



Article

# Evaluating the Performance of Eigenface, Fisherface, and Local Binary Pattern Histogram-Based Facial Recognition Methods under Various Weather Conditions

Md Manjurul Ahsan <sup>1,\*</sup>, Yueqing Li <sup>2,\*</sup>, Jing Zhang <sup>3</sup>, Md Tanvir Ahad <sup>4</sup> and Kishor Datta Gupta <sup>5</sup>

<sup>1</sup> Industrial and Systems Engineering, University of Oklahoma, Norman, OK 73019, USA

<sup>2</sup> Industrial and Systems Engineering, Lamar University, Beaumont, TX 77705, USA

<sup>3</sup> Computer Science, Lamar University, Beaumont, TX 77705, USA; jing.zhang@lamar.edu

<sup>4</sup> School of Aerospace and Mechanical Engineering, University of Oklahoma, Norman, OK 73019, USA; Md.Tanvir.Ahad-1@ou.edu

<sup>5</sup> Computer Science Department, University of Memphis, Memphis, TN 38152, USA; kgupta1@memphis.edu

\* Correspondence: ahsan@ou.edu (M.M.A.); yueqing.li@lamar.edu (Y.L.)

**Abstract:** Facial recognition (FR) in unconstrained weather is still challenging and surprisingly ignored by many researchers and practitioners over the past few decades. Therefore, this paper aims to evaluate the performance of three existing popular facial recognition methods considering different weather conditions. As a result, a new face dataset (Lamar University database (LUDB)) was developed that contains face images captured under various weather conditions such as foggy, cloudy, rainy, and sunny. Three very popular FR methods—Eigenface (EF), Fisherface (FF), and Local binary pattern histogram (LBPH)—were evaluated considering two other face datasets, AT&T and 5\_Celebrity, along with LUDB in term of accuracy, precision, recall, and F1 score with 95% confidence interval (CI). Computational results show a significant difference among the three FR techniques in terms of overall time complexity and accuracy. LBPH outperforms the other two FR algorithms on both LUDB and 5\_Celebrity datasets by achieving 40% and 95% accuracy, respectively. On the other hand, with minimum execution time of 1.37, 1.37, and 1.44 s per image on AT&T, 5\_Celebrity, and LUDB, respectively, Fisherface achieved the best result.

**Keywords:** face recognition; unconstrained weather; time complexity; accuracy



**Citation:** Ahsan, M.M.; Li, Y.; Zhang, J.; Ahad, M.T.; Gupta, K.D.

Evaluating the Performance of Eigenface, Fisherface, and Local Binary Pattern Histogram-Based Facial Recognition Methods under Various Weather Conditions.

*Technologies* **2021**, *9*, 31.

<https://doi.org/10.3390/technologies9020031>

Academic Editor: Manoj Gupta

Received: 14 March 2021

Accepted: 20 April 2021

Published: 27 April 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Facial recognition (FR) is one of the most widely researched fields and has been improved tremendously in the past few decades [1]. As a result, its application has been seen from security systems to the entertainment world. Recently, a lot of organizations within the commercial sector, such as mobile companies, software developers, and entertainment companies are developing and launching FR applications for various purposes. Nevertheless, FR techniques are still challenged by multiple factors such as low light conditions, facial expressions, and bad weather [2]. The lighting/illumination adjustment is one of the most complex and demanding problems in access control applications based on human face recognition [3]. Identifying a face in a crowd, on the other hand, raises significant concerns about individual liberties and ethical problems as well [4]. Jegham et al. (2020) provided an overview and real world challenges regarding vision-based human action recognition [5]. Safdar (2021) addressed illumination, pose variation, and misalignment as the challenging factors currently in facial recognition techniques [6]. With the advent of different FR techniques, it is possible to recognize faces almost 100% accurately in indoor conditions, even with different motion, illumination, and expression. However, no specific research has been conducted that focuses on FR systems under various weather conditions such as foggy, cloudy, and rainy. After reviewing several previous studies, it was observed that facial recognition applications are still challenging in different weather and somehow

ignored. Additionally, during this research work, it was found that there is no single dataset that contains face images captured under different weather conditions—foggy, cloudy, rainy, etc. Therefore, developing a new dataset based on different weather conditions is necessary to conduct such research. Consequently, we developed a new face dataset—Lamar university database (LUDB). Note that there are a lot of datasets for facial recognition, and new datasets are still developing every year. Each dataset contains some limitations and challenges. Since it was difficult to find a dataset containing all face images captured in unconstrained weather, it was easier and reasonable for us to develop a new dataset from the scratch and conduct the experiment. Additionally, two other datasets, 5\_Celebrity and AT&T, were chosen, which are available and free to use for research purposes. AT&T dataset was selected as it was one of the most widely used datasets in the facial recognition research field. The 5\_Celebrity dataset was chosen from the Kaggle repository.

As a means of analyzing the performance of FR techniques under various weather conditions, we chose three popular FR algorithms, namely Eigenface (EF), Fisherface (FF), and Local binary pattern histogram (LBPH), provided by OpenCV. In this study, these three FR techniques were evaluated using accuracy, precision, recall, F1 score, and execution time. To the best of our knowledge, this is the first time that the overall performance of FR methods, provided by OpenCV, has been evaluated under various weather conditions.

The EF, FF and LBPH algorithms have several advantages over other face recognition techniques [2,7]:

- Easy to implement and most widely used
- Demonstrate more stable performance on small dataset
- Run smoothly on an average CPU-based computer, chrome book, tab, and mobile device and require less computation time

Deep learning has recently become popular in the image processing field (e.g., ultrasound imaging). Recently, real-time semantic segmentation has gained much popularity using deep learning approaches due to its faster processing [8]. However, such techniques require a large amount of data for better performance. On the other hand, general machine learning-based techniques such as principal component analysis (PCA) and local binary pattern histogram (LBPH) can perform well with a minimal dataset [9]. For instance, Ahn et al. (2006) achieved 79% and 51% accuracy using weighted LBP and nonweighted LBP, respectively, for the FERET dataset [10]. Abuzneid and Mahmood (2018) used several methods, among them LBP + KNN approach. They tested their proposed method on two different dataset Yale and ORL. They achieved 100% accuracy on the Yale dataset when 90% of the data were used for the training and remaining 10% used for testing [11]. Adjabi et al. (2020) used multi-block color-Binarized statistical images for single sample face recognition [12].

Recently, Adeshina et al. (2020) showed that haar-like and LBP-based features outperformed deep learning-based methods such as ResNet101 and ResNet50 in terms of computation time for a smaller dataset [13]. As a result, due to the limited number of subjects and the dataset's size, big data-driven methods such as convolutional neural network (CNN) and recurrent neural network (RNN) were ignored for the initial pilot test.

The rest of the paper is organized as follows. Section 2 presents a summary of previous research and a literature review. Section 3 introduces a dataset used in the analysis of facial recognition. Sections 4–6 represent methodology, results, and discussion, respectively. Finally, in Section 7, a conclusion is drawn based on the overall study and the possibility of future work.

## 2. Literature Review

### 2.1. Indoor vs. Unconstrained Environment

Most facial recognition (FR) studies are conducted in indoor conditions [14–19]. However, none of these studies considered any sort of outdoor situation. For example, Salh and Nayef (2013) used ORL dataset images, which contain various pose variations and illumina-

tions [14]. However, the ORL dataset does not contain any images taken in unconstrained environments such as foggy and cloudy weather.

Qi et al. (2013) considered factors such as pose variance and expression but failed to consider the unconstrained environment [18]. Even some of the useful techniques described by Kim et al. (2002) and Deshpande and Ravishankar (2017) do not contain any information on how those methods would perform in unconstrained situations [20].

Additionally, unconstrained images contain different kinds of noise, which need to be reduced to allow the program to recognize the face. Some research addresses removing noise from images [21,22]. For example, Tiwari and Khandelwa (2017) used denoising techniques such as Fast Fourier Transform (FFT) methods to remove fog from an image [21]. The most efficient image denoising methods are based on wavelets and not FFT [23]. Zhang et al. (2012) tried to reduce the noise in images using filters such as Lee filters, SRAD, and s-function, but, again, they did not consider any real-time scenario [22].

Ruiz-del-Solar et al. (2009) indicated that outdoor experiments can pose significant challenges. They showed that most methods work fine on the FERET dataset in the indoor illumination conditions except SD [24]. However, in outdoor illumination conditions, the performance of all methods decreased dramatically.

Hermosilla, Ruiz-del-Solar, Verschae, and Correa (2012) used a database which allows for evaluating the methods in real-world conditions such as pose, accessories, and occlusions. They showed low accuracy in both indoor and unconstrained environments [25].

Recently, unmanned aerial vehicles (UAVs) show potential for face recognition and port security detection. Zhao et al. (2020) used three parameters (distance, height, and angle) to test UAV feasibility. The method was evaluated under daylight and various weather conditions [26].

In summary, while a lot of research has been conducted in indoor situations, real-time facial recognition in outdoor conditions is somewhat ignored due to low-quality results, and they require further investigation.

## 2.2. Effect of Fog on Real Time Images

Very few studies have considered face recognition with foggy, distorted images [27–29].

Schwarzlmüller et al. (2011) used CNN methods with foggy distorted images [30]. However, Surekha and Kumar (2016) and Nousheen and Kumar (2016) showed that their method (Retinex algorithm) outperformed the earlier study conducted by Schwarzlmüller et al. (2011) [27,28,30]. None of these studies showed how these techniques could improve the accuracy in real-time facial recognition.

Mohanram et al. (2014) used morphological operation and transmission ratio for distorted images [31]. Even though their results were quite satisfactory, their study did not mention anything regarding real-time object recognition.

Shabna and Manikandababu (2016) showed some promising results by using a haze removal algorithm for surveillance video [32]. However, the real drawback of their method is that they did not mention the accuracy rate and processing time.

After analyzing several papers, it is clear that real-time face recognition in foggy weather has never been studied significantly. Thus, it is necessary to study face recognition in foggy weather in order to understand the performance of existing facial recognition methods.

## 2.3. Effect of Rain on Real-Time Image

The impact of rain on the image causes loss of image contrast and color fidelity. As a result, it is difficult to make the image visible with adequate quality. While there is evidence of removing the effect of rain on video or on data with multiple frame images in literature [33,34], the studies only focused on a single image [35,36]. However, none of these studies have considered the real-time face recognition in rainy weather.

Kim et al. (2013) proposed single image de-raining techniques using an adaptive nonlocal means filter [37]. Although their proposed algorithm could remove rain streaks efficiently, it is not stable for real-time image processing. Kang et al. (2011) proposed an

image decomposition method (morphological component analysis), which performed well for simple structures but was not as effective for complex frameworks [38]. According to Luo et al. (2016), the bad performance could be due to the usage of bilateral filtering while processing the image, which erases specific details of the picture and makes it difficult to recover later [36].

Eigen et al. (2013) used a deep learning-based method for de-rained images [39]. Unlike other methods proposed by Kang et al. (2011) and Pei et al. (2014), deep learning does not rely on image pre- or post-processing module [38,40]. However, this method also has a limitation on the processing of complex image structure [36].

Barnum et al. (2010) suggested that detecting individual streaks in the rain is difficult even with an accurate model [33]. They proposed a better technique compared to pixel-based or patched-based methods, which showed better results with both image and camera motions. Bossu et al. (2011) proposed a Gaussian mixture model to detect the existence of rain or snow where histogram orientation methods were used [33,34]. Brewer and Liu (2008) used the shape characteristics of rain to identify and remove rain from video [41]. Neither study considered the real-time face recognition. Garg and Nayar (2004) presented a comprehensive analysis of rain on the image [42]. They used a correlation model to detect and remove the rain from the video. Although their study included moving objects and time-varying texture, it did not indicate whether their techniques were performed on the human face.

Liu and Piao (2016) showed that using rain characteristics to de-rain images is one of the best solutions [43]. Li et al. (2016) also dealt with rain streak removal from a single image [44]. According to their study, while dictionary learning methods and low-rank structure methods can improve the visibility of the image, they leave too many rain streaks in the background. Thus, in real time FR, if there are too many rain streaks in the background, there is a high chance of getting more errors, which will reduce the accuracy rate.

Li et al. (2016) used a simple patched-based method for both the background and rain layers, based on the Gaussian mixture model [44]. Their methods work better than any other technique when it comes to de-raining the image. However, Barnum et al. (2010) suggested that their approach is better when compared to pixel-based or patched-based methods, which is contradictory to the study conducted by Li et al. (2016) [33,44].

Lin et al. (2017) studied rain-removal systems, for their experiment on advanced driving assistance systems (ADAS) [45]. They proposed an orientation-adaptive non-local mean (OA-NLM) filter algorithm to improve rain removal performance. Their study also improved computational cost using orientation-adaptive and computing module (CPM), which is absent in other referenced literature [33,36,37].

Apart from this, their method implemented an FPGA system to improve the image processing speed. They showed that the proposed method reduced 94.21% of the execution time of image processing with the hardware–software design. Although Lin et al. (2017) dealt with ADAS, this study could potentially be implemented in real-time face recognition [45].

From the above discussion, it is clear that many studies have been conducted and are still ongoing related to noise reduction in image processing. While several techniques and algorithms are being used to improve image visibility, none of the studies solely focused in developing the application for real-time FR under rainy weather. As a result, without a complete investigation, it is difficult to conclude whether image filtering or a simple algorithm would be best for FR in a real-time situation during rainy weather.

To sum up, while facial recognition systems are studied extensively, unconstrained environments such as foggy, cloudy, and rainy conditions have not been given enough attention over the past few decades. Thus, one of the primary purposes of this research was to analyze the possibility to recognize a face in unconstrained weather and represent the overall performance of the selected techniques (Eigenface, Fisherface, and LBPH) using accuracy, precision, recall, F1 score, and execution time.

### 3. Dataset Creation

There is no existing dataset that contains face images captured in unconstrained weather. Large web-based dataset repositories such as Imagenet contain some images, but, unfortunately, those images are not enough for the experiment. Consequently, a new face dataset named “LUDB” was created for this study. For performance analysis, two other popular dataset, AT&T and 5\_Celebrity, were used with the LUDB dataset. The 5\_Celebrity dataset was taken from the popular machine learning website Kaggle [2]. The Scikit learn tool was used to calculate the accuracy, precision, recall, F1 score, and execution time of EF, FF, and LBPH.

#### *Dataset Description*

LUDB consists of 250 images of 17 participants—all images captured with different facial expressions, different illumination conditions, and different occlusions. Many recent studies show that a small dataset can be used with machine learning- and deep learning-based approaches. For instance, Narin et al. (2020) only used 100 images for the chest radiograph’s binary classification [46].

Our dataset includes fifteen male and two female participants and contains single images captured during sunny, cloudy, and rainy weather. In this study, demographic characteristics such as age, gender, and ethnicity were excluded; instead, we only focused on face images captured under different weather. All images were captured using a Note 8 phone that includes a 12 MP camera. Since the dataset was developed during summer, it was difficult to capture images considering foggy weather. Thus, some of the images were modified by adding artificial fog. The artificial fog was generated using adobe photoshop by adjusting the illumination and contrast value. Initially, the image was loaded. Then, a light gray shade color was chosen for the foreground color. After that, the horizontal line across the image was selected where we wanted the fog to appear. The opacity range was set to 50% to create the thickness of the fog. Finally, a Gaussian blur filter was used with the adjusted radius 5 px, as discussed in [47].

The number of images of each person varies, but each person has at least seven photos. All images are  $500 \times 500$  pixels in size. Figure 1 illustrates some examples of the face images from the dataset.



Figure 1. Image examples from LUDB dataset.

#### Database Comparison

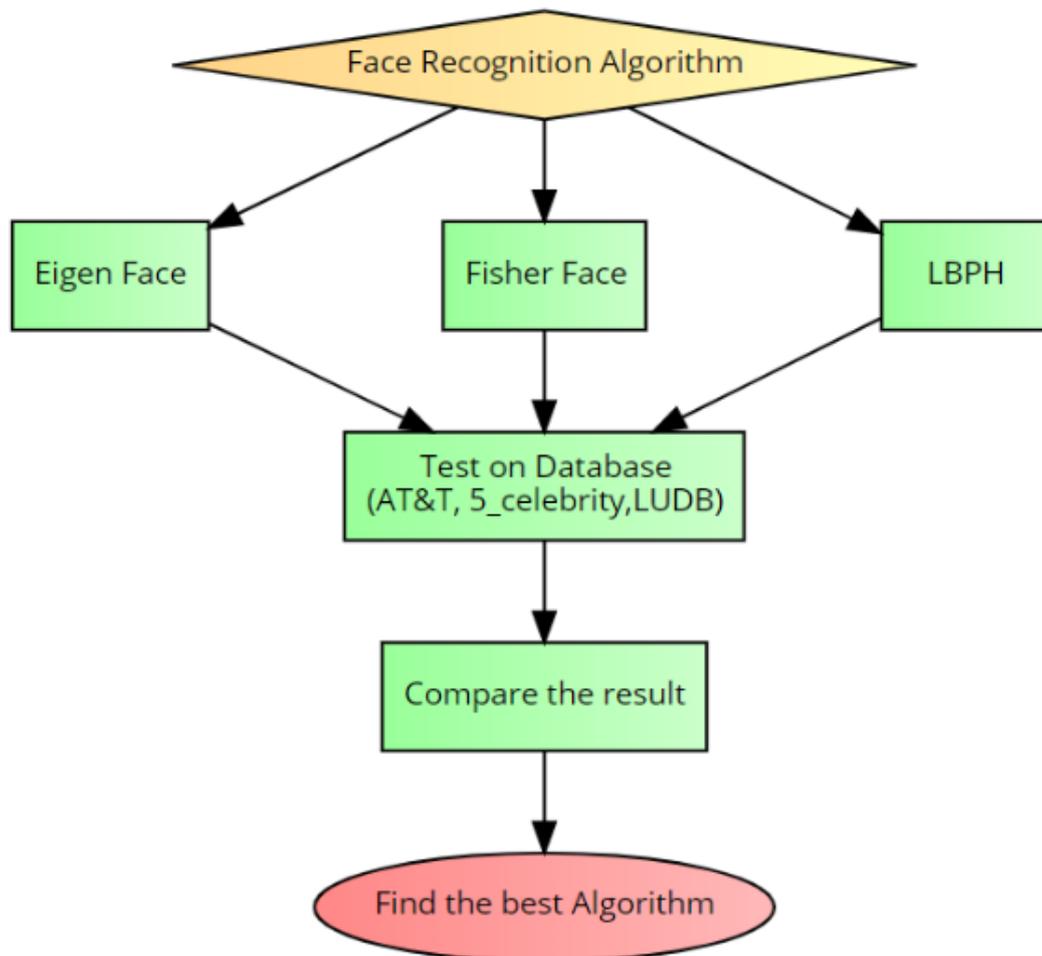
Table 1 compares different datasets used in this study. In Table 1, we can see that no specific image size was used to create the dataset. Some of the data use color images; others rely on gray scale images. For the LUDB dataset, we considered  $500 \times 500$  pixel images to make sure that the image contains enough information to extract. However, as a standard procedure, all the images were resized into  $100 \times 100$  pixels using the Python image library (PIL) tool Image “resize()” methods during the training and testing phase [1].

**Table 1.** Comparison between different dataset used in this study.

| Database    | Subjects | Images | Size (pixel) | Unconstrained |
|-------------|----------|--------|--------------|---------------|
| AT&T        | 40       | 400    | 92 × 112     | No            |
| 5_Celebrity | 5        | 118    | Not equal    | Yes           |
| LUDB        | 17       | 250    | 500 × 500    | Yes           |

#### 4. Methodology

To execute the experiment, a good updated computer and programming software with a web camera was used. The overall specification was as follows: Laptop Specifications, Dell Inspiron 7579, Window 10, processor (CPU) Intel Core I and 7500U 7th Generation, graphics Intel HD 620 Integrated graphics, System Ram 16 GB, Storage 512 GB, Camera Front, and VGA. As a software language, Python was used to conduct the whole experiment. Figure 2 demonstrates the overall experiment procedure of this study.

**Figure 2.** Flow chart of the Facial recognition experiment used in this study.

##### 4.1. Eigenfaces

Sirovich and Kirby used eigenfaces and principle component analysis (PCA) to identify face images [48]. One of PCA's main advantages is its low sensitivity to noise and its ability to reduce the dimension. The Euclidean distance method is used to calculate the distance between the eigenvector between eigenfaces. If the distance is small, then the subject is identified, whereas too large distance indicates that the model requires more training to identify the subject.

#### 4.2. Fisherfaces

Fisherface (FF) is the modified version of Eigenface (EF) method. In this method, the total average face per class is calculated. A very detailed explanation of EF can be found in the work by Delbiaggio (2017) [49].

#### 4.3. Local Binary Pattern Histogram

The LBP originally appeared as a texture descriptor. The operator allocates a label to each pixel value of the image by thresholding a  $3 \times 3$  neighborhood with a central pixel value, which counts as a binary number (Lopez, 2010) [50].

### 5. Results

In the experiment, three datasets (AT&T, 5\_Celebrity, and LUDB) were tested using Eigenface (EF), Fisherface (FF), and Local binary pattern histogram (LBPH). AT&T, LUDB, and 5\_Celebrity datasets contain 400, 250, and 100 images, respectively. However, 100 images were used from each dataset to keep the data ratio equal and to avoid any data imbalance-related complexity. K-fold cross validation (where  $k = 10$ ) was used to evaluate the overall result, and the final results are presented by averaging all of those folds. As the datasets are small, the 95% confidence interval is used to present the statistical outcomes [51,52]. The overall performance was measured using accuracy, precision, recall, F1 score, and execution time, which were calculated using the following formulas:

$$Accuracy = \frac{t_p + t_n}{t_p + t_n + f_p + f_n} \quad (1)$$

$$Precision = \frac{t_p}{t_p + f_p} \quad (2)$$

$$Recall = \frac{t_p}{t_n + f_p} \quad (3)$$

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (4)$$

Table 2 summarizes the accuracy, precision, recall, and F1 score with 95% confidence interval (CI) for Eigenface (EF), Fisherface (FF), and LBPH. The accuracy for FF is the highest at 100%. The accuracy for LBPH is the lowest at  $98\% \pm 1.24\%$ . Additionally, the highest F1 score was achieved for FF compared to the other two algorithms. To sum up, the performance of the algorithm in terms of accuracy and F1 score on the AT&T face dataset is ranked as follows: FF > EF > LBPH.

The execution time for EF is the highest at 2.74 s. On the other hand, FF took only 1.37 s to train and predict. From the result, the execution time of all three face recognition (FR) algorithms ranked as follows: FF < LBPH < EF. Considering both accuracy and execution time, FF recognition algorithm outperforms EF and LBPH face recognition algorithms.

**Table 2.** Accuracy, precision, recall, F1 score, and execution time on AT&T dataset.

| Algorithm  | Accuracy          | Precision        | Recall           | F1 Score          | Execution Time |
|------------|-------------------|------------------|------------------|-------------------|----------------|
| EigenFace  | $99\% \pm 0.87\%$ | $0.985 \pm 0.01$ | $0.99 \pm 0.009$ | $0.987 \pm 0.010$ | 2.74 s         |
| FisherFace | 100%              | 1                | 1                | 1                 | 1.37 s         |
| LBPH       | $98\% \pm 1.24\%$ | $0.97 \pm 0.015$ | $0.98 \pm 0.012$ | $0.97 \pm 0.015$  | 1.84 s         |

Table 3 summarizes the accuracy, precision, recall, F1 score, and execution time for EF, FF, and LBPH on 5\_Celebrity dataset. The accuracy for LBPH is the highest at  $40\% \pm 6.8\%$ . The accuracy for EF is the lowest at  $33\% \pm 7.18\%$ . Additionally, LBPH also achieved highest F1 score ( $0.358 \pm 0.07$ ). The performance of the algorithm in terms of accuracy and F1 score on 5\_Celebrity face dataset can be ranked as follows: LBPH > FF > EF. Considering

accuracy, LBPH outperforms EF and FF. Considering execution time, FF outperforms EF and LBPH.

**Table 3.** Accuracy, precision, recall, F1 score, and execution time on 5\_Celebrity dataset.

| Algorithm  | Accuracy        | Precision         | Recall           | F1 Score          | Execution Time |
|------------|-----------------|-------------------|------------------|-------------------|----------------|
| EigenFace  | 33% $\pm$ 7.18% | 0.318 $\pm$ 0.072 | 0.33 $\pm$ 0.072 | 0.305 $\pm$ 0.073 | 2.73 s         |
| FisherFace | 37% $\pm$ 6.96% | 0.398 $\pm$ 0.07  | 0.37 $\pm$ 0.07  | 0.352 $\pm$ 0.071 | 1.37 s         |
| LBPH       | 40% $\pm$ 6.8%  | 0.36 $\pm$ 0.07   | 0.4 $\pm$ 0.068  | 0.358 $\pm$ 0.07  | 1.95 s         |

Table 4 summarizes the accuracy, precision, recall, F1 score, and execution time for Eigenface (EF), Fisherface (FF), and LBPH on LUDB. The accuracy for LBPH is the highest at 95%  $\pm$  1.96% and EF is the lowest at 84%  $\pm$  3.51%. The highest F1 score is observed for LBPH (0.934  $\pm$  0.023). The performance of the algorithms on LUDB dataset in terms of accuracy and F1 score can be ranked as follows: LBPH > EF > FF. The execution time for EF is the highest (3.0167 s). On the other hand, FF took 1.44 s to train and predict. From the result, the execution time of the three face recognition algorithms can be ranked as follows: FF < LBPH < EF. Considering accuracy, LBPH outperforms EF and FF. Considering execution time, FF outperforms both LBPH and EF methods.

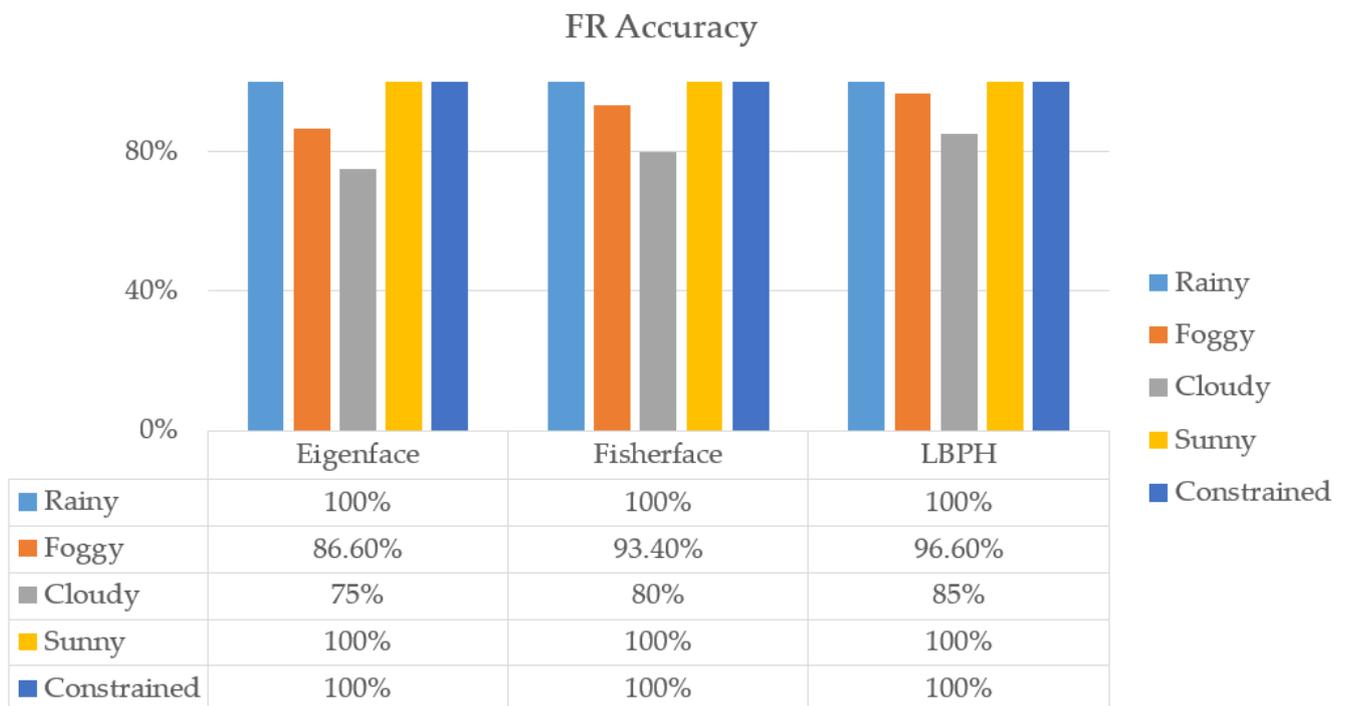
**Table 4.** Accuracy, precision, recall, F1 score, and execution time on LUDB.

| Algorithm  | Accuracy        | Precision         | Recall           | F1 Score          | Execution Time |
|------------|-----------------|-------------------|------------------|-------------------|----------------|
| EigenFace  | 86% $\pm$ 3.3%  | 0.803 $\pm$ 0.04  | 0.86 $\pm$ 0.033 | 0.82 $\pm$ 0.037  | 3.0167 s       |
| FisherFace | 84% $\pm$ 3.51% | 0.781 $\pm$ 0.041 | 0.84 $\pm$ 0.035 | 0.801 $\pm$ 0.04  | 1.44 s         |
| LBPH       | 95% $\pm$ 1.96% | 0.925 $\pm$ 0.024 | 0.95 $\pm$ 0.02  | 0.934 $\pm$ 0.023 | 2.09 s         |

## 6. Discussion

The overall accuracy of all facial recognition (FR) methods were analyzed in foggy, cloudy, rainy, sunny, and specific conditions (images captured when the weather was not sunny or cloudy). All three FR methods show 100% accuracy in recognizing face images captured under rainy, sunny, and constrained weather. However, the accuracy differs in foggy and cloudy weather. In foggy weather, LBPH outperformed all other algorithms by achieving an accuracy of around 96.60%. Eigenface had the lowest performance in terms of accuracy (86.60%). Similarly, in cloudy weather, LBPH had the best performance, while Eigenface displayed the lowest performance in terms of accuracy, as shown in Figure 3.

The accuracies of EF, FF, and LBPH were also measured on three datasets. On the AT&T dataset, Fisherfaces outperformed the other two FR methods. On 5\_Celebrity and LUDB datasets, LBPH outperformed EF and FF algorithms. Even though LBPH showed low accuracy on the AT&T dataset, it showed the highest accuracy on 5\_Celebrity and LUDB datasets, which contain unconstrained images. Thus, we found LBPH is the optimal algorithm for further experiment. Among all three FR methods, EF took the longest execution time at 2.74, 2.73, and 3.01 s, respectively, on AT&T, 5\_Celebrity, and LUDB datasets. FF took the lowest execution time on all three datasets. Based on the time complexity, the three FR methods can be ranked as follows: FF > LBPH > EF.



**Figure 3.** FR algorithm performance in unconstrained environment.

Three FR experiments with three datasets, namely AT&T, 5\_Celebrity, and LUDB dataset, were conducted. One main point from these experiments is that, among the three datasets, the lowest accuracy in context with any facial recognition method was found on 5\_Celebrity and LUDB datasets. While this experiment may help researchers and practitioners understand the performance of three FR methods under different weather conditions, it is also concluded with some limitations. The limitations can be summarized as follows, which shall be addressed in future work:

- Since the experiment was conducted during the summer, it was impossible to develop a dataset containing images captured under foggy weather. Thus, an adjustment was made using artificial fog on the image and overall experiments were conducted.
- The performance of all three algorithms on the 5\_Celebrity dataset was very poor due to the many pose variations and illumination conditions. Several possible FR techniques are available such as CNN and RNN that could be used for this experiment and might obtain higher accuracy than the results presented in this paper.
- The dataset was relatively small and contained only 15 male and 2 female participants. Although the number of images was enough to conduct the pilot test, due to the lack of female participants, the effect of gender on face recognition was ignored as well.
- The dataset contains images with different emotions as well, but the results are presented considering all the images instead of referring to individual emotions and face angles.

## 7. Conclusions

In this paper, a new face dataset (LUDB) is presented containing images captured in different weathers (i.e., artificially foggy, cloudy, rainy, and sunny). The experimental results show that facial recognition (FR) in an unconstrained situation using Eigenface (EF), Fisherface (FF), and Local binary pattern histogram (LBPH) is challenging but possible. LBPH showed the highest accuracy on both LUDB (images captured in different weather) and 5\_celebrity dataset (containing unconstrained images), 40% and 95%, respectively. FF showed the best result on AT&T, 5\_Celebrity, and LUDB in terms of execution time



15. Marami, E.; Tefas, A. Face detection using particle swarm optimization and support vector machines. In *Hellenic Conference on Artificial Intelligence*; Springer: Berlin, Germany, 2010; pp. 369–374.
16. Chen, L.; Zhou, C.; Shen, L. Facial expression recognition based on SVM in E-learning. *Ieri Procedia* **2012**, *2*, 781–787. [[CrossRef](#)]
17. Li, R.s.; Lee, F.f.; Yan, Y.; Qiu, C. Face Recognition Using Vector Quantization Histogram and Support Vector Machine Classifier. *DEStech Trans. Comput. Sci. Eng.* **2016**. [[CrossRef](#)]
18. Qi, Z.; Tian, Y.; Shi, Y. Robust twin support vector machine for pattern classification. *Pattern Recog.* **2013**, *46*, 305–316. [[CrossRef](#)]
19. Fontaine, X.; Achanta, R.; Süsstrunk, S. Face recognition in real-world images. In Proceedings of the 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA, USA, 5–9 March 2017; pp. 1482–1486.
20. Kim, K.I.; Kim, J.H.; Jung, K. Face recognition using support vector machines with local correlation kernels. *Int. J. Pattern Recog. Artif. Intell.* **2002**, *16*, 97–111. [[CrossRef](#)]
21. Tiwari, R.; Khandelwal, A. Fog Removal Technique with Improved Quality through FFT. *Int. J. Recent Trends Eng. Res.* **2017**, *3*. [[CrossRef](#)]
22. Zhang, Y.; Cheng, H.D.; Huang, J.; Tang, X. An effective and objective criterion for evaluating the performance of denoising filters. *Pattern Recog.* **2012**, *45*, 2743–2757. [[CrossRef](#)]
23. Ouahabi, A. A review of wavelet denoising in medical imaging. In Proceedings of the 2013 8th International Workshop on Systems, Signal Processing and their Applications (WoSSPA), Algiers, Algeria, 12–15 May 2013; pp. 19–26.
24. Ruiz-del Solar, J.; Verschae, R.; Correa, M. Recognition of faces in unconstrained environments: A comparative study. *EURASIP J. Adv. Signal Process.* **2009**, *2009*, 1–19. [[CrossRef](#)]
25. Hermosilla, G.; Ruiz-del Solar, J.; Verschae, R.; Correa, M. A comparative study of thermal face recognition methods in unconstrained environments. *Pattern Recognit.* **2012**, *45*, 2445–2459. [[CrossRef](#)]
26. Zhao, R.; Zhu, Z.; Li, Y.; Zhang, J.; Zhang, X. Use a UAV System to Enhance Port Security in Unconstrained Environment. In *International Conference on Applied Human Factors and Ergonomics*; Springer: Berlin, Germany, 2020; pp. 78–84.
27. Surekha, N.; Naveen Kumar, J. An improved fog-removing method for the traffic monitoring image. *Int. J. Mag. Eng. Technol. Manag. Res.* **2016**, *3*, 2061–2065.
28. Nousheen, S.; Kumar, S. Novel Fog-Removing Method for the Traffic Monitoring Image. 2016. Available online: <https://core.ac.uk/download/pdf/228549366.pdf> (accessed on 20 April 2021).
29. Deshpande, D.; Kale, V. Analysis of the atmospheric visibility restoration and fog attenuation using gray scale image. In Proceedings of the Satellite Conference ICSTSD 2016 International Conference on Science and Technology for Sustainable Development, Kuala Lumpur, Malaysia, 24–26 May 2016; pp. 32–37.
30. Schwarzmüller, C.; Al Machot, F.; Fasih, A.; Kyamakya, K. Adaptive contrast enhancement involving CNN-based processing for foggy weather conditions & non-uniform lighting conditions. In Proceedings of the Joint INDS'11 & ISTET'11, Klagenfurt am Wrthersee, Austria, 25–27 July 2011; pp. 1–10.
31. Mohanram, S.; Aarthi, B.; Silambarasan, C.; Hephzibah, T.J.S. An optimized image enhancement of foggy images using gamma adjustment. *Int. J. Adv. Res. Electr. Commun. Eng.* **2014**, *3*, 155–159.
32. Shabna, D.; Manikandababu, C. An efficient haze removal algorithm for surveillance video. *Int. J. Innov. Res. Sci. Eng. Technol.* **2016**, *5*.
33. Barnum, P.C.; Narasimhan, S.; Kanade, T. Analysis of rain and snow in frequency space. *Int. J. Comput. Vis.* **2010**, *86*, 256. [[CrossRef](#)]
34. Bossu, J.; Hautiere, N.; Tarel, J.P. Rain or snow detection in image sequences through use of a histogram of orientation of streaks. *Int. J. Comput. Vis.* **2011**, *93*, 348–367. [[CrossRef](#)]
35. Zhao, W.; Chellappa, R.; Phillips, P.J.; Rosenfeld, A. Face recognition: A literature survey. *ACM Comput. Surv.* **2003**, *35*, 399–458. [[CrossRef](#)]
36. Luo, Y.; Xu, Y.; Ji, H. Removing rain from a single image via discriminative sparse coding. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 3397–3405.
37. Kim, J.H.; Lee, C.; Sim, J.Y.; Kim, C.S. Single-image deraining using an adaptive nonlocal means filter. In Proceedings of the 2013 IEEE International Conference on Image Processing, Melbourne, VIC, Australia, 15–18 September 2013; pp. 914–917.
38. Kang, L.W.; Lin, C.W.; Fu, Y.H. Automatic single-image-based rain streaks removal via image decomposition. *IEEE Trans. Image Process.* **2011**, *21*, 1742–1755. [[CrossRef](#)]
39. Eigen, D.; Krishnan, D.; Fergus, R. Restoring an image taken through a window covered with dirt or rain. In Proceedings of the IEEE International Conference on Computer Vision, Sydney, Australia, 1–8 December 2013; pp. 633–640.
40. Pei, S.C.; Tsai, Y.T.; Lee, C.Y. Removing rain and snow in a single image using saturation and visibility features. In Proceedings of the 2014 IEEE International Conference on Multimedia and Expo Workshops (ICMEW), Chengdu, China, 14–18 July 2014; pp. 1–6.
41. Brewer, N.; Liu, N. Using the shape characteristics of rain to identify and remove rain from video. In *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*; Springer: Berlin, Germany, 2008; pp. 451–458.
42. Garg, K.; Nayar, S.K. Detection and removal of rain from videos. In Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Washington, DC, USA, 27 June–2 July 2004; Volume 41, p. 1.

43. Liu, S.; Piao, Y. A novel rain removal technology based on video image. In Proceedings of the Selected Papers of the Chinese Society for Optical Engineering Conferences, International Society for Optics and Photonics, Changchun, China, July 2016; Volume 10141, p. 101411O.
44. Li, Y.; Tan, R.T.; Guo, X.; Lu, J.; Brown, M.S. Rain streak removal using layer priors. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2736–2744.
45. Lin, Y.Y.; Hsiung, P.A. An early warning system for predicting driver fatigue. In Proceedings of the 2017 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW), Taipei, Taiwan, 12–14 June 2017; pp. 283–284.
46. Narin, A.; Kaya, C.; Pamuk, Z. Automatic detection of coronavirus disease (covid-19) using X-ray images and deep convolutional neural networks. *arXiv* **2020**, arXiv:2003.10849.
47. Anzalone, C. How to Make Fog with Photoshop. 2019. Available online: <https://yourbusiness.azcentral.com/chalky-look-photoshop-10803.html> (accessed on 20 April 2021).
48. üge Çarıkçı, M.; Özen, F. A face recognition system based on eigenfaces method. *Procedia Technol.* **2012**, *1*, 118–123. [CrossRef]
49. Delbiaggio, N. A Comparison of Facial Recognition's Algorithms. 2017. Available online: <https://www.theseus.fi/handle/10024/132808> (accessed on 20 April 2021).
50. Sánchez López, L. Local Binary Patterns Applied to Face Detection and Recognition. 2010. Available online: <https://upcommons.upc.edu/handle/2099.1/10772> (accessed on 20 April 2021).
51. Ahsan, M.M.; Ahad, M.T.; Soma, F.A.; Paul, S.; Chowdhury, A.; Luna, S.A.; Yazdan, M.M.S.; Rahman, A.; Siddique, Z.; Huebner, P. Detecting SARS-CoV-2 from Chest X-ray using Artificial Intelligence. *IEEE Access* **2021**, *9*, 35501–35513. [CrossRef]
52. Ahsan, M.M.; Gupta, K.D.; Islam, M.M.; Sen, S.; Rahman, M.; Shakhawat Hossain, M. COVID-19 Symptoms Detection Based on NasNetMobile with Explainable AI Using Various Imaging Modalities. *Mach. Learn. Knowl. Extract.* **2020**, *2*, 490–504. [CrossRef]