Head Pose Estimation in Complex Environment Based on Four-Branch Feature Selective Extraction and Regional Information Exchange Fusion Network

BIN-YU WANG1,2,3, KAI XIE1,2,3, SHENG-TAO HE1,2,3, CHANG WEN1,2,3,4, AND JIAN-BIAO HE5

1School of Electronic Information, Yangtze University, Jingzhou 434023, China
2National Demonstration Center for Experimental Electrical and Electronic Education, Yangtze University, Jingzhou 434023, China
3Western Institute, Yangtze University, Karamay 834000, China
4School of Computer Science, Yangtze University, Jingzhou 434023, China
5School of Computer Science and Engineering, Central South University, Changsha 410083, China

Corresponding author: Kai Xie (500646@yangtzeu.edu.cn)

This work was supported in part by the Natural Science Foundation of Xinjiang Uygur Autonomous Region under Grant 2020D01A131; in part by the Fund of Hubei Ministry of Education under Grant B2019039; in part by the Graduate Teaching and Research Fund of Yangtze University under Grant JXY201909; in part by the Teaching and Research Fund of Yangtze University under Grant JY2019011; in part by the Undergraduate Training Programs for Innovation and Entrepreneurship of Yangtze University under Grant Yz2020057, Grant Yz2020059, and Grant Yz2020156; and in part by the National College Student Innovation and Entrepreneurship Training Program under Grant 202110489003.

ABSTRACT Under the severe situation of the COVID-19 pandemic, masks cover most of the effective facial features of users, and their head pose changes significantly in a complex environment, which makes the accuracy of head pose estimation in some systems such as safe driving systems and attention detection systems impossible to guarantee. To this end, we propose a powerful four-branch feature selective extraction network (FSEN) structure, in which three branches are used to extract three independent discriminative features of pose angles, and one branch is used to extract composite features corresponding to multiple pose angles. By reducing the dimension of high-dimensional features, our method significantly reduces the amount of computation while improving the estimation accuracy. Our convolution method is an improved spatial channel dynamic convolution (SCDC) that initially enhances the extracted features. Additionally, we embed a regional information exchange network (RIEN) after each convolutional layer in each branch to fully mine the potential semantic correlation between regions from multiple perspectives and learn and fuse this correlation to further enhance feature expression. Finally, we fuse the independent discriminative features of each pose angle and composite features from the three directions of channel, space, and pixel to obtain perfect feature expression for each pose angle, and then obtain the head pose angle. We conducted extensive experiments on the controlled environment datasets and a self-built real complex environment dataset (RCE) and the results showed that our method outperforms state-of-the-art single-modality methods and performs on par with multimodality-based methods. This shows that our network meets the requirements of accurate head-pose estimation in real complex environments such as complex illumination and partial occlusion.

INDEX TERMS Head pose estimation, four-branch feature selective extraction, regional information exchange network, spatial channel dynamic convolution, multiple feature fusion, complex environment.

I. INTRODUCTION

With the gradual development of artificial intelligence, human-computer interaction has become particularly important in daily life. As a simple and natural method, head pose estimation has several applications in human-computer interaction, deriving its basis from the use of head pose estimation in computer vision, for estimating/predicting the direction that a human head is facing, that is, predicting the so-called Euler angles (yaw, pitch, and roll). Therefore, it can...
be applied to sight detection, assisted driving, and other purposes. It also lends head pose estimation to applications for mapping 3D objects to match the direction of a human head, examples of which can be seen on Snapchat, Instagram, and TikTok, as well as in 3D gaming and animation. Head pose estimation is becoming more popular in several fields, including health and fitness and content creation which involve animating 3D human models. Head pose estimation is also an important preprocessing step for further analysis of gaze estimation [1], [2], understanding human-object interaction [3]–[5], and human-robot interaction [6]. Furthermore, head pose estimation is increasingly used as an important indicator for evaluating whether a person's attention is concentrated as it reflects a person's mental activity. This has been tested in cognitive psychology and neurophysiology [7]. Real-time head-pose estimation in a complex environment on mobile devices is an important development direction. However, head pose estimation on mobile devices often has problems such as low computational power, difficulty in meeting real-time detection requirements, poor estimation accuracy, and in head pose estimation in complex environments. Therefore, a mobile-friendly algorithm that can efficiently and accurately estimate head pose in complex environments is required.

Classic head-pose estimation algorithms include methods based on appearance template matching [8], geometric models [9], [10], depth images [11], machine learning [12], human face landmarks [13], registration-based tracking [14], and multitasking [15]. Among them, the method based on appearance template matching was commonly used in the early days. This method compares a person's head-pose image against a sample set of pose images having known head poses to obtain the person's head pose [16]. This is generally suitable for estimating the pose of the person's face. The method based on the geometric model estimates head pose by accurately positioning the key points of the head and combining the shape of the head for face modeling [17], [18]. The method, based on depth images, is mainly used to estimate the head pose by capturing the missing 3D image information in the 2D image of the person's head [1], [11], [19]. This often requires a 3D depth camera that may not be always available. Algorithms based on machine learning include feature regression [20], random forest regression [21], [22], and manifold learning [16]. Because these methods have disadvantages such as being relatively expensive and having high time requirement, machine learning algorithms have been gradually developed to use convolutional neural networks to estimate face models [23]–[25]. Methods based on facial landmarks mainly use facial landmark detection and computer vision technology to estimate the head pose [23], [26]. However, this method requires manual labeling, and obtaining labeled landmarks is labor intensive. Even experts cannot accurately locate facial landmarks in some low-resolution images. Considering cost and accuracy, some scholars have proposed algorithms that do not require facial landmarks for face alignment [13], [27]. A method based on motion tracking works by judging the head deflection angle based on the relative movement of the human head in a video. This method has proven to have a high accuracy rate. However, a major difficulty associated with this method is the need to accurately initialize the head position to generate new patterns. Additionally, some scholars have proposed more advanced methods without face detection or landmark positioning, such as the img2pose [40] that can perform real-time, six degrees of freedom (6DoF) 3D face pose estimation. However, it is also popular to adopt different modalities to compensate for the loss of information [28], [29]. FacePoseNet uses a convolutional neural network (CNN) to perform 3D head pose regression, which improves the accuracy of the model. The model proposed by Nataniel et al. [30] combines ResNet50 with a multiloss architecture to obtain a robust neighborhood prediction of the pose. Fanelli et al. [21] used random forest regression for head-pose estimation. Meyer et al. [19] proposed a method for registering a 3D deformed model to a depth image for head pose estimation. Inspired by the similarity between Bayesian filters and recurrent neural networks (RNNs), Gu et al. [38] proposed a method that uses an RNN to track facial features over time. The latest research shows that, compared with single-task, multitask joint learning achieves better results [31], [32]. Hyperface [15], a multitask learning framework, simultaneously performs face detection, landmark localization, pose estimation, and gender recognition using deep convolutional neural networks. KEPLER [33] explored multiple structural dependencies using CNN. Multi-task neural network (MNN) [39] is a multitask image head-pose estimation method based in deep learning. We take advantage of the strong dependencies between the facial pose, alignment, and visibility to generate an optimally executed model.

Although the current head pose estimation algorithm has achieved a very high-performance level, meeting the requirements of accurate head pose estimation in complex environments such as those involving mask occlusion remains difficult. Therefore, this study proposes a head pose estimation algorithm for complex environments based on a four-branch feature-selective extraction and regional information exchange fusion network. The contributions of this study are as follows.

1. A more powerful four-branch feature selective extraction network (FSEN) was proposed, in which three branches were used to extract three independent discriminative features of pose angles, and one branch was used to extract features corresponding to multiple pose angles. By reducing the three-dimensional pose angle vector into three one-dimensional angle vectors, the amount of computation is significantly reduced, and the estimation accuracy is improved.

2. The improved spatial channel dynamic convolution method (SCDC) was used to initially enhance the extracted features, and a regional information exchange network (RIEN) was embedded after each convolutional layer to fully tap the potential between regions from multiple perspectives. Semantic correlation, learning, and fusing were applied to further enhance feature expression. Thus, the problem of
less effective features in the case of occlusion was effectively solved.

3. Integration of the independent discriminative features of each pose angle with the composite features from the three directions of the channel, space, and pixel enabled to obtain the most perfect feature expression for each pose angle, and the most accurate head pose angle.

The remainder of the study is organized as follows. Section 2 presents the algorithm of the proposed model. Section 3 presents the experimental setup, details, and analysis of the results. Section 4 presents the conclusions of the study and an outlook for future work.

II. METHOD

The head pose estimation algorithm proposed in this study is divided into four parts: image preprocessing based on an adaptive illumination processing network and face active region extraction network, based on a four-branch FSEN; SCDC, and RIEN feature extraction and enhancement; feature fusion based on multiple feature fusion network (MFFN); and head pose estimation based on a deep aggregation network and regression model. The flow of the specific algorithm is illustrated in Fig. 1. In the first step, we used the improved LIME (Local Interpretable Model-agnostic Explanations) and MSRCR (Multi Scale Retinex with Color Restoration) algorithms with adaptive brightness correction to perform adaptive illumination processing on the original image acquired using mobile devices, and simultaneously restore the color information of the image. The active area of the head was extracted using the MTCNN+Mobilenet and Gaussian skin-color models. In the second step, we fed the active head region into a four-branch FSEN to extract the independent discriminative features of each pose angle and the composite features corresponding to multiple pose angles. An improved SCDC was used to initially enhance the features, at the same time, the regional information exchange network (RIEN) is embedded after each convolutional layer to fully mine the potential semantic correlation between regions from multiple perspectives, and learn and fuse this correlation to further enhance the feature expression. In the third step, we fused the independent discriminative features of each pose angle with the composite features from the three directions of the channel, space, and pixel to obtain the perfect feature expression for each pose angle. In the last step, we performed deep feature aggregation and regression on the -perfect feature representation of each pose angle to obtain the head pose angle and then introduce the algorithm of each step-in detail.

A. IMAGE PREPROCESSING

Images collected by mobile devices may not successfully be used to perform face detection and subsequent head pose estimation owing to the lack of low light, strong light, and color information. Therefore, in the algorithm proposed in this study, we first perform adaptive lighting processing, and at the same time, restore the color information of the image to get a high-quality image and ensure progress in the subsequent stage of our work.

1) LOW LIGHT ENHANCEMENT

For low-light enhancement, we improved the LIME algorithm based on the retinex theory and applied it to the model proposed in this study. The core idea was to decompose the image in the dark to produce illuminated and reflected images. The illuminated image was enhanced and combined with the reflected image to achieve low-light enhancement. The specific steps are as follows:

First, we performed initial brightness estimation on the input low-exposure image, as shown in Eq. (1).

$$\hat{T}(x) \leftarrow \max_{c \in [R,G,B]} L^c(x)$$  \hspace{1cm} (1)

We then calculated the weight matrix W of the image, as shown in Eq. (2).

$$W_h(x) \leftarrow \sum_{y \in \Omega(x)} \frac{G_\sigma(x,y)}{\sum_{y \in \Omega(x)} G_\sigma(x,y)\nabla_h \hat{T}(y)} + \epsilon$$  \hspace{1cm} (2)

$G_\sigma(x,y)$ is produced by Gaussian kernel of standard deviation.

Then, $T$ is calculated when the expression value is the smallest, that is, the illumination image through the optimization equation, as shown in Eq. (3).

$$\min_T \| \hat{T} - T \|^2_F + \alpha \sum_x \frac{W_h(x)(\nabla_h T(x))^2}{|\nabla_h \hat{T}(x)| + \epsilon} + \frac{W_i(x)(\nabla_i T(x))^2}{|\nabla_i \hat{T}(x)| + \epsilon}$$  \hspace{1cm} (3)

where $\alpha$ represents the balance coefficient.

Finally, after obtaining the estimated value of $T$, an enhanced image was obtained using Eq. (4).

$$L = I \circ T$$  \hspace{1cm} (4)

where $L$ is a low-exposure image, $I$ is an enhanced image, and $T$ is an illuminated image.

2) STRONG LIGHT SUPPRESSION

For strong light suppression and color restoration, we merged the MSRCR algorithm with the gamma adaptive correction algorithm to propose an adaptive brightness correction MSRCR algorithm and applied it to this model. This algorithm can not only adjust the brightness and contrast of the image that is interfered with via strong light or even over-exposure but can also restore the color information of the image, and finally present a good visual effect. The steps of the algorithm are as follows:

Step 1. The image was converted from the RGB space to the HSV space, and the process can be expressed using
Eqs (5), (6), and (7).

\[
H = \begin{cases} 
0^\circ, & \text{if } V = \min(R, G, B); \\
60^\circ, & \text{if } V = R; \\
120^\circ, & \text{if } V = G; \\
240^\circ, & \text{if } V = B; \\
\end{cases} 
\]

(5)

\[
S = \begin{cases} 
0, & \text{if } V = \max(R, G, B), \\
\frac{V - \min(R, G, B)}{V}, & \text{if } V \neq 0; \\
\frac{1 - \min(R, G, B)}{V}, & \text{if } V = 0; \\
\end{cases} 
\]

(6)

\[
V = \max(R, G, B) 
\]

(7)

Step 2. The illumination component of the incident light was obtained via smooth filtering, and the algorithm process is as follows:

For an image, \( I(x, y) \) can be represented by both the incident light and reflected light images.

\[
I(x, y) = R(x, y) + L(x, y) 
\]

(8)

where \( L(x, y) \) represents the incident light image, \( R(x, y) \) represents the reflected light image.

Eq. (9) can be obtained by logarithmically transforming Eq. (8).

\[
\ln[I(x, y)] = \ln[R(x, y)] + \ln[L(x, y)] 
\]

(9)

Then, we used median filtering to reduce \( \ln[R(x, y)] \) to zero and exponentially transformed the remaining \( \ln[L(x, y)] \) to obtain the illumination component: \( L(x, y) \).

Step 3. We used adaptive brightness correction of the gamma function for the smoothed illumination components.

To cope with different lighting conditions, we performed a segmented gamma correction on the regions with different brightness values in the image. Suppose that \( \mu \) is the brightness of the image, \( \mu_L \) is the low brightness value, \( \mu_H \) is the high brightness value, and the value between \( \mu_L \) and \( \mu_H \) is the normal brightness value, then the relationship between the gamma value and the brightness value can be expressed as follows:

\[
\gamma = \begin{cases} 
1 + \frac{\mu_L - \mu}{2\mu} & \text{if } \mu \in [0, \mu_L]; \\
1 & \text{if } \mu \in [\mu_L, \mu_H]; \\
1 - \frac{\mu - \mu_H}{2(255 - \mu_H)} & \text{if } \mu \in [\mu_H, 255]; \\
\end{cases} 
\]

(10)

Then, we obtained the corrected light component:

\[
L'(x, y) = 255\left(\frac{I(x, y)}{255}\right)^{\gamma} 
\]

(11)

Step 4. Finally, we used the corrected brightness, saturation, and hue to resynthesize the RGB image and used the MSR algorithm with color restoration to reduce the color distortion of the image. The specific algorithm can be expressed using Eqs. (12)–(14).

\[
R_{MSRCR}(x, y) = C_i(x, y)R_{MSR}(x, y) 
\]

(12)

\[
C_i(x, y) = f[I'_i(x, y)] = \frac{1}{N} \sum_{j=1}^{N} I_j(x, y) 
\]

(13)

\[
f[I'_i(x, y)] = \beta \log[\alpha I'_i(x, y)] = \beta \left\{ \log[\alpha I'_i(x, y)] - \log \left[ \sum_{j=1}^{N} I_j(x, y) \right] \right\} 
\]

(14)
where \( C_j \) represents the color restoration factor of the \( j \)th channel, which is used to change the proportions of the three-color components. \( I_i(x, y) \) represents an image of the \( i \)th channel. \( f() \) represents the color-space mapping function, \( \beta \) is the gain function, and \( \sigma \) is the controlled nonlinear intensity.

Additionally, to avoid the case where the pixel value is less than zero, it is necessary to edit the image using gain and deviation, as shown in Eq. (15).

\[
R_{MSRCR}(x, y)^{'} = GR_{MSRCR}(x, y) + o \tag{15}
\]

3) MTCNN+MOBILENET FACE OCCLUSION DETECTION

After the completion of light processing and color restoration, we used the MTCNN (Multi-task convolutional neural network) + MobileNet face occlusion detection algorithm in deep learning to detect face regions. This algorithm can accurately detect human faces in real-time under complex lighting, head deflection, and partial occlusion. In practical applications, this detection method has better performance and robustness than other detection methods.

MTCNN is a network structure composed of three-layer cascaded CNNs (P-Net, R-Net, and O-Net). Each layer of the network was calibrated using bounding box regression and non-maximum value suppression filter candidates. The latter layer is more refined than the previous layer, and the network parameters are trained in the form of multiple tasks to realize the process of face detection from simple to refined. In network training, the face classification problem is a two-class classification problem and the cross-entropy loss function is often used. The equations are:

\[
L_{i}^{\text{det}} = - (y_{i}^{\text{det}} \log(p_{i}) + (1 - y_{i}^{\text{det}})(1 - \log(p_{i}))) \tag{16}
\]

\[
L_{i}^{\text{box}} = ||y_{i}^{\text{box}} - y_{i}^{\text{box}}||_{2}^{2} \tag{17}
\]

\[
L_{i}^{\text{landmark}} = ||y_{i}^{\text{landmark}} - y_{i}^{\text{landmark}}||_{2}^{2} \tag{18}
\]

The MobileNet model is based on depth-wise decomposable convolutions that can decompose standard convolutions into depth-wise and point convolutions (1 \( \times \) 1 kernel). Depthwise convolution applies each kernel to each channel, while a 1 \( \times \) 1 convolution is used to combine the outputs of the channel convolutions.

4) ACTIVE AREA EXTRACTION

In practical applications, occlusions are often caused by the user wearing sunglasses, masks, or rotating shadows on their heads. Because the statistical color information of the forehead (such as sunglasses and masks) is significantly different from that of the user’s skin area, this study adopts the Gaussian skin color in the HSV color space. The model estimates skin color similarity and uses the OTSU algorithm to obtain the skin active area.

The equation for converting the image from RGB space to HSV space is consistent with Eqs. (5)–(7) for glare suppression.

We make \( X = (H, S, V)^{T} \); then the covariance matrix is

\[
C = \frac{1}{m} \sum_{j=1}^{m} [X_{j} - \frac{1}{n} \sum_{i=1}^{n} X] [X_{j} - \frac{1}{n} \sum_{i=1}^{n} X]^{T} \tag{19}
\]

After obtaining the covariance matrix, the skin color similarity of a pixel can be calculated as follows:

\[
p(x, y) = \exp[-\frac{1}{2}(X - \sum_{i=1}^{n} X) C^{-1}(X - \sum_{i=1}^{n} X)] \tag{20}
\]

Skin color similarity was evaluated for each pixel, and a skin color similarity image was obtained. According to the image, the OTSU binary method was used to obtain the active skin area.

B. FEATURE EXTRACTION AND ENHANCEMENT

1) FOUR-BRANCH FEATURE SELECTIVE EXTRACTION NETWORK

A feature extraction network has often been used to extract all the features in an image for subsequent classification or regression operations. If the features extracted by this feature extraction network are used for head pose estimation and regression to obtain a three-dimensional angle vector, the amount of computation and accuracy cannot meet the ideal requirements. Therefore, we propose a new multi-branch feature-selective extraction network structure, which is a more powerful network structure that can selectively extract the required features from a single RGB image for subsequent correlation operations. While improving accuracy, the amount of computation is reduced. In this study, the network had four branches, three of which were used to extract the independent discriminative features of the pitch, yaw, and roll angles. However, these three pose angles are not completely independent, and there is a certain connection between them. Therefore, many features in the image are composite features corresponding to multiple pose angles. The fourth branch of our network extracts these composite features and uses them in a subsequent multi-feature fusion network. The general structure is shown in Fig. 1.

2) SPATIAL CHANNEL DYNAMIC CONVOLUTION MODEL

In our network structure, the first convolutional layer of each branch selectively extracts features from the active area according to different pose angles, and the subsequent convolutional layers are based on the spatial sum of the feature maps obtained by the previous convolutional layer. Channel attention dynamically aggregates multiple parallel convolution kernels to form more expressive convolution kernels, resulting in more expressive deep-feature maps. The spatial-channel dynamic convolution is shown in Fig. 2.

First, we send the feature map \( U \) obtained by the first convolution to the channel attention module (CAM) to obtain two 1 \( \times \) 1 \( \times \) \( C \) feature maps through max pooling and average pooling, respectively; they are then sent to a shared two-layer neural network (MLP). The number of neurons in the first layer is \( C/r \); the ReLU activation function was used; and the
number of neurons in the second layer is \( C \). Then, the MLP features were added, and the sigmoid activation operation was performed to obtain the final channel attention feature \( U_c \). As shown in Eq. (21).

\[
U_c = \sigma(W_2\delta(W_1U_{\text{avg}}) + W_2\delta(W_1U_{\text{max}}))
\]  

Similarly, we sent the feature map \( U \) obtained by the first convolution into the spatial attention module (SAM) and performed channel-based max pooling and average pooling on this feature map, respectively, to obtain two \( H \times W \times 1 \) feature maps. The two feature maps were then spliced in the channel direction, and after a \( 7 \times 7 \times 2 \) convolution operation, the dimension was reduced to one channel. The spatial attention feature \( U_s \) was then generated through the sigmoid activation operation. As shown in Eq. (22).

\[
U_s = \sigma(f^{7 \times 7 \times 2}(F_{\text{avg}}, F_{\text{max}}))
\]  

We then passed the channel attention feature \( U_c \) through the fully connected layer, and the output of the activation layer obtained the channel influence factor feature \( \alpha_c \), as shown in Eq. (23).

\[
\alpha_c = \sigma(WU_c)
\]  

Multiply \( \alpha_c \) and each convolution kernel in parallel according to the corresponding channel to initially enhance the expressive ability of each convolution kernel. As shown in Eq. (24).

\[
\text{conv}_{nc}' = \alpha_{cc} \cdot \text{conv}_{nc}
\]  

The subscript \( nc \) represents the \( c \)th channel of the parallel \( n \)th convolution kernel and \( \alpha_{cc} \) represents the value of the \( c \)th channel of the channel influence factor feature \( \alpha_c \).

Similarly, we fully connected the spatial attention feature \( U_s \), and the output of the softmax layer provides the spatial influence factor feature \( \alpha_s \), as shown in Eq. (25).

\[
\alpha_s = \text{soft} \ \max(WU_s)
\]  

The corresponding multiplication operations are performed on \( \alpha_s \) and all parallel convolution kernels to obtain a more expressive convolution kernel, \( \text{conv} \). As shown in Eq. (26).

\[
\text{conv} = \sum_{n=1}^{n} \alpha_{sn} \cdot \text{conv}_{n'}
\]

where \( \alpha_{sn} \) represents the parameter in \( \alpha_s \) corresponding to the \( n \)th convolution kernel.

Finally, we used the previously obtained convolution kernel \( \text{conv} \) to obtain a more expressive deep feature map.

3) REGIONAL INFORMATION EXCHANGE NETWORK

In traditional CNNs, the convolution operation is typically performed in a local area, and its receptive field is small. Even if the receptive field is increased through the iteration of multiple convolutional layers, it remains difficult to capture a wide range of information and different connections between regions. To fully mine the potential semantic correlation between regions and learn to integrate the feature information of the potential correlation of other regions, we embed a RIE after each convolutional layer of each branch, which fully mines the potential correlation between feature map regions, and learns and fuses this correlation to enhance feature expression.

For the feature map \( U \) after any convolutional layer, the network first sent the feature map \( U \) to the horizontal layer region information exchange module. The module performs adaptive horizontal segmentation on the input feature map to obtain \( K_1 \) horizontal feature maps (this Adaptive horizontal segmentation will maximize the integrity of each horizontal layer feature to prepare for the association mining and learning fusion between subsequent horizontal layer features) and obtains the maximum value and the average value of each horizontal feature map according to the channel direction to obtain two \( W^s \frac{H}{K_1} \) feature matrices, each with a channel number of one: the maximum value matrix \( V_{\text{max}} \) and the average value matrix \( V_{\text{avg}} \). The maximum value matrix of the two horizontal feature maps that want to mine the correlation is connected back and forth, and the average value matrix is connected back and forth, where the subject feature map is placed in the front and the explored feature map is...
placed at the back. The correlation is then mined through a $3 \times 3 \times 2$ convolutional layer, and the obtained results are added to obtain the correlation matrix $V_{vrel}$ between the two feature maps. The correlation matrix is then averaged, indexed, and sigmoid activated to obtain the correlation coefficient $\alpha_h$. As shown in Eq. (27).

$$\alpha_{hi} = \sigma(\text{exp}(f^{3\times 3\times 2}(\text{Maxpool}(U_{hi}); \text{Maxpool}(U_{hi}))) + f^{3\times 3\times 2}(\text{Avgpool}(U_{hi}); \text{Avgpool}(U_{hi}))))$$

(27)

where $U_{hi}$ and $U_{hj}$ represent two horizontal feature maps. The process of association mining of any two horizontal layer features is shown in Fig.3.

The correlation matrix $V_{ah}$ between all horizontal feature maps can be obtained by performing a correlation analysis between any two horizontal feature maps.

$$V_{ah} = \begin{pmatrix} \alpha_{11} & \alpha_{12} & \cdots & \alpha_{1K_1} \\ \alpha_{21} & \alpha_{22} & \cdots & \alpha_{2K_1} \\ \cdots & \cdots & \cdots & \cdots \\ \alpha_{K_11} & \alpha_{K_12} & \cdots & \alpha_{K_1K_1} \end{pmatrix}$$

According to the correlation matrix, information exchange between the horizontal feature maps was performed to enhance the feature expression.

$$\begin{pmatrix} U_{H1} & U_{H2} & \cdots & U_{HK_1} \end{pmatrix} = \begin{pmatrix} \alpha_{11} & \alpha_{12} & \cdots & \alpha_{1K_1} \\ \alpha_{21} & \alpha_{22} & \cdots & \alpha_{2K_1} \\ \cdots & \cdots & \cdots & \cdots \\ \alpha_{K_11} & \alpha_{K_12} & \cdots & \alpha_{K_1K_1} \end{pmatrix} \begin{pmatrix} U_{h1} \\ U_{h2} \\ \cdots \\ U_{hK_1} \end{pmatrix}$$

(28)

The enhanced horizontal feature maps are stacked horizontally to obtain the feature map $U_H$ that has passed through the regional information exchange module of the horizontal layer.

Then $U_H$ was sent to the vertical layer area information exchange module. The module performs adaptive vertical segmentation on the input feature map to obtain $K_2$ vertical feature maps and obtains the maximum value and the average value of each vertical feature map according to the channel direction to obtain two $H^*W^*K_2$ feature matrices, each with a channel number of one: the maximum value matrix $V_{v_{max}}$ and the average value matrix $V_{v_{avg}}$. The maximum value matrix of the two vertical feature maps that want to mine the correlation is connected back and forth, and the average value matrix is connected back and forth, where the subject feature map is placed in the front, and the explored feature map is placed at the back. The correlation is then mined through a $3 \times 3 \times 2$ convolutional layer, and the obtained results are added to obtain the correlation matrix $V_{vrel}$ between the two feature maps. The correlation matrix is then averaged, indexed, and sigmoid activated to obtain the correlation coefficient $\alpha_v$. As shown in Eq. (29).

$$\alpha_{vj} = \sigma(\text{exp}(f^{3\times 3\times 2}(\text{Maxpool}(U_{vi}); \text{Maxpool}(U_{vj}))) + f^{3\times 3\times 2}(\text{Avgpool}(U_{vi}); \text{Avgpool}(U_{vj}))))$$

(29)

where $U_{vi}$ and $U_{vj}$ represent two vertical feature maps.

The correlation matrix $V_{av}$ between all vertical feature maps can be obtained by performing a correlation analysis between any two vertical feature maps.

$$V_{av} = \begin{pmatrix} \alpha_{11} & \alpha_{12} & \cdots & \alpha_{1K_2} \\ \alpha_{21} & \alpha_{22} & \cdots & \alpha_{2K_2} \\ \cdots & \cdots & \cdots & \cdots \\ \alpha_{K_21} & \alpha_{K_22} & \cdots & \alpha_{K_2K_2} \end{pmatrix}$$

According to the correlation matrix, information exchange between the vertical feature maps was performed to enhance the feature expression.

$$\begin{pmatrix} U_{V1} & U_{V2} & \cdots & U_{VK_2} \end{pmatrix} = \begin{pmatrix} \alpha_{11} & \alpha_{12} & \cdots & \alpha_{1K_2} \\ \alpha_{21} & \alpha_{22} & \cdots & \alpha_{2K_2} \\ \cdots & \cdots & \cdots & \cdots \\ \alpha_{K_21} & \alpha_{K_22} & \cdots & \alpha_{K_2K_2} \end{pmatrix} \begin{pmatrix} U_{v1} \\ U_{v2} \\ \cdots \\ U_{vK_2} \end{pmatrix}$$

(30)

Then, the enhanced vertical feature maps are stacked vertically to obtain the feature map $U_V$ that has passed through the vertical layer regional information exchange module.

Then, we sent $U_V$ to the rectangular area information exchange module. Similarly, the module performs adaptive
equiangular segmentation on the input feature map to obtain $K^2_3$ rectangular feature maps and obtains the maximum and average values for each rectangular feature map according to the channel direction. The value is extracted to obtain two $H \times W \over K^2_3$ feature matrices, each with a channel number of one: the maximum value matrix $V_{r \max}$ and the average value matrix $V_{r \avg}$. The maximum value matrix of the two rectangular feature maps that want to mine the correlation is connected before and after, and the average value matrix is connected before and after, in which the main feature map is placed in the front and the explored feature map is placed at the back. The correlation is then mined through a $3 \times 3 \times 2$ convolutional layer, and the obtained results are added to obtain the correlation matrix $V_{rrel}$ between the two feature maps. The correlation matrix is then averaged, indexed, and activated by sigmoid to obtain the correlation coefficient $\alpha_{r}$. As shown in Eq. (31).

$$\alpha_{r} = \sigma(\exp(\alpha_{f}^{avg})(Maxpool(U_{r}))) + \exp(\alpha_{f}^{avg})(Avgpool(U_{r})) \) (31)$$

where $U_{r1}$ and $U_{r2}$ represent two rectangular feature maps.

The correlation matrix $V_{r}$ between all rectangular feature maps can be obtained by performing a correlation analysis between any two rectangular feature maps.

$$V_{r} = \begin{pmatrix} \alpha_{11} & \alpha_{12} & \cdots & \alpha_{1K^2_3} \\ \alpha_{21} & \alpha_{22} & \cdots & \alpha_{2K^2_3} \\ \cdots & \cdots & \cdots & \cdots \\ \alpha_{K^2_31} & \alpha_{K^2_32} & \cdots & \alpha_{K^2_3K^2_3} \end{pmatrix}$$

According to the correlation matrix, information exchange between the rectangular feature maps was performed to enhance the feature expression.

$$\begin{pmatrix} U_{R1} \\ U_{R2} \\ \vdots \\ U_{RK^2_3} \end{pmatrix} = \begin{pmatrix} \alpha_{11} & \alpha_{12} & \cdots & \alpha_{1K^2_3} \\ \alpha_{21} & \alpha_{22} & \cdots & \alpha_{2K^2_3} \\ \cdots & \cdots & \cdots & \cdots \\ \alpha_{K^2_31} & \alpha_{K^2_32} & \cdots & \alpha_{K^2_3K^2_3} \end{pmatrix} \begin{pmatrix} U_{r1} \\ U_{r2} \\ \vdots \\ U_{rK^2_3} \end{pmatrix}$$

The enhanced rectangular feature maps are then stacked in the original order to obtain the feature map $U_{R}$ that has passed through the rectangular area information exchange module.

Finally, we sent the feature map $U_{R}$ to the channel area information exchange module for information exchange between channels. The specific steps are as follows.

First, we performed feature compression on the features of each channel so that the feature map of each channel becomes a real number, which has a global receptive field to some extent. As shown in Eq. (33).

$$P_{c} = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} u_{c}(i,j)$$

where $u_{c}$ represents the feature of channel $c$.

We then passed the squeezed result $P$ through a first full connection, activation, and a second full connection to explore the correlation between channels. The first full connection compresses the $c$ channels into $c/r$ channels to reduce the number of computations. The second full connection restores the $c$ channels. The correlation result was then passed through a sigmoid function to obtain $S$, as shown in Eq. (34).

$$S = \sigma(W_{2}(W_{1}(P))) \) (34)$$

The correlation result $S$ was then considered because the importance of each feature channel was weighted on a channel-by-channel basis by multiplication with the previous features, as shown in Eq. (35).

$$\tilde{u}_{c} = s_{c} \cdot u_{c} \) (35)$$

Finally, the feature map $U_{F}$ was obtained after passing through the channel area information exchange module.

After a series of operations, such as four-branch feature selective extraction, dynamic convolution of spatial channels, and regional information exchange, we finally obtained the feature map $U_{p}$ of pitch angle, feature map $U_{r}$ of yaw angle, feature map $U_{c}$ of roll angle, and composite feature map $U_{cc}$ corresponding to multiple angles.

**C. MULTIPLE FEATURE FUSION NETWORK**

The number of features in the independent feature map for each pose angle was small. To obtain a more accurate head pose angle, we performed multiple feature fusions between the independent feature map of each pose angle and the multi-angle composite feature map. From the composite feature map, we extracted more features useful for single pose angle estimation.

Here, we used the feature map $U_{p}$ of the pitch angle and the composite feature map $U_{cc}$ of multiple angles to introduce the multiple feature fusion network. Its process is shown in Fig.4.

1) **CHANNEL FEATURE FUSION NETWORK**

First, we sent the feature maps $U_{p}$ and $U_{cc}$ into the CAM module to obtain the channel attention feature $U_{pc}$ of the pitch angle feature map and the channel attention feature $U_{cc}$ of the multi-angle composite feature map.

Next, to fully learn the features useful for the pitch angle in the fusion composite feature map from the direction of the channel, we connected the channel attention features $U_{pc}$ and $U_{cc}$ according to the channel direction, and then went through the fully connected and activation layers to obtain the fusion channel attention feature $U_{pcc}$, as shown in Eq. (36).

$$U_{pcc} = \sigma(W(U_{pc}; U_{cc})) \) (36)$$

Finally, we connected the feature maps $U_{p}$ and $U_{cc}$ according to the channel direction, and weighed the fused channel attention feature $U_{pcc}$ to the connected feature map through multiplication of channel by channel to obtain the feature map $U_{P1}$ of the pitch angle after channel fusion.
2) SPATIAL FEATURE FUSION NETWORK

First, we sent feature maps A and B into the SAM module to obtain spatial attention feature C of the pitch angle feature map and spatial attention feature D of the multi-angle composite feature map.

Then, to fully learn and fuse the features useful for the pitch angle in the composite feature map from the spatial direction, we connected the spatial attention features $U_{ps}$ and $U_{pe}$ before and after, and then went through the $1 \times 1 \times 2$ convolutional layer and the activation layer to obtain the fusion spatial attention feature $U_{pcs}$, as shown in Eq. (37).

$$U_{pcs} = \sigma(f^{11+2}(U_{ps}; U_{cs}))$$  \hspace{1cm} (37)

Finally, we connected the feature maps $U_p$ and $U_c$ before and after and weighed the fused channel attention feature $U_{pcs}$ to the connected feature map by spatial block by multiplication to obtain the feature map $U_{p2}$ of the pitch angle after spatial fusion.

3) PIXEL FEATURE FUSION NETWORK

Learning and merging the features useful for the pitch angle in the composite feature map from the space and channel directions alone cannot obtain the perfect feature expression. To further investigate the useful features in the composite feature map, we sent the feature map $U_{p1}$ of the pitch angle after channel fusion and the feature map $U_{p2}$ of the pitch angle after spatial fusion into the pixel feature fusion network for pixel-level feature fusion.

The network first evaluates the features of each pixel level of the feature maps, $U_{p1}$ and $U_{p2}$, and the evaluation function is expressed as follows:

$$A_{ij} = \alpha \cdot \delta(W^*U(i, j) + b) + \beta \cdot \sum_{m=1}^{c} \left(p_m - \frac{1}{c} \sum_{n=1}^{c} p_n \right)^2$$  \hspace{1cm} (38)

This evaluation function can fully evaluate the effect of each pixel-level feature on the discrimination of pose angles. Here, W is the weight to be learned, b is the bias, p represents the number of pixel-level features, and $\alpha$ and $\beta$ are the parameters that change adaptively according to pixel-level features. $i$ and $j$ represent the position of the pixel feature in the feature map, and A is the evaluation result of pixel features; by evaluating all pixel-level features in the two feature maps, $U_{p1}$ and $U_{p2}$, we can obtain two evaluation result maps: $A_{p1}$ and $A_{p2}$.

Next, we expanded the feature maps $U_{p1}$ and $U_{p2}$ in a two-dimensional plane and concatenated them to obtain the following matrix:

$$U_{p12} = [I, C]$$

Subsequently, according to the evaluation results, we fused the n-dimensional feature vectors that can best express the corresponding pose angle and denoted it as $U_F$. The relationship between $U_F$ and $U_{p12}$ is:

$$U_F = V_{p1}V_{p2}U_{p12}$$  \hspace{1cm} (39)

where $V_{p1} \in R^{n \times r}$ is obtained from the full connection and activation of the evaluation result $A_{p1}$ of the feature map $U_{p1}$, as shown in Eq. (40).

$$V_{p1} = \delta(W_{p1}A_{p1})$$  \hspace{1cm} (40)

$V_{p2} \in R^{r \times t}$ is obtained from the full connection and activation of the evaluation result $A_{p2}$ of the feature map $U_{p2}$, as shown in Eq. (41).

$$V_{p2} = \delta(W_{p2}A_{p2})$$  \hspace{1cm} (41)

Our method is based on the fusion of channel and spatial features. According to the evaluation coefficient of the pixel-level features, the neural network is trained to generate proportional weights for deeper feature fusion, and the most expressive feature vector is obtained, which greatly improves the follow-up accuracy of head pose estimation.

D. HEAD POSE ESTIMATION

After multiple feature fusion networks, we obtained the feature vectors $U_P$, $U_Y$, and $U_R$ corresponding to each pose angle.
and then performed deep feature aggregation regression on the feature vectors corresponding to each pose angle to obtain the head pose estimation result.

### III. EXPERIMENTAL RESULTS AND DISCUSSION

In this section, we introduce the entire experimental process, which is divided into four parts, as shown in Fig. 5. First, we provide indicators for evaluating the model. Subsequently, we introduce the basic experimental settings. We then compare our model with the advanced model and present the results from testing our model using a self-built real complex environment dataset. Finally, we describe the ablation research that was conducted on the key innovation part of our model.

#### A. EVALUATION STANDARD

In this study, we used the mean absolute error (MAE) as the evaluation standard for head pose estimation. Assume that the trained face image is, $X = \{x_n | n = 1, 2, \ldots, N\}$ and the head-pose vector $y_n$ corresponds to the face image. Each pose vector is $y_n$, which is a three-dimensional vector whose components correspond to the yaw, pitch, and roll angles. The head pose vector predicted by our algorithm is $y'_n$, and MAE is defined as follows:

$$\text{MAE} = \frac{1}{N} \sum_{n=1}^{N} ||y'_n - y_n||_1$$ (42)

At the same time, we update each parameter according to the mean absolute error (MAE) and minimize the prediction, so that the predicted value generated by the model is as close to the real value as possible.

#### B. EXPERIMENT SETTING

1) **INTRODUCTION TO DATASETS AND TRAINING AND TESTING PROTOCOLS**

Three popular head pose estimation datasets, 300 W-Lp [24], AFLW2000 [24], and BIWI [34], and a self-built real complex environment head pose estimation dataset RCE were used in the experiment.

The 300 W-Lp datasets [24] are based on the 300 W datasets [35], using facial contours and 3D image grids to generate 61,225 large pose samples, and using flips to further expand to 122,450 samples, a synthetic dataset aligned with 68 landmarks, which is called the 300 W (300 W lp) with a large pose.

The AFLW2000 dataset [24] provides ground-truth 3D faces and the corresponding 68 landmarks for the first 2,000 images of the AFLW2000 dataset [36]. The faces in the dataset had large pose variations under various illumination conditions and expressions.

The BIWI dataset [34] contains 24 videos of 20 subjects in a controlled environment. There are approximately 15,000 frames in the dataset. In addition to the RGB frames, the dataset provides a depth image for each frame.

The RCE dataset contains 20 subjects (10 men, 10 women) in the laboratory with video clips of head pose changes, and each subject was photographed wearing a mask in a complex lighting environment.

For training and evaluation on these datasets, we followed the following three protocols.

Protocol 1: Train on the synthetic 300 W-LP dataset and test two real datasets, AFLW2000 and BIWI, simultaneously. Simultaneously, when evaluating the BIWI dataset, we do not use tracking and only consider using MTCNN face detection samples whose rotation angles are in the range of $[-99^\circ, +99^\circ]$. We used this protocol for comparison with the most advanced landmark-based head-pose estimation methods.

Protocol 2: We divided the videos of the BIWI dataset into multiple categories by subject’s gender, ethnicity, and other characteristics, and used a portion (16 videos) of each category for training and the remaining 8 videos for testing. MTCNN uses experience tracking technology to detect faces in the BIWI dataset, avoiding the failure of face detection. This protocol is used by several pose estimation methods such as RGB, depth, and time, whereas our method uses only a single RGB frame.

Protocol 3: The self-built RCE dataset is classified according to the characteristics of subjects such as gender, and some videos (14 videos) in each category are used for training, and the rest (6 videos) are used for testing.

2) **EXPERIMENT PLATFORM**

In this study, all experiments were conducted on a Windows 10 operating system, NVIDIA GeForce RTX 2080Ti (11 GB video memory), and Intel Core i9-9900K (16 GB memory) platform. The software platform was Python 3.7.0 and the TensorFlow framework.
C. COMPARATIVE TEST

Under Protocol 1 conditions, we compared our network with state-of-the-art unimodal pose estimation methods, and our method achieved good results. Other pose estimation methods mainly include the following: Dlib [26] is a standard face library that includes various technologies such as face detection. KEPLER [33] used Google’s improved network to predict facial keypoints and poses, and uses rough pose monitoring to improve landmark detection. 3DDFA [24] evaluated the parameters related to the shape, converted the head into a dense 3D model and used a CNN to fit the 3D model to the RGB image. This method can effectively handle occlusion problems. The FAN obtains multiscale information by merging features multiple times across layers. Hope Net [30] used Res Net to separate yaw, roll, and pitch, and used the MSE and cross-entropy loss function to train the network. FSA-Net [37] used fine-grained structure mapping and a scoring function to screen important features, and used capsule network feature aggregation to estimate the head pose. Img2pose [40] enables real-time, 6DoF 3D face pose estimation without the need for face detection or landmark localization. MNN [39] is a multitask image head-pose estimation method based on deep learning. We took advantage of the strong dependencies between the facial pose, alignment, and visibility to generate an optimally executed model.

Under Protocol 1 conditions, the 300 W-LP dataset was used to train the pose estimation method. Tables 1 and 2 compare our model with the latest methods on the AFLW2000 and BIWI datasets. MAE was used as the evaluation index. In Protocol 1, the training dataset was synthetic, whereas the test dataset was real. The unmarked method can better adapt to the domain differences between training and testing. Therefore, on the AFLW2000 and BIWI datasets, landmark-free methods (img2pose, MNN, and Ours) were superior to landmark-based methods. Figure 6 compares our model with img2pose and MNN using several examples. However, from the perspective of error data, our model has improved accuracy of head pose estimation when compared with other models, and its performance is better than that of the other models. However, looking at the example in Fig. 6, our head pose estimation results are the most robust.

Under Protocol 2 conditions, we compared our model with advanced head-pose estimation methods that use multiple types of information. These methods are as follows: DeepHeadPose [27] focuses on low-resolution RGB-D images and combines classification and regression methods with CNN to estimate the head pose. Martin et al. [11] estimated the head pose from a depth image by constructing a 3D head model. VGG16 (RGB) and VGG16 + RNN (RGB + Time) were proposed by Gu et al. [38]. VGG16 + RNN (RGB + Time) is a multimodal head-pose estimation method. By analyzing Bayesian filters, they attempted multiple possibilities for CNN and RNN mergers.

In addition to providing color information, the BIWI dataset provides depth and time information. Table 3 shows the performance of the methods using different modes. The RGB-based group only uses a single RGB frame, while RGB + Depth and RGB + Time use color, depth, and time information, respectively. Our method only uses a single RGB frame, and the error data show that the performance is better than that of other methods based on RGB frames. Our network did not use a multimodal model, but the effect was not significantly different from that of the multimodal

### TABLE 1. Comparisons with the state-of-the-art methods on the AFLW2000 dataset. All are trained on the 300W-LP dataset.

| Model       | Yaw  | Pitch | Roll | MAE  |
|-------------|------|-------|------|------|
| Dlib (68 points) | 22.80 | 14.10 | 10.30 | 15.73 |
| 3DDFA       | 5.71 | 8.12  | 8.34 | 7.39 |
| FAN (12 points) | 6.58 | 11.56 | 8.58 | 8.90 |
| Hope-Net (α=1) | 6.85 | 6.64  | 5.69 | 6.39 |
| Hope-Net (α=2) | 6.42 | 6.61  | 5.46 | 6.16 |
| FSA-Net     | 4.58 | 6.09  | 4.71 | 5.12 |
| Img2pose    | 3.43 | 5.03  | 3.28 | 3.91 |
| MNN         | 3.36 | 4.68  | 3.47 | 3.84 |
| Ours        | 3.24 | 4.37  | 3.41 | 3.67 |

### TABLE 2. Comparisons with the state-of-the-art methods on the BIWI dataset. All are trained on the 300W-LP dataset.

| Model       | Yaw  | Pitch | Roll | MAE  |
|-------------|------|-------|------|------|
| Dlib (68 points) | 16.65 | 13.92 | 6.31 | 12.29 |
| KEPLER      | 8.62 | 17.14 | 16.67 | 14.14 |
| 3DDFA       | 35.65 | 13.31 | 8.75 | 19.24 |
| FAN (12 points) | 8.62 | 7.37  | 7.49 | 7.83 |
| Hope-Net (α=1) | 4.79 | 6.52  | 3.41 | 4.91 |
| Hope-Net (α=2) | 5.04 | 6.83  | 3.45 | 5.11 |
| FSA-Net     | 4.21 | 4.87  | 2.95 | 4.01 |
| Img2pose    | 4.57 | 3.55  | 3.24 | 3.79 |
| MNN         | 3.96 | 4.62  | 2.38 | 3.65 |
| Ours        | 3.34 | 3.45  | 3.29 | 3.36 |
TABLE 3. Comparisons with the-state-of-art methods in different modes on the BIWI dataset. 70% of videos are used for training (16 videos) and 30% for testing (8 videos).

| Model       | Yaw | Pitch | Roll | MAE  |
|-------------|-----|-------|------|------|
| VGG16       | 3.93| 4.02  | 3.05 | 3.67 |
| Ours        | 3.24| 3.28  | 3.31 | 3.27 |
| RGB + Depth |     |       |      |      |
| DeepHeadPose| 5.36| 4.72  | -    | -    |
| Martin      | 3.58| 2.61  | 2.63 | 2.94 |
| RGB + Time  |     |       |      |      |
| VGG16+RNN   | 3.26| 3.39  | 2.71 | 3.12 |

TABLE 4. Different models perform head pose estimation tests in real complex environments on self-built RCE datasets.

| Model  | Yaw | Pitch | Roll | MAE  |
|--------|-----|-------|------|------|
| Img2pose| 7.31| 7.53  | 7.42 | 7.42 |
| MNN    | 5.71| 6.54  | 5.62 | 5.96 |
| Ours   | 4.75| 5.09  | 4.86 | 4.90 |

Complex lighting and partial occlusions are common in real images or videos. To further verify the robustness of our model in such situations, we tested it in a real complex environment. That is, the conditions of Protocol 3 used the self-built RCE dataset for training and testing. Simultaneously, the Img2pose and MNN that performed well in the comparison experiment were used as references.

Table 4 reports the error of our model’s head pose estimation test in the real and complex environment. From the data, it can be concluded that our model can effectively handle images under complex lighting and partial occlusion conditions. It has better robustness in real complex environments. Figure 7 shows the excellent performance of our model in a real complex environment.

D. ABBLATION EXPERIMENT

1) ABBLATION STUDY OF SPATIAL CHANNEL DYNAMIC CONVOLUTION MODEL

In order to further prove the influence of the spatial channel dynamic convolution model on head pose estimation, we conduct head pose estimation experiments in two cases, the general static convolution and the spatial channel dynamic convolution, and the thermal schematic diagram of feature extraction in two cases is shown in Fig.8.

As shown in Fig.8, it is not difficult to find that the dynamic convolution of the spatial channel can extract richer features than the general static convolution through the heat map. At the same time, combined with the data in Table 5, we can conclude that the spatial channel dynamic convolution can improve the accuracy of head pose estimation by extracting more abundant feature.

2) ABBLATION STUDY OF REGIONAL INFORMATION EXCHANGE NETWORK

To study the influence of the RIEN on head pose estimation, we conducted the following ablation experiments and performed head pose estimation under different RIEN conditions. Simultaneously, we conducted a comparative analysis of the correlation of nine random features before and after regional information exchange.

In this study, the RIEN was divided into four modules: the horizontal layer regional information exchange module, vertical layer regional information exchange module, rectangular region information exchange module, and channel region information exchange module. Conditions 0, 1, 2, and 3 in Table 6 refer to the head pose estimation for only zero, one, two, and three regional information exchange modules, and Condition 4 refers to the situation of the RIEN in this study.

Head pose estimation was performed using the following (four regional information exchange modules): The MAE of the data result is the average of all possible cases. For example, the MAE in Condition 1 refers to the average value under the four cases (only the horizontal layer area information exchange module, only the vertical layer area information exchange module, only the rectangular area information exchange module, only the channel area information exchange module).
FIGURE 9. Comparison of feature correlation maps of nine random features before and after regional information exchange.

TABLE 5. Effects of spatial channel dynamic convolution models on head pose estimation under different protocols.

| Convolution method | Protocol1 (AFLW2000) MAE | Protocol1 (BIWI) MAE | Protocol2 MAE | Protocol3 MAE |
|--------------------|--------------------------|----------------------|---------------|---------------|
| Static             | 3.94                     | 3.61                 | 3.68          | 5.83          |
| SCDC               | 3.67                     | 3.36                 | 3.27          | 4.90          |

TABLE 6. Effects of regional information exchange networks under different conditions on head pose estimation.

| RIEN condition | Protocol1 (AFLW2000) MAE | Protocol1 (BIWI) MAE | Protocol2 MAE | Protocol3 MAE |
|----------------|--------------------------|----------------------|---------------|---------------|
| Condition 0    | 4.26                     | 4.12                 | 4.34          | 6.98          |
| Condition 1    | 4.07                     | 3.97                 | 4.13          | 6.32          |
| Condition 2    | 3.91                     | 3.74                 | 3.69          | 5.94          |
| Condition 3    | 3.72                     | 3.47                 | 3.48          | 5.46          |
| Condition 4    | 3.67                     | 3.36                 | 3.27          | 4.90          |

TABLE 7. Effects of different branches of feature extraction networks on head pose estimation.

| Number of branches | Protocol1 (AFLW2000) MAE | Protocol1 (BIWI) MAE | Protocol2 MAE | Protocol3 MAE |
|--------------------|--------------------------|----------------------|---------------|---------------|
| 1                  | 5.37                     | 5.03                 | 4.94          | 7.61          |
| 3                  | 4.01                     | 3.78                 | 3.62          | 5.85          |
| 4                  | 3.67                     | 3.36                 | 3.27          | 4.90          |

exchange module, and only the channel area information exchange module). Figure 9 shows the correlation of the nine segmented local features of the random regions before and after the regional information exchange. Table 6 shows that with an increase in the number of regional information exchange modules, the MAE gradually decreased. Figure 9 illustrates that, after regional information exchange, the connections between local features are closer. This indicates that the regional information exchange model can improve the accuracy of head pose estimation by enhancing the latent-connection enhancement features between the regions.

3) ABLATION STUDY OF FOUR-BRANCH FEATURE SELECTIVE EXTRACTION NETWORK

To further demonstrate the effectiveness of our proposed four-branch feature-selective extraction network structure for improving the accuracy of head pose estimation, we conducted the following ablation experiments, where we separately performed a single-branch (estimating three angles from global features). Under the network structure of three branches (decomposing the global feature into three parts, and then estimating three angles, respectively) and four branches (the network structure of this study), head
pose estimation was conducted under different protocol conditions.

Table 7 presents the head-pose estimation results for different protocols. The results show that the performance of our network structure is very impressive compared to other network structures, and it has made a significant contribution to improving the accuracy of head pose estimation.

4) ABSTRACTION STUDY OF MULTIPLE FEATURE FUSION NETWORK

To further explore the importance of multiple feature fusion networks for head pose estimation, we performed the following ablation experiments: channel space, and pixel fusion (triple fusion). Table 8 presents the head-pose estimation results under different protocols for the ablation experiment. From the experimental data, it is not difficult to determine that with an increase in the number of network fusions, the network performance is continuously enhanced, and the accuracy of the head-pose estimation gradually improves.

| Fusion times | Protocol1 (AFLW2000) | Protocol1 (BIWI) | Protocol2 | Protocol3 |
|--------------|-----------------------|-----------------|-----------|-----------|
| 0            | 5.19                  | 4.89            | 4.24      | 6.65      |
| 1            | 4.81                  | 4.43            | 4.08      | 6.47      |
| 2            | 4.58                  | 4.12            | 3.89      | 6.24      |
| 3            | 3.67                  | 3.36            | 3.27      | 4.90      |

IV. CONCLUSION AND FUTURE WORK

In this study, we investigated how to improve the accuracy of head-pose estimation in complex environments, such as those involving mask occlusion. Many experiments demonstrated that our algorithm can effectively deal with complex environments, such as those involving partial occlusion, improving the accuracy of head pose estimation. We believe that this study makes a significant contribution to the literature. We used a four-branch feature selective extraction network structure to extract three independent discriminative features of pose angles and composite features corresponding to multiple pose angles. This structure greatly reduces the amount of computation by reducing the dimension of high-dimensional features, improving the estimation accuracy. In terms of feature extraction and enhancement, we employed an improved SCDC and a RIEN to learn the potential semantic correlations between fusion regions to enhance feature representation to effectively solve the problem of less effective features in the case of occlusion. In terms of feature fusion, the multi-feature fusion network proposed in this study fuses the independent discriminative features of each pose angle and the composite features from the three directions of channel, space, and pixel to obtain the most accurate feature expression for each pose angle. This significantly improves the accuracy of the subsequent head-pose estimation. Furthermore, we believe that this study will be of interest to practitioners of head-pose estimation because accurate head-pose estimation in complex environments on mobile devices is an important development direction. The algorithm proposed in this study can effectively solve problems related to the estimation of head pose in complex environments, such as those associated with a lot of computation, few effective features, and low computational accuracy in complex environments.

However, our model has certain limitations. Although our multi-feature fusion network could obtain very accurate feature representations for each pose angle, there are still certain features of other pose angles in these feature representations, resulting in certain errors. Simultaneously, with the development of the field of artificial intelligence, the requirements for the accuracy of head-pose estimation in real and complex environments are increasing. Therefore, in future work, we will further optimize the multi-feature fusion network to reduce this error. Furthermore, we intend to classify the general direction of each pose angle first and then use our algorithm for regression, hoping to further improve the calculation speed and accuracy. Finally, we intend to use multimodal information, such as depth and temporal information, to estimate the head pose to improve the performance of the network.

AUTHOR CONTRIBUTIONS

Bin-Yu Wang conceived and initialized the research, conceived the algorithms, and designed the experiments; Kai Xie reviewed the paper; Chang Wen conducted comparative experiments; Sheng-Tao He and Jian-Biao He checked the spelling and provided advice.

ACKNOWLEDGMENT

(Bin-Yu Wang, Kai Xie, Sheng-Tao He, and Chang Wen contributed equally to this work.)

REFERENCES

[1] S. S. Mukherjee and N. M. Robertson, “Deep head pose: Gaze-direction estimation in multimodal video,” IEEE Trans. Multimedia, vol. 17, no. 11, pp. 2094–2107, Nov. 2015.
[2] Y. Lang, W. Liang, F. Xu, Y. Zhao, and L.-F. Yu, “Synthesizing personalized training programs for improving driving habits via virtual reality,” in Proc. IEEE Conf. Virtual Reality, Mar. 2018, pp. 297–304.
[3] S. Qi, W. Wang, B. Jia, J. Shen, and S.-C. Zhu, “Learning human-object interactions by graph parsing neural networks,” in Proc. ECCV, 2018, pp. 401–417.
[4] C. Li, W. Liang, C. Quigley, Y. Zhao, and L.-F. Yu, “Earthquake safety training through virtual drills,” IEEE Trans. Vis. Comput. Graphics, vol. 23, no. 4, pp. 1275–1284, Apr. 2017.
[5] W. Liang, J. Liu, Y. Lang, B. Ning, and L.-F. Yu, “Functional workspace optimization via learning personal preferences from virtual experiences,” IEEE Trans. Vis. Comput. Graphics, vol. 25, no. 5, pp. 1836–1845, May 2019.
[6] S. Sheikhi and J.-M. Odobez, “Combining dynamic head pose–gaze mapping with the robot conversational state for attention recognition in human–robot interactions,” Pattern Recognit. Lett., vol. 66, pp. 81–90, Nov. 2015.

[7] A. Doshi and M. M. Trivedi, “Head and eye gaze dynamics during visual attention shifts in complex environments,” J. Vis., vol. 12, no. 2, p. 9, Feb. 2012.

[8] R. Diaz-Chito, J. M. D. Rincón, A. Hernández-Sabaté, and D. Gil, “Continuous head pose estimation using manifold subspace embedding and multivariate regression,” IEEE Access, vol. 6, pp. 18325–18334, 2018, doi: 10.1109/ACCESS.2018.2817252.

[9] L. Liang, F. Wen, and J. Sun, “Face alignment via component-based discriminative search,” in Computer Vision—ECCV 2008. Berlin, Germany: Springer, 2012, doi: 10.1007/978-3-540-88688-4_6.

[10] W. Wang, “Face recognition and expression analysis based on AAM,” M.S. thesis, Qingdao Univ. Sci. Technol., Qingdao, China, 2014.

[11] M. Martin, F. Van De Camp, and R. Stiefelhagen, “Real time head model creation and head pose estimation on consumer depth cameras,” in Proc. 2nd Int. Conf. 3D Vis., Tokyo, Japan, Dec. 2014, pp. 641–648.

[12] V. Drouard, R. Horaud, A. Deleforge, S. Ba, and G. Evangelidis, “Robust head-pose estimation based on partially-late-mixture of linear regressions,” IEEE Trans. Image Process., vol. 26, no. 3, pp. 1428–1440, Mar. 2017.

[13] J.-F. Chang, A. T. Tran, T. Hassner, I. Masi, R. Nevatia, and G. Medioni, “FacePoseNet: Making a case for landmark-free face alignment,” in Proc. IEEE Int. Conf. Comput. Vis. Workshops (CVPR Workshops), Oct. 2017, pp. 1599–1608.

[14] G. Zhao, “Research on key technologies of vision based wide-angle head posture tracking,” M.S. thesis, Zhejiang Univ., Hangzhou, China, 2009.

[15] R. Ranjan, V. M. Patel, and R. Chellappa, “HyperFace: A deep multitask learning framework for face detection, landmark localization, pose estimation, and gender recognition,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 41, no. 1, pp. 121–135, Jan. 2019.

[16] Y.-J. Huang and F.-Z. Li, “Isospectral manifold learning algorithm,” J. Softw., vol. 24, no. 11, pp. 2656–2666, Jan. 2014.

[17] L. Liang, R. Xiao, F. Wen, and J. Sun, “Face alignment via component-based discriminative search,” in Computer Vision—ECCV 2008 (Lecture Notes in Computer Science), vol. 5303, D. Forsyth, P. Torr, and A. Zisserman, Eds. Berlin, Germany: Springer, 2008, doi: 10.1007/978-3-540-88688-4_6.

[18] I. Matthews and S. Baker, “Active appearance models revisited,” Int. J. Comput. Vis., vol. 60, no. 2, pp. 135–164, Nov. 2004.

[19] G. P. Meyer, S. Gupta, I. Frosio, D. Reddy, and J. Kautz, “Robust model-based 3D head pose estimation,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Dec. 2015, pp. 3649–3657.

[20] S. Yan, S. Shan, X. Chen, and W. Gao, “Locally assembled binary (LAB) feature with feature-centric cascade for fast and accurate face detection,” in Proc. CVPR, Jun. 2008, pp. 1–7.

[21] G. Fanelli, T. Weise, J. Gall, and L. Van Gool, “Real time head pose estimation from consumer depth cameras,” in Pattern Recognition (Lecture Notes in Computer Science), vol. 6835, R. Mester and M. Felsberg, Eds. Berlin, Germany: Springer, 2011, doi: 10.1007/978-3-642-23123-0_11.

[22] Y. Tang, Z. Sun, and T. Tan, “Face pose estimation based on integral slice features of single depth images,” in Proc. ACPR, Nov. 2011, pp. 530–534.

[23] A. Bulat and G. Tzimiropoulos, “How far are we from solving the 2D & 3D face alignment problem? (and a dataset of 230,000 3D facial Landmarks),” in Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCV Workshops), Nov. 2011, pp. 2144–2151.

[24] B. Albiero, X. Chen, Y. Yin, G. Pang, and T. Hassner, “img2pose: Face alignment and detection via 6DoF face pose estimation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2021, pp. 7617–7627.
SHENG-TAO HE was born in Hubei, China, in 2000. He joined the National Experimental Electrical and Electronic Education Demonstration Center, in 2020. He is currently an Assistant Researcher with the Yangtze University, Jingzhou, China. His current research interests include digital signal processing, deep research learning, image processing, and machine learning.

CHANG WEN received the B.S. degree in computer science from the Naval University of Engineering, Wuhan, China, in 2002, and the M.S. degree in computer science from Yangtze University, Jingzhou, China, in 2008. She is currently an Assistant Professor with the School of Computer Science, Yangtze University. She currently works in the field of image processing and signal processing.

JIAN-BIAO HE received the B.S. and M.S. degrees from the Huazhong University of Science and Technology, Wuhan, China, in 1986 and 1989, respectively. He is currently an Associate Professor with the School of Computer Science and Engineering, Central South University. His research interests include artificial intelligence, the Internet of Things, pattern recognition, mobile robots, and cloud computing.

***