# Accurate Head Pose Estimation-Based SO(3) and Orientation Tokens for Driver Distraction Detection

Xiong Zhao[1], Sarina Sulaiman[2], Wong Yee Leng*[3]
Faculty of Computing, Universiti Teknologi Malaysia, 81310 Johor Bahru, Johor, Malaysia[1,2,3]
Yunnan College of Business Management, Kunming, China[1]

***Abstract*—Driver distraction is an important cause of traffic accidents. By identifying and analyzing the driver's head posture through monitor images, the driver's mental state can be effectively judged, and early warnings or reminders can be given to reduce traffic accidents. We propose a novel dual-branch network named TokenFOE that combines Convolutional Neural Networks (CNN) and Transformer. The CNN branch uses an Multilayer Perceptron (MLP) to infer the image features from the backbone, then generating a rotation matrix based on SO(3) to represent head posture. The Dimension Adaptive Transformer branch uses learnable tokens to represent the head orientation of 9 categories. Integrate the losses of both branches for training, ultimately obtaining accurate head pose estimation results. The training dataset uses 300W-LP, and the quantatitive testing datasets are AFLW-2000 and BIWI. The experiment results show that the Mean Absolute Error is improved by 21.2% and 9.4% compared to the original SOTA model on the two datasets, and the Mean Absolute Error of Vectors is improved by 19.2% and 10.2%, respectively. Based on the model output and calibrated through the camera adapter module, we present the qualitative results on the largest driver distraction detection dataset currently available, the 100-driver dataset, robust and accurate detection results were achieved for four different camera perspectives in two modalities, RGB and Near Infrared. Additionally, the ablation study shows that the model inference speed (21 to 75fps) can be used for real-time detection.**

***Keywords*—Head pose; driver distraction detection; rotation matrix; token; transformer**

## I. Introduction

Distracted driving is defined as any behavior in which drivers focus their attention on activities unrelated to driving tasks [1]. The group of drivers who drive for a long time, such as long-distance bus drivers, full-time ride-hailing drivers, Taxi and truck drivers, are more prone to distraction, which is a significant safety hazard. When the driver is distracted, there is usually a phenomenon of abnormal head posture deviating from the direction of vehicle travel. Real time detection can be carried out using the existing monitoring device video in the terminal [2], and timely reminders can be given to the driver when abnormalities occur, which can effectively improve driving safety.

Face Orientation Estimation (FOE) also called Head pose estimation (HPE), is a challenging task in human-computer interaction [3], Human posture detection [4] and attention detection [5], [6]. The FOE problem can be conceptualized as a rotational dynamics challenge involving a 3D rigid body in space. To describe this 3D rotation, various mathematical techniques are available, including Euler Angles, Quaternions, Axis-Angle representations, Rotation Matrices, and Lie algebra. Each of these methods possesses unique strengths and limitations in terms of accuracy, efficiency, and ease of implementation. These advancements are attributed to the utilization of additional data sources such as facial landmark information [7], [8], [9], RGB-depth data [10], multi-task learning approaches [7], [11], and alternative parameterizations for orientation representation [11], [12], [13].

Traditional methods rely on manually extracting facial landmarks, such as the eyes, nose, and mouth, to estimate head pose. These methods are often sensitive to noise and variations in facial features. Deep learning has revolutionized FOE by enabling the direct prediction of head pose from raw images or videos. Deep learning models can learn complex patterns in facial data, leading to more robust and accurate head pose estimation, these methods use extra annotation to help FOE tasks, which usually could get good accuracy. Other methods trying to only use orientation labels, such as HopeNet [14], TriNet [11], FSA-Net [15], and TokenHPE [16], these methods also have much progress but slightly worse than aforementioned methods.

Some methods also try to use transformer architecture, such as TokenPose [17] uses a leaning token to represent a human pose, then regression the whole body key points. Following the same inspiration, TokenHPE [16] tried to use learning tokens to solve the HPE problem, they divided all the orientations into 9 or 11 regions, and every region set a token to predict.

Following these works, we introduce our work, TokenFOE, and try to estimate Head posture by a dual-branch network based SO(3) and Orientation Tokens for driver DDD (Driver Distraction Detection) task. The contributions are summarized as follows:

1) We proposed TokenFOE that adopts the dual-branch network with a joint loss that combines the CNN branch and Transformer branch, we also introduced DAT (Dimension Adaptive Transformer) architecture that could be suitable for any backbone.
2) Extensive experiments on AFLW2000 [18] and BIWI [19] show we achieved new SOTA, especially exceeding other Extra Annotation Free (EAF) methods by a wide margin.
3) To the best of our knowledge, it is the first time that SO(3) and attention mechanism are combined as a dual-branch network in a DDD task.

The paper is organized as follows: Section II introduces some related work in this field, and Section III describes in detail the

model structure of TokenFOE and provides specific important formulas, loss functions, and evaluation metrics. In Section IV, we first introduces the datasets used in this article, and then provides detailed experimental results to demonstrate the effectiveness and robustness of the method. Detailed ablation experimental results are also provided for important parameters. Finally, the limitations and conclusions of this method are discussed in Sections V and VI.

## II. Related Works

Prior to the advent of deep learning, estimating facial orientation or pose from RGB images without depth information presented significant challenges. This task involved dealing with a vast representation space encompassing diverse head poses. Over the past decades, numerous methodologies have been proposed to address this challenging problem. In this section, we present a concise overview of the FOE problem.

### A. Driver Distracted Detection Approaches

The accuracy of FOE has been greatly improved, and applications based on estimation results have also received more attention and research. Li et al. [20] try to review this research field use electroencephalography. Mou et al. [21] proposed a dual-channel network to try to solve the DDD problem, but it just tested on an early simulated small dataset, and the model's true generalization performance is difficult to verify, with few types of distraction detection. 100-driver [22] is a large-scale, diverse posture-based distracted diver dataset, with more than 470K images taken by 4 cameras observing 100 drivers over 79 hours from 5 vehicles. Gebert P. et al. use an attention module integrated into the network for adaptive feature extraction. Li et al. introduced an efficient system based on a Transformer to detect driving behavior.

### B. Extra Information-utilized HPE Approaches

PRNet [23] predicts 2D UV position maps that encode 3D points and utilizes the connectivity of the Basel Face Model (BFM) mesh to construct face models. Research [24] approach combines coarse and fine regression outputs within a deep neural network framework. Meanwhile, SynergyNet [7] explores a synergistic learning process that leverages both 3D Morphable Models (3DMM) and 3D facial landmarks to predict the entire 3D facial geometry, achieving highly accurate results. In contrast, [25] proposes a method that does not require training with head pose labels, instead relying on matching key points between a reconstructed 3D face model and the 2D input image. However, all prediction methods that rely on key points are significantly influenced by the quality of the input image.

Unlike traditional methods that rely on pre-labeled head poses for training, method [25] estimates pose by aligning a reconstructed 3D face model with the 2D input image, bypassing the need for explicit pose labels. However, this approach has a limitation: its accuracy can be affected by variations in image quality, as it depends on precise keypoint detection.

### C. Extra Annotation Free HPE Approaches

TokenHPE [16] was a Transformer-based method that is critically aware of minority relationships among facial parts. This approach specifically focuses on learning the intricate relationships between different facial components. To achieve this, they introduce several orientation tokens that are designed to explicitly encode the fundamental orientation regions of the face. Furthermore, they devise a novel token-guided multi-loss function that serves as a guide for the orientation tokens, enabling them to learn the desired regional similarities and relationships. This approach not only enhances the accuracy of head pose estimation but also provides a deeper understanding of the intricate facial geometry. 6dRepNet [13] tackles unconstrained head pose estimation. It overcomes the limitations of prior methods by using a continuous 6D rotation matrix representation for ground truth data. This allows it to learn the full range of head rotations, unlike previous approaches restricted to narrow angles. Additionally, a geodesic distance-based loss function ensures learned rotations adhere to real-world 3D rotation space geometry, boosting accuracy and robustness. LwPosr [26] utilizes a combination of depthwise separable convolutional and transformer encoder layers for efficient and fine-grained head pose prediction.

Overall, these methods have explored various technological routes in the field of FOE and achieved some good results. If new methods want to further improve performance or enhance generalization, they face significant challenges.

## III. Method

As shown in Fig. 1, estimating the driver's head posture in the monitored image can help solve DDBR problem.

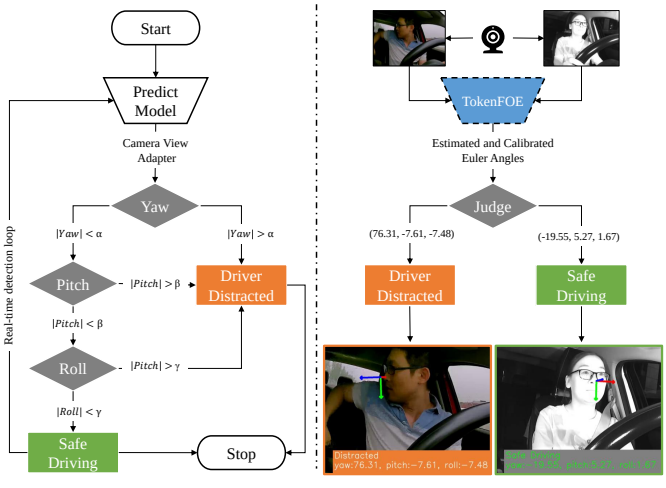


Fig. 1. Left: Overview of proposed driver distraction detection workflow. Right: One RGB and one Near Infrared (NIR) sample of detected results using TokenFOE.

In this section, we first introduce the overview of TokenFOE, then describe every part, including the representation methods of rotation, Gram-Schmidt process, MLP structure, Dimension Adaptive Transformer, and evaluation metric.

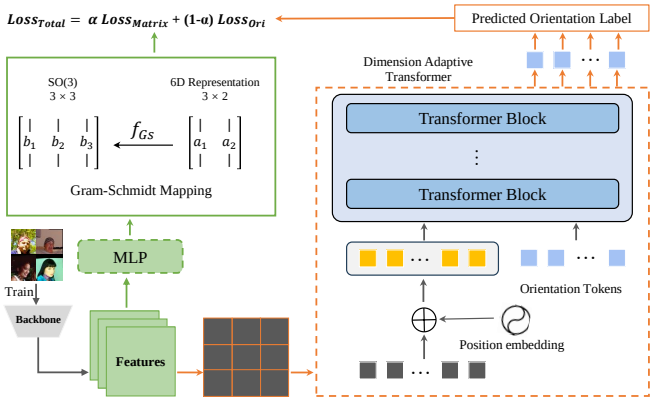


Fig. 2. The pipeline of proposed TokenFOE, a CNN and Transformer fusion dual-branch network based SO(3) and orientation tokens. The main module MLP (*Sec. III-C*), Gram-Schmidt Process (*sec. III-D*) and Dimension Adaptive Transformer (*sec. III-E*) will described respectively.

### A. Pipeline of TokenFOE

*1) Dual branch architecture:* The whole pipeline of TokenFOE is refer to Fig. 2, which first pre-processes the input image (Section IV-B1), then using a pre-trained CNN model to extract image features. The extracted image features are processed in two separate paths. One path is connected to the MLP to predict a 6d representation, follow a Gram-Schimidt process mapped it into a 3×3 rotation matrix belonging SO(3), while the other path adopts the Vit approach to divide features into 14×14 patches, overlay them with position embedding, and send them together with orientation tokens to the transformer block for attention calculation. The calculated results are generated through an fully connected (FC) layer to generate a set of predictions, which are the class of orientation. The calculation results of the two losses are fused using a hyper-parameter $\alpha$ to obtain the final output.

### B. Head Orientation and Representation of Rotation

*1) Head orientation partitioning:* According different orientation euler angles, we could divided head posture to several region. In this work, we followed TokenHPE [16] divided all posture into 9 classes, the illustration refer to Fig. 3.

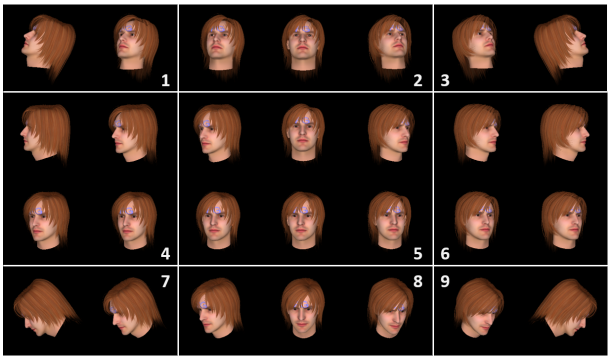


Fig. 3. According the neighbor image similarities, we divided head posture into 9 classes as the schematic diagram.

*2) Rotation matrix:* There are several methods available to represent a 3D rotation, such as Euler angles, Rotation matrix, Axis-angle, Quaternion, or Lie algebra. Among these, Euler angles are commonly used and intuitive. They decompose a 3D rotation into rotations along the three orthogonal coordinate axes of the object, known as Yaw, Pitch, and Roll. Axis-angle representation and Rotation matrix are closely related and can be converted into each other. If $v$ is a vector in $\mathbb{R}^3$ and $k$ is a unit vector describing an axis of rotation about which $v$ rotates by an angle $\theta$ according to the right-hand rule. As we know, an object rotation $\theta$ degrees by axis $x, y, z$ could described by followed Eq. 1 and Eq. 2 where the $c_\theta = cos(\theta), s_\theta = sin(\theta), \xi = 1 - cos(\theta)$.

$$R_{(\theta)} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_\theta & -s_\theta \\ 0 & s_\theta & c_\theta \end{bmatrix} \begin{bmatrix} c_\theta & 0 & s_\theta \\ 0 & 1 & 0 \\ -s_\theta & 0 & c_\theta \end{bmatrix} \begin{bmatrix} c_\theta & -s_\theta & 0 \\ s_\theta & c_\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (1)$$

$$R = \begin{bmatrix} c_\theta + u_x^2\xi & u_x u_y \xi - u_z s_\theta & u_x u_z \xi + u_y s_\theta \\ u_y u_z \xi + u_z s_\theta & c_\theta + u_y^2 \xi & u_y u_z \xi - u_x s_\theta \\ u_z u_x \xi - u_y s_\theta & u_z u_y \xi + u_x s_\theta & c_\theta + u_z^2 \xi \end{bmatrix} \quad (2)$$

### C. MLP Architecture

MLP is a classic artificial neural network model. It consists of an input layer, several hidden layers, and an output layer. In this work, we ultimately chose a single hidden layer MLP module instead of FC according to the ablation study result, with the dimension of the hidden layer set to 768. When using the default RepVgg_b2g4 as the backbone, the complete workflow is shown in Fig. 4.

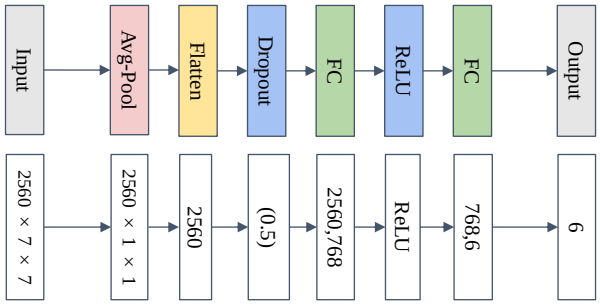


Fig. 4. Upper: MLP architecture with one hidden layer. Lower: the example tensor dimension when use RepVgg_b2g4 as backbone.

### D. Gram-Schmidt Process

A set of nonzero vectors $\mu_1, \mu_2, ..., \mu_n$ is called orthogonal if $\mu_i \cdot \mu_j \neq 0$ whenever $i \neq j$. It is orthonormal if it is orthogonal and in addition $\mu_i \cdot \mu_i = 1$ for all $i = 1, 2, 3, ..., n$. We can use the Gram-Schmidt Process to perform orthogonal transformations on the predicted rotation matrix, that is means we could convert the HPE problem to estimate the satisfied rotation matrix.

According TriNet [27] indicated, we could use map function $f_{Gs}$ (refer to Eq. 3) to map the output into $\hat{R}$, furthermore, predict 6 elements could get the best precision and guarantee

the representation space continuity than predict five or nine elements.

$$\hat{R} = f_{Gs}(\begin{bmatrix} | & | \\ a_1 & a_2 \\ | & | \end{bmatrix}) = \begin{bmatrix} | & | & | \\ b_1 & b_2 & b_3 \\ | & | & | \end{bmatrix} \tag{3}$$

The calculate method of the $f_{Gs}$ is using Eq. 4 and Eq. 5.

$$b_i = \begin{bmatrix} N(a_1) & i=1 \\ N(a_2 - (b_1 \cdot a_2)b_1) & i=2 \\ b_1 \times b_2 & i=3 \end{bmatrix}^T \tag{4}$$

$N(\cdot)$ denotes a Normalization function.

$$N(u_i) = \frac{u_i}{\|u_i\|} \tag{5}$$

### E. Dimension Adaptive Transformer

ViT [28] is a typical model and also is the first successful vision transformer model. In ViT, an input image undergoes a process of being segmented into smaller patches to avoid the exponential increase in computational complexity caused by too many patches, e.g. the default input image size is $224 \times 224 \times 3$, the author divides the image into $16 \times 16 \times 3$ patches, and maps each patch to a 768 dimensional 1D vector. The total number of patches is $14 \times 14 = 196$.

*1) Dimension adaptive layer:* In TokenFOE, the object processed by the transformer block is not the original image data, but the features output by the backbones. Different backbones output different feature dimensions and orders, for example, the output of resnet50 is $2048 \times 7 \times 7$, the output of RepVgg_b2g4 is $2560 \times 7 \times 7$, and the output of Swin_base_224 is $7 \times 7 \times 1024$. To be compatible with different feature dimensions without losing information, we introduce a DDT(Dimension Adaptive Transformer), as shown in Fig. 5. The input features undergo dimension adaptation while keeping the number of patches constant, and then are uniformly mapped to 128 dimensions as input for the transformer block. This operation allows our model to match the output of any backbone.

*2) Transformer block:* Given the 1D token embedding sequence $T = \{[visual], [euler\ angles]\}$ as input, the Transformer encoder learns pose feature representation by stacking M blocks. Each block contains a Multi-head Self-attention (MSA) module and a Multilayer Perceptron (MLP) module. In addition, layer norm (LN) is adopted before every module. Self-attention (SA) can be formulated as Eq. 6.

$$SA(T^{l-1}) = softmax(\frac{T^{l-1}W_Q(T^{l-1}W_K)^T}{\sqrt{d_h}})(T^{l-1}W_V) \tag{6}$$

where $W_Q, W_K, W_V \in R^{d \times d}$ are the learnable parameters of three linear projection layers, $T_{l-1}$ is the output of the $(l-1)_{th}$ layer, $d$ is the dimension of tokens, and $d_h = d \cdot MSA$ is an extension of SA with h self-attention operations which are called "heads". In MSA (Refer to Eq. 7), $d_h$ is typically set to $d/h$.

$$MSA(T) = [SA_1(T); SA_2(T); ...; SA(h)(T)]W_p \tag{7}$$

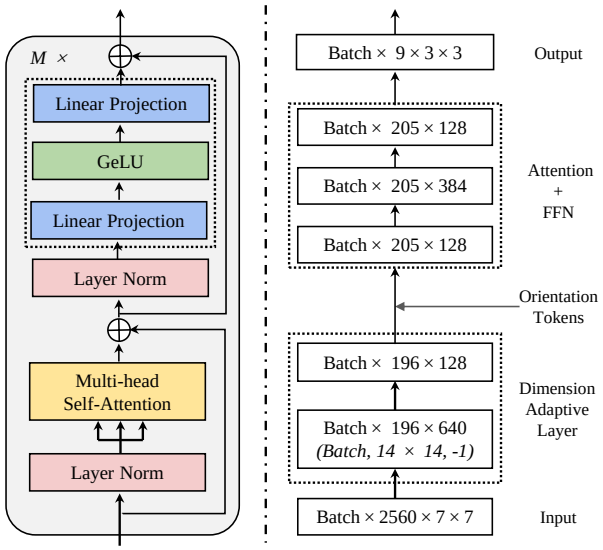


Fig. 5. Left: The architecture of Transformer block that consist of transformer branch. Right: The pipeline and example of dimension adaptive calculation process when use RepVgg_b2g4 as backbone.

Given that spatial relationships are essential for accurate HPE, positional embedding, $pos$, is added to the visual tokens to reserve spatial relationships, which can be expressed as Eq. 8:

$$[Visual] = \{v_1 + pos, v_2 + pos, ..., v_n + pos\} \tag{8}$$

where, n is the number of patches. Then, we obtain n×1D vectors symbolically presented by [visual] tokens. The position embedding use Eq. 9 followed [29].

$$PE = \begin{cases} \sin(pos/10000^{2i/d_{model}}) & (pos, 2i) \\ \cos(pos/10000^{2i/d_{model}}) & (pos, 2i+1) \end{cases} \tag{9}$$

### F. Evaluation Metric

*1) Loss function:* As we know, for any $R \in SO(3)$ must satisfy $R^T = R^{-1}$, $detR = \pm 1$. Given a specified image $x^i$ from the test data set, there is only one correct rotation matrix converted from the label we called $R_{gt}$, The model could predict a rotation matrix $\hat{R}$, we want the $\hat{R}$ as closely as possible of $R_{gt}$, the limitation is $\hat{R} = R_{gt}$, then $\hat{R}R_{gt}^T = I$, $I$ is an identity matrix. we use the Eq. 10 as the loss function followed paper [13].

$$\mathbb{L}_{pose} = \arccos\left(\frac{1}{2}(tr(\hat{R}_i R_{gt_i}^T) - 1)\right) \tag{10}$$

When $R_{gt_i} = \hat{R}_i$, the $\mathbb{L}_{pose}$ equal zero. If the result is not equal to zero, It means our model predicts an incorrect rotation matrix, we could use the result to penalty our model by back propagation algorithm. The final loss function is calculated by Eq. 11, $\alpha$ is a hyper-parameters combine the two branches.

$$Loss_{Total} = \alpha Loss_{matrix} + (1-\alpha)Loss_{ori} \tag{11}$$

*2) Evaluate:* MAE is a standard metric for HPE, it is defined as Eq. 12.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}(|x_i - \hat{x}_i|) \tag{12}$$

where N is the number of face images and $x_i$ and $\hat{x}_i$ represent the ground truth and predicted pose parameters, respectively.

MAEV is defined as Eq. 13. where N is the number of face images in the dataset and $v_i$ and $\hat{v}_i$ are the vector of ground truth and the predicted result.

$$MAEV = \frac{1}{N}\sum_{i=1}^{N} cos^{-1}\left(\frac{v_i \cdot \hat{v}_i}{|v_i||\hat{v}_i|}\right) \tag{13}$$

## IV. Experiments

### A. Datasets

We follow the methodologies employed in [13], [15], [11] and utilize well-established public face datasets for training and testing. Specifically, we use the widely recognized 300W-LP [8] as the training set, AFLW-2000 [18] and BIWI [19] for quantitative testing, 100-Driver [22] for DDD task.

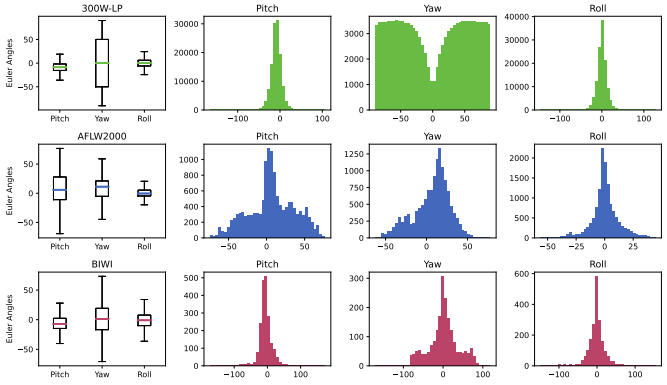


Fig. 6. Box and Hist plots of Training and Testing datasets, statistic by Euler angles. Green(300W-LP [8]), Blue(BIWI [19]) and Red(AFLW2000 [18]).

*1) Training:* The 300W-LP [8] dataset comprises over 60,000 face samples collected from multiple databases, the statistic results shown in the first line of Fig. 6. We convert ground truth Euler angle into rotation matrices for training.

*2) Quantitative testing:* AFLW-2000 [18] consists of the first 2000 images from the AFLW dataset. This dataset is annotated with ground truth 3D faces and corresponding 21 landmarks. It offers a diverse range of samples with varying lighting conditions and occlusion levels, providing a robust evaluation environment for our model. BIWI [19] includes more than 15K images of 20 individuals. The head pose range covers about ±75° yaw and ±60° pitch. The statistic data is shown as Fig. 6.

*3) Driver distracted detection:* 100-Driver [22] is the largest DDD dataset with more than 470K images taken by 4 cameras, observing 100 drivers, over 79 hours from 5 vehicles and including different vehicles, drivers, camera view, and modalities.

### B. Model Settings

*1) Pre-processing:* First, we resize all the images into 300×300 pixels, then random crop to 224×224 pixels as input, we don't use any other augment method. The model uses the pre-trained RepVgg_b2g4 provided by TIMM framework [30] as the backbone, it's one of powerful and popular CNN models. We use Top-Down mode and employ MTCNN [31] to deal with the face location detection task.

*2) Hyper-parameters settings:* The training and testing environment is Ubuntu 22.04, Python 3.10, cuda 12.6, Pytorch 1.13, and a Nvidia 2080Ti with 11G GPU memory. The core parameters of the training are: the batch size is 64, use the Adam optimizer, the total epochs is 30, and the initial Leaning Rate is set to 1e-4, which is reduced halve when $8^{th}$, $16^{th}$and $24^{th}$ Epoch. All the weights in our model are random initialization and the hyper-parameters $\alpha$ are set to 0.6 by the ablation study result.

### C. Experimental Results

Fig. 7 shows that the accuracy on AFLW2000 and BIWI consistently increases with the training progress. There are three periodic low points at $9^{th}$, $17^{th}$ and $27^{th}$ epoch, respectively, just accompanied by three times decrease in learning rate. The plot also shows an increasing trend until the $30^{th}$ Epoch, but the growth limit has not been explored yet.

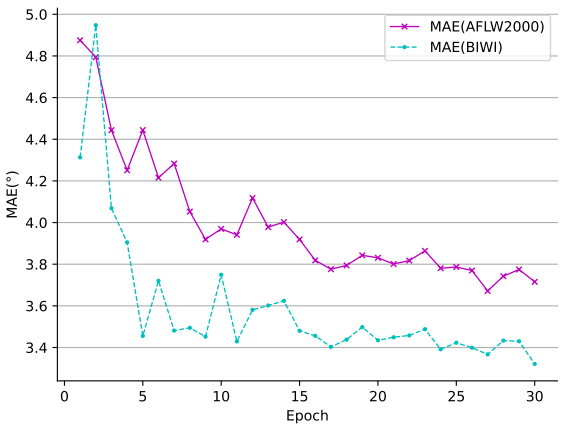


Fig. 7. Per epoch MAE plots tested on AFLW-2000 [18] and BIWI [19].

Follow the old SOTA method, TokenHPE [16], we also show the MAEV scores compared with other popular EAF methods, Refer to Fig. 8, our score exceeds others by a wide margin, specifically, improved by 19.2% on AFLW2000.

HPE is a classical problem that has been studied in many papers. According to literature [16], the methods are divided into two categories: pure HPE and using additional annotated data. Taking SynergyNet [7] as an example, facial key points are used for training. In fact, the coordinate of upper key points already contains head orientation information, which can greatly improve accuracy. As can be seen from Table I, this type of method generally has higher accuracy. Among all methods that do not use additional annotation information, our method has the highest accuracy. We achieved four new SOTA on AFLW-2000 and BIWI. Because TokenHPE [16] is

TABLE I. MEAN ABSOLUTE ERRORS OF EULER ANGLES AND VECTORS ON THE AFLW2000 [18] AND BIWI [19] . ALL METHODS ARE TRAINED ON THE 300W-LP [8] DATASET. COLUMN ‡ INDICATE WHETHER EXTRA ANNOTATIONS FREE. ✓MEANS JUST USING ORIENTATION LABELS FOR SUPERVISING, ✗MEANS USED EXTRA ANNOTATIONS, SUCH AS LANDMARKS. TOKENHPE [16](INDICATED BY UNDERLINE) IS MOST FAMILIAR AND TARGET MODEL. BOLD FONT INDICATE THE SOTA SCORE FOR EVERY COLUMN.

| Methods | ‡ | AFLW2000-Euler | | | | AFLW2000-Vector | | | | BIWI-Euler | | | | BIWI-Vector | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Yaw | Pitch | Roll | MAE | Left | Down | Front | MAEV | Yaw | Pitch | Roll | MAE | Left | Down | Front | MAEV |
| EVA-GCN [32] | ✗ | 4.46 | 5.34 | 4.11 | 4.64 | - | - | - | - | 4.01 | 4.78 | 2.98 | 3.92 | - | - | - | - |
| SynergyNet [7] | ✗ | 3.42 | 4.09 | 2.55 | 3.35 | - | - | - | - | - | - | - | - | - | - | - | - |
| img2Pose [33] | ✗ | 3.43 | 5.03 | 3.28 | 3.91 | - | - | - | - | - | - | - | - | - | - | - | - |
| HopeNet [14] | ✓ | 5.31 | 7.12 | 6.13 | 6.20 | 7.07 | 5.98 | 7.50 | 6.85 | 6.01 | 5.89 | 3.72 | 5.2 | 7.65 | 6.73 | 8.68 | 7.69 |
| FSA-Net [15] | ✓ | 4.96 | 6.34 | 4.78 | 5.36 | 6.75 | 6.22 | 7.35 | 6.77 | 4.56 | 5.21 | 3.07 | 4.28 | 6.03 | 5.96 | 7.22 | 6.40 |
| LwPosr [26] | ✓ | 4.8 | 6.38 | 4.88 | 5.35 | - | - | - | - | | | | | | | | |
| Quatnet [34] | ✓ | 3.97 | 5.62 | 3.92 | 4.50 | - | - | - | - | 4.01 | 5.49 | 2.94 | 4.15 | - | - | - | - |
| Trinet [11] | ✓ | 4.2 | 5.77 | 4.04 | 4.67 | 5.78 | 5.67 | 6.52 | 5.99 | 3.05 | 4.76 | 4.11 | 3.97 | 5.57 | 5.46 | 6.57 | 5.86 |
| TokenHPE-v1 [16] | ✓ | 4.53 | 5.73 | 4.29 | 4.85 | 6.16 | 5.21 | 6.97 | 6.11 | | | | | | | | |
| TokenHPE-v2 [16] | ✓ | 4.36 | 5.54 | 4.08 | 4.66 | 6.01 | 5.10 | 6.82 | 5.98 | 3.95 | 4.51 | 2.71 | 3.72 | 5.41 | 5.17 | 6.23 | 5.60 |
| 6DRepNet [13] | ✓ | 3.63 | 4.91 | 3.37 | 3.97 | - | - | - | - | **3.24** | 4.48 | 2.68 | 3.47 | - | - | - | - |
| zhao et al [35] | ✓ | 3.72 | 4.52 | 3.16 | 3.80 | - | - | - | - | 3.45 | 4.32 | 2.75 | 3.51 | - | - | - | - |
| **TokenFOE(Ours)** | ✓ | **3.49**↓ 20.0% | **4.48**↓ 19.1% | **3.04**↓ 25.5% | **3.67**↓ 21.2% | **4.64**↓ 22.8% | **4.37**↓ 14.3% | **5.30**↓ 22.3% | **4.83**↓ 19.2% | 3.67↓ 7.1% | **3.91**↓ 13.3% | **2.52**↓ 7.0% | **3.37**↓ 9.4% | **4.79**↓ 11.5% | **4.56**↓ 11.8% | **5.72**↓ 8.2% | **5.03**↓ 10.2% |

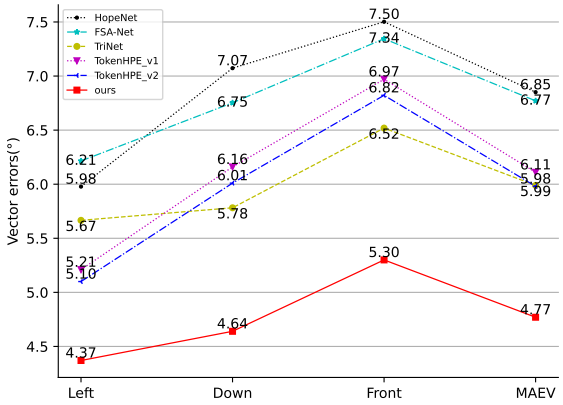


Fig. 8. Comparison with other methods on AFLW-2000 measured by MAEV. Our accuracy exceeds other EAF methods by a wide margin. The lower the better, best viewed in color.

the most familiar with our method, we conducted a careful comparison with it (indicated by underline), using downward arrows(↓) and percentages(%) to indicate the specific values of accuracy improvement.

We conducted extensive ablation experiments to investigate the impact of each module on overall performance.

Table II shows the impact of different backbones on model performance. The results show that the backbone has a significant impact on the final performance. Combining the training and inference speed experimental results in Table III, a trade-off between performance and speed can be made according to specific needs.

TABLE II. ABLATION STUDY OF DIFFERENT BACKBONE. THE OVERALL PERFORMANCE OF THE CNN MODEL IS BETTER THAN THAT OF THE TRANSFORMER MODEL

| Backbone | Type | MAE(AFLW2000) | MAE(BIWI) |
|---|---|---|---|
| Swin_base_224 | Transformer | 5.03 | 3.44 |
| ResNet50 | CNN | 4.40 | 3.69 |
| RepVgg_b2g4 | CNN | 3.67 | 3.37 |

The MLP structure also has a significant impact on accuracy. We compared and tested FC, single hidden layer MLP with different dimensions, and double hidden-layer MLP, and found that the single-layer MLP with dimension 768 had the best performance. All the data refer to Table IV.

TABLE III. ABLATION STUDY RESULTS OF TRAINING AND INFERENCE SPEED. 'M' AND 'MS' ABBREVIATION FOR MINUTES AND MILLISECONDS, RESPECTIVELY

| Model | Backbone | Image Size (C,H,W) | Training (M/epoch) | Inference (ms/image) |
|---|---|---|---|---|
| TokenFOE | Swin_base_224 | 3,224,224 | 29 | 33.9 |
| TokenFOE | ResNet50 | 3,224,224 | 13 | 13.3 |
| TokenFOE | RepVgg_b2g4 | 3,224,224 | 31 | 48.7 |

TABLE IV. ABLATION STUDY RESULTS OF DIFFERENT MLP ARCHITECTURES. SINGLE HIDDEN LAYER ARE THE BEST MODULE THAN FC AND TWO HIDDEN LAYER ARCHITECTURE

| Module | Hidden Layers | Hidden Dimensions | MAE(AFLW2000) |
|---|---|---|---|
| FC | 0 | - | 3.77 |
| MLP | 1 | 128 | 4.05 |
| MLP | 1 | 256 | 4.03 |
| MLP | 1 | 512 | 3.76 |
| MLP | 1 | 768 | 3.67 |
| MLP | 1 | 1024 | 3.95 |
| MLP | 2 | 1024,256 | 4.26 |

In the end, we tested different loss weights refer to Fig. 9, that is say the different values of $\alpha$. According to our design, the Transformer branch performs classification tasks, which can improve the overall model accuracy(improved about 11.2%) but is not suitable for independent work. When $\alpha$ is set to 1, means only the CNN branch worked.

So far, our method can accurately estimate the head pose angle, but there is still a problem when facing DDD tasks. The training images are all based on the front profile view as the initial position, and the output Euler angles are also the deviation values relative to this initial position.

However, in the DDD task, the cameras may be arranged in different positions, and due to the influence of the view angle, completely different Euler angles will be output for the same driver posture. This article simply sets up a Camera Adapter layer to solve it. Through cluster analysis for 100-driver [22],

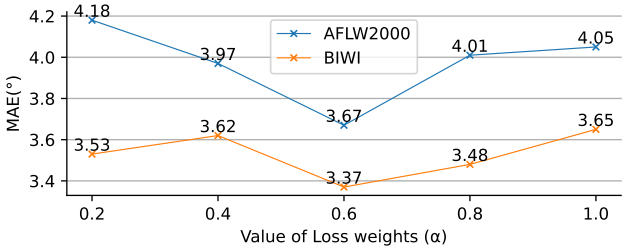


Fig. 9. Ablation study results of model architectures. $Loss_{Matrix}$ is more important than $Loss_{Ori}$.

it was determined that Pitch and Roll directions greater than $\pm 15°$, Yaw direction greater than $50°$, or yaw less than $-75°$ with an absolute value greater than $25°$ are Distracted. Fig. 10 shows the visual detection results of different camera views, modalities, lighting, vehicles, and drivers, further demonstrating the effectiveness and robustness of this method.

## V. Limitation and Discussion

In this section, we will discuss the limitations of our method in this article.

First, TokenFOE adopts a dual-branch structure to calculate two sets of Lose separately, which theoretically enhances the model's expressive power. However, compared to CNN, especially lightweight CNN models represented by Mobile-net [36], the computational complexity and resource consumption of the transformer model have significantly increased, which is not cost-effective from an efficiency perspective. The advantage of the transformer scheme is that the model performs better when training with large amounts of data. If we want to deploy the model to a vehicle terminal device for terminal inference, maybe need to fine-tune the model, such as using a lighter backbone and reducing the number of parameters appropriately.

Second, sleeping and yawning is also an important sign of fatigue and distraction, our method can only detect head posture, making it difficult to make quick judgments in such scenarios (refer to Fig. 11).

## VI. Conclusion

We proposed a novel dual-branch network named TokenFOE, that combines CNN and Transformer, one is the classic CNN path, and the other is a transformer model based on a self-attention mechanism, the dimension adaptive algorithm suitable uses any pre-trained backbone for the feature extractor.

We train the model on 300W-LP, quantitative test on AFLW-2000 and BIWI. The experiment results show that the MAE score is improved by 21.2% and 9.4% compared to the original SOTA model, and the MAEV score is improved by 19.2% and 10.2%, respectively. Based on the model output and calibrated through the camera adapter module, we present the visualization results on the largest DDD dataset currently available, the 100-driver [22] dataset. Robust and accurate detection results were achieved for four different camera perspectives in daytime (RGB) and Night time(NIR). Additionally, the ablation study shows that the model inference speed (21 to 75fps) can be used for real-time detection.

The main limitation of this method is that the heavy model leads to high training and inference costs, and the computational overhead of the self-attention part is too high. In theory, more tasks can be completed by adding a small number of additional tokens, such as key points. In addition to further improving accuracy and reducing costs, the fusion of multi-task and multi-modality is also a future research direction.

## Conflict of Interest

The authors declare no conflict of interest.

## Authors' Contributions

Xiong Zhao conceived the main idea, conducted the experiments and wrote the manuscript. Sarina Sulaiman guided the research direction. Wong Yee Leng analyze the results and proposed improvement suggestions. All authors had reviewed and approved the final version.

## Funding

This work were supported by Chinese Universities' Industry-University-Research Innovation Fund—BeiChuang Teaching Assistant Project (Phase II) (2021BCF01006), Yunnan Provincial Department to Education Science Research Fund Project (2022J1281; 2024J1352) and The Innovation Team of Intelligent Manufacturing and New Power System Research, Yunnan College of Business Management (2022XKJS02).

## References

[1] J. D. Lee, K. L. Young, and M. A. Regan, "Defining driver distraction," *Driver distraction: Theory, effects, and mitigation*, vol. 13, no. 4, pp. 31–40, 2008.

[2] E. Wassef, H. E. Abd El Munim, S. Hammad, and M. Ghoneima, "Robust real-time head pose estimation for 10 watt sbc," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 7, 2021.

[3] Y. Wang, W. Liang, J. Shen, Y. Jia, and L.-F. Yu, "A deep coarse-to-fine network for head pose estimation from synthetic data," *Pattern Recognition*, vol. 94, pp. 196–206, 2019.

[4] Y. Shu and L. Hu, "A vision-based human posture detection approach for smart home applications," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 10, 2023.

[5] H. Tang, M. Dai, X. Du, J.-L. Hung, and H. Li, "An eeg study on college students' attention levels in a blended computer science class," *Innovations in Education and Teaching International*, pp. 1–13, 2023.

[6] J. Mo, G. Jiang, H. Yuan, Z. Shou, and H. Zhang, "Adaptive target region attention network-based human pose estimation in smart classroom," *International Journal of Advanced Computer Science and Applications*, vol. 15, no. 4, 2024.

[7] C. Y. Wu, Q. G. Xu, U. Neumann, and I. C. Soc, "Synergy between 3dmm and 3d landmarks for accurate 3d facial geometry," in *9th International Conference on 3D Vision (3DV)*, ser. International Conference on 3D Vision. LOS ALAMITOS: Ieee Computer Soc, 2021, Conference Proceedings, pp. 453–463. [Online]. Available: ¡Go to ISI¿://WOS:000786496000045

[8] X. Zhu, Z. Lei, X. Liu, H. Shi, and S. Z. Li, "Face alignment across large poses: A 3d solution," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 146–155.

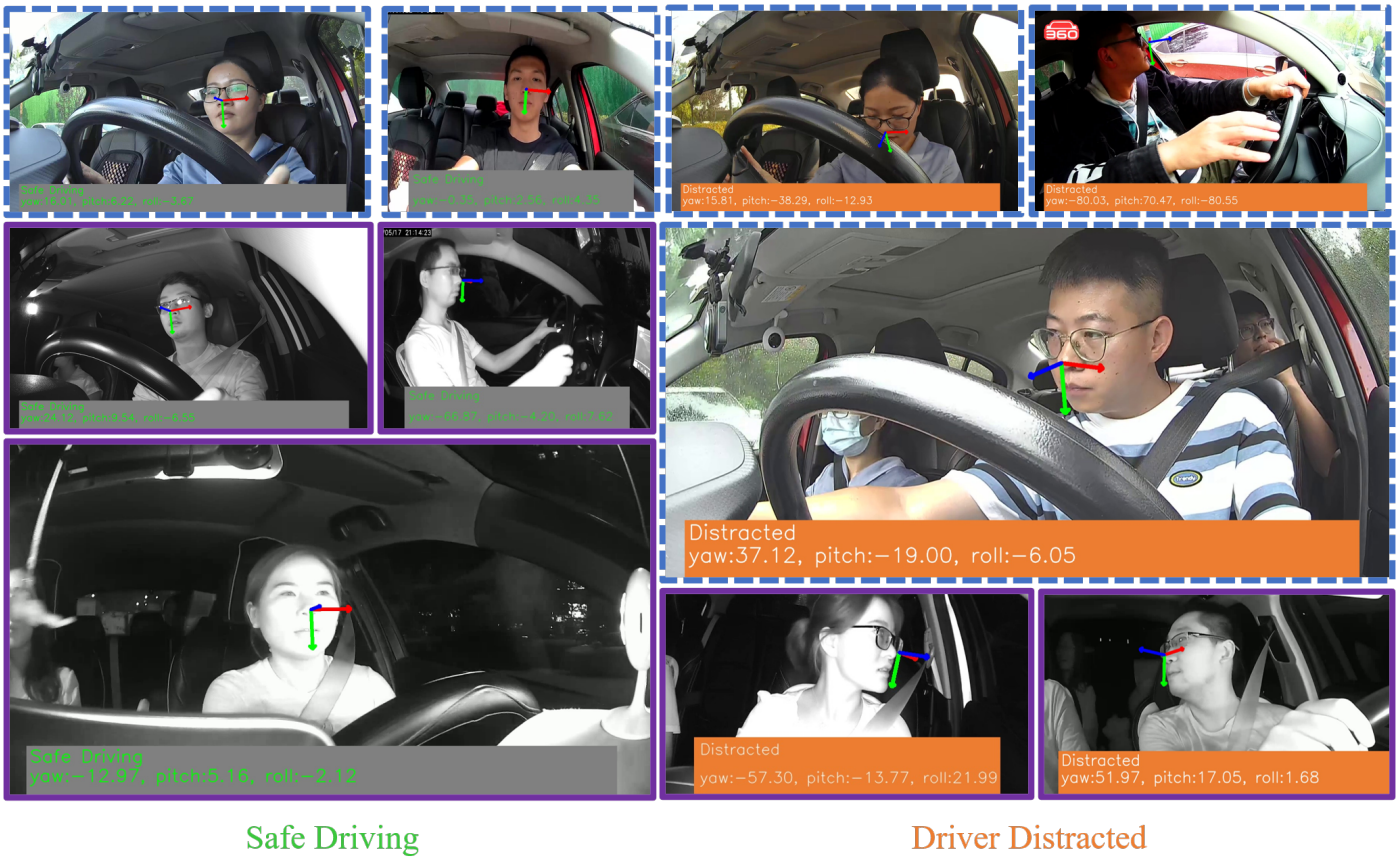


Fig. 10. Visualization of more driver distracted detection results. Blue dashed line, purple bold line, green text and white text indicate Day(RGB), Night(NIR), safe driving and driver distracted, respectively. All images from 100-driver [22].

Fig. 11. Two failure samples.

[9] J. Guo, X. Zhu, Y. Yang, F. Yang, Z. Lei, and S. Z. Li, "Towards fast, accurate and stable 3d dense face alignment," in *European Conference on Computer Vision*. Springer, 2020, Conference Proceedings, pp. 152–168.

[10] Y. Yun, M. H. Changrampadi, and I. Y. Gu, "Head pose classification by multi-class adaboost with fusion of rgb and depth images," in *2014 International Conference on Signal Processing and Integrated Networks (SPIN)*. IEEE, 2014, pp. 174–177.

[11] Z. W. Cao, Z. C. Chu, D. F. Liu, Y. J. Chen, and Ieee, "A vector-based representation to enhance head pose estimation," in *IEEE Winter Conference on Applications of Computer Vision (WACV)*, ser. IEEE Winter Conference on Applications of Computer Vision. LOS ALAMITOS: Ieee Computer Soc, 2021, Conference Proceedings, pp. 1187–1196. [Online]. Available: ¡Go to ISI¿://WOS:000692171000118

[12] X. Geng and Y. Xia, "Head pose estimation based on multivariate label distribution," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, Conference Proceedings, pp. 1837–1842.

[13] T. Hempel, A. A. Abdelrahman, and A. Al-Hamadi, "6d rotation representation for unconstrained head pose estimation," in *2022 IEEE International Conference on Image Processing (ICIP)*, 2022, Conference Proceedings, pp. 2496–2500.

[14] B. Doosti, S. Naha, M. Mirbagheri, and D. J. Crandall, "Hope-net: A graph-based model for hand-object pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[15] T.-Y. Yang, Y.-T. Chen, Y.-Y. Lin, and Y.-Y. Chuang, "Fsa-net: Learning fine-grained structure aggregation for head pose estimation from a single image," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1087–1096.

[16] C. Zhang, H. Liu, Y. Deng, B. Xie, and Y. Li, "Tokenhpe: Learning orientation tokens for efficient head pose estimation via transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, Conference Proceedings, pp. 8897–8906.

[17] Y. Li, S. Zhang, Z. Wang, S. Yang, W. Yang, S.-T. Xia, and E. Zhou, "Tokenpose: Learning keypoint tokens for human pose estimation," in *Proceedings of the IEEE/CVF International conference on computer vision*, 2021, Conference Proceedings, pp. 11 313–11 322.

[18] X. Zhu, Z. Lei, J. Yan, D. Yi, and S. Z. Li, "High-fidelity pose and expression normalization for face recognition in the wild," in

*Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 787–796.

[19] G. Fanelli, M. Dantone, J. Gall, A. Fossati, and L. Van Gool, "Random forests for real time 3d face analysis," *International journal of computer vision*, vol. 101, pp. 437–458, 2013.

[20] G. Li, Y. Yuan, D. Ouyang, L. Zhang, B. Yuan, X. Chang, Z. Guo, and G. Guo, "Driver distraction from the eeg perspective: A review," *IEEE Sensors Journal*, 2023.

[21] L. Mou, J. Chang, C. Zhou, Y. Zhao, N. Ma, B. Yin, R. Jain, and W. Gao, "Multimodal driver distraction detection using dual-channel network of cnn and transformer," *Expert Systems with Applications*, vol. 234, p. 121066, 2023.

[22] J. Wang, W. Li, F. Li, J. Zhang, Z. Wu, Z. Zhong, and N. Sebe, "100-driver: a large-scale, diverse dataset for distracted driver classification," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 7, pp. 7061–7072, 2023.

[23] Y. Feng, F. Wu, X. Shao, Y. Wang, and X. Zhou, "Joint 3d face reconstruction and dense alignment with position map regression network," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 534–551.

[24] H. Wang, Z. Chen, and Y. Zhou, "Hybrid coarse-fine classification for head pose estimation," *arXiv preprint arXiv:1901.06778*, 2019.

[25] L. Liu, Z. Ke, J. Huo, and J. Chen, "Head pose estimation through keypoints matching between reconstructed 3d face model and 2d image," *Sensors*, vol. 21, no. 5, p. 1841, 2021.

[26] N. Dhingra, "Lwposr: Lightweight efficient fine grained head pose estimation," in *Proceedings of the IEEE/CVF Winter Conference on applications of computer vision*, 2022, Conference Proceedings, pp. 1495–1505.

[27] Y. Zhou, C. Barnes, J. Lu, J. Yang, and H. Li, "On the continuity of rotation representations in neural networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5745–5753.

[28] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, and S. Gelly, "Vit: An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[29] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, l. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[30] R. Wightman, "Pytorch image models," https://github.com/rwightman/pytorch-image-models, 2019.

[31] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE signal processing letters*, vol. 23, no. 10, pp. 1499–1503, 2016.

[32] M. Xin, S. Mo, and Y. Lin, "Eva-gcn: Head pose estimation based on graph convolutional networks," in *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, 2021, Conference Proceedings, pp. 1462–1471.

[33] V. Albiero, X. Chen, X. Yin, G. Pang, and T. Hassner, "img2pose: Face alignment and detection via 6dof, face pose estimation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, Conference Proceedings, pp. 7617–7627.

[34] H.-W. Hsu, T.-Y. Wu, S. Wan, W. H. Wong, and C.-Y. Lee, "Quatnet: Quaternion-based head pose estimation with multiregression loss," *IEEE Transactions on Multimedia*, vol. 21, no. 4, pp. 1035–1046, 2019.

[35] X. Zhao, S. Sulaiman, L. Chen, M. Dong, Y. Duo, and H. Song, "Continuity rotation representation for head pose estimation without keypoints," in *Proceedings of the 2023 9th International Conference on Computing and Artificial Intelligence*, 2023, Conference Proceedings, pp. 358–363.

[36] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.