An Enhanced LBPH Algorithm for Robust Face Feature Extraction

Solomon Anab[1], Godbless Mensah[2], Edward Yellakuor Baagyere[3], Mustapha Adamu Mohammed[4]

[1,2] Lecturer, Computer Science Department, Koforidua Technical University
[3]Lecturer, Computer Science, C.K. Tedam University of Tech. and App. Science
[4]Lecturer, Computer Science Department, Koforidua Technical University

Abstract

From security systems to human-computer interaction, face recognition technology is a key component in many different applications. In this domain, the Local Binary Pattern Histogram (LBPH) algorithm has shown great promise due to its ability in texture-based feature extraction and robustness.

In this study, the performance of LBPH algorithm is improved by optimising important parameters: radius, number of neighbours and grid configuration. Tweaking these parameters systematically enable us to get the most success out of this algorithm. LBPH algorithm parameters has been tuned into a 5x7 dimensional matrix, which gives us total of 35 grids with equal width and height pixels. In other words, one central pixel plus 34 neighbouring pixels where the radii of 3 square neighbourhoods can be adjusted.

The study combines the experimental exploration with fine-tuning via machine learning approaches to optimize LBPH algorithm. Finally, we have provided experimental results on intra-class and inter-class feature distribution analysis conducted on selected images taken under constraints and unconstraint environments. The performance of our novel 34N-LBPH algorithm showed very low values by obtaining intra-class average Means of 0.031, 0.078, and 0.101 for three classes under constraint environment indicating the proposed 34N-LBPH is robust for facial feature extraction. The findings suggest that our enhanced algorithm, 34 Neighbour Linear Binary Pattern Histogram (34N-LBPH) can effectively handle variations in lighting, expressions and occlusions, contributing to the advancement of facial feature extraction for face recognition.

Keywords: Face recognition, LBPH, Algorithm, Parameters, Intra and Inter-Class, Analysis.

1.0 Introduction

The Local Binary Patterns Histogram (LBPH) algorithm is a widely used texture descriptor in computer vision, particularly in facial recognition. It uses Local Binary Patterns (LBP) to describe the local spatial structure of an image. The algorithm generates binary numbers representing patterns, which are then converted into decimal values. This process generates a histogram, which uniquely characterizes the image's texture. LBPH is invariant to monotonic grayscale transformations and computationally efficient, making it suitable for real-time applications like facial recognition in surveillance systems or user authentication in mobile devices. It is particularly effective in low-resolution images. LBPH is widely used in facial recognition, object detection, texture classification, and image retrieval [1].

Different methods have been used to categorize faces based on learned patterns, such, as Support Vector Machines (SVMs) and Linear Discriminant Analysis (LDA) [2]. The Fisherface algorithm, introduced by Belhumeur et al. in 1997 improved upon eigenfaces by integrating discriminant analysis.
Face recognition have been revolutionised through Convolutional Neural Networks (CNNs) by learning features in a hierarchy directly from raw pixel data [4]. The introduction of Siamese Networks and Triplet Loss also enable end-to-end learning for face recognition tasks [5].

1.1 Applications

Security: Access control, surveillance and forensics are fields that often apply Face recognition as a security measure [6].
Human-Computer Interaction: Facial recognition is used in user authentication for devices, and it enables interactions between humans and various applications [7].
While a multitude of enhancement strategies have been extensively investigated for mitigating the limitations of this algorithm, there has been little focus on compensating for its collapse in discriminative ability to address complicated appearance changes that introduce intra-class variations.
In this study, we improve the LBPH algorithm with a parameter optimization to strengthen its discriminative power for handling facial appearance variations in complex scenarios.
This study focuses on enhancing the LBPH algorithm by optimising its parameters to bolster its discriminative power to handle complex variations in facial appearance.
This work has included studying many facets of LBPH, such as parameter tuning, and innovative feature extraction methods. We further look into integrating machine learning frameworks including deep neural networks to extend the discriminative capabilities of LBPH. This is in hope to strengthen the ongoing quest of state-of-the-art face recognition systems by testing the limits of LBPH.

2.0 Related Research Work

The LBPH algorithm is commonly used for facial texture pattern extraction in computer vision, but its performance depends on its parameters, which can be optimized. Researchers are working to improve the technique for more robust facial features extraction in constrained and unconstrained environments.

2.1 The Local Binary Pattern Histogram Algorithm

The Local Binary Pattern Histogram (LBPH) algorithm, originally introduced by Ojala et al. in 1996 [8], is an important face recognition algorithm because of its flexibility, user friendly, and reliable results even under different conditions and environment. LBP operator extract facial texture information by comparing pixel intensities with neighbour pixels [8]. Facial texture details are captured by local patterns making them robust to variations in illumination conditions and facial expressions. Transformation of LBP values into histograms enables efficient and effective feature extraction. Histograms create a concise representation of facial features, facilitating faster recognition [9]. The LBP operator produces a binary pattern with a value between 0 and 1 [10]. Cells are the discrete, smaller sections that make up the image. Each of the cell's eight neighbours is compared to each pixel therein. The centre pixel's value will serve as the threshold value [11]. If the value of each of the eight neighbour's pixel is more than or equal to the centre pixel, it will be set to one; otherwise, it will be set to zero. As a result, to create the LBP code for the centre pixel, the values of its eight neighbours' pixels (either ones or zeros) are concatenated into a binary code. This binary code is then conveniently translated to a 256-dimensional decimal to serve as the centre pixel's texture descriptor. The window of 3*3 defines the first LBP algorithm. The median pixel value is employed as the window's threshold, comparing it to the value of each of the eight pixels surrounding it. The pixel location value is recorded as 1 if the neighbourhood pixel value is greater or equal to the median pixel value, else as 0. This method compares 8 pixels in the 3*3 neighbours to get 8-bit binary values. After being converted to decimal numbers, the LBP values of the window's central pixel points are obtained. The texture features of the region are then displayed using these values. Several empirical studies have been carried to improve the LBPH algorithm for better face recognition. These studies have shown that, tuning parameters improves recognition accuracy, robustness to environmental changes, and also makes it more generalize to new dataset. Nevertheless, the LBPH algorithm cannot supply sufficient information to handle different facial appearances which vary in more complex ways based on landmarks [12].

2.1.1 Mathematical Formulation

The threshold function of the LBPH algorithm is represented by the formula in equation 1

$$S(x) = \begin{cases} 0, & x < 0 \\ 1, & x \geq 0 \end{cases} \qquad (1)$$

Here, s(x) is the threshold function

The general $LBP_{P,R}$ operator, is therefore defined as:

$$LBP_{P,R}(x_c, y_c) = \sum_{P=0}^{P-1} S(g_p - g_c)2^p \qquad (2)$$

From equation (2), the texture can be defined as:

$$T \square\, t(LBP_{P,R}(x_c, y_c)) \qquad (3)$$

In summary, equation 1 and 2 illustrates the threshold function and the general LBP operator respectively.

Figure1: The First LBP Operator [8]

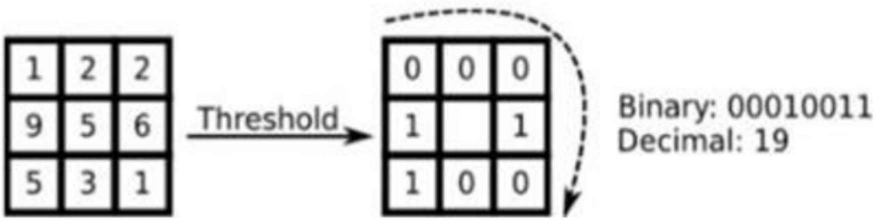

Figure2: LBP Operator for different set of P and R [13]

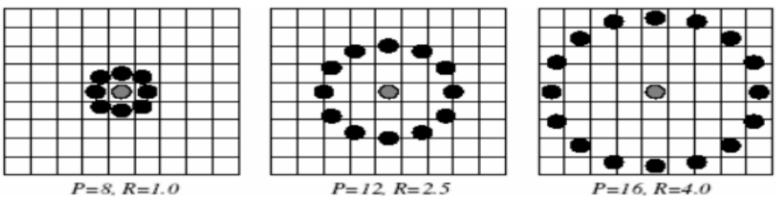

Figure 1 shows a diagram of window of 3*3 which defines the first LBP algorithm introduced by Ojala et al. in 1996. The median pixel value is employed as the window's threshold, comparing it to the value of each of the eight pixels surrounding it. The pixel location value is recorded as 1 if the neighbourhood pixel value is greater or equal to the median pixel value, else as 0. The diagram in Figure 2 illustrates LBP operator for different set of Pixels and Radii for multiresolution grey-scale and rotation invariant texture classification by Ojala et al. [13].

2.2 Building on Strengths and Addressing Weaknesses

Several methods have been introduced in related works to deal with the weaknesses of LBPH and leverage its strengths. Parameter Optimization Tweaking parameters such as radius, number of neighbours, and grid configuration can help mitigate some of LBPH's limitations. Radius and Neighbour of LBPH Neighbour value and Radius are two important parameters in the LBPH algorithm with radius 1 as the default radius [13]. Increasing the radius may help to extract more details. The neighbour value should be a positive integer ranging from 4 to 24 with 8 as default value [13]. Cells or Grid An image has multiple cells or grid in Horizontal and vertical directions. There is no set range for the number of the grid, and none of the preceding research considered these issues.

2.3 Cells or Grid

An LBPH-based image processing technique that takes into account radius and neighbour values has been proposed [14]; A study by Tiryaki et al. [15] studied varying window sizes from 3 x 3 to 5 x 5 to 7 x 7 and obtained accuracy rates of 86.7%, 91.4%, and 94.09%, respectively; however, grid size was not considered during the study, same as it was absent in [14].

Inspired by [15], distinctive grid/cell sizes of 5 x 7 have been introduced in this work along with 3 variable radii to obtain 35 grid/cells making neighbour cells value of 34 with 1 central cell with the focus on enhanced LBPH face recognition techniques. Even though the current LBPH models demonstrate high accuracy, there are still limitations and gaps, particularly images taken under unconstrained environments.

2.4 Intra-Class and Inter-Class Distance Analysis

Intra-class and Inter-class distance analysis measures the average distance between samples within the same class and samples from different classes [16]. Euclidean distance calculates the average Mean distance between data points in data visualisation [17]. In Intra-class distance analysis, the values 0-0.1: Very low distances, indicating high similarity or almost identical samples within the class [18], 0.1-0.3: Low distances, indicating moderate similarity within the class [19], 0.3-0.6: Moderate distances, indicating some diversity within the class [20]. In inter-class distance analysis, the values 0.1-0.3 very low distances, indicating almost identical classes, 0.3-0.6: Low distances, indicating moderate similarity between classes, 0.6-1.0: Moderate distances, indicating noticeable differences between classes, >1.0: High distances, indicating dissimilarity between classes, >2.0: Very high distances, indicating almost unrelated classes.

3.0 Methodology

3.1 Proposed Enhancements

Introduction of 34-neighbour pixels, one (1) central pixel and three (3) adjustable radii to capture more global patterns at different scales to provide a more detailed representation of the image.

3.1.1 Proposed 34N-LBP Operator

Reference is made to equations 1 and 2, representing the threshold function and traditional/original LBP operator respectively introduced by Ojala et al., in 1996.

Optimising the original LBP operator, we get

$$34N - LBP = \sum_{n=0}^{n-1} S(i_n - i_c)2^n \qquad (4)$$

Here,

in = Value of Neighbour Pixel

ic = Value of Centre Pixel
n = sampling points (n = 0, 1. . . , 34 for a 5x7 cell, where n = 34)
r = radius (for 5x7 cell, it is 3)

$$S(Z) = \begin{cases} 1 \; if \; Z \; is \; \geq \; 0 \\ 0 \; if \; Z \; is \; < \; 0 \end{cases} \quad (5)$$

Where,
s(z)= threshold function

$$S(z) = \begin{cases} 1, if \;\; i_n \; \geq i_c \\ 0, if \;\; i_n \; < i_c \end{cases} \quad (6)$$

From equations 4 and 6, we get the texture as:

$$T \,\square\, t(34N - LBP) \quad (7)$$

Now, introduce 5x7 grid to yield 35 sampling points, we get the proposed 34N-LBP as

$$34N - LBP = \sum_{n=0}^{34} S(i_n - i_c)2^n \quad (8)$$

Now, introduce the threshold function as,

$$S(z) = \begin{cases} 1, if \;\; i_n \; \geq i_c \\ 0, if \;\; i_n \; < i_c \end{cases} \quad (9)$$

3.1.2 How The Proposed 34N-LBPH Algorithm Works

1. The given grayscale image is divided into 5 x 7 matrix of equal sizes forming 35 cells
2. Select the centre pixel in the image
3. Define 3 circular neighbourhoods around the centre pixel with 34 equally spaced neighbours. The radius of the circular neighbourhood determines the distance between the centre pixel and its neighbours.
4. Compare the value of each neighbour with the value of the centre pixel.
5. If a neighbour's intensity value is greater or equal to the intensity value of the centre pixel, assign a binary digit of one (1), otherwise assign zero (0)
6. This process is repeated for all the 34 neighbours resulting in a binary code of 34 bits which represents the local texture pattern around the central pixel. This binary pattern is treated as a binary number, and the values are read in a clockwise order to create the binary number.
7. Concatenate the binary values to form a 34-bit binary code
8. Convert the binary code to decimal to represent the local texture pattern
9. This process is repeated for every pixel in the given image
10. A Histogram is constructed for the local texture patterns to capture the distribution of different patterns in the image.
11. Normalize the histogram to make the LBPH descriptor robust to variations in illumination. This is done by dividing the frequency of each pattern by the total number of patterns in the image.
12. The final LBPH feature vector is constructed by concatenating the normalized histogram bins. This feature vector serves as a representation of the local texture patterns in the given image.

Figure 3: 34N-LBP Operator

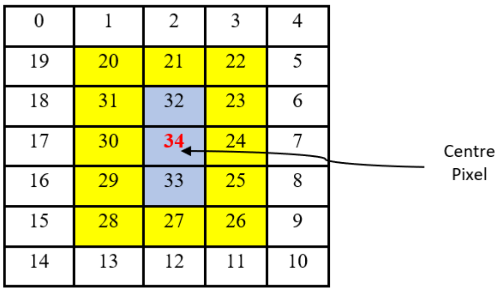

Figure 3 illustrate the proposed 34N-LBP operator with 5x7 matrixes of cells. Cell 34 being the centre pixel cell and cells zero to thirty-three being the neighbours.

Figure 4: P=20, R=3

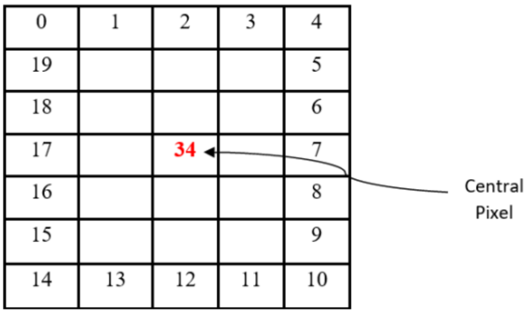

Figure 4 illustrate the Radius three with twenty pixels neighbours as shown in the figure.

Figure 5: P=12, R=2

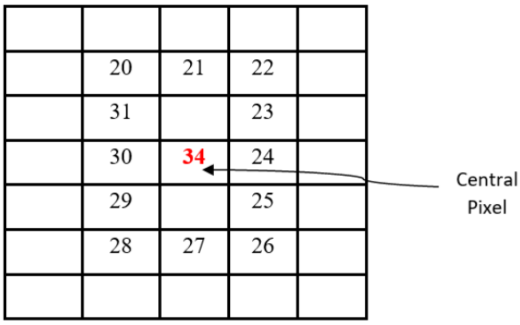

Figure 5 illustrate Radius two with twelve pixels neighbours

Figure 6: P=2, R=1

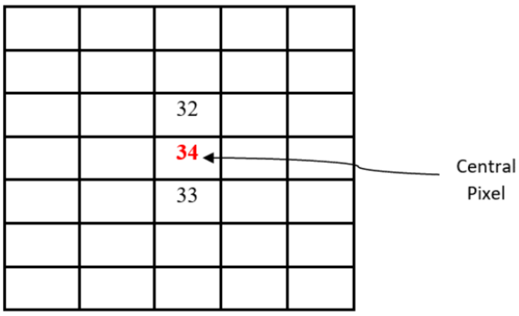

Figure 6 represent Radius one with two neighbours as shown.

3.2 Framework For Evaluation Of The Proposed 34N-LBPH Algorithm On Images Selected From The ORL Dataset

Figure 7: Framework for Evaluation of the proposed 34N-LBPH Algorithm on ORL dataset

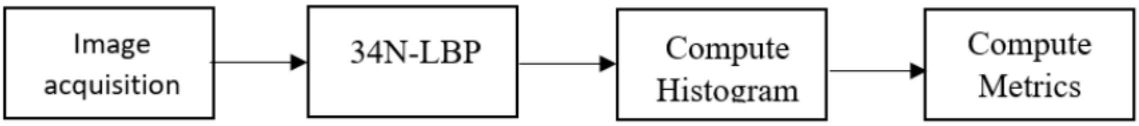

Figure 7 shows the framework for evaluating the proposed 34N-LBPH algorithm on ORL dataset. The gray scale images acquired is passed to the proposed 34N-LBP to extract feature values from the images, then the histogram of the image is computed before using Euclidean distance to compute the Means for Intra-class analysis and Inter-class analysis to determine the robustness of the proposed algorithm under constrained environment. Euclidean distance is applied to the histogram to measure the distances between the data points within the same class and distances between data points of different classes for intra-class and inter-class analysis respectively.

3.3 Framework for Evaluation of The Proposed 34N-LBPH Algorithm on Images Selected from 5_Celebrity Dataset

Figure 8: Framework for Evaluation of the proposed 34N-LBPH Algorithm on 5_Celebrity dataset

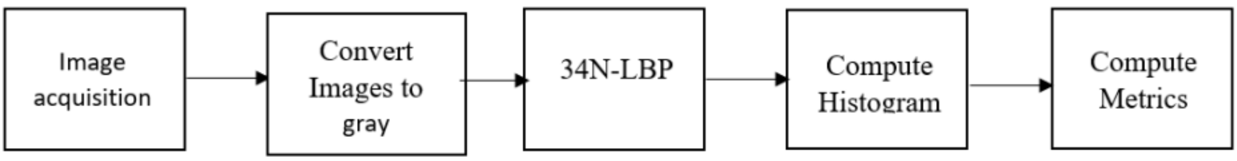

Figure 8 shows the framework for evaluating the proposed 34N-LBPH algorithm on 5_celebrity dataset. The acquired images are converted to gray scale before it is passed to the proposed 34N-LBP to extract feature values from the images, then the histogram of the image is computed before using Euclidean distance to compute the Mean for Intra-class analysis and Inter-class analysis to determine the robustness of the proposed algorithm under unconstrained environment. Euclidean distance is applied to the histogram to measure the distances between the data points within the same class and distances between data points of different classes for intra-class and inter-class analysis respectively

4.0 Results

4.1 Research Tools and Materials

The experiment was executed using a well updated computer system. The overall specification was as follows:

4.1.1 Laptop Specifications:

HP EliteBook 745 G5, Windows 11 Pro 64-Bit, Processor: AMD Ryzen 5 PRO 2500U w/ Radeon Vega Mobile Gfx (8CPUs), 2.0GHz, Graphics: AMD Radeon™ Vega 8 Graphics, System Ram 16 GB, Storage 512 GB, Camera: HP HD camera.

4.1.2 Programming languages and Libraries:

Python: programming language was used to develop the module and conduct the whole experiment.
OpenCV: it was used to identify and verify faces, and also for extraction of features from the histogram.
NumPy: a highly optimized library for numerical operations. It was used for handling arrays to the OpenCV library and also from the OpenCV library.
SciPy: it was used to calculate intra-class and inter-class Means using Euclidian distance
Jupyter IDE: was used to combine code, visualization of results, and analysis.

4.2 Datasets Used for The Experiment

The proposed 34N-LBPH was experimented using images selected from the ORL and 5_Celebrity public datasets to evaluate the performance by conducting feature distribution analysis. The ORL datasets contains facial images taken under constrained environments and the 5_Celebrity dataset contains facial images taken under unconstrained environments. Euclidean distance was applied to the proposed 34N-LBPH to measure the distances between the data points within the same class and distances between data points of different classes for intra-class and inter-class analysis respectively.

4.2.1 Sample Images of The Datasets Used

Figure 8: Sample images of the ORL dataset

Figure 8 illustrates sample images selected from the ORL dataset for testing the proposed algorithm. These images were taken under constrained environment.

Figure 9: Sample images of the 5_Celebrity dataset

Figure 9 illustrates sample images selected from the 5_Celebrity dataset for testing the proposed algorithm. These images were taken under unconstrained environment.

4.3 Experimental Results

To achieve our objective, Intra-class and Inter-class Feature distribution analysis was conducted on constrained and unconstrained environment. By analysing the Mean values of the distribution of LBP features across different images, a robust feature extractor should produce features with a consistent distribution.

4.4 Breakdown of Images Used for Testing the Proposed 34N-LBPH Algorithm

Table 1: Breakdown of Images Used for Testing 34N-LBPH

Datasets	No. of Classes	Image Per Class	Total Images Used
ORL	3	3	9
5_Celebrity	3	3	9

Table 1 represent the breakdown of selected images from the ORL and 5_Celebrity datasets for testing the proposed algorithm.

4.5 Original And 34N-LBP Extracted Images of The 3 Classes On Orl Dataset

Figure 10: Original and 34N-LBP Images for Class A, B, and C on the ORL Dataset

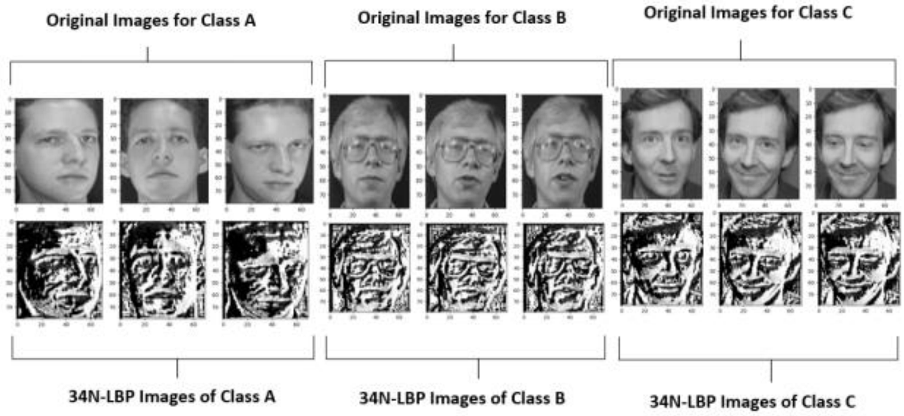

Figure 10 depicts three original images (top row) each and their corresponding 34N-LBP feature extracted images (down row) for the three classes used for testing the 34N-LPBH Algorithm on the ORL dataset.

4.5.1 Histogram Representation of The Images on Orl Datasets

Figure 11: Histogram representation of Class A pictures on ORL

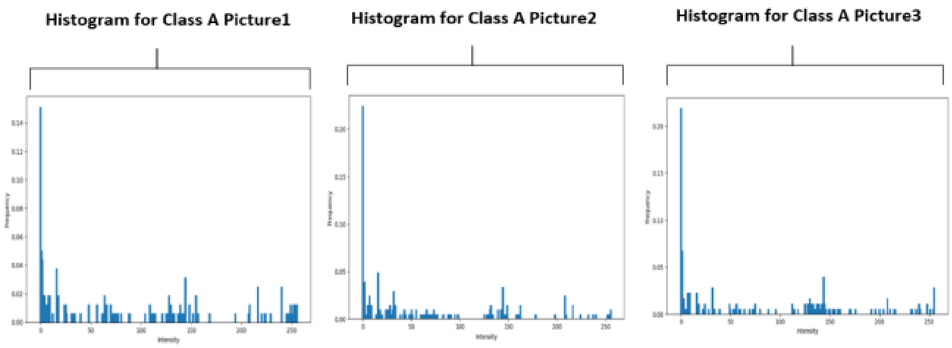

Figure 11 shows the Histogram representing the three pictures of Class A taken from the ORL dataset.

Figure 12: Histogram representation of Class B pictures on ORL

Figure 12 shows the Histogram representing the three pictures of Class B taken from the ORL dataset.

Figure 13: Histogram representation of Class C pictures on ORL

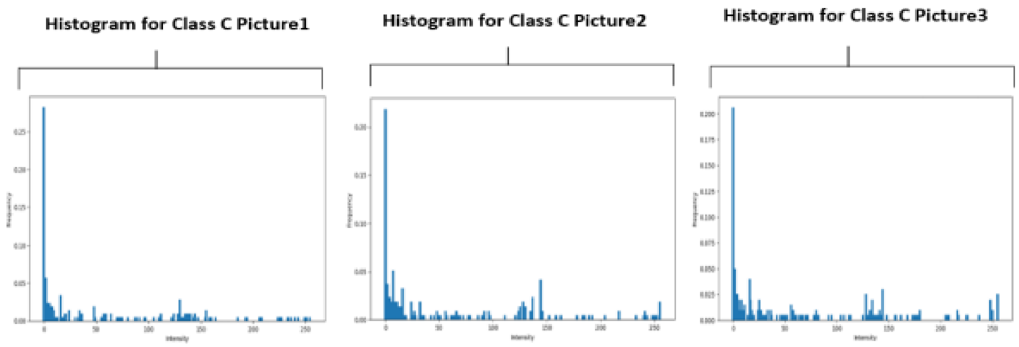

Figure 13 shows the Histogram representing the three pictures of Class C taken from the ORL dataset

4.5.2 Summary of Results for Intra-Class and Inter-Class Distance Analysis on Orl Dataset

Table 2: Intra-Class Summary Statistics on ORL dataset

CLASS	MEAN DISTANCE	STANDARD DEVIATION
A	0.031	0.003
B	0.078	0.007
C	0.101	0.018

Table 3: Inter-Class Summary Statistics on ORL dataset

CLASS	MEAN DISTANCE	STANDARD DEVIATION
Class A Vs. Class B	0.126	0.023
Class A Vs. Class C	0.107	0.015
Class B Vs. Class C	0.115	0.023

Table 2 and 3 shows the summary of statistics of the distances between the data points within the same class and distances between data points of different classes for Intra-class and Inter-class analysis on ORL dataset respectively

4.5 Original And 34N-LBP Extracted Images of the 3 Classes On 5_Celebrity Dataset

Figure 14: Original and 34N-LBP Images for Class A, B, and C on the 5 Celebrity Dataset

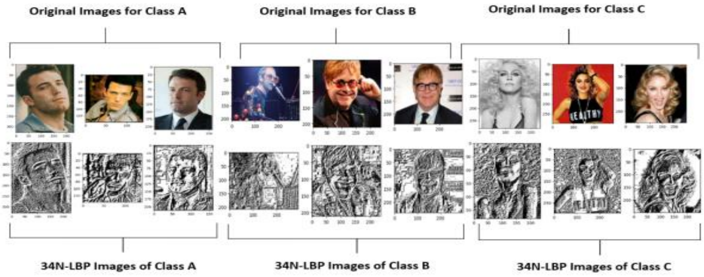

Figure 14 depicts three original images (top row) each and their corresponding 34N-LBP feature extracted images (down row) for the three classes used for testing the proposed 34N-LBPH algorithm on the 5_Celebrity dataset.

4.6.1 Histogram Representation of The Images On 5 Celebrity Datasets

Figure 15: Histogram representation of Class A pictures on 5 Celebrity dataset

Figure 15 shows the Histogram representing the three pictures of Class A taken from the 5 Celebrity dataset.

Figure 16: Histogram representation of Class B pictures on 5 Celebrity dataset

Figure 16 shows the Histogram representing the three pictures of Class B taken from the 5 Celebrity dataset.

Figure 17: Histogram representation of Class C pictures on 5_Celebrity dataset

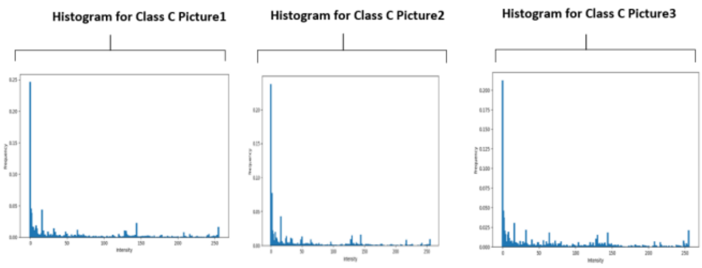

Figure 17 shows the Histogram representing the three pictures of Class C taken from the 5_Celebrity dataset.

4.6.2 Summary of Results for Intra-Class and Inter-Class Distance Analysis On 5_Celebrity Dataset

Table 4: Intra-class Summary Statistics on 5_Celebrity dataset

CLASS	MEAN DISTANCE	STANDARD DEVIATION
A	0.072	0.022
B	0.066	0.023
C	0.048	0.013

Table 5: Inter-class Summary Statistics on 5_Celebrity dataset

CLASS	MEAN DISTANCE	STANDARD DEVIATION
Class A Vs. Class B	0.072	0.027
Class A Vs. Class C	0.057	0.017
Class B Vs. Class C	0.066	0.018

Table 4 and 5 shows the summary of statistics of the distances between the data points within the same class and distances between data points of different classes for Intra-class and Inter-class analysis on 5_Celebrity dataset respectively.

4.7 Comparison of the Performance of The Proposed 34N-LBPH Algorithm Under Constrained and Unconstrained Environment

Table 6: Intra-class comparison of performance under constrained and unconstrained environment

Class	Mean (ORL)	Mean (5_Celeb)	STDV (ORL)	STDV (5_Celeb)
A	0.031	0.072	0.003	0.022
B	0.078	0.066	0.007	0.023
C	0.101	0.048	0.018	0.013

Table 7 shows Intra-class comparison of the performance of the proposed 34N-LBPH algorithm on images under constrained environment to images under unconstrained environment.

Table 7: Inter-class comparison of performance under constrained and unconstrained environment

Class	Mean (ORL)	Mean (5_Celeb)	STDV (ORL)	STDV (5_Celeb)
Class A vs. B	0.126	0.072	0.023	0.027
Class A Vs. C	0.107	0.057	0.015	0.017
Class B Vs. C	0.115	0.066	0.023	0.018

Table 7 shows Inter-class comparison of the performance of the proposed 34N-LBPH algorithm on images under constrained environment to images under unconstrained environment.

5.0 Discussion

Results obtained from the experiment conducted on the ORL and 5_Celebrity datasets representing images taken under constrained and unconstrained environments respectively showed that, the proposed 34N-LBPH algorithm extracts more robust features from images taken under different environments.

Intra-class summary statistics on the ORL dataset showed an average Mean of 0.031, 0.078, and 0.101 for Classes A, B, and C (Table 2), with all 3 Classes having very low Intra-class distance Mean, indicating high similarity or almost identical samples [20].
Inter-class summary statistics on the 5_Celebrity dataset showed an average Mean of 0.126, 0.107, and 0.115 (Table 5) for Class A vs Class B, Class A vs Class C, and Class B vs Class C respectively.
All Class pairs had relatively high Inter-class distance Mean, indicating good separation and distinctness and low Standard Deviation, indicating consistent separation between classes [20].
However, comparing the performance of the proposed 34N-LBPH algorithm on images taken under constrained and unconstrained environments, the results showed that, the algorithm performs better by extracting more robust facial features from images taken under constrained environment than images taken under unconstrained environment (Table 6 and 7).
Nevertheless, the proposed 34N-LBPH algorithm showed acceptable performance for facial feature extraction tasks in both constrained and unconstrained environments.
Future research should focus on incorporating the 34N-LBPH algorithm with CNN for further feature extraction and face recognition.

6.0 Conclusion

In the pursuit to enhance the LBPH algorithm to extract more robust facial features under constrained and unconstrained environments, this study has undertaken a comprehensive investigation into parameter optimization and their relationship in maximizing the algorithm's effectiveness in extracting more robust facial features.
The results obtained from our experimentation and parameter tuning is promising. By selecting appropriate parameter values, we have been able to significantly enhance the algorithm's performance in extracting more robust facial features under different environments.
Notably, the optimized variant, 34N-LBPH algorithm has demonstrated robustness in facial image extraction under constrained environment by handling variations in lighting, facial expressions, and head poses, all of which are pivotal challenges in real-world face recognition scenarios. Also, the optimised variant, 34N-LBPH algorithm has shown greater success in unconstrained environment such as rainy, sunny, foggy, cloudy, and occlusions.
As technology advances and real-world applications grow, the pursuit of enhanced algorithms remains pivotal, and our research contributes to this ongoing journey.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

7.0 List of References

1. Turk, M. A., & Pentland, A. P. (1991, January). Face recognition using eigenfaces. In Proceedings. 1991 IEEE computer society conference on computer vision and pattern recognition (pp. 586-587). IEEE Computer Society.
2. He, X., Yan, S., Hu, Y., Niyogi, P., & Zhang, H. J. (2005). Face recognition using laplacianfaces. IEEE transactions on pattern analysis and machine intelligence, 27(3), 328-340.
3. Belhumeur, P. N., Hespanha, J. P., & Kriegman, D. J. (1997). Eigenfaces vs. fisherfaces: Recognition

using class specific linear projection. IEEE Transactions on pattern analysis and machine intelligence, 19(7), 711-720.
4. Taigman, Y., Yang, M., Ranzato, M. A., & Wolf, L. (2014). Deepface: Closing the gap to human-level performance in face verification. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1701-1708).
5. Schroff, F., Kalenichenko, D., & Philbin, J. (2015). Facenet: A unified embedding for face recognition and clustering. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 815-823).
6. Jain, A. K., Ross, A., & Prabhakar, S. (2004). An introduction to biometric recognition. IEEE Transactions on circuits and systems for video technology, 14(1), 4-20.
7. Baltrušaitis, T., Ahuja, C., & Morency, L. P. (2018). Challenges and applications in multimodal machine learning. The Handbook of Multimodal-Multisensor Interfaces: Signal Processing, Architectures, and Detection of Emotion and Cognition-Volume 2, 17-48.
8. Ojala, T., Pietikäinen, M., & Harwood, D. (1996). A comparative study of texture measures with classification based on featured distributions. Pattern recognition, 29(1), 51-59.
9. Ahonen, Timo, Abdenour Hadid, and Matti Pietikainen (2006). "Face description with local binary patterns: Application to face recognition." IEEE transactions on pattern analysis and machine intelligence 28.12 (2006): 2037–2041.
10. Singh, C., Walia, E., & Kaur, K. P. (2018). Color texture description with novel local binary patterns for effective image retrieval. Pattern recognition, 76, 50-68.
11. Ahonen, T., Hadid, A., & Pietikäinen, M. (2004). Face recognition with local binary patterns. In Computer Vision-ECCV 2004: 8th European Conference on Computer Vision, Prague, Czech Republic, May 11-14, 2004. Proceedings, Part I 8 (pp. 469-481). Springer Berlin Heidelber
12. Liao S, Lei Z, Yi D, Li SZ. A benchmark study of large-scale unconstrained face recognition. In IEEE international joint conference on biometrics 2014 Sep 29 (pp. 1-8). IEEE.
13. Ojala, T., Pietikainen, M., & Maenpaa, T. (2002). Multiresolution gray-scale 67 and rotation invariant texture classification with local binary patterns. IEEE Transactions on pattern analysis and machine intelligence, 24(7), 971-987.
14. Urdal, J. (2016). Image processing and classification of urothelial carcinoma using tissue sample images (Master's thesis, University of Stavanger, Norway).
15. Tiryaki, V. M., Adia-Nimuwa, U., Ayres, V. M., Ahmed, I., & Shreiber, D. I. (2015). Texture-based segmentation and a new cell shape index for quantitative analysis of cell spreading in AFM images. Cytometry Part A, 87(12), 1090-1100.
16. Hu, S., Zhang, B., Lu, J., Jiang, Y., Wang, W., Kong, L., ... & Jiang, T. (2022, September). WideResNet with Joint Representation Learning and Data Augmentation for Cover Song Identification. In Interspeech (pp. 4187-4191).
17. Li, C., Liu, Z., Ren, J., Wang, W., & Xu, J. (2020). A feature optimization approach based on inter-class and intra-class distance for ship type classification. Sensors, 20(18), 5429.
18. Rousseeuw, P. J. (1990). Robust estimation and identifying outliers. Handbook of statistical methods for engineers and scientists, 16, 16-11.
19. Jain, A. K. (1988). Algorithms for clustering data.
20. Hastie, T., Tibshirani, R., Friedman, J. H., & Friedman, J. H. (2009). The elements of statistical learning: data mining, inference, and prediction (Vol. 2, pp. 1-758). New York: springer.

21. Chen, Y., & Hu, H. (2019). Facial expression recognition by inter-class relational learning. IEEE Access, 7, 94106-94117.