Proceedings of 10th IOE Graduate Conference
Peer Reviewed
ISSN: 2350-8914 (Online), 2350-8906 (Print)
Year: 2021 Month: October Volume: 10

Head Pose Estimation by Few Shot Learning Techniques

Manoj Joshi [a], Dibakar Raj Pant [b], Bikram Acharya [c]

[a, b, c] *Department of Electronics and Computer Engineering, IOE, Pulchowk Campus*
Corresponding Email: [a] 075mscsk009.manoj@pcampus.edu.np, [b] drpant@ioe.edu.np, [c] 075mscsk004.bikram@pcampus.edu.np

Abstract
Head pose estimation is used in a variety of human-computer interface applications, like stare tracking, driving assistance, impaired assistance and entertainment. Other range of application include biometric recognition, mainly in video surveillance. The advancement in convolutional neural networks has significantly improved the performance of head pose estimation in recent works. However, difficulties in capturing well labelled head pose data, massive training time and differences in the facial features of different persons makes them difficult to use. The proposed work addresses the problem of estimating the head pose from RGB images. An approach to learn latent representation of head pose features has been obtained using ResNet-50 architecture. BIWI Kinect head pose dataset has been used to train the network. The latent embeddings are further passed into a quick, adaptable head pose estimator trained using one shot learning in few shot settings. A mean absolute error (MAE) of 6.405 has been achieved for five-way one-shot settings in predicting the 3D head pose angles (yaw, pitch and roll).

Keywords
Head Pose Estimation, Few Shot Learning, Deep Learning, Resnet

1. Introduction

In recent times, a lot of progress has been made in the field of image processing with effective application in day to day activities. One such application of image processing is head pose estimation, used in variety of human-machine interface applications and tasks like including stare tracking, driving assistance, impaired assistance and entertainment, robotics, VR, driving assistance,etc. Head posture is generally used for understanding human focus, actions, or intentions, and has been widely researched and explored in the cognitive psychology and neurophysiology communities. Head pose estimation is generally concerned with predicting the Euler angles of a human head which predicts yaw and pitch angles in 2D space and optionally a roll angle is also predicted for analyzing the 3D space [1]

Table 1: Average Head movement

Angles	Values
Yaw	79.8° to +75.3°
Pitch	60.04° to +69.6°
Roll	40.9° to +36.3°

The task of estimating these angles is difficult due to the wide variance in head appearance features such as facial expressions, race, gender and many other environmental factors such as occlusion, noisy images and varying illumination. While some success have been achieved in estimation of Euler angles for head pose estimation using traditional approach as histogram of oriented gradients(HOG) [2], modern deep learning have been used due to robustness and flexibility of model. The high efficiency of deep learning model is highly reliant on training a network with a wide number of named instances with a variety of visual variations. Modern deep learning methods should be able to achieve the high proficiency by rapidly understanding and adapting from a few examples and continuing to adapt as more evidence becomes available. This type of rapid and scale able learning is difficult since the agent must combine previous knowledge with a limited volume of new information while avoiding over-fitting to the new details but fails to generalize for the new tasks. Recently, proposed meta-learning algorithms [3] are used to rapidly learn a new task from limited amount of data. A few shot learner is used to train model and is capable of learning from large amount of new

different tasks and the model performs optimally on a new set of tasks when the modified parameters are computed after few more gradient steps using a limited number of data from the new task[4]. However, meta-learning algorithms do not increase the number of learned parameters or impose restrictions on the model architecture. As the work requires huge dataset, there is a challenge of high computation power and memory in training the model. Working under few shot setting, we can use very few samples to learn important features and then use that knowledge to adapt to novel tasks much more quickly and efficiently and also infers very quickly on unseen tasks.

2. Related Literature

Some early method for estimating head pose are based on appearance template methods [5] which used image based comparison metrics. The query image of a person is matched with some templates having head pose labels and based on similarity, label is assigned. Detector arrays based method were developed for frontal face detection [6] where instead of directly comparing images to templates, it used detector trained on many images using supervised learning algorithms and simultaneously detected head and pose from the images. To train these model, we need to train many detectors needed to be trained for discrete poses. Nonlinear regression methods were used to detect head poses by learning a nonlinear function that can map an image space to one or more head pose direction as in literature [7]. High dimensional image data were handled by principal component analysis (PCA) whereas support vector regressor (SVR) and multi layer perceptron's (MLP) were used for learning nonlinear function. Nonlinear regression models used in detectors exhibited excellent results but they were prone to error if head localization was not properly achieved. Manifold embedding methods were proposed for CNN implementation, which considered high dimensional image space as low dimensional continuous manifolds, which preserves head pose variations and used for head pose estimation using regression models[8].

CNN based regression methods provides a high tolerance to shift and distortion variance. Osadchy et al. [9] proposed a real-time CNN based approach for head pose estimation to simultaneously detect face and head pose. Their work is based on CNN's and energy-based models. Their CNN architecture is similar to LeNet-5 [10] but had more feature maps. They claimed to have an accuracy of over 85% in yaw angle range and 95% accuracy for in-plane rotation. In 2014, Ahn et al.[11] published worked with BIWI Kinect head pose dataset using four convolution layers and two fully connected layers to design their network. Their model used the RGB images and got very promising results.

Other methods proposed use of RGB images along with the depth information as seen in literature [12], where GoogleLeNet [13] used RGB and depth images to train the model. Venturelli et al. [14] proposed a shallow network with five convolutional layers and three fully connected layer with improved operformace over [12]. Ruiz et al. [15] used ResNet50 architecture with three mean squared error (MSE) and cross entropy loss for each head pose angles as evaluation metrices. Recent work on head pose estimation has been proposed by [16] on Prima and AFLW datasets.

These above stated literature with CNN based methods give excellent results for head pose estimation but requires a lot of training samples to find the best estimator, lack generalization in unknown task and have poor adaptation to new set of task.

3. Methodology

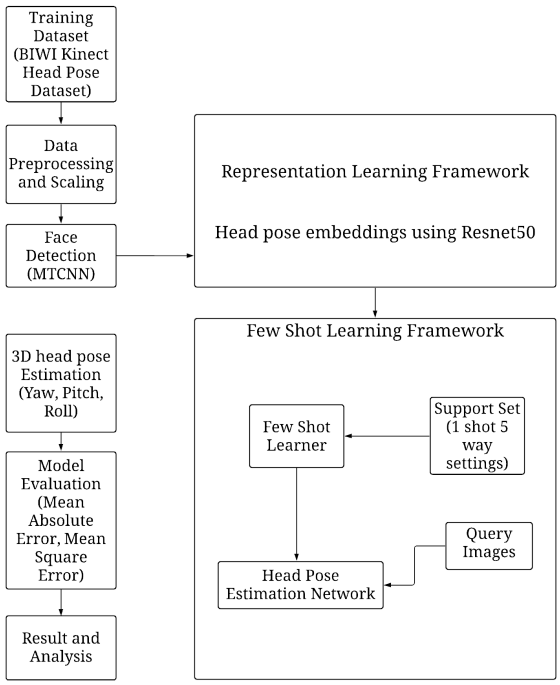

Figure 1: Methodology

The proposed methodology formulates how process of face detection, data preprocessing, embedding image features in latent space and training a robust one shot learner is achieved. Fig 1 shows the block diagram for one shot training and testing a head pose estimation framework.

3.1 Dataset Description

The research work uses well known BIWI Kinect Head Pose [17] benchmark dataset for 3D head pose estimation.

Figure 2: Example of images from Biwi Kinect Dataset [18]

Each image's ground truth is given in the form of the 3D position and rotation of the head. Fig 2 represents an example of BIWI dataset with head poses (pitch,yaw,roll)

Table 2: Dataset Description

Parameter	Values
Number of images	15678
Total person(Class)	20
Depth RGB image size	640 × 480 pixels
Head pose range (yaw)	±75°
Head pose range (pitch)	±60°(pitch)

3.2 Face Detection

Detecting face and face alignment is a difficult task due to challenges posed by varying lighting conditions, visual variations in human faces and extreme head pose variations. The task includes face detection, localization and computation of bounding box coordinates to get the exact coordinates of face. This work uses multi-task cascaded convolutional neural network (MTCNN) [19] for face classification, bounding box regression and facial landmark localization.

- **Proposal Network(P-net):** peforms a face detection and bounding box regression in input images.
- **Network (R-Net):** peforms finetuning to remove false features and performs calibration with bounding box regression
- **O-Net:** describe face in details to produce five facial landmark position

Face classification: A fully connected convolution neural network is trained to classify the face from given image. The learning objective of the CNN is two-class classification i.e. face or non-face. For a given sample x_i, a cross entropy loss is computed to classify the face. The cross-entropy loss is computed as:

$$L_i^{detector} = -(y_i^{detector} log(p_i)) + (1 - y_i^{detector})(1 - log(p_i) \quad (1)$$

where p_i is the probablity of sample being a face and $y_i^{detector}$ is the ground truth.

Bounding box regression: The second stage region network (R-Net) predicts an offset between real image and predicted candidate window. An Euclidean loss for given sample is computed using:

$$L_i^{box} = ||\hat{y}_i^{box} - y_i^{box}||_2^2 \quad (2)$$

where $\hat{y}_i^{box}$ is the regression result from R-Net and y_i^{box} is the ground truth. Bounding box y_i^{box} comprises four coordinates: left, top, height, width and hence $y_i^{box} \varepsilon \mathbb{R}^4$

Facial landmark localization: The third stage network (O-Net) describes detected face in more detail by learning a regression function shown below:

$$L_i^{landmark} = ||\hat{y}_i^{landmark} - y_i^{landmark}||_2^2 \quad (3)$$

where $\hat{y}_i^{landmark}$ are the facial landmarks coordinates predicted by the network and $y_i^{landmark}$ are the ground truth coordinates. Left eye, right eye, nose, left mouth corner and right mouth corner are the predicted facial landmarks. The dimension of landmarks is $y_i^{landmark} \varepsilon \mathbb{R}^{10}$ MTCNN uses multiple tasks on three different CNN's, the training samples are very different in each networks. Thus some of the above mentioned losses are skipped during training and are set to zero. The combined learning target used by MTCNN is computed as:

$$L_{mtcnn} = min \Sigma_{i=1}^{N} \Sigma_{j \varepsilon (detector, box, landmark)} \alpha_j \beta_i^j L_i^j \quad (4)$$

where N denotes total number of training samples. The value of α_j is choosen as per the importance of task. ($\alpha_{detector} = 1, \alpha_{box} = 0.5, \alpha_{landmark} = 0.5$) is chosen to detect face, ($\alpha_{detector} = 1, \alpha_{box} = 0.5, \alpha_{landmark} = 1$) is chosen for facial landmark detection.

3.3 Data Preprocessing and Scaling

Noise, incomplete data (points), and spurious data are major hindrances that impede the use of RGB images for training a model. Because of large size of RGB images and unnecessary background object, the faces of persons are detected and cropped using MTCNN model and stored separately for training and testing the one shot learner. This helps to eliminate spurious data. The support and query sets for training are created by using the first fifteen persons and the rest five persons are used for validating the model performance. The data is then normalized as:

$$Z = \frac{x}{255} \tag{5}$$

where x are the orginal RGB training images. This method normalizes the image to produce $Z\ \varepsilon$ [0, 1]

3.4 Representation Learning

Representation learning is used before training the one learner to get the useful preserved facial and head pose features from cropped face images. Representation Learning disentangles the head pose features from the dataset using CNN based encoder-decoder. ResNet50 [20] model is used to compute the latent features from the head pose images and feed into the one shot learner which then generalizes for new tasks. Identification of latent features led to better generalization in person-specific head pose estimation. Representation learning is used as a robust feature extractor for the one shot learning network as shown in figure 3

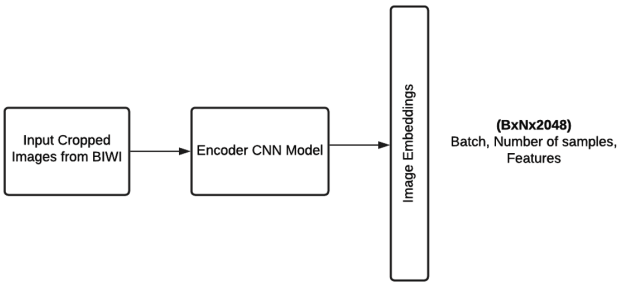

Figure 3: Architecture of Representation Learning

The ResNet model efficiently learns the important features from training images and gives 2048 latent features. Leagacy vanishing gradient issue of CNN is solved in ResNet architecture using skip connections which skips training in few layers and connects directly to the output layer with inherent regularization.

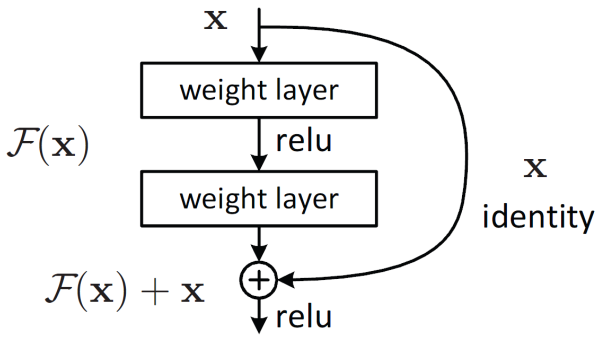

Figure 4: A building block of Residual Network [20]

The output classification layers of ResNet50 model is removed and few linear layers are added and trained to get latent features from the images. The ResNet50 model is used as feature extractor in this research work.

3.5 Model Evaluation

The performance of the the proposed models is dependent is done based on different parameter metric.

3.5.1 Mean Squared Error

Mean squared error (MSE) or mean squared deviation (MSD) of an estimator computes the average of the squares of the errors that represents a probability function that represents the estimated value of squared error loss.

$$MSE = \frac{1}{R}\sum_{i=1}^{R} ||S_i - \hat{S}_i||^2 \tag{6}$$

where S_i is the orginal input and $\hat{S}_i$ is the estimated output.

3.5.2 Mean Absoulte Error

Mean Absolute Error (MAE) computes similarity between two sets. It computes differences between to given set used to compute the difference between ground truth head pose angles and predicted head pose angles.

$$MAE = (\frac{1}{n})\sum_{i=1}^{n} |y_i - x_i| \tag{7}$$

4. Results and Analysis

The experiments are run for 25 episodes with one shot batch size of 32 and learning rate of 0.001. Adam optimizer is used for training the network. The activation function for all layers is taken as ReLU. The experiment is 5 way 1 shot few shot learning experiment. The summary of the other model parameters is shown below:

Table 3: Model Parameters

Parameter	Value
One Batch Size	32
Episodes	25
One shot learning rate	0.001
N-way	5
K-shot	1
Number of Updates	5
Support Set Size	(32, 5, 3, 224, 224)
Query Set Size	(32, 5, 3, 224, 224)

After a number of experimentation over aforementioned number of iterations, Euler angles for head pose in images in predicted. This result also confirmed that one shot can be used for complex regression tasks.

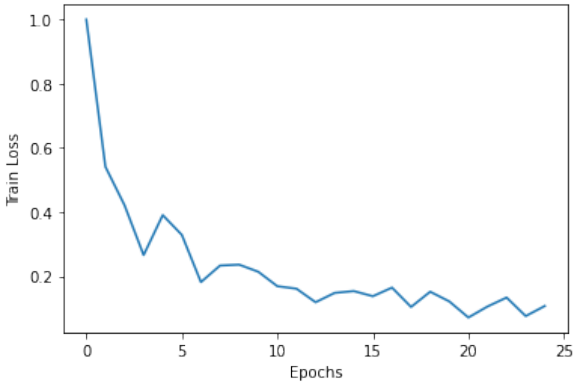

Figure 5: One Shot training loss

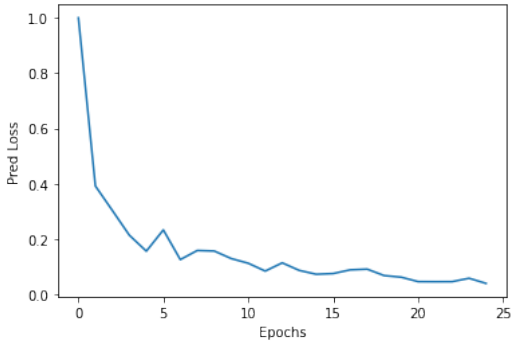

Figure 6: One Shot testing loss

As seen in Figure 5 and 6, over the epochs, it can be noted that the MSE loss is decreasing rapidly at first few epochs and grows steady, but still decreases after that. This trendline shows that the model is learning well. The model works on 5-way 1-shot learning setting to learning during training with 1 image of 5 different person in the support set. Query set consists of 1 image each of 5 different person

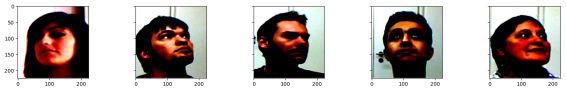

Figure 7: Support set in five way one shot settings

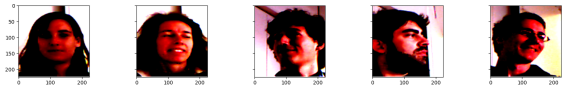

Figure 8: Query set in five way one shot settings

as seen in figure 7 and 8

The predictions of 3D head poses from the proposed architecture can be summarized in the Fig 9, 10 and 11 respectively. The yaw, pitch and roll angles are represented by red, blue and green lines respectively. The direction of red line shows the frontal face direction in the test images. The test image is of an entirely new class/person and is not previously seen in support and query sets. These image have varying visual cues such as glasses and different head orientations. This result shows that proposed model is able to adapt to new task easily only using very few samples for training the network.

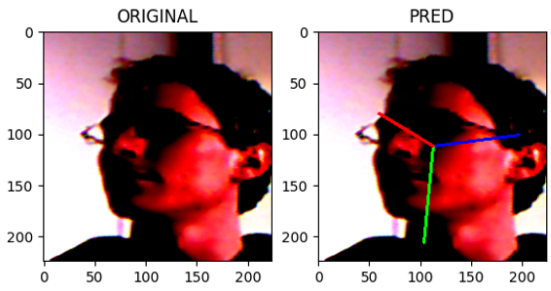

Figure 9: Original image and Predicted head poses

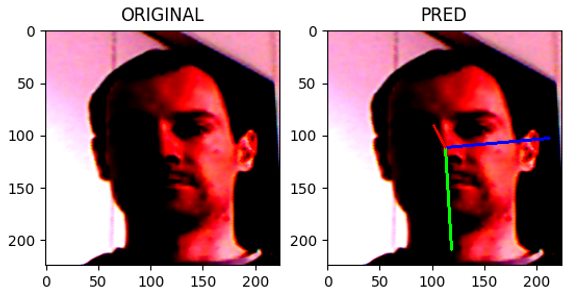

Figure 10: Original image and Predicted head poses

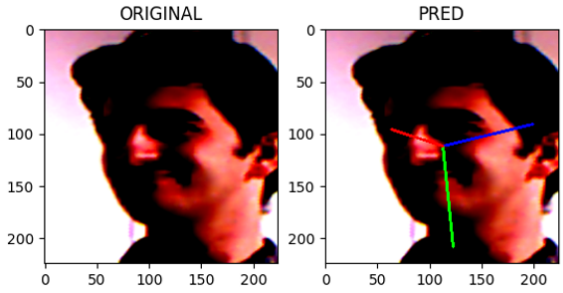

Figure 11: Original image and Predicted head poses

The performance of the model is compared with other standard head pose estimation models as shown below in Table 4. This shows that One shot Learning is efficiently predicting head poses. The best performance is computed in terms of mean absolute error (MAE) between the head pose angles and is shown in bold:

Table 4: Model performance validation in terms MAE

Model	MAE
One Shot based model	6.405
WHENet-V [21]	**3.475**
Dlib (68 Points) [21]	12.249

The implementation of one shot learning based model for head pose estimation gave good results as compared with state of the art deep learning models. However, we achieved good estimations of Euler angles using very few samples rather than a large number of training examples. When using 5 way 1 shot learning approach, the model was successful to estimate Euler's angles for unseen human faces.

5. Conclusion

The implementation of one shot learning based model for head pose estimation gave good results as compared with state of the art deep learning models. However, we achieved good estimations of Euler angles using very few samples rather than a large number of training examples. When using 5 way 1 shot learning approach, the model was successful to estimate Euler's angles for unseen human faces. One Shot algorithm is successfully implemented for head pose regression problem and the results have proved that the model is adaptive for new sets of tasks from the dataset with very few steps of gradient updates during training.

6. Future Works

The work can be further extended to realtime video frames for 3D head pose estimation. One shot learning can be further improved by latest meta learning algorithms.

References

[1] Heikki Huttunen, Ke Chen, Abhishek Thakur, Artus Krohn-Grimberghe, Oguzhan Gencoglu, Xingyang Ni, Mohammed Al-Musawi, Lei Xu, and Hendrik Veen. Computer vision for head pose estimation: Review of a competition. pages 65–75, 06 2015.

[2] Dinh Tuan Tran and Joo-Ho Lee. A robust method for head orientation estimation using histogram of oriented gradients. In Tai-hoon Kim, Hojjat Adeli, Carlos Ramos, and Byeong-Ho Kang, editors, *Signal Processing, Image Processing and Pattern Recognition*, pages 391–400, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[3] Qianru Sun, Yaoyao Liu, Tat-Seng Chua, and Bernt Schiele. Meta-transfer learning for few-shot learning, 2019.

[4] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks, 2017.

[5] J. Ng and Shaogang Gong. Multi-view face detection and pose estimation using a composite support vector machine across the view sphere. In *Proceedings International Workshop on Recognition,*

Analysis, and Tracking of Faces and Gestures in Real-Time Systems. In Conjunction with ICCV'99 (Cat. No.PR00378), pages 14–21, 1999.

[6] E. Osuna, R. Freund, and F. Girosit. Training support vector machines: an application to face detection. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 130–136, 1997.

[7] Yongmin Li, Shaogang Gong, and H. Liddell. Support vector regression and classification based multi-view face detection and recognition. In *Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580)*, pages 300–305, 2000.

[8] Zhu Li, Yun Fu, Junsong Yuan, Thomas S. Huang, and Ying Wu. Query driven localized linear discriminant models for head pose estimation. In *2007 IEEE International Conference on Multimedia and Expo*, pages 1810–1813, 2007.

[9] Margarita Osadchy, Yann Le Cun, and Matthew L. Miller. *Synergistic Face Detection and Pose Estimation with Energy-Based Models*, pages 196–206. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.

[10] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[11] Byungtae Ahn, Jaesik Park, and In So Kweon. Real-time head orientation from a monocular camera using deep neural network. In Daniel Cremers, Ian Reid, Hideo Saito, and Ming-Hsuan Yang, editors, *Computer Vision – ACCV 2014*, pages 82–96, Cham, 2015. Springer International Publishing.

[12] Sankha S. Mukherjee and Neil Martin Robertson. Deep head pose: Gaze-direction estimation in multimodal video. *IEEE Transactions on Multimedia*, 17(11):2094–2107, 2015.

[13] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions, 2014.

[14] Marco Venturelli, Guido Borghi, Roberto Vezzani, and Rita Cucchiara. Deep head pose estimation from depth data for in-car automotive applications, 2017.

[15] Nataniel Ruiz, Eunji Chong, and James M. Rehg. Fine-grained head pose estimation without keypoints, 2018.

[16] Massimiliano Patacchiola and Angelo Cangelosi. Head pose estimation in the wild using convolutional neural networks and adaptive gradient methods. *Pattern Recognition*, 71, 06 2017.

[17] Gabriele Fanelli, Thibaut Weise, Juergen Gall, and Luc Van Gool. Real time head pose estimation from consumer depth cameras. In Rudolf Mester and Michael Felsberg, editors, *Pattern Recognition*, pages 101–110, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[18] Gabriele Fanelli, Juergen Gall, and Luc Van Gool. Real time 3d head pose estimation: Recent achievements and future challenges. pages 1–4, 05 2012.

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

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

[21] Yijun Zhou and James Gregson. Whenet: Real-time fine-grained estimation for wide range head pose, 2020.