Local Binary Pattern in the Frequency Domain: Performance Comparison with Discrete Cosine Transform and Haar Wavelet Transform

Eren Sen
Department of Information Technologies, Istanbul Nisantasi University, Istanbul, Turkiye
20231530002@std.nisantasi.edu.tr (corresponding author)

Ibrahim Furkan Ince
Department of Software Engineering, Istinye University, Istanbul, Turkiye
furkan.ince@istinye.edu.tr

Ali Ozkurt
Department of Computer Engineering, Istanbul Nisantasi University, Istanbul, Turkiye
ali.ozkurt@nisantasi.edu.tr

Furkan Ilker Akin
Department of Management, Faculty of Economics, Administrative and Social Sciences, Istanbul Nisantasi University, Istanbul, Turkiye
furkan.akin@nisantasi.edu.tr

Received: 1 January 2025 | Revised: 19 January 2025 | Accepted: 29 January 2025

 DOI: https://doi.org/10.48084/etasr.10110

ABSTRACT

This study presents a new method that aims to improve iris recognition performance by amplifying high-frequency components in the frequency domain, considering that iris images naturally contain high-frequency details. The Haar Wavelet Transform (HWT) and Discrete Cosine Transform (DCT) are used to enhance these components and an inverse transformation is applied to obtain iris images with more details. As input, the brightness values of the 8 neighboring pixels around each central pixel are used. These values are transformed into the frequency domain, the high-frequency band is amplified, and the data are reconstructed. Feature vectors are then generated using the Local Binary Pattern (LBP) algorithm, which is fed with the enhanced images. These feature vectors are formed using a combination of local histograms rather than a global LBP histogram, which are normalized to ensure consistency. The generated feature vectors are divided into a 70% training set and a 30% test set and are tested using K-Nearest Neighbors (KNN), Support Vector Machines (SVM), and Random Forest (RF) algorithms. The proposed method provides a significant performance improvement in terms of accuracy compared to traditional approaches. While both HWT and DCT yield similar results, it has been observed that HWT is much faster. In this study, a comparison is made in terms of both speed and accuracy. Two different public iris datasets, MMU1 and MMU2, are used. This work not only introduces an innovative approach to iris recognition, but also makes a significant contribution to the manipulation of pixel brightness values in the frequency domain, with the findings being expected to guide future research.

Keywords-iris recognition; frequency domain; high-frequency amplification; Haar Wavelet Transform (HWT); Discrete Cosine Transform (DCT); Local Binary Pattern (LBP)

I. INTRODUCTION

Iris recognition uses pattern recognition techniques on iris images to uniquely identify an individual [1]. The foundations of this technology were first proposed by Flam and Safi in 1987 [2]. Since then, iris recognition systems have been widely utilized in biometric authentication processes, and more than 50 million people worldwide have been identified deploying iris code recognition methods [1, 2]. LBPs capture local texture information by converting grayscale variations around a pixel into binary codes [3]. This feature has gained prominence because it allows the effective representation of micropatterns in an image, making the method successful in areas, such as texture classification, face recognition, and object detection [4, 5]. First introduced in the late 1990s [3], LBP has been successfully used in a wide range of applications due to its simplicity, computational efficiency, and flexibility [6-14]. Authors in [15] proposed a new descriptor called Feature-based Local Binary Pattern (FbLBP), which decomposes the difference vector into sign and magnitude components and improves classification accuracy by more than 10% compared to conventional LBP. Similarly, authors in [16] proposed a method called Local Binary Pattern Histogram Fourier features (LBP-HF), which uses the Fourier transform of LBP histograms. LBP-HF provides a more robust representation against rotation, delivering effective results in applications where rotation variations are significant [17]. Authors in [18] proposed a novel iris recognition approach that addresses challenges, such as illumination variation, occlusion, and noise, by extracting features from modified pixel values and combining radial and circumferential features using a Block Local Binary Pattern (BLBP) method. The experimental results on multiple publicly available iris databases demonstrate the effectiveness of the system. This method has shown a high success rate especially in object detection applications [19]. Authors in [20] proposed the Multi-Scale Block Local Binary Patterns (MB-LBP) method by combining Haar wavelets with LBP. MB-LBP gained the ability to perform multi-scale analysis by integrating Haar-like features with LBP. This method has shown a high success rate especially in object detection applications. It is thus expected that the integration of new algorithms with LBP and its role in frequency-based analysis will contribute to the development of more robust and powerful systems in the future.

This study presents a novel approach that significantly improves the performance of iris recognition by amplifying the high-frequency components in the frequency domain. Unlike conventional methods that rely primarily on spatial domain features, the focus is on utilizing HWT and DCT to boost the high-frequency details of the iris image. The motivation behind this approach lies in the fact that iris images naturally contain a wealth of high-frequency information, particularly in the fine textures of the iris, which are critical for accurate identification. By applying HWT and DCT to the image, the present study can selectively amplify the frequency components that capture these intricate patterns, while suppressing lower-frequency components that may introduce redundancy or noise into the feature extraction process. The core of the proposed method involves the extraction of pixel intensity values from the eight neighboring pixels surrounding each central pixel in the iris image. These pixel values are then transformed into the frequency domain, where the high-frequency bands are amplified to enhance the finer details of the iris texture. Once the frequency components have been modified, an inverse transformation is applied to reconstruct the image with the amplified high-frequency details. This reconstructed image retains the essential features of the iris while highlighting the subtle patterns that are often critical to distinguishing between individuals. The enhanced image is then fed into the feature extraction process, where the LBP algorithm is employed to generate robust feature vectors.

The contributions of this work are twofold. First, it introduces a new method for enhancing iris recognition performance by amplifying the high-frequency components of the iris image in the frequency domain. This approach addresses the limitations of traditional methods, which often overlook the importance of high-frequency details that are critical for accurate recognition. Second, it makes a significant contribution to the field by demonstrating the practical application of HWT and DCT in improving iris recognition accuracy and computational efficiency. By integrating these techniques with advanced feature extraction algorithms, like LBP, a comprehensive solution that improves both the speed and precision of iris recognition systems is provided.

II. MATERIALS AND METHODS

This study uses the Multimedia University (MMU1) iris image dataset, which is accessible via [21], and the MMU2 iris image dataset, which was previously accessible via [22], for the proposed iris authentication system. The iris images in this dataset are openly available, with no license required, and were collected from students at Multimedia University. The dataset is widely used in biometric systems, such as for attendance and authentication purposes. The images are originally stored in BMP format, and for this study's processing they were converted to PNG format because the utilized Java processing IDE could not handle BMP files.

In the proposed method, the brightness values of the 8 neighbors around each central pixel are collected into a one-dimensional array of length 8. Each of these arrays is processed using frequency transformations, such as the HWT or DCT. After transitioning to the frequency domain, the high-frequency components are enhanced. Then an inverse transformation is applied, and the resulting frequency data are returned to the original space. This process allows for the enhancement of fine details and structural information in the image. The goal of using HWT and DCT is to enhance subtle details in biometric images, such as irises, which are more prominent.

A. Haar Wavelet Transform

This method collects the brightness values of 8 neighbors around each central pixel, arranges them in a one-dimensional array, and then applies the HWT to this array. The purpose of HWT is to enhance the high-frequency components of the image and attenuate the low-frequency components, thus making the fine details in the image more distinct. The main advantage of HWT is its computational speed. Next, an inverse transform is applied, which returns the frequency domain image to the original space. This step places the information obtained from the frequency transform back into the spatial domain. The brightness values of the 8 neighbors around each pixel are calculated and stored in the neighbors' array, as shown in:

$$\text{neighbors}[i] = \text{brightness}(p_{xi,yi}) \quad (1)$$

where $p_{xi,yi}$ is the brightness value of the neighboring pixel. HWT is a transformation technique used to extract the fundamental structure of a signal or image. This process involves converting the data into averages and differences and involves a two-step operation. The average of two consecutive values is taken, and the first value is subtracted by the average. Mathematically, this is expressed in (2) and (3). This operation is performed on $\text{input}[n]$ over multiple steps. The first step is applied to the $\text{input}[2]$ elements, the second step to $\text{input}[4]$, and so on, continuing until n. At each step, the size of the data is halved. The algorithm structure is:

$$\text{input}\left[\frac{i}{2}\right] = \frac{\text{input}[i]+\text{input}[i+1]}{2} \quad (2)$$

$$output\left[\frac{i+k}{2}\right] = \text{input}[i] - \text{output}[i] \quad (3)$$

The inverse Haar transform restores the data back to their original form. By recombining the averages and differences, each data component is recalculated. Mathematically, this is expressed by:

$$\text{output}[i * 2] = \text{original}[i] + \text{original}[i + k] \quad (4)$$

$$\text{output}[i * 2 + 1] = \text{original}[i] - \text{original}[i + k] \quad (5)$$

HWT allows a signal or image to be decomposed into its frequency components. These components are divided into low-frequency components (averages) and high-frequency components (differences). The high frequencies contain fine details, edges, and textures of the image, whereas the low frequencies reflect the overall structure and main outlines. Before performing the inverse transformation, the high-frequency components in the frequency domain are scaled by a certain factor as demonstrated in the code below:

```
//High frequency boost code in HWT
for( int i = 0; i < 8; i++){
   for(int j = i; j < 8; j++ ){
      if(i != 0 )
         neighbors[j] *= 8D / (i + j);   }}
```

This loop performs the operation of enhancing the high-frequency components. Each high-frequency component is multiplied by a ratio based on its index and its associated neighbors. The scaling factor applied to the high-frequency components increases as the frequency increases. More specifically, each high-frequency component is amplified by the factor $\frac{8}{i+j}$. This multiplier depends on the sum of the values i and j. Larger values of the sum of i and j result in a smaller scaling factor. However, since $i \neq 0$, the first component (i.e., $i = 0$) remains unchanged; this component generally represents the low-frequency components. The main purpose of this operation is to enhance the high-frequency components with a larger multiplier, making fine details in the image more prominent. With this enhancement, the edges, and fine details (high-frequency components) in the image become more noticeable. Increasing the high-frequency components is typically used in operations, such as image sharpening or emphasizing details. As a result, the details in the image become more visible and clearer.

B. Discrete Cosine Transform

DCT is a mathematical transformation used to decompose a signal into its frequency components. This transformation allows the data to be converted to the frequency domain (low and high-frequency components) for analysis. DCT is widely utilized, particularly in applications such as image compression (e.g., JPEG format). Assuming a sequence is represented as $x = [x_0, x_1, \ldots, x_{N-1}]$, the DCT transformation is performed as in:

$$X_k = \sqrt{\frac{2}{N}} \cdot s_k \sum_{n=0}^{N-1} x_n \cdot \cos\left(\frac{\pi\left(n+\frac{1}{2}\right)k}{N}\right) \quad (6)$$

where X_K is the frequency component obtained from the DCT, N is the length of the data, $s_k = \sqrt{\frac{1}{2}}(\text{if } k = 0, \text{else } 1)$ is the normalization coefficient, x_n represents the elements of the input sequence, and k is the frequency component in the output. The inverse DCT is the transformation applied in reverse to DCT, bringing the data obtained in the frequency domain back to the original time or spatial domain. The mathematical formula for the inverse DCT is given in:

$$x_n = \sqrt{\frac{2}{N}} \sum_{k=0}^{N-1} X_k \cdot s_k \cdot \cos\left(\frac{\pi\left(n+\frac{1}{2}\right)k}{N}\right) \quad (7)$$

where x_n is the original data element obtained by the inverse DCT, X_k are the frequency components in the DCT transformation, and s_k is the normalization coefficient used in the transformation. The brightness values of the 8 neighboring pixels around each pixel are taken and placed into the array $\text{neighbors}[i]$. The DCT is then applied to this array of neighboring pixels by applying a factor to the frequency components to make them more prominent, as shown in:

$$\text{dct\_freq}[j] \times= \frac{8}{i+j}, \text{ if } i \neq 0 \quad (8)$$

where the indices i and j represent different frequency components. DCT separates the low and high frequencies in the image, making smaller and less perceptible details more visible. The weights applied to the high frequencies reveal the hidden details, such as fine edges or textures. Before the inverse transformation, the goal is to make these details more prominent by assigning more weight to the high-frequency components. To achieve this, each $\text{dct\_freq}[j]$ value is recalculated with a special multiplier in the following loop:

```
//High frequency boost code in DCT
for( int i = 0; i < 8; i++){
   for (int j = i; j < 8; j++ ){
      if(i != 0 )
         dct_freq[j] *= 8D / (i + j); }}
```

The multiplier gives more weight to the high-frequency components. When $i = 0$, no change is made, as $i = 0$ typically represents the low-frequency (DC component). In other cases, when $i \neq 0$, as $i + j$ increases, the multiplier $\frac{8}{i+j}$ decreases, giving more weight to the high-frequency components. This makes the high-frequency components more prominent. The weight applied to the high-frequency components, $\frac{8}{i+j}$, ensures that fine details, especially sharp edges, and textures, are emphasized more. Low frequencies are less emphasized because no change is made when $i = 0$, and when $i + j$ is small, the multiplier remains close to 1. This helps preserve the main structure of the image (such as the background or general colors). This method is particularly useful for sharpening or emphasizing visual details.

C. Local Binary Pattern

LBP is a method of assigning a bit value to each neighbor by comparing the brightness values of a pixel with those of its neighbors. These bit patterns show the relationship of each pixel to its surroundings and are commonly used in areas such as face recognition and texture analysis. Each pixel is compared to the values of its 8 neighbors. If the brightness value of a neighboring pixel is greater than the brightness value of the center pixel, the corresponding bit for that neighbor becomes "1", otherwise it becomes "0". These bits are then combined to form a number. Mathematically, this is shown in:

$$\mathrm{LBP} = \sum_{i=0}^{7} 2^{i} \cdot 1(p_i > p_{center}) \quad (9)$$

where p_i is the brightness value of the neighboring pixels, p_{center} is the brightness value of the center pixel, and $1(\cdot)$ is a sign function that returns 1 when $p_i > p_{center}$ and 0 otherwise.

D. Proposed Method

The proposed method is a combination of enhancing high-frequency components in the image using HWT or DCT and then extracting local features using LBP. This method provides high accuracy, especially for biometric applications such as iris recognition. The brightness values of the 8 neighbors surrounding each center pixel are collected to form a one-dimensional array. This array undergoes either HWT or DCT, which transforms the image into the frequency domain. This transformation emphasizes high frequencies and attenuates low frequencies. After processing in the frequency domain, an inverse transformation is applied to return to the original space. Following the inverse transformation, the LBP pattern is created by comparing the neighbors of each pixel to the center pixel. Local histograms of the LBP image are extracted, merged, and normalized. The normalized histogram is then converted into a feature vector for use in Machine Learning (ML) algorithms.

E. Complexity Analysis

HWT works with data pairs at each stage. It processes 2 elements in the first stage, 4 elements in the second stage, and so on, reducing the size of the data by half at each stage. This transformation is typically considered as an algorithm with linear time complexity. When the length of the data is N, the time complexity of HWT is $O(N)$. DCT generally involves cosine functions and discretization processes. This transformation is mainly used to decompose images into frequency components. The most used DCT algorithm is the Fast Cosine Transform (FCT) or Fast Fourier Transform (FFT). However, since the FFT has a time complexity of $O(N\ log\ N)$, DCT is expected to have a similar complexity of $O(N\ log\ N)$.

III. EXPERIMENTAL RESULTS

The primary goal of the experiments is to evaluate the impact of enhancing high-frequency components in the frequency domain on the accuracy of iris recognition. To this end, the effectiveness of the improvements made using HWT and DCT was measured using three ML classifiers: SVM, KNN, and RF. The results are presented in terms of accuracy, processing time, and computational efficiency. Tables I and II display the comparisons made between the proposed method and the traditional approaches.

TABLE I. ACCURACY RESULTS

Conventional LBP					
SVM		KNN		RF	
MMU1	MMU2	MMU1	MMU2	MMU1	MMU2
72.1805%	74.9164%	75.1880%	79.5987%	56.3910%	75.2508%
DCT - LBP					
SVM		KNN		RF	
MMU1	MMU2	MMU1	MMU2	MMU1	MMU2
85.7143%	85.2843%	91.7293%	82.6087%	79.6992%	80.9365%
HWT - LBP					
SVM		KNN		RF	
MMU1	MMU2	MMU1	MMU2	MMU1	MMU2
84.9624%	85.2843%	90.9774%	84.9498%	83.4586%	80.9365%

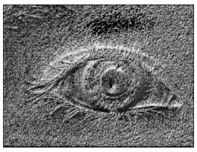
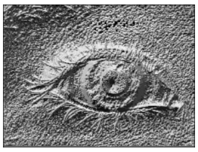
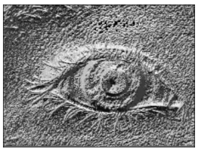
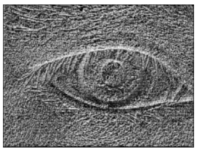
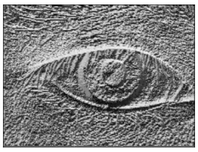
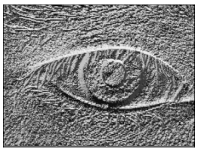
In the DCT-based LBP method, high accuracy rates were achieved with all ML algorithms. In particular, SVM achieved an accuracy rate of 85.71% (MMU1) and 85.28% (MMU2), whereas KNN and RF achieved 91.73% (MMU1) and 82.61% (MMU2), and 79.70% (MMU1) and 80.94% (MMU2), respectively. These results indicate that enhancing the high-frequency components in the frequency domain can effectively improve the iris recognition performance. Similarly, the HWT-based LBP method produced strong results. SVM achieved an accuracy rate of 84.96% (MMU1) and 85.28% (MMU2), whereas KNN achieved an accuracy rate of 90.98% (MMU1) and 84.95% (MMU2). RF achieved accuracy rates of 83.46% (MMU1) and 80.94% (MMU2).

TABLE II. PROCESSING TIME RESULTS

Method	Dataset	
	MMU1 (ms)	MMU2 (ms)
Conventional - LBP	19	17
DCT - LBP	165	159
HWT - LBP	51	48

In addition to the accuracy rates, processing times are a critical factor when evaluating the performance of the methods. Table II presents the processing times of different methods on the MMU1 and MMU2 datasets. These times are important for comparing the computational complexity and speed of the transformation methods. The HWT is the fastest method in terms of processing time. An average processing time of 19 ms was measured for the MMU1 dataset and 17 ms for the MMU2 dataset. The low processing time of HWT is a significant advantage for real-time applications. Additionally, Table III portrays the images taken from the output of the algorithms. Comparing the LBP images produced by the conventional method with the LBP images enhanced by DCT and HWT, the effect of enhancing the high-frequency components in the frequency domain is clearly observed. In the conventional LBP images, the high-frequency details in the iris texture were not sufficiently emphasized. The fine details of the iris structure were lost due to low contrast and limited sharpness, which negatively affected the accuracy of the classification algorithms.

TABLE III. IMAGES BEFORE AND AFTER PROCESSING

Conventional - LBP	DCT - LBP	HWT - LBP
		
		

IV. CONCLUSION

This study presents an innovative approach to improve the performance of iris recognition systems in terms of accuracy and speed by manipulating high-frequency components in the frequency domain. By enhancing the high-frequency components of iris images, more detailed images were generated, and these details were transformed into feature vectors using the Local Binary Patterns (LBP) algorithm. The Discrete Cosine Transform (DCT)-based LBP method achieved high accuracy rates with all Machine Learning (ML) algorithms. Specifically, Support Vector Machines (SVM) achieved 85.71% (MMU1) and 85.28% (MMU2) accuracy, whereas K-Nearest Neighbors (KNN) and Random Forest (RF) achieved 91.73% (MMU1) and 79.70% (MMU1), 82.61% (MMU2) and 80.94% (MMU2) accuracy, respectively. These results demonstrate that enhancing high-frequency components in the frequency domain can improve iris recognition performance. Similarly, the Haar Wavelet Transform (HWT)-based LBP method produced strong results, with SVM reaching 84.96% (MMU1) and 85.28% (MMU2) accuracy, KNN reaching 90.98% (MMU1) and 84.95% (MMU2) accuracy, and RF reaching 83.46% (MMU1) and 80.94% (MMU2) accuracy. The results obtained using HWT and DCT showed similar accuracy levels; however, HWT was found to be significantly more efficient in terms of processing speed. The proposed method achieved a significant accuracy improvement over conventional methods on both the MMU1 and MMU2 datasets. Furthermore, the experiments using KNN, SVM, and RF demonstrated a significant improvement in classification performance.

ACKNOWLEDGMENT

This paper is derived from the master's thesis of a graduate student named Eren Şen, under the supervision of faculty members İbrahim Furkan İnce and Ali Özkurt. The work is carried out at the Graduate School of İstanbul Nişantaşı University. In addition, the authors would like to thank the research assistant Furkan İlker Akin for his contributions to the paper.

REFERENCES

[1] B. O. Connor and K. Roy, "Iris Recognition Using Level Set and Local Binary Pattern," *International Journal of Computer Theory and Engineering*, vol. 6, no. 5, pp. 416–420, Oct. 2014, https://doi.org/10.7763/IJCTE.2014.V6.901.

[2] A. Sharma and M. R. Gupta, "Iris Recognition Based Learning`Vector Quantization and Local Binary Patterns on Iris Matching," *International Journal of Technical Research and Applications*, vol. 3, no. 5, pp. 7–14, Sep.-Oct. 2015.

[3] W. Li, C. Chen, H. Su, and Q. Du, "Local Binary Patterns and Extreme Learning Machine for Hyperspectral Imagery Classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 7, pp. 3681–3693, Jul. 2015, https://doi.org/10.1109/TGRS.2014.2381602.

[4] C.-H. Chan, J. Kittler, and K. Messer, "Multi-scale Local Binary Pattern Histograms for Face Recognition," in *2007 International Conference on Biometrics*, Seoul, Korea, 2007, pp. 809–818, https://doi.org/10.1007/978-3-540-74549-5_85.

[5] R. Y. Lad, S. Mapari, and F. N. Sibai, "A Novel Approach to Image Classification for Detecting Abnormalities in Neuroimages based on the Structural Similarity Index Measure," *Engineering, Technology & Applied Science Research*, vol. 14, no. 5, pp. 17382–17387, Oct. 2024, https://doi.org/10.48084/etasr.8384.

[6] X. Qian, X.-S. Hua, P. Chen, and L. Ke, "PLBP: An effective local binary patterns texture descriptor with pyramid representation," *Pattern Recognition*, vol. 44, no. 10–11, pp. 2502–2515, Oct.-Nov. 2011, https://doi.org/10.1016/j.patcog.2011.03.029.

[7] D. Huang, C. Shan, M. Ardabilian, Y. Wang, and L. Chen, "Local Binary Patterns and Its Application to Facial Image Analysis: A Survey," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 41, no. 6, pp. 765–781, Nov. 2011, https://doi.org/10.1109/TSMCC.2011.2118750.

[8] Y. Hong, C. Leng, X. Zhang, Z. Pei, I. Cheng, and A. Basu, "HOLBP: Remote Sensing Image Registration Based on Histogram of Oriented Local Binary Pattern Descriptor," *Remote Sensing*, vol. 13, no. 12, Jun. 2021, Art. no. 2328, https://doi.org/10.3390/rs13122328.

[9] Y. Kaya, Ö. F. Ertuğrul, and R. Tekin, "Two novel local binary pattern descriptors for texture analysis," *Applied Soft Computing*, vol. 34, pp. 728–735, Sep. 2015, https://doi.org/10.1016/j.asoc.2015.06.009.

[10] D. G. R. Kola and S. K. Samayamantula, "A novel approach for facial expression recognition using local binary pattern with adaptive window," *Multimedia Tools Appl.*, vol. 80, no. 2, pp. 2243–2262, Jan. 2021, https://doi.org/10.1007/s11042-020-09663-2.

[11] C. Li, W. Zhou, and S. Yuan, "Iris recognition based on a novel variation of local binary pattern," *The Visual Computer*, vol. 31, no. 10, pp. 1419–1429, Oct. 2015, https://doi.org/10.1007/s00371-014-1023-5.

[12] L. Nanni, A. Lumini, and S. Brahnam, "Local binary patterns variants as texture descriptors for medical image analysis," *Artificial Intelligence in Medicine*, vol. 49, no. 2, pp. 117–125, Jun. 2010, https://doi.org/10.1016/j.artmed.2010.02.006.

[13] Z. Pan, S. Hu, X. Wu, and P. Wang, "Adaptive center pixel selection strategy in Local Binary Pattern for texture classification," *Expert Systems with Applications*, vol. 180, Oct. 2021, Art. no. 115123, https://doi.org/10.1016/j.eswa.2021.115123.

[14] J. Ren, X. Jiang, and J. Yuan, "Noise-Resistant Local Binary Pattern With an Embedded Error-Correction Mechanism," *IEEE Transactions on Image Processing*, vol. 22, no. 10, pp. 4049–4060, Oct. 2013, https://doi.org/10.1109/TIP.2013.2268976.

[15] Z. Pan, Z. Li, H. Fan, and X. Wu, "Feature based local binary pattern for rotation invariant texture classification," *Expert Systems with Applications*, vol. 88, pp. 238–248, Dec. 2017, https://doi.org/10.1016/j.eswa.2017.07.007.

[16] T. Ahonen, J. Matas, C. He, and M. Pietikäinen, "Rotation Invariant Image Description with Local Binary Pattern Histogram Fourier Features," in *16th Scandinavian Conference on Image Analysis*, Oslo, Norway, 2009, pp. 61–70, https://doi.org/10.1007/978-3-642-02230-2_7.

[17] X. Qi, R. Xiao, C.-G. Li, Y. Qiao, J. Guo, and X. Tang, "Pairwise Rotation Invariant Co-Occurrence Local Binary Pattern," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 11, pp. 2199–2213, Nov. 2014, https://doi.org/10.1109/TPAMI.2014.2316826.

[18] A. Nigam, V. Krishna, A. Bendale, and P. Gupta, "Iris recognition using block local binary patterns and relational measures," in *IEEE International Joint Conference on Biometrics*, Clearwater, FL, USA, 2014, pp. 1–6, https://doi.org/10.1109/BTAS.2014.6996263.

[19] G. Zhao and M. Pietikainen, "Dynamic Texture Recognition Using Local Binary Patterns with an Application to Facial Expressions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 915–928, Jun. 2007, https://doi.org/10.1109/TPAMI.2007.1110.

[20] Y. Liu, K. Xu, and J. Xu, "An Improved MB-LBP Defect Recognition Approach for the Surface of Steel Plates," *Applied Sciences*, vol. 9, no. 20, Oct. 2019, Art. no. 4222, https://doi.org/10.3390/app9204222.

[21] "MMU1 Iris Dataset." Kaggle, [Online]. Available: https://www.kaggle.com/datasets/naureenmohammad/mmu-iris-dataset.

[22] "MMU2 Iris Dataset." Advanced Source Code, [Online]. Available: http://www.advancedsourcecode.com/irisdatabase.asp.