerprint classification tasks, while the resolution analysis suggests that increasing the input image size does not necessarily provide additional performance benefits. The proposed architecture achieved competitive classification performance while maintaining low computational complexity, demonstrating the potential of lightweight deep learning for biometric security applications. In particular, its low-complexity design makes it a promising candidate for supporting secure access control in industrial environments.

References:

- Abdou, M. A. (2022). Literature review: Efficient deep neural networks techniques for medical image analysis. *Neural Computing and Applications*, 34(8), 5791-5812. doi: 10.1007/s00521-022-06960-9
- Ali, M. S., Akram, A., Rashid, J., Jaffar, M. A., Shah, D., Ali, S., & Tahir, M. (2025). Fake fingerprint classification using hybrid features learning with gradient boosting. *Applied Computational Intelligence and Soft Computing*, 2025(1), 8442143. doi: 10.1155/acis/8442143
- Bedari, A., Wang, S., & Yang, W. (2022). A secure online fingerprint authentication system for industrial IoT devices over 5G networks. *Sensors*, 22(19), 7609. doi: 10.3390/s22197609
- Cherrat, E. M., Alaoui, R., & Bouzahir, H. (2020). Convolutional neural networks approach for multimodal biometric identification system using the fusion of fingerprint, finger-vein and face images, *PeerJ Computer Science*, 6, e248. doi: 10.7717/peerj-cs.248
- Dhillon, A., & Verma, G. K. (2020). Convolutional neural network: a review of models, methodologies and applications to object detection. *Progress in Artificial Intelligence*, 9, 85-112. doi: 10.1007/s13748-019-00203-0

- Efrizoni, L., Armoogum, S., & Zakaria, M. Z. (2024). Deep Learning Innovations in Fingerprint Recognition: A Comparative Study of Model Efficiencies. *International Journal of Advances in Artificial Intelligence and Machine Learning*, 1(1), 28-35. doi: 10.58723/ijaauml.v1i1.294
- Fabricius, R., Šuch, O., & Tarabek, P. (2023). Deep neural network ensembles using class-vs-class weighting. *IEEE Access*, 11, 77703-77715. doi: 10.1109/ACCESS.2023.3298057
- Fernando, K. R. M., & Tsokos, C. P. (2021). Dynamically weighted balanced loss: class imbalanced learning and confidence calibration of deep neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 33(7), 2940-2951. doi: 10.1109/TNNLS.2020.3047335
- Ghosh, A., Sufian, A., Sultana, F., Chakrabarti, A., & De, D. (2019). Fundamental concepts of convolutional neural network. In Balas, V., Kumar, R., Srivastava, R. (Eds) *Recent trends and advances in artificial intelligence and Internet of Things* (pp. 519-567). Cham: Springer International Publishing. doi: 10.1007/978-3-030-32644-9_36
- Giudice, O., Litrico, M., & Battiato, S. (2020, October). Single architecture and multiple task deep neural network for altered fingerprint analysis. In 2020 IEEE international conference on image processing (ICIP) (pp. 813-817). IEEE.
- Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., ... & Chen, T. (2018). Recent advances in convolutional neural networks. *Pattern Recognition*, 77, 354-377. doi: 10.48550/arXiv.1512.07108
- Hameed, S. S., Ahmed, I. T., & Al Okashi, O. M. (2023). Real and altered fingerprint classification based on various features and classifiers. *Computers, Materials and Continua*, 74(1), 327-340. doi: 10.32604/cmc.2023.031622
- Heydarian, M., Doyle, T. E., & Samavi, R. (2022). MLCM: Multi-label confusion matrix. *IEEE Access*, 10, 19083-19095. doi: 10.1109/ACCESS.2022.3151048
- Ibitayo, F. B., Olanrewaju, O. A., & Oyeladun, M. B. (2022). A fingerprint based gender detector system using fingerprint pattern analysis. *International Journal of Advanced Research in Computer Science*, 13(4), 35-47. doi: 10.26483/ijarcs.v13i4.6885
- Kalunga, J., Tembo, S., & Phiri, J. (2020). Development of Access Control Mechanism Based on Fingerprint Biometrics and Mobile Phone Identity for Industrial Internet of Things Critical Infrastructure Protection. *International Journal of Advances in Scientific Research and Engineering*, 6(12), 15-36. doi: 10.31695/IJASRE.2020.33940
- Kanth Gudala, R., Kumar Suggala, R., Nagaraju Datla, V., & Krishna Adusumalli, S. (2025). Design of smart tourism systems to forecast foreign tourist arrival rate using deep learning techniques. *Proceedings on Engineering Sciences*, 7(1), 223-232 doi: 10.24874/PES07.01B.001
- Ketkar, N., & Moolayil, J. (2021). Convolutional neural networks. In N. Ketkar & J. Moolayil (Eds.), *Deep learning with Python: Learn best practices of deep learning models with PyTorch* (pp. 197-242). New York, NY: Apress. doi: 10.1007/978-1-4842-5364-9_6
- Kim, P. (2017). Convolutional neural network. In P. Kim (Ed.), *MATLAB deep learning: With machine learning, neural networks and artificial intelligence* (pp. 121-147). doi: 10.1007/978-1-4842-2845-6_6
- Krichen, M. (2023). Convolutional neural networks: A survey. *Computers*, 12(8), 151. doi: 10.3390/computers12080151
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25.
- Knapcikova, L., Sukic, E., Behunova, A., Tauberova, R., Ljajic, S., & Dedic, N. (2024). Digitalization and technology as main tools to assure the company's performance. *International Journal for Quality Research*, 18(4), 953-964. . doi: 10.24874/IJQR18.04-02
- Kumar, N., Agarwal, P., Bansal, S., Yadav, V. K., & Bhowmik, D. (2024). Sentiment analysis using novel deep learning methods. *Proceedings on Engineering Sciences*, 6(2), 853-862. doi: 10.24874/PES06.02A.012
- Lata, S., Sangwan, P., Sheoran, D., & Chaudhary, P. (2025). Improved long-range dependency capture in speech sentiment recognition: a comparative study of mfcc-bilstm and mfcc-lstm. *Proceedings on Engineering Sciences*, 7(3), 1955. doi: 10.24874/PES07.03A.021
- Li, Z., Liu, F., Yang, W., Peng, S., & Zhou, J. (2021). A survey of convolutional neural networks: analysis, applications, and prospects. *IEEE transactions on neural networks and learning systems*, 33(12), 6999-7019. doi: 10.1109/TNNLS.2021.3084827
- Liu, H., Tan, C., Tang, J., & Deng, M. (2026). Improved ZFNet for accurate building shape recognition in map generalization. *Journal of Geovisualization and Spatial Analysis*, 10(1), 8. doi: 10.1007/s41651-026-00249-4
- Loganathan, R., & Latha, S. (2025). The empirical comparison of deep neural network optimizers for binary classification of OCT images. *Proceedings on Engineering Sciences*, 7(1), 547-554. doi: 10.24874/PES07.01D.011

- Lopez Pinaya, W. H., Vieira, S., Garcia-Dias, R., & Mechelli, A. (2020). Convolutional neural networks. In A. Mechelli& S. Vieira (Eds.), *Machine learning* (pp. 173–191). Academic Press. doi: 10.1016/B978-0-12-815739-8.00010-9
- Maltoni, D., Maio, D., Jain, A. K., & Prabhakar, S. (2009). Handbook of fingerprint recognition. London: Springer London.
- Mahajan, A., Siwan, D., Krishan, P., Rana, A., Verma, R., Guleria, A., ... & Krishan, K. (2026). Deep learning-based CNN model for multiclass classification of fingerprint patterns. *Medicine, Science and the Law*, 66(2), 114-121. doi: 10.1177/00258024251355042
- Manda, B., Bhaskare, P., & Muthuganapathy, R. (2021). A convolutional neural network approach to the classification of engineering models. *IEEE Access*, 9, 22711-22723. doi: 10.1109/ACCESS.2021.3055826
- Militello, C., Rundo, L., Vitabile, S., & Conti, V. (2021). Fingerprint classification based on deep learning approaches: experimental findings and comparisons. *Symmetry*, 13(5), 750. doi: 10.3390/sym13050750
- Mohammed, T. J., Chew, X., Khaw, K. W., Albahri, A. S., & Malik Alnoor, A. (2025, December). Lightweight Multi-ML-Based Fingerprint Spoof Detection Framework for Secure Digital Banking Applications with VGG16-Based Feature Extraction. In *International Multi-Disciplinary Conference-Integrated Sciences and Technologies* (pp. 119-141). Cham: Springer Nature Switzerland.
- Muschelli III, J. (2020). ROC and AUC with a binary predictor: a potentially misleading metric. *Journal of classification*, 37(3), 696-708. doi: 10.1007/s00357-019-09345-1
- Papi, S., Ferrara, M., Maltoni, D., & Anthonioz, A. (2016, September). On the generation of synthetic fingerprint alterations. In *2016 International Conference of the Biometrics Special Interest Group (BIOSIG)* (pp. 1-6). IEEE.
- Peralta, D., Triguero, I., García, S., Saeys, Y., Benitez, J. M., & Herrera, F. (2017). On the use of convolutional neural networks for robust classification of multiple fingerprint captures. *International Journal of Intelligent Systems*, 33(1), 213–230. doi: 10.1002/int.21948
- Perišić, N., & Jovanović, R. (2023). Konvolucione neuronske mreže u sistemima automatskog upravljanja-pregled stanja u oblasti istraživanja. *Tehnika*, 78(4), 433-441. doi: 10.5937/tehnika2304433P
- Ruiz-Garcia, A. (n.d.). *SOCOFing* [Data set]. Kaggle. Retrieved from <https://www.kaggle.com/datasets/ruizgara/socofing>
- Saleem, M. A., Senan, N., Wahid, F., Aamir, M., Samad, A., & Khan, M. (2022). Comparative analysis of recent architecture of convolutional neural network. *Mathematical Problems in Engineering*, 2022(1), 7313612. doi: 10.1155/2022/7313612
- Saipidinov, I. M., Khamdamov, O. N., Bandurina, I. P., Fomenko, N. M., & Karanina, E. V. (2025). Environmental management of quality: the modern vision of sustainable business. *International Journal for Quality Research*, 19(2), 535–548. doi: 10.24874/IJQR19.02-12
- Shehu, Y. I., Ruiz-Garcia, A., Palade, V., & James, A. (2018). Sokoto Coventry Fingerprint Dataset. *arXiv preprint arXiv:1807.10609*.
- Shehu, Y. I., Ruiz-Garcia, A., Palade, V., & James, A. (2018a). Detection of fingerprint alterations using deep convolutional neural networks. In *International conference on artificial neural networks* (pp. 51-60). Cham: Springer International Publishing. doi: 10.1007/978-3-030-01418-6_6
- Simonyan, K., & Zisserman, A. (2015). *Very deep convolutional networks for large-scale image recognition*. In *3rd international conference on learning representations*. ICLR 2015. Computational and Biological Learning Society. Retrieved from <https://arxiv.org/pdf/1409.1556>
- Singh, R., Sharma, N., Chauhan, R., Choudhary, A., & Gupta, R. (2023). Enhanced fingerprint alteration detection using lightweight CNN model trained on SOCOFing dataset. In *2023 3rd International Conference on Smart Generation Computing, Communication and Networking (SMART GENCON)* (pp. 1-6). IEEE. doi: 10.1109/SMARTGENCON60755.2023.10441980
- Singhal, N., Gupta, N., & Singh, A. K. (2025). Facial detection in computer vision: bridging gap between CNN haar cascade and MTCNN. *Proceedings on Engineering Sciences*, 7(1), 427-436. doi: 10.24874/PES07.01C.009
- Swapna, M., Sharma, Y. K., & Prasad, B. M. G. (2020). CNN Architectures: Alex Net, Le Net, VGG, Google Net, Res Net. *International Journal of Recent Technology and Engineering*, 8(6), 953-959. doi: 10.35940/ijrte.F9532.038620
- Wang, C. (2023). A review on 3D convolutional neural network. In *2023 IEEE 3rd International Conference on Power, Electronics and Computer Applications (ICPECA)* (pp. 1204-1208). doi: 10.1109/ICPECA56706.2023.10075760
- Yamashita, R., Nishio, M., Do, R. K. G., & Togashi, K. (2018). Convolutional neural networks: an overview and application in radiology. *Insights into imaging*, 9, 611-629. doi: 10.1007/s13244-018-0639-9
- Yoon, S., Feng, J., & Jain, A. K. (2012). Altered fingerprints: Analysis and detection. *IEEE transactions on pattern analysis and machine intelligence*, 34(3), 451-464. doi: 10.1109/TPAMI.2011.161

- Zeiler, M. D., & Fergus, R. (2014). *Visualizing and understanding convolutional networks*. In Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (Eds) *Computer Vision – ECCV 2014* (pp. 818–833). Springer. doi: 10.1007/978-3-319-10590-1_53
- Zhang, X., Yang, Y. H., Han, Z., Wang, H., & Gao, C. (2013). Object class detection: A survey. *ACM Computing Surveys (CSUR)*, 46(1), 1-53. doi: 10.1145/2522968.2522978
- Zhao, X., Wang, L., Zhang, Y., Han, X., Deveci, M., & Parmar, M. (2024). A review of convolutional neural networks in computer vision. *Artificial Intelligence Review*, 57, 99. doi: 10.1007/s10462-024-10721-6

Jelena Dencic

University of Kragujevac,
Faculty of engineering,
Kragujevac,
Serbia
dencicjelena5@gmail.com

Marija Zahar Djordjevic

University of Kragujevac,
Faculty of engineering,
Kragujevac,
Serbia
maja_199@yahoo.com
ORCID 0000-0003-4905-2728

Tijana Petrovic

University of Kragujevac,
Faculty of engineering,
Kragujevac,
Serbia
t.cvetic@kg.ac.rs
ORCID 0000-0001-5563-8982
