Deep Multi-Task Learning to Recognise Subtle Facial Expressions of Mental States

Guosheng Hu[1,2], Li Liu[3], Yang Yuan[1], Zehao Yu[4], Yang Hua[2], Zhihong Zhang[4], Fumin Shen[5], Ling Shao[3], Timothy Hospedales[6], Neil Robertson[2,1], Yongxin Yang[6,7]

[1] Anyvision, Queens Road, Belfast, UK huguosheng100@gmail.com
[2] ECIT, Queens University of Belfast, UK
[3] Inception Institute of Artificial Intelligence, Abu Dhabi, UAE
[4] Software Department, Xiamen University, Xiamen, China
[5] University of Electronic Science and Technology of China, Chengdu, China
[6] School of Informatics, University of Edinburgh, Edinburgh, UK
[7] Yang's Accounting Consultancy Ltd, London, UK

Abstract. Facial expression recognition is a topical task. However, very little research investigates subtle expression recognition, which is important for mental activity analysis, deception detection, etc. We address subtle expression recognition through convolutional neural networks (CNNs) by developing multi-task learning (MTL) methods to effectively leverage a side task: facial landmark detection. Existing MTL methods follow a design pattern of shared bottom CNN layers and task-specific top layers. However, the sharing architecture is usually heuristically chosen, as it is difficult to decide which layers should be shared. Our approach is composed of (1) a novel MTL framework that automatically learns which layers to share through optimisation under tensor trace norm regularisation and (2) an invariant representation learning approach that allows the CNN to leverage tasks defined on disjoint datasets without suffering from dataset distribution shift. To advance subtle expression recognition, we contribute a Large-scale Subtle Emotions and Mental States in the Wild database (LSEMSW). LSEMSW includes a variety of cognitive states as well as basic emotions. It contains 176K images, manually annotated with 13 emotions, and thus provides the first *subtle* expression dataset large enough for training deep CNNs. Evaluations on LSEMSW and 300-W (landmark) databases show the effectiveness of the proposed methods. In addition, we investigate transferring knowledge learned from LSEMSW database to traditional (non-subtle) expression recognition. We achieve very competitive performance on Oulu-Casia NIR&Vis and CK+ databases via transfer learning.

1 Introduction

Facial expressions convey important information about the emotional and mental states of a person. Facial expression understanding has wide applications and has been most widely studied in the form of emotion recognition. The classic problem

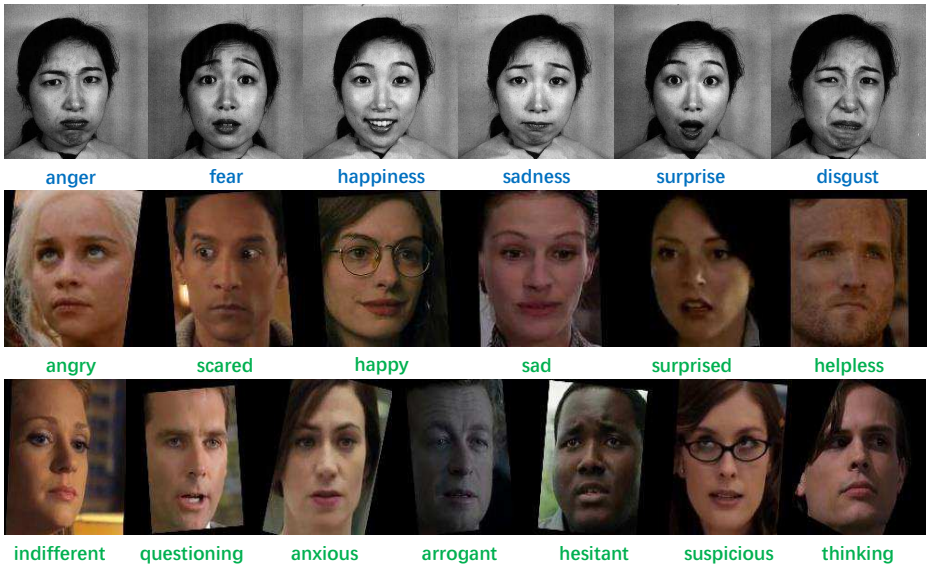


Fig. 1: Conventional emotion recognition (Row 1) vs our LSEMSW (Row 2 and 3). Our dataset contains mental states that are richer – including a variety of cognitive states, conveyed by *subtle* expressions, and exhibited in the wild.

is to recognise six basic emotions [39] (Fig. 1, top) based on facial expressions [21]. This problem is now considered solved for posed, exaggerated expressions in lab conditions (Fig. 1, top); but is still an open question for realistic subtly exhibited expressions where, for example, even a slight tightening of the lips can be a sign that someone is angry. Subtle expressions (Fig. 1, Row 2&3) are important for mental activity analysis and deception detection [55].

In this paper we go significantly beyond existing work on emotion recognition in two ways, to address: Recognition of *subtly* expressed rather than exaggerated emotions; and recognition of a wider range of mental states beyond the basic six emotions, including for the first time cognitive states. To address these goals we work from two directions: providing (1) improved deep learning algorithms, and (2) a big dataset covering subtle emotions and cognitive states. Specifically, *first*, we introduce a new deep learning approach to subtle expression recognition, based on a novel multi-task learning architecture to exploit a side task: landmark detection. *Second*, in order to benchmark the new proposed task, and to train our deep learning model, we also contribute a new large scale dataset: Large-scale Subtle Emotions and Mental States in the Wild (LSEMSW). The expressions of this database are much more realistically subtle compared to existing posed and exaggerated benchmarks (Fig 1). The dataset is also much richer than most existing benchmarks – containing 13 emotions and cognitive states (Fig. 1 Row 2 & 3) defined by 2 psychologists.

Deep MTL Algorithms We build on deep CNNs, which have achieved great success in many vision tasks. As per classic studies that use landmark locations [54] and distances [50] for basic emotion recognition; we observe that when emotional/mental state is conveyed by subtle expressions, a salient cue is often slight movements of facial landmarks (e.g., eye widening). To provide this prior knowledge as an inductive bias in our deep network design, we aim to detect landmarks and mental state simultaneously via multi-task learning (MTL).

Classic MTL methods focused on improving performance by cross-task knowledge transfer through shared task representations in linear models [2] or kernel-based nonlinear models [10]. More recently, interest in MTL for neural networks has been grown in popularity (again [5]), in order to combine the power of deep networks and knowledge sharing [61, 24, 27]. The conventional deep MTL predefines the first few CNN layers as shared by multiple tasks, and then forks into different layers for different tasks with different losses. However, this approach is heuristic and, without theoretical guidance about how to choose sharing structure, it is often left to 'grad student descent'. This is particularly tricky with increasingly deep CNN architectures. For example ResNet [17] has 156 layers, leading to 156 possible architectures assuming exactly one fork, or $(B_T)^{156}$ architectures (where T is the number of tasks and $B_\cdot$ is the Bell number, i.e., the number of partitions of a set) more generally. To address this, we develop a new tensor trace norm approach that automatically determines how much every layer should be shared, and without assuming a single fork point.

Furthermore we address the issue that MTL typically requires all tasks to be annotated on a single dataset to be effective. If the tasks are associated with different datasets, MTL can still be applied but it is ineffective due to the negative effect of cross-dataset distribution shift outweighing the benefit of MTL-based knowledge sharing. By integrating a distribution alignment strategy [13], we can use disjoint training sets (tasks defined on different datasets), thus making MTL much more flexible and widely applicable. In the context of emotion recognition, this allows us to leverage existing datasets to provide auxiliary tasks such as facial landmark localisation in 300-W dataset [45].

***Subtle* Expression Database** Most existing expression databases [8, 20, 11, 37] only contain images with strong expression of exaggerated emotions, and *subtle* expression analysis is rarely investigated. To address this gap, we contribute LSEMSW, the first big database for *subtle* expression analysis. LSEMSW only contains images with realistically subtle expressions. In addition, the existing databases have some limitations: they either contain only emotions without other mental states [8, 11], are noisy due to automated annotation [11], or are too small for deep learning [8]. Our LSEMSW contains other (non-emotional) cognitive mental states, compared to existing datasets focusing on six basic emotions [8]. LSEMSW contains 176K images, making it multiple orders of magnitude larger than some alternatives (E.g., 1500 images in AFEW [8]), and all images are manually labelled rather than automatically annotated by algorithms [11]. Finally, we contrast *micro-expression* recognition, which is to recognise an emotion

that a person is trying to conceal [25]. This is related in addressing subtle cues, but different in that it is typically performed on video rather than images.

Contributions In summary, our contributions are: (i) Unlike standard heuristically designed deep MTL, we propose an end-to-end soft sharing strategy that flexibly learns where, what and how much to share by optimising the trace norm regularised parameters. We further embed a distribution alignment method in order to maintain good performance when the per-task training sets are disjoint. (ii) We contribute our LSEMSW dataset consisting of 176K images manually annotated with 13 emotions and cognitive states. This is the first database for *subtle* expression analysis, the first database for recognising *cognitive* states from facial expressions, and it is big enough for deep CNN training. We will release this database to advance mental state recognition in the deep learning era. In addition, the source code and trained models will be made publicly available. (iii) We show that LSEMSW can benefit Traditional (non-subtle) expression recognition (TNER), by using transfer learning to achieve very competitive TNER performance on Oulu-Casia NIR&Vis [62] and CK+ [29] databases.

2 Methodology

2.1 Preliminaries

Matrix-based Multi-Task Learning Matrix-based MTL is usually built on linear models, i.e., each task is parameterised by a D-dimensional weight vector $\mathbf{w}$, and the model prediction is $\hat{y} = \mathbf{x}^T\mathbf{w}$, where $\mathbf{x}$ is a D-dimensional feature vector representing an instance. The objective function for matrix-based MTL can be written as $\sum_{i=1}^{T}\sum_{j=1}^{N^{(i)}} \ell(y_j^{(i)}, \mathbf{x}_j^{(i)} \cdot \mathbf{w}^{(i)}) + \lambda\Omega(W)$. Here $\ell(y, \hat{y})$ is a loss function of the true label $\mathbf{y}$ and predicted label $\hat{\mathbf{y}}$. T is the number of tasks, and for the i-th task there are $N^{(i)}$ training instances. Assuming the dimensionality of every task's feature is the same, the models $\mathbf{w}^{(i)}$ are of the same size. Then the collection of $\mathbf{w}^{(i)}$s forms a $D \times T$ matrix W where the i-th column is a linear model for the i-th task. A regulariser $\Omega(W)$ is exploited to encourage W to be a low-rank matrix. Some choices include the $\ell_{2,1}$ norm [2], and trace norm [19].

Tensor-based Multi-Task Learning In standard MTL, each task is indexed by a single factor. But in some real-world problems, tasks are indexed by multiple factors. The collection of linear models for all tasks is then a 3-way tensor $\mathcal{W}$ of size $D \times T_1 \times T_2$, where T_1 and T_2 are two task indices. In this case, tensor norm regularisers $\Omega(\mathcal{W})$ have been used [51]. For example, sum of the trace norms on all matriciations [44] and scaled latent trace norm [56]. However, such prior tensor norm-based regularisers have been limited to shallow models. We develop methods to allow application of tensor norms end-to-end in deep networks.

Deep Multi-Task Learning With the success of deep learning, many studies have investigated deep MTL [28, 61, 41, 36, 58]. E.g., using a CNN to find facial landmarks as well as recognise facial attributes [61, 41]. The standard approach [28, 61, 41] is to share the bottom layers of a deep network and use ask-specific parameters for the top layers. We call this type of 'predefined' sharing strategy 'hard' sharing. This 'hard' sharing based architecture can be traced back to 2000s

[3]. However, it is impossible to try every hard sharing possibility in modern CNN architectures with many layers. Limited very recent work on automating deep MTL [58, 36] suffers from the need to specify discrete ranks at every layer. This introduces an additional sharing-strength hyper-parameter per-layer, and crucially prevents knowledge sharing when working with only two tasks, as it increases rather than decreases the number of parameters. Our approach learns soft sharing at all layers controlled by a single sharing strength hyper-parameter.

2.2 Trace Norm-based Knowledge Sharing for Deep MTL

In this work, we focus on deep MTL, in particular, CNN-based MTL. One CNN contains multiple convolution layers, each consisting of a number of convolutional kernels. A convolutional layer is parameterised by a 4-way tensor of size $H \times W \times C \times M$ where H, W, C, M are the height, width, number of channels, number of filters respectively. Since convolutional layers are structured as tensors, we use tensor-based theory, in particular, tensor trace norm, to achieve knowledge sharing. Unlike 'hard sharing' strategy, we propose a flexible 'soft' parameter sharing strategy that automatically learns where, what and how much to share by optimising the tensor trace norm regularised parameters.

Knowledge Sharing To *learn* a parameter sharing strategy, we propose the following framework: For T tasks, each is modelled by a neural network of the same architecture. The T networks are stacked horizontally in a layer-wise fashion, i.e. we assume the architectures of different tasks' networks are the same, so that we can collect the parameters in the same level (layer) then stack them to form a one-order higher tensor, e.g., for convolution layer, $4D \rightarrow 5D$. This process is repeated for every layer. With this stacking of parameters into higher order tensors, we can apply a tensor trace norm regulariser to each in order to achieve knowledge sharing. A schematic example with 2-task learning is illustrated in Fig. 2. Learning the CNN with tensor trace norm regularisation means that the ranks of these tensors are minimised where possible, and thus knowledge is shared where possible. Since trace norm is performed on the stacked parameters of all the layers, we can control the parameter sharing for all layers with a single hyperparameter of regularisation strength.

Tensor Norms Since tensor trace norm is the core of our approach, we review this topic. Matrix trace norm is the sum of a matrix's singular values $||X||_* = \sum_{i=1} \sigma_i$. It is the tightest convex relation of matrix rank [42]. Thus when directly restricting the rank of a matrix is challenging, trace norm serves as a good proxy. The trace norm of a tensor can be formulated as the sum of trace norms of matrices. However unlike for matrices, the trace norm of a tensor is not unique because tensors can be factored in many ways e.g., Tucker [53] and Tensor-Train [38] decompositions. We propose three tensor trace norms here, corresponding to three variants of the proposed method.

For an N-way tensor $\mathcal{W}$ of size $D_1 \times D_2 \times \cdots \times D_N$. We define

$$\textbf{Last Axis Flattening (LAF)} \quad ||\mathcal{W}||_* = \gamma ||\mathcal{W}_{(N)}||_* \tag{1}$$

where $\mathcal{W}_{(i)} := \text{reshape}(\text{permute}(\mathcal{W}, [i, 1, \ldots, i-1, i+1 \ldots, N]), [D_i, \prod_{j \neg i} D_j])$ is the mode-i tensor flattening. This is the simplest definition. Given that in our framework, the last axis of tensor indexes the tasks, i.e., $D_N = T$, it is the most straightforward way to adapt matrix-based MTL – i.e. by reshaping the $D_1 \times D_2 \times \cdots \times T$ tensor to $D_1 D_2 \cdots \times T$ matrix.

To advance, we define two kinds of tensor trace norm that are closely connected with Tucker-rank (obtained by Tucker decomposition) and TT-rank (obtained by Tensor Train decomposition).

$$\textbf{Tucker} \quad ||\mathcal{W}||_* = \sum_{i=1}^{N} \gamma_i ||\mathcal{W}_{(i)}||_* \tag{2}$$

$$\textbf{TT} \quad ||\mathcal{W}||_* = \sum_{i=1}^{N-1} \gamma_i ||\mathcal{W}_{[i]}||_* \tag{3}$$

Here $\mathcal{W}_{[i]}$ is yet another way to unfold the tensor, which is obtained by $\mathcal{W}_{[i]} = \text{reshape}(\mathcal{W}, [D_1 D_2 \ldots D_i, D_{i+1} D_{i+2} \ldots D_N])$. Note that unlike LAF, Tucker and TT also encourage within-task parameter sharing, e.g., sharing across filters in a neural network context.

Optimisation For the regularisers defined in Eqs. (1)-(3), we see that tensor trace norm is formulated as the sum of matrix trace norms. Gradient-based methods are not commonly used to optimise matrix trace norm. However in order to apply trace norm-based regularisation end-to-end in CNNs, we wish to optimise trace norm and standard CNN losses using a single gradient-based optimiser such as Tensorflow [1]. Thus we derive a (sub-)gradient descent method for trace norm minimisation.

We start from an equivalent definition of trace norm instead of the sum of singular values, $\|W\|_* = \text{Trace}((W^T W)^{\frac{1}{2}}) = \text{Trace}((W W^T)^{\frac{1}{2}})$ where $(\cdot)^{\frac{1}{2}}$ is the matrix square root. Given the property of the differential of the trace function, $\partial \,\text{Trace}(f(A)) = f'(A^T) : \partial A$, where the colon : denotes the double-dot (a.k.a. Frobenius) product, i.e., $A : B = \text{Trace}(AB^T)$. In this case, $A = W^T W$, $f(\cdot) = (\cdot)^{\frac{1}{2}}$ thus $f'(\cdot) = \frac{1}{2}(\cdot)^{-\frac{1}{2}}$, so we have,

$$\partial \,\text{Trace}((W^T W)^{\frac{1}{2}}) = \frac{1}{2}(W^T W)^{-\frac{1}{2}} : \partial(W^T W) = W(W^T W)^{-\frac{1}{2}} : \partial W$$

Therefore we have $\frac{\partial \|W\|_*}{\partial W} = W(W^T W)^{-\frac{1}{2}}$. In the case that $W^T W$ is not invertible, we can derive that $\frac{\partial \|W\|_*}{\partial W} = (W W^T)^{-\frac{1}{2}} W$ similarly. To avoid the check on whether $W^T W$ is invertible, and more importantly, to avoid the explicit computation of the matrix square root, which is usually not numerically safe, we use the following procedure.

First, we assume W is an $N \times P$ matrix ($N > P$) and let the (full) SVD of W be $W = U \Sigma V^T$. Σ is an $N \times P$ matrix in the form of $\Sigma = [\Sigma_*; \mathbf{0}_{(N-P)\times P}]$. Then we have

$$\begin{aligned} W(W^T W)^{-\frac{1}{2}} &= U \Sigma V^T (V \Sigma_*^2 V^T)^{-\frac{1}{2}} = U \Sigma V^T V \Sigma_*^{-1} V^T \\ &= U \Sigma \Sigma_*^{-1} V^T = U [I_P; \mathbf{0}_{(N-P)\times P}] V^T \end{aligned}$$

This indicates that we only need to compute the truncated SVD, i.e., $W = U_* \Sigma_* V_*^T$, and $W(W^TW)^{-\frac{1}{2}} = U_* V_*^T$. For the case when $N < P$, we have the same result as,

$$\begin{aligned}(WW^T)^{-\frac{1}{2}}W &= (U\Sigma_*^2 U^T)^{-\frac{1}{2}} U\Sigma V^T = U\Sigma_*^{-1}U^T U\Sigma V^T \\ &= U\Sigma_*^{-1}\Sigma V^T = U[I_N, \mathbf{0}_{(P-N)\times N}]V^T\end{aligned}$$

Now we have an agreed formulation: $\frac{\partial \|W\|_*}{\partial W} = U_* V_*^T$ that we can use for gradient descent. Though exact SVD is expensive, we find that a fast randomized SVD [16] works well in practice.

2.3 Adversarial Domain Alignment (ADA)

In our application, the main task's dataset (LSEMSW) is disjoint to the auxiliary task's (300-W) [45]. This leads to the distribution shift problem across the two tasks, reducing the performance of MTL. Inspired by [14] and [13], we propose to confuse the dataset identity for dealing with this problem.

We use ADA to solve this problem: One classifier aims to distinguish which distribution (dataset) the features of each task are from. If features are distinguishable the domain shift is clearly greater than if they are indistinguishable. ADA trains them to be indistinguishable. We assume $T \geq 2$ tasks (indexed by t), each with its own dataset $\{X_t, y_t\}$. Task t is modelled by a CNN parametrised by $\Theta_t = \{\theta_t^{(1)}, \theta_t^{(2)}, \ldots, \theta_t^{(L)}\}$ where L is the number of layers, and we split Θ_t into two sets at the l-th layer. Conventionally we choose $l = L-1$, i.e., the penultimate layer, so we have $\Theta_t = \Theta_t^* \cup \{\theta_t^{(L)}\}$ where $\Theta_t^* = \{\theta_t^{(1)}, \theta_t^{(2)}, \ldots, \theta_t^{(L-1)}\}$. We then build a multi-class classification problem that uses a neural network parametrised by Φ to predict the database identity from $f_{\Theta_t^*}(X_t)$, the penultimate layer representation. Letting Z be the stacked features for all tasks i.e., $Z = [f_{\Theta_t^*}(X_1) \ldots f_{\Theta_t^*}(X_T)]$, we optimise

$$\max_{\Theta_1^* \ldots \Theta_T^*} \min_{\Phi} \ell(g_\Phi([f_{\Theta_1^*}(X_1) \ldots f_{\Theta_T^*}(X_T)]), y) \tag{4}$$

where y is one-hot label to indicate which distribution the feature is from; g_Φ is a classifier, e.g. softmax; ℓ is a cross entropy loss.

For our application, we have 2 tasks in total, so it is reduced to a binary classification problem. For the task identity prediction neural network, we use a 2-hidden-layer MLP (multilayer perceptron) with 512 (input feature) -128 (hidden layer)-64 (hidden layer) -2 (classifier) architecture.

2.4 CNN Architecture for Deep MTL

In this study, we implement our deep MTL based on the well known Residual Network (ResNet) architecture [17]. We use the compact 34-layer ResNet with 33 convolutional layers and 1 fully connected layer detailed in [17]. We perform trace norm on the weights of all the 33 shareable convolutional layers of the

stacked networks. In addition, the original 34-layer ResNet has a 7×7 global average pooling before loss layer, adapting to the 224×224 input. To adapt to our 96×96 input, we use 3×3 average pooling instead. The Adversarial Domain Alignment is performed on the activations (feature map) of this average pooling. The classification loss for mental state recognition is softmax cross-entropy loss, while the loss for landmark detection is l_1 regression loss. The architecture is shown in Fig. 2.

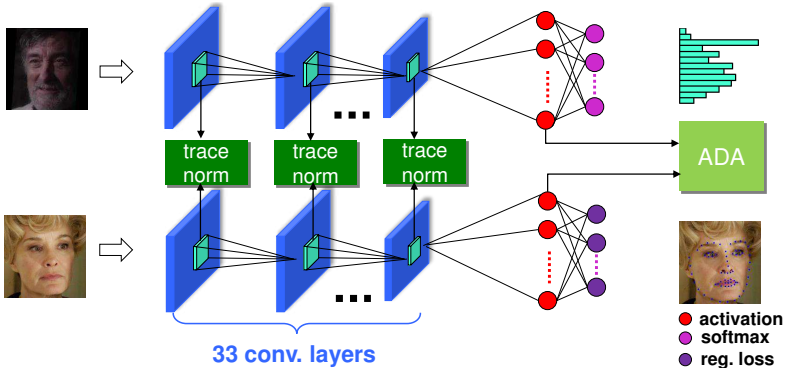


Fig. 2: Our deep MTL framework. For simplicity, layers such as pooling, relu, etc. are not visualised. 'activation'$\in R^{512}$ denotes the feature map after global average pooling.

3 Large-scale Subtle Emotions and Mental States in the Wild (LSEMSW) Database

Motivation Subtle expressions occur when a person's emotional response to surroundings is of low intensity. People usually exhibit subtle expressions when they start to feel an emotion. Subtle expression recognition has many applications such as mental activity analysis and deception detection [55]. However, subtle expressions are rarely investigated and the existing facial expression analysis techniques mainly focus on strong or exaggerated expressions. To advance research on subtle expression analysis, we collect the new LSEMSW database.

Collection and Annotation LSEMSW was collected from more than 200 movies and TV serials such as Big Bang Theory, Harry Potter, Game of Thrones, etc. For each video/clip, we selected the first frame of every 5 ones. Then face detection was performed on the selected frames using MTCNN [60]. The images that contain faces were manually annotated via Amazon Mechanical Turk over nine months. To achieve accurate annotation, we provided detailed instructions to annotators and used Amazon MT Master service to select well-performing reliable annotators based on their historical performance. Each image is assigned

to 3 workers for annotation. During the annotation, the subtitle (if available) on the frame is shown to help the workers make their decision. An annotation is accepted only if more than two workers agree on the annotation. The images with strong expressions were manually filtered. More details of our database are shown in Table 1 and 2 and in the supplementary material.

Table 1: Attribute Distribution.

Gender	Male	64.1%
	Female	35.9%
Age	Child	1.5%
	Young	55.9%
	Adult	42.6%
Ethinity	Black	1.3%
	White	31.9%
	Asian	66.2%
	Mixed	0.6%

Table 2: Expression Distribution.

Expression	# Images	Expression	# Images
Happy	22,378	Surprised	13,712
Anxious	11,776	Arrogant	11,240
Sad	10,392	Thinking	31,645
Scared	12,190	Helpless	10,699
Angry	9,014	Suspicious	12,666
Hesitant	7,365	Questioning	10,288
Indifferent	12,314	Total	175,679

Comparison with Existing Databases We compare LSEMSW with existing well known expression/emotion databases in Table 3. We can see that our LSEMSW is the only one with *subtle* expressions rather than strong expressions. Although this research focuses on *subtle* expression recognition, the knowledge learned from LSEMSW can be transferred to standard strong expression recognition, as verified in Section 4.2. In terms of size, LSEMSW is smaller than EmotioNet [11] and AffectNet [37]. However while EmotioNet [11] contains 1 million images, only 50K are manually annotated and the labels of the remaining images are noisily predicted by algorithm [4]. Therefore, our database is the second largest with manual expression annotations. It is the only database with cognitive state annotations.

4 Experiments

4.1 Databases and Settings

Expressions We explore two types of expression recognition: (1) subtle expression recognition and (2) traditional (non-subtle) expression recognition (TNER). For (1), our LSEMSW database is used for evaluation. Specifically, the database is divided to training, validation and test sets according to the ratio: 80%, 10% and 10%. The rank 1 recognition rate on test set is reported. For (2), we explore transferring representation learned from LSEMSW to TNER. Specifically, we train TNER networks by finetuning from the *subtle* expression network trained with LSEMSW. We use two well known TNER databases, Oulu-Casia NIR&Vis (OCNV) facial expression database [62] and Extended Cohn-Kanade database (CK+) [29], for this evaluation. OCNV contains 480 sequences taken under 3 lightings: dark, strong and weak. Following [9], we use VIS videos with

Table 3: Comparison of manually annotated facial expression databases.

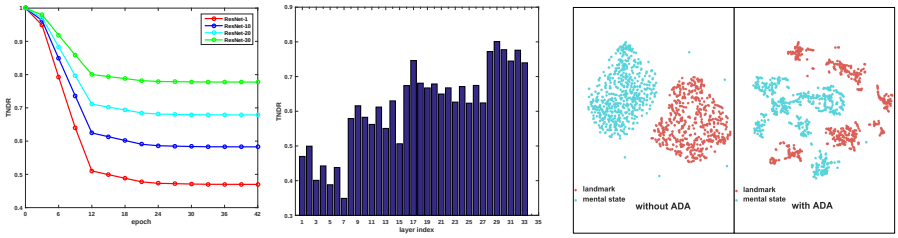
Database	Expr. Intensity	Expr. Type	# Expr.	# Images	Environment
JAFFE [32]	strong	emotions	7	213	controlled
SFEW [7]	strong	emotions	7	663	uncontrolled
DISFA [34]	strong	emotions	7	4,845	controlled
FER2013 [15]	strong	emotions	7	36K	uncontrolled
RAF-DB [23]	strong	emotions	18	30K	uncontrolled
EmotioNet [11]	strong	emotions	16	50K (950K)[1]	uncontrolled
AffectNet [37]	strong	emotions	7	450K (1M)[2]	uncontrolled
LSEMSW	subtle	emotions & cognitive states	13	176K	uncontrolled

[1] 50K images are manually annotated, and the labels of 950K images are predicted by algorithm [4].
[2] 450K of 1M images are manually annotated with emotions, valence and arousal.

strong lighting (80 identities and 6 expressions). Each image sequence varies from neutral to peak formation of one particular expression. The last three frames (strongest expression) are used. 10-fold cross validation is conducted as [9]. On the other hand, CK+ includes 593 video sequences of 123 subjects. Subjects displayed 7 basic (non subtle) expressions in different sequences. We use only the last (strongest) frame of the sequence. Following [23], 5-fold cross validation is conducted. During training, data augmentation (flip, crop, rotation), which we find is very important, is performed. We finetune the LSEMSW-pretrained network using the augmented training images of OCNV and CK+ and evaluate the performance on the testing images of these 2 databases. Evaluations are reported on task (1) except where explicitly specified.

Facial Landmarks We use 68-point annotations [46] for landmark detection. Our training set consists of the training images of 300 Faces In-the-Wild Challenge (300-W) [46] and Menpo benchmark [59]. Face detection using MTCNN [60] is performed on original training images. The detected bounding boxes are extended with a scale ratio of 0.2, aiming to cover the whole face area. Due to the limited training images, data argumentation is important. The detected faces are flipped, rotated (-30°, 30°), and disturbed via translation and scaling (0.8, 1.2). During training, the landmark coordinates are normalised to (0, 1). Following [52, 18], the test set contains 3 parts: common subset (554 images), challenging subset (135 images) and full set (689 images). Where not explicitly specified, we reports the results on the full set. Following [52, 18], we use the normalised mean error (the distance between estimated landmarks and the ground truths, normalised by the inter-pupil distance) to evaluate the result.

Implementation Details Our end-to-end deep MTL framework is implemented in TensorFlow [1]. The training images for mental state recognition are aligned and cropped to 96×96. Similarly, the images for landmark detection are resized to 96×96 and landmark coordinates are whitened following [43]. The landmark detection data is augmented by horizontal flip, rotation, scale, shift, and adding Gaussian noise following [12]. Only horizontal flip is used for emo-



(a) TNDR against epochs (b) TNDR after optimisation (c) Feature distributions of 2 test sets

Fig. 3: Trace Norm and ADA Analysis: Trace Norm changes during (a) and after (b) network optimisation. The feature distributions with and without ADA (c).

tion recognition. The learning rates for both networks are set 0.01, and batch sizes are both 256.

4.2 Results

Analysis of Discovered Sharing To analyse the learned sharing strategy, we define the trace norm decrease rate (TNDR) as $\frac{\text{Norm of Optimised Para}}{\text{Norm of Initialised Para}}$. The smaller TNDR is, the more knowledge one convolutional layer shares. We take ResNet+LAF as an example to investigate the properties of trace norm optimisation. Fig. 3a shows the TNDR decreases against network optimisation epochs. We choose the 1st, 10th, 20th, 30th layers LAF trace norms for analysis. Clearly, the 1st layer LAF decreases more dramatically than others, implying more knowledge is shared in the 1st layer. This is consistent with the common intuition that the lower layers capture more broadly useful low-level features. Fig. 3b shows the TNDR of all layers after learning. We observe that: (i) Overall TNDR is smaller (information sharing is greater) at earlier layers as expected. However this trend is continuous rather than discontinuous, supporting the value of continuously varying soft sharing rather than a discrete all-or-nothing fork. Surprisingly (ii) within each residual block TNDR decreases (sharing is less) at higher layers. By *learning* the parameter sharing, our method has discovered a surprising strategy – related to the ResNet block architecture – that a human engineer is unlikely to have tried.

Comparison with Other Deep MTL Methods Traditional Deep MTL methods use hand designed 'hard' parameter sharing which manually defines which layers are shared and which not. To contrast the manual approach, we compare 4 predefined architectures: 34-layer ResNets with the first {6, 14, 26, 32} convolutional layers shared and the rest task-specific. These increments are chosen to correspond to 4 residual units/blocks in [17]. From Table 4, we see that our automatic soft sharing (without ADA) works much better than 'hard' sharing in both tasks. Among 'hard' methods, ResNet (6) with the first 6 layers shared is best. The fact that such fairly limited sharing works best implies the cross-dataset domain-shift between the two tasks is strong, further

Table 4: Accuracy (%) of Mental State Recognition on LSEMSW using 34-layer ResNet. RN(#) indicates the number of shared layers in standard MTL baseline.

	Single task	Our soft sharing			Hard sharing				Cross stitch [36]
	RN	LAF	Tucker	TT	RN(6)	RN(14)	RN(26)	RN(32)	RN
No ADA	28.39	33.43	33.39	33.41	30.07	28.11	26.90	24.69	30.96
ADA	-	**36.72**	36.51	36.64	33.97	31.95	30.58	28.18	-

motivating our solution for domain invariant feature learning. We also implement the recent deep MTL method 'cross stitch MTL' [36] using the same ResNet. From Table 4, we can see that our MTL strategy outperforms 'cross stitch MTL'. This is because our trace-norm based strategy provides more fine-grained control of information sharing compared to discrete rank setting.

Trace Norm Comparison A key contribution of this work is multi-task parameter sharing through trace norm. Here we compare the three trace norms (LAF, Tucker, TT) introduced in Section 2.2 without ADA. The baseline single task method is 34-layer ResNet without any parameter sharing. From the results in Table 4, we can see our MTL methods (LAF, Tucker, TT) perform significantly better than single-task learning. Specifically, for mental state recognition, LAF, Tucker and TT achieve recognition accuracy around 33.4%, compared with 28.39% of single task learning. For landmark detection, LAF, Tucker and TT reduce the mean error rates vs single task by around 7%. The three trace norms achieve very similar performance. This means that our strategy is not very sensitive to the type of norm/factorisation. The similar performance of TT and Tucker to LAF also mean that there is not much gain from compressing across filters rather than tasks – suggesting that ResNet is not overly 'wide' for mental state recognition. Thus we choose the simplest LAF for subsequent comparisons.

Adversarial Domain Alignment We proposed ADA to reduce the domain shift across training sets from the tasks. As shown in Table 4, our method ResNet+LAF+ADA achieved 36.72% mental state recognition accuracy and 4.64% mean error rate of landmark detection, compared with ResNet+LAF (33.43%, 4.67%), showing the effectiveness of ADA. To further investigate the effect of ADA, we visualise the data distributions using t-SNE [33] technique. From Fig. 3c, we compare the feature distributions of two test sets (mental state and landmark) with (ResNet+LAF+ADA) and without (ResNet+LAF) using ADA. Clearly, ADA can effectively solve the domain shift problem.

Subtle Expression Recognition (SoA) Finally, we compare to prior state-of-the-art (SoA) methods. The historical lack of big training data, means that most prior approaches to expression/emotion recognition use handcrafted features such as LPQ [6], LBP [47], EOH [35]. A very recent study [40] empirically showed deep learning methods (AlexNet, VGGNet, ResNet) to be effective. Therefore, we compare the proposed method with all these networks. As subtle expression recognition is very challenging, handcrafted features (LPQ, LBP and EOH) do not achieve promising performance. From Table 5, we see that

EOH [35] is the best handcrafted feature because EOH captures both spatial and texture information while LBP and LPQ only capture texture information. Nevertheless deep learning methods work better than handcrafted features because the deep features are trained end-to-end to capture subtle facial variations. Our proposed ResNet+LAF+ADA approach performs best overall. The superiority of ResNet+LAF+ADA against ResNet shows the effectiveness of our MTL strategy (LAF) and domain alignment strategy (ADA).

Table 5: Comparison of SoA methods on LSEMSW

Method		Acc (%)
Hand-crafted Features	LPQ [6]	10.86
	LBP [47]	10.53
	EOH [35]	13.47
Deep Learning	AlexNet [40, 22]	26.77
	VGGNet [40, 49]	28.07
	RN	28.39
	RN+LAF+ADA	**36.72**

Table 6: Error Rate (%) of Landmark detection on 300-W database.

Method	Common Subset	Challenging Subset	Full Set
TCDCN [61]	4.80	8.60	5.54
TSR [30]	4.36	7.56	4.99
RAR [57]	4.12	8.35	4.94
MSLPR [18]	**3.83**	**7.46**	**4.54**
Ours(l_2 loss)	4.09	7.51	4.76
Ours(l_1 loss)	**3.99**	**7.28**	**4.64**

Landmark Detection (SoA) Facial landmark detection is primarily performed to provide an auxiliary task for our main subtle expression recognition task. We nevertheless also evaluate landmark detection here. Some qualitative results are shown in Fig. 4. The images illustrate strong variation of expression, illumination, occlusion, and poses. We can see that our method (ResNet+LAF+ADA) is very robust to these variations. Some failure cases are also shown in Fig. 4. These are mainly caused by the combination of different strong variations, e.g expression+pose (row 2, col 5 & 6) and expression+pose+illumination (row 2, col 7). We also perform quantitative comparison to SoA methods in Table 6. From the results we can see that our method (RN+LAF+ADA) achieves very promising landmark detection performance. Specifically, we achieve the 2nd best performance on common subset and full set, and best on challenging subset, showing the robustness of our method on various challenging scenarios such as strong pose, illumination and expression variations – as illustrated in Fig. 4. The promising performance results from (1) the strong nonlinear modelling (regression) capacity of ResNet and (2) the effectiveness of LAF and ADA. Both (1) and (2) are also supported by Table 4. We also compare the different loss functions used by landmark detection. From Table 6, we can see that l_1 loss achieves better performance than l_2 loss.

Traditional (non-subtle) Expression Recognition (TNER) It is interesting to investigate transferring knowledge learned from LSEMSW to TNER. We finetune the LSEMSW-pretrained network using the augmented training images of Oulu-Casia NIR&Vis (OCNV) [62] and CK+ [29] facial expression databases and also 300-W for multi-task learning. From the results in Table 7, we can draw the following conclusions: (i) Finetuning from LSEMSW works

Fig. 4: Samples of Landmark Detection: Faces with expressions (row 1, col 1-2), illuminations (row 1, col 3-4), occlusions (row 1, col 5-6), poses (row 2, col 1-3) and failed cases (row 2, col 4-6)

Table 7: Comparison against state of the art on traditional non-subtle expression recognition. (FT) indicates fine-tuning from LSEMSW. (S) means training from scratch.

OCNV database		CK+ database	
Method	Acc. (%)	Method	Acc. (%)
LOMO [48]	82.1	FP+SAE [31]	91.1
PPDN [63]	84.6	AUDN [26]	93.7
FN2EN [9]	**87.7**	RAF [23]	**95.8**
RN (FT)	82.9	RN (FT)	93.2
RN+LAF (FT)	85.8	RN+LAF (FT)	95.3
RN+LAF+ADA (FT)	**87.1**	RN+LAF+ADA (FT)	**96.4**
RN+LAF+ADA (S)	76.0	RN+LAF+ADA (S)	86.3

significantly better than training from scratch: 87.1% vs 76.0% on OCNV and 96.4% vs 86.3%, thus confirming its benefit as a source of data for representation learning, even if the final goal is TNER. (ii) Our MTL based on LAF and ADA is also beneficial for this TNER task (RN+LAF+ADA (FT) vs RN (FT) scores 87.1% vs 82.9% on OCNV and 96.4% vs 93.2% on CK+), as well as subtle expression recognition. (iii) In terms of comparison to prior state of the art, we achieve very competitive TNER performance via our soft MTL method and fine-tuning from LSEMSW (although it exclusively contains subtle expressions). Our RN+LAF+ADA (FT) achieves state of the art performance on CK+ and second best on OCNV.

5 Conclusion

In summary we have contributed a large new database to advance subtle expression recognition in the deep learning era. A trace norm based MTL learning method is proposed and ADA is used for domain alignment. Extensive experiments have verified the effectiveness of the propose methods.

References

1. Abadi, M., Agarwal, A., Barham, et al.: Tensorflow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org
2. Argyriou, A., Evgeniou, T., Pontil, M.: Convex multi-task feature learning. Machine Learning (2008)
3. Bakker, B., Heskes, T.: Task clustering and gating for bayesian multitask learning. Journal of Machine Learning Research **4**, 83–99 (2003)
4. Benitez-Quiroz, C.F., Srinivasan, R., Feng, Q., Wang, Y., Martinez, A.M.: Emotionet challenge: Recognition of facial expressions of emotion in the wild. arXiv preprint arXiv:1703.01210 (2017)
5. Caruana, R.: Multitask learning. In: Learning to learn. Springer (1998)
6. Dhall, A., Asthana, A., Goecke, R., Gedeon, T.: Emotion recognition using phog and lpq features. In: Automatic Face & Gesture Recognition and Workshops (FG) (2011)
7. Dhall, A., Goecke, R., Lucey, S., Gedeon, T.: Static facial expression analysis in tough conditions: Data, evaluation protocol and benchmark. In: ICCV Workshops (2011)
8. Dhall, A., Goecke, R., Lucey, S., Gedeon, T.: Collecting large, richly annotated facial-expression databases from movies. IEEE MultiMedia **19**(3), 0034 (2012)
9. Ding, H., Zhou, S.K., Chellappa, R.: Facenet2expnet: Regularizing a deep face recognition net for expression recognition. In: FG (2017)
10. Evgeniou, T., Pontil, M.: Regularized multi–task learning. In: KDD (2004)
11. Fabian Benitez-Quiroz, C., Srinivasan, R., Martinez, A.M.: Emotionet: An accurate, real-time algorithm for the automatic annotation of a million facial expressions in the wild. In: CVPR. pp. 5562–5570 (2016)
12. Feng, Z.H., Kittler, J., Christmas, W., Huber, P., Wu, X.J.: Dynamic attention-controlled cascaded shape regression exploiting training data augmentation and fuzzy-set sample weighting. arXiv preprint arXiv:1611.05396 (2016)
13. Ganin, Y., Lempitsky, V.S.: Unsupervised domain adaptation by backpropagation. In: ICML (2015)
14. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: NIPS (2014)
15. Goodfellow, I.J., Erhan, e.a.: Challenges in representation learning: A report on three machine learning contests. In: ICONIP (2013)
16. Halko, N., Martinsson, P., Tropp, J.: Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. SIAM Review **53**(2), 217–288 (2011)
17. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
18. Huang, Z., Zhou, E., Cao, Z.: Coarse-to-fine face alignment with multi-scale local patch regression. arXiv preprint arXiv:1511.04901 (2015)
19. Ji, S., Ye, J.: An accelerated gradient method for trace norm minimization. In: ICML. pp. 457–464 (2009)
20. Kossaifi, J., Tzimiropoulos, G., Todorovic, S., Pantic, M.: Afew-va database for valence and arousal estimation in-the-wild. Image and Vision Computing (2017)
21. Krause, R.: Universals and cultural differences in the judgments of facial expressions of emotion. Journal of Personality and Social Psychology **5**(3), 4–712 (1987)
22. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: NIPS. pp. 1097–1105 (2012)

23. Li, S., Deng, W., Du, J.: Reliable crowdsourcing and deep locality-preserving learning for expression recognition in the wild. CVPR, July (2017)
24. Li, S., Liu, Z.Q., Chan, A.B.: Heterogeneous multi-task learning for human pose estimation with deep convolutional neural network. In: CVPR Workshops (2014)
25. Li, X., HONG, X., Moilanen, A., Huang, X., Pfister, T., Zhao, G., Pietikainen, M.: Towards reading hidden emotions: A comparative study of spontaneous micro-expression spotting and recognition methods. IEEE Transactions on Affective Computing (2017)
26. Liu, M., Li, S., Shan, S., Chen, X.: Au-inspired deep networks for facial expression feature learning. Neurocomputing **159**, 126–136 (2015)
27. Liu, W., Mei, T., Zhang, Y., Che, C., Luo, J.: Multi-task deep visual-semantic embedding for video thumbnail selection. In: CVPR (2015)
28. Liu, X., Gao, J., He, X., Deng, L., Duh, K., Wang, Y.Y.: Representation learning using multi-task deep neural networks for semantic classification and information retrieval. In: HLT-NAACL. pp. 912–921 (2015)
29. Lucey, P., Cohn, J.F., Kanade, T., Saragih, J., Ambadar, Z., Matthews, I.: The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In: CVPR Workshops (CVPRW) (2010)
30. Lv, J., Shao, X., Xing, J., Cheng, C., Zhou, X.: A deep regression architecture with two-stage re-initialization for high performance facial landmark detection. In: CVPR (2017)
31. Lv, Y., Feng, Z., Xu, C.: Facial expression recognition via deep learning. In: International Conference on Smart Computing (SMARTCOMP) (2014)
32. Lyons, M.J., Akamatsu, S., Kamachi, M., Gyoba, J., Budynek, J.: The japanese female facial expression (jaffe) database
33. Maaten, L.v.d., Hinton, G.: Visualizing data using t-sne. Journal of Machine Learning Research **9**(Nov), 2579–2605 (2008)
34. Mavadati, S.M., Mahoor, M.H., Bartlett, K., Trinh, P., Cohn, J.F.: Disfa: A spontaneous facial action intensity database. IEEE Transactions on Affective Computing **4**(2), 151–160 (2013)
35. Meng, H., Romera-Paredes, B., Bianchi-Berthouze, N.: Emotion recognition by two view svm_2k classifier on dynamic facial expression features. In: Automatic Face & Gesture Recognition and Workshops (FG), (2011)
36. Misra, I., Shrivastava, A., Gupta, A., Hebert, M.: Cross-stitch networks for multi-task learning. In: CVPR (2016)
37. Mollahosseini, A., Hasani, B., Mahoor, M.H.: Affectnet: A database for facial expression, valence, and arousal computing in the wild. arXiv preprint arXiv:1708.03985 (2017)
38. Oseledets, I.V.: Tensor-train decomposition. SIAM Journal on Scientific Computing **33**(5), 2295–2317 (2011)
39. Pantic, M., Rothkrantz, L.J.M.: Automatic analysis of facial expressions: the state of the art. TPAMI (2000)
40. Pramerdorfer, C., Kampel, M.: Facial expression recognition using convolutional neural networks: State of the art. arXiv preprint arXiv:1612.02903 (2016)
41. Rajeev Ranjan, Vishal M. Patel, R.C.: Hyperface: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition. arXiv (2016)
42. Recht, B., Fazel, M., Parrilo, P.A.: Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. SIAM review **52**(3), 471–501 (2010)

43. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: NIPS. pp. 91–99 (2015)
44. Romera-Paredes, B., Aung, H., Bianchi-Berthouze, N., Pontil, M.: Multilinear multitask learning. In: ICML (2013)
45. Sagonas, C., Antonakos, E., Tzimiropoulos, G., Zafeiriou, S., Pantic, M.: 300 faces in-the-wild challenge: database and results. Image and Vision Computing **47** (2016)
46. Sagonas, C., Tzimiropoulos, G., Zafeiriou, S., Pantic, M.: 300 faces in-the-wild challenge: The first facial landmark localization challenge. In: ICCV Workshops (2013)
47. Shan, C., Gong, S., McOwan, P.W.: Facial expression recognition based on local binary patterns: A comprehensive study. Image and Vision Computing **27**(6), 803–816 (2009)
48. Sikka, K., Sharma, G., Bartlett, M.: Lomo: Latent ordinal model for facial analysis in videos. In: CVPR (2016)
49. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
50. Tang, H., Huang, T.S.: 3d facial expression recognition based on automatically selected features. In: CVPR Workshops. pp. 1–8 (2008)
51. Tomioka, R., Hayashi, K., Kashima, H.: On the extension of trace norm to tensors. In: NIPS Workshop on Tensors, Kernels, and Machine Learning (2010)
52. Trigeorgis, G., Snape, P., Nicolaou, M.A., Antonakos, E., Zafeiriou, S.: Mnemonic descent method: A recurrent process applied for end-to-end face alignment. In: CVPR (2016)
53. Tucker, L.R.: Some mathematical notes on three-mode factor analysis. Psychometrika **31**(3), 279–311 (1966)
54. Walecki, R., Rudovic, O., Pavlovic, V., Pantic, M.: Variable-state latent conditional random fields for facial expression recognition and action unit detection. In: Automatic Face and Gesture Recognition (FG) (2015)
55. Warren, G., Schertler, E., Bull, P.: Detecting deception from emotional and unemotional cues. Journal of Nonverbal Behavior **33**(1), 59–69 (2009)
56. Wimalawarne, K., Sugiyama, M., Tomioka, R.: Multitask learning meets tensor factorization: task imputation via convex optimization. In: NIPS. pp. 2825–2833 (2014)
57. Xiao, S., Feng, J., Xing, J., Lai, H., Yan, S., Kassim, A.: Robust facial landmark detection via recurrent attentive-refinement networks. In: ECCV (2016)
58. Yang, Y., Hospedales, T.: Deep multi-task representation learning: A tensor factorisation approach. ICLR (2017)
59. Zafeiriou, S., Trigeorgis, G., Chrysos, G., Deng, J., Shen, J.: The menpo facial landmark localisation challenge: A step towards the solution. In: Computer Vision and Pattern Recognition Workshop (2017)
60. Zhang, K., Zhang, Z., Li, Z., Qiao, Y.: Joint face detection and alignment using multitask cascaded convolutional networks. IEEE Signal Processing Letters **23**(10), 1499–1503 (2016)
61. Zhang, Z., Luo, P., Loy, C.C., Tang, X.: Facial landmark detection by deep multi-task learning. In: ECCV (2014)
62. Zhao, G., Huang, X., Taini, M., Li, S.Z., Pietikäinen, M.: Facial expression recognition from near-infrared videos. Image and Vision Computing **29**(9), 607–619 (2011)
63. Zhao, X., Liang, X., Liu, L., Li, T., Han, Y., Vasconcelos, N., Yan, S.: Peak-piloted deep network for facial expression recognition. In: ECCV (2016)