Video-based Blepharospasm Detection

Anonymous WACV Applications Track submission

Paper ID 1937

Abstract

This paper presents how Blepharospasm, a neurological disorder, can be detected from a patient's video. Persons suffering from this disorder suddenly squeeze their eyes involuntarily while blinking. We call them tight blinks. Since there has been no prior work on modeling such a phenomenon, we design multiple features which model the phenomenon and helps detect Blepharospasm. We observe that these tight blinks are usually longer than normal ones. It prompts us to incorporate temporal analysis of blinks alongside spatial analysis in our feature design. We aggregate these analyses at video level for yielding our proposed features. Our experiments on videos downloaded from the internet demonstrate that our features are not only useful in detecting Blepharospasm via supervised learning but also via unsupervised learning.

1. Introduction

Digital healthcare is a significant application area of video analytics. Neurological disorders [2, 4], which affect the nervous system—including the brain, spinal cord, and nerves—often exhibit visible symptoms that can be modeled for automated diagnosis. Blepharospasm is a neurological disorder characterized by sudden, involuntary squeezing of the eyes, referred to as tight blinks. While extensive research has been conducted on normal blink detection, tight blinks remain underexplored. The complete deformation of the eyes during tight blinks poses a substantial challenge for spatial analysis. In this paper, we propose a novel video analytics-based approach for detecting this disorder.

Blepharospasm is a rare neurological condition characterized by involuntary, repeated, and forcible contractions of the eyelids and forehead muscles [1]. Although the exact cause remains unclear, it is often associated with dysfunction in the Basal Ganglia, a region of the brain.

We observe that the distance between the eye center and eyebrows varies significantly during blinks involving such contractions. Notably, not all blinks in Blepharospasm patients exhibit this contraction; however, this phenomenon occurs frequently among these patients. This paper models this observation to develop a video-level feature essential for constructing an eye-based detection system for Blepharospasm.

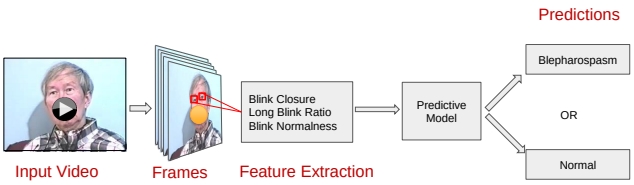

Figure 1. Our objective is to automatically detect Blepharospasm from a subject's video using our novel eye-based feature, Blink Normalness.

Building such a detection system presents several challenges: (i) There is no publicly available dataset of videos featuring Blepharospasm patients. (ii) There are no existing video-level features explicitly designed for Blepharospasm detection. (iii) The system should be robust and natural, without imposing constraints such as specific video lengths or requiring subjects to perform predefined expressions, as seen in previous works [3].

We address these challenges as follows: (i) We construct a novel video dataset consisting of videos of both Blepharospasm patients and healthy individuals. (ii) We develop a novel eye-based feature, termed blink normalness, based on blink dimensions and duration. This required us to create an additional dataset comprising images of open, closed, and tightly closed eyes. (iii) To make our system duration-independent, we incorporate the ratio of counters approach. Additionally, eye blinks are natural, requiring no artificial, predefined actions from the subjects.

To the best of our knowledge, this is the first work on the detection of Blepharospasm in videos. We propose a hybrid feature that effectively combines learning-based and hand-crafted approaches. Finding Blepharospasm patients and recording numerous videos of them is nearly impossible, making it challenging to learn features for this disorder

in an end-to-end manner. To address this, we divide our feature extraction into two stages: blink detection and blink normalness measurement. Notably, end-to-end blink detection is feasible due to the abundance of human images and videos available online. Once blinks are detected, we can hand-craft a feature that incorporates the specifics of the disorder, primarily exploiting blink dimensions and duration. Our experiments show that the proposed hybrid feature performs exceptionally well, demonstrating the effectiveness of this design.

Our contributions are as follows: (1) We develop two novel benchmark datasets: one for Blepharospasm detection and another for blink detection. (2) We introduce a novel hybrid (hand-crafted + deep learning) eye-based feature, termed blink normalness. Our disorder-focused feature design enables robust performance even with very few labeled samples and can indicate the severity degree of the disorder. (3) We are the first to propose an automatic diagnosis system for Blepharospasm based solely on eye analysis.

2. Proposed Method

As discussed earlier, this study focuses on three blink-related symptoms of Blepharospasm: (i) frequent blinking, (ii) long blinks, and (iii) eyelid squeezing. To quantify these symptoms, we extract three corresponding blink-related features: (i) blink closure, (ii) long blink ratio, and (iii) blink abnormalness. In this section, we detail the development of these features and their integration into a comprehensive video-based diagnostic tool for predicting Blepharospasm.

2.1. Blink Closure

Eye blinking is a vital physiological function that plays several critical roles in maintaining ocular health and vision. Blinks help spread the tear film evenly across the surface of the eye, keeping it moist and protecting it from irritants like dust and foreign particles. This reflexive action also provides brief rest periods for the brain's visual processing centers, which are continuously active during wakefulness. On average, a person blinks about 15-20 times per minute, though this rate can vary due to environmental conditions, emotional state, and health status.

In individuals without neurological or ocular conditions, blinks are typically smooth, symmetrical, and consistent in frequency. Each blink involves the coordinated action of the orbicularis oculi muscle, which closes the eyelid, and the levator palpebrae superioris muscle, which opens it. This natural blinking rhythm ensures optimal eye lubrication and protection.

Conversely, individuals with Blepharospasm experience abnormal, involuntary contractions of the muscles around the eyes, leading to excessive blinking and prolonged eyelid closure. This condition disrupts normal blinking patterns, resulting in:

Increased Blink Frequency: Patients may blink more frequently than usual, with episodes of rapid blinking.
Prolonged Eye Closure: Eyelids may remain closed for extended periods, causing functional blindness during these spasms.
Irregular Blink Patterns: Blinking can become erratic, with unpredictable intervals and durations. These disruptions not only affect visual clarity and comfort but also impact daily activities, often leading to social and functional impairments.

To quantify the differences in eye behavior between normal individuals and those with Blepharospasm, we introduce the Eye Closure feature. This feature captures the proportion of time the eye remains open compared to when it is closed within a video sequence. By analyzing frame-level predictions of eye states, we can derive meaningful insights into blinking patterns and eye behavior.

- **Definition:** Blink Closure represents the ratio of the number of frames where the eye was open to the total frames where the eye was either open or closed.
- **Calculation:**

$$\text{Blink Closure} = \frac{\text{ECF}}{\text{ECF} + \text{EOF}} \tag{1}$$

where, ECF denotes the number of frames for which the eye is closed (Eye Closure Frame), and EOF denotes the number of frames for which the eye is open (Eye Open Frame).

The process of calculating the Blink Closure ratio begins with the extraction of frames from a video sequence. Each frame is processed to detect and isolate the eye regions using MediaPipe, as shown in Figure 2. To ensure consistency in orientation, left-eye images are horizontally flipped for subsequent analysis. These processed eye images are then input into a Blink Detector, depicted in Figure 6, which classifies each frame as either an eye-open or eye-closed frame. The total counts of eye-open and eye-closed frames are then calculated separately. Finally, Equation 1 computes the Eye Closure ratio. This ratio remains small for normal blinking and approaches 1 in individuals with Blepharospasm.

2.2. Long Blink Ratio

The Blink Detector provides valuable insights into the frequency and duration of eye closures. In normal blinking patterns, blinks are typically short and frequent. However, in conditions like Blepharospasm, long blinks become more prevalent. By employing the Long Blink Detector, we can quantify long blinks relative to the length of the video,

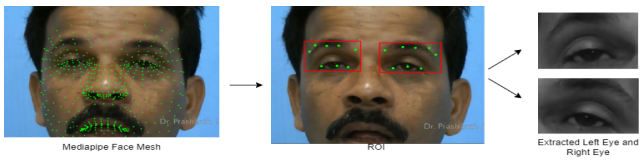

Figure 2. Landmark Detection and Extraction of Eye Using Mediapipe

gaining a deeper understanding of the subject's blinking behavior.

Normal Blinking:
Duration: A typical blink lasts about 100-400 milliseconds (0.1-0.4 seconds).
Frequency: The average person blinks approximately 15-20 times per minute.
Normal blinks are quick and involuntary, providing efficient eye protection and lubrication without significantly interrupting vision.

Long Blinks:
Duration: A long blink is defined as an eye closure lasting longer than the typical blink duration, generally over 0.4 seconds.
Frequency: The frequency of long blinks can increase due to factors such as fatigue, drowsiness, or neurological conditions like Blepharospasm.
Individuals with Blepharospasm exhibit an abnormal increase in the duration and frequency of eye closures, characterized by involuntary, forceful, and often prolonged eyelid closures.

The Long Blink Ratio, quantifies the ratio of long blinks to the total number of blinks, thus providing insights into blink patterns.

- **Definition:** The Long Blink Ratio quantifies the ratio of long blinks to the total number of blinks in a video, offering insights into blink behavior.

- **Calculation:**

$$\text{Long Blink Ratio} = \frac{\text{LB}}{\text{TB}} \tag{2}$$

where, LB (Long Blink) represents the number of long blinks detected, TB (Total Blink) is the total number of blinks detected, and TB is calculated as Long Blink plus Normal Blink.

To determine the Long Blink Ratio, we count the number of long blinks (eye closures lasting longer than 0.4 seconds) and divide it by the total number of blinks in the video. This metric offers a glimpse into the frequency of abnormal blinks relative to the video's length.

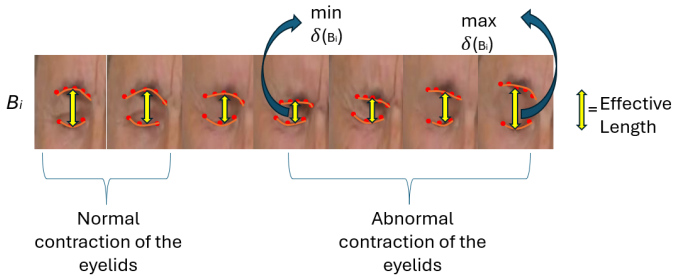

Figure 3. Here is an illustration to differentiate between normal and abnormal contraction of eyelids during a blink. Due to the abnormal condition of Blepharospasm, the distance δB_i between the eyebrows (0-4) and the eyes (5-7) considerably decreases compared to the normal condition. The min-max ratio of these distances during a blink acts as a plausible indicator to represent the normality of blinks.

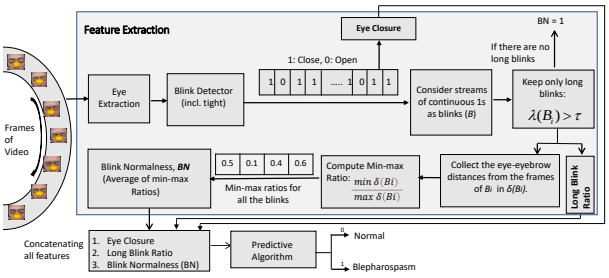

Figure 4. Proposed approach for the extraction of our novel blink normalness feature and the end-to-end workflow for the detection of Blepharospasm.

2.3. Blink Normalness

In Blepharospasm, a patient's blinks often appear as if they are squeezing the eyes due to forceful contractions, as shown in Figure 3. This is evident from the noticeable decrease in the eye-eyebrow distance in the first and fourth frames compared to each other. There is significant variation in this distance during the blink. In contrast, normal blinking does not exhibit such squeezing, and thus, no significant variation is observed. Additionally, blinks in Blepharospasm tend to last longer than normal blinks.

To address these observations, we first detect long blinks and compute the min-max ratio of the eye-eyebrow distance for these extended blinks. Since a video may contain multiple long blinks, we average these ratios to obtain the blink-normalness score. If no long blinks are detected, a score of 1 is assigned, indicating normal behavior. Thus, a high score suggests normal blinking, while a lower score indicates the presence of the disorder.

To calculate this ratio, we first define the Average Normalized Distance. We use five points on each eyebrow and three points along the midline of each eye, as shown in Fig-

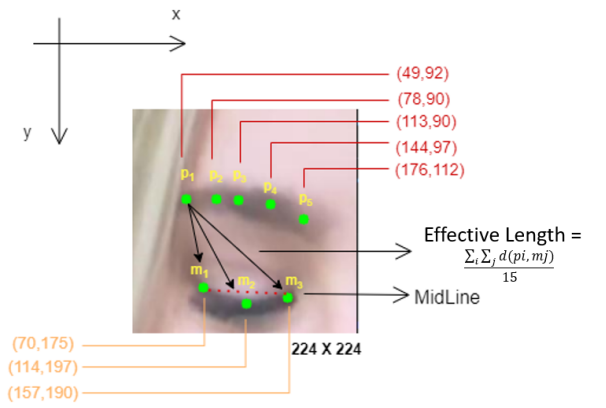

Figure 5. Landmarks on Region of Interest to calculate Blink Normalness

ure 5. These points are represented by their (X, Y) coordinates within a 224 x 224 pixel image.

So we calculate Euclidean distance between each point on the eyebrow p_i to every point on the eye m_j and we normalize it by dividing it by the length of the eye which is the distance from m_1 to m_3. Next, we collect all distances for every point on the eyebrow and calculate the average normalized distance.

- **Calculation:**

$$\text{Normalized Distance}(x_i) = \frac{\sum_j d(p_i, m_j)}{L} \tag{3}$$

where, p_i is the point on the eyebrow, m_j is the point on the midline of the eye, L is the overall length of the eye's midline from m_1 to m_3, and $d(p_i, m_j)$ is the Euclidean distance between p_i and m_j.

$$\text{Avg. Normalized Distance}(\delta(B_i)) = \frac{\sum x_i}{15} \tag{4}$$

where, x_i represents the normalized distance.

Let's assume $B = \{B_1, B_2, ...\}$ denote the set of all the blinks in a video. Let $\delta(B_i)$ be the set of all the eye-eyebrow distances computed from the frames of the blink B_i, and let $\lambda(B_i)$ denotes the duration of the blink B_i. Then, our blink normalness feature is computed in the following manner:

$$BN = \begin{cases} 1, \text{ if } \Big(\sum\limits_{i \in (1,\cdots,|B|)} \lambda(B_i) > \tau \Big) = 0 \\ avg_{i \in (1,\cdots,|B|)} \Big(\{ \frac{\min \delta(B_i)}{\max \delta(B_i)} | \lambda(B_i) > \tau \} \Big), otherwise, \end{cases} \tag{5}$$

where the condition on λ is to check if the blink is long enough. τ denotes the threshold time for a blink to be considered as a long blink. We set τ as 200ms, which is a bit more than average blink duration of 150 ms, to cover some normal blinks that are little more than the average. We give

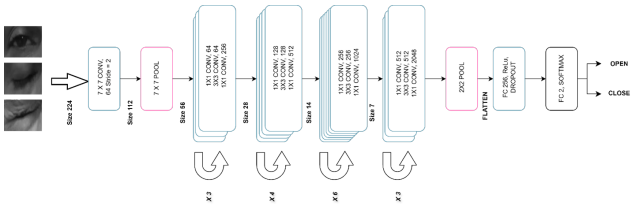

Figure 6. ResNet50-based CNN architecture used for developing the Eye State Detector.

the entire end-to-end work-flow process in Fig. 4. As the blink closure detects blinks, a continuous stream of 1s will appear during the blink. The size of the continuous stream of 1s can be converted into the time domain using the following equation:

$$\lambda(B_i) = |B_i| \times \frac{L}{F}, \tag{6}$$

where L and F are the video's length and the total number of frames in the video. $|B_i|$ is the total number of frames across which the blink B_i occurred. After such conversion, we have the duration of the blink in time-domain to do the comparison.

2.4. Intermediate Model

2.4.1 CNN BASED CLASSIFICATION MODEL (Eye State Detector)

To detect eye blinks, we employ a Convolutional Neural Network (CNN) based on the ResNet50 architecture. This network takes eye images as input and outputs the state of the eye—whether it is closed or open. For each frame, we extract the person's two eyes and horizontally flip the left eye, as the CNN is trained on images of the right eye only. We construct a blink detection dataset (details provided in the next section) comprising images of both open and closed eyes to train the CNN.

As shown in Figure 6, the CNN architecture includes a convolutional base with 49 convolutional layers. The output of this base is flattened and fed into fully connected layers to generate the final output. The last fully connected layer employs the softmax activation function, while an intermediate fully connected layer with 256 neurons uses ReLU activation. An additional Dropout layer with a rate of 0.45 is included to prevent overfitting. Input images for the blink detector are initially 50x50 pixels and are resized to 224x224 pixels for input into the ResNet network. The CNN is trained using categorical cross-entropy loss (CEL), as detailed below:

$$\text{CEL}(y, \hat{y}) = -\sum_{j=1}^{C} \sum_{i=1}^{|I|} (y_{ij} \log (\hat{y}_{ij})) \tag{7}$$

where y_{ij} is the ground-truth label value of the i-th image with respect to the j-th class, and $\hat{y}_{ij}$ is the predicted

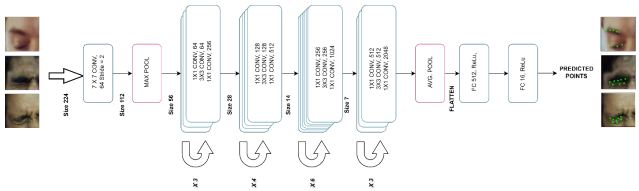

Figure 7. ResNet50-based CNN architecture used for developing the key-point detector.

label value of the i-th image with respect to the j-th class. C is the number of classes, which is two (2) in this case. $|I|$ is the total number of images in a mini-batch. We use the Adamax optimization algorithm to train this CNN.

2.4.2 CNN BASED REGRESSION MODEL (Key-Point Detector)

To predict the 16 coordinates from eye images, we use a Convolutional Neural Network (CNN), specifically a variant of the ResNet architecture adapted for regression tasks. This model takes images as input and outputs the predicted coordinates, which represent the positions of key points around the eyes on a 224x224 pixel image. The 16 predicted coordinates include both x and y positions of these key points. These coordinates are crucial for tasks such as detecting the state of the eye (open or closed) and analyzing eye movements and behaviors. Accurate detection of these key points is essential for identifying conditions such as Blepharospasm.

As shown in Figure 7, the architecture's convolutional base consists of 49 convolutional layers. The output from these convolutional layers is flattened and passed through a fully connected (FC) layer with 512 neurons, utilizing the ReLU activation function. The final output layer is a fully connected layer with 16 neurons, corresponding to the 16 coordinates we aim to predict.

The trained CNN model uses the mean squared error (MSE), as mentioned below:

$$\text{MSE}(y, \hat{y}) = \sum_{i=1}^{|I|} (y_i - \hat{y}_i)^2 \tag{8}$$

where y_{ij} is the ground-truth label value of the i-th sample, and $\hat{y}_i$ is the predicted label value of the i-th sample. By minimizing the MSE loss, the model learns to make predictions that are as close as possible to the actual key point coordinates, ensuring high accuracy in the regression task. This approach allows the model to efficiently capture the spatial hierarchies and intricate patterns in the input images.

3. Experiments

In this section, we will first discuss the datasets we have built, then we will discuss our blink closure and Blepharospasm detection results based on some experiment.

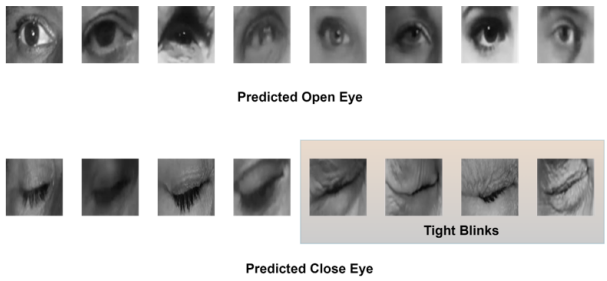

Figure 8. Sample results of our blink detector.

3.1. Datasets

We build three datasets: one for blink closure, another for blink normalness and third for Blepharospasm detection.

Blink (incl. tight) Closure Dataset: It consists of 17760 open-eye images and 30700 closed-eye images, some of which are tightly closed. The purpose is to detect if a given eye is closed, including tightly closed.

Note that this dataset consists solely of right eye images, which means we flip the left eye images to maintain consistency. Most of these images were collected from YouTube videos by manually cropping the eyes frame-by-frame.

Landmark Annotated Dataset: This dataset comprises 5,886 manually annotated landmark points on regions of interest (ROI). Each image is 224 by 224 pixels, with (x, y) coordinates marked for 8 points—5 on the eyebrows and 3 on the eyes.

Blepharospasm Video Dataset: This dataset includes 70 videos of patients suffering from Blepharospasm and 74 videos of normal individuals.

We report cross-validation and classification accuracy throughout this work, defined as the percentage of correct predictions. Due to the disparity in the availability of labeled data for the intermediate and final detectors, different evaluation approaches are employed.

For blink closure, with abundant labeled data, we divide the data into three parts: training, validation, and testing. Specifically, we use the hold-out approach for validation, where a portion of the dataset is reserved for validation. We optimize hyperparameters for training the CNNs based on validation scores and then test the trained CNN on the testing dataset.

For the key points detector model, we split the dataset into two parts: 80% for training and 20% for testing. During the model training process, we compute the mean squared loss on the test data. The **resulting test loss is 7.8961**.

For the Blepharospasm detector, we reserved 25% of the data for testing and employed k-fold cross-validation on the remaining 75% of the labeled dataset. In k-fold cross-validation, the dataset is divided into k parts, training on

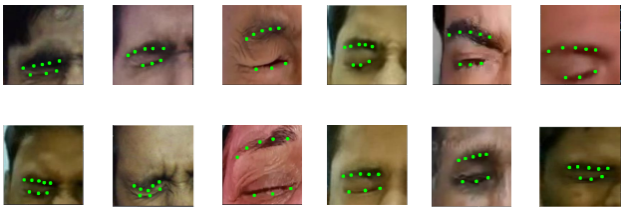

Figure 9. Sample results of our CNN-based keypoint detector model.

$k-1$ parts and validating on the remaining part. We used $k=5$ for the Blepharospasm dataset. This cross-validation helps identify the best algorithm among seven basic machine learning algorithms: Random Forests (RF), Support Vector Machines (SVM), Logistic Regression (LR), k-Nearest Neighbors (kNN), Decision Tree (DT), Multi-Layer Perceptron (MLP), and eXtreme Gradient Boosting (XGBoost). After selecting the best algorithm, we trained on the 75% labeled dataset to develop our final detectors. In addition to classification accuracy from testing on the reserved test dataset, we also report confusion matrices to evaluate detector performance.

We then compare our Blepharospasm detector with various competitive models to assess its robustness against different methods.

3.2. Blink Closure (Open/Close Detection Result)

We present quantitative results on the accuracy of our blink detector, which is designed to detect both tight and normal blinks, as shown in Table 1. The performance is compared with another detector trained exclusively on normal blinks. For this comparison, we created two datasets derived from the original dataset, which includes 17,760 open-eye images and 30,700 closed-eye images.

The **first dataset** contains 16,123 closed-eye images with both normal and tight blinks, along with 15,000 open-eye images. The **second dataset** includes 16,024 closed-eye images with only normal blinks and 15,000 open-eye images. We evaluated the detectors on our test data which includes 2,010 normal closed-eye images, 2,760 tightly closed-eye images (tight blinks), and 2,673 tightly open-eye images.

The Blink detector achieves a classification accuracy of over 99% across all three cases, while the blink detector (w/o tight blinks) struggles to accurately detect tight blinks, as shown in Table 1. Sample detection results from our blink detector are presented in Figure 8. This indicates that the blink detector model, which includes tight blinks, effectively distinguishes between open and closed eyes, including tightly closed ones.

Therefore, including tight blinks improves our model's accuracy in detecting whether an eye is open or closed, particularly in cases of Blepharospasm where patients exhibit signs of tight blinks. To further enhance our model, we have developed a final blink detector model (Eye State Detector) using a dataset consisting of 26,167 closed-eye images (16,123 normal closed-eye images and 10,044 tight closed-eye images) along with 15,000 open-eye images. This comprehensive dataset allows our final model to leverage the strengths of both individual datasets mentioned above. The confusion matrix obtained by testing on the same test dataset using this blink detector is provided in Table 2. We evaluated our model on the same test dataset achieving an **accuracy of 99.81%**.

Detector	Tight Close Eye	Normal Close Eye	Open Eye
Blink Detector	100.00%	99.65%	99.49%
Blink Detector (w/o Tight blinks)	90.66%	99.65%	99.39%

Table 1. Accuracy of different detectors on various eye states on Test Dataset.

Eye State Detector		Predicted Open	Predicted Close
	Actually Open	2749	11
	Actually Close	3	4674

Table 2. Confusion matrices in the testing phase while building our intermediate detectors for detecting blink.

3.3. Keypoint Detection

In this section, we compare the results of keypoint detection using deep learning-based regressor model,Key-Point Detector (Figure 7) with those obtained through Mediapipe keypoints.

Mediapipe is an open-source framework developed by Google that provides a range of machine learning solutions for live and offline media processing. Among its popular applications is face mesh detection, which offers a comprehensive set of facial keypoints (or landmarks) for tasks such as facial recognition, expression analysis, and augmented reality.

Mediapipe's face mesh model can detect 468 3D facial landmarks in real time, even on mobile devices. These landmarks include key regions of the face such as the eyes, eyebrows, nose, lips, and jawline, making it a valuable tool for various facial analysis tasks. By comparing our results from Eye State Detector with those from Mediapipe, we aim to evaluate the performance and accuracy of our method. While Mediapipe provides a well-established and detailed keypoint detection system, our method aims to offer a more targeted approach, potentially providing advantages in specific applications such as improved accuracy or computational efficiency.

We build the feature Blink Normalness (BN) using key points detected by both methods and calculate classification accuracy. We use the same number of key points for both methods. BN was calculated using Equation 5. Comparing results from both methods based on cross-validation score, as shown in Table 3, we find that key points extracted using our CNN model consistently outperform those obtained through Mediapipe. Our CNN model predicts key points more accurately, leading to a better classifier. Also, we present classification accuracy obtained on test data in Table 4 using the algorithm underlined in Table 3. The accuracy improvement is significant, demonstrating that our CNN model provides a more robust and accurate representation for classification tasks, particularly in blepharospasm analysis.

Keypoint Models	RF	SVM	LR	kNN	DT	MLP	XGBoost
Mediapipe Keypoints	0.5797	0.6786	0.664	0.6465	0.5507	0.6916	0.5749
Key-Point Detector	0.8611	0.8465	0.8410	0.8592	0.8341	0.8437	0.8545

Table 3. Comparison of CV Score on the basis of BN calculated using Mediapipe keypoints and keypoints obtained using our CNN model (Red shows the best result and Blue shows second best result across all classifier).

3.4. Video Detection

In this section, we compare the results from different methods to classify videos accurately.

3.4.1 Classification Methods

We employed five distinct methods to classify videos using seven machine learning algorithms. **(1) Mediapipe Coordinates** involves using Mediapipe to predict the coordinates for the eyes and eyebrows in every frame of each video. **(2) ResNet50 Embeddings** utilizes a modified version of the ResNet50 model to obtain embeddings for the video frames. **(3) 3D Convolutional Neural Network (3D CNN)** employs a 3D CNN to capture spatiotemporal patterns within the video data. **(4) Comprehensive Feature Set Method (Blepharospasm Detector)** utilizes all features extracted during the feature extraction process. Lastly, **(5) Threshold-Based Unsupervised Classification** introduces a novel threshold-based unsupervised learning approach to detect blepharospasm from video data.

		Predicted No	Predicted Yes
Mediapipe Keypoints(52.77)	Actual No	9	7
	Actual Yes	10	10
		Predicted No	Predicted Yes
Key-Point Detector (72.22)	Actual No	13	3
	Actual Yes	7	13

Table 4. Confusion Matrix obtained on Blepharospasm Detection using Mediapipe keypoints and Key-Point Detector on Test Dataset

3.4.2 Mediapipe Coordinates

In the first method, Mediapipe is used to predict keypoints on every frame of each video. We use 5 points for the eyebrow and 3 points for the eye, with each point represented by x and y coordinates. For each point, we calculate the average values of the x and y coordinates and their corresponding standard deviations (SD) across all frames in a video. This results in a 64-dimensional vector for each video, where each point is represented by $[X_{\text{avg}}, X_{\text{sd}}, Y_{\text{avg}}, Y_{\text{sd}}]$ (32 dimensions for each eye). The extracted features, which include the average and SD of keypoint positions, are then used as input to a classification algorithm. Table 5 shows the cross-validation (CV) score obtained using these inputs for classification.

3.4.3 ResNet50 Embeddings

In the second method, we use a modified version of the ResNet50 model to obtain embeddings for video frames. Typically used for image classification, the ResNet50 model is adapted by removing the final fully connected (fc) layer, allowing it to output embeddings from the last average pooling layer. For each video, 60 frames are randomly selected (30 from the left eye and 30 from the right eye) and passed through the modified ResNet50 model. The output from this layer is a 2048-dimensional vector, representing the learned features or "embeddings" for each frame. After obtaining embeddings for all 60 frames, we calculate the average and standard deviation (SD) of each of the 2048-dimensional embeddings across all frames, resulting in a 4096-dimensional vector for each video which is used as a feature vector for detecting blepharospasm.

Models	**RF**	**SVM**	**LR**	**kNN**	**DT**	**MLP**	**XGBoost**
Mediapipe	0.7942	0.7175	0.7277	0.4053	0.6512	0.6121	0.7219
ResNet50	0.8503	0.8512	0.8594	0.7941	0.7552	0.8577	0.8235
3D CNN	0.6854	0.6845	0.6539	0.4089	0.6071	0.6606	0.6518
Blepharospasm Detector	0.9684	0.9528	0.9637	0.9628	0.9138	0.9583	0.9573

Table 5. Performance of various classifiers using different competitive models during cross-validation (Red shows the best result and Blue shows the second best result across all classifiers).

3.4.4 3D Convolutional Neural Network (3D CNN)

In the third method, we employed a 3D Convolutional Neural Network (3D CNN) to extract embeddings from videos of blepharospasm patients and normal subjects. Each video is divided into three segments, each consisting of 16 consecutive frames. These frames are resized to 64x64 pixels and normalized to ensure uniform input dimensions for

the model. The network begins with an input video sequence and consists of three 3D convolutional layers, each followed by max pooling. These layers progressively reduce spatial dimensions while extracting features. The convolutional layers use ReLU activation and maintain input size with same padding. The final embeddings are generated by a dense layer with 128 neurons, resulting in a 128-dimensional embedding vector for each segment. The embeddings from all three segments are concatenated into a single 384-dimensional vector for each video.

These embeddings capture the temporal and spatial features of the video segments, encapsulating patterns indicative of blepharospasm. By transforming the raw video frames into fixed-size embeddings, the model effectively condenses complex video information into a compact representation. This transformation highlights specific features and patterns in eye movements and blinks that are characteristic of the condition, enabling accurate and automated diagnosis based on the video data.

The loss function used in the model is binary cross-entropy, as defined by Equation 9.

$$\text{Loss} = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \log(p_i) + (1-y_i)\log(1-p_i)\right] \quad (9)$$

Where N is the number of samples, y_i is the actual label of the i-th sample (either 0 or 1), and p_i is the predicted probability that the i-th sample belongs to the class labeled as 1.

Mediapipe (80.55)		Predicted No	Predicted Yes
	Actual No	15	1
	Actual Yes	6	14
ResNet50 (88.88)		Predicted No	Predicted Yes
	Actual No	15	1
	Actual Yes	3	17
3D CNN (69.44)		Predicted No	Predicted Yes
	Actual No	11	5
	Actual Yes	6	14
Unsupervised Classification (86.11)		Predicted No	Predicted Yes
	Actual No	13	3
	Actual Yes	2	18
Blepharospasm Detector (94.44)		Predicted No	Predicted Yes
	Actual No	14	2
	Actual Yes	0	20

Table 6. Confusion Matrix of different competitive models on Test Data (Red shows the best result and Blue shows the second best result).

3.4.5 Threshold-Based Unsupervised Classification

In this study, we introduce a threshold-based unsupervised learning approach for detecting blepharospasm from video data. The dataset includes features computed from eye metrics: Left Eye Avg and Right Eye Avg, derived by averaging three individual features: Blink Closure, Long Blink Ratio, and (1 - Blink Normalness). In these features, values close to 1 indicate the presence of blepharospasm, while values closer to 0 suggest normal conditions.

The method involves splitting the dataset into training and testing subsets. We then apply a series of thresholds to cluster the data based on the average feature values. Clustering is performed by assigning data points to clusters based on whether their average feature values exceed the threshold. The optimal threshold is determined by maximizing clustering accuracy on the training set. This threshold is then applied to the test set to assess performance through accuracy and confusion matrix metrics, providing insights into the effectiveness of the clustering in distinguishing between blepharospasm and normal conditions. The optimal **threshold** for clustering was determined to be **0.22**. Using this threshold, the method achieved a test accuracy of approximately 86.11%.

3.4.6 Comprehensive Feature Set (Blepharospasm Detector)

In the fourth method, we use all features extracted during the feature extraction process to classify the videos. Table 5 shows the performance of various classifiers obtained using k-fold cross-validation with k = 5 followed by GridSearchCV.

3.4.7 Results:

For classification, the dataset was split into training and testing sets, with 25% of the data reserved for testing. The model's performance was evaluated using accuracy, confusion matrix metrics in Table 6 using the classifier underlined in Table 5.

Mediapipe Features: The performance metrics for classifiers using Mediapipe features are relatively lower compared to those using ResNet50 and the Blepharospasm Detector model. The best-performing classifier with Mediapipe features is Random Forest (RF), achieving an accuracy of 79.42% on cross-validation and 80.55% on test data. This suggests that while Mediapipe features provide useful information, their effectiveness is limited compared to other feature extraction methods.

ResNet50 Features: Classifiers using embeddings extracted from ResNet50 demonstrate a significant improvement over those using Mediapipe features. The best-performing classifier with ResNet50 features is Logistic Regression (LR), with an accuracy of 85.03% on cross-validation and 88.88% on test data. This indicates that ResNet50 features are more robust and informative for classification tasks, showing a clear enhancement in performance compared to Mediapipe features.

3D CNN: The best-performing classifier using the 3D Convolutional Neural Network (3D CNN) achieves an ac-

curacy of 68.54% on cross-validation and 69.44% on test data. This performance is notably lower compared to the comprehensive feature set used for blepharospasm detection.

Unsupervised Classification: This method utilizes average feature value (Left Eye Avg and Right Eye Avg), and achieves an accuracy of 86.11% on the test data. The improvement over Mediapipe, ResNet50 and 3D CNN suggests that the selected features are effective in capturing the characteristics of blepharospasm, but it may not fully exploit all the discriminative information available in the data compared to the comprehensive feature set used for blepharospasm detection.

Blepharospasm Detector: The method utilizes a comprehensive set of features extracted during the feature extraction process discussed above, demonstrating the highest performance across all classifiers. For instance, the Random Forest (RF) classifier achieves an accuracy of 96.84% on cross-validation and 94.44% on test data, significantly surpassing the accuracies of Mediapipe features, ResNet50 features, 3D CNN method and Unsupervised classification method. This underscores the effectiveness of the feature extraction approach, leading to superior classification outcomes.

Results from Table 5 and Table 6 clearly indicate that the features extracted and utilized in the Blepharospasm Detector Model method offer superior classification performance compared to those from Mediapipe, ResNet50, 3D CNN and Unsupervised classification. The substantial accuracy increase across all classifiers when using the Model method highlights the importance of effective feature extraction in enhancing model performance, with Random Forest (RF) achieving the highest accuracy.

3.5. Blepharospasm Detection Results

For our final Blepharospasm detector, we built a classification model using **Random Forest**, which achieved the highest accuracy. The model was trained using 75% of the video dataset and tested on the remaining 25%. The confusion matrix is provided in Table 6.

From the table, it is evident that the proposed system predicts Blepharospasm with an accuracy of $(\frac{14+20}{17+2+20+0}) \times 100 = 94.4\%$. Notably, there are no false negatives, which is advantageous from a clinical perspective, indicating that all individuals with Blepharospasm are detected by the system.

3.5.1 Comparison of Classification Methods using Table 6.

Mediapipe: The Mediapipe method shows decent performance with an accuracy of 80.55%, but it struggles with a higher number of misclassifications compared to the proposed model. **ResNet50:** The ResNet50 method performs slightly better than Mediapipe with an accuracy of 88.88%, but it still falls short of the accuracy achieved by the proposed method. **3D CNN:** The 3D CNN method has the lowest accuracy among the compared methods, at 69.44%. This result highlights the limitations of using 3D CNN for this specific task compared to the proposed model. **Threshold-based unsupervised** classification method provides a reasonable improvement with an accuracy of 86.11%, over traditional methods like Mediapipe, ResNet50, and 3D CNN, it does not achieve the level of accuracy reached by the proposed model. **Proposed Model (Blepharospasm Detector):** The proposed model significantly outperforms the other methods with an accuracy of 94.44%, making it the most effective approach. The model achieves perfect classification for the "Yes" category, demonstrating its robustness and reliability in identifying Blepharospasm cases.

All features (94.44)		Predicted No	Predicted Yes
	Actual No	14	2
	Actual Yes	0	20
Blink Closure Only (77.77)		Predicted No	Predicted Yes
	Actual No	12	4
	Actual Yes	4	16
Long Blink Only (80.55)		Predicted No	Predicted Yes
	Actual No	11	5
	Actual Yes	2	18
BN Only (72.22)		Predicted No	Predicted Yes
	Actual No	13	3
	Actual Yes	7	13
Blink Closure + Long Blink Ratio (86.11)		Predicted No	Predicted Yes
	Actual No	14	2
	Actual Yes	3	17
Blink Closure + BN (77.77)		Predicted No	Predicted Yes
	Actual No	13	3
	Actual Yes	5	15
Long Blink Ratio + BN (69.44)		Predicted No	Predicted Yes
	Actual No	13	3
	Actual Yes	8	12

Table 7. Ablation Study

Features	RF	SVM	LR	kNN	DT	MLP	XGBoost
All features	0.9684	0.9528	0.9637	0.9628	0.9138	0.9583	0.9573
Blink Closure Only	0.8582	0.9025	0.8935	0.8666	0.8518	0.8924	0.8676
Long Blink Only	0.790	0.8215	0.8198	0.8205	0.7939	0.8206	0.8389
BN Only	0.8628	0.8519	0.8454	0.8592	0.8563	0.8491	0.8581
Blink Closure + Long Blink Ratio	0.8852	0.9159	0.9083	0.9029	0.8721	0.9166	0.8851
Blinl Closure + BN	0.9046	0.9140	0.9008	0.9142	0.8515	0.9103	0.8988
Long Blink Ratio + BN	0.8797	0.8725	0.8705	0.8741	0.8424	0.8714	0.8805

Table 8. CV scores of different combinations of features for detecting Blepharospasm (Red shows the best result and Blue shows the second best result across all classifiers).

Conclusion

We attempted video-based detection of Blepharospasm by extracting a novel eye-based feature called Blink Normalness from subjects' videos, along with other key features such as Blink Closure and Long Blink Ratio, to develop our predictive model. We designed and implemented two CNN models: one for detecting blinks, including tight blinks, and another for keypoint detection on the eyebrow and eye. Our extensive experiments on the developed datasets demonstrate excellent performance in detecting Blepharospasm.

References

[1] Talmage J Broadbent, Ralph E Wesley, and Louise A Mawn. A survey of current blepharospasm treatment patterns among oculoplastic surgeons. *Ophthalmic Plastic & Reconstructive Surgery*, 32(1):24–27, 2016. 1

[2] D Narendhar Singh, Mohammad Farukh Hashmi, and Sudhir Kr Sharma. Predictive analytics & modeling for modern health care system for cerebral palsy patients. *Multimedia Tools and Applications*, pages 1–23, 2019. 1

[3] Anping Song, Zuoyu Wu, Xuehai Ding, Qian Hu, and Xinyi Di. Neurologist standard classification of facial nerve paralysis with deep neural networks. *Future Internet*, 10(11):111, 2018. 1

[4] Gozde Yolcu, Ismail Oztel, Serap Kazan, Cemil Oz, Kannappan Palaniappan, Teresa E Lever, and Filiz Bunyak. Facial expression recognition for monitoring neurological disorders based on convolutional neural network. *Multimedia Tools and Applications*, 78(22):31581–31603, 2019. 1