OsGG-Net: One-step Graph Generation Network for Unbiased Head Pose Estimation

Shentong Mo*
Carnegie Mellon University
Pittsburgh, USA
shentonm@andrew.cmu.edu

Xin Miao†
Institute of Automation, Chinese Academy of Sciences
Beijing, China
miao.xin@ia.ac.cn

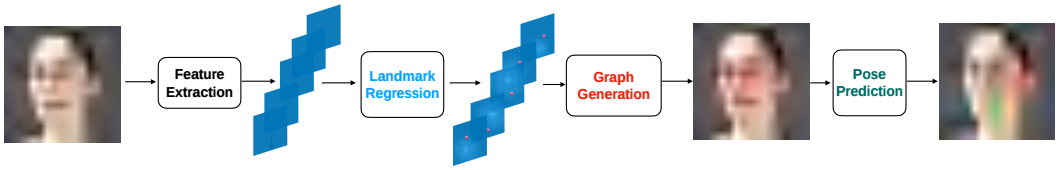

Figure 1: Pipeline of OsGG-Net for head pose estimation by generating a landmark-connection graph to model the 3D angle associated with the landmark distribution robustly in *one step*.

ABSTRACT

Head pose estimation is a crucial problem that involves the prediction of the Euler angles of a human head in an image. Previous approaches predict head poses through landmarks detection, which can be applied to multiple downstream tasks. However, previous landmark-based methods can not achieve comparable performance to the current landmark-free methods due to lack of modeling the complex nonlinear relationships between the geometric distribution of landmarks and head poses. Another reason for the performance bottleneck is that there exists biased underlying distribution of the 3D pose angles in the current head pose benchmarks. In this work, we propose **OsGG-Net**, a One-step Graph Generation Network for estimating head poses from a single image by generating a landmark-connection graph to model the 3D angle associated with the landmark distribution robustly. To further ease the angle-biased issues caused by the biased data distribution in learning the graph structure, we propose the UnBiased Head Pose Dataset, called UBHPD, and a new unbiased metric, namely UBMAE, for unbiased head pose estimation. We conduct extensive experiments on various benchmarks and UBHPD where our method achieves the state-of-the-art results in terms of the commonly-used MAE metric and our proposed UBMAE. Comprehensive ablation studies also demonstrate the effectiveness of each part in our approach.

CCS CONCEPTS

• **Computing methodologies** → *Artificial intelligence*; *Computer vision*; Computer vision tasks.

*This work was done during his intern at CASIA.
†Corresponding author.

MM '21, October 20–24, 2021, Virtual Event, China

ACM ISBN 978-1-4503-8651-7/21/10…$15.00
https://doi.org/10.1145/3474085.3475417

KEYWORDS

head pose estimation, graph generation, unbiased datasets

ACM Reference Format:
Shentong Mo and Xin Miao. 2021. OsGG-Net: One-step Graph Generation Network for Unbiased Head Pose Estimation. In *Proceedings of the 29th ACM International Conference on Multimedia (MM '21), October 20–24, 2021, Virtual Event, China.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3474085.3475417

1 INTRODUCTION

Head pose estimation [5, 10, 25, 32] has long been active research in computer vision, as it can be applied in lots of real problems, such as driver behavior monitoring, human attention modeling, and human-computer interaction. This paper addresses the head pose estimation problem from a single image, where we need to predict a 3D vector containing the angles of yaw, pitch, and roll.

In the head pose estimation literature, most previous methods [17, 23] apply landmarks as the intermediate step to regress the 3 degree-of-freedom angles from an image. Therefore, landmarks are typically useful for several downstream tasks such as face recognition or identification, face alignment, expression transfer, and so on. However, previous methods rarely apply the robustness of a facial graph for head pose estimation. In this work, we explore the robustness of facial graph generation for modeling the 3D pose angles associated with the landmark distribution efficiently.

In terms of graph generation, researchers [13, 24] often apply *two-step* methods to generate graphs from feature maps. Specifically, they first regress the spatial location of nodes from the feature map and then link edges by generating the adjacent matrix according to relationships of nodes. With this kind of two-step method, we reveal it hard to model the complex nonlinear relationships between the geometric distribution of landmarks and head poses robustly, leading to the performance bottleneck of landmark-based models. Another cause leading to this performance bottleneck is that *there exist underlying distribution biases of the 3D pose angles lying in the current head pose datasets*. Therefore, we are wondering if the

negative effect brought by such angle-biased distribution existing in current benchmarks [3, 6, 7, 16, 36] can be mitigated.

In this work, we propose an *end-to-end* and *image-to-graph* architecture to generate a landmark-connection graph that takes the generated facial landmarks as the vertexes to model the 3D angle associated with the landmark distribution robustly. Specifically, we adopt a *one-step* approach for face graph generation network, namely OsGG-Net, where we take a single image as input, generate the facial landmarks first, and then apply the spatial graph convolution network (GCN) to regress three directions of pose angles. Our OsGG-Net consists of three pipelines, that is, landmark, graph, and pose pipelines. In the landmark pipeline, we apply an integral landmark regression to automatically localize the face landmarks from an image. Then the graph pipeline is proposed to generate a face graph for pose estimation. In the end, the pose pipeline focuses on extracting the spatial extent of the face graph, by directly regressing the yaw, pitch, and roll of head poses.

To further ease the angle-biased problem caused by the biased distribution lying in the current benchmarks, we propose the UnBiased Head PoseDataset by human-cleaning, called UBHPD, for unbiased head pose estimation. Specifically, we explore the underlying distribution histogram of existing datasets in Section 4. In order to generate an unbiased head pose dataset, we uniformly sampling the images from BIWI, AFLW, 300W-LP, UPNA and SynHead according to the density histogram for each direction of head pose angle. Taking this underlying distribution difference into account, we propose an unbiased metric, namely UBMAE, to have a balanced evaluation of each angle for head pose estimation.

The major contributions of this paper are summarized as follows:

- We propose an end-to-end and image-to-graph framework to generate a robust facial graph by applying CNN and GCN jointly.
- We propose a novel one-step face graph generation network, called OsGG-Net, instead of using a two-step framework, for estimating head poses from a single image.
- We are the first to reveal the potential bias lying in current head pose benchmarks and propose an UnBiased Head Pose Dataset, namely UBHPD, and a new metric called UBMAE.
- Extensive experiments demonstrate that OsGG-Net achieves state-of-the-art results on various benchmarks and UBHPD in terms of both MAE and UBMAE metrics.

2 RELATED WORKS

2.1 Head Pose Estimation

In the head pose estimation community, previous works fall into two categories, *i.e.* landmark-free and landmark-based.

Landmark-free. These methods typically predict three Euler angles directly from a single image without landmarks involved. For example, HopeNet [25] applies cross-entropy and Mean Square Error (MSE) losses to train a deep convolutional neural network (CNN). QuatNet [10] adopts a quaternion-based multi-regression loss method to avoid ambiguity problem in the commonly used Euler angle representation. FSA-Net [32] proposes a stage-wise regression mechanism with a CNN model and an attention mechanism combined with a feature aggregation module to group global spacial features. Liu *et al.* [20] present the head pose estimation as a label distribution learning paradigm with a multi-loss function by regressing a Gaussian label distribution rather than a single label. Concurrently, a novel model based on the characteristic of representations with three vectors in a rotation matrix is developed in TriNet [5] to address the discontinuity in annotations and the Mean Absolute Error (MAE) of Euler angles based metric.

Landmark-based. This kind of methods often detect facial landmarks first and then use them to estimate the head pose. For example, Hyperface [23] applies a multi-task model for face detection, landmarks localization, pose estimation and gender recognition using deep CNNs. KEPLER [17] proposes a modified GoogLeNet [29] architecture for landmark detection and pose estimation of unconstrained faces by regression, where landmark prediction is improved by the coarse pose supervision. A number of landmark-based methods attract much attention in the literature, since facial landmarks are important intermediate results shared by multiple downstream tasks. However, current landmark-based methods fails to achieve comparable performance to the landmark-free methods due to the lack of the abilities to model the complex nonlinear relationships between the geometric distribution of landmarks and head poses robustly and efficiently. Therefore, it is necessary to take full advantage of landmarks to solve the current performance bottleneck by designing such nonlinear mappings. In this work, we focus on modeling the 3D pose angles associated with the landmark distribution efficiently by generating a facial graph.

2.2 Graph Generation

Graph generation is a crucial problem in many areas, such as generating social networks, drug discovery, designing electric circuits, and so on. Most previous methods [15, 18, 19, 27] focus on applying generative models, *i.e.* variational autoencoders and generative adversarial networks to predict a probabilistic fully-connected graph. Since our goal is to apply graph generation for head pose estimation, we relate our work to the domain-specific graph generation literature, where previous methods mainly use two-step approaches. For example, DGR [24] applies a Spatial Location Regression Net to regress the position of nodes directly and generate the feature vector of each node from the feature map by bilinear interpolation according to spatial coordinates. Then a Gaussian kernel is used to generate the adjacent matrix according to the spatial location of nodes. Kim *et al.* [13] propose a dynamic programming framework to build the graph structure in an online manner by incorporating the adjacency matrix in the graph theory to propagate message through the known structure of the graph. In this paper, we aim to generate a static and robust facial graph for head pose estimation.

2.3 Head Pose Benchmarks

Recently, researchers have been putting their efforts into collecting the images of humans and head poses in the wild or in the lab setting. Zhu *et al.* [36] expand 61, 225 samples across large poses in the 300W dataset [26] with flipping to 122, 450 samples for synthesizing the 300W across Large Poses (300W-LP). The AFLW2000 dataset [36] is proposed to include large pose variations with various illumination conditions and expressions, where the first 2, 000 images of the AFLW dataset [16] with ground-truth 3D faces and the corresponding 68 landmarks are provided. The BIWI dataset [6] is recorded with a Kinect sensor in the controlled environment,

containing 24 videos of 20 different subjects (14 men, 6 women, 4 people with glasses), *i.e.* totally 15, 000 frames. To get rid of errors and noises in the ground truth annotations due to various difficulties in ground truth collection, a large-scale synthetic head pose dataset, SynHead [7], is created by gathering head motion tracks from the BIWI and the ETH dataset [3] and recording additional depth video sequences with the Kinect and SoftKinetic sensors. After failure cases are discarded by manual inspection, the SynHead contains 10 subjects, 70 motion tracks, and 510,960 frames in total.

However, existing datasets [3, 6, 7, 16, 36] share the fundamental problem that the collected samples and the ground truth annotations have the underlying distribution bias for three directions of pose angles. For data-driven learning, the existing biased distribution has a negative impact on the performance of the trained model. To address this problem, we propose the UnBiased Head Pose Dataset by uniformly sampling the samples from the existing head pose benchmarks according to the density histogram for each pose angle, called UBHPD. Taking the biased underlying distribution difference into account, we propose an unbiased metric, namely UBMAE, to have a balanced evaluation of each angle.

3 METHOD

Overview. Head pose estimation aims at predicting head poses given an image. We present a new head pose detector, coined as One-step Graph Generation Netwrok (OsGG-Net), by viewing a face as a graph to extract its features. As shown in Figure 2, in our OsGG-Net, we take an image as input and feed it into a backbone to extract the spatial landmark heatmaps of a face. Then, we design three pipelines to perform head pose estimation in an end-to-end manner. 1) *Landmark Pipeline* is defined on the extracted landmark heatmaps. This Landmark Pipeline automatically localizes the face landmarks from an image. 2) *Graph Pipeline* is defined over all landmarks extracted from Landmark Pipeline. This Graph Pipeline tries to relate landmarks to generate a face graph for pose estimation. The generated face graph would capture the dependencies of each landmark on the estimated pose. 3) *Pose Pipeline* operates on the generated face graph. This pipeline focuses on extracting the spatial extent of the face graph, by directly regressing the yaw, pitch, and roll of head poses. These three pipelines collaborate together to yield head poses from an image in an end-to-end manner. **Backbone.** In our OsGG-Net, we input one RGB image with the resolution of $H_{img} \times W_{img} \times 3$, where 3 denotes the number of channels. This RGB image is fed into a Fully Convolutional Network [21] with a simple ResNet [9] backbone. Specifically, we choose ResNet-34 with light-weighted decoder architecture as our OsGG-Net feature backbone. This architecture employs an encoder-decoder architecture to extract heatmaps for each landmark. The extracted spatial heatmaps $\mathbf{H}$ are with the resolution of $K \times H \times W$, where K denotes the number of regressed landmarks. The extracted heatmaps are successively input into three pipelines. Next, we present the technical details of these pipelines.

3.1 Landmark Pipeline

The first pipeline is Landmark Pipeline, which is defined on the extracted landmark heatmaps $\mathbf{H}$ with $K \times H \times W$. This Landmark Pipeline automatically localizes the face landmarks from an image. Specifically, we apply an integral landmark regression to make it feasible to train our model in a differential way. Previous methods on keypoints heatmap regression need to prepare ground-truth gaussian heatmaps with a small radius, where the center value with 1 represents the location of the key points. Given a learned heat map $\mathbf{H}_k$ for kth keypoint, each location in the map represents the probability of the location being the key point. The final keypoint location coordinate $\mathbf{N}_k$ is obtained as the location $\mathbf{p}$ with the maximum likelihood as

$$\mathbf{N}_k = \arg\max_{\mathbf{p}} \mathbf{H}_k(\mathbf{p}). \tag{1}$$

This landmark regression method has two drawbacks. 1) It is non-differential. Since we are not able to use keypoints coordinates to train our model in an end-to-end manner, we need to regress gaussian heatmaps and generate landmarks by post-processing heatmaps according to Equation 1. 2) The keypoint localization precision is limited by the lower resolution of heatmaps than that of the original image due to down-sampling steps in a deep neural network. Using larger resolution of input images would bring more computer resources. In this work, in order to localize the face landmarks accurately, we employ the idea from integral pose regression on human pose estimation [28, 34] to replace the arg max in Equation 1 with the integration of all locations $\mathbf{p}$ in its domain Ω weighted by their probabilities. In this way, $\mathbf{J}_k$ is defined as

$$\mathbf{N}_k = \int_{\mathbf{p}\in\Omega} \mathbf{p} \cdot \tilde{\mathbf{H}}_k(\mathbf{p}), \tag{2}$$

where $\tilde{\mathbf{H}}_k$ denotes the normalized heatmaps. In order to make all elements of $\tilde{\mathbf{H}}_k$ non-negative and sum to one, we apply a softmax operator in this work to $\mathbf{H}_k(\mathbf{p})$ defined as

$$\tilde{\mathbf{H}}_k = \frac{e^{\mathbf{H}_k(\mathbf{p})}}{\int_{q\in\Omega} e^{\mathbf{H}_k(\mathbf{q})}} \tag{3}$$

For the discrete form in the generated heatmaps, Equation 2 is relaxed to

$$\mathbf{N}_k = \sum_{p_y=1}^{H} \sum_{p_x=1}^{W} \mathbf{p} \cdot \tilde{\mathbf{H}}_k(\mathbf{p}), \tag{4}$$

where H, W denote the height, width of heatmaps, respectively. As shown in Figure 2, applying the softmax operator to the extracted landmark heatmaps $K \times H \times W$ outputs x and y coordinate for each landmark. Then, we have a coordinate matrix $\mathbf{J}$ with $K \times 2$ generated from this landmark pipeline. For the landmark loss, we use the Mean Absolute Error (MAE) loss to regress landmarks directly for more accurate localization.

$$\mathcal{L}_{land} = \sum_{k=1}^{K} MAE(\mathbf{N}_k, \mathbf{T}_k), \tag{5}$$

where $\mathbf{N}_k, \mathbf{T}_k$ denote the prediction and ground-truth of the kth landmark, and K is the total number of landmarks.

3.2 Graph Pipeline

The second pipeline is Graph Pipeline, which is defined over all landmarks heatmaps $\mathbf{H}$ and the landmark coordinates matrix $\mathbf{J} \in \mathbb{R}^{K\times 2}$ extracted from Landmark Pipeline. This Graph Pipeline tries to relate landmarks to generate a face graph for pose estimation. The generated face graph captures the dependencies of each landmark on the estimated pose. In order to generate this desired face graph,

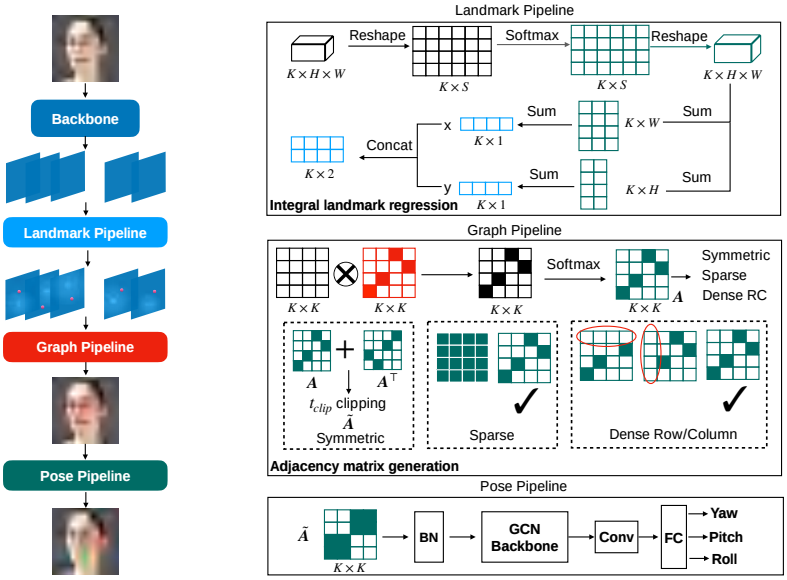

Figure 2: Illustration of OsGG-Net. On the left, we present the overall OsGG-Net framework. The blue cuboids represent the generated landmark heatmaps, the blue, sky blue, red, and green boxes denote the backbone, the Landmark Pipeline, the Graph Pipeline, and the Pose Pipeline. On the right, we show the detailed design of each pipeline.

we need to generate nodes and edges on each heatmap $\mathbf{H}_k$. For nodes in the graph, we take the predicted landmark spatial coordinates $\mathbf{J}$ as each node. Then we have K nodes in this desired graph. For generating edges in this graph, we first define the corresponding landmark feature maps $\mathbf{F}_k$ as nodes and generate edges $\mathbf{E}_k$ on each heatmap. The corresponding landmark feature maps $\mathbf{F}_k$ is defined as

$$\mathbf{F}_k = \mathbf{H}_k \circ \mathbf{J}, \tag{6}$$

where $\circ$ denotes matrix indexing, such that $\mathbf{F}_k \in \mathbb{R}^{K\times K}$.

Edges linking. In order to train our model in an end-to-end manner, we search S "alternative" nodes k_a^s among the left $K-1$ landmarks for a "master" node k_m on $\mathbf{F}_k$, where $s \in \{1, 2, ..., S\}$. Then we link S edges between the "master" node k_m and "alternative" nodes k_a^s. In this work, we define a mask matrix $\mathbf{M}_{search} \in \mathbb{R}^{K\times K}$ to constrain the search range of each "master" node on each landmark feature map $\mathbf{F}_k$. As a result, we get the valid landmark feature map $\tilde{\mathbf{F}}_k$ within our search range $\mathbf{M}_{search}$ as

$$\tilde{\mathbf{F}}_k = \mathbf{F}_k \odot \mathbf{M}_{search}, \tag{7}$$

where $\odot$ denotes the element-wise multiplication, and entries of $S \times S$ in $\mathbf{M}_{search}$ are 1 and others are 0.

Adjacent matrix generation. The adjacent matrix $\mathbf{A}$ is generated by applying a softmax operator to the $\tilde{\mathbf{F}}_k \in \mathbb{R}^{K\times K}$ along one dimension according to the spatial position of nodes. The generated facial graph has three properties, that is, being symmetric and sparse for the full matrix, but being dense for each row and column. 1) **Symmetric.** To assure it symmetric, we add $\mathbf{A}^\top$ to $\mathbf{A}$, and apply a clipping method to prune edges with small weights. The adjacency matrix $\tilde{\mathbf{A}}$ is defined as:

$$\tilde{\mathbf{A}} = \max\left(\frac{1}{2}(\mathbf{A} + \mathbf{A}^\top), t_{clip}\right), \tag{8}$$

where $\tilde{\mathbf{A}}$ is the $K \times K$ adjacency matrix, K is number of landmarks, t_{clip} is the threshold of clipping. 2) **Sparse.** In order to generate a sparse adjacency matrix for getting rid of redundancy entries in the adjacent matrix of a facial graph, we apply a ℓ_1 norm to each element of $\tilde{\mathbf{A}}$. The sparse loss is defined as

$$\mathcal{L}_{sparse} = \sum_{i=1}^{K}\sum_{j=1}^{K} |\tilde{\mathbf{A}}_{i,j}|, \tag{9}$$

where $|\cdot|$ denotes the absolute value of each element in $\tilde{\mathbf{A}}$. 3) **Dense RC.** Using the sparse loss, the model would collapse to the oversimplified solutions to make many zeros in the matrix, which is not desirable. Instead, we aim to make it *Dense* for entries in each Row and Column, namely, Dense RC. In this way, each "master" node links to its "important" nodes without missing all "alternative" nodes. Furthermore, to constrain the Dense RC of the generated adjacency matrix, we apply a dense loss to $\tilde{\mathbf{A}}$. And the dense loss is defined as

$$\mathcal{L}_{dense} = \sum_{i=1}^{K}|\sum_{j=1}^{K} \tilde{\mathbf{A}}_{i,j}|, \tag{10}$$

where $\sum_{j=1}^{K} \tilde{\mathbf{A}}_{i,j}$ is a $K \times 1$ matrix. Putting sparse loss and dense loss together, we define the overall graph loss as

$$\mathcal{L}_{graph} = \beta_s \mathcal{L}_{sparse} + \beta_d \mathcal{L}_{dense}, \tag{11}$$

where β_s, β_d denote the weight hyper-parameter of the sparse loss and dense loss. In order to explore how much each loss affects the final performance of our OsGG-Net, we perform extensive experiments on each parameter in Section 6.3.

3.3 Pose Pipeline

The third pipeline is Pose Pipeline, which operates on the generated face graph. This pipeline focuses on extracting the spatial extent of the face graph, by directly regressing the yaw, pitch, and roll of head poses. Specifically, we take the predicted landmark coordinates $\mathbf{J} \in \mathbb{R}^{K\times 2}$ as the node feature and feed it together with generated adjacency matrix $\tilde{\mathbf{A}}$ into the Pose Pipeline. First, we apply a batch normalization layer to $\mathbf{J}$, and then input the result together with $\tilde{\mathbf{A}}$ into 5 graph convolution layers. Finally, we use one 2D convolution layer and one fully connected layer to generate $\hat{\boldsymbol{y}}$ including yaw, pitch, roll. For accurate pose estimation, we adopt the MAE loss to regress poses directly, which is defined as

$$\mathcal{L}_{pose} = MAE(\hat{\boldsymbol{y}}, \boldsymbol{y}), \tag{12}$$

where $\boldsymbol{y}$ is the ground-truth of head poses.

Overall loss. Putting landmark, graph, and pose losses together, we define the overall loss of our OsGG-Net model as

$$\mathcal{L}_{total} = \alpha\mathcal{L}_{land} + \beta\mathcal{L}_{graph} + \gamma\mathcal{L}_{pose} \tag{13}$$

where α, β, γ denote the weight hyper-parameter of landmark, graph, and pose loss, respectively. In order to explore the effect of each loss on the final performance, we perform extensive experiments on each parameter for our ablation study in Section 6.1.

4 THE UNBIASED HEAD POSE DATASET

In this section, we describe the proposed UnBiased Head Pose Dataset by human-cleaning, namely UBHPD, in detail.

Existing dataset. There are four real-world datasets for head pose estimation, including BIWI [6], AFLW2000 [16], 300W-LP [36], and SynHead [7]. However, they all share the fundamental problem that samples have the underlying distribution bias for three directions of head poses. For data-driven learning, the existing biased distribution has a significantly negative impact on the learned model's performance. This is broadly acknowledged in previous works on object recognition [30] and facial expression recognition [8] but has not yet been addressed for head pose estimation.

To address this problem, we explore the underlying distribution histogram of existing head pose estimation benchmarks. Specifically, we calculate the counts of head pose frames, where three directions of head pose angles (yaw, pitch, roll) are within [-90°, +90°]. Then we split the range into U segments to visualize the density histogram for each head pose angle, as shown in Figure 3. In our case, we set $U = 15$ to cover the range of valid head pose angles within [-90°, +90°]. As can be seen, for the yaw angle, BIWI, AFLW2000, and SynHead have a large density within [-18°, +18°], while having a small density within [-90°, -54°] and [+54°, +90°]. The same density trend has been shown in the pitch and roll angle. Although 300W-LP has a "uniform" density within [-90°, -42°] and [+42°, +90°], the density of other segments are such low that the distribution of the yaw angle is distinct from that of pitch and roll angles on 300W-LP. Moreover, the yaw's distribution of 300W-LP dataset shows an obvious difference from that of BIWI and AFLW2000 datasets, which influences the cross-dataset evaluations.

UBHPD. In order to generate an unbiased head pose dataset, we sample the images from BIWI, AFLW2000, 300W-LP, UPNA and SynHead according to the density histogram for each direction of head pose angle in a Gaussian manner. As a result, the 15 segments (-90°, -78°, -66°, -54°, -42°, -30°, -18°, -6°, +6°, +18°, +30°, +42°, +54°, +66°, +78°, +90°) consist of 25,000 frames in total in our new dataset. After sampling the data, we visualize the density histogram of our new dataset for comparison with the existing datasets. As can be seen, our new dataset has an "unbiased" distribution for yaw, pitch, and roll such that we call our new dataset **UBHPD**. We split 70% of the total data as the training dataset and 30% as the test dataset. To the best of our knowledge, we are the first to propose an unbiased dataset in the head pose estimation community.

UBMAE. Most previous methods adopt the mean absolute error (MAE) for head pose estimation. However, this metric ignores the underlying distribution difference between the source and the target dataset. For example, the MAE metric would be biased to the angles with more samples in the source dataset, while our UBMAE is still balanced to the pose angles with even rare samples. Taking the underlying distribution difference between S angle segments into account, we define the UBMAE metric as

$$UBMAE = \frac{1}{3U}\left(\sum_{i=1}^{U}\sum_{j=1}^{3}\frac{d_i^j}{s_i^j}(\hat{y}-y)\right), \tag{14}$$

where we denote i, j for the index of the segment sets, and three directions of head pose angles, $\hat{y}, y$ for the prediction and the ground-truth of the jth head pose angle within the ith segment set, and d_i^j, s_i^j for the density of the jth angle within the ith segment set in the target and source dataset, respectively. This means that $d_i^j = s_i^j$ if the training set and test set ideally share the same underlying distribution of each angle within given U segments.

5 EXPERIMENTS

5.1 Experiments Settings

Datasets. We adopt three biased datasets for head pose estimation in the experiments: the 300W-LP, AFLW2000, BIWI, UPNA datasets, and our proposed UBHPD dataset. The 300W across Large Poses (300W-LP) dataset is synthesized by Zhu *et al.* [36] with expanding 61,225 samples across large poses in the 300W dataset [26] with flipping to 122,450 samples. The AFLW2000 dataset [16] provides ground-truth 3D faces and the corresponding 68 landmarks for the first 2,000 images of the AFLW dataset, where the faces in the dataset have large pose variations with various illumination conditions and expressions. The BIWI Kinect Head Pose Database [6] contains 24 videos of 20 subjects in the controlled environment. There are a total of roughly 15,000 frames in the dataset. In addition to RGB frames, the dataset also provides the depth image for each frame. The UPNA Head Pose Database [2] contains 10 subjects, where each user has 12 videos and 300 frames per video. Our proposed UBHPD dataset is an unbiased version of all existing datasets in the head pose estimation community. See more details about our new unbiased dataset in Section 4.

Protocols. 1) **Protocol 1:** Following Hopenet [25] and FSA-Net [32], we train on the synthetic 300W-LP dataset while testing on the two real-world datasets, the AFLW2000 and BIWI datasets. Notice that, when evaluating on the BIWI dataset, we only consider frames with pose angles within the range of [-99°, +99°] with MTCNN [35] face detection, following Hopenet [25] and FSA-Net [32]. We compare several state-of-the-art landmark-based pose estimation methods using this protocol. The batch size we used for this protocol is 16. 2)

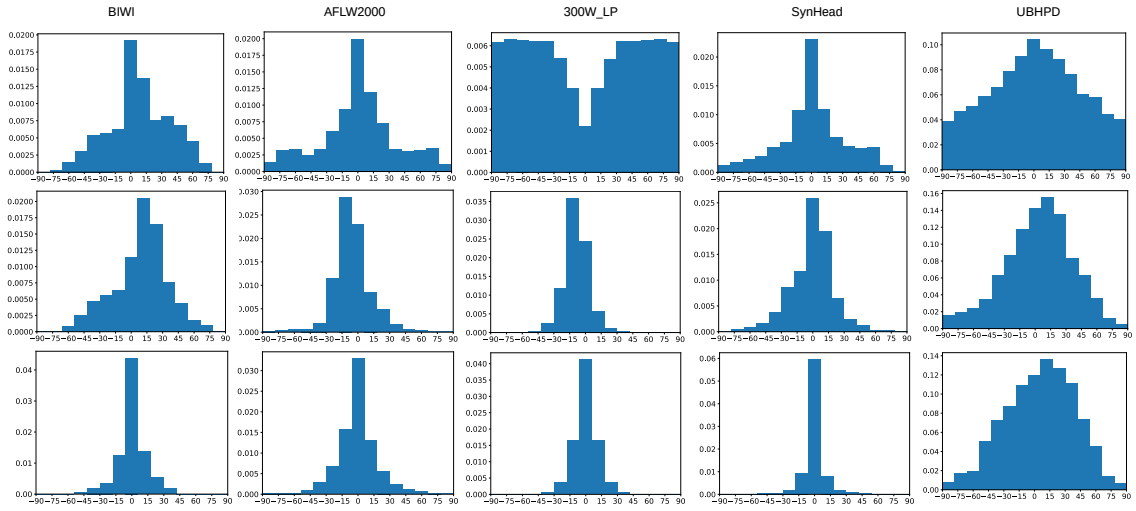

Figure 3: Distribution of BIWI, AFLW2000, 300W-LP, SynHead, and UBHPD. Top, middle, and bottom row denote yaw, pitch and roll, respectively.

Protocol 2: For BIWI dataset, we use 70% of videos (16 videos) in the BIWI dataset for training, and the others (8 videos) for testing. The faces in the BIWI dataset are detected by MTCNN with the empirical tracking technique to avoid failure of face detection. We used a batch size of 8 for training in this metric. For UPNA dataset, we use 70% of subjects (7 subjects) for training, and the others (4 subjects) for testing. 3) **Protocol 3:** For our new UBHPD dataset, we used 70% of the data in the dataset for training, and the others for testing. We excluded the target dataset in our UBHPD for cross-dataset evaluation settings.

Implementation Details. We use PyTorch [22] backend for implementing the proposed OsGG-Net [1]. For data augmentation in training, we closely follow FSA-Net [32] and apply random cropping and random scaling ($0.8 \sim 1.2$) to training images. We use 300 epochs to train the network with the Adam [14] optimizer with the initial learning rate of 0.001. All experiments are conducted on one NVIDIA Titan RTX GPU. The total training time is 8.3 hours, and the inference time of our model is 2.5 ms per image.

5.2 Comparison with State-of-the-art Methods

In this section, we implement extensive experiments to compare the proposed OsGG-Net with state-of-the-art methods.

Protocol 1. Following HopeNet [25] and FSA-Net [32], we train on the synthetic 300W-LP dataset while testing on the two real-world datasets, the AFLW2000 and BIWI datasets. We report the comparison results of both MAE and UBMAE on the AFLW2000 and BIWI datasets in Table 1. When transferred to the AFLW2000 and BIWI datasets, our OsGG-Net achieves the best performance in terms of both metrics. When evaluating on the AFLW2000 and BIWI dataset, img2pose [1] using the large face detection benchmark [31] for pre-training achieves MAE=4.27 (Yaw:4.41, Pitch:5.57, Roll:2.82) and MAE=3.90 (Yaw:3.97, Pitch:5.27, Roll:2.46), respectively. Although we do not pre-train our model on auxiliary datasets, we can achieve comparable results to their performance. This infers the excellent generalizability of the proposed OsGG-Net.

[1]The code and dataset are released at https://github.com/stoneMo/OsGG-Net.

Table 1: Comparisons with the state-of-the-art methods on the AFLW2000 and BIWI dataset evaluated on Protocol 1. Bold and underline denote the first and second place.

Methods	300W-LP → *AFLW2000*					300W-LP → *BIWI*				
	Yaw	Pitch	Roll	MAE	UBMAE	Yaw	Pitch	Roll	MAE	UBMAE
Dlib (68 points) [12]	23.10	13.60	10.50	15.80	19.60	16.80	13.80	6.19	12.20	23.92
FAN (12 points) [4]	6.36	12.30	8.71	9.12	11.71	8.53	7.48	7.63	7.89	16.31
Landmarks [25]	5.92	11.86	8.27	8.65	11.14	4.87	9.85	7.38	7.37	9.52
3DDFA [36]	5.40	8.53	8.25	7.39	9.73	36.20	12.30	8.78	19.10	35.21
Hopenet ($\alpha = 2$) [25]	6.47	6.56	5.44	6.16	7.87	5.17	6.98	3.39	5.18	10.58
Hopenet ($\alpha = 1$) [25]	6.92	6.64	5.67	6.41	8.20	4.81	6.61	3.27	4.90	10.02
SSR-Net-MD [33]	5.14	7.09	5.89	6.01	7.80	4.49	6.31	3.61	4.65	9.93
HPE [11]	4.80	6.18	4.87	5.28	6.78	<u>3.12</u>	5.18	4.57	4.29	9.23
FSA-Net [32]	4.50	6.08	4.64	5.07	6.50	4.27	4.96	**2.76**	4.00	<u>8.13</u>
QuatNet [10]	**3.92**	**5.62**	3.97	<u>4.50</u>	<u>5.74</u>	**2.94**	5.49	4.01	4.15	8.88
TriNet [5]	4.04	5.77	4.20	4.67	5.96	4.11	**4.76**	<u>3.05</u>	<u>3.97</u>	8.15
OsGG-Net (ours)	<u>3.96</u>	<u>5.71</u>	**3.51**	**4.39**	**5.53**	3.26	<u>4.85</u>	3.38	**3.83**	**8.05**

Protocol 2. In this protocol, 70% of videos are used for training (16 videos for BIWI, 7 subjects for UPNA) and 30% for testing (8 videos for BIWI, 3 subjects for UPNA). The comparison results with state-of-the-art methods on the BIWI and UPNA datasets are reported in Table 2. When testing on the BIWI and UPNA dataset, our OsGG-Net outperforms FSA-Net [32] and Tri-Net [5] in terms of MAE and UBMAE, which further validates the effectiveness of our OsGG-Net.

Protocol 3. In Table 3, we compare our OsGG-Net with state-of-the-art methods in terms of both MAE and UBMAE metrics on the proposed UBHPD dataset. All are trained on the UBHPD-train and tested on the UBHPD-test. We can observe that the difference ($\triangle$) between MAE and UBMAE is smaller than the two aforementioned protocols, which shows the "unbiased" distribution lying in our proposed UBHPD. In this protocol, our OsGG-Net achieves the best results in terms of MAE and UBMAE. This also demonstrates the advantage of our OsGG-Net over current methods.

Table 2: Comparisons with state-of-the-art methods on the BIWI and UPNA dataset evaluated on Protocol 2.

Methods	BIWI-*train* → BIWI-*test*					UPNA-*train* → UPNA-*test*				
	Yaw	Pitch	Roll	MAE	UBMAE	Yaw	Pitch	Roll	MAE	UBMAE
SSR-Net-MD [33]	4.24	4.35	4.19	4.26	5.18	4.56	4.85	4.96	4.79	6.11
VGG16 [7]	3.91	4.03	3.03	3.66	4.41	4.08	3.95	3.27	3.77	4.60
VGG16+RNN [7]	3.14	3.48	2.60	3.07	3.71	3.23	3.67	2.75	3.22	4.03
FSA-Net [32]	**2.89**	4.29	3.60	3.60	4.37	**3.05**	4.53	3.34	3.64	4.80
TriNet [5]	2.93	**3.04**	2.44	2.80	3.39	3.15	**3.24**	2.55	2.98	3.67
OsGG-Net (ours)	2.95	3.01	2.24	**2.73**	**3.30**	3.14	3.32	**2.32**	**2.93**	**3.58**

Table 3: Comparisons with state-of-the-art methods on the UBHPD dataset using Protocol 3.

Method	Yaw	Pitch	Roll	MAE	UBMAE	$\triangle(UBMAE - MAE)$
SSR-Net-MD [33]	5.05	5.53	5.12	5.23	5.44	0.21
VGG16 [7]	4.63	4.25	3.62	4.17	4.33	0.16
VGG16+RNN [7]	3.45	3.75	2.85	3.35	3.48	0.13
FSA-Net [32]	3.52	4.72	3.45	3.90	4.04	0.14
TriNet [5]	**3.23**	3.45	2.63	3.10	3.22	**0.12**
OsGG-Net (ours)	3.25	**3.36**	**2.53**	**3.05**	**3.17**	**0.12**

Cross-dataset Evaluation. We evaluate our OsGG-Net and previous work trained on 300W-LP and UBHPD separately on the BIWI dataset in Table 4. When trained on UBHPD, all methods perform better than the one trained on 300W-LP, which validates the effectiveness of our UBHPD in cross-dataset settings. In this cross-dataset setting, our OsGG-Net outperforms FSA-Net [32] and Tri-Net [5] by a large margin (0.19 and 0.21) in terms of MAE, which further shows our method's decent generalization capability.

Table 4: Comparisons with two different source datasets (300W-LP and UBHPD) for the cross-dataset settings.

Method	Source	Target	Yaw	Pitch	Roll	MAE	UBMAE
Hopenet [25]	300W-LP	BIWI	4.81	6.61	3.27	4.90	10.02
FSA-Net [32]	300W-LP	BIWI	4.27	4.96	**2.76**	4.00	8.13
Tri-Net [5]	300W-LP	BIWI	4.11	**4.76**	3.05	3.97	8.15
OsGG-Net(ours)	300W-LP	BIWI	**3.26**	4.85	3.38	**3.83**	**8.05**
Hopenet [25]	UBHPD	BIWI	4.65	6.45	3.23	4.78(↓ 0.12)	6.48(↓ 3.54)
FSA-Net [32]	UBHPD	BIWI	4.06	4.63	**2.52**	3.74(↓ 0.26)	5.05(↓ 3.08)
Tri-Net [5]	UBHPD	BIWI	3.96	**4.43**	2.88	3.76(↓ 0.21)	5.12(↓ 3.03)
OsGG-Net(ours)	UBHPD	BIWI	**3.02**	4.58	3.06	**3.55**(↓ 0.28)	**4.92**(↓ 3.13)

5.3 Visualization

In Figure 4, we visualize the qualitative examples of head pose estimation and generated face graphs on 300W-LP, AFLW2000, BIWI, and UNPA datasets. By comparison, our OsGG-Net model achieves competitive head pose estimation performance. Some failure cases are reported on the Bottom Row in the Figure 4. Our model sometimes misses the cases where there exists a large area of occlusion on the human face. For visualization on the robust facial graph, we also show the generated facial graphs used for head pose estimation on the BIWI dataset, as shown in Figure 5. As can be seen, the generated facial graphs have decent robustness to both the same pose angles of different subjects (Top Row) and different head pose angles for the same subject (Bottom Row).

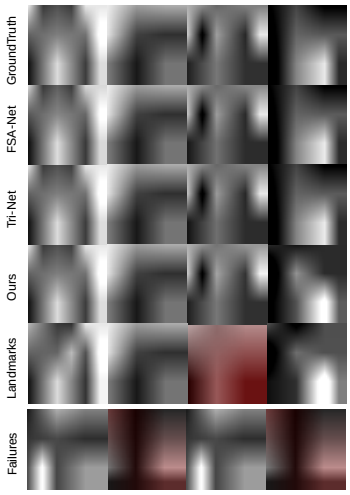

Figure 4: Visualization results. The *blue, green, red* lines denote the yaw, pitch, and roll. Failures Row: left two for predictions, right two for GT. Best viewed on screen.

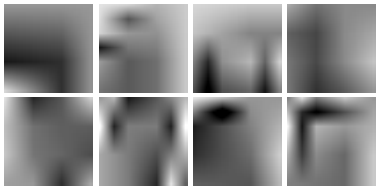

Figure 5: Visualization of the generated facial graph (Top Row: the same head pose angles of different subjects, Bottom Row: different pose angles for the same subject).

6 ABLATION STUDY

In this section, we explore extensive ablation studies on each part of our OsGG-Net, including landmark regression in Landmark Pipeline, edge linking strategies and adjacency matrix generation in Graph Pipeline, and the Pose Pipeline. Unless specified, all models for our ablation study are trained on the proposed UBHPD dataset, and then tested on BIWI dataset.

6.1 Effect of each loss (landmark, graph, pose)

In this section, we explore how much each of the three proposed losses affects the final performance of head pose estimation, as shown in Table 5. We can observe the obvious performance drop without using the proposed landmark loss and graph loss. When increasing the weighting coefficient α and β, we can achieve better results in terms of MAE and UBMAE. This demonstrates the importance of each loss proposed in our OsGG-Net.

Furthermore, we conduct extensive experiments to explore the sparse and dense loss used in Graph Pipeline in Table 6. As can be seen, with the rise of the weight importance of the sparse and dense loss, our method's performance increase and then decrease. This result further validates the rationality of introducing the sparse loss and dense loss to constrain the sparsity and denseness of the generated facial graphs.

Table 5: Exploration study on the importance weight of each designed loss. + denotes if the loss is used or not; ++ is used for the increase of loss weight (α, β, γ) by a factor of 10.

$\mathcal{L}_{Land}$	$\mathcal{L}_{Graph}$	$\mathcal{L}_{Pose}$	Yaw	Pitch	Roll	MAE	UBMAE
+		+	3.45	4.79	3.48	3.91 (↑0.36)	5.41 (↑0.49)
	+	+	3.31	4.65	3.34	3.77 (↑0.22)	5.22 (↑0.30)
+	+	+	3.02	4.58	3.06	3.55 (–)	4.92 (–)
++	+	+	3.01	4.52	3.03	3.52 (↓0.03)	4.87 (↓0.05)
+	++	+	2.97	4.36	3.08	3.47 (↓0.08)	4.81 (↓0.11)
++	++	+	**2.93**	**4.32**	**3.01**	**3.42** (↓0.13)	**4.74** (↓0.18)

Table 6: Exploration study on sparse loss and dense loss.

β_{sparse}	β_{dense}	Yaw	Pitch	Roll	MAE	UBMAE
1	1	2.93	4.32	3.01	3.42 (–)	4.74 (–)
1	5	2.88	4.27	2.96	3.37 (↓0.05)	4.67 (↓0.07)
1	10	**2.86**	**4.21**	**2.95**	**3.34** (↓0.08)	**4.63** (↓0.11)
1	15	2.96	4.39	3.15	3.50 (↑0.08)	4.86 (↑0.12)
1	20	3.28	4.68	3.46	3.81 (↑0.39)	5.28 (↑0.54)
5	1	2.91	4.28	3.13	3.44 (↑0.02)	4.78 (↑0.04)
10	1	2.85	4.21	3.03	3.36 (↓0.06)	4.67 (↓0.07)
15	1	3.18	4.49	3.28	3.65 (↑0.23)	5.06 (↑0.32)
20	1	3.46	4.62	3.45	3.84 (↑0.42)	5.32 (↑0.58)

6.2 Number and search range of nodes

In this part, we first explore the number of landmarks (nodes), *i.e.* K, used in our OsGG-Net. In order to pick the salient and stable landmarks, we first select 34 candidates by calculating the moving distances of landmark locations along with the head pose changes for the identical person. Then we compare the stableness of these candidates for the same head pose from diverse people and set $K = 19$ in our experiments. Using $K = 68, 34$, our model achieves worse performance (K=68: MAE=4.10, UBMAE=5.71; K=34: MAE=3.63, UBMAE=5.05;) than $K = 19$ in terms of MAE and UBMAE. See more results in the supplementary.

In order to analyze the effect of the node search range, *i.e.* S, on linking edges, we set the number of the search range of nodes to 3, 5, 7, 9, 18 empirically. We report the quantitative results in Table 7. With the increase of the search range of nodes for linking edges of facial graphs dynamically, our method's performance rises and drops then. This is because a large search range makes the network hard to generate a facial graph robust to predicting head pose angles.

Table 7: Study on the search range of nodes to link edges.

Search range (S)	Yaw	Pitch	Roll	MAE	UBMAE
3	2.86	4.21	2.95	3.34 (–)	4.63 (–)
5	2.81	4.16	2.87	3.28 (↓0.06)	4.54 (↓0.09)
7	**2.75**	**4.13**	**2.83**	**3.24** (↓0.10)	**4.48** (↓0.15)
9	2.93	4.25	2.99	3.39 (↑0.05)	4.70 (↑0.07)
18	3.15	4.47	3.23	3.62 (↑0.28)	5.01 (↑0.38)

6.3 Adjacency matrix generation

In Table 8, we explore the threshold of clipping, t_{clip}, used for adjacency matrix generation in Graph Pipeline by experimenting with different numerical values. We can observe that with the incorporation of a threshold to clip the importance score in the adjacency matrix, our method improves in terms of MAE and UBMAE, which indeed validates the importance of the generated adjacency matrix for estimating head poses.

Table 8: Exploration study on adjacency matrix generation.

t_{clip}	Yaw	Pitch	Roll	MAE	UBMAE
0	2.75	4.13	2.83	3.24 (–)	4.48 (–)
0.05	2.71	4.08	2.78	3.19 (↓0.05)	4.42 (↓0.06)
0.10	**2.69**	**4.06**	**2.75**	**3.17** (↓0.07)	**4.38** (↓0.10)
0.15	2.77	4.16	2.83	3.25 (↑0.01)	4.50 (↑0.02)
0.16	2.87	4.21	2.85	3.31 (↑0.07)	4.58 (↑0.10)

6.4 Pose Pipeline

In order to better understand the influence of the Pose Pipeline on the final performance of our OsGG-Net, we further explore the number of graph convolution networks (GCN) used in this pipeline. The quantitative results are reported in Table 9. When increasing the number of GCNs to 5, we achieve the best performance in terms of MAE and UBMAE. However, the method's performance drops with a large number of GCNs used in Pose Pipeline. This is due to the vanishing gradient problem lying in deeper GCN models.

Table 9: Exploration study on Pose Pipeline.

Pose decoder	Yaw	Pitch	Roll	MAE	UBMAE
GCN ×3	2.69	4.06	2.75	3.17 (–)	4.38 (–)
GCN ×5	**2.63**	**3.98**	**2.71**	**3.11** (↓0.06)	**4.30** (↓0.08)
GCN ×7	2.75	4.15	2.87	3.26 (↑0.09)	4.51 (↑0.13)
GCN ×9	2.96	4.56	3.02	3.51 (↑0.34)	4.86 (↑0.48)
GCN ×11	3.01	4.69	3.28	3.66 (↑0.49)	5.08 (↑0.70)

7 CONCLUSION

In this paper, we propose OsGG-Net, a One-step Graph Generation Network for estimating head poses from a single image. Our OsGG-Net consists of landmark, graph, and pose pipelines. An integral landmark regression in the landmark pipeline is applied to localize the face landmarks automatically, then the graph pipeline is used to generate a face graph for modeling the complex nonlinear relationships between the geometric distribution of landmarks and head poses, and finally the pose pipeline is proposed to extract the spatial extent of the face graph, by directly regressing the yaw, pitch, and roll of head poses. Furthermore, we also propose the UBHPD by human-cleaning, and a new unbiased metric, namely UBMAE, for unbiased head pose estimation, to address the biased underlying distribution issues lying in current benchmarks. We conduct extensive experiments on various benchmarks and UBHPD where our method achieves the state-of-the-art results in terms of the commonly-used MAE metric and our proposed UBMAE. Comprehensive ablation studies also demonstrate the effectiveness of each part of our approach.

ACKNOWLEDGMENTS

This work was supported in part by National Natural Science Foundation of China (No. 61906195). The work was also sponsored by CCF-Baidu open fund.

REFERENCES

[1] Vítor Albiero, Xingyu Chen, Xi Yin, Guan Pang, and Tal Hassner. 2021. img2pose: Face Alignment and Detection via 6DoF, Face Pose Estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 7617–7627.

[2] Mikel Ariz, José J. Bengoechea, Arantxa Villanueva, and Rafael Cabeza. 2016. A novel 2D/3D database with automatic face annotation for head tracking and pose estimation. *Computer Vision and Image Understanding* 148 (2016), 201–210.

[3] Michael D. Breitenstein, Daniel Kuettel, Thibaut Weise, Luc van Gool, and Hanspeter Pfister. 2008. Real-time face pose estimation from single range images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 1–8.

[4] Adrian Bulat and Georgios Tzimiropoulos. 2017. How far are we from solving the 2d and 3d face alignment problem?(and a dataset of 230,000 3d facial landmarks). In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 1021–1030.

[5] Zhiwen Cao, Zongcheng Chu, Dongfang Liu, and Yingjie Chen. 2021. A Vector-Based Representation to Enhance Head Pose Estimation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. 1188–1197.

[6] Gabriele Fanelli, Matthias Dantone, Juergen Gall, Andrea Fossati, and Luc Van Gool. 2013. Random forests for real time 3d face analysis. *International Journal of Computer Vision* 101, 3 (2013), 437–458.

[7] Jinwei Gu, Xiaodong Yang, Shalini De Mello, and Jan Kautz. 2017. Dynamic Facial Analysis: From Bayesian Filtering to Recurrent Neural Network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 1531–1540.

[8] Byungok Han, Woo-Han Yun, Jang-Hee Yoo, and Won Hwa Kim. 2020. Toward Unbiased Facial Expression Recognition in the Wild via Cross-Dataset Adaptation. *IEEE Access* 8 (2020), 159172–159181.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 770–778.

[10] Heng-Wei Hsu, Tung-Yu Wu, Sheng Wan, Wing Hung Wong, and Chen-Yi Lee. 2019. QuatNet: Quaternion-Based Head Pose Estimation With Multiregression Loss. *IEEE Transactions on Multimedia* 21, 4 (2019), 1035–1046.

[11] Bin Huang, Renwen Chen, Wang Xu, and Qinbang Zhou. 2020. Improving head pose estimation using two-stage ensembles with top-k regression. *Image and Vision Computing* 93 (2020), 103827.

[12] Vahid Kazemi and Josephine Sullivan. 2014. One millisecond face alignment with an ensemble of regression trees. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 1867–1874.

[13] Daesik Kim, YoungJoon Yoo, Jeesoo Kim, Sangkuk Lee, and Nojun Kwak. 2018. Dynamic Graph Generation Network: Generating Relational Knowledge from Diagrams. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 4167–4175.

[14] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[15] Thomas N Kipf and Max Welling. 2016. Variational graph auto-encoders. In *Proceedings of the Advances in Neural Information Processing Systems Workshops (NeurIPSW)*.

[16] Martin Köstinger, Paul Wohlhart, Peter M. Roth, and Horst Bischof. 2011. Annotated Facial Landmarks in the Wild: A large-scale, real-world database for facial landmark localization. In *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCVW)*. 2144–2151.

[17] Amit Kumar, Azadeh Alavi, and Rama Chellappa. 2017. KEPLER: keypoint and pose estimation of unconstrained faces by learning efficient H-CNN regressors. In *Proceedings of the IEEE International Conference on Automatic Face Gesture Recognition (FG)*. 258–265.

[18] Renjie Liao, Yujia Li, Yang Song, Shenlong Wang, Will Hamilton, David K Duvenaud, Raquel Urtasun, and Richard Zemel. 2019. Efficient Graph Generation with Graph Recurrent Attention Networks. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*.

[19] Qi Liu, Miltiadis Allamanis, Marc Brockschmidt, and Alexander Gaunt. 2018. Constrained Graph Variational Autoencoders for Molecule Design. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*.

[20] Zhaoxiang Liu, Zezhou Chen, Jinqiang Bai, Shaohua Li, and Shiguo Lian. 2019. Facial Pose Estimation by Deep Learning from Label Distributions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*. 1232–1240.

[21] Jonathan Long, Evan Shelhamer, and Trevor Darrell. 2015. Fully Convolutional Networks for Semantic Segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 3431–3440.

[22] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*.

[23] Rajeev Ranjan, Vishal M. Patel, and Rama Chellappa. 2019. HyperFace: A Deep Multi-Task Learning Framework for Face Detection, Landmark Localization, Pose Estimation, and Gender Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41, 1 (2019), 121–135.

[24] Min Ren, Yunlong Wang, Zhenan Sun, and Tieniu Tan. 2020. Dynamic Graph Representation for Occlusion Handling in Biometrics. *Proceedings of the AAAI Conference on Artificial Intelligence*, 11940–11947.

[25] Nataniel Ruiz, Eunji Chong, and James M. Rehg. 2018. Fine-Grained Head Pose Estimation Without Keypoints. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 2155–215509.

[26] Christos Sagonas, Georgios Tzimiropoulos, Stefanos Zafeiriou, and Maja Pantic. 2013. 300 Faces in-the-Wild Challenge: The First Facial Landmark Localization Challenge. In *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCVW)*. 397–403.

[27] Martin Simonovsky and Nikos Komodakis. 2018. GraphVAE: Towards Generation of Small Graphs Using Variational Autoencoders. *arXiv preprint arXiv:1802.03480* (2018).

[28] Xiao Sun, Bin Xiao, Fangyin Wei, Shuang Liang, and Yichen Wei. 2018. Integral Human Pose Regression. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 536–553.

[29] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going deeper with convolutions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 1–9.

[30] Antonio Torralba and Alexei A. Efros. 2011. Unbiased look at dataset bias. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 1521–1528.

[31] Shuo Yang, Ping Luo, Chen-Change Loy, and Xiaoou Tang. 2016. WIDER FACE: A Face Detection Benchmark. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 5525–5533.

[32] Tsun-Yi Yang, Yi-Ting Chen, Yen-Yu Lin, and Yung-Yu Chuang. 2019. 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 (CVPR)*. 1087–1096.

[33] Tsun-Yi Yang, Yi-Hsuan Huang, Yen-Yu Lin, Pi-Cheng Hsiu, and Yung-Yu Chuang. 2018. SSR-Net: A Compact Soft Stagewise Regression Network for Age Estimation. In *Proceedings of the International Joint Conference on Artificial Intelligence, (IJCAI)*. 1078–1084.

[34] Kwang Moo Yi, Eduard Trulls, Vincent Lepetit, and Pascal Fua. 2016. LIFT: Learned Invariant Feature Transform. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 467–483.

[35] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. 2016. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters* 23, 10 (2016), 1499–1503.

[36] Xiangyu Zhu, Zhen Lei, Xiaoming Liu, Hailin Shi, and Stan Z. Li. 2016. Face Alignment Across Large Poses: A 3D Solution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 146–155.