

# Description

---

## MULTIMODAL GAZE ESTIMATION METHOD, SYSTEM, ELECTRONIC DEVICE AND STORAGE MEDIUM BASED ON LIGHTWEIGHT GRAPH NEURAL NETWORK

### **TECHNICAL FIELD**

The present application relates to the technology fields of human-computer interaction application and gaze estimation, specifically relates to a multimodal gaze estimation method, system, electronic device and storage medium based on a lightweight graph neural network.

### **BACKGROUND**

There is a growing demand for eye gaze estimation technology in human-computer interaction, virtual and augmented reality systems, eye tracking applications, and assistive technologies.

However, the growing demand also exposes key limitations of existing solutions: 1) Current appearance-based gaze estimation systems often process multimodal inputs through independent computational branches, preventing the effective integration of spatial relationships between facial features, eye position, and head orientation data; 2) Model-based require specialized hardware, such as infrared illumination and spot detection systems, making them unsuitable for consumer devices and unconstrained environments; 3) Existing state-of-the-art systems typically require 7-10 million parameters, creating a huge computational burden that prevents the deployment of gaze estimation related technologies on mobile devices, embedded systems, and AR/VR wearables; 4) Current architecture has the problem of poor generalization ability across datasets, which requires a large number of subject-specific calibration procedures that are time-consuming and expensive to calculate.

Based on this, the present application proposes a multimodal gaze estimation technology based on a lightweight graph neural network in order to solve one or more of the problems mentioned above.

## SUMMARY

### (1) Technical problems solved

In view of the shortcomings of the prior art, the present application provides a multimodal gaze estimation method, system, electronic device and storage medium based on a lightweight graph neural network, which is suitable for lightweight and accurate gaze estimation suitable for mobile deployment, and at the same time realizes effective multimodal feature integration and rapid specific subject adaptation, and solves one or more of the problems mentioned above.

### (2) Technical solutions

To achieve the above objectives, the present invention is implemented through the following technical solutions.

In a first aspect, a multimodal gaze estimation method based on a lightweight graph neural network (GNN) is provided, wherein the method comprises:

obtaining multimodal heterogeneous data related to and complementary for gaze direction estimation, wherein the multimodal heterogeneous data comprises visual appearance data, geometric orientation information and spatial positioning coordinates;

extracting visual features from the visual appearance data based on depthwise separable convolution and squeeze-and-excitation module;

projecting the visual features, the geometric orientation information, and the spatial positioning coordinates into a common vector space;

constructing a graph structure, wherein the graph structure comprises a plurality of nodes representing the multimodal heterogeneous data or features corresponding to the multimodal heterogeneous data;

processing the constructed graph using a multi-head attention graph neural network having a two-layer architecture to obtain graph-level features, wherein a first layer of the two-layer architecture expands dimensionality of node features, and a second layer of the two-layer architecture reduces dimensionality of the expanded dimension;

aggregating the graph-level features and fusing the aggregated graph-level features with the visual features, the geometric orientation information, and the spatial positioning coordinates;

performing gaze estimation prediction based on the fused features using a fully connected layer; and

converting the gaze estimation prediction into a normalized three-dimensional gaze vector based on coordinate conversion.

In an embodiment, the visual appearance data comprises facial image data, eye cropping image data, the geometric orientation information comprises head rotation data, and the spatial positioning coordinate comprises three-dimensional eye center position.

In an embodiment, constructing a graph structure comprises:

constructing a fully connected graph with six nodes; wherein the six nodes comprise nodes corresponding to: facial features, left eye features, right eye features, head rotation data, left eye three-dimensional position, and right eye three-dimensional position; and

establishing learnable edge weights between pairs of the six nodes.

In an embodiment, coordinate conversion comprises converting spherical coordinates into Cartesian coordinates.

In an embodiment, the method further comprises:

when estimating the gaze for a specific subject, incorporating subject-specific calibration data into the multimodal heterogeneous data.

In an embodiment, the subject-specific calibration comprises:

updating only the specific subject calibration embedding and final fully connected layer parameters during calibration, while maintaining all other network parameters in a frozen state.

In an embodiment, processing the constructed graph structure using a multi-head attention graph neural network having a two-layer architecture to obtain graph-level features comprises:

Step 4.1: initializing the six graph nodes using the projected feature embedding;

Step 4.2: calculating attention coefficients of all node pairs using linear projection;

Step 4.3: applying LeakyReLU activation and softmax normalization to the attention coefficients;

Step 4.4: aggregating neighboring features using weighted attention coefficients;

Step 4.5: determining whether current processing corresponds to a first GNN layer,

and if yes, proceeding to step 4.6; if no, proceeding to step 4.7

Step 4.6: expanding node features to 256 dimensions;

Step 4.7: reducing node features to 32 dimensions;

Step 4.8: applying a second GNN layer processing;

Step 4.9: flattening and concatenating all node features into a unified representation;

and

Step 4.10: outputting a final graph-level feature vector.

In an embodiment, converting the gaze estimation prediction into a normalized three-dimensional gaze vector based on coordinate conversion comprises:

constructing an orthogonal normalized coordinate frame using a cross product; and  
generating a parametric ray emanating from the three-dimensional eye center.

Preferably, the parametric ray is capable of performing intersection calculations with objects in spatial applications, the objects comprising virtual objects, user interfaces, and physical surfaces.

In an embodiment, the multi-head attention comprises four attention heads processed in parallel, wherein:

a first attention head is configured to capture specific relationship patterns among the six input modalities represented within the six-node graph structure;

a second attention head is configured to capture alternative relationship patterns among the input modalities;

a third attention head is configured to provide additional parallel processing capacity for comprehensive relationship modeling; and

a fourth attention head is configured to complete a parallel processing array by providing a fourth independent attention mechanism.

In a second aspect, a multimodal gaze estimation system based on a lightweight graph neural network is provided, wherein the system comprises:

a multimodal input module configured to obtain multimodal heterogeneous data related to and complementary for gaze direction estimation, wherein the multimodal heterogeneous data comprises visual appearance data, geometric orientation information, and spatial positioning coordinates;

a feature extraction module configured to extract visual features from the visual appearance data based on depthwise separable convolution and squeeze-and-excitation module;

a projection module configured to project the visual features, the geometric orientation information, and the spatial positioning coordinates into a common vector space;

a graph construction module configured to construct a graph structure, wherein the graph structure comprises a plurality of nodes representing the multimodal heterogeneous data or features corresponding to the multimodal heterogeneous data;

a multi-head attention graph neural network configured to process the constructed graph using a multi-head attention graph neural network having a two-layer architecture to obtain graph-level features, wherein a first layer of the two-layer architecture expands dimensionality of node features, and a second layer of the two-layer architecture reduces dimensionality of the expanded dimension;

a feature fusion module configured to aggregate the graph-level features and fuse the aggregated graph-level features with the visual features, the geometric orientation information, and the spatial positioning coordinates;

a regression module configured to perform gaze estimation prediction based on the fused features using a fully connected layer; and

a three-dimensional reconstruction module configured to convert the gaze estimation prediction into a normalized three-dimensional gaze vector based on coordinate conversion.

In a third aspect, an electronic device is provided, wherein the electronic device comprises a memory, a processor, and a computer program stored on the memory and executable on the processor, wherein the processor, when executing the program, implements the steps of the multimodal gaze estimation method based on a lightweight graph neural network according to any one above.

In a fourth aspect, a non-transitory computer-readable storage medium having stored thereon a computer program is provided, wherein the computer program, when executed by a processor, implements the steps of the multimodal gaze estimation method based on

a lightweight graph neural network according to any one above.

### (3) Beneficial effects

The present application provides a multimodal gaze estimation method, a system, an electronic device and a storage medium based on a lightweight graph neural network. Compared with the prior art, the present application has the following beneficial effects:

1. The present application processes heterogeneous inputs through a unified feature coding framework, comprising facial images, eye crop images, head pose data and 3D eye center position, which solves the problem in the prior art by using independent branches of convolutional neural networks to process multimodal inputs, resulting in independent processing of facial images, eye crop images and head pose data, and the inability to meaningfully model the spatial relationship across modalities. This approach can effectively capture complex relationship structures, with better feature fusion and cross-modal interaction.

2. The present application constructs a fully connected graph with six nodes, representing different modalities, and adopts a multi-head attention mechanism to model cross-modal relationships and dependencies.

3. The present application can unify feature projection, map all heterogeneous inputs (vision, geometry, and calibration data) into a common 32-dimensional embedding space, and realize coherent reasoning across modalities. Two-layer graph neural networks employ attention-based reasoning to capture spatial and semantic relationships between facial components, head orientation, and eye position.

4. The present application reconstructs the 3D gaze vector, converts the predicted pitch angle and yaw angle into mathematically stable 3D rays, which is suitable for intersecting calculations with screens, surfaces, and augmented reality or virtual reality coordinate frames.

5. The present application combines the calibration embedding of specific subjects, which can achieve rapid personalization with the least number of training samples, and improve the generalization ability across data sets while maintaining computational efficiency.

6. The complete system of the present application has fewer parameters (about 3.48 million) and realizes real-time performance, making it suitable for deployment in mobile devices, embedded systems, and low-power AR/VR platforms.

## **BRIEF DESCRIPTION OF THE DRAWINGS**

In order to more clearly illustrate the technical solutions in the embodiments of the present application or prior art, the following will briefly introduce the drawings required in the embodiment or the description of the prior art. It is obvious that the drawings in the following description are only some embodiments of the present application, and for ordinary skilled persons in the field, other drawings can be obtained according to these drawings without paying creative labor.

Fig. 1 is a flow chart of a multimodal gaze estimation method based on lightweight graph neural network in the present application.

FIG. 2 is a system architecture diagram of a LightGazeNet system shown in an embodiment.

FIG. 3 is a flow chart of a calibration method for a specific body shown in an embodiment.

FIG. 4 is a flow chart of the graph neural network processing method shown in an embodiment.

FIG. 5 is a flow chart of the three-dimensional gaze vector reconstruction method shown in an embodiment.

FIG. 6 is an architecture of a feature embedding component shown in an embodiment.

FIG. 7 is a system diagram of a gaze estimation architecture shown in an embodiment.

FIG. 8 is a flow chart of a gaze estimation process shown in the embodiment.

FIG. 9 is a flow chart of a graph neural network processing method shown in an embodiment.

FIG. 10 is a flow chart of a gaze estimation processing line shown in an embodiment.

FIG. 11 is a sequence diagram of a multimodal gaze estimation process shown in an embodiment.

FIG. 12 is a sequence diagram of a multimodal gaze estimation data processing workflow shown in an embodiment.

FIG. 13 is a sequence diagram of a graph neural network processing workflow shown in an embodiment.

FIG. 14 is a sequence diagram of a multimodal feature integration and three-dimensional gaze vector reconstruction process shown in an embodiment.

FIG. 15 is a system diagram of a LightGazeNet deployment system shown in an embodiment.

FIG. 16 is a multi-head attention architecture for processing multimodal gaze estimation data shown in an embodiment.

## **DETAILED DESCRIPTION OF THE INVENTION**

In order to make the purpose, technical solution and advantages of the embodiment of the present application clearer, the technical solutions in the embodiments of the present application are described clearly and completely. It is obvious that the described embodiment is a part of the embodiment of the present application, not all of the embodiments. Based on the embodiments in the present application, all other embodiments obtained by ordinary skill in the art without performing creative labor fall within the protection scope of the present application.

It should be noted that a multimodal gaze estimation method, system, electronic device and storage medium based on a lightweight graph neural network proposed in the present application can be applied in, but are not limited to, the following scenarios:

Implemented in virtual and augmented reality systems to provide accurate real-time gaze tracking for immersive user interfaces, enabling natural interaction with virtual objects, menu navigation, and foveated rendering optimization, reducing computational load by rendering high-resolution graphics only where the user is looking.

Proposed as a lightweight gaze estimation system for mobile devices and smartphones, enabling hands-free control of applications, providing accessibility features for users with motor disabilities, and enhancing user experience in games and social media applications through gaze-based interaction mechanisms.

Implemented in automotive systems to provide driver attention monitoring and drowsiness detection, improving vehicle safety by warning the drivers when the driver's gaze pattern indicates distraction or fatigue, and enabling a gaze-controlled infotainment system that allows the driver to stay focused on the road.

Implemented in human-computer interaction systems on desktop and laptop computers, providing a more intuitive user interface with gaze-based cursor movement, text selection, and application switching, which is particularly beneficial for users with physical disabilities who may have limited mobility or limited flexibility.

Proposed as gaze estimation solutions for eye tracking research and medical diagnostics, enabling researchers and clinicians to study visual attention patterns, diagnose neurological disorders, assess cognitive function, and monitor rehabilitation progress using portable, cost-effective devices that do not require specialized infrared hardware.

Implemented in educational technology and e-learning platforms, it analyzes students' attention and engagement levels during online courses, provides teachers with valuable feedback on the effectiveness of content, and enables adaptive learning systems that adjust presentation based on students' attention focus.

Implemented in retail and marketing applications to provide personalized content recommendations, measure advertising effectiveness, and create engaging shopping experiences in stores and public spaces through digital signage and interactive displays that respond to customer gaze patterns.

### **Example 1:**

In a first aspect, the present application first proposes a multimodal gaze estimation method based on a lightweight graph neural network, see FIG. 1, the method comprises:

S1. Obtaining multimodal heterogeneous data related to and complementary for gaze direction estimation, wherein the multimodal heterogeneous data comprises visual appearance data, geometric orientation information and spatial positioning coordinates;

S2. Extracting a visual feature from the visual appearance data based on depthwise separable convolution and squeeze-and-excitation module;

S3. Project the visual feature, the geometric orientation information, and the spatial

positioning coordinate into a common vector space;

S4. Constructing a graph structure, wherein the graph structure comprises a plurality of nodes representing the multimodal heterogeneous data or features corresponding to the multimodal heterogeneous data;

S5. Processing the constructed graph using a multi-head attention graph neural network having a two-layer structure to obtain a graph-level feature; wherein a first layer of the two-layer structure expands dimensionality of node features, and a second layer of the two-layer structure reduces dimensionality of the expanded dimension.

S6. Aggregating the graph-level feature, and fusing them with the visual feature, the geometric orientation information, and the spatial positioning coordinate;

S7. Utilizing a fully connected layer to perform gaze estimation prediction based on the fused features; and

S8. Converting the gaze estimation prediction into a normalized three-dimensional gaze vector based on coordinate conversion.

The multimodal gaze estimation method based on the lightweight graph neural network proposed in this embodiment realizes real-time gaze estimation performance through system dimension transformation and lightweight processing operations implemented in sequential steps. The method can achieve computational efficiency suitable for mobile deployments while maintaining cross-modal relationship modeling capabilities, enhancing gaze estimation accuracy compared to traditional independent processing methods.

The implementation process of an embodiment of the present application will be described in detail below with FIG. 1-16 and the explanation of the specific steps S1-S8.

S1. Obtaining a multimodal heterogeneous data related to and complementary for a gaze direction estimation, wherein the multimodal heterogeneous data comprises visual appearance data, geometric orientation information, and spatial positioning coordinates.

The multimodal heterogeneous data are obtained, which provide complementary information for the gaze direction estimation. The multimodal heterogeneous data comprises visual appearance data from facial areas, geometric orientation information, and spatial positioning coordinates. In some embodiments, the visual appearance data

from the facial area comprises but is not limited to facial image data, and eye cropping image data; the geometric orientation information comprises but is not limited to user head rotation data; and the spatial positioning coordinate comprises but is not limited to the three-dimensional eye center position.

S2. Extracting a visual feature from the visual appearance data based on the depthwise separable convolution and a squeeze-and-excitation module.

In this step, the visual appearance data, comprising data such as a facial image and an eye crop, is processed by lightweight convolutional operations and the squeeze-and-excitation module to extract visual features. In some embodiments, this step can eventually generate 576-dimensional feature vectors that capture unique visual features from input data from the facial and eye regions.

It should be noted that, with the help of the squeeze-and-excitation module, feature channel response is adaptively recalibrated through squeeze (global information aggregation) and excitation (channel weight learning), which improves the feature expression ability of the convolutional neural network without increasing the computational cost. This operation reduces computational effort while maintaining the effectiveness of feature extraction.

S3. Projecting the visual feature, the geometric orientation information, and the spatial positioning coordinate into a common vector space based on a linear transformation with a GELU activation function.

In this step, all heterogeneous inputs are projected into the same vector space (preferably a common 32-dimensional embedding space) and converted into a unified dimensional representation, enabling coherent cross-modal processing. The heterogeneous inputs, comprising the extracted visual feature, the geometric orientation information, and the spatial positioning coordinate, are converted into a unified dimensional representation, enabling coherent cross-modal processing. In some embodiments, a linear transformation with a GELU activation function is applied to map 576-dimensional visual features into a standardized 32-dimensional space, while geometric orientation information (e.g., a head rotation matrix) and spatial positioning coordinate (e.g., a 3D eye center coordinate) are processed through separate projection

paths. For example, the head rotation data is flattened into a nine-dimensional three-by-three matrix, which is projected into a 32-dimensional embedded space. The three-dimensional eye center position comprises left eye center coordinate and right eye center coordinate, both projected from three dimensions to 32 dimensions.

S4. Constructing a graph structure, wherein the graph structure comprises a plurality of nodes representing the multimodal heterogeneous data or features corresponding to the multimodal heterogeneous data.

In this step, a graph structure is created, and a fully connected graph is constructed, which comprises a plurality of nodes representing the multimodal heterogeneous data or features corresponding to the multimodal heterogeneous data, representing the input of multimodal heterogeneous data. In some embodiments, the fully connected graph comprises at least six nodes, each node corresponding to a processed input type, specifically comprising facial features, left eye features, right eye features, head rotation data, left eye 3D position, and right eye 3D position. Preferably, learnable edge weights can be established between all node pairs to realize comprehensive relationship modeling between different input modes.

S5. Processing the constructed graph using a multi-head attention graph neural network having a two-layer structure to obtain a graph-level feature; wherein a first layer of the two-layer structure expands dimensionality of node features, and a second layer of the two-layer structure reduces dimensionality of the expanded dimension.

In this step, the fully connected graph constructed in the above steps is processed by a multi-head attention graph neural network having a two-layer structure to obtain a graph-level feature. An attention mechanism is applied, and relational information such as attention coefficients for all node pairs is calculated using linear projection, connection operation, and LeakyReLU activation function to obtain the graph-level feature. In some embodiments, a parallel attention head is used to capture multiple relational patterns at the same time, with the first layer expanding the node features to 256 dimensions and the second layer reducing the features back to 32 dimensions to improve computational efficiency. Preferably, multi-head attention graph neural networks comprise four parallel attention heads that operate using independent parameter sets to capture different

relational patterns simultaneously.

S6. Aggregating the graph-level feature, and fusing them with the visual feature, the geometric orientation information, and the spatial positioning coordinates.

In this step, graph-level features are fused with original CNN features and geometric inputs. Specifically, the relational information captured by graph neural network processing (i.e., graph-level feature) is combined with original feature representations from the original CNN features (i.e., extracted visual feature) and original geometric features (i.e., geometric orientation information and spatial positioning coordinate). In some embodiments, a unified 384-dimensional relationship vector resulting from graph processing is integrated with data based on visual appearance features and spatial orientation to create a comprehensive multimodal representation.

S7. Utilizing a fully connected layer and performing gaze estimation prediction based on the fusion features.

In this step, a fully connected layer is used to predict the gaze estimation angle. The predicted angle at least comprises a gaze pitch angle and a gaze yaw angle. The fused feature representation is processed by the fully connected processing network, and the integrated feature is mapped to the angular gaze direction parameter by applying nonlinear transformation of the fully connected processing network. Pitch and yaw angle predictions can be generated, which represent the direction of the gaze in spherical coordinate format for subsequent three-dimensional conversion operations.

S8. Converting the gaze estimation prediction into a normalized three-dimensional gaze vector based on coordinate conversion.

In this step, a spherical to Cartesian transformation is used to convert the predicted angle into a normalized three-dimensional gaze vector. In some embodiments, the trigonometric function is applied by the three-dimensional gaze vector reconstruction module to convert the angle prediction into a Cartesian coordinate representation, generating a unit vector representing the direction of the gaze in three-dimensional space, and can work in tandem with the orthogonal base building unit and the camera coordinate transformation unit to generate parametric rays suitable for spatial interaction applications and coordinate system integration.

Referring to Figure 2, a comprehensive system architecture for multimodal gaze estimation based on lightweight graph neural network according to the present method is provided. The architecture converts heterogeneous input data into accurate three-dimensional gaze prediction through four sequential processing stages. This method implements a hierarchical structure that systematically handles diverse input modalities through specialized components, including processing feature extraction, dimensional normalization, relational modeling, and vector reconstruction operations.

### 2.1) Multimodal input processing.

Multimodal input processing receives and organizes six different types of input data for comprehensive gaze estimation processing. In specific implementation, multimodal input processing comprises:

Facial image input, which provides overall facial appearance information captured from the user's facial area. Facial image input can provide visual features that capture facial structure and orientation, as well as contextual information around the eye area.

Left eye crop and right eye crop, which provide specialized visual data from a single eye area. Left crop and right eye crop can provide detailed appearance information from each eye that enables a centralized analysis of eye-specific features, comprising pupil position, iris pattern, and eyelid configuration. In some embodiments, the left eye crop and the right eye crop can be generated by precise positioning of the key points of the eye, and then the cropping operation is performed to extract the rectangular area centered on each eye.

Head rotation data can provide a geometric orientation information that represents the user's three-dimensional attitude relative to the camera's coordinate system. Head rotation data can comprise a rotation matrix information that describes the spatial relationship between the user's head position and the imaging device reference frame. Head rotation data can be expressed as a flattened 9-dimensional vector derived from a 3x3 rotation matrix that captures head pose features.

The 3D eye center can provide spatial positioning coordinates that represent the three-dimensional position of each eye center within the camera's coordinate system. The 3D eye center can provide the center coordinates of the left and right eyes calculated by

geometric analysis of key points on the face combined with depth estimation techniques. In some embodiments, the 3D eye center can provide 6-dimensional coordinate data, which represents the spatial position of the two eye centers in three-dimensional space.

Calibration embeddings can provide subject-specific personalized parameters that enable personalized gaze estimation model adaptation. Calibration embeddings can comprise a 6-dimensional learnable embedding vector that represents individual user characteristics and gaze patterns without modifying the underlying network architecture. Calibration embeddings can be quickly personalized with selective parameter updates that improve the accuracy of gaze estimation for specific subjects.

## 2.2) Unified feature encoding.

Unified feature encoding converts heterogeneous input data from multimodal input processing into standardized dimensional representations suitable for subsequent graph-based processing. Unified feature encoding applies system transformation operations to map diverse input modalities into a common embedding space while preserving the unique characteristics of each input type. In specific implementation, the unified feature encoding comprises:

Depthwise separable convolution, which processes visual input from facial image input, left eye crop, and right eye crop. Depthwise separable convolutions enable the application of lightweight convolution operations that extract unique visual features while maintaining computational efficiency suitable for mobile deployment scenarios. Depthwise separable convolution reduces the number of parameters and computational requirements compared to standard convolutional operations while maintaining feature extraction effectiveness.

The squeeze-and-excitation module can enhance the quality of feature representations generated by depthwise separable convolutions through the channel attention mechanism. The squeeze-and-excitation module can apply global average pooling, followed by a fully connected layer and sigmoid activation to generate channel attention weights. In some embodiments, the squeeze-and-excitation module can improve feature discrimination by emphasizing the informational channel while suppressing the less relevant feature channels in the extracted visual representation.

The linear projection layer can convert enhanced visual features from the squeeze-and-excitation module into a unified dimensional space. The linear projection layer can apply a learned linear transformation to map 576-dimensional visual features extracted from facial and eye inputs into a standardized 32-dimensional embedding representation. The linear projection layer ensures that the visual features are dimensionally compatible with the geometry and calibration data for subsequent graph processing operations.

GELU activation can provide nonlinear transformation capabilities within the linear projection layer to enhance the quality of feature representation. GELU activation can apply Gaussian error linear unit activation functions that provide smooth nonlinearity and improved gradient flow characteristics compared to traditional activation functions. GELU activation enables efficient feature transformation while maintaining training stability and convergence characteristics.

### 2.3) Graph neural network.

Graph neural network models cross-modal relationships and dependencies between the unified feature representations generated by unified feature encoding. Graph neural networks can implement attention-based processing operations that capture spatial and semantic interactions between different input modes through system relational modeling. In implementation, graph neural network comprises:

A six-node graph structure that represents each input modality as a different node within a fully connected graph topology. The six-node graph structure can establish a comprehensive connection between all input modes, enabling each node to directly affect every other node within the graph framework. In some embodiments, the six-node graph structure can correspond to facial features, left eye features, right eye features, head rotation data, left eye 3D position, and right eye 3D position. All node pairs have learnable edge weights.

Multi-head attention can process six-node graph structures through parallel attention mechanisms that capture multiple types of relational patterns simultaneously. Multi-head attention can be calculated for attention coefficients for all node pairs using the learned linear projection, join operations, and activation functions. Multi-head attention can manipulate multiple parallel attention heads with independent parameter sets to achieve

comprehensive cross-modal relationship modeling.

Two-layer processing enables hierarchical feature refinement through dimensional expansion and reduction operations, which enhance representation capabilities while maintaining computational efficiency. Two-layer processing can extend node features to 256 dimensions in the first layer to capture complex relationship patterns, and can reduce extended features back to 32 dimensions in the second layer. In some embodiments, two-layer processing can apply a full attention mechanism in two processing layers to achieve system feature refinement across graph structures.

#### 2.4) Output generation.

Output generation converts the processed graph representation into accurate three-dimensional gaze vector predictions suitable for spatial applications. Output generation can integrate relational information captured through graph processing with complementary feature representations to generate comprehensive gaze predictions. In specific implemented, the output generation comprises:

Feature fusion module, which combines graph-level representations from graph neural networks with original CNN features, original geometry features, and subject-specific calibration embeddings. The feature fusion module can integrate diverse feature representations through fusion operations that ensure proper weighting and combination of heterogeneous input modalities. The feature fusion module can create a unified feature representation that maximizes the accuracy of gaze estimation under different subject and environmental conditions.

The regression network can process integrated features from the feature fusion module to generate angular gaze predictions. The regression network can apply a fully connected layer with a nonlinear activation function to map the fused multimodal features to the gaze parameters in spherical coordinate format. In some embodiments, the regression network can produce predicted pitch and yaw angles that represent the direction of the user's gaze relative to the camera coordinate system.

3D vector reconstruction can convert angular predictions from regression networks into normalized three-dimensional gaze vectors suitable for spatial applications and coordinate system integration. 3D vector reconstruction can use sine and cosine functions

to apply trigonometric operations to generate 3D coordinate components from predicted pitch and yaw angles. 3D vector reconstruction can work in tandem with the 3D gaze vector reconstruction module, orthogonal base building unit, and camera coordinate transform unit to produce parametric rays with defined origin and direction vectors that can be computed at the intersection of virtual objects, user interface elements, and physical surfaces in augmented and virtual reality environments.

The comprehensive system architecture for multimodal gaze estimation based on lightweight graph neural networks has a hierarchical structure that can realize the system transformation of heterogeneous input data through specialized processing stages, which capture comprehensive cross-modal relationships while maintaining computational efficiency. The system architecture achieves real-time performance with approximately 3.48 million parameters, making the system suitable for deployment in mobile devices, embedded systems, and low-power AR/VR platforms, while maintaining gaze estimation accuracy through integrated processing of visual appearance features, geometric orientation information, spatial positioning coordinates, and subject-specific calibration parameters.

### **Example 2:**

On the basis of the multimodal gaze estimation method proposed by embodiment 1 and the preferred embodiment, the present embodiment further comprises a step of initializing a six-dimensional calibration embedding vector for a specific subject, which enables personalized adaptation with a minimum of training samples. Referring to Figure 3, this specific body calibration method comprises the following steps:

3.1) Initializing the specific subject's 6-dimensional calibration embedding vector for target body.

Learnable parameters that represent individual user characteristics and gaze patterns are created without modifying the underlying network architecture.

3.2) Collecting a minimum of calibration samples from the target body.

Gaze data points are collected that provide subject-specific training information for personalized gaze estimation models. In some embodiments, as few as 32 calibration samples can be collected to achieve effective personalization while minimizing user

burden and calibration time requirements.

3.3) Determining whether sufficient calibration samples are available for effective model personalization.

After sample collection, the quantity and quality of collected samples can be assessed against predetermined thresholds that ensure sufficient training data for reliable calibration performance. This step provides a decision point to guide the calibration process to different paths based on sample adequacy assessments: when it is determined that available samples are insufficient, proceed to step 3.5); and when it is determined that sufficient samples are available, proceed to step 3.4).

3.4) Updating only the calibration embedding and final FC layer parameters.

The specific body calibration embedding and final fully connected layer weights within full connection processing network are modified, while all other network parameters remain frozen. In some embodiments, this step avoids retraining the complete network architecture, thereby reducing computational requirements and preventing overfitting on limited calibration data.

It should be noted that only the calibration embedding and final FC layer (Fully Connected Layer) parameters are updated, and this selective parameter update method reduces the calibration time to less than 2 minutes.

3.5) Requesting additional calibration samples from the target body.

A user is prompted to provide more gaze data points to meet the minimum requirements for effective personalization. When the user provides more gaze data points, return to step 3.2) to collect additional calibration samples, creating a feedback loop that ensures sufficient training data is obtained before the model adaptation.

3.6) Validating calibration performance on test samples.

The effectiveness of the personalized model was evaluated by testing the gaze estimation accuracy on validation data that was not used during calibration parameter updates. This step allows you to measure the performance improvements achieved through specific subject adaptation and can verify that the calibration process has successfully enhanced the accuracy of gaze estimation for the target user.

3.7) Deploying a personalized gaze estimation model for the target body.

Activate the calibration system with updated body-specific calibration embedding parameters for real-time gaze tracking applications. In some embodiments, this step enables the deployment of a personalized model that provides up to 11.8% error reduction compared to an uncalibrated baseline system.

The subject-specific calibration method proposed in the present embodiment realizes rapid personalization by updating only the 6-dimensional calibration embedding vector and final layer weights instead of retraining the entire network architecture, which can provide an efficient calibration workflow, minimize user interaction time, and maximize the benefits of personalization, so as to improve the gaze estimation accuracy under different subjects and usage scenarios.

### **Example 3:**

The present embodiment shows a specific implementation method for obtaining graph-level features based on the processing of a multi-head attention graph neural network with a two-layer structure. Referring to Figure 4, graph neural network processing operations transform multimodal input embeddings through attention-based inference mechanisms. Cross-modal relationships between heterogeneous input modalities are modeled through a two-layer graph neural network architecture with dimensional expansion and reduction operations. Specifically, graph neural network processing operations comprise the following steps:

4.1) Initializing the six graph nodes using projected feature embeddings.

Uniform 32-dimensional embeddings generated by previous projection operations are received, and each embedding can be assigned to a corresponding node within the fully connected graph structure. In some embodiments, this step can establish initial node states that represent facial features, left eye features, right eye features, head rotation data, left eye 3D position, and right eye 3D position within the graph neural network processing framework.

4.2) Calculating the attention coefficient of all node pairs using linear projection.

Each attention head applies a learned linear transformation to the source node and target node features, and can join transformed representations to create a combined feature vector for attention coefficient calculations. The generated attention coefficient

quantifies the strength of the relationship between each pair of nodes in the fully connected graph structure.

4.3) Applying LeakyReLU activation and softmax normalization to attention coefficients.

After the attention coefficient is calculated, the original attention coefficient is processed by the LeakyReLU activation function to introduce nonlinearity, and softmax normalization can subsequently be applied to all neighbor (adjacent) connections of each node. In some embodiments, this step ensures that the attention weights of each node are summed to one, while maintaining the gradient flow by activating the function during the training operation.

4.4) Aggregating neighboring features using weighted attention coefficients.

Based on the normalized attention coefficient generated in the above steps, the weighted sum of the features of neighboring nodes is calculated. This step allows you to combine features from all connected nodes based on attention-weighted importance, enabling each node to incorporate information from related modalities within the graph structure.

4.5) Determining whether the current processing corresponds to a first GNN layer.

This step provides a decision point to guide dimensional transformation operations to different paths based on the position of the layers within the two-layer graph neural network architecture. Using this step, the current processing stage can be evaluated to determine the appropriate dimensional scaling operation for the aggregation node features. When it is determined that the current processing corresponds to the first GNN layer, proceed to step 4.6; and when it is determined that the current processing does not correspond to the first GNN layer, proceed to step 4.7).

4.6) Expanding node features to 256 dimensions.

A linear transformation is applied to increase the dimensional representation of each node from the input 32-dimensional space to the extended 256-dimensional intermediate representation. In some embodiments, this step can achieve the capture of complex relational patterns by increasing the representation power during the first-layer processing operation.

#### 4.7) Reducing node features to 32 dimensions.

A linear transformation is applied to reduce the dimensional representation from the middle 256-dimensional space back to a standardized 32-dimensional output format. This step maintains computational efficiency while preserving the enhanced relationship information captured during the first layer expansion operation.

#### 4.8) Applying a second GNN layer processing.

Extended 256-dimensional node representation are processed through a second layer of the graph neural network architecture. This step can be used to repeat attention factor calculations, normalization, and feature aggregation operations using the expanded dimension representation as input features.

#### 4.9) Flattening and concatenating all node features into a unified representation.

Six 32-dimensional node embeddings are converted into a single flattening vector by connecting operations. In some embodiments, this step can generate a 192-dimensional unified representation that captures processing relational information from all input modalities within a single feature vector.

#### 4.10) Outputting a final graph-level feature vector for fusion operations.

A unified graph representation is provided to the downstream processing module for integration with raw CNN features, raw geometry features, and body-specific calibration embeddings. This step enables the multimodal feature integration layer to combine graph-based relational features with other feature representations for comprehensive gaze estimation processing.

It should be noted that the graph neural network processing operation method realizes hierarchical feature refinement through a two-layer processing structure, which expands the representation ability in the first layer and integrates information in the second layer. In some embodiments, dimensional expansion and reduction operations can capture complex cross-modal relationships while maintaining computational efficiency suitable for real-time deployment scenarios of mobile devices and embedded systems.

#### **Example 4:**

The present embodiment provides a specific implementation method for converting gaze estimation prediction into a normalized three-dimensional gaze vector based on

coordinate transformation. The coordinate transformation step involves constructing an orthogonal coordinate frame using a cross product and generating parametric rays emitted from the center of the 3D eye for intersecting calculations with augmented or virtual reality coordinate frames.

Referring to Fig. 5, the three-dimensional gaze vector reconstruction process is shown, which converts angular gaze prediction into a spatial orientation vector representation suitable for augmented reality and virtual reality applications. This method can realize the coordinate transformation and mathematical operation of the system, and convert the gaze direction parameters of the spherical surface into a parametric ray representation with defined origin and direction vectors. Specifically, the three-dimensional gaze vector reconstruction process comprises the following steps:

5.1) Receiving a predicted gaze pitch and a yaw angle from a regression module.

Angular predictions generated by the fully connected processing network are obtained that represent the direction of the gaze in spherical coordinate format. In some embodiments, this step receives pitch and yaw angle values calculated by the multimodal feature fusion and regression operations described in the processing phase described above.

5.2) Converting spherical coordinates to Cartesian coordinates using trigonometric transformations.

Sine and cosine functions are applied to the received pitch and yaw angles to generate three-dimensional coordinate components. In this step, the x, y, and z coordinate values representing the gaze vectors in Cartesian space can be calculated, and the angular representation can be converted into a format suitable for three-dimensional space calculations.

5.3) Normalizing a resulting 3D vector to unit length.

The amplitude of the Cartesian coordinate vector is calculated, and each coordinate component can be divided by the calculated amplitude to ensure the unit vector property. In some embodiments, this step can ensure that the length of the normalized gaze vector is exactly one unit, providing mathematical stability for subsequent coordinate frame construction operations.

5.4) Constructing an orthogonal normalized coordinate frame using a cross product.

An orthogonal basis vector is generated by cross product operations between a normalized gaze vector and a reference coordinate axis. This step allows the creation of a complete three-dimensional coordinate system that establishes the spatial orientation relationship of the gaze vector within the target coordinate frame.

5.5) Constructing a rotation matrix based on the orthogonal basis vector.

Generated orthogonal basis vectors are arranged into a rotation matrix format that implements coordinate system transformations. In some embodiments, this step can construct a 3x3 rotation matrix that provides a mathematical framework for transforming gaze vectors between different coordinate reference systems.

5.6) Determining whether camera coordinate transformation is required.

Target application requirements are evaluated to determine whether the gaze vector should be expressed in the camera coordinate space or remain in the original coordinate system. This step provides a decision point to guide the processing process to different paths based on the coordinate system requirements of the downstream application. When it is determined that a camera coordinate transformation is required, proceed to step 5.7); when it is determined that the camera coordinate transformation is not required, proceed to step 5.8).

5.7) Transforming the gaze vector into the camera coordinates using a rotation matrix.

The rotation matrix constructed in step 5.5) is applied to convert the normalized gaze vector from the original coordinate system to the coordinate space of camera alignment. This step enables integration with imaging systems and spatial tracking applications operating within the frame of the camera's coordinates.

5.8) Outputting the gaze vector in the original coordinate system.

A normalized gaze vector is provided without coordinate system transformation, maintaining the original spatial orientation relationship. In some embodiments, this step may be suitable for applications operating within the same coordinate framework as the initial gaze estimation processing.

5.9) Generating parametric rays with the 3D eye center as the origin.

After the coordinate transformation decision and processing in step 5.7) or step 5.8), a processed gaze direction vector is combined with the three-dimensional eye center coordinates to create a complete ray representation. This step defines the start point and direction vector of the gaze, enabling intersecting calculations with virtual objects, user interface elements, and physical surfaces in spatial applications.

It should be noted that the three-dimensional gaze vector reconstruction method realizes accurate three-dimensional gaze vector reconstruction by applying trigonometric transformation, normalization operation and coordinate system transformation. This method can work in tandem with the 3D gaze vector reconstruction module, orthogonal building unit, and camera coordinate transformation unit to provide mathematically stabilized gaze ray representations suitable for augmented reality, virtual reality, and spatial interaction applications. In some embodiments, the method can generate parametric rays that support gaze intersection calculations for user interface control, diagnostic applications, and immersive environment interaction systems.

**Example 5:**

The present embodiment provides a specific implementation method for projecting the visual feature, the geometric orientation information, and the spatial positioning coordinate into the same vector space based on a linear transformation with a GELU activation function.

Referring to FIG. 6, the method of the present embodiment provides a feature embedding component architecture that processes heterogeneous multimodal inputs for gaze estimation applications through four specialized processing modules. Disparate input modalities are converted into unified dimensional representations that enable coherent cross-modal inference and lightweight deployment on mobile devices and embedded systems. In specific implemented, the feature embedding components comprise:

6.1) A visual feature processing module, which processes appearance-based information from facial and eye area inputs. The visual feature processing module can process visual data through specialized extraction operations that capture unique features from facial images and eye cropping inputs while maintaining computational efficiency

suitable for real-time applications.

A facial feature extractor within the visual feature processing module can process images of facial areas to extract appearance-based features that capture overall facial structure, orientation, and contextual information. The facial feature extractor can apply the depthwise separable convolution and squeeze-and-excitation module to generate feature representations that capture facial appearance patterns related to gaze direction estimation. In some embodiments, the facial feature extractor can process facial image input through lightweight convolutional operations that reduce computational requirements while maintaining feature extraction effectiveness.

An eye feature extractor within the visual feature processing module can process individual eye area images to extract detailed appearance information from each eye. The eye feature extractor can handle both left and right eye cropping with specialized processing operations that capture eye-specific features, comprising pupil position, iris pattern, and eyelid configuration. Eye feature extractors can apply convolutional processing operations similar to facial feature extractors while focusing on isolating unique visual features present within the eye area.

The facial feature extractor and eye feature extractor can generate a 576D feature vector that captures visual features from processed facial and eye area inputs. The 576D feature vectors can represent appearance-based information extracted from a combination of facial images and eye cropping inputs through convolutional processing operations. In some embodiments, the 576D feature vector may contain unique visual features that provide texture, shape, and appearance information that complements the geometric and spatial input modalities.

A 32D projection layer in the visual feature processing module can convert the 576D feature vector into a standardized dimensional space for subsequent processing operations. The 32D projection layer can be applied with a linear projection layer with GELU activation to map high-dimensional visual features into a unified 32-dimensional embedded space. The 32D projection layer ensures that visual features are dimensionally compatible with geometry and calibration data for subsequent graph processing operations within graph neural networks.

6.2) Geometric feature processing module, which processes head orientation data through system transformation operation. The geometric feature processing module can process three-dimensional rotation matrix information that represents the spatial relationship between the user's head position and the camera's coordinate system.

A head rotation processor within the geometric feature processing module can receive head rotation data containing three-dimensional rotation matrix information describing the head pose features. The head rotation processor can process rotation matrix data to extract geometric orientation parameters that provide spatial context for gaze estimation. In some embodiments, the head rotation processor can process a rotation matrix that describes the angular relationship between the orientation of the user's head and the reference frame of the imaging device.

A 9D matrix flattening component within the geometric feature processing module converts the three-dimensional rotation matrix into a flattening vector representation suitable for subsequent processing operations. The 9D matrix flattening component can convert the 3x3 rotation matrix into a 9-dimensional vector format, which retains the geometric orientation information and realizes linear processing. The 9D matrix flattening component ensures that the head rotation information is represented in a format compatible with dimensional normalization operations applied to other input modalities.

A 32D embedded space component within the geometric feature processing module can project flattened geometry data into a unified dimensional framework used across all input modalities. The 32D embedded space component can apply a linear transformation operation to map the 9-dimensional head rotation vector into a standardized 32-dimensional space. In some embodiments, the 32D embedding spatial component can ensure that the geometric orientation data shares the same dimensional features as the visual feature and spatial localization information for coherent cross-modal processing within graph neural networks.

6.3) Spatial feature processing module, which manages three-dimensional eye position data through specialized coding operations. The spatial feature processing module can process spatial coordinates that represent the three-dimensional position of each eye center in the camera coordinate system.

A left eye center component within the spatial feature processing module can process the three-dimensional coordinate data of the left eye center position. The left eye center component can process spatial positioning information calculated through geometric analysis of facial key points combined with depth estimation techniques. The center of the left eye component manages coordinate data that represents the spatial position of the center of the left eye within the coordinate frame of the three-dimensional camera.

A right eye center component within the spatial feature processing module can process the three-dimensional coordinate data of the right eye center position. The right eye center component can process spatial positioning information that complements the left eye center component to provide comprehensive eye positioning data. In some embodiments, the right eye center component can work in tandem with the left eye center module to provide 6-dimensional coordinate data representing the positions of the two eye centers within the 3D eye center input modality.

A 3D position coding component within the spatial feature processing module can process the center of the eye coordinates from the center of the left eye component and the center of the right eye component. The 3D position coding component can apply encoding operations to convert 3D spatial coordinates into representations suitable for dimensional standardization. The 3D position encoding component ensures that the spatial positioning information is properly formatted in a format suitable for subsequent projection into a unified embedded space.

A 32D unified spatial component within the spatial feature processing module can convert the encoded spatial information into a common embedding dimension used across all input modalities. The 32D unified space component can apply a linear transformation operation to map the 6-dimensional eye center coordinate data into a standardized 32-dimensional space. In some embodiments, the 32D unified spatial component can ensure dimensional compatibility of spatial positioning information with visual feature and geometric orientation data for effective integration within a six-node graph structure.

6.4) Calibration component module, which provides personalized capabilities through specific body adaptation mechanisms. The calibration component module enables personalized gaze estimation model adaptation without the need for full network

retraining.

A body-specific embedding components within the calibration components module can generate personalized parameters for each user that capture the gaze patterns and characteristics of a particular subject. Body-specific embedding components can create learnable embedding vectors that represent individual user characteristics without modifying the underlying network architecture. The body-specific embedding components can be quickly personalized with selective parameter updates that improve the accuracy of gaze estimation for specific body.

A 6D parametric vector component within the Calibration Component module can represent calibration data in a standardized format suitable for integration with other input modalities. The 6D parameter vector component can contain learnable parameters optimized for the target body through a calibration process. In some embodiments, the 6D parametric vector component can provide a calibration embedding that enables a specific body adaptation while maintaining compatibility with the unified dimensional framework used throughout the system.

An individual module components within the calibration component module enable rapid adaptation to specific body without the need for full system retraining. The personalization module component enables the selective parameter update method, which modifies only the specific body calibration embeddings and final layer weights while keeping all other network parameters frozen. The personalization module component can provide a calibration workflow that minimizes user interaction time while maximizing the benefits of personalization to improve the accuracy of gaze estimation across subjects and use cases.

The above four processing modules can work together to convert heterogeneous input modalities into unified dimensional representations that enable effective cross-modal inference within graph neural networks. The visual feature processing module can process appearance-based information through the face feature extractor and eye feature extractor, while the geometric feature processing module can process head orientation data through the head rotation processor and dimension transformation operations. The spatial feature processing module can manage eye positioning coordinates through the

left and right eye center components, while the calibration component module can provide personalization capabilities through body-specific embedding and personalization module components. In some embodiments, the implementation of system dimension standardization across all four modules ensures that diverse input modalities can be effectively integrated within a six-node graph structure for comprehensive cross-modal relationship modeling and accurate gaze prediction.

**Example 6:**

In a second aspect, the present application further provides a multimodal gaze estimation system based on a lightweight graph neural network, the system comprising:

A multimodal input module configured to obtain multimodal heterogeneous data related to and complementary for a gaze direction estimation, wherein the multimodal heterogeneous data comprises visual appearance data, geometric orientation information, and spatial positioning coordinates;

A feature extraction module configured to extract the visual feature from the visual appearance data based on depthwise separable convolution and a squeeze-and-excitation module;

A projection module configured to project the visual feature, the geometric orientation information, and the spatial positioning coordinate into a common vector space;

A graph construction module configured to construct a graph structure, wherein the graph structure comprises a plurality of nodes representing the multimodal heterogeneous data or features corresponding to the multimodal heterogeneous data;

A multi-head attention GNN configured to process the constructed graph based on a multi-head attention graph neural network having a two-layer structure to obtain graph-level features, wherein a first layer of the two-layer structure expands dimensionality of node features, and a second layer of the two-layer structure reduces dimensionality of the expanded dimension.

A feature fusion module configured to aggregate the graph-level feature, and fuse them with the visual feature, the geometric orientation information, and the spatial positioning coordinates;

A regression module configured to use a fully connected layer to perform gaze estimation prediction based on the fusion features.

A 3D reconstruction module configured to convert the gaze estimation prediction into a normalized three-dimensional gaze vector based on coordinate conversion.

Referring to Figure 7, the system can provide a multimodal gaze estimation architecture based on a lightweight graph neural network, which processes heterogeneous inputs to predict the direction of the gaze. The system comprises:

A multimodal input module, which receives various types of input data, comprising a facial image, an eye crop image, a head rotation data, and a 3D eye center position. The multimodal input module can capture image data through a device camera and process the captured data through a normalized camera module that performs data normalization operations.

A feature extraction module, which receives normalized facial and eye image data from the multimodal input module. The feature extraction module can generate facial features and eye features from processed image inputs using depthwise separable convolution and squeeze-and-excitation module. In some embodiments, the feature extraction module can process the 3D eye center data obtained from the captured image as well as the head rotation matrix information representing the head pose data.

A projection module can receive input from multiple sources within the system, comprising facial features and eye features from the feature extraction module. The projection module can convert heterogeneous input modes into a unified representation through embedding operations. The projection module can generate calibration embeddings represented as vectors, and can create facial feature embeddings, 3D eye center embeddings, head rotation embeddings, and eye feature embeddings to map different input types into a common dimensional space.

A graph construction module, which receives embedded output from the projection module. The graph construction module can construct a fully connected graph structure with six interconnected nodes, representing different input modes. In some embodiments, the six nodes can correspond to a right eye, a left eye, a face, a right eye center, a left eye center, and a head rotation data. The graph construction module can establish directed

edges between nodes to indicate relationships between modalities.

A multi-head attention GNNs can process graph structures created by the graph construction module. The multi-head attention GNNs can include a multi-head attention processing mechanism, followed by attention aggregation operations. The multi-head attention GNNs can model cross-modal relationships and dependencies through attention-based reasoning, which captures spatial and semantic interactions between facial components, head orientation, and eye position.

A feature fusion module, which combines outputs from multiple processing paths. The feature fusion module can receive calibration embeddings, 3D eye center data, and context-aware encoding from multi-head attention GNNs from the projection module. The feature fusion module can integrate diverse inputs into a unified representation that combines relational information from graph processing with the original feature representation.

A regression module can process the fusion features from the feature fusion module through the fully connected output layer. The regression module can generate normalized gaze direction predictions expressed in yaw and pitch angles. In some embodiments, the regression module can generate an angle prediction that represents the direction of the gaze in a format suitable for subsequent three-dimensional transformations.

A 3D reconstruction module, which receives angle prediction from the regression module. The 3D reconstruction module can use the Spherical to Cartesian transform operation to convert the normalized gaze direction angle into a three-dimensional gaze ray representation. The 3D reconstruction module can generate gaze vectors suitable for space applications and coordinate system integration.

The system has a total parameter count of approximately 3.48 million parameters, making the system 50-70% smaller than competing transformers and CNN models. In some embodiments, the reduced parameter count enables real-time performance in mobile devices, embedded systems, and low-power AR/VR platforms while maintaining gaze estimation accuracy. The system can work with RGB webcams, mobile front cameras, and world cameras on headsets without the need for specialized IR hardware, enabling deployment in a wide range of consumer devices and unconstrained

environments.

Referring to Figure 8, a camera device can initiate the gaze estimation process by capturing image data from the user. The camera device can transmit the captured image data to a normalized camera interface, which standardizes the input data for subsequent processing operations. The normalized camera interfaces can perform normalization operations to ensure consistent data formats across different camera types and imaging conditions.

The normalized camera interface can forward the processed image data to a data processing module, which performs comprehensive preprocessing operations. The data processing module can perform facial key point detection, eye position recognition, resizing, cropping, and normalization operations on the received image data. In some embodiments, the data processing module can generate multiple output streams branching into parallel processing paths of different input modalities.

The data processing module can generate a head rotation vector that represents three-dimensional head orientation information. The data processing module can generate facial images that capture the overall facial appearance features and can create left-eye crop and right-eye crop images that isolate the eye area for detailed analysis. The data processing module can provide 3D position data for the left eye and 3D position of the right eye, representing the spatial coordinates of each eye center in three-dimensional space.

A feature extractor can receive facial images, left eye crop, and right eye crop from the data processing module. The feature extractor can extract 576-dimensional features from face and eye crops using depthwise separable convolution combined with a squeeze-and-excitation module to process vision inputs. The feature extractor can apply global averaging pooling operations to generate facial and eye features that capture unique visual features from the input image.

The feature extractor can project the 576-dimensional visual feature into a unified embedding space using linear transformations with GELU activation. In some embodiments, the feature extractor can convert heterogeneous visual data into standardized dimensional representations, enabling coherent cross-modal processing at subsequent stages.

A multi-stream linear module can receive output from the data processing module and the feature extractor. The multi-stream linear module can process head rotation vector, facial features, eye features, left eye 3D position, and right eye 3D position data to produce uniform embeddings. The multi-stream linear module can generate head rotation embeddings, facial feature embeddings, eye feature embeddings, and 3D eye center embeddings, which represent various input modalities in a common dimensional space.

The multi-stream linear modules can enable coherent cross-modal inference by mapping heterogeneous input types into standardized embedding representations. In some embodiments, the multi-flow linear module can facilitate subsequent graph-based processing for attention-based relational modeling by ensuring that all input modalities share compatible dimensional features.

Referring to Figure 9, a multimodal input module can provide node embeddings, including face, left and right eyes, head rotation, and eye position data, to initiate graph neural network processing. The multimodal input module can provide heterogeneous embedding representations that are converted to a unified dimensional space through previous feature extraction and projection operations.

A graph construction module can receive node embeddings from multimodal input modules and can create fully connected network structures with six nodes. The graph construction module can establish static topology where all nodes are connected to all other nodes with learnable edge weights. In some embodiments, the graph construction module can implement comprehensive relationship modeling between different input modes by ensuring that each node can directly influence every other node in the network.

The graph construction module can forward the constructed graph structure to the multi-head attention mechanism for attention-based processing. The multi-head attention mechanism can perform linear transformations of source and target features, and can calculate the attention coefficient using the learned linear projection. The multi-head attention mechanism can apply the connection of transform features, followed by LeakyReLU activation, to generate attention coefficients with learnable parameters.

The multi-head attention mechanism can use parallel attention heads to capture multiple relationship patterns between graph nodes simultaneously. In some embodiments,

the multi-head attention mechanism enables the system to model different types of relationships and dependencies in the same processing step, allowing for comprehensive cross-modal interaction analysis.

An attention normalization module can receive the calculated attention coefficient from the multi-head attention mechanism and can apply the coefficient normalization on a per-node basis. The attention normalization module can perform exponential scaling of scores and can perform softmax normalization between neighbors to ensure proper weighting of node influences. The attention normalization module generates a normalized attention coefficient that represents the relative importance of each node connection.

A feature aggregation module can process the normalized attention coefficients from the attention normalization module to perform node feature processing. The feature aggregation module calculates the weighted sum of neighbor features and can apply nonlinear transformations while maintaining the original node dimension. In some embodiments, the feature aggregation module can maintain the spatial and semantic relationships between input modalities through a combination of features based on attention weighting.

The feature aggregation module can provide aggregated node features to the multi-head processing module to generate an enhanced representation.

A multi-head processing module can manipulate multiple parallel attention heads with independent parameter sets and can aggregate the results by connecting them to form rich node representations. The multi-head processing module can realize the simultaneous processing of different relationships in the graph structure.

A two-layer inference module can receive rich node representations from the multi-head processing module for hierarchical processing operations. The two-layer inference module can convert node features into intermediate representations in the first layer and can extend node features to 256 dimensions in this initial processing phase. The two-layer inference module can then convert the intermediate representation into output embeddings in the second layer, and can reduce the extended features back to 32 dimensions to maintain computational efficiency.

The two-layer inference module can apply a full attention mechanism in both

processing layers to achieve hierarchical feature refinement across graph structures. In some embodiments, dimensional expansion and reduction operations can allow a two-layer inference module to capture complex relational patterns while maintaining manageable computational requirements for mobile deployment scenarios.

A graph-level generation module can receive processed embeddings from the two-layer inference module for final feature vector creation. The graph-level generation module can embed all six updated nodes flattened and connected into a unified graph representation vector. The graph-level generation module can generate a uniform 384-dimensional relationship vector that captures cross-modal dependencies and relationships between all input modalities.

The graph-level generation module can output the final graph-level feature vector for subsequent fusion with other feature representations in the downstream processing module. In some embodiments, the unified relational vector can provide comprehensive cross-modal relational information, enhancing the accuracy of gaze estimation compared to independent processing methods used in conventional systems.

Referring to Figure 10, a multimodal feature integration layer can receive multiple input streams to create a composite representation for gaze estimation processing. The multimodal feature integration layer can combine diverse input modalities from four different sources that provide different types of feature information for comprehensive gaze prediction.

A graph neural network outputs can provide relational features to the multimodal feature integration layer. The graph neural network outputs can contain graph-based relational information that captures cross-modal dependencies and spatial relationships between facial components, eye position, and head orientation data processed through attention mechanisms.

Raw CNN features can provide appearance-based features from the face and eye processing branches to the multimodal feature integration layer. The raw CNN features can include visual features extracted from facial areas and eye crops that provide texture, shape, and appearance information that complements the relational features of graph processing.

Raw geometry features can transmit rotation and eye position data to the multimodal feature integration layer. The raw geometry features can include head rotation matrix information and three-dimensional eye center coordinates, which provide spatial orientation and positioning context for gaze direction estimation.

Subject-specific calibration embeddings can provide personalized calibration parameters to the multimodal feature integration layer. The subject-specific calibration embeddings can include a 6-dimensional learnable embedding vector that adapts and personalizes specific subjects without the need for full network retraining.

The multimodal feature integration layer can combine the graph neural network outputs, the raw CNN features, the raw geometric features, and the subject-specific calibration embeddings into a unified feature representation. In some embodiments, the multimodal feature integration layer can ensure that the heterogeneous input modalities are appropriately weighted and integrated to maximize the accuracy of gaze estimation under different subject and environmental conditions.

A fully connected processing network can receive integrated features from the multimodal feature integration layer for angle prediction processing. The fully connected processing networks can perform nonlinear transformations on composite feature representations through fully connected layers that map fusion features to gaze parameters. The fully connected processing network can generate predicted pitch and yaw angles that represent the direction of gaze in spherical coordinate format.

The fully connected processing network can output the angle prediction to the 3D gaze vector reconstruction module for three-dimensional vector transformation. The 3D gaze vector reconstruction module can use the Spherical to Cartesian transform operation to convert the predicted pitch and yaw angles into normalized 3D vectors. In some embodiments, the 3D gaze vector reconstruction module can apply trigonometric functions to convert angular representations into Cartesian coordinate vectors suitable for spatial applications.

A 3D reconstruction module can work in tandem with the 3D gaze vector reconstruction module to perform comprehensive 3D gaze vector processing. The 3D reconstruction module can convert pitch and yaw angles into normalized 3D vectors using

a spherical to Cartesian transform, which ensures mathematical stability and unit vector properties.

An orthogonal basis construction unit can receive a normalized gaze vector from the 3D gaze vector reconstruction module to establish a coordinate frame. The orthonormal basis construction unit can be constructed using the fork product and vector normalization operations to construct orthogonal coordinate frameworks. The orthonormal basis construction unit can perform cross-product calculations to derive orthogonal axes, and can normalize and verify orthogonality to establish a stable coordinate reference system.

A camera coordinate transform unit can process the orthogonal base vector from the orthogonal base building block for final coordinate system alignment. The camera coordinate transform unit can transform the normalized gaze vector into the camera coordinates using a rotation matrix built from orthogonal base vectors. The camera coordinate transformation unit can build a rotation matrix from the base vector and can invert the matrix to produce a camera alignment output suitable for integration with imaging systems and spatial tracking applications.

The 3D gaze vector reconstruction module outputs parametric rays emitted from the center of the 3D eye, enabling intersecting calculations with screens, surfaces, and AR/VR coordinate frames. In some embodiments, parametric ray representations can provide origin and orientation information, allowing downstream applications to compute the intersection point of gaze with virtual objects, user interface elements, and physical surfaces in augmented and virtual reality environments.

Referring to Figure 11, the system can coordinate a series of comprehensive processing operations to convert the multimodal input data into an accurate three-dimensional gaze vector representation through the system's module interaction. The system can orchestrate data flows that specialize in processing data flows between components that handle feature extraction, dimension normalization, graph-based relational modeling, and vector reconstruction operations in a sequential manner.

A multimodal input module can initiate a processing sequence by providing multimodal inputs to the system, including facial images, eye cropping, head rotation data, and 3D eye center position. In some embodiments, the basis for subsequent processing

operations is established by providing heterogeneous input data that captures visual appearance features, the geometric orientation information, and the spatial positioning coordinates. In some embodiments, ensuring that all necessary input modalities are available for comprehensive gaze estimation processing.

The system forwards the received image input to a feature extraction module for visual feature processing.

The facial image and eye crop data are directed to specialized processing operations that extract unique visual features from the input image. The feature extraction module processes the image input using the depthwise separable convolution and squeeze-and-excitation module and can return the processed features to the system.

The extracted visual features capture appearance-based information from facial areas and eye crops. In some embodiments, when the processed features are returned to the system, 576-dimensional feature vectors can be generated, which represent unique visual features suitable for subsequent dimensional transformation operations. The feature extraction module can complete the vision processing operations and can transfer the extracted features back to the system for integration with other processing paths.

The system then forwards all heterogeneous inputs to the projection module for dimension normalization.

The visual features, the geometric data, and calibration parameters are directed to unified embedding operations that convert different input modalities into a common dimensional space. The projection module can project all inputs into a common 32D embedding space, and can be uniformly embedded and returned to the system.

When performing this step, the projection module can provide standardized representations of dimensions that enable coherent cross-modal processing in subsequent graph-based operations. The projection module can apply linear transformations with GELU activation functions to map heterogeneous input types into a unified 32-dimensional space. In some embodiments, it is possible to ensure that the visual feature, head rotation matrix, and 3D eye center coordinates are represented in a compatible dimensional format for graph neural network processing.

The system sends the projection embeddings to the graph building block for graph

structure creation.

The uniform embeddings are directed to graph construction operations that establish relational connections between different input modalities. The graph construction module can build a fully connected graph with six nodes, representing the input modalities, and can return the graph structure to the system.

When performing related operations, the graph construction module can provide a fully connected graph structure, where each node corresponds to an input mode, including facial features, left eye features, right eye features, head rotation data, left eye 3D position, and right eye 3D position. In some embodiments, learnable edge weights can be established between all node pairs to realize comprehensive relationship modeling between different input modes within the framework of graph neural networks.

The system forwards the constructed graph to the multi-head attention GNN for relational processing.

The graph structures are directed to attention-based processing operations that model cross-modal relationships and dependencies through parallel attention mechanisms. The multi-head attention GNN can process the graph through a two-layer structure with an attention mechanism and return the graph-level feature to the system.

When performing related operations, the multi-head attention GNNs can provide processing relational features that capture cross-modal dependencies and spatial relationships between different input modes. The multi-head attention GNN can apply attention factor calculation, normalization, and feature aggregation operations in two processing layers. In some embodiments, the process can generate a unified 384-dimensional relational vector that represents comprehensive relational information captured through graph-based attention processing.

The system sends graph features, CNN features, and geometric inputs to the feature fusion module for integration.

Multiple feature representations are directed to fusion operations that combine relational information from graph processing with appearance-based feature and geometry data. The feature fusion module can aggregate and fuse all feature representations and return the unified representation to the system.

When the feature fusion module performs related operations, it can provide an integrated and feature representation that combines a graph neural network outputs, raw CNN features, raw geometric features, and specific subject calibration embeddings into a comprehensive multimodal representation. In some embodiments, it is possible to ensure that different input modes are properly weighted and integrated to maximize gaze estimation accuracy under different subject and environmental conditions.

The system forwards the fusion features to a regression module for angle prediction.

The integrated feature representation is directed to regression operations that map the fusion features to the gaze direction parameters. The regression module can use the fully connected layer to predict the pitch and yaw angles of the gaze, and can return the angle prediction to the system.

When the regression module performs related operations, it can provide predicted pitch and yaw angles, which represent the direction of the gaze in spherical coordinate format, which is suitable for subsequent 3D transformation operations. The regression module can apply nonlinear transformation to predict from the perspective of fusion multimodal feature generation through the fully connected processing network. In some embodiments, gaze direction parameters can be generated that are used as inputs for the three-dimensional vector reconstruction process.

The system sends the predicted angle to a 3D reconstruction module for vector transformation.

The angular predictions are directed to 3D reconstruction operations that convert spherical coordinate representations into Cartesian vector format. The 3D Reconstruction module can use the spherical to Cartesian transform to convert the angle into a normalized 3D gaze vector and can return the final gaze vector to the system.

When performing the relevant operations, the 3D reconstruction module can provide normalized 3D gaze vectors that represent the direction of the gaze in Cartesian coordinate space, making it suitable for spatial applications and coordinate system integration. The 3D reconstruction module can work in tandem with the 3D gaze vector reconstruction module, the orthonormal basis construction unit, and the camera coordinate transform unit to generate parametric rays with defined origin and direction

vectors. In some embodiments, gaze ray representations are generated that enable intersecting computations with virtual objects, user interface elements, and physical surfaces in augmented and virtual reality environments.

Referring to Figure 12, a camera device captures image data from the user and transmits the captured image data to a normalized camera interface to initiate a multimodal gaze estimation data processing workflow.

An initial data acquisition phase of the processing workflow is established by providing raw image inputs containing facial features, eye area, and head orientation information. The camera devices can work with a variety of imaging hardware, including RGB webcams, mobile front cameras, and world cameras on headsets, without the need for specialized infrared lighting systems.

The normalized camera interfaces can receive captured image data from the camera device and can perform standardized operations to ensure consistent data formats across different camera types and imaging conditions. A normalization program is applied that takes into account changes in camera resolution, color space representation, and geometric distortion features. In some embodiments, image data can be prepared for subsequent processing operations by establishing uniform data attributes that enable reliable feature extraction and analysis in different imaging hardware configurations.

The normalized camera interface can also forward standardized image data to a data processing module for comprehensive preprocessing operations. Normalized image inputs are directed to specialized processing operations that extract multiple types of information from captured images. The data processing module can perform facial keypoint detection, eye position recognition, resizing, cropping, and normalization operations on the received image data to generate multiple output streams that provide different aspects of the visual and geometric information contained in the original image.

The data processing module can generate a head rotation vector representing the three-dimensional head direction information, and can output the head rotation vector to the multi-stream linear module. In this process, geometric orientation data can also be provided, which captures the spatial relationship between the user's head position and the camera's coordinate system. The head rotation vector can be expressed as a flattened 9-

dimensional vector derived from a 3x3 rotation matrix that describes the head pose relative to the imaging device reference frame.

After the head rotation vector is generated, the data processing module can produce facial images that capture the overall facial appearance features and can transfer the facial images to the feature extractor.

Cropped facial area data is provided that isolates facial areas from the original captured image while maintaining sufficient contextual information for appearance-based feature extraction. In some embodiments, a cropping of a face image can be generated that is adjusted to a standardized dimension suitable for subsequent convolutional neural network processing operations.

The data processing module can create left-eye crop and right-eye crop images that isolate a single eye area for detailed analysis, and can send eye crop images to the feature extractor.

Specialized eye area data is provided that can be used to estimate the most informative areas from the direction of the gaze for centralized feature extraction. Left-eye cropping and right-eye cropping images can be generated by pinpointing eye key points and then cropping, extracting rectangular areas centered on each eye while maintaining spatial relationships and scale factors.

The data processing module can also provide the spatial coordinate data of the left eye 3D position and the right eye 3D position to the multi-stream linear module.

Three-dimensional eye center coordinates are provided that represent the spatial position of each eye within the camera's coordinate system. 3D eye position data can be calculated by geometric analysis of facial key points combined with depth estimation techniques, which provide spatial positioning information that complements appearance-based features extracted from eye cropped images.

The feature extractor can receive facial images, left eye cropping, and right eye cropping from the data processing module, and can process visual inputs using depthwise convolution, squeeze-and-excitation module, and global average pooling operations.

Lightweight convolutional operations are applied that extract unique visual features while maintaining computational efficiency suitable for mobile deployment scenarios.

Depthwise separable convolution reduces the number of parameters and computational requirements compared to standard convolutional operations, while the squeeze-and-excitation module can enhance the quality of feature representation through a channel attention mechanism.

The feature extractor can generate facial and eye features that capture unique visual features from the processed image, and can transfer the extracted features to the multi-stream linear module.

576-dimensional feature vectors are provided that represent appearance-based information extracted from facial and eye area inputs. In some embodiments, feature representations are provided that capture texture, shape, and appearance features that are complementary to the geometric information provided through head rotation vectors and 3D eye position data.

The multi-stream linear module can receive all outputs from the data processing workflow, including head rotation vectors, facial features, eye features, and 3D eye position, for unified processing.

The different input modes generated by the aforementioned processing operations are integrated into a single processing module, which handles dimension normalization and embedding generation. The multi-stream linear module can handle heterogeneous input types through parallel processing paths that include geometric orientation data, appearance-based visual features, and spatial positioning coordinates, while maintaining the unique characteristics of each input modality.

The multi-stream linear module generates head rotation embedding, facial feature embedding, eye feature embedding, and 3D eye center embedding in a unified dimensional space.

Standardized 32-dimensional embedding representations are generated for all input modalities through linear transformation operations that map heterogeneous input data into a common dimension framework. A separate linear projection layer with GELU activation function is used to convert the 9-dimensional head rotation vector, 576-dimensional visual feature and 6-dimensional eye position coordinates into a unified 32-dimensional embedding space, which realizes coherent cross-modal inference in

subsequent graph neural network processing operations.

Referring to Figure 13, a multimodal input module provides node embeddings (including face, left and right eyes, head rotation, and eye position data) to the graph building block to initiate the graph neural network processing workflow.

A heterogeneous embedding representation converted to a unified 32-dimensional space through previous feature extraction and projection operations is provided. The multimodal input module ensures that all six input modalities are properly formatted and dimensionally normalized before the graph construction operation begins.

The graph construction module can receive node embeddings from the multimodal input module and create a fully connected network of six nodes with a static topology, wherein all nodes are connected to all other nodes by learnable edge weights.

A comprehensive connection is established between all input modes, allowing each node to directly affect every other node within the graph structure. The graph construction module can initialize learnable parameters for each edge connection that will be optimized during training to capture meaningful relationships between different input modalities.

After graph construction, the graph construction module can send the constructed graph structure to the multi-head attention mechanism for processing. Step S906 can guide the fully connected graph to attention-based processing operations that calculate the strength of the relationship between all node pairs. The multi-head attention mechanism can receive graph structures with initialized node embeddings and edge weights for subsequent attention coefficient calculations.

The multi-head attention mechanism can perform linear transformations of source and target features, and can calculate attention coefficients through learnable parameters using parallel attention heads.

Linear projections of learning are applied to the features of the source and target nodes, and representations of transformations can be joined to create combined feature vectors for attention coefficient calculations. In some embodiments, multiple parallel attention headers are operated simultaneously to capture patterns of different types of relationships between graph nodes.

The multi-head attention mechanism can forward the calculated attention coefficient to the attention normalization module for coefficient normalization.

The raw attention coefficient is directed to normalization operations that ensure proper weighting of node influences within the graph structure. The Attention Normalization module can receive attention coefficients from all parallel attention headers for subsequent exponential scaling and softmax normalization operations.

The attention normalization module can apply exponential scaling of fractions and can perform softmax normalization on a per-node basis for neighbor nodes. The original attention coefficient is processed by an exponential function, and then a softmax operation is performed to normalize the attention weight to the sum of each node to one. The attention normalization module ensures that each node's attention distribution on its neighbor nodes forms an efficient probability distribution while maintaining a gradient flow for the training operation.

After normalization, the attention normalization module can send the normalized attention coefficient to the feature aggregation module for node feature processing. Provide appropriately weighted attention coefficients that quantify the relative importance of each node connection within the graph structure.

The feature aggregation module can receive normalized attention weights as well as original node features for weighted feature combination operations.

The feature aggregation module calculates the weighted sum of neighbor features and can apply nonlinear transformations while maintaining the original node dimension. Features from all connected nodes are combined based on attention-weighted importance, enabling each node to incorporate information from relevant modalities within the graph structure. In some embodiments, the 32-dimensional representation of each node is maintained, and the feature content is enriched by the attention-weighted aggregation of neighbor information.

The feature aggregation module can provide aggregated node features to the multi-head processing module to generate enhanced representations.

Attention aggregation node features that contain rich relational information from the first stage of graph processing are provided. The multi-head processing module can

receive updated node representations for parallel multi-head attention processing, which captures additional relational aspects within the graph structure.

The multi-head processing module can manipulate multiple parallel attention heads with independent parameter sets and can aggregate results through connections. Aggregate node features are processed by using separate attention mechanisms operated by different learning parameters, which makes it possible to capture different relationship patterns simultaneously. The multi-head processing module can connect outputs from all attention heads to form rich node representations that contain multiple perspectives across modal relationships.

The multi-head processing module can send rich node representations to the two-layer inference module for hierarchical processing. Enhanced node features processed through multiple attention mechanisms are provided to capture synthetic relationship information. The two-layer inference module can receive rich representations for system dimension transformations through a hierarchical two-layer architecture.

The two-layer inference module can convert node features into intermediate representations at the first layer and output embeddings at the second layer. This operation enables dimensional expansion and reduction operations, which enable hierarchical feature refinement in the graph structure. In the first layer of processing, the two-layer inference module can extend the node features to 256 dimensions to capture complex relational patterns, while in the second layer of processing, the block can reduce the extended features back to 32 dimensions to maintain computational efficiency for subsequent processing operations.

After the hierarchical processing, the two-layer inference module can forward the processed embeddings to the graph-level generation module for final feature vector creation. Fine-grained point embeddings with two-layer processing through a hierarchical inference architecture are provided. The graph-level generation module can receive the final processed node representations and integrate them into a unified graph-level feature vector.

The graph-level generation module can embed all six updated nodes flattened and connected into a unified graph representation vector.

A single 32-dimensional node embedding is converted into a single flattening vector through join operations that hold processing relationship information from all input modalities. The graph-level generation module can generate a unified representation that captures the comprehensive relational information processed by the graph neural network architecture and provides an integrated feature vector suitable for integration with other feature representations in the downstream processing module.

The workflow shown in Figure 13 enables system transformations of heterogeneous input embeddings through specialized graph processing operations that model across modal relationships and dependencies. The above sequential processing steps ensure that the relationship information between different input modes is captured and refined through attention mechanisms, dimensional transformations, and hierarchical processing operations that enhance the representation quality of the final graph-level feature vectors.

Referring to Fig. 14, a multimodal feature integration layer can coordinate a series of comprehensive processing operations that integrate heterogeneous input modalities and convert integrated features into accurate three-dimensional gaze vector representations. The multimodal feature integration layer can receive multiple input streams from specialized processing components that provide different types of feature information for comprehensive gaze prediction and spatial vector reconstruction.

A graph neural network outputs can provide relational features to the multimodal feature integration layer.

Graph-based relational information is provided that captures cross-modal dependencies and spatial relationships between facial components, eye position, and head orientation data, which have been processed by attention mechanisms within graph neural network architectures. The graph neural network output can contain a unified 384-dimensional relationship vector that represents the comprehensive relationship information captured by the two-layer graph processing operation.

Raw CNN features can transmit appearance-based features from facial and eye branches to the multimodal feature integration layer. Step S922 can provide visual features extracted from facial areas and eye crops that provide texture, shape, and appearance information that complements the relational features of graph processing. In

some embodiments, step S922 can provide 576-dimensional feature vectors that capture unique visual features from facial images, left eye crop, and right eye crop inputs processed by the depthwise separable convolution and squeeze-and-excitation module.

Raw geometry features can send rotation and eye position data to the multimodal feature integration layer.

Head rotation matrix information and 3D eye center coordinates are provided, which provide spatial orientation and positioning context for gaze direction estimation. The raw geometry features can include a 9-dimensional flattened head rotation vector and 6-dimensional eye center position data, which represent the geometric relationship between the user's head pose and eye position within the camera coordinate system.

Subject-specific calibration embeddings can provide personalized calibration parameters to the multimodal feature integration layer.

A 6-dimensional learnable embedding vector is provided, which realizes subject-specific adaptation and personalization without the need for full network retraining operations. In some embodiments, personalized parameters optimized for the target subject by calibration process are provided.

The multimodal feature integration layer can combine diverse input modalities from graph neural network outputs, raw CNN features, raw geometric features, and subject-specific calibration embeddings into composite representations. Relational features, appearance-based features, geometric data, and personalized parameters are integrated through fusion operations, ensuring proper weighting and combination of heterogeneous input modalities. Multimodal feature integration layers can create a unified feature representation that maximizes gaze estimation accuracy under different subject and environmental conditions.

After feature integration, the multimodal feature integration layer can forward the composite representation to the fully connected processing network in step S930. Step S930 can guide the integrated features to regression processing operations that convert the fused multimodal representations into angular gaze direction parameters. The fully connected processing network can receive a synthetic feature representation that combines all input modalities for subsequent nonlinear transformation operations.

The fully connected processing network can perform nonlinear transformations on the composite feature representation and can generate predicted pitch and yaw angles in step S932. Step S932 can apply a fully connected layer with a nonlinear activation function to map the integrated features to the gaze parameters in spherical coordinate format. The fully connected processing network can process the fused multimodal features through learned transformations that produce angular predictions that represent the direction of the user's gaze relative to the camera coordinate system.

The fully connected processing network transmits the angle prediction to the 3D gaze vector reconstruction module.

Predicted pitch and yaw angles are provided that represent the direction of the gaze in spherical coordinate format for subsequent three-dimensional vector conversion operations. In some embodiments, angular parameters are provided that serve as inputs to the system coordinate transformation process, which converts the spherical representation into Cartesian vector format.

The 3D gaze vector reconstruction module can convert spherical coordinates to Cartesian coordinates and normalize vectors.

Trigonometric transforms are applied using the sine and cosine functions to generate 3D coordinate components from the received pitch and yaw angles. The 3D gaze vector reconstruction module can calculate x, y, and z coordinate values, and can normalize the resulting vectors to unit lengths to ensure the mathematical stability of subsequent coordinate frame construction operations.

The 3D gaze vector reconstruction module can send a normalized gaze vector to the orthogonal base building unit.

A gaze vector per unit length that has been converted from spherical coordinates to Cartesian coordinate representation is provided. The orthogonal basis construction unit can receive normalized 3D vectors for subsequent orthogonal coordinate framework establishment operations.

The orthonormal basis construction unit can perform fork product operations and establish orthogonal coordinate frames.

The orthogonal basis vector is generated by cross product calculations between a

normalized gaze vector and a reference coordinate axis. The Orthogonal basis construction unit can create a complete three-dimensional coordinate system that establishes the spatial orientation relationship of the gaze vector within the target coordinate framework through orthogonality normalization and verification operations.

The orthogonal basis construction unit can transfer the orthogonal basis to the camera coordinate transform unit in step S942. An orthogonal basis vector can be provided for a stable coordinate reference system that defines the spatial orientation of the gaze vector. In some embodiments, a mathematical framework is provided that implements the transformation of the coordinate system between the original gaze vector space and the representation of the camera alignment coordinates.

The camera coordinate transform unit can construct a rotation matrix and transform vectors into the camera coordinates.

A 3x3 rotation matrix is constructed from orthonormal basis vectors received from the orthonormal basis construction unit, and a rotation matrix can be applied to convert the normalized gaze vector into a camera-aligned coordinate space. The camera coordinate transformation unit enables integration with imaging systems and spatial tracking applications operating within the camera coordinate frame.

The camera coordinate transformation unit can output the final 3D gaze ray for space applications.

A parametric ray representation is provided that combines the processed gaze direction vector with the three-dimensional eye center coordinates to create a complete ray definition.

A start vector and a direction vector of gaze rays are generated, enabling intersecting calculations with virtual objects, user interface elements, and physical surfaces in augmented and virtual reality environments. In some embodiments, mathematically stable gaze ray representations are generated that support gaze intersection calculations for user interface control, diagnostic applications, and immersive environment interaction systems.

Referring to Figure 15, a LightGazeNet deployment system can provide multi-platform implementation capabilities, which can implement gaze estimation based on

lightweight graph neural networks in different hardware configurations and deployment scenarios. The LightGazeNet deployment system demonstrates the versatility and adaptability of the gaze estimation architecture through three different platform configurations that accommodate different computing resources, imaging hardware, and application requirements while maintaining consistent gaze estimation capabilities across all deployment environments.

The LightGazeNet deployment systems can include a mobile device platform that provides gaze estimation capabilities for smartphone and tablet applications. The mobile device platforms enable real-time eye tracking on consumer mobile devices with optimized processing operations that leverage the lightweight architecture of graph neural networks and efficient feature extraction operations through depthwise separable convolution and squeeze-and-excitation module.

A smartphone camera within the mobile device platform can capture visual input data for gaze estimation processing using standard front-facing camera hardware commonly found on consumer mobile devices. The smartphone cameras can provide RGB image data without the need for specialized infrared lighting systems or additional hardware components. In some embodiments, the smartphone camera can work with a variety of mobile camera specifications and resolutions while maintaining compatibility with normalized camera interface processing operations that standardize input data for subsequent feature extraction and analysis.

The smartphone camera can be connected to an embedded processor that handles the computational requirements of the gaze estimation algorithm within the mobile device environment. Embedded processors can use the processing power of mobile devices to perform multimodal input processing, uniform feature encoding, graph neural networks, and output generation operations. The embedded processor can reduce the number of parameters by about 3.48 million parameters to achieve computing efficiency suitable for mobile deployment scenarios.

Embedded processors can be connected to a real-time engine that manages real-time processing and output generation for mobile apps. The real-time engine can coordinate sequential processing operations and can ensure that gaze estimation results are generated

at the appropriate frame rate for the interactive application. In some embodiments, the real-time engine can achieve real-time performance of more than 30 frames per second on mobile devices through lightweight processing operations implemented across feature extraction, graph neural networks, and vector reconstruction stages.

The LightGazeNet deployment systems can further include AR/VR platforms, which provide specialized capabilities for augmented and virtual reality applications. AR/VR platforms can include components designed for immersive environments that require precise 3D gaze vector reconstruction and spatial tracking integration.

Head-mounted displays within AR/VR platforms can provide visual output to users while serving as a platform for eye tracking hardware integration. The head-mounted displays can house imaging components and processing hardware that enable gaze estimation in an immersive virtual environment. The head-mounted displays can work in tandem with 3D vector reconstruction operations, providing gaze representations suitable for interacting with virtual objects and user interface elements within augmented and virtual reality coordinate frames.

The head-mounted display can be connected to a panoramic camera that captures environmental and user vision data for comprehensive gaze estimation processing. Panoramic cameras can provide imaging capabilities that capture the user's facial features and eye area, as well as contextual information about the environment. In some embodiments, the panoramic camera can operate in a constrained power consumption and computing environment typical of head-mounted display systems while maintaining the visual data quality required for accurate feature extraction by facial feature extractors and eye feature extractors.

Both head-mounted displays and panoramic cameras can be connected to a spatial tracker that handles three-dimensional positioning and gaze vector reconstruction for immersive applications. The spatial trackers can integrate gaze estimation results with spatial tracking systems that monitor head position and orientation within a virtual environment. The spatial tracker can work in tandem with the 3D gaze vector reconstruction module, the orthogonal base building unit, and the camera coordinate transform unit to generate parametric rays that can be computed intersecting with virtual

objects and surfaces within augmented and virtual reality coordinate systems.

The LightGazeNet deployment systems can additionally include a desktop platform that provides traditional computer-based implementations with standard webcam hardware and desktop application integration. The desktop platforms enable gaze estimation capabilities for desktops and laptops without the need for specialized hardware beyond common webcam devices.

A webcam interface within the desktop platform can capture user visual data through standard webcam hardware commonly found on desktop and laptop systems. The webcam interface can provide RGB image data that is processed through the same multimodal input processing operations used in other platform configurations. In some embodiments, the webcam interface can be adapted to various webcam specifications and mounting configurations while maintaining compatibility with feature extraction and processing operations implemented throughout the gaze estimation pipeline.

The webcam interface can be connected to the application layer, which handles software integration and processing coordination within the desktop computing environment. The application layer can manage the execution of unified feature encoding, graph neural networks, and output generation operations within the desktop application framework. The application layer can orchestrate the data flow between webcam input processing and gaze estimation algorithms, while providing an interface to integrate with desktop software applications.

The application layer can be connected to the user interface, which manages user interaction and system control functions within the desktop app. The user interface provides control over calibration operations, system configuration, and visual visibility of gaze estimation results. In some embodiments, the user interface provides access to a specific subject calibration capability implemented through calibration embeddings and can provide feedback on gaze estimation performance and accuracy.

The LightGazeNet deployment systems can include cross-platform connectivity capabilities, illustrated by connections between the real-time engine and the spatial tracker and application layer. These connections enable interoperability and data sharing between different platform implementations, allowing gaze estimation systems to operate

on multiple device types while maintaining consistent functionality. Cross-platform connectivity facilitates deployment scenarios where gaze estimation results are shared between mobile devices, AR/VR systems, and desktop platforms for collaborative applications or multi-device interaction systems.

The real-time engine can communicate with spatial trackers to enable integration between mobile device gaze estimation and AR/VR spatial tracking systems. This connectivity can support applications where mobile devices are used as input sources for immersive environments, or applications where gaze estimation results are transferred between mobile and AR/VR platforms. In some embodiments, the connection between the real-time engine and the spatial tracker enables synchronized eye tracking across multiple device types within a shared virtual environment.

The real-time engine can also communicate with the application layer to enable integration between mobile gaze estimation and desktop application environments. This connection can support scenarios where mobile devices provide gaze input for desktop applications, or where gaze estimation results are processed on mobile and desktop platforms. The connectivity between the real-time engine and the application layer facilitates cross-platform data sharing and coordinated processing operations that leverage the computing power of multiple device types.

The LightGazeNet deployment system can achieve consistent gaze estimation capabilities across all three platform configurations through a unified architecture implemented in graph neural networks and standardized processing operations applied in the multimodal input processing, unified feature encoding, and output generation phases. The system can maintain a model size of less than 5 megabytes across all deployment platforms, enabling cloud-free device gaze estimation that operates independently of network connectivity requirements. In some embodiments, the LightGazeNet deployment system can operate at a low power consumption of less than 500 milliwatts, making the system suitable for battery-powered mobile devices, embedded systems, and low-power AR/VR platforms while maintaining real-time performance characteristics across all supported hardware configurations.

Referring to Figure 16, a multi-head attention architecture can provide specialized

attention-based processing capabilities that enable comprehensive cross-modal relationship modeling within graph neural networks. The multi-head attention architectures can implement systematic attention mechanisms that capture diverse relational patterns between heterogeneous input modalities through parallel processing operations and coordinated feature transformation phases.

The multi-head attention architecture can include an array of attention heads that contain four parallel processing units designed to capture different types of relationship patterns and dependencies within the input data simultaneously. Attention head arrays can enable the system to model multiple aspects of a cross-modal relationship through independent attention mechanisms that operate with different learning parameters while processing the same input representation.

The attention head array can include a first head, which is embedded in the input node through a first set of learned attention parameters. The first head captures specific relationship patterns between a six input modalities represented within a six-node graph structure, including the relationship between facial features, eye features, head rotation data, and spatial positioning coordinates. In some embodiments, the first head can focus on specific types of cross-modal dependencies that differ from the patterns captured by other attention heads within the array.

A second head within the attention head array can be manipulated using an independent parameter sets to capture alternative relationship patterns between input modalities. The second head can handle the same node embeddings as the first head, while applying different learning transformations that capture complementary relational information. The second head can work in parallel with the first head, providing diverse perspectives on the cross-modal interactions that exist within the graph structure.

The attention head array can further include a third head, which provides additional parallel processing capabilities for comprehensive relational modeling. The third head can apply different attention mechanisms that capture relational aspects that are not covered by the first head or second head. In some embodiments, the third head enables the multi-head attention architecture to model complex multiplexing relationships between different input modal combinations through specialized attention parameter

configuration.

A fourth head within the attention head array can complete the parallel processing array by providing a fourth independent attention mechanism. The fourth head can capture additional relationship patterns that complement processing operations performed by the first head, the second head, and the third head. The fourth head can ensure comprehensive coverage of the model across modal relationships through the combination of all four parallel attention heads.

The multi-head attention architecture can include a linear projection layer that receives input from all four attention heads within the attention head array. The linear projection layer can convert input features into queries, keys, and value representations that are used for subsequent attention calculation operations. The linear projection layer can apply the learned linear transformation to prepare node embeddings for attention coefficient calculation and feature aggregation processes.

The linear projection layer can include a query projection that converts input node features into a query representation for attention coefficient calculations. The query projection can embed a linear transformation of application learning on nodes received from the attention head array to generate query vectors that represent the information-seeking aspects of each node within the attention mechanism. In some embodiments, query projections can create query representations that enable each node to identify relevant information from other nodes within the graph structure.

A key projection within the linear projection layer can convert input node features into key representations that serve as targets for attention coefficient calculations. The key projections can be applied to different linear transformations to generate key vectors that represent the information provided by each node within the attention frame. The key projections can work in tandem with query projections to enable the calculation of attention coefficients that quantify the strength of relationships between different node pairs.

The linear projection layer can further include a value projection that converts input node features into value representations that contain actual information aggregated through attention mechanisms. The value projection can apply learned transformations to

generate value vectors that represent feature content, which will be combined based on attention weights calculated through the interaction between the query and the key representation. In some embodiments, the value projection can provide feature information that is weighted and aggregated according to attention weights to produce an eventual attention-based node representation.

The multi-head attention architectures can include an attention computing module that processes projection features from linear projection layers to generate attention-weighted outputs. The attention computing module can implement the system's attention factor calculations, normalization, and feature aggregation operations that combine queries, keys, and value representations into enhanced node embeddings that capture cross-modal relationship information.

The attention calculation module can include a score calculator that calculates the attention factor between different input elements using queries and key representations generated by the linear projection layer. The score calculator can calculate the dot product or other similarity measure between the query vector from the query projection and the key vector from the key projection to generate the raw attention coefficient. The Score Calculator quantifies the strength of the relationship between each pair of nodes within a six-node graph structure through the system score calculation operation.

A Softmax normalizer within the attention calculation module can process the raw attention coefficient from the score calculator to create a normalized probability distribution. The Softmax normalizer can apply exponential scaling followed by normalization operations that ensure that the attention weights of each node add up to one while maintaining the proper gradient flow for the training operation. In some embodiments, the Softmax normalizer can generate normalized attention coefficients that represent the relative importance of each node connection within the graph structure.

The attention computing module can further include a feature aggregator that combines weighted features to produce the final attention output. The feature aggregator can receive the normalized attention coefficient from the Softmax normalizer, as well as the value representation from the value projection to calculate the weighted combination of node features. The feature aggregators can generate enhanced node embeddings that

merge information from relevant modalities based on attention-weighted importance determined through score calculation and normalization processes.

Connections within the multi-head attention architecture can establish a systematic information flow from the attention head through linear projection to attention computing operations. The first head, the second head, the third head, and the fourth head can all be connected to the linear projection layer to provide parallel processing inputs for feature transformation operations. Parallel joining enables simultaneous processing of the same input embeddings through multiple attention mechanisms while maintaining an independent set of parameters for each attention head.

Both the query projections and the key projections can be connected to the score calculator to provide the query and key representations needed for attention factor calculations. Connections enable the score calculator to compute similarity measures between queries and key vectors that quantify the strength of the relationship between different node pairs within the graph structure. In some embodiments, the connection can facilitate the systematic computation of attention coefficients that capture cross-modal dependencies between heterogeneous input modalities.

The value projections can be connected directly to feature aggregators to provide feature content that will be combined based on calculated attention weights. Connections enable feature aggregators to access representations of values that contain actual information aggregated through attention-weighted combinatorial operations. Direct connections ensure that the feature content from the value projection is properly integrated with the attention coefficients generated through the fraction calculation and normalization process.

The score calculator can be connected to the Softmax normalizer to provide the raw attention coefficient used for normalization operations. Connections enable the transfer of calculated attention coefficients from the score calculation stage to the normalization stage, where exponential scaling and softmax operations generate appropriately weighted attention distributions. The Softmax normalizer can then be connected to the feature aggregator to provide a normalized attention coefficient for weighted feature combination operations.

The linear projection layer can be connected to the attention computing module to establish a holistic flow of information from feature transformations to attention-based aggregations. Connections can orchestrate the transition of queries, keys, and value representations from the projection operation to the attention computation process that generates the final enhancement node embedding. In some embodiments, the connection ensures that the transformation features from the linear projection layer are properly integrated within the attention computing framework implemented by the attention computing module.

The multi-head attention architecture can achieve the capture of diverse relationship patterns through the parallel operation of four attention heads in the attention head array, combined with the system feature transformation and aggregation operations implemented through the linear projection layer and the attention computing module. The architecture can provide comprehensive cross-modal relationship modeling capabilities that enhance the representation quality of node embeddings within the six-node graph structure while maintaining computational efficiency suitable for real-time deployment scenarios on mobile devices, AR/VR platforms, and desktop platforms.

It is understandable that the multimodal gaze estimation system based on the lightweight graph neural network provided by the embodiment of the present application corresponds to the above-mentioned multimodal gaze estimation method based on the lightweight graph neural network, and the explanation, examples, beneficial effects and other parts of the relevant content can refer to the corresponding content in the multimodal gaze estimation method based on the lightweight graph neural network, which will not be repeated here.

#### **Example 6:**

In a third aspect, the present application further provides an electronic device, comprising a memory, a processor, and a computer program stored on the memory and executable on the processor, wherein the processor, when executing the program, implements steps of the multimodal gaze estimation method based on a lightweight graph neural network as described in any of the above embodiments and its preferred embodiment.

It is understandable that the multimodal gaze estimation electronic device based on the lightweight graph neural network provided by the embodiment of the present application corresponds to the above-mentioned multimodal gaze estimation method and system based on the lightweight graph neural network, and the explanation, examples, beneficial effects and other parts of the relevant content can refer to the multimodal gaze estimation method based on the lightweight graph neural network and the corresponding content in the system, which will not be repeated here.

**Example 7:**

In a fourth aspect, the present application further provides a non-transient computer-readable storage medium on which a computer program is stored, wherein the computer program, when executed by the processor, the steps for implementing a multimodal gaze estimation method based on a lightweight graph neural network described in any of the above embodiments and its preferred embodiments are implemented.

It is understandable that the multimodal gaze estimation storage medium based on the lightweight graph neural network provided by the embodiment of the present application corresponds to the above-mentioned multimodal gaze estimation method and system based on the lightweight graph neural network, and the explanation, examples, beneficial effects, and other parts of the relevant content can refer to the multimodal gaze estimation method based on the lightweight graph neural network and the corresponding content in the system, which will not be repeated here.

The above embodiments are only used to illustrate the technical solution of the present application, not to limit it. Although the present application is described in detail with reference to the aforementioned embodiments, ordinary technicians in this field should understand that it can still modify the technical solutions recorded in the aforementioned embodiments, or replace some of the technical features therein; These modifications or substitutions do not make the essence of the corresponding technical solution out of the spirit and scope of the technical solutions of each embodiment of the present application.

## Claim

---

1. A multimodal gaze estimation method based on a lightweight graph neural network (GNN), wherein the method comprises:

obtaining multimodal heterogeneous data related to and complementary for gaze direction estimation, wherein the multimodal heterogeneous data comprises visual appearance data, geometric orientation information and spatial positioning coordinates;

extracting visual features from the visual appearance data based on depthwise separable convolution and squeeze-and-excitation module;

projecting the visual features, the geometric orientation information, and the spatial positioning coordinates into a common vector space;

constructing a graph structure, wherein the graph structure comprises a plurality of nodes representing the multimodal heterogeneous data or features corresponding to the multimodal heterogeneous data;

processing the constructed graph using a multi-head attention graph neural network having a two-layer architecture to obtain graph-level features, wherein a first layer of the two-layer architecture expands dimensionality of node features, and a second layer of the two-layer architecture reduces dimensionality of the expanded dimension;

aggregating the graph-level features and fusing the aggregated graph-level features with the visual features, the geometric orientation information, and the spatial positioning coordinates;

performing gaze estimation prediction based on the fused features using a fully connected layer; and

converting the gaze estimation prediction into a normalized three-dimensional gaze vector based on coordinate conversion.

2. The method of claim 1, wherein the visual appearance data comprises facial image data, eye cropping image data, the geometric orientation information comprises head rotation data, and the spatial positioning coordinate comprises three-dimensional eye center position.

3. The method of claim 1, wherein constructing a graph structure comprises:

constructing a fully connected graph with six nodes; wherein the six nodes comprise nodes corresponding to: facial features, left eye features, right eye features, head rotation data, left eye three-dimensional position, and right eye three-dimensional position; and establishing learnable edge weights between pairs of the six nodes.

4. The method of claim 1, wherein coordinate conversion comprises converting spherical coordinates into Cartesian coordinates.

5. The method of claim 1, further comprising:

when estimating the gaze for a specific subject, incorporating subject-specific calibration data into the multimodal heterogeneous data.

6. The method of claim 5, wherein the subject-specific calibration comprises:

updating only the specific subject calibration embedding and final fully connected layer parameters during calibration, while maintaining all other network parameters in a frozen state.

7. The method of claim 3, wherein processing the constructed graph structure using a multi-head attention graph neural network having a two-layer architecture to obtain graph-level features comprises:

Step 4.1: initializing the six graph nodes using the projected feature embedding;

Step 4.2: calculating attention coefficients of all node pairs using linear projection;

Step 4.3: applying LeakyReLU activation and softmax normalization to the attention coefficients;

Step 4.4: aggregating neighboring features using weighted attention coefficients;

Step 4.5: determining whether current processing corresponds to a first GNN layer, and if yes, proceeding to step 4.6; if no, proceeding to step 4.7

Step 4.6: expanding node features to 256 dimensions;

Step 4.7: reducing node features to 32 dimensions;

Step 4.8: applying a second GNN layer processing;

Step 4.9: flattening and concatenating all node features into a unified representation;

and

Step 4.10: outputting a final graph-level feature vector.

8. The method of claim 1, wherein converting the gaze estimation prediction into a

normalized three-dimensional gaze vector based on coordinate conversion comprises:

- constructing an orthogonal normalized coordinate frame using a cross product; and
- generating a parametric ray emanating from the three-dimensional eye center.

9. The method of claim 8, wherein the parametric ray is capable of performing intersection calculations with objects in spatial applications, the objects comprising virtual objects, user interfaces, and physical surfaces.

10. The method of claim 1, wherein the multi-head attention comprises four attention heads processed in parallel, wherein:

- a first attention head is configured to capture specific relationship patterns among the six input modalities represented within the six-node graph structure;

- a second attention head is configured to capture alternative relationship patterns among the input modalities;

- a third attention head is configured to provide additional parallel processing capacity for comprehensive relationship modeling; and

- a fourth attention head is configured to complete a parallel processing array by providing a fourth independent attention mechanism.

11. A multimodal gaze estimation system based on a lightweight graph neural network, wherein the system comprises:

- a multimodal input module configured to obtain multimodal heterogeneous data related to and complementary for gaze direction estimation, wherein the multimodal heterogeneous data comprises visual appearance data, geometric orientation information, and spatial positioning coordinates;

- a feature extraction module configured to extract visual features from the visual appearance data based on depthwise separable convolution and squeeze-and-excitation module;

- a projection module configured to project the visual features, the geometric orientation information, and the spatial positioning coordinates into a common vector space;

- a graph construction module configured to construct a graph structure, wherein the graph structure comprises a plurality of nodes representing the multimodal heterogeneous

data or features corresponding to the multimodal heterogeneous data;

a multi-head attention graph neural network configured to process the constructed graph using a multi-head attention graph neural network having a two-layer architecture to obtain graph-level features, wherein a first layer of the two-layer architecture expands dimensionality of node features, and a second layer of the two-layer architecture reduces dimensionality of the expanded dimension;

a feature fusion module configured to aggregate the graph-level features and fuse the aggregated graph-level features with the visual features, the geometric orientation information, and the spatial positioning coordinates;

a regression module configured to perform gaze estimation prediction based on the fused features using a fully connected layer; and

a three-dimensional reconstruction module configured to convert the gaze estimation prediction into a normalized three-dimensional gaze vector based on coordinate conversion.

12. An electronic device comprising a memory, a processor, and a computer program stored on the memory and executable on the processor, wherein the processor, when executing the program, implements the steps of the multimodal gaze estimation method based on a lightweight graph neural network according to any one of claims 1-10.

13. A non-transitory computer-readable storage medium having stored thereon a computer program, wherein the computer program, when executed by a processor, implements the steps of the multimodal gaze estimation method based on a lightweight graph neural network according to any one of claims 1-10.