

Received September 2, 2021, accepted September 13, 2021, date of publication September 28, 2021, date of current version October 11, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3115961

Classification of Eye Movement and Its Application in Driving Based on a Refined Pre-Processing and Machine Learning Algorithm

XIAN-SHENG LI¹, ZHI-ZHEN FAN¹, YUAN-YUAN REN¹,
XUE-LIAN ZHENG¹, AND RAN YANG²

¹Transportation College, Jilin University, Changchun 130022, China

²Kingfar International Inc., Beijing 100089, China

Corresponding author: Yuan-Yuan Ren (renyy@jlu.edu.cn)

This work was supported by the National Key Research and Development Program of China under Grant 2018YFB1600502.

ABSTRACT The eyes are the first channel used by humans to obtain various types of visual information from the outside world and, especially when driving, 80-90% of information is received through the eyes. Eye movement behaviors are generally divided into six types, but attention is often paid to fixation, saccade, and smooth pursuit. Due to their importance, it is essential to classify eye movement behaviors accurately. The classification of eye movements should be a complete process, including the three steps of pre-processing, classification, and post-processing. However, it is very uncommon for all of these steps to be included in the eye-tracking literature when eye movement classification is discussed. Therefore, first, this paper proposes a refined eye movement data pre-processing framework and an improved method consisting of three steps is introduced. Second, an eye movement classification algorithm based on an improved decision tree that is independent of the threshold setting and application environment is proposed, and a post-processing consisting of merging adjacent fixations and discarding short fixations is described. Finally, the application of the classified eye movement behavior in the driving field is described, including the estimation of preview time using fixation and the estimation of time-to-collision using smooth pursuit. Two important results are obtained in this paper. One concerns the classification accuracy of eye movement behavior, the F1-scores of fixation, saccade, and smooth pursuit being respectively 92.63%, 93.46%, and 65.29%, which are higher than the scores of other algorithms. The other relates to the application to driving. On the one hand, the preview time calculated by fixation is mostly distributed around 1-6s, which is closer to reality than the traditional setting of 1s. At the same time, the regression relationship between the preview time and the road turning radius is also quantitatively analyzed and their regression function is obtained. On the other hand, the average estimated error of time-to-collision used by smooth pursuit is 7.37%. These results can play an important role in the development of ADAS and the improvement of traffic safety.

INDEX TERMS Keywords eye movement data, refined pre-processing, behavior classification, feature construction, improved decision tree.

I. INTRODUCTION

The eyes are the first channel used by humans to obtain various types of visual information from the outside world, and represent an important way of searching for and receiving information. For example, statistical studies show that

80-90% of driver information is received through vision in driving [1]. Therefore, the extraction and analysis of human visual information has theoretical significance and application value in many fields. It is therefore relevant to study eye movement characteristics and their classification.

The irises can be used to obtain human gaze information as well for biometric identification as they are stable, unique, and non-contact identification objects. At present,

The associate editor coordinating the review of this manuscript and approving it for publication was Ikramullah Lali.

iris detection technology is an important research channel for obtaining eye movement information. Moreover, the classification of eye movement data after iris detection is also an important research branch, which can be used to quantify visual attention and to mine the area of interest of the individual, and even to evaluate their mental load. Nowadays, it is widely used in the fields of market research, learning and education, driving safety, etc. In this paper, we focus on the classification of eye movement and its application in the driving field.

Studies have shown that the fovea, the most sensitive and clear-sighted part of the eye, occupies only a small area of the retina. In order to see and locate an object of interest, humans need to adjust the position of the eyeballs so that light can be focused in the fovea as much as possible. According to the characteristics of eye movements and the role they play in visual imaging, human eye movements can be divided into six types: fixation, saccade, smooth pursuit, optokinetic reflex, vestibulo-ocular reflex, and vergence [2]. However, in the application of eye movements to scientific research, attention is often paid to only three of these types: fixation, saccade, and smooth pursuit. Therefore, determining how to classify these three eye movement types correctly and accurately from raw eye movement data is very important.

Early eye movement classification algorithms focused on distinguishing between fixation and saccade, because the stimulus materials used are mostly static, for example, images or text. More recently, the application of dynamic stimulation has attracted growing attention. Unlike static stimuli, which are relatively still (fixation) or move rapidly from one fixation point to another (saccade), for dynamic stimuli the subject's eyes track the stimulus and keep it in the fovea, creating a smooth eye movement known as smooth pursuit. The presence of smooth pursuit affects the performance of classification algorithms in distinguishing between fixation and saccade, as smooth pursuit points are eventually classified into one of these two categories. Therefore, in the study of eye movement involving dynamic stimuli, developing an automatic and effective algorithm to distinguish between fixation, saccade, and smooth pursuit is of great importance.

Traditional eye movement classification models are divided into three main categories: threshold-based, probabilistic-based, and hybrid algorithms [3].

A threshold-based algorithm takes eye movement behavior characteristic parameters (such as the speed or spatial dispersion of eye movement data) as the threshold, and then uses them to generate classification results. It is a comparative classification algorithm and is still widely used. Classic threshold-based algorithms include I-VT (velocity threshold identification) and I-DT (dispersion threshold identification) [3]. The I-VT algorithm separates the fixation point from the saccade point by setting a speed threshold [4]. The advantage of I-VT is its fast-processing speed, but the disadvantage is that it can easily produce classification errors because it strongly depends on the threshold setting. The principle of the I-DT algorithm is to calculate the dispersion of

eye movement points within a sliding time window, compare the calculated dispersion in the window at each time point with a preset threshold for the dispersion, and finally, realize the separation of fixation and saccade points.

The I-VVT (Identification with Velocity and Velocity Threshold) and I-VDT (Identification with Velocity and Dispersion Threshold) algorithms have since been developed based on the above two algorithms. I-VDT distinguishes eye movement behavior by using speed and dispersion thresholds [5]. The speed threshold is firstly used to distinguish saccades, and then the dispersion threshold is used to classify fixation and smooth pursuit points. The purpose of the I-VVT algorithm is to classify eye movement behavior using two velocity thresholds [6]. It initially uses the first speed threshold to separate the saccades from the others, and then uses the second speed threshold to identify the fixation and smooth pursuit points. Although I-VVT can classify the three kinds of eye movement behaviors, its disadvantage is that it can easily produce errors in classification. Moreover, its classification performance on eye movement data with high sampling frequency is also poor.

Probability-based algorithms are used to build a characteristic probability distribution model for each kind of eye movement state (for example, the distribution of speed), estimate the posterior probability using the prior probability, recalculate the probability distribution parameters to minimize them, and finally obtain the classification results. Algorithms of this type include the Markov model algorithm (I-HMM), the Bayesian theory algorithm (I-BDT), and the Kalman filtering algorithm. The advantages of the I-HMM are its performance at state prediction and its probability characteristics [3]. In the I-BDT, a Gaussian mixture model is used to fit the velocity distribution based on the eyeball velocity [7]. This is used to distinguish between fixation and saccade behaviors because the basic parameters of the model are determined by eye movement velocity calculated from two sample points, so the model parameters are constantly updated as the number of samples increases. This means the classification effect is continually improved.

Hybrid algorithms are fusions of threshold-based and probability-based algorithms, and include I-DFCM (Distance threshold and FCM Identification), AFKF (Attention Focus Kalman Filter), I-VMPray (Velocity and Movement Pattern Rayleigh), and I-BMM (Bayesian Mixture Model). The I-DFCM first identifies the saccade points using a distance threshold and then distinguishes the fixation and smooth pursuit points using the space characteristics and fuzzy C-means [8]. In the AFKF algorithm, one firstly distinguishes saccades from other eye movements using a Kalman filter and chi-squared test, and then classifies fixation and smooth pursuit points using velocity and time thresholds.

In recent years, with the rapid development of computer technology, machine learning algorithms have become popular, and another kind of novel eye movement classification algorithm based on machine learning has been developed. These methods can achieve real-time classification by

extracting features from eye movement data and training the classification model. Taking some classic algorithms, Melodie Vidal proposed a novel set of shape features including slope, range, mean velocity, variance, integral, energy, and waveform length of data, and used the machine learning method to obtain a classification model [9]. Then, the classification accuracy of models with different time windows was calculated. Finally, the model with the highest level of accuracy was selected to classify eye movement behavior. Mikhail Startsev used a combination of the ID-CNN and BLSTM to carry out a mapping of data to obtain eye movement behavior results [10]. The method had good real-time performance, but the classification accuracy of the saccades and the smoothing tracking was low. Sabrina Hoppe used a convolutional neural network to achieve end-to-end learning [11]. The model proposed has strong learning ability and fast classification speed, but does not carry out pre-processing of raw eye movement data.

Thus far, most eye movement classification algorithms have been based on manually set thresholds and rules [12], and the classification results are subjective and may not match the actual situation very well. Although machine learning methods can be used to avoid some of the above problems and achieve a high level of classification accuracy, they are always limited by the application environment. Therefore, the development of an accurate eye movement classification algorithm that is not affected by the environment is urgently required. In addition, the quality of the eye movement data collected is inevitably affected by factors such as the irregular sampling frequency of eye-trackers, the instability of data transmission, errors associated with iris detection algorithms, and eye tremor, jitter, or even abnormal eye movements or head turning influenced by the experimental environment. In other words, determining how to pre-process raw eye movement data, improve its quality, and then better apply it to the classification and application of eye movement behavior is also crucial.

Based on the above problems, we propose a refined eye movement data pre-processing framework and an eye movement classification algorithm that is independent of the threshold setting and application environment. The application of the classified eye movement behavior to the driving field is also described. The main contributions of this paper are as follows:

(1) We propose a refined pre-processing framework and an improved method involving three key steps for optimizing the quality of raw eye movement data. The processed data are more similar to reality and will be beneficial for the construction of different kinds of eye movement behavior. The key steps are sampling frequency correction, small gap fill-in, and fusion filtering.

(2) Five new features of eye movement behavior classification are constructed and proposed after we have fully mined the movement and distribution characteristics of the fixation, saccade, and smooth pursuit points.

(3) In our machine learning method, we use the decision tree algorithm for eye movement classification. In order to improve the generalizability of the decision tree model, a post-pruning method is proposed to optimize the model, considering the depth of the tree and the number of classified samples.

(4) The driver's eye movement data are collected, pre-processed, and classified. Finally, the fixation and smoothing pursuit points are used to estimate the preview time and collision time when driving.

II. TRACKING PRINCIPLES AND QUALITY ANALYSIS OF RAW EYE MOVEMENT DATA

A. EYE MOVEMENT DATA TRACKING AND MEASUREMENT

Eye movement tracking and measurement has been gaining popularity for decades as a way to understand and study an observer's visual characteristics as well as their cognitive ability [13], [14]. Common methods of eye movement tracking and measurement include the electro-oculography method and the camera method [15]. The electrical potential around the eyes changes as they move, which can be used to track and measure eye movements [16]. Compared with the method above, the camera method is more stable. It uses a camera to record a subject's eye movement, extract eye feature information, and estimate the eye direction or determine the eye movement points. Based on the location of the eye-tracker at the time of eye movement tracking and measurement, they can be divided into two categories: head-mounted and remote. Until now, non-contact eye-trackers combined with video cameras have been the method most commonly used to record eye movement [17].

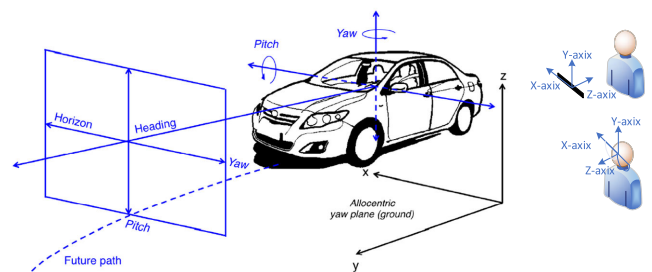


FIGURE 1. Eye-tracker-fixed coordinates used to describe driver eye movement.

B. TYPES OF PARAMETERS COLLECTED BY EYE TRACKERS

In the transportation field, eye trackers are widely used to collect drivers' eye movement data. The collected data can be used to analyze drivers' eye movement characteristics. Generally, the remote eye tracker is attached to the top of the vehicle dashboard, just in front of the driver. The origin of the coordinates is located in the center of the eye tracker, as shown in Figure 1. The x -axis is in the direction of the driver's left side, the z -axis is parallel to vehicle's longitudinal axis and in the direction of the driver, and the y -axis is

TABLE 1. Parameters for tracking and measuring using eye trackers.

Parameter		Label	Parameter specification
Remote	Head-mounted		
Timestamp	Timestamp	Time	Timestamp from the beginning of tracking
Eye Position	—	Left Eye Position (3D)	The 3D position of the eyes in the eye-tracker—fixed coordinate
		Right Eye Position (3D)	
Eye Relative Position	—	Left Eye Relative Position (2D)	The 2D position of the eyes in the eye-tracker—fixed coordinate
		Right Eye Relative Position (2D)	
Gaze Position	Gaze Position	Left Gaze Position (3D)	The 3D position of the gaze point in the eye-tracker—fixed coordinate
		Right Gaze Position (3D)	
Gaze Relative Position	Gaze Relative Position	Left Gaze Relative Position (2D)	The 2D position of the gaze point in the eye-tracker—fixed coordinate
		Right Gaze Relative Position (2D)	
Pupil Diameter	Pupil Diameter	Left Pupil Diameter	Pupil diameter in millimeters
		Right Pupil Diameter	
Record Validity	Record Validity	Left Eye Validity	The accuracy degree of eyes tracked by the device
		Right Eye Validity	

perpendicular to the XZ plane and in the upward direction. The head-mounted tracker is worn on the driver’s head. Here, the origin of the coordinates is on the driver’s head, which can also be seen in Figure 1. Regardless of the kind of eye-tracker used, the parameters used to describe the eye movement are quite similar. The only difference is in the parameters used to describe the eye position. The essential parameters are listed and described in Table 1.

C. QUALITY ANALYSIS OF RAW EYE MOVEMENT DATA

There are three possible reasons for noise or a lack of eye movement data in the tracking process:

1. Noise caused by the eye tracker, such as image distortion, data transmission, different collecting frequencies, etc.
2. Errors caused by an algorithm embedded in the eye-tracker, such as a location error for the center of the iris caused by the camera algorithm.
3. Tremor or jitter caused by the eyeball itself.

Moreover, the tracking quality might be influenced by the experimental environment or by movement of the subject’s head. Therefore, it is essential to conduct a quality analysis of the raw eye movement data in two ways: a validity check and correctness checking of the tracking frequency.

1) PRELIMINARY VALIDITY CHECK

The record validity is an important parameter, which can be used to describe the accuracy of eye movement data tracking. If the eye tracker cannot track the subject’s eye movement well, effective eye movement data cannot be collected, which means that the data collected will be invalid. Invalid data are useless for subsequent feature construction and eye movement classification. Therefore, a preliminary validity check is an essential procedure for an eye tracker. Taking the Tobii eye tracker as an example, a larger value indicates a lower tracking accuracy, and a smaller value, a higher tracking accuracy.

2) CORRECTNESS CHECK OF ACTUAL FREQUENCY

To capture eye movement accurately, the eye tracker should have a high sampling frequency. Generally, the theoretical

sampling frequency of an eye-tracking device is between 40 and 120 Hz. However, the actual sampling frequency is often different from the theoretical sampling frequency. In general, the actual sampling frequency fluctuates around the theoretical sampling frequency, and its random fluctuations cannot be predicted. Taking eye trackers with 40 and 60 Hz frequencies as examples, the actual tracking sampling frequencies are shown in Figure 2. Taking the eye tracker with 40 Hz sampling frequency as an example, its actual sampling frequency ranges from 5 to 120 Hz. The probabilities of these frequencies occurring are also presented in Figure 2. Sampling frequencies of 30–35 Hz account for a large proportion of the total, at about 70%. For the 60 Hz eye tracker, the sampling frequencies of 50–80 Hz account for a large proportion of 95% of the total.

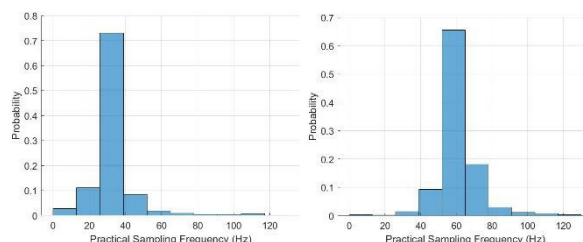


FIGURE 2. Practical sampling frequency distributions of eye trackers with 40 Hz (left) and 60 Hz (right) sampling frequencies in data collection.

Accordingly, the amount of eye movement data recorded per second differs due to random changes in the practical sampling frequency, seriously affecting the accuracy of subsequent eye movement analyses. Therefore, it is necessary to check and correct the sampling frequency before the classification and application of eye movement.

III. THE ENTIRE EYE MOVEMENT CLASSIFICATION PROCESS

The classification of eye movement behavior has an influence on the accuracy of research results regarding human visual characteristics. Therefore, in order to obtain accurate classification results for eye movement behavior, each step of

the eye movement behavior classification process is essential, including the pre-processing of the raw eye movement data, the classification (classification model construction), and the post-processing of the labeled eye movement data.

After these three steps have been applied, eye movement behavior can be accurately classified and used to extract further indicators, such as the fixation duration, fixation number, smooth tracking duration, fixation trajectory, tracking trajectory, and effective attention area. The complete process is shown in Figure 3.

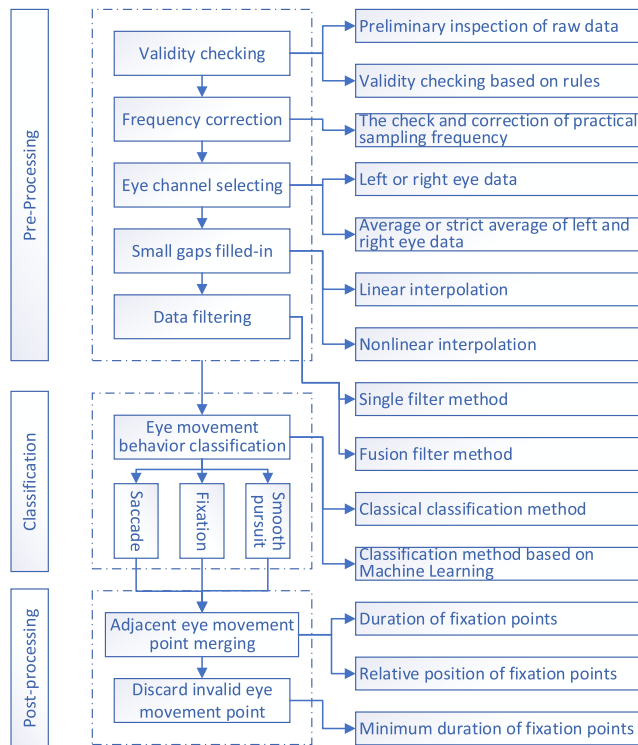


FIGURE 3. A complete process for eye movement behavior classification.

A. PRE-PROCESSING OF RAW EYE MOVEMENT DATA

Due to environmental noise, hardware problems, user blinking, head movement, and other behaviors during eye tracking, the raw eye movement data obtained may not be ideal. Therefore, pre-processing of the raw eye movement data is carried out. Five steps are included, namely, validity checking, frequency correction, channel selection, gap filling, and data filtering. This will be discussed in Part IV.

B. CLASSIFICATION

This step has three sub-steps. First, features containing rich and multi-dimensional information are constructed. Second, a model that is suitable for eye movement behavior classification is selected. Third, the constructed features are input into the classification model and the eye movement behavior is finally classified. This will be discussed in Parts V and VI.

C. POST-PROCESSING ON THE LABELED EYE MOVEMENT DATA

In this step, the classified and labeled eye movement data are merged into adjacent points, and any invalid points are discarded according to certain principles. This will be discussed in Part VII.

IV. EYE MOVEMENT PRE-PROCESSING

A. THE PURPOSE

In order to process the raw eye movement data, we designed a pre-processing framework consisting of four steps: validity checking, frequency correction, missing data filling, and data filtering.

B. THE PRE-PROCESSING PROCEDURE AND ITS ALGORITHM

1) VALIDITY CHECKING OF RAW EYE MOVEMENT DATA

After the collection of eye movement data, it is necessary to check its validity. Different eye trackers have different coordinate reference systems, as shown in Figure 4. Therefore, the collected eye movement data should be checked according to the position of the coordinate axis.

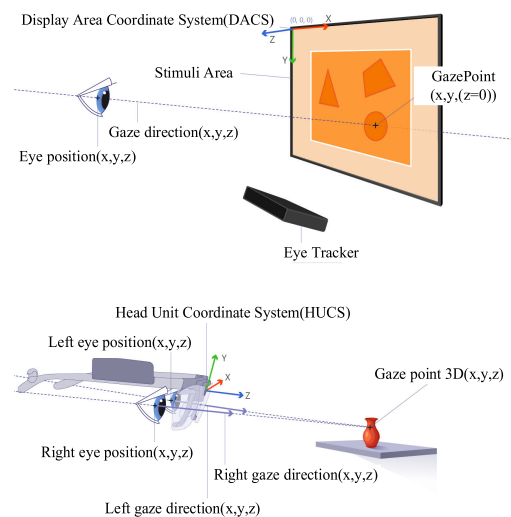


FIGURE 4. Remote eye-tracker (up) and head mounted eye-tracker (down).

2) CHECKING AND CORRECTION OF ACTUAL SAMPLING FREQUENCY

Irregular sampling frequency may easily lead to the complication of data processing, especially in terms of the synchronization of eye movement data and other data types. As a comparison with theoretical sampling, a schematic diagram of a section of actual records is presented in Figure 5 (first line). It shows that there can be several types of discrepancy between the actual sampling point and the theoretical sampling point.

The problems described above can be solved by correcting the sampling frequency. Then, there will be one record for

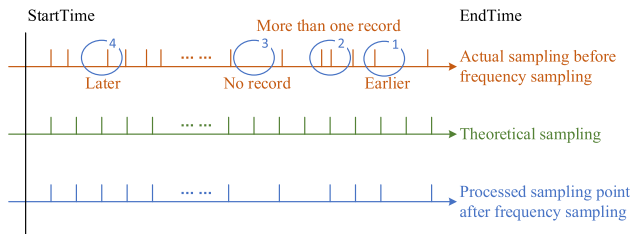


FIGURE 5. A schematic diagram of a section of actual records.

each ideal sampling moment, regardless of whether there are any data, as shown in Figure 5 (last line). If there are no data, the record is marked as blank. The correction process is as follows:

1. Calculate the ideal number of data points:

$$N = f(T_{end} - T_{start}) + 1 \quad (1)$$

where T_{start} and T_{end} are the start and end times for the eye-tracking experiment. f is the designed sampling frequency of the eye-tracker.

2. Calculate the ideal timestamp for each record:

$$t_i = T_{start} + \frac{i - 1}{f}, \quad i = 1, 2, \dots, N \quad (2)$$

3. Calculate the index value of the theoretical sampling points for the raw eye movement data:

$$Index_{ideal} = [t - T_{start}] \times f + 1 \quad (3)$$

It is important to note that there is only one sampling moment for each ideal record. At the beginning of the frequency correction, the values for each record are set to blanks.

4. Assign the closest practical record to each ideal one and fill in the blanks using the values from the assigned practical one. The allocation will follow the following principle,

$$\left\{ \begin{array}{l} Data_{Index_{ideal}}^{(i)} = Data_{Index_{practical}}^{(i)} \\ |t_{Index_{practical}} - t_{Index_{ideal}}| \leq \frac{1}{2}f \\ Data_{Index_{ideal}}^{(i)} = 0 \text{ or blank} \\ |t_{Index_{practical}} - t_{Index_{ideal}}| \geq \frac{1}{2}f \end{array} \right. \quad (4)$$

A selection of records obtained following the frequency correction is presented in Figure 6.

It is clear that the practical sampling time is the same as the theoretical sampling time after frequency correction. There is a small time offset for each practical record after frequency correction. Following frequency correction, the number of records collected in each second is equal to the designed sampling frequency. However, there are gaps between adjacent eye movement data points, so it is necessary to fill in the gaps to improve the integrity of the eye movement data.

3) FILLING IN SMALL GAPS

After completing the above steps, there will be some gaps between the adjacent eye movement data points.

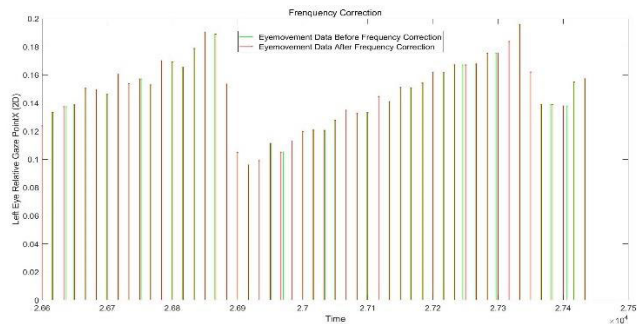


FIGURE 6. A section of records before and after frequency correction.

However, the reasons for the missing data points should be determined before data filling is implemented.

If the gaps are caused by the subject's movement, such as blinking, head rotating, etc., or by the eye tracker being blocked by something else, the gaps do not need to be filled. However, if the gaps are caused by sampling frequency corrections, it is necessary to fill them.

Therefore, the gaps should be filled according to certain rules. The duration of a blink is 75–425 ms. Therefore, a threshold of 75 ms can be used to judge whether the gaps should be filled or not [3]. For gaps with a duration of less than 75 ms, fill-in methods should be applied.

In general, gaps are filled by linear interpolation or by taking the average value based on the data points at both ends. The gap-filling process is described in reference [18], and the formula is as follows:

$$P_{gap} = P_f + \frac{P_f - P_a}{T_f - T_a}(T_{gap} - T_f) \quad (5)$$

where P_{gap} and T_{gap} are the parameter value and the sampling moment of the record in the gap, P_f and T_f are the parameter value and the sampling moment of the valid record before the gap, P_a and T_a are the parameter value and the sampling moment of the valid record after the gap.

Although the method described in references [18] and [19] is simple and easy to implement, the data that are filled in are greatly affected by the beginning and end eye movement data points, and there is no sampling noise or random error. As a result, the filled-in eye movement data are not realistic.

Therefore, another gap-filling method is proposed that involves the construction of a Fourier series equation. It is called the 3F (Fit Fourier Function) method and has the following steps:

1. Mark the missing data as $\{X_{(1)}, X_{(2)}, \dots, X_{(i)}\}$. These data should satisfy $(1000 \cdot i)/f \leq 75ms$.

2. Construct a Fourier series equation $f(x) = a_0 + \sum_{n=1}^{\infty} (a_n \cos nx + b_n \sin nx)$ with k data points before the missing data point and substitute the timestamp corresponding to the missing data point into the equation. The result of the calculation is that data point 1 is filled in and marked as $\{X_{(1)}^1, X_{(2)}^1, \dots, X_{(i)}^1\}$.

3. Construct a Fourier series equation $f(x) = a_0 + \sum_{n=1}^{\infty} (a_n \cos nx + b_n \sin nx)$ with k data points after the missing data point and substitute the timestamp corresponding to the missing data point into the equation. The result of the calculation is that data point 2 is filled in and marked as $\{X_{(1)}^2, X_{(2)}^2, \dots, X_{(i)}^2\}$.

4. Calculate the average value of the filled-in data points 1 and 2 to obtain the final filled-in data points: $\left\{ \frac{X_{(1)}^1 + X_{(1)}^2}{2}, \frac{X_{(2)}^1 + X_{(2)}^2}{2}, \dots, \frac{X_{(i)}^1 + X_{(i)}^2}{2} \right\}$

The contrasting data points filled in using the method proposed in this paper and linear interpolation can be seen in Figure 7. It shows that the data filled in by the 3F method are more realistic in terms of random noise.

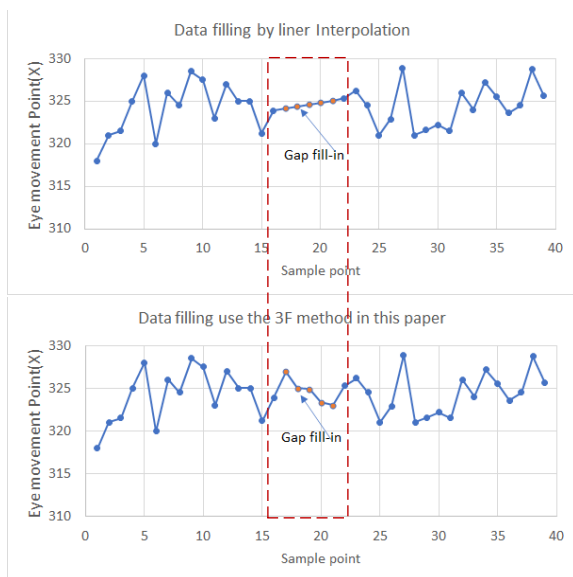


FIGURE 7. Gaps filled-in by linear interpolation (top) and the method proposed in this paper (bottom).

4) FUSION FILTERING

Noise is common in eye tracking, no matter what kind of eye tracker is used. There are many causes of noise, including the eye movement itself, as well as other environmental influences. Therefore, data filtering or smoothing is necessary. In order to reduce the level of noise, we propose a fusion filter method for improved bilateral convolution filtering and wavelet filtering for eye movement data [20]. The main processes are as follows:

1. Record the eye movement data to be denoised as $\{(X_1, Y_1)(X_2, Y_2)(X_3, Y_3) \dots (X_n, Y_n)\}$ and calculate the Euclidean distance $\{L_1, L_2 \dots L_{n-1}\}$ between adjacent eye movement points using the following formula:

$$L = \sqrt{(X_i - X_{i+1})^2 + (Y_i - Y_{i+1})^2} \quad (6)$$

2. Utilize the improved three-point bilateral convolution filter method to initially reduce the noise in the eye

movement data. The filtered eye movement data are recorded as $\{(X_{(1)}, Y_{(1)}), (X_{(2)}, Y_{(2)}), (X_{(3)}, Y_{(3)}) \dots (X_{(n)}, Y_{(n)})\}$, and the formulas used are as follows:

$$X_{(i+1)} = \alpha_1 * X_{(i)} + \alpha_2 * X_{(i+1)} + \alpha_3 * X_{(i+2)} \quad (7)$$

$$\alpha_1 = \frac{L_2}{2 * (L_1 + L_2)} \quad \alpha_3 = \frac{L_1}{2 * (L_1 + L_2)} \quad (8)$$

$$X_{(1)} = \beta_1 * X_{(1)} + \beta_2 * X_{(2)} \quad (9)$$

$$X_{(n)} = \beta_1 * X_{(n)} + \beta_2 * X_{(n-1)} \quad (10)$$

where $\alpha_1, \alpha_2, \alpha_3$ are the convolution kernel coefficients, $\alpha_2 = 0.5$; β_1, β_2 are the weight coefficients, $\beta_1 = 0.8, \beta_2 = 0.2$.

3. Perform wavelet filtering on the denoised eye movement data to obtain wavelet coefficients composed of the detailed noise and eye movement features. Based on the differences in the properties of the wavelet coefficients of the noise and eye movement data on different scales, the threshold is set to remove the wavelet coefficients in the noise and then obtain the denoised data through inverse transformation. The values of the wavelet denoising parameters are given in Table 2:

TABLE 2. Values of wavelet denoising parameters.

Parameter	Value
Type of wavelet base	db4
Wavelet threshold	Bayes
Wavelet decomposition layer	6
Threshold function	Soft

The processed data obtained by different filtering methods are compared with the raw eye movement data in Figure 8. It can be seen that the fluctuation characteristics of the raw data are perfectly preserved after the fusion filtering. The filtered result used in this paper is smoother than that obtained when filtering by the moving average method. The precision errors shown in Table 3 before and after data filtering show that the fusion filtering is much better, which could be beneficial for future studies involving the extraction of features from eye movement data.

TABLE 3. Contrasting effects of different filter methods.

Filter method	Precision error
Without filter	1.6095
Moving average	1.4044
Fusion filter	1.4034

V. FEATURE CONSTRUCTION FOR EYE MOVEMENT CLASSIFICATION

Raw eye movement data can be collected with an eye tracker. These data include the coordinate values of the gaze position, the coordinate values of the eye position, and the pupil diameter. Although these raw data can show different aspects of eye movement behavior, neither the coordinate values nor the pupil diameter can be used directly for the analysis of eye movement behavior. Therefore, based on these types

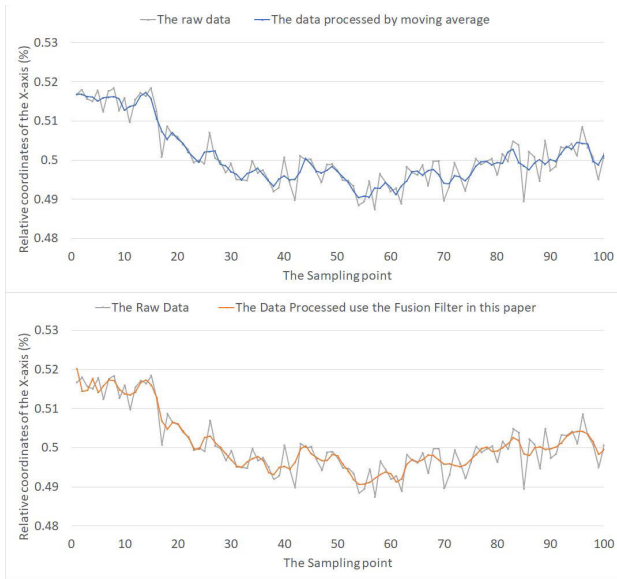


FIGURE 8. Eye movement data filtered by moving average method (top) and the fusion filter proposed in this paper (bottom).

of raw data, it is essential to extract the primary variables related to eye movement behavior, such as speed and distance. However, the meanings expressed by the above variables are too specific to be used to describe eye movement behavior comprehensively and deeply from multiple perspectives. It is therefore necessary to construct features with more depth and breadth based on the raw data and primary variables so as to accurately and reliably express different eye movement behaviors. This can also be beneficial for the classification of eye movement behaviors.

A. EXTRACTION OF TEMPORAL AND SPATIAL VARIABLES

The movement characteristics of the three kinds of eye movement are obviously different. The empirical values are shown in Table 4. According to previous studies, the most basic features used for eye movement classification are velocity and distance. These two features are generally called the primary features or variables. Most of the threshold-based eye movement classification algorithms determine the values of the thresholds using these two features.

TABLE 4. Empirical values of movement characteristics of the three kinds of eye movements.

	Velocity	Duration	Span
Fixation	0-30°/s	80~600ms	<0.5-1°
Saccade	30~500°/s	20~40ms	>0.5-1°
Smooth pursuit	1-30°/s	—	—

*Smooth pursuit has no duration, only a preparation time of about 90~150 ms

A schematic diagram of the gaze angle of eye movement is shown in Figure 9.

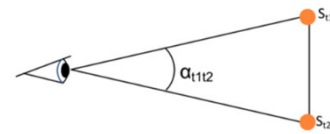


FIGURE 9. Gaze angle of eye movement.

The eye movement velocity V can be expressed by the following formula:

$$V = \frac{L}{|t_1 - t_2|} = \frac{\sqrt{(X_i - X_{i+1})^2 - (Y_i - Y_{i+1})^2}}{|t_1 - t_2|} \quad (11)$$

where t is the sampling time and L is the Euclidean distance between adjacent eye movement points.

The eye movement acceleration a can be expressed by the following formula:

$$a = \frac{V_{i+1} - V_i}{\Delta t} \quad (12)$$

where V is the eye moment velocity.

B. FEATURE CONSTRUCTION

Features derived from raw eye movement data are used as inputs for a machine learning algorithm. High-quality features (e.g., informative, relevant, non-redundant, interpretable, etc.) improve the understandability of algorithms, which is important for problem-solving [21]. Since the learning process relies on the exact information being delivered into the algorithms, features are the key to generating reliable and convincing classification results.

As the primary features of eye movement, speed and distance are highly correlated. For eye movement data collected by an eye tracker with a fixed sampling frequency, the following relationship exists between speed and distance:

$$V = L \cdot f \quad (13)$$

Therefore, these two features (variables) play the same role in the classification algorithm. Although the classification effect is better when eye movement behavior is based on speed and distance thresholds, it is limited to distinguishing between fixation and saccade points. It is difficult to classify smooth tracking behavior correctly and accurately using only the above two features.

To solve the above problems, in this paper, new features are constructed based on the primary features. At present, there are three methods of feature construction: the variable extraction method, the functional method, and the statistical analysis method. The most commonly used are the functional and statistical analysis methods. The former substitutes the data as independent variables into the function to obtain new variables; the latter obtains the statistical characteristics of the data (such as maximum values, extreme values, and variance).

The features constructed in this paper are based on the functional method and can be used to describe and measure

eye movement behavior in depth and from multiple perspectives. The features include

- the minimum coverage circle radius R ;
- the weighted average sum of the Euclidean distance ratios β and variance ΔL^2 of the adjacent eye movement points;
- the direction of movement α , the rate of change in the direction of movement $\Delta\alpha$, and the radius of curvature r of adjacent eye movement points;
- the acceleration of eye movement a .

These features are extracted to produce in-depth and multi-view descriptions and measures of eye movement behaviors. We not only focus on the movement characteristics (the range of movement) and movement trend (the directionality of movement) but also consider the information provided by the features in different sliding time windows for the classification of eye movement behavior. Finally, the feature set that can provide the largest amount of information in a specific time window is selected as the input for the classification model.

1) MINIMUM COVERAGE CIRCLE RADIUS R

The movement ranges of the three eye movement behaviors are all different. Compared with the fixation and smooth pursuit points, the movement range of saccades per unit of time is much larger. In other words, the faster an eye movement point moves, the greater the distance it travels per unit of time. If fixations are mixed with saccades, the movement range of these points will be larger. In order to quantitatively describe the eye movement range in the coordinate plane, the use of the minimum covering circle radius for a unit of time is proposed. This can be used to describe an eye movement point directly using the geometric shape in the coordinate plane. The feature not only considers the movement characteristics of the three eye movement behaviors but also considers their geometric characteristics in the coordinate plane. A geometrical diagram of the minimum coverage circle radius is shown in Figure 10.

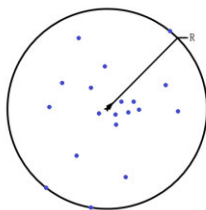


FIGURE 10. Geometrical diagram of the minimum coverage circle radius.

The specific calculation method used for this feature is as follows:

- The points collected for a certain unit of time are recorded as a point set $G_i = \{g_1, g_2, \dots, g_k\}$, where $k \geq 3$.
- Taking the line g_1g_2 as the diameter, the initial circle C_2 is obtained.

- The points in the point set $G_i = \{g_1, g_2, \dots, g_k\}$ are added in order. The current point is set as g_i , and if the point is in circle C_2 , then the radius of circle C_2 is the radius of the minimum coverage circle of the point set per unit of time.
- If not, a circle C_i with a diameter of g_1g_i is temporarily obtained, and the insertion point g_i must be on the boundary of the circle C_i .
- The circle C_i may not contain all points 1 to, but a point $g_j (j < i)$ that is not in C_i can be found. If a circle C_i is temporarily obtained for the diameter g_ig_j , g_i, g_j must be on the boundary of the circle.
- The circle C_j may not include all points 1 to j [as above]. If a point $g_k (k < j < i)$ that is not in C_i is found, g_i, g_j, g_k is used to establish a new circle. g_i, g_j, g_k must be on the boundary of the new circle. The radius of the new circle is the radius of the minimum coverage circle of the point set per unit of time, and this is denoted as R .

2) THE WEIGHTED AVERAGE SUM OF THE RATIOS AND THE VARIANCE OF THE EUCLIDEAN DISTANCE BETWEEN ADJACENT EYE MOVEMENT POINTS

The Euclidean distance between adjacent fixation points is small, while the Euclidean distance between adjacent saccade points is large. Therefore, the dispersion of Euclidean distance values between adjacent points is an important feature when describing differences in eye movement behavior. In the unit of time window, the lower the Euclidean distance dispersion, the simpler the type of eye movement in the window, and vice versa. Therefore, the Euclidean distance of the primary feature can be constructed, and its statistical features, including the ratios and the variance of adjacent Euclidean distances, can be used to describe the dispersion features of the distance. Therefore, the increment in the Euclidean distance can be used to describe the differences in eye movement behavior, so the use of the weighted average sum of the Euclidean distance ratios of adjacent eye movement points and the variance of the Euclidean distance within the unit of time window is proposed. These can describe the variation characteristics of eye movement points from different perspectives. The features above are beneficial for distinguishing different eye movement behaviors. The proposed features take full account of the sudden changes in distance caused by the appearance of a saccade, which can effectively distinguish fixation points (smooth tracking points) from saccade points. The specific calculation method used for the two above features is as follows:

- The eye movement dataset is denoted by $G = \{g_1, g_2, \dots, g_n\}$, where g_i is the i th eye movement tracking point, the corresponding coordinate point is $(X, Y, t), i = 1, 2, \dots, n$, and n is the number of tracking points contained in the set.
- The Euclidean distance L between adjacent eye movement points is calculated by

$$L(i, i + 1) = \sqrt{(X_i - X_{i+1})^2 + (Y_i - Y_{i+1})^2} \quad (14)$$

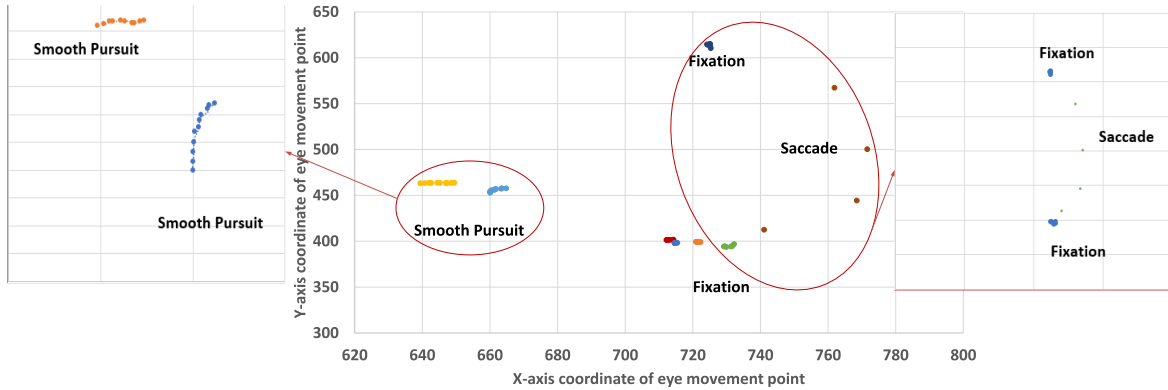


FIGURE 11. The movement characteristics of fixation, saccade, and smooth pursuit.

- The weighted average sum of the ratios of adjacent Euclidean distances per unit of time β can be calculated by

$$\beta = \frac{1}{n} \left(\frac{L(2, 3)}{L(1, 2)} + \dots + \frac{L(n, n+1)}{L(n-1, n)} \right) \quad (15)$$

- The variance in the Euclidean distance ΔL^2 in the input data can be calculated for a fixed unit of time. The Euclidean distance set $L_i = \{L_2, \dots, L_k\}$ of adjacent points can be calculated, where $k \geq 3$. The calculation formula is

$$\Delta L^2 = \frac{1}{k-1} \sum_{i=2}^k (L_i - \bar{L})^2 \quad (16)$$

3) THE DIRECTION AND RATE OF CHANGE IN THE DIRECTION OF EYE MOVEMENT POINTS AND THE CURVATURE RADIUS COMPOSED OF EYE MOVEMENT POINTS IN THE COORDINATE PLANE

When the eye follows a dynamic stimulus, it produces continuous feedback movement, which is called smooth pursuit. Due to the different causes of eye movements, the smooth pursuit movement trend is different from those of fixation and saccade, as can be seen in Figure 11. The movement trends of the fixation and saccade are disorganized and cannot form a fixed shape in the coordinate system plane, while the movement trend of smooth pursuit has a certain direction and can form a banded structure in the coordinate plane.

The tracking speed of smooth pursuit is slow but similar to the speed of the dynamic stimulus. In addition, the curves formed from eye movement in the coordinate plane have strong directivity, which means the curve-bending degree changes very little. Therefore, based on the movement characteristics of the smooth pursuit mentioned above, the use of the movement direction of adjacent eye movement points in the unit of time window and the curvature radius composed of eye movement in the coordinate plane is proposed. The former feature can be used to describe the eye movement trend, and the latter to describe the degree of curve-bending change.

In addition, during the process of smooth pursuit, the eye movement direction is the same as the motion direction of the observed dynamic stimulus, and there is no mutation in the direction. Once a mutation occurs, the eye movement behavior changes from smooth pursuit to saccade. That is to say, the rate of change of the eye movement direction will remain within a certain range during this period. Therefore, the rate of change of eye movement can be used to describe the eye movement behavior trend.

The specific calculation method for the above two features is as follows:

- When calculating the eye movement direction for continuous eye movement input data point α , the movement direction of the i th point is α_i , and the calculation formula is

$$\alpha_i = \arctan\left(\frac{Y_i - Y_{i-1}}{X_i - X_{i-1}}\right) \quad (17)$$

- The radius r of curvature is calculated for every three points for a fixed unit of time for the input data. By utilizing the point set $G_i = \{g_1, g_2, \dots, g_k\}$ included in a certain unit of time i , the distance between three consecutive points can be calculated. If the three points are not collinear, then

$$r = \frac{a}{2\sqrt{1 - \left(\frac{b^2 + c^2 - a^2}{2bc}\right)^2}} \quad (18)$$

VI. EYE MOVEMENT BEHAVIOR CLASSIFICATION USING A DECISION TREE

A. INTRODUCTION TO DECISION TREES

At present, the decision tree algorithm is one of the most popular machine-learning algorithms used in data mining. It is a nonparametric approach for building classification models. It does not require any prior assumption about the probability distribution governing the class and attributes of the data, and is thus applicable to a wide variety of datasets. Another appealing feature of the decision tree classifier is that the induced trees, especially the shorter ones, are relatively easy to interpret. The accuracies of the trees are also quite

comparable to other classification techniques for many simple datasets. Moreover, the employed techniques can quickly construct a reasonably good decision tree even when the training set size is very large, without requiring expensive computational efficiency. Finally, the decision tree algorithm is robust to noise [22].

Therefore, for the classification problem of eye movement behavior studied in this paper, the decision tree can be regarded as a very cost-effective algorithm. At present, examples of some well-known decision tree algorithms include CART, ID3, and C4.5, distinguishable mainly by different methods of splitting functions [23]–[25]. In this paper, the C4.5 algorithm, which can deal with continuous data, is selected. This algorithm uses the information gain rate as the splitting function method, which can overcome the shortcomings of other algorithms regarding inaccurate splitting.

B. SELECTION OF THE DATA SET

At present, there are few eye movement datasets published, because the annotation of eye movement data is complex and time-consuming. Therefore, we used GazeCom recordings for both training and testing through a strict cross-validation procedure. The GazeCom dataset was collected in Karl Gegenfurtner’s lab at the University of Giessen, and the GazeCom project is funded by the European Commission (contract no. IST-C-033816) within the Information Society Technologies (IST) priority of the 6th Framework Programme [26].

All eye movement recordings were made with an SR Research EyeLink II eye tracker using information from pupil and corneal reflections to estimate gaze at a frequency of 250 Hz. The 54 subjects were students (age range 18 to 34 years) at the Psychology Department of Giessen University who were paid for their participation. The total number of individual labels was about 4.3 million, of which 72.5%, 10.5%, 11%, and 5.9% were labeled as fixations, saccades, smooth pursuits, and noise, respectively. The eye movement data labels were manually assigned by two experts, and the format of the data was.arff. The data content included the sampling time, x,y coordinates of the gaze point, confidence level of the gaze data, labels from the two experts, and final labels. A segment of the dataset is shown in Table 5.

Given that the amount of data in each folder is sufficient (about 200,000-250,000 data points) and the number of subjects in each folder is the same (54 subjects), eye movement data from two independent dynamic videos in the GazeCom dataset are selected for the final dataset used in this paper. One dataset is split into a training set and a verification set in a ratio of 7:3. The training set is used to fit the data samples, and the verification set is used to adjust the parameters of the decision tree model and evaluate the classification ability of the model. The other dataset is used as the test set, with the data run through the constructed classification model to evaluate its accuracy.

TABLE 5. Information in a GazeCom segment.

Label	0	UNKNOWN
	1	FIX(Fixation)
	2	SACCADE
	3	SP(Smooth Pursuit)
	4	NOISE
METADATA	width px	1280.0
	height px	720.0
	width mm	400.0
	height mm	225.0
	distance mm	450.0
	relation	gaze_labels
ATTRIBUTE	time	INTEGER
	x	NUMERIC
	y	NUMERIC
	confidence	NUMERIC
	handlabeler1	INTEGER
	handlabeler2	INTEGER
	handlabeler_final	INTEGER
Data sample	2000	Time
	620.8	width_px
	289.2	height_px
	1.0	confidence
	0	handlabeler1
	0	handlabeler2
	1	handlabeler_final

C. DETERMINATION OF INITIAL PARAMETERS

To set the model parameters, the following aspects need to be considered: window length, window moving step size, and window label selection rules. These are shown in Table 6.

TABLE 6. Model parameter setting.

window length	three eye movement points
window moving step	one eye movement point
window label	last eye movement point’s label in a window

D. MODEL OPTIMIZATION AND TEST ANALYSIS

Overfitting is one of the main challenges with the decision tree algorithm. Without any restrictions, the established model can provide 100% accuracy for the training set, but this accuracy level does not translate to the test set. Therefore, it is important to prevent the model from overfitting. Generally, the overfitting problem can be solved by two methods: one is to constrain the size of the tree; the other is tree pruning. In addition, the correlations among attributes in the dataset can easily be ignored in the modeling process. To address the above problem, we first analyzed the correlations among the features used for the classification of eye movement behavior using the Pearson correlation coefficient method. Features with low correlations were then selected as the inputs to the decision tree model. The Pearson correlations between the features are shown in Table 7.

Table 7 shows that the correlation coefficients between the primary features and constructional features exceed 0.5. Therefore, the primary features are removed because they contain less information and cannot be explained from multiple views.

TABLE 7. Pearson correlations between features.

β	β	$\Delta\alpha$	R	ΔL^2	r	V	α	a
β	1							
$\Delta\alpha$	-0.0076	1						
R	0.2438	-0.0050	1					
ΔL^2	0.2547	-0.0109	0.5869	1				
r	0.0004	-0.0099	-0.0073	-0.0007	1			
V	0.4166	-0.0003	0.8397	0.4840	-0.0065	1		
α	-0.0055	-0.5984	-0.0736	-0.0173	0.0058	-0.0608	1	
a	0.3919	0.0086	0.0057	-0.0091	0.0001	0.5468	0.0004	1

The final feature set ($R, \beta, \Delta L^2, \Delta\alpha, r$) is obtained as the feature inputs for the decision tree.

In order to improve the generalizability of the decision tree model and avoid overfitting, we adopt the tree pruning method to reduce the complexity of the tree by deleting the branches of unimportant features. This further improves the performance of the tree and the predictive ability of the established model. An improved cost-complexity pruning (CCP) algorithm is proposed. Based on the traditional CCP algorithm, the improved method considers the influence of the model depth on the pruning process. If a decision tree is deep enough, then the deeper the level is, the greater the likelihood of it being pruned will be.

The improved CCP method includes the following steps:

(1) A sequence of subtrees $\{T_0, T_1, \dots, T_n\}$ is generated from the original decision tree T_0 . Among them, T_{i+1} is generated from T_i , and T_n is the root node.

(2) The optimal decision tree is selected based on the true error estimation of the tree from the sequence of subtrees generated in step 1.

In step 1, the basic idea of generating a sequence of subtrees $\{T_0, T_1, \dots, T_n\}$ is to cut branch T_i with the smallest increase in the training dataset error to obtain T_{i+1} .

The increase in the error rate after the tree branch is clipped is

$$\theta = \frac{[R(t) - R(T_i)]}{(N_1 - N_2) * h} \tag{19}$$

where N_1 and N_2 are the number of nodes before and after pruning, respectively; $R(t)$ is the error cost of the node, $R(t) = r(t) * p(t)$; $r(t)$ is the misclassification sample rate of node t ; $p(t)$ is the proportion of all samples that fall into node t ; $R(T_i)$ is the subtree error cost, $R(T_i) = \sum R(i)$; i is the leaf node of subtree T_i ; and h is the depth of the decision tree at which the pre-pruned node is located.

Once θ has been calculated for each non-leaf node of decision tree T_0 , the subtree with the smallest θ value is circulated and cut until the root node is left. A series of pruned trees $\{T_0, T_1, T_2 \dots T_m\}$ can be obtained, and the optimal decision tree can be selected according to the true error.

The pseudo code for this procedure is shown in Figure 12.

The evaluation indexes of the classification algorithm usually include the accuracy, precision, recall, and F1-score, which is a comprehensive evaluation index. The confusion matrix after classification is constructed to calculate the

Pseudocode: Improved Pruning

```

Input: Decision Tree  $T_0$ ;
Process:
1: let  $T = T_0, k = 0$ ;
2: start function TreePruning( $T, k$ )
3:   Initialize  $\alpha = +\infty$ 
4:   if  $T$  is the root node then
5:     return
6:   end if
7:   for each of the internal node  $t$  in  $T$  do
8:     calculate  $\theta = \frac{[R(t) - R(T_i)]}{(N_1 - N_2) * h}$ ;
9:      $\alpha = \min(\alpha, \theta)$ ;
10:  end for
11:  for each of the internal node  $t$  in  $T$  (traversal from top to bottom) do
12:    if  $\theta = \alpha$  then
13:      the node  $t$  is pruned, using majority method to determine the category of node  $T$ , get a decision tree; break
14:    end if
15:  end for
16:   $k = k + 1, \alpha_i = \alpha, T_i = T$ 
17:   $\{T_{i+1}, \alpha_{i+1}\} = \text{TreePruning}(T, k)$ 
18:   $T_{i+1} = (T_i, T_{i+1}), \alpha_{i+1} = (\alpha_i, \alpha_{i+1})$ 
19:  return  $\{T_{i+1}, \alpha_{i+1}\}$ 
20: end Function
21:  $T_{i+1} = (T_i, T_{i+1}), \alpha_{i+1} = (0, \alpha_{i+1})$ ;
22: Using test set  $Test$ , using cross validation method to choose a best subtree  $T_n$  in the sequence of subtrees  $T_{i+1} = (T_i, T_{i+1}, \dots, T_i)$ 
Output:  $T_n$ 
    
```

FIGURE 12. A brief description of the improved pruning.

TABLE 8. The evaluation indexes of the constructed decision tree model before pruning.

Label	Fixation	Saccade	Smooth pursuit	Accuracy
Precision	0.9087	0.9799	0.9872	0.9215
Recall	0.9983	0.8933	0.5823	
Specificity	0.6865	0.9979	0.9845	
F1-score	0.9514	0.9346	0.7323	

TABLE 9. The evaluation indexes of the constructed decision tree model after pruning.

Label	Fixation	Saccade	Smooth pursuit	Accuracy
Precision	0.8640	0.9799	0.9859	0.8876
Recall	0.9982	0.8933	0.4879	
Specificity	0.5941	0.9979	0.9988	
F1-score	0.9263	0.9346	0.6529	

above evaluation indexes. The evaluation indexes of the constructed decision tree model before and after pruning are shown in Tables 8 and 9, respectively.

The numbers of layers in the constructed decision tree model are, respectively, 32 and 20 before and after pruning. The runtime comparison can be seen in the following table.

TABLE 10. The evaluation indexes of the constructed decision tree model after pruning.

	Model before pruning	Model after pruning
runtime	39.8747s	38.9149s

By comparing the classification accuracy of the decision tree model and the number of layers on the tree before and after pruning, it can be observed that, although the classification accuracy of the model is reduced by pruning, the number of decision tree model layers is also reduced. Compared with the unpruned model, the pruned model is simpler, and its generalizability is improved.

The F1-score's comparison with other algorithms is shown in Table 11. Those of the method proposed in this paper are higher than those of the other algorithms for all three eye movement behaviors.

TABLE 11. The F1-score comparison between the proposed method and other algorithms.

Labels	F1-score		
	Fixations	Saccades	Pursuits
Final model	92.63%	93.46%	65.29%
[Agtzidis et al.2016] ^[26]	88.60%	86.40%	64.60%
I-VMP ^[27]	90.90%	68.53%	58.10%
[Larsson et al.2015] ^[12]	91.20%	86.10%	45.90%
[Berg et al.2009] ^[28]	88.30%	69.70%	42.20%

The test set accuracy of the eye movement classification model proposed in this paper is shown in Table 12.

TABLE 12. The test set accuracy of the eye movement classification model proposed in this paper.

Label	Fixation	Saccade	Smooth pursuit
Accuracy	76.19%	93.3%	81.03%

The classification algorithm proposed in this paper has higher accuracy for saccade and smooth pursuit, but slightly lower accuracy for fixation. As fixation and smooth pursuit are similar, the classification accuracy of these two eye movement behaviors is mutually restricted. Although the algorithm proposed in this paper has slightly lower classification accuracy for fixation, the classification accuracy for smooth pursuit is greatly improved. The overall accuracy of the improved decision tree algorithm is better than that of the existing classification algorithms. This is beneficial for the study of visual characteristics in dynamic stimulus scenarios.

VII. THE POST-PROCESSING OF EYE MOVEMENT BEHAVIOR CLASSIFICATION

A. MERGE ADJACENT FIXATIONS

After the eye movement behavior has been classified, the duration of the fixation points is longer than that of the other eye movement behaviors. Therefore, a fixation may be divided into two shorter ones by noise or saccades, which means that some post-processing of the data is required after

the eye movement classification has been completed in order to identify and merge fixations that are very close in time and space.

The combination of adjacent fixation points can be judged based on two parameters: the interval time and the span angle between the two fixation points. The values set for the two parameters are shown in Table 13.

TABLE 13. The parameter values for merging adjacent fixations.

Parameter	Value
Max time between fixations	75ms
Max angle between fixations	0.5°

According to the physiological characteristics of human eyes, a blink takes at least 75 ms. When the interval between two fixations is greater than 75 ms, there may have been a blink or an eye closure or some other reason for the eye movement not having been picked up by the eye tracker. Therefore, when the interval is less than 75 ms, the adjacent fixation points should be merged. Since the angle between fixations is usually less than 0.5–1°, the fixations should also be merged when the angle between them is less than 0.5°.

B. DISCARD SHORT FIXATIONS

Fixations with very short durations can also exist after the merging operation. Based on the characteristics of visual information acquisition, short-term fixation behavior is meaningless because the short term is not long enough to obtain useful information. Therefore, a second judgment can be made according to the duration of fixations, and the value is suggested to be 60 ms [19].

VIII. ESTIMATION OF THE PREVIEW DISTANCE AND TIME-TO-COLLISION IN DRIVING

A. THE APPLICATION OF EYE MOVEMENT TECHNOLOGY IN DRIVING

Drivers need to get all kinds of information from the road environment when driving, and this information is received through the sense organs [1]. Statistical studies show that 80-90% of information related to the traffic environment is received through vision when driving [13]. Drivers' cognitive ability, attention, and mental state while driving is very important to driving safety. Fortunately, these aspects can be quantitatively analyzed and studied based on eye movement [29], [30]. At present, the study and application of eye movements in driving mainly focus on the following aspects:

- Visual characteristics of drivers, such as saccade amplitude, distribution of fixation points, fixation duration, etc., in different driving environments [31].
- Cognitive characteristics of drivers, such as focus target, estimation of distance or speed etc. [32].
- Visual search characteristics of drivers, such as area of interest, visual search scope, fixation transfer, etc. [33], [34].

- Preview characteristics of drivers, such as preview distance, preview time, etc. [35].
- Attention characteristics of drivers, such as concentration of attention, fixation area, gaze duration, etc. [36], [37].
- Fatigue and distraction state monitoring in driving, such as frequency of blink, Perclos, fixation direction etc. [38].
- Intention recognition in driving, such as lane changing or overtaking intention recognition, etc. [39].
- Visual load assessment, such as changes in pupil diameter, visual tremor and so on [31].

No matter what the application or type of research, drivers mainly obtain traffic information through fixation and smooth pursuit behaviors when driving [40]. In addition, based on the motion direction, moving objects in the longitudinal direction are detected through fixation points, and moving objects in the lateral direction are obtained through smooth tracking [41]. Therefore, the application of eye movement behavior to driving in this paper mainly focuses on two aspects. One is to study the driver's preview characteristics; that is, the driver's preview time or preview distance can be estimated from fixations. The other application is innovative, in that traffic conflicts and collision risk can be estimated from smooth pursuits [42]. Next, the solution algorithm for the preview time and the estimation of the time-to-collision in traffic conflicts will be introduced.

B. SOLUTION ALGORITHM FOR THE PREVIEW TIME

The driver's preview behavior refers to the information perception process by which the driver obtains driving information by observing the road traffic environment during the driving process. Here, the driving information includes the vehicle's motion state, road width, changes in road curvature, pedestrians or other vehicles, etc. Preview behavior is the most important information source for drivers, so it is an important parameter to be considered in driver modeling. Typical models are Guo Konghui's original preview optimal curvature model and Professor MacAdam's optimal seedling control model [43], [44].

1) THEORETICAL DEDUCTION OF PREVIEW-TIME-SOLVING ALGORITHM FOR A STRAIGHT SECTION AND A CURVED SECTION

The driver's preview time on a straight section can be calculated by using the eye's pitch angle and the vertical distance from the driver's eyes to the ground. The calculation diagram is shown in Figure 13 and the formula is as follows:

$$PreviewDistance_s = H / \tan \varphi \quad (20)$$

where H is the vertical distance from the driver's eyes to the ground, the unit is m, and the height needs to be measured before the experiment because it will be different for each driver. φ is the pitch angle of the eyes.

The preview time can be expressed as the ratio of the driver's preview distance to the vehicle's current velocity:

$$PreviewTime_s = H/V \cdot \tan \varphi \quad (21)$$

where V is the instantaneous velocity of the vehicle and the unit is m/s.

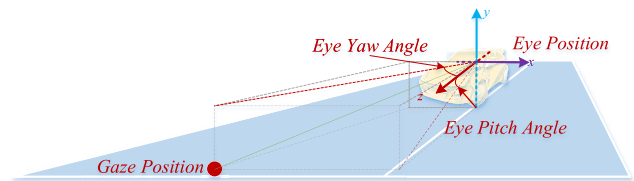


FIGURE 13. Preview distance calculation diagram on a straight section.

When driving on a curved section, the driver's preview time can be calculated by using the pitch angle and the yaw angle of the eyes and the vertical distance from the driver's eyes to the ground. The calculation diagram is shown in Figure 14 and the formula is as follows:

$$PreviewTime_c = H/V \cdot \tan(\tan(\varphi) \cos(\beta)) \quad (22)$$

where β is the yaw angle of the eyes.

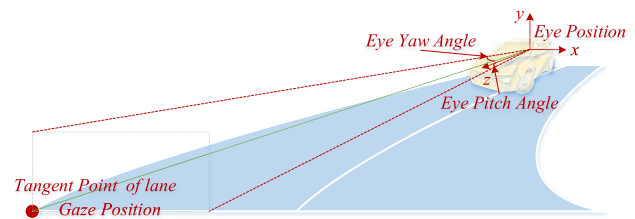


FIGURE 14. Preview distance calculation diagram for a curved section.

2) REGRESSION ANALYSIS OF PREVIEW TIME AND ROAD TURNING RADIUS

The preview time is influenced by many factors, such as the radius of the curvature of the road, the speed of driving, and the characteristics of the driver, among which the most important factor is the road turning radius. In order to obtain the mathematical relationship between the preview time and the road turning radius, an experiment was carried out in this research. The driving simulator, experiment scene, experiment route, and some subjects are shown in Figure 15.

The driving simulator has eight degrees of freedom. The experimental scenario generated using UC-win Road is a section of city expressway of length 10km. The road section includes straight sections and curved sections with various turning radii. The vehicle dynamics model comes from CarSIM. The driver's eye movements were collected using the remote eye-tracker of FaceLab and 17 subjects' data were collected in this experiment.

By processing the eye movement data, the preview time of the drivers under different road curvature radii can be calculated using formulas 20-22, and the solution formula for the road turning radius is as (23), shown at the bottom of the next page, where R is the road turning radius, x , y are



FIGURE 15. Pictures of actual simulation experiment.

the coordinates of real-time trajectory points, and the number of trajectory points in the sliding window is $2n + 1$.

Because the raw calculated preview times do not obey a normal distribution, the regression analysis cannot be carried out directly. A logarithmic transformation of the raw calculated preview times must be carried out so that they obey a normal distribution. The distribution histograms of PT and $\ln(PT)$ can be seen in Figure 16.

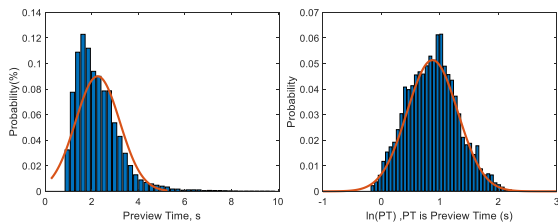


FIGURE 16. The distributions of PT (left) and $\ln(PT)$ (right).

The road turning radii are grouped and the median of $\ln(PT)$ is calculated in each group. The scatter diagram of $\ln(PT)$ and each group of road turning radii is shown in Figure 17. It can be seen from the figure that the scatter plot can be fitted by an exponential function with the following formula:

$$PT = 3.071 - 1.515 \times \exp\left(-\left(\frac{R}{465.9}\right)^2\right) \quad (24)$$

The coefficient of determination is 0.995, which shows that the formula above can explain 99.5% of the samples in Figure 17.

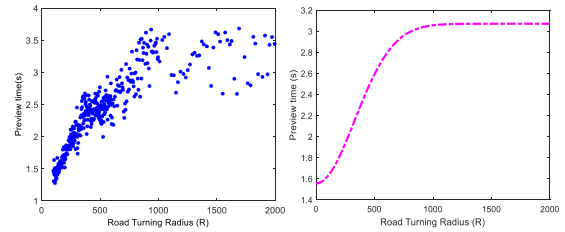


FIGURE 17. The scatter diagram (left) and regression curve (right) of $\ln(PT)$ and each group of road turning radii.

C. ESTIMATION OF TIME-TO-COLLISION BASED ON SMOOTH PURSUIT

In the natural environment, humans can recognize and track the movement of an object. Even if the object disappears or is blocked for a short time, humans can express and predict its space and time characteristics (such as speed and position). This slow, smooth tracking behavior is called smooth pursuit eye movement (SPEM). The visual information received in the SPEM process can be used to judge the position, velocity, moving trajectory, and time required for a moving object to reach a specific position. Therefore, the driver can estimate the initial motion of the lateral object and the final time-to-collision using SPEM. The calculation formula is as follows:

$$TTC = \beta / \omega_{SPEM} \quad (25)$$

where ω_{SPEM} is the angular velocity of smooth pursuit.

In this study, a collision experiment between vehicles and pedestrians is designed to verify the accuracy of the estimated time-to-collision. The eye movements are collected by Tobii Glasses2, and the experiment is conducted on a real road. To reduce the error introduced by other parameters in the experiment, the values of some parameters are fixed. For example, the yaw angle of the driver's eyes is fixed by fixing the start positions (20m and 30m away from the intersection) of the vehicle and the pedestrian, in order to reduce the psychological pressure on the pedestrian caused by vehicle movement and the impact of the yaw angle collection error on the driver's smooth pursuit. A schematic diagram of the experiment is shown in Figure 18.

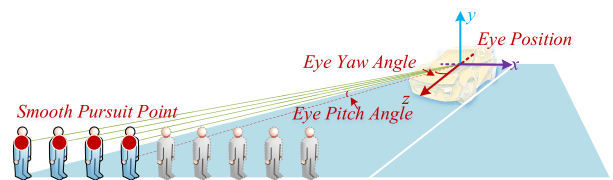


FIGURE 18. Schematic diagram of the conflict experiment.

In the experiment, pedestrians cross a road at three speeds (fast, medium, and slow), and the experiment is repeated

$$\lambda = \frac{1}{R} = \frac{\arccos\left(\frac{(x_{i+n}-x_i)(x_{i+2n}-x_{i+n})+(y_{i+n}-y_i)(y_{i+2n}-y_{i+n})}{\sqrt{(x_{i+n}-x_i)^2+(y_{i+n}-y_i)^2}\sqrt{(x_{i+2n}-x_{i+n})^2+(y_{i+2n}-y_{i+n})^2}}\right)}{\sqrt{(x_{i+2n}-x_i)^2+(y_{i+2n}-y_i)^2}} \quad (23)$$

TABLE 14. The experimental data and calculation results of the conflict experiment between a pedestrian and a vehicle.

Sample	Longitudinal distance of vehicle	Lateral distance of pedestrians	Velocity of pedestrians	Yaw angle	Angular velocity of smooth pursuit	Estimated time-to-collision	Actual arrival time	Time error
1	20	5	fast	14.04	2.5861	5.42	4.97	9.05%
2	20	5	fast	14.04	2.7661	5.07	4.60	10.22%
3	20	5	fast	14.04	3.2768	4.28	4.90	12.65%
4	20	5	medium	14.04	2.5976	5.40	5.47	1.22%
5	20	5	medium	14.04	2.1012	6.68	6.20	7.74%
6	20	5	slow	14.04	2.0115	6.98	7.26	3.88%
7	30	5	fast	9.46	1.7520	5.40	5.01	7.80%
8	30	5	fast	9.46	1.57123	6.02	5.51	9.29%
9	30	5	medium	9.46	1.42789	6.63	5.96	11.19%
10	30	5	medium	9.46	1.407	6.73	6.58	2.20%
11	30	5	slow	9.46	1.44256	6.56	7.06	7.10%
12	30	5	slow	9.46	1.3935	6.79	7.23	6.07%



FIGURE 19. Actual photos of conflict experiment.

five times for each speed. Thus, 15 experiments are conducted. Those with poor eye movement data are removed, leaving 12 experiments whose data are finally collected. Smooth pursuit behavior data are collected when the drivers are tracking the moving pedestrians, and the average tracking angular velocity of the data in the first 0.5–1s is calculated to estimate the time-to-collision, that is the time it will take the pedestrians to reach the conflict point. Actual pictures of the experiment are shown in Figure 19.

The eye movement data from one experiment is shown in Figure 20. It can be seen from Figure 20 that, when the pedestrian is still, the driver’s eye movement behavior is that of fixation (blue points in Figure 20). When the pedestrian begins to move, the eye movement of the driver changes from fixation (blue points) to smooth pursuit (red points) to track the moving pedestrian, during which some saccades (green points) are mixed in. Therefore, when the smooth pursuit can be accurately classified, its velocity can be used to estimate the time-to-collision. The experimental data and calculation results are shown in Table 14.

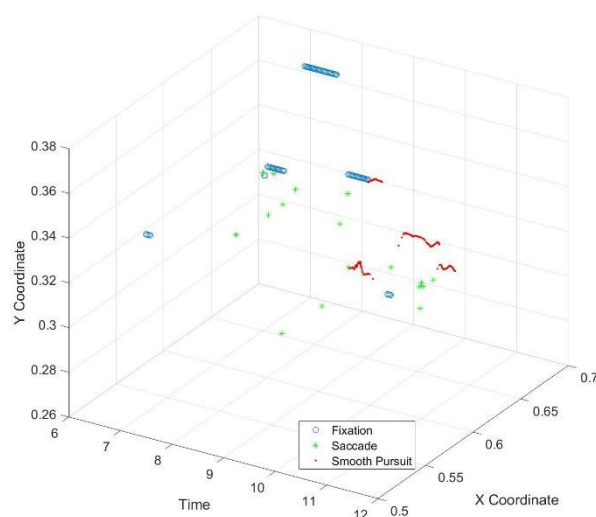


FIGURE 20. Real eye movement data showing fixation, saccade, and smooth pursuit points from a sample of the conflict experiment between a pedestrian and a vehicle.



FIGURE 21. Line chart of estimated and actual times-to-collision.

The estimated time-to-collision is calculated using equation (25), and the actual arrival time is recorded by a timer. The average estimated accuracy error is found to be 7.37%, and a line chart of the estimated and actual times-to-collision can be seen in Figure 21.

IX. DISCUSSION AND CONCLUSION

Classifying raw data into the different types of eye movements is an important part of eye-tracking research. It is essential to classify eye movement behavior accurately using an appropriate classification algorithm. Importantly, the classification of eye movements should be a complete process, including the three steps of pre-processing, classification, and post-processing. However, it is very uncommon for all of these steps to be included in the eye-tracking literature when eye movement classification is discussed. Therefore, in this paper, a refined pre-processing of eye movements and classification of eye movement behavior is studied. Moreover, the application of the classified eye movement behaviors to the driving field is described. We will now discuss and summarize what we have done in this paper.

Firstly, a universal procedure for raw data pre-processing has been designed. The three steps of this pre-processing have been improved and optimized. We proposed a sampling correction method, a 3F method for filling in gaps, and a fusion filtering method for the last step. Although we encountered difficulties in the process of algorithm improvement, for example, some of the parameters in the fusion filter are determined by an empirical method, the improved methods are more effective than the existing methods.

Secondly, five new features of eye movement behavior classification have been constructed, and the movement and distribution characteristics of three eye movement behaviors mined, especially for smooth pursuit. The features proposed not only consider the movement ranges and trends of different eye movement behaviors but also consider the information provided by each feature under different time windows. This is critical for the subsequent machine learning, because the quality of the features constructed will directly determine the effect of the classification.

Thirdly, for the machine learning, the decision tree algorithm was used for the classification of eye movement behaviors. Considering that overfitting is the most challenging problem for the decision tree algorithm, in order to improve the generalizability of the decision tree algorithm, we proposed a post-pruning method that considers the depth of the tree. The F1-scores of the proposed algorithm for the classification of fixation, saccade, and smooth pursuit are 92.63%, 93.46%, and 65.2% respectively. The results show that the proposed method is highly accurate in classifying eye movement behaviors.

Finally, the application of fixation and smooth pursuit behaviors in driving has been presented. One application is the estimation of the preview time using fixation points. The preview times calculated using fixation are mostly distributed around 1-6s, which is more practical than the traditional setting of 1s. Also, a regression function between the preview time and the road turning radii was derived, which is very important for the study of the guiding mechanism of drivers' visual perception features for vehicle steering. Another application described in this paper was the estimation of time-to-collision using the smooth pursuit eye movements of the

driver. The average estimated accuracy error was found to be 7.37%. Although only a few samples were collected due to the limitations of the test conditions, and the reliability of the data needs to be further verified, this is still a very pleasing result, which implies that smooth pursuit can be used to develop driver warning assistance systems in lateral confliction.

REFERENCES

- [1] T. H. Rockwell, "Skill, judgment and information acquisition in driving," in *Human Factors in Highway Traffic Safety Research*, T. W. Forbes, Ed. New York, NY, USA: Wiley, 1972, pp. 133–164.
- [2] S. Cannon, "The neurology of eye movements (contemporary neurology series)," *Arch. Ophthalmol.*, vol. 110, no. 3, p. 326, Mar. 1992, doi: [10.1001/archophth.1992.01080150024016](https://doi.org/10.1001/archophth.1992.01080150024016).
- [3] D. D. Salvucci and J. H. Goldberg, "Identifying fixations and saccades in eye-tracking protocols," in *Eye Tracking Research & Applications*. New York, NY, USA: ACM Press, 2000, pp. 71–78.
- [4] O. V. Komogortsev, D. V. Gobert, U. K. S. Jayarathna, D. H. Koh, and S. M. Gowda, "Standardization of automated analyses of oculomotor fixation and saccadic behaviors," *IEEE Trans. Biomed. Eng.*, vol. 57, no. 11, pp. 2635–2645, Nov. 2010, doi: [10.1109/TBME.2010.2057429](https://doi.org/10.1109/TBME.2010.2057429).
- [5] O. V. Komogortsev and A. Karpov, "Automated classification and scoring of smooth pursuit eye movements in the presence of fixations and saccades," *Behav. Res. Methods*, vol. 45, no. 1, pp. 203–215, 2013, doi: [10.3758/s13428-012-0234-9](https://doi.org/10.3758/s13428-012-0234-9).
- [6] M. Dorr, T. Martinetz, K. R. Gegenfurtner, and E. Barth, "Variability of eye movements when viewing dynamic natural scenes," *J. Vis.*, vol. 10, no. 10, p. 28, 2010.
- [7] T. Santini, W. Fuhl, T. Kübler, and E. Kasnecki, "Bayesian identification of fixations, saccades, and smooth pursuits," in *Eye Tracking Research & Applications*. Charleston, SC, USA: ACM Press, 2016, pp. 163–170.
- [8] C. Z. Bao, "Analysis of eye tracking data and algorithms research on data fusion," M.S. thesis, ECE, Eng., XDU, Xi'an, China, 2018.
- [9] M. Vidal, A. Bulling, and H. Gellersen, "Detection of smooth pursuits using eye movement shape features," in *Eye Tracking Research and Applications*. Charleston, SC, USA: ACM Press, 2012, pp. 177–180.
- [10] M. Startsev, I. Agtzidis, and M. Dorr, "1D CNN with BLSTM for automated classification of fixations, saccades, and smooth pursuits," *Behav. Res. Methods*, vol. 51, no. 2, pp. 556–572, Apr. 2019, doi: [10.3758/s13428-018-1144-2](https://doi.org/10.3758/s13428-018-1144-2).
- [11] S. Hoppe and A. Bulling, "End-to-end eye movement detection using convolutional neural networks," 2016, *arXiv:1609.02452*. [Online]. Available: <https://arxiv.org/abs/1609.02452>
- [12] L. Larsson, M. Nyström, R. Andersson, and M. Stridh, "Detection of fixations and smooth pursuit movements in high-speed eye-tracking data," *Biomed. Signal Process. Control*, vol. 18, pp. 145–152, Apr. 2015, doi: [10.1016/j.bspc.2014.12.008](https://doi.org/10.1016/j.bspc.2014.12.008).
- [13] M. Carrasco, "Visual attention: The past 25 years," *Vis. Res.*, vol. 51, no. 13, pp. 1484–1525, 2011, doi: [10.1016/j.visres.2011.04.012](https://doi.org/10.1016/j.visres.2011.04.012).
- [14] M. F. Land, "Eye movements and the control of actions in everyday life," *Prog. Retinal Eye Res.*, vol. 25, no. 3, pp. 296–324, 2006, doi: [10.1016/j.preteyeres.2006.01.002](https://doi.org/10.1016/j.preteyeres.2006.01.002).
- [15] G. L. Yan and H. J. Tian, "A review of eye movement recording methods and techniques," *Chin. J. Appl. Psychol.*, vol. 10, no. 2, pp. 55–58, 2004.
- [16] L. Chamberlain, "Eye tracking methodology: Theory and practice," *Qualitative Market Res., An Int. J.*, vol. 10, no. 2, pp. 217–220, 2007, doi: [10.1108/13522750710740862](https://doi.org/10.1108/13522750710740862).
- [17] J. K. Y. Ong and T. Haslwanter, "Measuring torsional eye movements by tracking stable iris features," *J. Neurosci. Methods*, vol. 192, no. 2, pp. 261–267, Oct. 2010, doi: [10.1016/j.jneumeth.2010.08.004](https://doi.org/10.1016/j.jneumeth.2010.08.004).
- [18] B. Liu, Q.-C. Zhao, Y.-Y. Ren, Q.-J. Wang, and X.-L. Zheng, "An elaborate algorithm for automatic processing of eye movement data and identifying fixations in eye-tracking experiments," *Adv. Mech. Eng.*, vol. 10, no. 5, May 2018, Art. no. 168781401877367, doi: [10.1177/1687814018773678](https://doi.org/10.1177/1687814018773678).
- [19] A. Olsen. (2012). *The Tobii I-VT Fixation Filter: Algorithm description*. Tobii Technology. [Online]. Available: https://www.tobii.com/Global/Analysis/Training/WhitePapers/Tobii_WhitePaper_TobiiIVTFixationFilter.pdf
- [20] Z. W. Yang, L. L. Zheng, H. S. Qu, and Y. Wu, "Image denoising research of combining mean filtering with Poisson kernel improved bilateral filtering," *Comput. Simul.*, vol. 37, no. 9, pp. 460–463, 2020.

- [21] X. Shi, Y. D. Wong, M. Z.-F. Li, C. Palanisamy, and C. Chai, "A feature learning approach based on XGBoost for driving assessment and risk prediction," *Accident Anal. Prevention*, vol. 129, pp. 170–179, Aug. 2019.
- [22] P. N. Tan, M. Steinback, and V. Kumar, "Decision tree," in *Introduction to Data Mining*, 2nd, ed. New York, NY, USA: Post & Telecom Press, 2011.
- [23] L. Breiman, J. H. Friedman, R. Olshen, and C. J. Stone, *Classification and Regression Trees*. Belmont, CA, USA: Wadsworth, 1984.
- [24] G. V. Kass, "An exploratory technique for investigating large quantities of categorical data," *Appl. Statist.*, vol. 29, no. 2, pp. 119–127, 1980, doi: [10.2307/2986296](https://doi.org/10.2307/2986296).
- [25] J. R. Quinlan, "Discovering rules by induction from large collection of examples," in *Expert Systems in the Micro Electronic Age*. Edinburgh, U.K.: Edinburgh Univ. Press, 1979.
- [26] I. Agtzidis, M. Startsev, and M. Dorr, "Smooth pursuit detection based on multiple observers," in *Eye Tracking Research & Applications*. Columbia, SC, USA, 2016, pp. 303–306.
- [27] J. S. Agustin, "Off-the-shelf gaze interaction," Ph.D. dissertation, Dept. IT, ITU, Copenhagen, Denmark, 2009.
- [28] D. J. Berg, S. E. Boehnke, R. A. Marino, D. P. Munoz, and L. Itti, "Free viewing of dynamic stimuli by humans and monkeys," *J. Vis.*, vol. 9, no. 5, p. 19, 2009, doi: [10.1167/9.5.19](https://doi.org/10.1167/9.5.19).
- [29] S. W. Savage, D. D. Potter, and B. W. Tatler, "The effects of cognitive distraction on behavioural, oculomotor and electrophysiological metrics during a driving hazard perception task," *Accident Anal. Prevention*, vol. 138, Apr. 2020, Art. no. 105469, doi: [10.1016/j.aap.2020.105469](https://doi.org/10.1016/j.aap.2020.105469).
- [30] C. Braunagel, D. Geisler, W. Stolzmann, W. Rosenstiel, and E. Kasneci, "On the necessity of adaptive eye movement classification in conditionally automated driving scenarios," in *Proc. 9th Biennial ACM Symp. Eye Tracking Res. Appl.*, Columbia, SC, USA, Mar. 2016, pp. 19–26.
- [31] X. S. Li, F. S. Meng, X. L. Zheng, Y. Y. Ren, and J. H. Yan, "Driver's visual characteristics based on stress response," *J. Jilin Univ.*, vol. 47, no. 5, pp. 1403–1410, 2017.
- [32] D. M. Zhuang and X. R. Wanyan, "Research on pilot's visual attention allocation modeling and validation experiment based on cognitive characteristics," *Sci. Technol. Innov. Herald*, vol. 13, no. 13, pp. 178–179, 2016, doi: [10.3969/j.issn.1674-098X.2016.13.096](https://doi.org/10.3969/j.issn.1674-098X.2016.13.096).
- [33] C. Robbins and P. Chapman, "How does drivers' visual search change as a function of experience? A systematic review and meta-analysis," *Accident Anal. Prevention*, vol. 132, Nov. 2019, Art. no. 105266, doi: [10.1016/j.aap.2019.105266](https://doi.org/10.1016/j.aap.2019.105266).
- [34] C. Xia, J. Han, and D. Zhang, "Evaluation of saccadic scanpath prediction: Subjective assessment database and recurrent neural network based metric," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Jun. 15, 2020, doi: [10.1109/TPAMI.2020.3002168](https://doi.org/10.1109/TPAMI.2020.3002168).
- [35] X. L. Zheng, X. S. Li, Y. Y. Ren, D. W. Xing, J. H. Yan, and Q. C. Zhao, "Modelling of driver's preview time based on visual characteristics," *China J. Highway Transp.*, vol. 31, no. 5, pp. 160–166, 2018, doi: [10.19721/j.cnki.1001-7372.2018.05.019](https://doi.org/10.19721/j.cnki.1001-7372.2018.05.019).
- [36] L. F. Zhang, "Research on situation-awareness attention distribution system," M.S. thesis, ME, Eng. BUPT, Beijing, China, 2020.
- [37] D. Zhang, B. Wang, G. Wang, Q. Zhang, J. Zhang, J. Han, and Z. You, "Onfocus detection: Identifying individual-camera eye contact from unconstrained images," 2021, *arXiv:2103.15307*. [Online]. Available: <https://arxiv.org/abs/2103.15307>
- [38] X. Hu and G. Lodewijks, "Detecting fatigue in car drivers and aircraft pilots by using non-invasive measures: The value of differentiation of sleepiness and mental fatigue," *J. Saf. Res.*, vol. 72, pp. 173–187, Feb. 2020, doi: [10.1016/j.jsr.2019.12.015](https://doi.org/10.1016/j.jsr.2019.12.015).
- [39] J. Feng, "Research on driving characteristics of lane-changing vehicle based on driver behavior," M.S. thesis, DOA, Eng. WUT, Wu'han, China, 2018.
- [40] A. Rasouli, I. Kotseruba, and J. K. Tsotsos, "Agreeing to cross: How drivers and pedestrians communicate," in *Proc. IEEE Intell. Vehicles Symp.*, Los Angeles, CA, USA, Jun. 2017, pp. 264–269.
- [41] C.-J. Chang and M. Jazayeri, "Integration of speed and time for estimating time to contact," *Proc. Nat. Acad. Sci. USA*, vol. 115, no. 12, pp. E2879–E2887, Mar. 2018, doi: [10.1073/pnas.1713316115](https://doi.org/10.1073/pnas.1713316115).
- [42] E. Kasneci, G. Kasneci, T. Kubler, and W. Rosenstiel, "Online recognition of fixations, saccades, and smooth pursuits for automated analysis of traffic hazard perception," *Artif. Neural Netw.*, vol. 4, pp. 411–434, Jan. 2015, doi: [10.1007/978-3-319-09903-3_20](https://doi.org/10.1007/978-3-319-09903-3_20).
- [43] K. H. Guo, "Vehicle handling stability evaluation," in *Principles of Vehicle Handling Dynamics*, 1st ed. Jiangsu, China: Jiangsu Science and Technology Press, 2011.
- [44] C. C. Macadam and G. E. Johnson, "Application of elementary neural networks and preview sensors for representing driver steering control behaviour," *Vehicle Syst. Dyn.*, vol. 25, no. 1, pp. 3–30, Jan. 1996, doi: [10.1080/00423119608968955](https://doi.org/10.1080/00423119608968955).



XIAN-SHENG LI received the bachelor's degree in automobile application engineering from Jilin University, Changchun, China, in 1982, and the Ph.D. degree in vehicle operation engineering from the School of Transportation, Jilin University. His research interests include driving safety and reliability and transportation system resources optimization.



ZHI-ZHEN FAN received the bachelor's degree in transportation (automotive application engineering) from Jilin University, in 2019, where he is currently pursuing the master's degree in vehicle application engineering with the School of Transportation. His research interests include the characteristics of driver visual and driver assistance system development which includes vehicle detection and warning.



YUAN-YUAN REN was born in Jilin City, Jilin Province, in 1982. She received the B.S. and M.S. degrees in traffic information engineering and control from Jilin University, in 2006 and 2008, respectively, and the Ph.D. degree in vehicle operation engineering from Jilin University, in 2011. Her research interests include driving stability and safety technology, intelligent analysis of driving behavior, and intelligent vehicle planning and control research.



XUE-LIAN ZHENG was born in Shandong, in 1987. She received the B.S. degree in automobile engineering from Northeast Forestry University, in 2009, and the Ph.D. degree in vehicle operation engineering from Jilin University, in 2014. She currently works with the School of Transportation, Jilin University, as the Vice President. Her research interests include vehicle dynamics and control, driving behavior, and autonomous vehicle design and control.



RAN YANG received the master's degree (French engineering degree) in electronics and embedded systems from the École Nationale Supérieure de l'Électronique et de ses Applications (ENSEA). After returning to China, he is fully responsible for the engineering software and hardware research and development team project and technological independent innovation. He is currently a Research and Development Director of Kingfar Company Ltd. He has participated in the research and development of a number of equipment in the field of military industry and national defense ergonomics. He has published many academic papers and obtained many national invention patents. He has won provincial and ministerial science and technology awards.