Weakly-supervised Deep Convolutional Neural Network Learning for Facial Action Unit Intensity Estimation

Yong Zhang[1,2], Weiming Dong[1], Bao-Gang Hu[1], and Qiang Ji[3]*
[1]National Laboratory of Pattern Recognition, CASIA
[2]University of Chinese Academy of Sciences
[3]Rensselaer Polytechnic Institute
zhangyong201303@gmail.com, weiming.dong@ia.ac.cn, hubg@nlpr.ia.ac.cn, qji@ecse.rpi.edu

Abstract

Facial action unit (AU) intensity estimation plays an important role in affective computing and human-computer interaction. Recent works have introduced deep neural networks for AU intensity estimation, but they require a large amount of intensity annotations. AU annotation needs strong domain expertise and it is expensive to construct a large database to learn deep models. We propose a novel knowledge-based semi-supervised deep convolutional neural network for AU intensity estimation with extremely limited AU annotations. Only the intensity annotations of peak and valley frames in training sequences are needed. To provide additional supervision for model learning, we exploit naturally existing constraints on AUs, including relative appearance similarity, temporal intensity ordering, facial symmetry, and contrastive appearance difference. Experimental evaluations are performed on two public benchmark databases. With around 2% *of intensity annotations in FERA 2015 and around* 1% *in DISFA for training, our method can achieve comparable or even better performance than the state-of-the-art methods which use* 100% *of intensity annotations in the training set.*

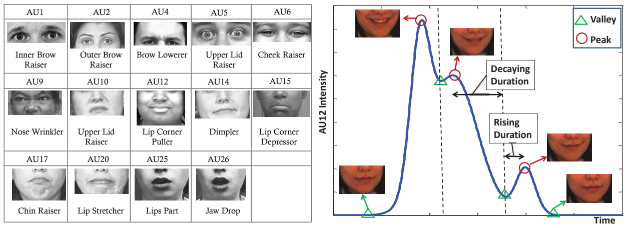
1. Introduction

Expressions are conveyed through facial appearance which is produced by the movements of facial muscles under the skin. Facial Action Coding System (FACS) was developed by Ekman and Friesen [5] to depict these muscle movements. It defines AUs as a contraction or relaxation of one or a group of muscles (see Fig. 1a). Nearly any anatomically possible facial expression can be coded by a combination of AUs. FACS also divides AU intensity into 6 discrete ordinal levels and provides rules to annotate the intensity from neutral to maximum, i.e., *Neutral (0)* $<$ *Trace (A)* $<$ *Slight (B)* $<$ *Pronounced (C)* $<$ *Extreme (D)* $<$ *Maximum (E)*. The goal of AU intensity estimation is to predict the AU intensity for an unseen image.

Recently, deep neural networks (DNNs) have achieved breakthroughs in a variety of computer vision tasks, including image classification [14], image segmentation [18], action recognition [37], etc. The millions of parameters are learned through the end-to-end strategy with raw images as the input and targets as the output. DNNs have also been applied to facial behavior analysis, including expression recognition [13, 4], AU recognition [52, 16, 39], and AU intensity estimation [6, 42]. Since DNNs contain a huge number of parameters, a large set of training samples are required for model learning to avoid overfitting. However, AU annotation needs strong domain expertise and is time-consuming. Hence, it requires great effort to construct a large database with AU intensity annotations to meet the requirement of fully supervised DNNs.

Instead of requiring a large set of annotations, we propose a knowledge-based semi-supervised deep convolutional neural network (CNN) for AU intensity estimation. Prior knowledge can be used to facilitate the model learning and reduce the dependence on data [9, 8, 26]. In emotional sequences, only the AU intensity annotations of peak and valley frames are needed. Since peak and valley frames account for a rather small proportion of whole sequences, using semi-supervised learning can save great effort for intensity annotation. To leverage unlabeled frames, we exploit four types of domain knowledge on AU intensity to provide additional supervision for model learning, including relative appearance similarity, temporal intensity ordering, facial symmetry, and contrastive appearance difference. They come from the observation on AU intensity in emotional sequences. Fig. 1b shows the annotated peak and valley frames of AU12 in a sequence. Firstly, local appearance of AU changes smoothly since the physical movements of muscles are smooth. AU intensity gradually increases from

*Corresponding author.



(a) (b)

Figure 1. (a) Facial appearance of facial action units. (b) The intensity curve of AU12 in a sequence from [20].

a valley frame to its neighbor peak frame (*rising duration*) and it gradually decreases from a peak frame to its neighbor valley frame (*decaying duration*). The learned deep network is encouraged to provide intensity predictions that retain such relations. Secondly, during both rising and decaying durations, the closer two frames are, the more similar their appearance looks. The learned feature representation is encouraged to retain such property. Thirdly, human face is symmetric. For spontaneous expression, the facial appearance is always nearly the same on the left and right faces. So do the AU occurrence and intensity. The left and right faces should have close feature representations. Though there exist different head poses in face images, the learned representation is encouraged to be invariant to head pose. Fourthly, the facial appearance of an emotional face is different from a neutral face. The learned representation is encouraged to be able to tell apart emotion faces from neutral faces. The domain knowledge provides weak supervision and makes it feasible to exploit the unlabeled frames.

Our main contributions are summarized as follows. Firstly, we propose a knowledge-based semi-supervised deep convolutional neural network for AU intensity estimation. Only annotations of peak and valley frames in sequences are needed for model learning, which significantly reduces the effort for intensity annotation. Secondly, we identify four types of domain knowledge to provide weak supervision for model learning, including relative appearance similarity, temporal intensity ordering, facial symmetry, and contrastive appearance difference. The knowledge builds connections between labeled and unlabeled samples. Thirdly, we propose to use *5-element tuples* for model learning instead of individual frames or frame pairs, which leverages high-order relationships among multiple frames. Fourthly, we evaluate the proposed method on two public benchmark databases.

2. Related Work

Shallow models for AU intensity estimation. Several frame-based methods treat each frame independently without considering relationships among frames. For single AU intensity estimation, SVMs were used in [19, 41] and SVRs were used in [34]. Kaltwang *et al.* proposed to use Relevance Vector Regression for AU intensity prediction in [10] and [12]. To consider relationships among multiple AUs, Sandbach *et al.* [33] used the tree structured Markov Random Field to capture relationships among intensities of AUs. Walecki *et al.* proposed to jointly estimate the intensities of multiple AUs by using copula functions in [43] and [44]. Kaltwang *et al.* [11] used a latent tree model and learned the structure from the input features and labels. Temporal dynamics have been used to model sequential data in different vision tasks [23, 48, 3, 24, 47, 46, 50], which can also be used for AU intensity estimation. Probabilistic graphical models are used to capture temporal and spatial relationships among the AU intensities for joint estimation such as [21, 17, 28, 29, 1].

Several works focus on learning model with limited annotations. Ruiz *et al.* [30] proposed Multi-instance Dynamic Ordinal Random Fields for AU intensity estimation by exploiting the idea of the multi-instance learning to treat each sequence as a bag. Zhao *et al.* [53] formulated expression intensity estimation as a weakly supervised learning problem by combining the unsupervised ordinal regression and SVR. They consider only pairwise relationships while we consider relationships among multiple frames and more types of knowledge. One issue of these methods is that they have to extract hand-craft features first and then perform the model learning. The dynamics are applied only for AU intensity. Unlike them, we apply the dynamics to both AU intensity and image representation. Furthermore, The complexity of shallow models is limited. They are unable to handle high-dimensional image features and millions of training samples.

Deep models for AU intensity estimation. Recently, few works applied DNNs for AU intensity estimation. Gudi *et al.* [6] exploited a 7-layer CNN for both AU intensity estimation and AU detection. Walecki *et al.* [42] combined conditional random field (CRF) and CNN. CRF is used to capture the dependencies among the intensities of multiple AUs. CRF is parameterized by using copula functions to allow non-linear AU intensity relations while CNN is used to learn deep representation. They are learned simultaneously. Tran *et al.* [40] combined variational auto-encoder (VAE) and non-parametric ordinal Gaussian Process (OGPs) for joint learning of latent representations and classifiers of multiple ordinal outputs. Zhou *et al.* [54] and Batista *et al.* [2] applied DNN to estimate AU intensity under multiple head poses. However, these methods use fully supervised DNNs which require a large set of training samples. They tend to overfit the training set when the number of annotated samples is limited. On the contrary, our method not only uses the limited annotations, but also exploits four types of domain knowledge to take advantage of unlabeled samples to avoid overfitting.

Semi-supervised deep neural networks. Several works use the semi-supervised training paradigm to learn deep models. Lee *et al.* [15] used the predicted label by the current model as the pseudo-labels for unlabeled samples. Then, labeled and unlabeled samples are used to train the model. Mehdi *et al.* [31] exploited an unsupervised regularization term to force the classifiers prediction for multiple classes to be mutually exclusive and to effectively guide the location of the decision boundary. They then proposed to add regularization term with stochastic transformations and perturbations for semi-supervised learning [32]. Haeusser *et al.* [7] used the associations between label and unlabeled samples through two-step walking to learn neural networks. [32, 45, 51] introduced an auto-encoder to an existing network to learn efficient representations. Rasmus *et al.* [27] combined supervised learning with unsupervised learning by minimizing the sum of supervised cost and denoising cost. These methods are originally proposed for image classification, rather than AU intensity estimation. Unlike them, we use unlabeled samples through domain knowledge on AU intensity rather than adding perturbations or using reconstruction cost of auto-encoder. The domain knowledge contains the relationships among labeled and unlabeled frames on image representation and AU intensity.

3. Proposed Method

The pipeline of the proposed method is shown in Fig. 2a. We first present the four types of domain knowledge in Sec. 3.1 and then explain the training tuples and encode the knowledge in Sec. 3.2. We present the structure of CNN in Sec. 3.3 and the learning and inference in Sec. 3.4.

As shown in Fig. 1b, given the annotated peak and valley frames, the sequences can be split into segments by using the locations of peak and valley frames. Segments can be divided into three groups according to the trend of AU intensity, *i.e.*, evolving from a valley frame to a peak frame, evolving from a peak frame to a valley frame, and keeping AU intensity unchanged. To make the trend consistent, we reverse the order of frames for the segments that evolve from a peak frame to a valley frame. Then, each training segment either evolves from a valley frame to a peak frame or keeps AU intensity unchanged.

Notation: The training set for one AU is denoted as $\mathcal{D} = \{\mathbf{X}_m, y_m^1, y_m^{N_m}\}_{m=1}^M$, where $\mathbf{X}_m = \{X_m^n\}_{n=1}^{N_m}$. X_m^n is the n-th frame of the m-th training segment, which represents a raw image. M is the number of training segments. N_m is the length of the m-th sequence. Only the first frame and the last frame in each segment are annotated with AU intensities. y_m^1 denotes the intensity of X_m^1 while $y_m^{N_m}$ denotes the intensity of $X_m^{N_m}$. Let Θ denote the parameters of CNN. Let $\tilde{y}_m^n = f(X_m^n; \Theta)$ denote the predicted intensity of X_m^n and f_m^n denote the extracted features of X_m^n, *i.e.*, the last fully connected layer of CNN. Let $d(a, b)$ denote the distance $d(a, b) = ||a - b||^2$. Given the partially annotated training set, our goal is to learn the parameters Θ. We train a CNN for each AU individually since the locations of peak and valley frames are different for different AUs.

3.1. Domain Knowledge

Relative appearance similarity Since facial appearance changes smoothly, in a segment, the closer two frames are, the more similar they look. We encourage image representations extracted from the CNN to satisfy that the closer two frames are, the closer their representations are, namely,

$$d(f_m^i, f_m^j) \leq d(f_m^i, f_m^k), 1 \leq i < j < k \leq N_m, \qquad (1)$$

where $d(f_m^i, f_m^j) = ||f_m^i - f_m^j||^2$. When i, j, and k contain the first or last frame, Eq. 1 builds the connections between labeled and unlabeled frames on image representation.

Temporal intensity ordering During a facial action, the facial appearance changes smoothly as the physical movements of muscles are smooth. So does the AU intensity. Neighboring frames have the similar facial appearance and AU intensity. Though there exist multiple peak and valley frames in an expression sequence, the whole sequence can be split into segments. AU intensity of each segment changes monotonically. After the rearrangement, AU intensity in each segment changes without decreasing along time. To leverage the intensity ordering to supervise model learning, we encourage the predictions of AU intensity in a segment to satisfy the following constraint, *i.e.*,

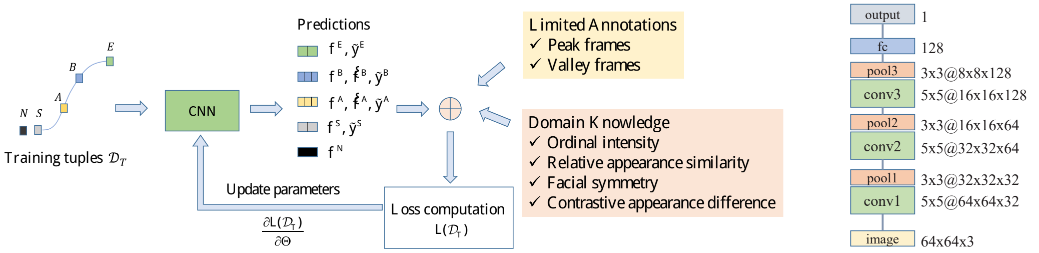
$$\tilde{y}_m^1 \leq \tilde{y}_m^2 \leq \cdots \leq \tilde{y}_m^{N_m}, m = 1, 2, \cdots, M. \qquad (2)$$

It contains the first and last labeled frames and also unlabeled frames between them, which builds the connections between labeled and unlabeled frames on AU intensity.

Facial symmetry Human face has the symmetry property. When performing an expression spontaneously, the appearance of the left face is always similar to the right. So do the AU occurrence and AU intensity. Though the captured faces might not be symmetric in image if they have a non-frontal head pose, it does not change the AU occurrence and AU intensity. When the corresponding regions of AUs are visible on both left and right faces, distinct patterns of AUs still appear on both sides. The appearance of these patterns is different due to head pose, but is similar. We encourage the CNN to provide head pose invariant representation for AU intensity estimation. For an aligned face by two eye centers, the horizontally flipped face should have the close representation to the original one, *i.e.*, the distance

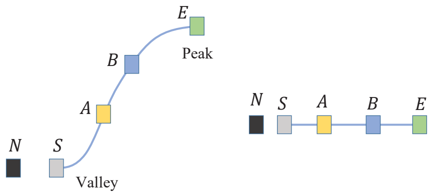
$$d(f_m^n, \hat{f}_m^n) = ||f_m^n - \hat{f}_m^n||^2, \qquad (3)$$

should be small. $\hat{f}_m^n$ is the representation of the flipped face.



(a) (b)

Figure 2. (a) The pipeline of the proposed method. Each frame in a tuple and also flipped face images of A and B go through the same CNN. The predictions are collected, including the representation from the fully collected layer and the predicted AU intensity. The loss is computed with using the predictions and the supervisory information, i.e., limited annotations and domain knowledge. The gradient of the loss is used to update the parameters of CNN. (b) The structure of CNN



(a) (b)

Figure 3. Training tuples. (a) $y^S < y^E$. (b) $y^S = y^E$.

Contrastive appearance difference For every subject, the appearance of an emotional face is different from a neutral face. We encourage the CNN to provide representations that can tell emotional faces apart from neutral faces. Since the neutral faces differ from subjects, we compute the distance between representations of emotional and neutral faces from the same subject. The distance should satisfy

$$d(f_m^n, f_m^N) = ||f_m^n - f_m^N||^2 \geq \eta, \tag{4}$$

where $\eta \geq 0$ is the threshold. f_m^N is the representation of a labeled neutral frame from the subject of the m-th segment.

3.2. Encoding knowledge

Training tuples Instead of directly using Eq. 2∼ 4 to form the objective, we propose to encode the knowledge based on training tuples (see Fig. 3). A tuple sampled from a segment is denoted as $T = \{S, A, B, E, N\}$ where $S < A < B < E$ and N are the frame indices. It consists of the first (S) and the last (E) frame of the segment, two frames (A and B) between S and E, and a neutral frame (N). The intensities of S, E, and N are labeled while A and B are unlabeled. Given a training segment, we can generate a large set of such tuples. Tuples from the same segment share the same S and E. The labeled neural frame N can be from other segments of the same subject. The advantages of using tuples are shown as follows. Given the training set $\mathcal{D}$, we can obtain the training tuples $\mathcal{D}_T = \{T_m^k\}_{m=1,k=1}^{m=M,k=K_m}$, where K_m is the number of tuples from the m-th segment.

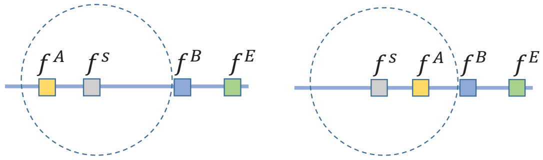
Encoding labels In each segment, only the first frame and the last frame are annotated. The intensity annotations provide strong supervisory information for model learning. They can not only supervise the model to give accurate intensity predictions on the two labeled frames during training, but also indirectly provide the upper and lower bounds of intensity for unlabeled frames through $\tilde{y}^S$ and $\tilde{y}^E$. Given a tuple T, the loss of provided labels is defined as

$$\ell_{lb}(T) = d(\tilde{y}^S, y^S) + d(\tilde{y}^E, y^E). \tag{5}$$

Encoding relative appearance similarity In a segment, instead of directly applying Eq. 1 to arbitrary three frames, we design the loss on a tuple to make full use of the annotations of the first and the last frames. The loss should have such properties: (a) captures the evolution of facial appearance in a segment, *i.e.*, from the first to the last frame, the representation gets more different from the first and gets more similar to the last, (b) makes full use of the annotated frames, and (c) considers high-order relationships among multiple frames. Given a tuple T, the loss is defined as

$$\begin{aligned}\ell_{rel}(T) = &\max(d(f^S, f^A) - d(f^S, f^B) + \alpha, 0)\\ &+ \max(d(f^E, f^B) - d(f^E, f^A) + \alpha, 0)\\ &+ \max(d(f^B, f^A) - d(f^B, f^S), 0)\\ &+ \max(d(f^A, f^B) - d(f^A, f^E), 0),\end{aligned} \tag{6}$$

where $\alpha \geq 0$ is the margin. Each term is a triplet loss [35] with respect to an anchor frame. Note that one triplet loss can not encode the correct evolution of facial appearance. As shown in Fig. 4, the two simplified cases show the distances between representations of frames. When S is the anchor frame of (S, A, B), the triple losses of both cases are 0, *i.e.*, $\max(d(\tilde{f}^S, \tilde{f}^A) - d(\tilde{f}^S, \tilde{f}^B) + \alpha, 0) = 0$, but only the second case (Fig. 4b) is the one that captures the right evolution of facial appearance. Hence, we introduce another term to ensure the right evolution, *i.e.*, $\max(d(\tilde{f}^B, \tilde{f}^A) - d(\tilde{f}^B, \tilde{f}^S), 0)$. To build more connections between the unlabeled frames A and B and labeled



(a) (b)

Figure 4. Simplified distances between representations of frames. When S is the anchor, the triplet losses of both (a) and (b) are 0, but only (b) is the correct one.

frames, we simultaneously consider S and E as two anchor frames. The loss encodes the relationships among four frames. We use different margins for two types of tuples (Fig. 3), *i.e.*, $\alpha = 0$ if $y^S = y^E$. Otherwise, $\alpha > 0$.

Encoding temporal intensity ordering A straightforward way of directly using Eq. 2 is to enforce frame pairs to satisfy the knowledge. Given a tuple T, the ordinal information is that the predictions are supposed to satisfy $\tilde{y}^S \leq \tilde{y}^A \leq \tilde{y}^B \leq \tilde{y}^E$. It can be encoded as

$$\ell = \max(\tilde{y}^S - \tilde{y}^A, 0) + \max(\tilde{y}^A - \tilde{y}^B, 0) + \max(\tilde{y}^B - \tilde{y}^E, 0). \tag{7}$$

However, the issue of using pairwise relationships is that it adjusts the parameters to make the local violated pair satisfy the constraint without considering other frames out of the pair. For example, when the predictions are $\tilde{y}^S \leq \tilde{y}^B \leq \tilde{y}^A \leq \tilde{y}^E$, only the second term in Eq. 7 is used to compute the gradient. The gradient involves only A and B, but the labeled S and E are not used.

To alleviate this issue, we consider the high-order relationships among multiple frames in the similar way to encoding relative appearance similarity. Instead of directly comparing intensity predictions, we use the distances between intensity predictions to encode the ordinal information. Given a tuple T, the loss of ordinal intensity is

$$\begin{aligned}\ell_{ord}(T) = &\max(d(\tilde{y}^S, \tilde{y}^A) - d(\tilde{y}^S, \tilde{y}^B) + \beta, 0)\\ &+ \max(d(\tilde{y}^E, \tilde{y}^B) - d(\tilde{y}^E, \tilde{y}^A) + \beta, 0)\\ &+ \max(d(\tilde{y}^B, \tilde{y}^A) - d(\tilde{y}^B, \tilde{y}^S), 0)\\ &+ \max(d(\tilde{y}^A, \tilde{y}^B) - d(\tilde{y}^A, \tilde{y}^E), 0),\end{aligned} \tag{8}$$

where $\beta \geq 0$ is the margin. When the predictions are $\tilde{y}^S \leq \tilde{y}^B \leq \tilde{y}^A \leq \tilde{y}^E$, at least the first and the second terms are used to compute the gradient. All the frames are jointly considered to update the parameters, including the labeled S and E. We use different margins for the two types of tuples, *i.e.*, $\beta = 0$ if $y^S = y^E$. Otherwise, $\beta > 0$.

Encoding facial symmetry Facial symmetry involves one face image and its horizontally flipped image. Given a tuple T, Eq. 3 can be directly used to define the loss of facial symmetry, *i.e.*,

$$\ell_{sym}(T) = d(f^A, \hat{f}^A) + d(f^B, \hat{f}^B). \tag{9}$$

We compute loss for only A and B because S and E are the same for tuples from the same segment. A and B can be very close to S and E in some tuples and they can cover the similar information from S and E. Since S could be a neutral frame of some tuples, N is also not included.

Encoding contrastive appearance difference Contrastive appearance difference involves one emotional face image and one neutral image. Given a tuple T, the loss is

$$\begin{aligned}\ell_{con}(T) = &\max(\eta - d(f^A, f^N), 0)\\ &+ \max(\eta - d(f^B, f^N), 0),\end{aligned} \tag{10}$$

where $\eta \geq 0$ is the threshold. If $y^S = y^E = 0$, $\eta = 0$; otherwise, $\eta > 0$. We consider the loss only for A and B for the same reason as facial symmetry.

3.3. CNN structure

We use a CNN with 3 convolutional layers, 3 max pooling layers, and 1 fully connected layer. The CNN structure is shown is Fig. 2b. The image has the size of 64×64 and 3 channels. The number of kernels is 32, 64, and 128 for the convolutional layers respectively. All convolution kernels have the same size of 5×5. Each convolutional layer is followed by a ReLu (Rectified Linear Unit) activation layer. All pooling kernels have the size of 3×3 and their stride is 2. The fully connected layer has 128 nodes. The dimension of the output is 1 since we train a CNN for each AU individually. The total number of parameters is $1,307,457$.

3.4. Learning and inference

Learning. Given training tuples $\mathcal{D}_T$, the goal is to learn the parameters Θ that minimize the objective $L(\mathcal{D}_T)$. As shown in Fig. 2a, each frame of a tuple is fed to the CNN. The predictions including the AU intensity and the representation are collected. Since only using the limited annotations leads to overfitting, we introduce domain knowledge to provide additional supervision. With the definitions of losses for the knowledge, the total loss of a training tuple can be computed as

$$\begin{aligned}\ell(T) = &\ell_{lb}(T) + \lambda_1 \ell_{rel}(T) + \lambda_2 \ell_{ord}(T)\\ &+ \lambda_3 \ell_{sym}(T) + \lambda_4 \ell_{con}(T)\end{aligned} \tag{11}$$

where $\lambda_1, ..., \lambda_4$ are the weights for losses. The total loss of all training tuples can be computed as

$$L(\mathcal{D}_T) = \frac{1}{G} \sum_{m=1}^{M} \sum_{k=1}^{K_m} \ell(T_m^k), \tag{12}$$

where $G = \sum_{m=1}^{M} K_m$. In Eq. 11, the loss is computed by using the predictions and the supervision from two sources, *i.e.*, limited annotations and the domain knowledge. The gradient of the loss with respect to Θ is

$$\frac{\partial \ell(T)}{\partial \Theta} = \sum_{z \in \mathbf{V}} \left[\frac{\partial \ell_{lb}}{\partial z} + \lambda_1 \frac{\partial \ell_{rel}}{\partial z} + \lambda_2 \frac{\partial \ell_{ord}}{\partial z} + \lambda_3 \frac{\partial \ell_{sym}}{\partial z} + \lambda_4 \frac{\partial \ell_{con}}{\partial z} \right] \frac{\partial z}{\partial \Theta}, \quad (13)$$

where $\mathbf{V} = \{f^S, f^A, f^B, f^E, f^N, \tilde{f}^A, \tilde{f}^B, \tilde{y}^S, \tilde{y}^A, \tilde{y}^B, \tilde{y}^E\}$. Each loss term is a function of Θ. We first compute the gradients with respect to elements in $\mathbf{V}$ and then update the CNN parameters through backpropogation. The detailed gradients are presented in the supplementary material.

Inference. Though the CNN is trained by using tuples, it can be used to predict AU intensity for a single image. Given a testing image, we feed it to the CNN and the predicted AU intensity is $y = f(X; \Theta)$. For evaluation, we discretize the continuous prediction into discrete intensity label, *i.e.*, 0 ($y < 0.5$), 1 ($0.5 \leq y < 1.5$), 2 ($1.5 \leq y < 2.5$), 3 ($2.5 \leq y < 3.5$), 4 ($3.5 \leq y < 4.5$), and 5 ($4.5 \leq y$). We also report the performance of using continuous prediction for evaluation in the supplementary material.

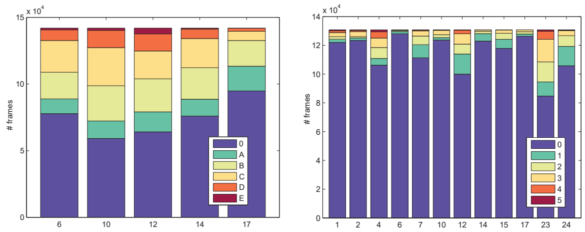
4. Experiments

4.1. Settings

Data. The BinghamtonPittsburgh 4D database [49] is a spontaneous expression database, which is used as a benchmark in **FERA 2015** challenge [41]. It consists of 328 sequences from 41 subjects. Around $140,000$ frames are annotated with AU intensity for 5 AUs. The official Training split of FERA 2015 contains 21 subjects while the Development split contains the other 20 subjects. In our experiments, we use the offical Training/Development splits.

The Denver Intensity of Spontaneous Facial Action (**DISFA**) [22] database is a spontaneous expression database, which consists of 27 sequences from 27 subjects when watching videos. Around $130,000$ frames are annotated with AU intensity for 12 AUs. In our experiments, we perform 3-fold subject independent cross validation with 18 subjects for training and 9 subjects for testing.

The AU intensity is qualified into 6 discrete levels. The distributions of AU intensities in FERA 2015 and DISFA are shown in Fig. 5a and Fig. 5b. The number of peak and valley frames of each AU in **the training set** is shown in Table 1. Note that the percentage of peak and valley frames is around 2% in FERA 2015 while the percentage is around 1% in DISFA. Our method uses only the annotations of peak and valley frames and unlabeled frames for learning.



(a) FERA 2015 (b) DISFA

Figure 5. Distribution of AU intensities.

Data preprocessing. Data preprocessing includes cropping face, image normalization, and data augmentation. We first register face images according to two eye centers. The eye centers are obtained by using facial landmarks provided in each database. Then, we crop the face and resize it to the size of 64×64. We perform per-image contrast normalization to alleviate the influence of illumination changes. Since the model has about 1.3 million parameters, to avoid overfitting via data augmentation, we randomly shrink or enlarge face images to $90\% \sim 110\%$ of its original size and crop the center part with the size of 64×64.

Tuple generation. Given peak and valley frames which can be identified according to their definitions in [20], training sequences can be split into segments. We sample training tuples from each segment. As shown in Fig. 3, each tuple consists of 5 frames. Given a segment, the first and last frames are S and E. We select two frames between S and E and treat them as A and B. Then, we select an annotated neutral frame as N. A subset of valley frames are neutral frames. As both databases have a high frame rate, close frames have similar appearance. We collect one frame every 5 frames in the segment and then use each frame pair from the collected frames as A and B to form a tuple.

Evaluation metrics. For evaluation, we use Intra-Class Correlation (ICC(3,1) [36]) and Mean Absolute Error (MAE) as the measures. The hyperparameters are $\{\alpha, \beta, \eta, \{\lambda_i\}_{i=1}^{i=5}\}$. To tune them, we use 70% of the training subjects for training and 30% for evaluation. MAE is used as measure and the grid search strategy is used. For the margins, $\alpha \in \{0.1, 0.5, 1\}$ and $\beta \in \{0.01, 0.05, 0.1\}$ if $y^S \neq y^E$. Otherwise, $\alpha = 0$ and $\beta = 0$. If $y^S = y^E = 0$, $\eta = 0$. Otherwise, $\eta \in \{0.1, 0.5, 1\}$. For penalty factors, $\{\lambda_i\}_{i=1}^{i=5} \in \{0.01, 0.1, 1\}$. The learning rate is 0.0002.

Models. (i) We compare our method (**KBSS**) to the baseline methods. To verify the effectiveness of each type of knowledge, we learn the model without using one type of knowledge, including removing relative appearance similarity (**KBSS-NR**), removing temporal intensity ordering (**KBSS-NO**), removing facial symmetry (**KBSS-NS**), and removing contrastive appearance difference (**KBSS-NC**). We also compare to **CNN-K** which uses only the knowledge without intensity annotations, and compare to **CNN-P** which uses only limited annotations of peak and valley frames, and **CNN-F** which uses annotations of all frames. We then

Table 1. The number of peak and valley frames in the training set. The total numbers of training frames in FERA and DISFA are **74906** and **87209** respectively. 'Percentage' represents the percentage of peak and valley frames in the training set.

Dataset	FERA 2015					DISFA											
AU	6	10	12	14	17	1	2	4	5	6	9	12	15	17	20	25	26
Peak&Valley	1527	1563	1636	1811	2830	871	769	1031	776	841	711	1072	745	1069	734	1129	1165
Percentage	2.04%	2.09%	2.18%	2.42%	3.78%	1.00%	0.88%	1.18%	0.89%	0.96%	0.82%	1.23%	0.85%	1.23%	0.84%	1.29%	1.34%

Table 2. Comparison to the baseline methods. Numbers in bracket and bold indicate the best performance; numbers in bold indicate the second best.

		FERA 2015						DISFA												
	AU	6	10	12	14	17	avr.	1	2	4	5	6	9	12	15	17	20	25	26	avr.
ICC(3,1)	CNN-F	[**.76**]	.69	.83	.29	[**.53**]	.62	.02	.06	.43	.02	[**.56**]	.22	.75	.13	.18	[**.09**]	.79	.28	.29
	CNN-P	.72	.47	.79	.18	.29	.49	.05	.02	.11	.01	.51	.09	.72	.00	.15	.00	.71	.28	.22
	CNN-K	.39	.18	.38	.12	-.09	.20	.00	.00	-.05	.00	-.10	.05	.24	-.01	.01	.01	-.05	-.07	.00
	KBSS-Pair	.73	**.73**	**.84**	.40	.43	.62	.15	.07	**.50**	.21	.47	.20	**.76**	[**.25**]	.19	.03	.77	.33	.33
	KBSS-Tri	.69	.68	**.84**	.40	.48	.62	.08	.09	.41	.25	.45	.23	.72	[**.25**]	.23	.05	.82	.32	.33
	KBSS-NO	.70	.69	**.84**	.32	.49	.61	.15	.09	.43	.23	.51	.19	.69	.14	.14	.04	.81	[**.41**]	.32
	KBSS-NR	.75	.65	.82	.38	.50	.62	.11	.07	.40	**.25**	.44	**.26**	[**.78**]	.18	.23	[**.09**]	.82	.19	.32
	KBSS-NS	.73	.72	**.84**	.44	.48	.64	.11	.08	[**.54**]	**.25**	.48	[**.29**]	.70	.20	[**.26**]	.07	[**.83**]	.29	**.34**
	KBSS-NC	.73	.72	.82	**.45**	**.52**	**.65**	**.16**	**.10**	.32	[**.28**]	**.54**	.22	.70	.23	.19	.01	[**.83**]	.40	.33
	KBSS	[**.76**]	[**.75**]	[**.85**]	[**.49**]	.51	[**.67**]	[**.23**]	[**.11**]	.48	**.25**	.50	.25	.71	.22	**.25**	.06	[**.83**]	[**.41**]	[**.36**]
MAE	CNN-F	.69	.71	.51	1.02	.70	.73	.57	.36	.67	.09	.27	.31	.31	.17	.46	**.18**	.50	.58	.37
	CNN-P	.62	.78	.57	**.96**	.70	.73	**.47**	**.26**	.95	[**.07**]	.30	.21	.32	[**.09**]	**.23**	[**.07**]	.65	.51	**.34**
	CNN-K	.62	.69	.51	1.06	.72	.72	[**.38**]	[**.15**]	.89	.17	.39	[**.15**]	.41	.66	[**.22**]	.34	.91	.71	.45
	KBSS-Pair	.65	.75	.51	1.12	.90	.79	.78	.53	.69	**.07**	.28	.21	**.30**	**.13**	.53	.21	.56	.51	.40
	KBSS-Tri	.62	.76	.52	**.96**	[**.63**]	**.70**	.55	.51	.76	.16	.30	.24	.34	.18	.38	.18	.48	**.43**	.38
	KBSS-NO	.64	.70	.53	1.03	.69	.72	.78	.65	.95	.12	.31	.32	.46	.33	.82	.68	.47	.55	.54
	KBSS-NR	**.61**	**.68**	.51	[**.93**]	.76	**.70**	1.01	.84	1.20	.11	.39	**.19**	[**.27**]	.21	.44	.33	.47	.61	.51
	KBSS-NS	.62	.76	.49	1.04	.75	.73	.77	.60	**.65**	.10	.42	.20	.35	.21	.42	.29	.45	.44	.41
	KBSS-NC	.65	.79	[**.48**]	1.03	.66	.72	.86	.52	.90	[**.07**]	[**.26**]	.23	.37	.15	.64	.39	[**.38**]	.46	.44
	KBSS	[**.56**]	[**.65**]	[**.48**]	.98	[**.63**]	[**.66**]	.48	.49	[**.57**]	.08	[**.26**]	.22	.33	.15	.44	.22	**.43**	[**.36**]	[**.33**]

compare to **KBSS-Pair** and **KBSS-Tri**. KBSS-Pair uses pairwise relationships (Eq. 7) instead of high-order relationships (Eq .8). KBSS-Tri uses triplets instead of 5-element tuples. **(ii)** We compare our method to the state-of-the-art AU intensity estimation methods. **CNN** [6] was proposed for AU intensity estimation in FERA 2015. **CCNN-IT** [42] uses CRF to capture relationships between continuous variables for AU intensity estimation. In [42], CCNN-IT(*) combines multiple databases for training while CCNN-IT uses one database. For fair comparison, we compare to CCNN-IT. **2DC** [40] combines variational auto-encoder and Gaussian Process for AU intensity estimation. **OR-CNN** [25] transforms an ordinal regression problem to a series of binary classification sub-problems for age estimation, which can also be used for AU intensity estimation. We also adapt **VGG16** [38] for AU intensity estimation by fine-tuning the last 3 layers of the pre-trained model. **(iii)** We compare our method to the state-of-the-art semi-supervised methods, including **Ladder** [27] , **RSTP** [32], and **LBA** [7]. Though they are originally proposed for image recognition, we apply them to AU intensity estimation.

4.2. Results

Our method vs. baseline methods The results are shown in Table 2. A visual example of our method is shown in Fig. 6. Results are analyzed as follows. Firstly, compared to methods with dropping one type of knowledge, KBSS achieves better performance. It demonstrates that each type of knowledge contributes. The results show that relative appearance similarity and temporal intensity ordering are more important than the other two types of knowledge. Secondly, compared to CNN-P, our method is much better. CNN-P uses only peak and valley frames, which tends to overfit these samples. It generalizes poorly to the testing samples. Besides labeled frames, our method also uses unlabeled frames through the knowledge, which ensures its generalization ability. Thirdly, CNN-K that uses only the knowledge achieves poor performance. This shows the importance of the limited annotations. Fourthly, compared to CNN-F, surprisingly, our method even outperforms it in both ICC and MAE though CNN-F uses much more annotations. We attribute the improvement to the usage of the knowledge on AU intensity and the relationships among multiple frames. We explicitly use the knowledge as supervision with considering relationships among multiple

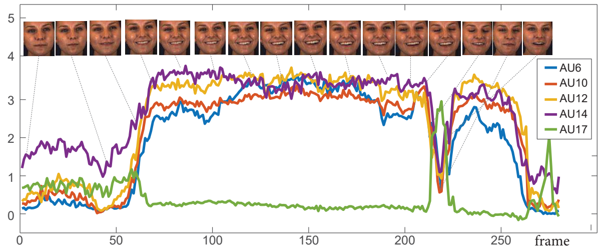


Figure 6. Prediction of each AU in a sequence of FERA 2015

Table 3. Comparison to the state-of-the-art AU intensity estimation methods. Only our method is a semi-supervised method.

		FERA 2015						DISFA												
	AU	6	10	12	14	17	avr.	1	2	4	5	6	9	12	15	17	20	25	26	avr.
ICC(3,1)	CCNN-IT [42]*	.75	.69	[**.86**]	.40	.45	.63	.20	.12	.46	.08	.48	**.44**	.73	**.29**	[**.45**]	[**.21**]	.60	.46	**.38**
	2DC [40]*	[**.76**]	**.71**	**.85**	**.45**	[**.53**]	**.66**	[**.70**]	[**.55**]	[**.69**]	.05	[**.59**]	[**.57**]	[**.88**]	[**.32**]	.10	.08	[**.90**]	**.50**	[**.50**]
	CNN [6]	.72	.64	.82	.22	**.52**	.58	.07	.03	.39	.11	.49	.30	**.76**	.20	.20	**.12**	.74	.41	.32
	VGG [38]	.68	.63	.75	.35	.37	.56	**.31**	**.29**	.40	**.13**	.39	.13	.58	.02	.16	.03	.63	.22	.27
	OR-CNN [25]	.74	.70	**.85**	.34	.51	.63	-.01	.02	.21	.10	.47	.30	**.76**	.14	.21	.07	**.84**	[**.59**]	.31
	KBSS (ours)	[**.76**]	[**.75**]	**.85**	[**.49**]	.51	[**.67**]	.23	.11	**.48**	[**.25**]	**.50**	.25	.71	.22	**.25**	.05	.82	.41	.36
MAE	CCNN-IT [42]*	1.17	1.43	.97	1.65	1.08	1.26	.73	.72	1.03	.21	.72	.51	.72	.43	.50	.44	1.16	.79	.66
	2DC [40]*	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
	CNN [6]	.66	.84	.56	1.12	.65	.77	**.40**	**.41**	**.56**	.10	**.27**	**.20**	**.30**	.13	[**.36**]	**.16**	.60	.45	**.33**
	VGG [38]	.63	.80	.66	[**.91**]	[**.61**]	.72	[**.24**]	[**.22**]	[**.51**]	[**.04**]	**.27**	[**.13**]	.37	[**.10**]	**.41**	.17	.63	.47	[**.30**]
	OR-CNN [25]	[**.56**]	**.72**	**.49**	**.95**	.69	**.68**	.48	.45	.95	[**.04**]	.28	.23	[**.27**]	**.12**	.47	[**.12**]	[**.40**]	[**.32**]	.34
	KBSS (ours)	[**.56**]	[**.65**]	[**.48**]	.98	**.63**	[**.66**]	.48	.49	.57	.08	[**.26**]	.22	.33	.15	.44	.22	**.43**	**.36**	**.33**

Table 4. Comparison to the state-of-the-art semi-supervised methods.

		FERA 2015						DISFA												
	AU	6	10	12	14	17	avr.	1	2	4	5	6	9	12	15	17	20	25	26	avr.
ICC(3,1)	Ladder [27]	.65	.63	.79	.24	.45	.55	-.01	.03	.16	.01	**.50**	.10	.64	-.01	.06	.00	.57	.22	.19
	RSTP [32]	.68	.63	.77	.24	.48	.56	.00	.05	.20	.05	.42	.11	.59	.09	.13	.05	.68	.38	.23
	LBA [7]	.71	.65	.80	.28	.50	.59	.04	.06	.39	.01	.41	.12	**.73**	.13	**.27**	**.10**	**.82**	**.43**	.29
	KBSS (ours)	**.76**	**.75**	**.85**	**.49**	**.51**	**.67**	**.23**	**.11**	**.48**	**.25**	**.50**	**.25**	.71	**.22**	.25	.05	**.82**	.41	**.36**
MAE	Ladder [27]	.72	.82	.62	1.15	.66	.79	.68	.39	.94	.14	**.26**	.29	.34	.17	**.26**	**.13**	.78	.52	.41
	RSTP [32]	.73	.93	.70	1.24	**.61**	.84	1.17	.80	1.23	.25	.34	.38	.42	.23	.66	.39	.59	.39	.57
	LBA [7]	.63	.79	.60	1.07	.62	.74	**.43**	**.29**	**.51**	.10	.30	**.19**	**.30**	**.11**	.31	.14	**.40**	.38	**.29**
	KBSS (ours)	**.56**	**.65**	**.48**	**.98**	.63	**.66**	.48	.49	.57	**.08**	**.26**	.22	.33	.15	.44	.22	.43	**.36**	.33

frames while CNN-F treats each frame independently. Finally, our method outperforms KBSS-Pair and KBSS-Tri. It further shows that the effectiveness of high-order relationships within tuples. The study of increasing annotations and detailed comparison to CNN-F are shown in the supplementary material.

Our method vs. AU intensity estimation methods The results are shown in Table 3. (*) means that the results of CCNN-IT and 2DC are adapted from [42] and [40]. As shown in Table 3, though using limited annotations, our semi-supervised method achieves comparable or even better performance than the state-of-the-art fully supervised methods. On FERA 2015, our method achieves the best performance in both ICC and MAE on average. On DISFA, our method outperforms CNN, VGG, and OR-CNN in ICC and achieves close performance to them in MAE. Compared to the reported performance of CCNN-IT, our method is close to CCNN-IT in ICC and much better in MAE. These supervised methods treat frames independently while our method consider relationships among multiple frames on both AU intensity and image representation. The results further demonstrate the effectiveness of the proposed semi-supervised method.

Our method vs. the semi-supervised methods. As shown in Table 4, our method achieves the best performance on FERA 2015. On DISFA, our method achieves the best performance in ICC and the second best in MAE. LBA tends to predict the intensity to be 0 which is the majority AU intensity (see Fig. 5b). It can achieves good MAE performance, but its ICC is much worse than ours. Ladder and RSTP treat each frame independently and use unlabeled samples by denoising or permutation. LBA uses unlabeled samples by walking from a labeled sample to a unlabeled one and then walking back. Differently, our method considers high-order relationships among multiple frames on both intensity and representation. The results show the effectiveness of the domain knowledge incorporated in our model.

5. Conclusion

We propose a knowledge-based semi-supervised deep CNN for AU intensity estimation. The proposed method requires only the annotations of peak and valley frames in training sequences, which significantly reduces the requirement of annotations for training CNN. We identify four types of knowledge and encode them to provide additional supervision. Particularly, the designed losses for relative appearance similarity and temporal intensity ordering consider high-order relationships among multiple frames in training tuples. Evaluations on FERA 2015 and DISFA demonstrate that though using around 1% or 2% of intensity annotations in the training set, our method can achieve comparable or even better to the state-of-the-art methods that use 100% of intensity annotations.

Acknowledgments: The work was accomplished when the first author was visiting Rensselaer Polytechnic Institute (RPI), through a scholarship from China Scholarship Council. The support of CSC and RPI is greatly appreciated. This work was also supported in part by the National Natural Science Foundation of China (NSFC) under Grant Nos. 61573348, 61702488, and the Beijing Natural Science Foundation under Grant No. 4162056.

References

[1] T. Baltrušaitis, P. Robinson, and L.-P. Morency. Continuous conditional neural fields for structured regression. In *ECCV*, 2014. 2

[2] J. C. Batista, V. Albiero, O. R. Bellon, and L. Silva. Aumpnet: simultaneous action units detection and intensity estimation on multipose facial images using a single convolutional neural network. In *FG*, 2017. 2

[3] J. Chen, S. Nie, and Q. Ji. Data-free prior model for upper body pose estimation and tracking. *TIP*, 2013. 2

[4] H. Ding, S. K. Zhou, and R. Chellappa. Facenet2expnet: Regularizing a deep face recognition net for expression recognition. In *FG*, 2017. 1

[5] P. Ekman and W. V. Friesen. *Manual for the facial action coding system*. Consulting Psychologists Press, 1978. 1

[6] A. Gudi, H. E. Tasli, T. M. den Uyl, and A. Maroulis. Deep learning based facs action unit occurrence and intensity estimation. In *FG workshop*, 2015. 1, 2, 7, 8

[7] P. Haeusser, A. Mordvintsev, and D. Cremers. Learning by association-a versatile semi-supervised training method for neural networks. In *CVPR*, 2017. 3, 7, 8

[8] B.-G. Hu, H.-B. Qu, Y. Wang, and S.-H. Yang. A generalized-constraint neural network model: Associating partially known relationships for nonlinear regressions. *Information Sciences*, 2009. 1

[9] Q. Ji. Combining knowledge with data for efficient and generalizable visual learning. *PRL*, 2017. 1

[10] S. Kaltwang, O. Rudovic, and M. Pantic. Continuous pain intensity estimation from facial expressions. In *ISVC*, 2012. 2

[11] S. Kaltwang, S. Todorovic, and M. Pantic. Latent trees for estimating intensity of facial action units. In *CVPR*, 2015. 2

[12] S. Kaltwang, S. Todorovic, and M. Pantic. Doubly sparse relevance vector machine for continuous facial behavior estimation. *TPAMI*, 2016. 2

[13] P. Khorrami, T. Paine, and T. Huang. Do deep neural networks learn facial action units when doing expression recognition? In *ICCVW*, 2015. 1

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012. 1

[15] D.-H. Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *ICML workshop*, 2013. 3

[16] W. Li, F. Abitahi, and Z. Zhu. Action unit detection with region adaptation, multi-labeling learning and optimal temporal fusing. In *CVPR*, 2017. 1

[17] Y. Li, S. M. Mavadati, M. H. Mahoor, Y. Zhao, and Q. Ji. Measuring the intensity of spontaneous facial action units with dynamic bayesian network. *PR*, 2015. 2

[18] Z. Liu, X. Li, P. Luo, C.-C. Loy, and X. Tang. Semantic image segmentation via deep parsing network. In *ICCV*, 2015. 1

[19] M. H. Mahoor, S. Cadavid, D. S. Messinger, and J. F. Cohn. A framework for automated measurement of the intensity of non-posed facial action units. In *CVPRW*, 2009. 2

[20] M. Mavadati, P. Sanger, and M. H. Mahoor. Extended disfa dataset: Investigating posed and spontaneous facial expressions. In *CVPRW*, 2016. 2, 6

[21] S. M. Mavadati and M. H. Mahoor. Temporal facial expression modeling for automated action unit intensity measurement. In *ICPR*, 2014. 2

[22] S. M. Mavadati, M. H. Mahoor, K. Bartlett, P. Trinh, and J. F. Cohn. Disfa: A spontaneous facial action intensity database. *IEEE Transactions on Affective Computing*, 2013. 6

[23] S. Nie and Q. Ji. Capturing global and local dynamics for human action recognition. In *ICPR*, 2014. 2

[24] S. Nie, Z. Wang, and Q. Ji. A generative restricted boltzmann machine based method for high-dimensional motion data modeling. *CVIU*, 136, 2015. 2

[25] Z. Niu, M. Zhou, L. Wang, X. Gao, and G. Hua. Ordinal regression with multiple output cnn for age estimation. In *CVPR*, 2016. 7, 8

[26] Y.-J. Qu and B.-G. Hu. Generalized constraint neural network regression model subject to linear priors. *TNN*, 2011. 1

[27] A. Rasmus, M. Berglund, M. Honkala, H. Valpola, and T. Raiko. Semi-supervised learning with ladder networks. In *NIPS*, 2015. 3, 7, 8

[28] O. Rudovic, V. Pavlovic, and M. Pantic. Multi-output laplacian dynamic ordinal regression for facial expression recognition and intensity estimation. In *CVPR*, 2012. 2

[29] O. Rudovic, V. Pavlovic, and M. Pantic. Context-sensitive dynamic ordinal regression for intensity estimation of facial action units. *TPAMI*, 2015. 2

[30] A. Ruiz, O. Rudovic, X. Binefa, and M. Pantic. Multi-instance dynamic ordinal random fields for weakly-supervised pain intensity estimation. In *ACCV*, 2016. 2

[31] M. Sajjadi, M. Javanmardi, and T. Tasdizen. Mutual exclusivity loss for semi-supervised deep learning. In *ICIP*, 2016. 3

[32] M. Sajjadi, M. Javanmardi, and T. Tasdizen. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. In *NIPS*, 2016. 3, 7, 8

[33] G. Sandbach, S. Zafeiriou, and M. Pantic. Markov random field structures for facial action unit intensity estimation. In *ICCVW*, 2013. 2

[34] A. Savran, B. Sankur, and M. T. Bilge. Regression-based intensity estimation of facial action units. *IVC*, 2012. 2

[35] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, 2015. 4

[36] P. E. Shrout and J. L. Fleiss. Intraclass correlations: uses in assessing rater reliability. *Psychological bulletin*, 1979. 6

[37] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. In *NIPS*, 2014. 1

[38] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 7, 8

[39] Z. Tősér, L. A. Jeni, A. Lőrincz, and J. F. Cohn. Deep learning for facial action unit detection under large head poses. In *ECCV Workshop*, 2016. 1

[40] D. L. Tran, R. Walecki, S. Eleftheriadis, B. Schuller, M. Pantic, et al. Deepcoder: Semi-parametric variational autoencoders for facial action unit intensity estimation. In *ICCV*, 2017. 2, 7, 8

[41] M. F. Valstar, T. Almaev, J. M. Girard, G. McKeown, M. Mehu, L. Yin, M. Pantic, and J. F. Cohn. Fera 2015-second facial expression recognition and analysis challenge. In *FG workshop*, 2015. 2, 6

[42] R. Walecki, V. Pavlovic, B. Schuller, M. Pantic, et al. Deep structured learning for facial action unit intensity estimation. In *CVPR*, 2017. 1, 2, 7, 8

[43] R. Walecki, O. Rudovic, M. Pantic, and V. Pavlovic. Copula ordinal regression for joint estimation of facial action unit intensity. In *CVPR*, 2016. 2

[44] R. Walecki, O. Rudovic, M. Pantic, V. Pavlovic, and J. F. Cohn. A framework for joint estimation and guided annotation of facial action unit intensity. In *CVPRW*, 2016. 2

[45] J. Weston, F. Ratle, H. Mobahi, and R. Collobert. Deep learning via semi-supervised embedding. In *Neural Networks: Tricks of the Trade*, pages 639–655. Springer, 2012. 3

[46] B. Wu, B.-G. Hu, and Q. Ji. A coupled hidden markov random field model for simultaneous face clustering and tracking in videos. *Pattern Recognition*, 2017. 2

[47] B. Wu, S. Lyu, B.-G. Hu, and Q. Ji. Simultaneous clustering and tracklet linking for multi-face tracking in videos. In *ICCV*, 2013. 2

[48] B. Wu, Y. Zhang, B.-G. Hu, and Q. Ji. Constrained clustering and its application to face clustering in videos. In *CVPR*, 2013. 2

[49] X. Zhang, L. Yin, J. F. Cohn, S. Canavan, M. Reale, A. Horowitz, P. Liu, and J. M. Girard. Bp4d-spontaneous: a high-resolution spontaneous 3d dynamic facial expression database. *IVC*, 2014. 6

[50] Y. Zhang, Z. Tang, B. Wu, Q. Ji, and H. Lu. A coupled hidden conditional random field model for simultaneous face clustering and naming in videos. *TIP*, 2016. 2

[51] J. Zhao, M. Mathieu, R. Goroshin, and Y. LeCun. Stacked what-where auto-encoders. In *ICLR workshop*, 2016. 3

[52] K. Zhao, W.-S. Chu, and H. Zhang. Deep region and multi-label learning for facial action unit detection. In *CVPR*, 2016. 1

[53] R. Zhao, Q. Gan, S. Wang, and Q. Ji. Facial expression intensity estimation using ordinal information. In *CVPR*, 2016. 2

[54] Y. Zhou, J. Pi, and B. E. Shi. Pose-independent facial action unit intensity regression based on multi-task deep transfer learning. In *FG*, 2017. 2