Dynamic Multi-Task Learning for Face Recognition with Facial Expression

Zuheng Ming, Junshi Xia, Muhammad Muzzamil Luqman, Jean-Christophe Burie, Kaixing Zhao

▶ **To cite this version:**

Zuheng Ming, Junshi Xia, Muhammad Muzzamil Luqman, Jean-Christophe Burie, Kaixing Zhao. Dynamic Multi-Task Learning for Face Recognition with Facial Expression. Lightweight Face Recognition Challenge Workshop during the 2019 International Conference on Computer Vision (ICCV 2019), Oct 2019, Séoul, South Korea. ⟨hal-02889907⟩

HAL Id: hal-02889907

https://hal.science/hal-02889907v1

Submitted on 5 Jul 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

HAL Authorization

Dynamic Multi-Task Learning for Face Recognition with Facial Expression

Zuheng Ming[1] Junshi Xia[2] Muhammad Muzzamil Luqman[1] Jean-Christophe Burie[1] Kaixing Zhao[3]
[1]L3i, University of La Rochelle, France
[2]RIKEN Center for Advanced Intelligence Project (AIP), RIKEN,Tokyo 103-0027, Japan
[3]IRIT, University of Toulouse
{zuheng.ming, mluqma01, jcburie}@univ.lr-fr, jushi.xia@riken.jp, kaixing.zhao@irit.fr

Abstract

Benefiting from the joint learning of the multiple tasks in the deep multi-task networks, many applications have shown the promising performance comparing to single-task learning. However, the performance of multi-task learning framework is highly dependant on the relative weights of the tasks. How to assign the weight of each task is a critical issue in the multi-task learning. Instead of tuning the weights manually which is exhausted and time-consuming, in this paper we propose an approach which can dynamically adapt the weights of the tasks according to the difficulty for training the task. Specifically, the proposed method does not introduce the hyperparameters and the simple structure allows the other multi-task deep learning networks can easily realize or reproduce this method. We demonstrate our approach for face recognition with facial expression and facial expression recognition from a single input image based on a deep multi-task learning Conventional Neural Networks (CNNs). Both the theoretical analysis and the experimental results demonstrate the effectiveness of the proposed dynamic multi-task learning method. This multi-task learning with dynamic weights also boosts of the performance on the different tasks comparing to the state-of-art methods with single-task learning. [1]

1. Introduction

Multi-task learning has been used successfully across many areas of machine learning [27], from natural language processing and speech recognition [6, 7] to computer vision [10]. By joint learning in multiple tasks in the related domains with different information, especially from information-rich tasks to information-poor ones, the multi-task learning can capture a representation of features being difficult learned by one task but can easily learned by another task [24]. Thus the multi-task learning can be conducted not only for improving the performance of the systems which aims to predict multiple objectives but also can utilise for improving a specific task by leveraging the related domain-specific information contained in the auxiliary tasks. In this work, we explore the multi-learning for face recognition with facial expression. Thanks to the progress of the representing learning with the deep CNNs, face recognition has made remarkable progress in the recent decade [32, 25, 28, 20]. These works have achieved or beyond the human-level performance on the benchmarks LFW[14], YTF[35]. The challenges of face recognition such as the variation of the pose, the illumination and the occlusion have been well investigated in many researches, nevertheless face recognition for the face with the non-rigid deformation such as the ones introduced by the facial expression has not been sufficiently studied especially in the 2D face recognition domain. Some 3D based methods have been proposed to deal with this issue such as [42, 16, 3], in which [42] presents the method by using the 3D facial model to normalise the facial expression and then maps the normalised face to the 2D image to employ face recognition. In order to leverage the promising progress in the numerous 2D face recognition and facial expression recognition researches particularly based on the deep neural networks, we propose to combine the face recognition task and the facial expression recognition task in the unified multi-task framework aiming to jointly learn the sharing features and task-specific features to boost the performance of each task. Figure 1 shows the multi-task framework proposed in this work.

How to set the weights of tasks is a crucial issue in the multi-task learning. The weights determine the importance of the different tasks in the holistic networks. Many works simply set the equal values for all tasks or experimentally set the weights of the tasks. In [4], the authors assign equal weights to the ranking task and the binary classification task for the person re-identification. However the multi-task learning is an optimization problem for multiple objectives.

[1] https://github.com/hengxyz/Dynamic_multi-task-learning.git

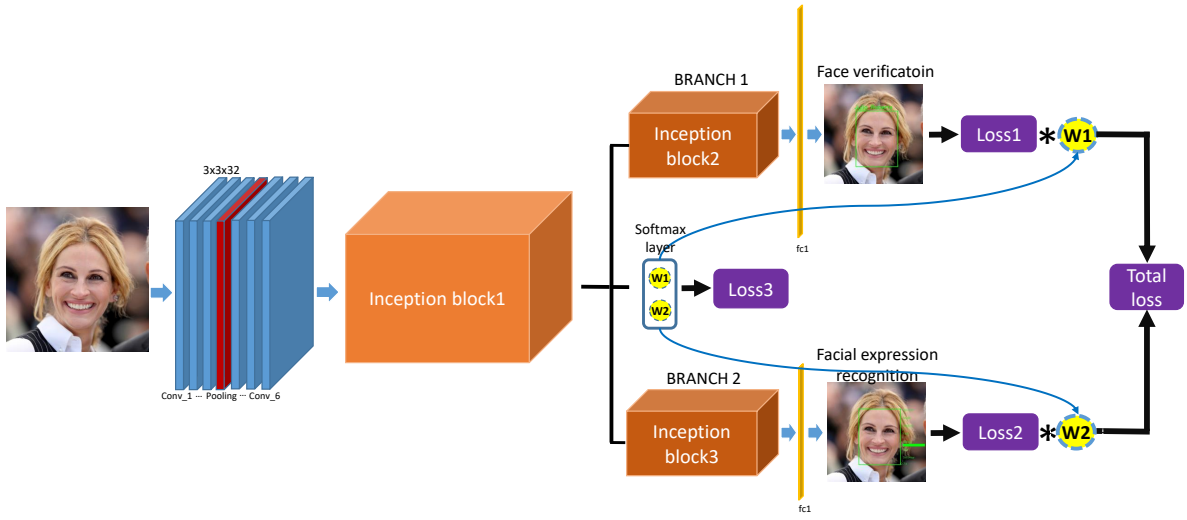
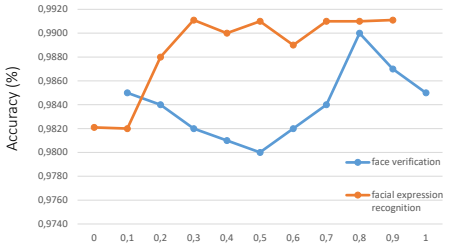


Figure 1. The proposed multi-task framework with dynamic weights of tasks to simultaneously perform face recognition with facial expression and facial expression recognition. The dynamic weights of tasks can adapt automatically according to the difficulty of the training of tasks.

The main task and the side tasks with different objective have different importance in the overall loss meanwhile the difficulty of the training of each task is also different. Thus it is arbitrary to assign equal weights for tasks for multi-task learning. We also verified this point in our work by manually setting the weights of tasks from 0 to 1 with the interval of 0.1. As shown in Figure 2, either for the facial expression recognition task or for the face recognition task, the best performance are obtained with the different weights of tasks rather than the equal weights of each task. Most of the multi-task learning methods search the optimal weights of the tasks by the experimental methods, for instance Hyperface [26] manually set the weights of the tasks such as the face detection, landmarks localization, pose estimation and gender recognition according to their importance in the overall loss, and [33] obtain the optimal weights by a greedy search for pedestrian detection tasks with the different attributes. Besides the cost of time and being exhausting, these methods setting the weights as the fix values to optimize the tasks ignore the variation of the the importance or the difficulty of the tasks during the training processing. Rather than the methods with fix weights which can be so called static weights methods, the methods [5, 17, 39] update the weights or part of the weights of the tasks during the training of the networks. [39] set the weight of the main task as 1 while the auxiliary tasks are weighted by the dynamic weights λ_t which are updated by an analytical solution. [17] introduces a uncertainty coefficient θ to revise the softmax loss function of each task. Unlike these methods which need to introduce the additional hyperparameters to update the weights of tasks, we propose to use a softmax layer adding to the end of the hidden sharing layers of the multi-task networks to generate the dynamic weights of the tasks (see Figure 1). Each unit of this softmax layer is corresponding to a weight of a task and no more hyperparameter is introduced for updating the tasks weights. Rather than [37] updating simultaneously the dynamic weights of tasks and the filters weights of the networks by the unify total loss of the networks, we propose a new loss function to update the dynamic weights which enable the networks to focus on the training of the hard task by automatically assigning a larger weight. On the contrary, [37] always updates the smaller weight for the hard task and the larger weight for the easy task which results the hard task is far from being fully trained and the networks stuck in the worthless training of the over trained easy task. This is due to use the total loss of the networks to simultaneously update the weights of tasks, in which the dynamic weights of the tasks are also in the function of the weights of networks, i.e. $\mathcal{L}_{total}(\Theta) = w_1(\Theta_0)\mathcal{L}_1(\Theta_1) + w_2(\Theta_0)\mathcal{L}_2(\Theta_2) \quad s.t. \quad w_1 + w_2 = 1$ and $\{\Theta_0, \Theta_1, \Theta_2\} = \Theta$ are the weights of the networks. The optimization of Θ by the total loss $\mathcal{L}_{total}$ aims to decrease the total loss as much as possible, thus when the hard task has a large loss the fastest way to decrease the total loss is to shrinkage its weight w_i so that the weighted loss of the hard task can be cut down rapidly. This is why the hard task always has a small task weight while the easy task has a large weight.

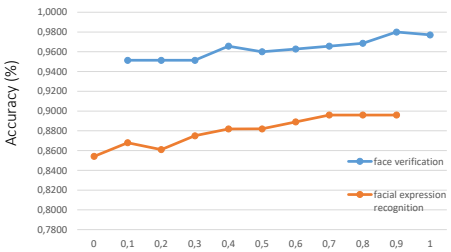
In a summary, our main contributions of this paper are.

- We propose a dynamic multi-task learning method which can automatically update the weight of task according to the importance of task during the training.



w1	FV(%)	EX(%)
0	–	98.21
0.1	98.50	98.20
0.2	98.40	98.80
0.3	98.20	99.11
0.4	98.10	99.00
0.5	98.00	99.10
0.6	98.20	98.90
0.7	98.40	99.10
0.8	99.00	99.10
0.9	98.70	99.11
1.0	98.50	-

(a) CK+



w1	FV(%)	EX(%)
0	–	85.42
0.1	95.14	86.80
0.2	95.14	86.10
0.3	95.14	87.50
0.4	96.57	88.19
0.5	96.00	88.20
0.6	96.28	88.90
0.7	96.57	89.60
0.8	96.86	89.60
0.9	98.00	89.60
1.0	97.71	-

(b) OuluCASIA

Figure 2. Performances of the task of face verification for facial expression images (FV / blue) with the manually setting weight $w1$ and the task of facial expression recognition (EX / red) on the different datasets CK+ and OuluCASIA. The weight of the facial expression recognition is $w2$ = 1-$w1$.

- Both the theoretical analysis and the experimental results demonstrate the proposed dynamic multi-task learning enable to focus on the training of the hard task to achieve better efficiency and performance.
- We have demonstrated that, for both face verification with facial expression and facial expression recognition tasks, the proposed multi-task learning can outperform the state-of-the-art performance on the datasets CK+ [22], OuluCASIA [40].
- The proposed method is simple and does not introduce the hyperparameters,which can be easily realized and reproduce in the other deep multi-task learning frameworks.

The remainder of this paper is organized as follows: Section II briefly reviews the related works; Section III describes the architecture of the dynamic multi-task network. Section IV presents the approach of multi-task learning with dynamic weights following by Section V where the experimental results are analyzed. Finally, in Section VI, we draw the conclusions and present the future works.

2. Related works

Multi-task learning not only helps to learn more than one task in a single network but also can improve upon your main task with an auxiliary task [27]. In this work, we focus on the multi-task learning in the context of the deep CNNs. According to the means of updating the weights of tasks, the multi-task learning can be divided into two categories: the static method and dynamic method. In the static methods, the weights of tasks are set manually before the training of the networks and they are fixed during the whole training of the networks [26, 10, 4, 36, 33]; while the dynamic methods initialize the weights of the tasks at the beginning of the training and update the weights during the training processing [5, 17, 39, 37]. There are two ways for setting the weights in the static methods. The first way is to simply set the equal weights of task such as Fast R-CNN [10] and [4, 36]. In Fast R-CNN, the author uses a multi-task loss to jointly train the classification and bounding-box regression for object detection. The classification task is set as the main task and the bounding-box regression is set as the side task weighted by λ. In the experiments the λ is set to 1. The second way to set the weights is manually searching the optimal weights by the experimental methods. Hyperface [26] proposed a multi-task learning algorithm for face detection, landmarks localization, pose estimation and gender recognition using deep CNNs. The tasks have been set the different weights according to the importance of the task. [4] integrated the classification task and the ranking task in a multi-task networks for person re-identification problem. Each task has been set with a equal weight to jointly optimizing the two tasks simultaneously. Tian et al. [33] fix the weight for the main task to 1, and obtain the weights of all side tasks via a greedy search within 0 and 1. In [5] an additional loss in function of the gradient of the weighted losses of tasks is proposed to update the weights meanwhile an hyperparameter is introduced for balancing the training of different tasks. [17] introduces a uncertainty coefficient θ to combine the multiple loss functions. The θ can be fixed manually or learned based on the total loss. Zhang et al. [39] propose a multi-task networks for face landmarks detection and the recognition of the facial attributes. The face landmarks detection is set as the main task with the task weight 1 and the tasks for recognition of the different facial attributes are set as auxiliary tasks with dynamic weights λ_t. An hyperparameter ρ as a scale factor is introduced to calculate the weight λ_t. Yin et al. [37] proposed a multi-task model for face pose-invariant recognition with an automatic learning of the weights for each task. The main task of is set to 1 and the auxiliary tasks are sharing the dynamic tasks generated by the softmax layer. However, the update of the weights of tasks by the total loss of the networks runs counter to the objective of the multi-task learning. Thanks to the progress of the representation learning based on the deep neural networks, the methods based on the deep CNNs such as Deep-

Face [32] , DeepIDs [31], Facenet [28], VGGFace [30], SphereFace [20] have made a remarkable improvement comparing to the conventional methods based on the handcrafted features LBP, Gabor-LBP, HOG, SIFT [1, 8, 2, 29]. The situation is same as facial expression recognition based on deep CNNs [15, 41, 23]. Even so, the studies on the face recognition with the facial expression images are limited. [16, 42, 3] propose the 3D based methods to deal with this issue. Kakadiaris et al. [16] present a fully automated framework for 3D face recognition using the Annotated Face Model to converted the raw image of face to a geometric model and a normal map. Then the face recognition is based on the processed image by using the Pyramid and Haar. Zhu et al. [42] presents the method by using the 3D facial model to normalise the facial expression and then maps the normalised face to the 2D image to employ face recognition. Chang et al. [3] describe a method using three different overlapping regions around the nose to employ the face recognition since this region is invariant in the presence of facial expression.

3. Architecture

The proposed multi-task learning with dynamic weights is based on the deep CNNs (see Figure 1). The hard parameter sharing structure is adopted as our framework, in which the sharing hidden layers are shared between all tasks [27]. The task-specific layers consisting of two branches are respectively dedicated to face verification and facial expression recognition. The two branches have almost identical structures facilitate the transfer learning of facial expression recognition from the pretrained face recognition task. Specifically, the BRANCH 1 can extract the embedded features of bottleneck layer for face verification and the BRANCH 2 uses the fully connected softmax layer to calculate the probabilities of the facial expressions. The deep CNNs in this work are based on the Inception-ResNet structure which have 13 million parameters of about 20 hidden layers in terms of the depth and 3 branches to the maximum in terms of the large. By the virtue of the Inception structure, the size of the parameters is much fewer than other popular deep CNNs such as VGGFace with 138 million parameters.

Dynamic-weight-unit The dynamic weights of tasks are generated by the softmax layer connecting to the end of the sharing hidden layers, which can be so called the Dynamic-weight-unit. Each element in the Dynamic-weight-unit is corresponding to a weight of a task, thus the size of the Dynamic-weight-unit is equal to the number of weights of tasks, e.g. the size is 2 in this work. Since the weights are generated by the softmax layer, $w1 + w2 = 1$ which can well indicate the relative importance of the tasks. The parameters of this softmax layer are updated by the independent loss function $\mathcal{L}_3$ during the training of the networks, which can automatically adjust the weights of tasks in light of the variation of the loss of tasks and drive the networks to always train the hard task firstly by assigning a larger weight.

4. Multi-task learning with dynamic weights

The total loss of the proposed multi-task CNNs is the sum of the weighted losses of the multiple tasks.

(I) **Multi-task loss** $\mathcal{L}$: The multi-task total loss $\mathcal{L}$ is defined as follows:

$$\mathcal{L}(\mathbf{X}; \Theta; \Psi) = \sum_{i=1}^{T} w_i(\Psi)\mathcal{L}_i(\mathbf{X}_i; \Theta_i) \tag{1}$$

where T is the number of the tasks, here $T = 2$. X_i and Θ_i are the feature and the parameters corresponding to each task, $\Theta = \{\Theta_i\}_{i=1}^{T}$ are the overall parameters of the networks to be optimized by $\mathcal{L}$. Ψ are the parameters of the softmax layer in the Dynamic-weight-unit used to generate the dynamic weights $w_i \in [0, 1]$ s.t. $\sum w_i = 1$. Thus $\{\mathbf{X}_i, \Theta_i\} \in \mathbb{R}^{d_i}$, where d_i is the dimension of the features X_i, and $\{\mathcal{L}_i, w_i\} \in \mathbb{R}^1$. Particularly, when w_1 = 1, w_2 = 0 the multi-task networks are degraded as the single-task networks for face verification (i.e. Branch 1 and sharing hidden layers) while w_1 = 0, w_2 = 1 is corresponding to the single-task networks for facial expression recognition (i.e. Branch 2 and sharing hidden layers).

(II) **Face verification task loss** $\mathcal{L}_1$: The loss for face verification task is measured by the center loss [34] joint with the cross-entropy loss of softmax of Branch 1. The loss function of face verification task $\mathcal{L}_1$ is given by:

$$\mathcal{L}_1(\mathbf{X}_1; \Theta_1) = \mathcal{L}_{s1}(\mathbf{X}_1; \Theta_1) + \alpha\mathcal{L}_c(\mathbf{X}_1; \Theta_1) \tag{2}$$

where $\mathcal{L}_{s1}$ is the cross-entropy loss of softmax of Branch 1, $\mathcal{L}_c$ is the center loss weighted by the hyperparameter α. The $\mathcal{L}_c$ can be treated as a regularization item of softmax loss $\mathcal{L}_{s1}$ which is given by:

$$\begin{aligned} \mathcal{L}_{s1}(\mathbf{X}_1; \Theta_1) &= \sum_{k=1}^{K} -y_k log P(y_k = 1|\mathbf{X}_1, \theta_k) \\ &= -\sum_{k=1}^{K} y_k log \frac{e^{f^{\theta_k}(\mathbf{X}_1)}}{\sum_{k'}^{K} e^{f^{\theta_{k'}}(\mathbf{X}_1)}} \end{aligned} \tag{3}$$

where K is the number of the classes, i.e. the number of identities in the training dataset, $y_k \in \{0, 1\}$ is the one-shot label of the feature $\mathbf{X}_1$, $P(y_k|\mathbf{X}_1, \theta_k)$ is softmax function over the activation function $f^{\theta_k}(\mathbf{X}_1)$ where $\{\theta_k\}_{k=1}^{K} = \Theta_1$, $\theta_k \in \mathbb{R}^{d_1}$. The bottleneck layer of BRANCH 1 is extracted as the feature $\mathbf{X}_1$ of the input image. The center loss $\mathcal{L}_c$ is given by:

$$\mathcal{L}_c(\mathbf{X}_1; \Theta_1) = ||\mathbf{X}_1 - C_{y_k}|| \tag{4}$$

Where the C_{y_k} is the center of the class which $\mathbf{X}_1$ belonging to, $C_{y_k} \in \mathbb{R}^{d_1}$.

(III) **Facial expression recognition task loss** $\mathcal{L}_2(\mathbf{X}_2; \Theta_2)$: The loss function of facial expression recognition task $\mathcal{L}_2$ is the cross-entropy loss of the softmax layer of BRANCH 2. The equation of $\mathcal{L}_2$ is as same as Equation 3 except the K in $\mathcal{L}_2$ is the number of the categories of the facial expressions, $\mathbf{X}_2$ is the bottleneck layer of BRANCH 2, $\Theta 2$ is corresponding parameters of this task.

(IV) **Generation of the dynamic weights** $w_i(\Psi)$: The dynamic weights w_i are generated by the softmax layer of the dynamic-weight-unit which is given by:

$$w_i(\mathbf{Z}; \Psi) = \frac{e^{f^{\psi_i}(\mathbf{Z})}}{\sum_{i'}^{T} e^{f^{\psi_{i'}}(\mathbf{Z})}} \tag{5}$$

where the $\mathbf{Z} \in \mathbb{R}^{d_z}$ is the flat output of the last layer of the sharing hidden layers.T is the number of the tasks, here T=2. ψ_i is parameters in the softmax layer of the dynamic-weight-unit $\{\psi_i\}_{i=1}^{T} = \Psi$, $\psi_i \in \mathbb{R}^{d_z}$. $f^{\psi_i}(\mathbf{Z})$ is activation function which is given by:

$$f^{\psi_i}(\mathbf{Z}) = \psi_i \mathbf{Z}^T + b_i \tag{6}$$

Note that, we do not use the Relu function as the activation function since Relu discards the values minors zero. This shrinks the range of the variation of the dynamic weights w_i.

(V) **Update of the dynamic weights** w_i: Rather than using the total loss to update the dynamic weights, we propose a new loss function to update the dynamic weights which can drive the networks always train the hard task. The proposed new loss function for updating the dynamic weights is given by:

$$\mathcal{L}_3(\mathbf{Z}; \Psi) = \sum_{i=1}^{T} \frac{w_i(\psi_i)}{\mathcal{L}_i(\Theta_i)} \quad s.t. \quad \sum w_i = 1 \tag{7}$$

Note that, $\mathcal{L}_i\{\Theta_i\}$ is independent with $w_i(\psi_i)$ since $\Theta_i \cap \psi_i = \emptyset$, $i \in [1, .., T]$, thus $\mathcal{L}_i$ is constant for the dynamic weight update loss function $\mathcal{L}_3$.

(VI) **Qualitative analysis** shows that when the loss of the task $\mathcal{L}_i$ is small, i.e. the reciprocal of the $\mathcal{L}_i$ is large, thus loss $\mathcal{L}_3$ will try to reduce the loss by decreasing the value of w_i. That is to say, when the task is easy with a small loss the weight of the task will be assigned by a small value. On the contrary, the hard task with a large loss will be assigned by a large weight, which enable the networks always focus on training the hard task firstly. The update of the dynamic weights w_i is essentially the update of the parameters ψ_i which generate the dynamic weights.

(VII) **Quantitative analysis**: Considering the Equation 5 and Equation 6, the gradient of the ψ_i can be given by

$$\nabla\psi_i = \frac{\partial \mathcal{L}_3}{\partial \psi_i} = \frac{1}{\mathcal{L}_i} \frac{\partial w_i(\psi_i)}{\partial \psi_i} = \frac{1}{\mathcal{L}_i} \frac{a_i \sum_{j \neq i}^{T} a_j}{(\sum_i^T a_i)^2} \mathbf{Z} \tag{8}$$

where $a_i = e^{\psi_i \mathbf{Z}^T + b_i}$, and the update of the parameters is $\psi_i^{t+1} = \psi_i^t - \eta \nabla {\psi_i}^t$ where η is the learning rate. Then the new value of the dynamic weight w_i^{t+1} can be obtained by the Equation 5 and 6 with the ψ_i^{t+1}.

If we assume the $b_i^0 = 0, \psi_i^0 = 0$ (this is possible if we initialize the ψ_i, b_i by zero), the ψ_i^t can be given by

$$\psi_i^t = -\sum \frac{1}{\mathcal{L}_i} \frac{a_i \sum_{j \neq i}^{T} a_j}{(\sum_i^T a_i)^2} \mathbf{Z} \tag{9}$$

if we consider the case for two tasks w_1 and w_2 when $t = 1$:

$$\begin{aligned} \frac{w_1^t}{w_2^t} &= e^{(\psi_1^t - \psi_2^t)\mathbf{Z}^T} \\ &= e^{(\frac{1}{\mathcal{L}_2} - \frac{1}{\mathcal{L}_1}) \frac{a_1 a_2}{(a_1 + a_2)^2} \mathbf{Z}\mathbf{Z}^T} \end{aligned} \tag{10}$$

We can see that $a_i > 0$ and $ZZ^T \geq 0$, so if $\mathcal{L}_2 < \mathcal{L}_1$ the $\frac{w_1}{w_2} > 1$ namely $w_1 > w_2$. It means if the loss of task1 larger than the loss of task 2, the weight of the task1 is larger than the one of task2. It indicates that the proposed loss function $\mathcal{L}_3$ can well update the weights of tasks to drive the networks always train the hard task firstly.

(VIII) **Training protocol**: The training of the entire deep CNNs includes two independent training: the training of the parameters of the networks Θ by the multi-task loss $\mathcal{L}(\Theta) = \sum_{i=1}^{2} \mathcal{L}_i(\theta_i)$ and the training of the parameters of weight-generate-module Ψ by the loss $\mathcal{L}_3(\Psi)$. These can be conducted simultaneously in a parallel way.

$$\Theta^{t-1} - \eta \frac{\partial \mathcal{L}(\Theta)}{\partial \Theta} \mapsto \Theta^t \tag{11}$$

$$\Psi^{t-1} - \eta \frac{\partial \mathcal{L}_3(\Psi)}{\partial \Psi} \mapsto \Psi^t \tag{12}$$

where $\eta \in (0, 1)$ is the learning rate.

5. Experiments and analysis

5.1. Datasets

Since the proposed multi-task networks performs the face verification task and the facial expression recognition task simultaneously, the datasets including both identity labels and facial expression labels are necessary to the training and the evaluation of the model. However, the large-scale datasets such as Celeb-A [21] and FER2013 [12] either do not include the facial expression or the identity labels. Finally 5184 (positive or negative) pairs of face images with both identity labels and facial expression labels

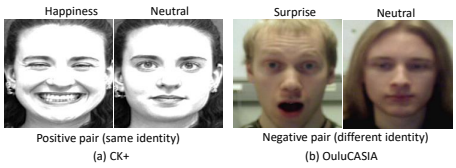


Figure 3. The image pairs extracted from CK+ and OuluCASIA.

Table 1. The datasets used in the multi-task learning for face verification and facial expression recognition in this work. The labels of images is: ID (identities), Neutral (Ne), Anger (An), Disgust (Di), Fear (Fe), Happy (Ha), Sad (Sa), Surprise (Su), Contempt (Co).

	ID	Ne	An	Di	Fe	Ha	Sa	Su	Co
CK+	123	327	135	177	75	147	84	249	54
OuluCASIA	560	560	240	240	240	240	240	240	-

are extracted from OuluCASIA as well as 2099 pairs of images are extracted from CK+ to form two datasets respectively (see Fig. 3 and Table 1).

5.2. Experimental configuration

In both training and evaluation phase, the faces have been detected by the MTCNN [38] from the given raw images. The RMSprop with the mini-batches of 90 samples are applied for optimizing the parameters. The learning rate is started from 0.1, and decay by 10 at the different iterations depends on the different tasks. The networks are initialized by Xavier [11] and biases values are set to zero at beginning. The momentum coefficient is set to 0.99. The dropout with the probability of 0.5 and the weight decay of 5e-5 are applied. The weight of the center loss α is set to 1e-4.

5.3. Pretrained model

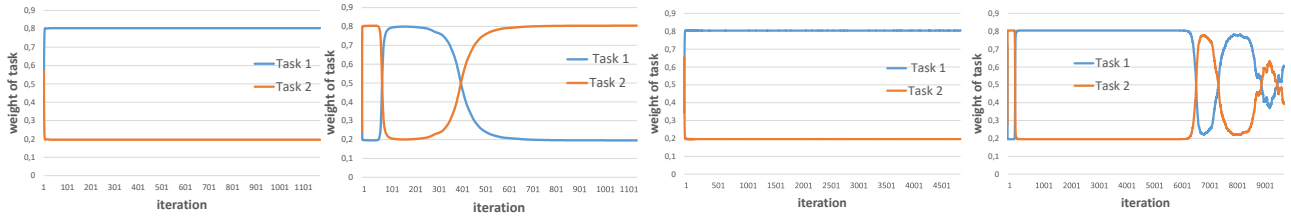
Before the training of the proposed multi-task CNNs, a single-task network constituted of the sharing hidden layers and the BRANCH 1 is pretrained for face verification-task with large-scale dataset by loss function $\mathcal{L}_1$. Then the training of the dynamic multi-task CNNs can handling on the pretrained model. Moreover, in order to compare the multi-task learning with the single-task learning, the BRANCH 2 is also trained independently by transferring the learning of the pretrained BRANCH 1 for facial expression recognition with loss function $\mathcal{L}_2$. Finally we obtain two models pretrained by the single-task learning for face verification (sharing layers + BRANCH 1) and facial expression recognition (sharing layers + BRANCH 2) respectively.

5.4. Dynamic multi-task learning / training

In order to distinguish our proposed method, we call the method in [37] as naive dynamic method. Comparing to the naive dynamic method, the proposed dynamic method can adjust the weights of tasks according to their importance/training difficulty as shown in Figure 4. The training difficulty of the task is presented by its training loss. Figure 5 shows the variation of the loss of tasks corresponding to the two different methods. From Figure 4 and Figure 5, we can see that the naive dynamic method always train the easy task namely facial expression recognition (denoted as Task 1) with smaller loss by assigning a large weight as shown in (a) on dataset CK+ or (c) on dataset OuluCASIA . However, the hard task namely face verification (denoted as Task 2) with large loss is always assigned by small weight less than 0.2. Contrarily, the weight of task generated by the proposed method can effectively adapt to the varied importance of the task in the multi-task learning. For instance, as shown in the (b) on dataset CK+, the hard task which is face verification (Task 2) with a large loss is assigned a large weight at the beginning of the training. The large weight of task drive the networks to fully train the hard task so that the loss of the hard task decreases rapidly and soon it is lower than the loss of the task of facial expression recognition (Task 1). Once the previous easy task become the hard task with a larger loss, the proposed method automatically assigns a larger weight to the current hard task as shown in (b) that the weight of the facial expression recognition (Task 1) augment promptly from the bottom to the top when the loss of the task becomes the larger one. Thus the networks are capable to switch to fully train the current hard task with the proposed dynamic method. Figure 6 shows how the multi-task learning decreases the losses of the tasks with the proposed dynamic weights and the naive dynamic weights on dataset CK+ and OuluCASIA respectively. It suggests that the proposed dynamic method can decrease the loss of the hard task, i.e. the face verification task, more quickly and achieve lower value of loss. For the easy task, namely the facial expression recognition task, these two methods decrease the loss similarly since the easy task can be sufficiently trained by both of the methods. Thus the proposed dynamic method is superior to the naive dynamic method in terms of the training efficiency.

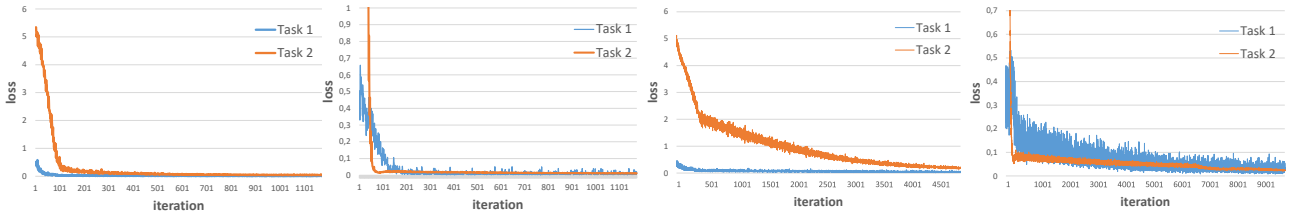
5.5. Gradient vanishing problem

In this section we analyse the problem of the gradient vanishing for updating the dynamic weights. From Equation 8, we can see that if the $a_i \sum_{j \neq i}^{T} a_j << (\sum_{i}^{T} a_i)^2$, the $\nabla \psi_i \rightarrow 0$ which means that the gradient vanishes. In this work, $T = 2$, if $0 << a_1^2 + a_2^2 + a_1 a_2$, it will cause the problem of the gradient vanishing. Since the $a_i = e^{\psi_i \mathbf{Z}^T + b_i} > 0$, the condition of gradient vanishing is easy to satisfy provided the a_i is relative large. In order to mitigate the problem of grand vanishing, we normalize the a_i as the embedding feature for calculating the weights. As shown in (a) and (b) of Figure 7, the gradient vanishes



(a) Naive dynamic - CK+ (b) Proposed dynamic - CK+ (c) Naive dynamic - Oulu (d) Proposed dynamic - Oulu

Figure 4. The weights of tasks generated by our proposed method and the naive dynamic method during the training. Task 1 is facial expression recognition and Task 2 is face verification. The training is conducted on dataset CK+ and OuluCASIA respectively.



(a) Naive dynamic - CK+ (b) Proposed dynamic - CK+ (c) Naive dynamic - Oulu (d) Proposed dynamic - Oulu

Figure 5. The loss of tasks corresponding to our proposed method and the naive dynamic method during the training. Task 1 is facial expression recognition and Task 2 is face verification. The training is conducted on dataset CK+ and OuluCASIA respectively.

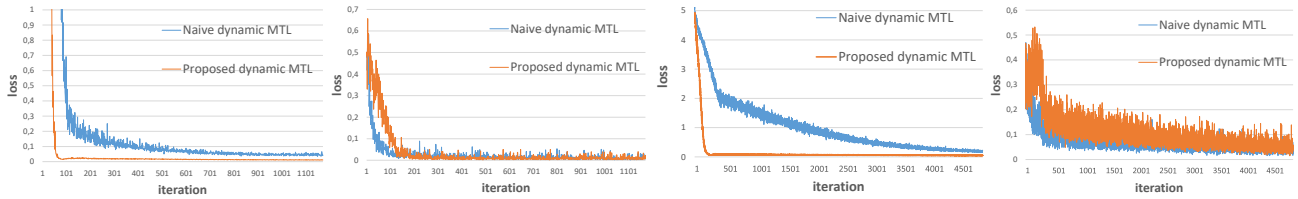
when the a_i is large than $8*10^6$. By applying the normalization of a_i as shown in (d). the gradient return to the normal values as shown in (c).

5.6. Evaluation and ablation analysis

(I) Dynamic multi-task learning for face verification with facial expression To evaluate the effectiveness of the proposed dynamic multi-task learning method for face verification with facial expression, we firstly analyse the results from the single-task method with pretrained models trained on the general datasets and then the fine-tuning model based on the datasets used in this work. Furthermore, we compare the multi-task learning methods with manually setting weights (i.e. static multi-task learning), naive dynamic weights and our proposed dynamic weights to the single-task learning method.

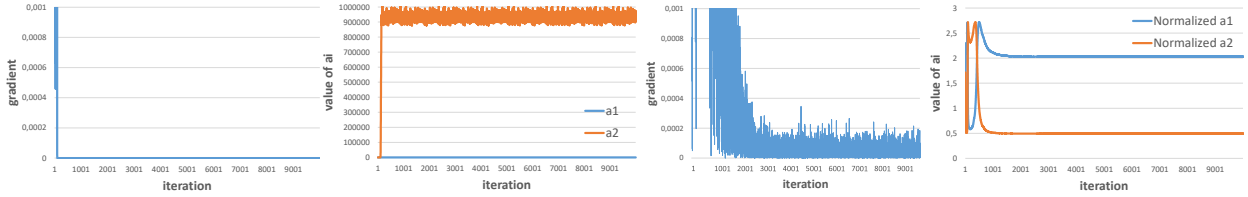
Table 2 firstly shows that the performance of the state-of-art methods such as DeepID, VGGFace, FaceNet, etc. with pretrained models for face verification with facial expression. Comparing to the performance on the general dataset such as LFW or YTF, we can see that the performances on the face images with facial expression in CK+ or OuluCASIA have degraded obviously, e.g. the face verification accuracy of DeepID has decreased from 99.47% on LFW to 91.70% on CK+, VGGFace has decreased from 98.95% on LFW to 92.20% on CK+. Even FaceNet trained on the very large dataset has decreased slightly from 99.63% on LFW to 97.50% on OuluCASIA as well as our single-task model pretrained on the large-scale dataset MSCeleb [13] whose verification accuracy decrease from 99.41% on LFW to 92.60% on OuluCASIA. This is quite probably resulted by the lack of the facial expression images in the general datasets for the training of the models. By fine-tuning our pretrained model with the facial expression datasets, the performance has improved (evaluating by the 10 folds cross-validation) from 92.6% to 97.71% on OuluCASIA. Thanks to the capacity of learning the features between the tasks, the static multi-task learning further improve the performance comparing to the fine-tuning single task model from 97.71% to 98.0% on OuluCASIA. However, the performance of the naive dynamic multi-task learning is inferior to the static multi-task learning and even the fine-tuning single-task model. This is due to the face verification task is the hard task comparing to facial expression recognition. Thus the face verification task assigned by a small weight has not been sufficiently trained. Finally, the proposed dynamic multi-task learning method with the appropriately assigned task weights achieves the best results both on the datasets CK+ and OuluCASIA. Rather than the fix weights of static multi-learning method, the dynamic method can real-time update the weights of tasks with a adequate fine variation. That is why the proposed dynamic multi-task learning is superior to the static multi-task learning method.

(II) Dynamic multi-task learning for facial expression recognition Table 3 and Table 4 compare the proposed dynamic multi-task learning for facial expression recognition with other methods on CK+ and OuluCASIA respectively. As well as the face verification task, the proposed dynamic multi-task learning achieves the best performance on both datasets. Since the facial expression recognition is the easy task, the naive dynamic multi-task learning has sufficiently trained this task and achieved the comparable results as the proposed method. The multi-task learning also show the significant improvement to the single-task methods.



(a) Face verification - CK+ (b) Facial expression - CK+ (c) Face verification - Oulu (d) Facial expression - Oulu

Figure 6. The training efficiency for decreasing the loss of task by the proposed dynamic multi-learning method and the naive dynamic multi-learning method on different tasks, i.e. face verification and facial expression recognition. The experiments are conducted on CK+ and OuluCASIA respectively.



(a) Original gradient (b) Original a_i (c) Normalized gradient (d) Normalized a_i

Figure 7. The gradient value before and after the normalization of a_i. The normalization of the a_i can mitigate the gradient vanishing problem caused by the large value of a_i.

Table 2. The evaluation of face verification on facial expressions datasets with different methods (accuracy%).

Method	Images	LFW	YTF	CK+	Oulu.
DeepFace [32]	4M	97.35	91.4	-	-
DeepID-2,3 [31]	-	99.47	93.2	91.70	96.50
FaceNet [28]	200M	**99.63**	**95.1**	98.00	97.50
VGGFace [30]	2.6M	98.95	91.6	92.20	93.50
Centerloss [34]	0.7M	99.28	94.9	94.00	95.10
SphereFace [20]	0.7M	99.42	95.0	93.80	95.50
Single-task (pretrained)	1.1M	99.41	95.0	98.00	92.60
Single-task (fine-tuning)	1.1M	99.10	94.2	98.50	97.71
Static MTL	1.1M	99.23	94.1	98.50	98.00
Naive dynamic MTL	1.1M	99.23	94.1	98.15	95.14
Proposed dynamic MTL	1.1M	99.21	94.3	**99.00**	**99.14**

Table 3. The evaluation of proposed multi-task networks for facial expression recognition task on dataset CK+.

Method	Accuracy(%)
LBPSVM [9]	95.1
Inception [23]	93.2
DTAGN [15]	97.3
PPDN [41]	97.3
AUDN [19]	92.1
Single-task	98.21
Static MTL	99.11
Naive dynamic MTL	99.10
Proposed dynamic MTL	**99.50**

Table 4. The evaluation of proposed multi-task networks for facial expression recognition task on dataset OuluCASIA.

Method	Accuracy(%)
HOG3D [18]	70.63
AdaLBP [40]	73.54
DTAGN [15]	81.46
PPDN [41]	84.59
Single-task	85.42
Static MTL	**89.60**
Naive dynamic MTL	88.89
Proposed dynamic MTL	**89.60**

6. Conclusion

In this work, we propose a dynamic multi-task learning method which allows to dynamically update the weight of task according to the importance of the task during the training process. Comparing to the other multi-task learning methods, our method does not introduce the hyperparameters and it enables the networks to focus on the training of the hard tasks which results a higher efficiency and better performance for training the multi-task learning networks. Either the theoretical analysis or the experimental results demonstrate the effectiveness of our method. This method can be also easily applied in the other deep multi-task learning frameworks such as Faster R-CNN for object detection.

Acknowledgment

This work was supported by the MOBIDEM project, part of the "Systematic Paris-Region" and "Images & Network" Clusters, funded by the French government.

References

[1] T. Ahonen, A. Hadid, and M. Pietikainen. Face description with local binary patterns: Application to face recognition. *IEEE transactions on pattern analysis and machine intelligence*, 28(12):2037–2041, 2006.

[2] M. Bicego, A. Lagorio, E. Grosso, and M. Tistarelli. On the use of sift features for face authentication. In *Computer Vision and Pattern Recognition Workshop, 2006. CVPRW'06. Conference on*, pages 35–35. IEEE, 2006.

[3] K. I. Chang, K. W. Bowyer, and P. J. Flynn. Multiple nose region matching for 3d face recognition under varying facial expression. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(10):1695–1700, 2006.

[4] W. Chen, X. Chen, J. Zhang, and K. Huang. A multi-task deep network for person re-identification. In *AAAI*, pages 3988–3994, 2017.

[5] Z. Chen, V. Badrinarayanan, C.-Y. Lee, and A. Rabinovich. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. *arXiv preprint arXiv:1711.02257*, 2017.

[6] R. Collobert and J. Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167. ACM, 2008.

[7] L. Deng, G. Hinton, and B. Kingsbury. New types of deep neural network learning for speech recognition and related applications: An overview. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 8599–8603. IEEE, 2013.

[8] O. Déniz, G. Bueno, J. Salido, and F. De la Torre. Face recognition using histograms of oriented gradients. *Pattern Recognition Letters*, 32(12):1598–1603, 2011.

[9] X. Feng, M. Pietikäinen, and A. Hadid. Facial expression recognition based on local binary patterns. *Pattern Recognition and Image Analysis*, 17(4):592–598, 2007.

[10] R. Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[11] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In *13th International Conference on Artificial Intelligence and Statistics*, pages 249–256, 2010.

[12] I. J. Goodfellow, D. Erhan, and et al. Challenges in representation learning: A report on three machine learning contests. In *International Conference on Neural Information Processing*, pages 117–124, 2013.

[13] Y. Guo, L. Zhang, Y. Hu, X. He, and J. Gao. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In *European Conference on Computer Vision*, pages 87–102. Springer, 2016.

[14] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical report, Technical Report 07-49, University of Massachusetts, Amherst, 2007.

[15] H. Jung, S. Lee, and et al. Joint fine-tuning in deep neural networks for facial expression recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2983–2991, 2015.

[16] I. A. Kakadiaris, G. Passalis, G. Toderici, M. N. Murtuza, Y. Lu, N. Karampatziakis, and T. Theoharis. Three-dimensional face recognition in the presence of facial expressions: An annotated deformable model approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(4):640–649, 2007.

[17] A. Kendall, Y. Gal, and R. Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7482–7491, 2018.

[18] A. Klaser, M. Marszałek, and C. Schmid. A spatio-temporal descriptor based on 3d-gradients. In *BMVC 2008-19th British Machine Vision Conference*, pages 275–1. British Machine Vision Association, 2008.

[19] M. Liu, S. Li, and et al. Au-aware deep networks for facial expression recognition. In *Automatic Face and Gesture Recognition, 2013 IEEE International Conference and Workshops on*, pages 1–6. IEEE, 2013.

[20] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song. Sphereface: Deep hypersphere embedding for face recognition. In *The CVPR*, volume 1, page 1, 2017.

[21] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2015.

[22] P. Lucey, J. F. Cohn, and et al. The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In *CVPR Workshops, 2010 IEEE Computer Society Conference on*, pages 94–101. IEEE, 2010.

[23] A. Mollahosseini, D. Chan, and et al. Going deeper in facial expression recognition using deep neural networks. In *Applications of Computer Vision, 2016 IEEE Winter Conference on*, pages 1–10. IEEE, 2016.

[24] K. Murugesan, H. Liu, J. Carbonell, and Y. Yang. Adaptive smoothed online multi-task learning. In *Advances in Neural Information Processing Systems*, pages 4296–4304, 2016.

[25] O. M. Parkhi, A. Vedaldi, A. Zisserman, et al. Deep face recognition. In *BMVC*, page 6, 2015.

[26] R. Ranjan, V. M. Patel, and R. Chellappa. 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*, 2017.

[27] S. Ruder. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*, 2017.

[28] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, pages 815–823, 2015.

[29] K. Simonyan and O. M. e. a. Parkhi. Fisher vector faces in the wild. In *BMVC*, page 4, 2013.

[30] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[31] Y. Sun, X. Wang, and X. Tang. Deeply learned face representations are sparse, selective, and robust. In *CVPR*, pages 2892–2900, 2015.

[32] Y. Taigman, M. Yang, and et al. Deepface: Closing the gap to human-level performance in face verification. In *CVPR*, pages 1701–1708, 2014.

[33] Y. Tian, P. Luo, X. Wang, and X. Tang. Pedestrian detection aided by deep learning semantic tasks. In *Proceedings of the CVPR*, pages 5079–5087, 2015.

[34] Y. Wen, K. Zhang, Z. Li, and Y. Qiao. A discriminative feature learning approach for deep face recognition. In *European Conference on Computer Vision*, pages 499–515. Springer, 2016.

[35] L. Wolf, T. Hassner, and I. Maoz. Face recognition in unconstrained videos with matched background similarity. In *CVPR, 2011 IEEE Conference on*, pages 529–534. IEEE, 2011.

[36] J. Yim, H. Jung, B. Yoo, C. Choi, D. Park, and J. Kim. Rotating your face using multi-task deep neural network. In *Proceedings of the CVPR*, pages 676–684, 2015.

[37] X. Yin and X. Liu. Multi-task convolutional neural network for pose-invariant face recognition. *IEEE Transactions on Image Processing*, 27(2):964–975, 2018.

[38] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *Signal Processing Letters*, 23(10):1499–1503, 2016.

[39] Z. Zhang, P. Luo, C. C. Loy, and X. Tang. Learning deep representation for face alignment with auxiliary attributes. *IEEE transactions on pattern analysis and machine intelligence*, 38(5):918–930, 2016.

[40] G. Zhao, X. Huang, and et al. Facial expression recognition from near-infrared videos. *Image and Vision Computing*, 29(9):607–619, 2011.

[41] X. Zhao, X. Liang, and et al. Peak-piloted deep network for facial expression recognition. In *European Conference on Computer Vision*, pages 425–442. Springer, 2016.

[42] 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*, pages 787–796, 2015.