Deep structured learning for facial action unit intensity estimation

R. Walecki, O. Rudovic, V. Pavlovic, Björn Schuller, M. Pantic

Angaben zur Veröffentlichung / Publication details:

Walecki, R., O. Rudovic, V. Pavlovic, Björn Schuller, and M. Pantic. 2017. "Deep structured learning for facial action unit intensity estimation." In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 21-26 July 2017, Honolulu, Hawaii*, 3405–14. Piscataway, NJ: IEEE. https://doi.org/10.1109/CVPR.2017.605.

2017 IEEE Conference on Computer Vision and Pattern Recognition

Deep Structured Learning for Facial Action Unit Intensity Estimation

Robert Walecki[1], Ognjen (Oggi) Rudovic[2],
Vladimir Pavlovic[3], Björn Schuller[1] and Maja Pantic[1,4]
[1] Department of Computing, Imperial College London, UK
[2] MIT Media Lab, Cambridge, USA
[3] Department of Computer Science, Rutgers University, USA
[4] EEMCS, University of Twente, The Netherlands

Abstract

We consider the task of automated estimation of facial expression intensity. This involves estimation of multiple output variables (facial action units — AUs) that are structurally dependent. Their structure arises from statistically induced co-occurrence patterns of AU intensity levels. Modeling this structure is critical for improving the estimation performance; however, this performance is bounded by the quality of the input features extracted from face images. The goal of this paper is to model these structures and estimate complex feature representations simultaneously by combining conditional random field (CRF) encoded AU dependencies with deep learning. To this end, we propose a novel Copula CNN deep learning approach for modeling multivariate ordinal variables. Our model accounts for ordinal structure in output variables and their non-linear dependencies via copula functions modeled as cliques of a CRF. These are jointly optimized with deep CNN feature encoding layers using a newly introduced balanced batch iterative training algorithm. We demonstrate the effectiveness of our approach on the task of AU intensity estimation on two benchmark datasets. We show that joint learning of the deep features and the target output structure results in significant performance gains compared to existing deep structured models for analysis of facial expressions.

1. Introduction

Automated analysis of human facial expressions aims to make inference about affective states, emotion expressions, pain levels, etc., from face images of the target person. Facial expressions are typically described in terms of configuration and intensity of facial muscle actions using the Facial Action Coding System (FACS) [11]. FACS defines a unique set of 30+ atomic non-overlapping facial muscle actions named Action Units (AUs) [27], with rules for scoring their intensity on a six-point ordinal scale. Using FACS, nearly any anatomically possible facial expression can be described as a combination of AUs and their intensities.

Recent advances in deep neural networks (DNN), and, in particular, convolutional models (CNNs) [14], have allowed to completely remove or highly reduce the dependence on physics-based models and/or other pre-processing techniques, by enabling the "end-to-end" learning in the pipeline directly from input images. While the effectiveness of these models has been demonstrated on many general vision problems [19, 40, 38], only baseline tasks such as expression recognition and AU detection [23, 44, 17] and AU intensity estimation [14] have been investigated. All of them, however, follow the traditional "blind deep learning" paradigm that relies on large labeled training datasets (e.g., 100K+ samples in [32]). Yet, in the facial data domain obtaining accurate and comprehensive labels is typically prohibitive. For instance, it takes more than an hour for an expert annotator to code AUs' intensity for 1 sec of face video. Even then, the annotations are highly biased and have low inter-annotator agreement. Coupled with large variability in imaging conditions, facial morphology, dynamics of expressions, this has resulted in the lack of suitable large datasets for effective deep model learning.

To improve deep learning for facial expression analysis and, in particular, intensity estimation of facial AUs, from available (annotated) facial images, we exploit and combine two modeling approaches: structured learning and data-sharing (e.g., between multiple datasets). We focus on the AU intensity estimation as the intensities are very difficult to annotate manually (high number of AUs and their intensity levels) but are of critical importance for high level interpretation of facial expressions. This, inevitably, entails a scarcity in available annotated data. Furthermore, the AU intensities are highly imbalanced due to the highest intensity levels occurring rarely and varying considerably among subjects. Finally, the dynamics of AUs also vary across contexts (e.g., in facial expressions of pain vs expressions of basic emotions).

To tackle these challenges, we first constrain the deep CNN models by imposing their structure at different levels. Specifically, we model the network output (i.e., different AUs) jointly as ordinal variables to account for the monotonicity constraints in the (discrete) intensity levels of each AU. Also, explicitly modeling the relations between co-occurrences of AU intensity levels has been addressed for binary outputs only (e.g., for object detection [22]), and not for multi-level intensities. In this work, we model the AU intensity relations by allowing them to be $non\text{-}linearly$ related – in contrast to present models that account only for linear dependencies. We do so by means of copula functions [4], known for their ability to capture highly non-linear dependencies through a simple parametrization. The notion of the copula functions has previously been explored for modeling of structured output [42] but not in the context of structured deep learning.

To efficiently model these two types of structure within our deep CNN model, we borrow the modeling approach of conditional graph models (Conditional Random Fields – CRFs) to define (**ordinal**) unary and (**copula**) binary cliques in the output graph (i.e., the output layer of the deep net), which are then learned jointly with the CNN layers. Note that several approaches to combining CNNs and CRFs have been proposed [35, 45, 6]. However, these model a different type of (spatial) dependencies, and, more importantly, deal only with (object) detection tasks- thus can not be directly scaled to the multi-class ordinal classification problems, as addressed here. Our main contributions can be summarized as follows:

- We propose a novel structured deep CNN-CRF model for joint learning of multiple ordinal outputs. The data structure is seamlessly embedded in the deep CNN via an output graph, capturing the ordinal structure in AU intensity levels via ordinal unary cliques, and non-linear dependencies between the network outputs via the copula binary cliques. We show that this model learns better the target AUs from scarce and highly imbalanced data compared to existing deep models.

- Joint learning of the deep CNN and target dependency structure (CRF) in our model is challenging and can easily lead to overfitting if standard learning is applied. To ameliorate this, we propose a novel approximate training: balanced-batch iterative training that carefully feeds the model with balanced variety of subjects, AU intensity levels and their co-occurrences during learning. We show that this is critical for the model's performance and leads to efficient learning.

- To leverage annotations from multiple datasets efficiently, our approach augments the learning of the shared marginals (AUs) across multiple datasets. This, in turn, results in models that are more robust to imbalanced and scarce data.

We show on benchmark datasets of naturalistic facial expressions, coded in terms of AU intensity, that our approach outperforms by a large margin related deep models applicable to the target task.

2. Related Work

2.1. Facial Action Unit Intensity Estimation

Estimation of AU intensity is often posed as a multi-class problem approached using Neural Networks [16], Adaboost [2], SVMs [26] and belief network classifeirs [24]. Yet, these methods are limited to a single output, thus, a separate classifier is learned for each AU – ignoring the AU dependencies. This has been addressed using the multi-output learning approaches. For example, [29] proposed a multi-task learning for AU detection where a metric with shared properties among multiple AUs was learned. Similarly, [34] proposed a MRF-tree-like model for joint intensity estimation of AUs. This method performs a two step learning – by first obtaining the intensity scores for each AU independently, followed by the MRF-graph optimization – aimed at capturing the AU relations. The proposed Latent-Trees (LTs) [15] for joint AU-intensity estimation capture higher-order dependencies among the input features and multiple target AU intensities. More recently, [42] proposed a multi-output Copula Ordinal Regression approach for estimation of AU intensity, where the co-occurring AU intensity levels are modeled using the statistical framework of copula functions. However, these methods are highly dependent on the feature pre-processing, involving (dense) facial point tracking and extraction of hand-crafted image features. More importantly, these cannot deal with high-dimensional input features. To this end, this paper investigates alternative approaches based on CNNs for the target task.

2.2. CNN Models for Facial Expression Analysis

So far, only a few works addressed the task of facial expression recognition using CNNs. [23] introduced an AU-aware receptive field layer in a deep network, designed to search subsets of the over-complete representation, each of which aims at best simulating the combination of AUs. Its output is then passed through additional layers aimed at the expression classification, showing a large improvement over the traditional hand-crafted image features such as LBPs, SIFT and Gabors. Another example is [14], where a CNN is jointly trained for detection and intensity estimation of multiple AUs. The authors proposed a network architecture composed of 3 convolutional and 1 max-pooling layers. More recently, [44] introduced an intermediate region layer that is able to learn region specific weights of

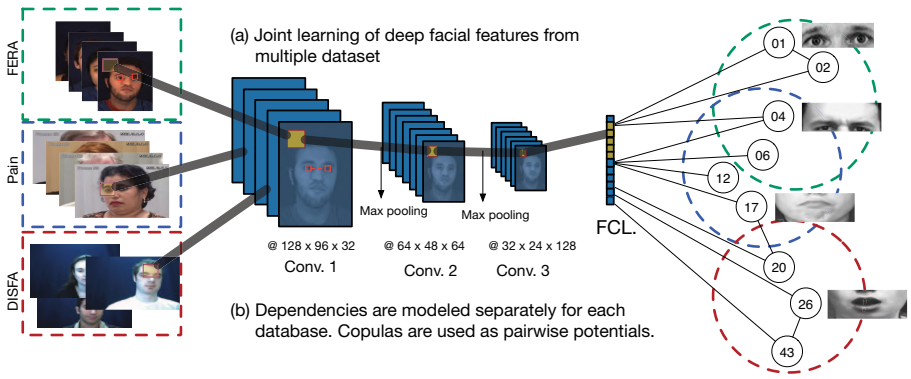


Figure 1: The proposed model pipeline. The input is a preprocessed face image, and the outputs are the model likelihood values for each intensity level of each AU. The CNN features are jointly learned for estimation of intensities of all AUs and the parameters of the unary potentials are shared. The pairwise potentials, however, model the AU dependencies that are specific to the context of the database.

CNNs. The region layer returns an importance map for each input image and the network is trained for joint AU detection. All these methods focus solely on feature extraction while the network output remains unstructured. By contrast, we capture the output structure by means of a CRF graph that explicitly accounts for ordinal and non-linear relations between multiple outputs. Note that ordinal modeling has been attempted in the context of deep networks for the age estimation task [30]. However, in contrast to our model, this method does not handle multi-output multi-class problems.

2.3. Structured Deep Models

Structured models allow us to learn task-specific constraints and relations between output variables directly from the data (see [31]). Recently, this has been a focus of research within deep learning – an attempt to regulate the network output. This is typically achieved by combining multi-output CNNs and graph models such as CRFs and MRFs. For instance, DeepLab-CRF [35] combines a CNN and fully connected CRF, where the binary cliques are used to model relations between image color and location. More recently, [22] proposed a fully connected CRF with *linear* binary cliques to capture semantic correlations between neighboring image patches, showing its effectiveness on the image segmentation task. Other applications of structured CNNs include image restoration [10], image super-resolution [8], depth-estimation [9], and image-tagging [5]. However, to the best of our knowledge, the deep structred learning has not been attempted before in context of facial expression analysis, and in particular, intensity estimation. Also, while the structured models mentioned above may be applicable to the target task, the key difference to our CCNN model is that they fail to model ordinal structure in their CRF model - which is critical when dealing with ordinal variables. Also, since these methods deal with binary outputs, they assume linear relations in the binary cliques of a CRF. This can easily be violated when dealing with multi-class outputs, as in our case. To this end, we propose non-linear dependence modeling using the framework of copula functions.

3. Structured Deep CRFs: Methodology

Fig.1 summarizes our deep structured learning approach. We assume here several settings. In the first setting, given an input face image, we first apply a pre-defined CNN network layer to the (normalized) input image, in order to generate a feature map. The learned deep features are of a lower resolution than the original image because of the down-sampling operations in the pooling layers. To embed the target structure, we place a CRF graph on the (fully) connected output layer of our network. Here, each output (AU) of the network represents a node in this graph, and relations between different nodes (AUs) are modeled using pairwise connections in this CRF. To leverage information from multiple datasets, we propose a data-augmented learning approach (the second setting). In this approach, the CNN layers are trained using data from multiple datasets simultaneously, resulting in enriched feature representation. As these datasets may contain non-overlapping sets of AUs, the model output will be a union of all these AUs, thus, instead of having multiple "weak" models, we arrive at a single shared model for multiple AUs. However, it is important to mention that for each combination of AUs (dataset-specific), we learn different dependencies in CRF pairwise connections, as their dynamics may vary considerably across the datasets. On the other hand, modeling of the marginals/nodes in the graphs is performed jointly, by sharing the model parameters of the overlapping AUs in these datasets.

For simplicity, we start with the notation that describes a

single dataset as $\mathcal{D} = \{\mathbf{Y}, \mathbf{X}\}$ (we extend this to multiple datasets in Sec.3.3). $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_i, \ldots, \mathbf{y}_N]^T$ is comprised of N instances of multivariate outputs stored in $\mathbf{y}_i = \{\mathbf{y}_i^1, \ldots \mathbf{y}_i^q, \ldots \mathbf{y}_i^Q\}$, where Q is the number of AUs, and $\mathbf{y}_i^q$ takes one of $\{1, ..., L^q\}$ discrete intensity levels of the q-th unary potential. Furthermore, $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_i, \ldots, \mathbf{x}_N]^T$ are input images that correspond to the combinations of labels in $\mathbf{Y}$.

Deep Facial Features: In our experiments, we first use a CNN to extract the feature map $f_d(\mathbf{x}, W)$ from an input image $\mathbf{x}$, where the network parameters are defined by W. We used 3 convolutional layers containing 32, 64 and 128 filters. The filter size was set to 9×9 pixel followed by ReLu (Rectified Linear Unit) activation functions. We also used max pooling layers with a filter size of 2×2 after each convolutinonal layer. The last component of the CNN is the fully connected layer (fcl) which returns 128 facial features. These parameters have been found via a validation procedure (Sec.4).

Structured CRFs: We assume a graph with unary and binary cliques in our CRF[20]. Specifically, we introduce a random field with an associated graph $\mathcal{G} = (V, \mathcal{C})$, where nodes $v \in V, |V| = Q$, correspond to individual AUs and cliques $c \in \mathcal{C}$ correspond to subsets of dependent AUs modeled using the copula functions. The conditional likelihood for image $\mathbf{x}$ having the labels y is then defined as follows:

$$P(\mathbf{y}|\mathbf{x}, \Omega) = \frac{1}{Z(\mathbf{x})} \exp\big[-E(\mathbf{y}, \mathbf{x}, \Omega)\big]. \quad (1)$$

Here, $Z(\mathbf{x}) = \sum_{\mathbf{y}^*} \exp\big[-E(\mathbf{y}^*, \mathbf{x}, \Omega)\big]$ is the partition function and the energy function is defined by a set of unary and pairwise potential functions.

$$E(\mathbf{y}, \mathbf{x}, \Omega) = \sum_{q \in V} U(\mathbf{y}^q, f_d, \phi^q) + \sum_{(r,s) \in E} V(\mathbf{y}^r, \mathbf{y}^s, f_d, \theta^{r,s}). \quad (2)$$

where U is the unary potential function and V the pairwise potential function. The parameters of U and V are ϕ and θ, respectively. The input features are computed using $f_d(\mathbf{x}, W)$ where x is the input and W the weights of the network.

3.1. Unary potentials

To impose increasing monotonicity constraints on the AU intensity levels, we formulate the unary potentials using the notion of ordinal regression [1]. Let $l \in \{1, \ldots, L\}$ be the ordinal label for the intensity level of the q-th AU. We employ the standard threshold model:

$$\mathbf{y}_*^q = \beta^q f_d(x, W)^T + \varepsilon^q, \mathbf{y}^q = l \text{ iff } \psi_{l-1}^q < \mathbf{y}_*^q \leq \psi_l^q. \quad (3)$$

where β^q is the ordinal projection vector, ψ_l^q is the lower bound threshold for count level l ($\psi_0^q = -\infty < \psi_1^q < \psi_2^q ... < \psi_{L-1}^q < \psi_L^q = +\infty$). By assuming that the error (noise) terms ε^q are Gaussian with zero mean and variance $(\sigma^q)^2$, their normal cumulative density function (cdf) is $F(z^q) = Pr(\varepsilon^q < z^q) = \int_{-\infty}^{z^q} \mathcal{N}(\xi; 0, 1) d\xi$, and the probability of AU q having intensity l is defined as:

$$\Pr(\mathbf{y}^q = l | f_d(x, W), \phi^q) = F(z_l^q) - F(z_{l-1}^q). \quad (4)$$

where $z_k^q = \frac{(\psi_k^q - \beta^q f_d(x,W)^T)}{\sigma^q}$. The model parameters are stored in $\phi^q = \{\psi_1^q, \psi_2^q, \ldots, \psi_{L-1}^q, \beta^q, \sigma^q\}$. Finally, the unary node potentials in our structured deep CRF are defined as:

$$U(\mathbf{y}^q, \mathbf{x}, W, \phi^q) = \Pr(\mathbf{y}^q = l | f_d(x, W), \phi^q). \quad (5)$$

Note that these ordinal potentials embed the label structure in our graph – this is in contrast to existing structured deep CRFs [35, 10, 8, 5], which typically use the softmax/sigmoid function.

3.2. Pairwise potentials

The structured deep CRFs reviewed in Sec.3 focus on modeling of binary co-occurrence patterns, and the use of linear binary potentials. Yet, in case of multiple intensity levels, various and highly non-linear co-occurrence patterns are expected (e.g., for two AUs, there are 6×6 possible configurations). To this end, we propose a more powerfull modeling of these dependencies using the copula functions[36].

The main idea of copulas is closely related to that of histogram equalization: for a random variable y^q with (continuous) cdf F, the random variable $u^q := F(y^q)$[1] is uniformly distributed on the interval $[0, 1]$. Using this property, the marginals can be separated from the dependency structure in a multivariate distribution [3]. In the context of structured learning, the copula functions allow us (i) to easily model non-linear dependencies among the outputs, and (ii) do so independently of their marginal models. The latter is particularly important when designing efficient learning algorithms for deep learning (see Sec.3.3).

Formally, a copula $C(u^1, u^2, \ldots u^Q)$: $[0, 1]^Q \rightarrow [0, 1]$ is a multivariate distribution function on the unit cube with uniform marginals [43]. When the random variables are discrete, as is the case with the AU intensity levels, we can construct the joint distribution for discrete variables as:

$$\begin{aligned} &\Pr(y^1 = l^1, \ldots, y^Q = l^Q) = \\ &\Pr(\psi_{l^1-1} < y_*^1 < \psi_{l^1}, \ldots, \psi_{l^Q-1} < y_*^Q < \psi_{l^Q}) \\ &= \sum_{c_1=0}^{1} \cdots \sum_{c_Q=0}^{1} (-1)^{c_1+\ldots+c_Q} F(z_{l^1-c_1}^1, ..., z_{l^Q-c_Q}^Q) \\ &= \sum_{c_1=0}^{1} \cdots \sum_{c_Q=0}^{1} (-1)^{c_1+\ldots+c_Q} C_\theta(u_{l^1-c_1}^1, ..., u_{l^Q-c_Q}^Q). \end{aligned} \quad (6)$$

[1] Sometimes we omit dependence on $f_d(x|W)$ for notational simplicity.

where $u^q_{l^q-c_q} = F(z^q_{l^q-c_q}), c_q \in \{0,1\}$, is defined in Sec.3.1, and θ are the copula parameters, as defined below. It is important to note that the the joint density model induced by the copula is conditioned on the deep features $f_d(x|W)$, i.e., $F(y^1,\ldots,y^Q) \leftarrow F(y^1,\ldots,y^Q|f_d(x|W))$. This, in contrast to the models in [34, 21] that rely solely on the AU labels, allows the deep features to directly influence the dependence structure of AUs, and the other way round, during learning. Under this formulation, for the binary case, the model reduces to:

$$\Pr(y^r = l^r, y^s = l^s) = F(z^r_{l_r}, z^s_{l_s}) + F(z^r_{l_r-1}, z^s_{l_s-1}) - F(z^r_{l_r-1}, z^s_{l_s}) - F(z^r_{l_r}, z^s_{l_s-1}). \tag{7}$$

We use these joint probabilities to define binary cliques in our CRF model as:

$$V(\mathbf{y}^r, \mathbf{y}^s, \mathbf{x}, W, \theta^{r,s}) = \Pr(y^1 = l^1, y^2 = l^2|f_d(x|W), \theta^{r,s}). \tag{8}$$

One specific benefit of copulas is that they can model different forms of (non-linear) dependency using simple parametric models for $C(\cdot)$. We limit our consideration to the commonly used Frank copula [12] from the class of Archimedean copulas, defined as:

$$C_\theta(u^r, u^s) = -\frac{1}{\theta}\ln\left(1 + \frac{(\exp(-\theta u^r) - 1)(\exp(-\theta u^s) - 1)}{\exp(-\theta) - 1}\right). \tag{9}$$

The dependence parameter $\theta \in (-\infty, +\infty)\backslash\{0\}$, and the perfect positive/negative dependence is obtained if $\theta \rightarrow \pm\infty$. When $\theta \rightarrow 0$, we recover the ordinal model in Eq.4.

3.3. Learning and Inference

Optimizing the network parameters can be done a in naive way by minimizing the (regularized) negative log-likelihood of Eq. (1). However, this is prohibitively expensive as it involves computation of the normalization constant Z, which, in case of 10 AUs, would involve 2^{10} evaluations of the copula functions. We mitigate this by resorting to the approximate methods based on piece-wise training of CRFs [22, 39], that allows us to define a composite likelihood function (instead of fully normalized pdf in Eq.(1)):

$$P(\mathbf{Y}|\mathbf{X}) = \prod_{q\in V} P(\mathbf{y}^q|\mathbf{x}) \prod_{r,s\in E} P(\mathbf{y}^r, \mathbf{y}^s|\mathbf{x}). \tag{10}$$

We include $l2$ regularization on the Unary potential. Finally, the overall cost is then given by:

$$\min_{\Omega} \lambda||\phi||_2^2 - \sum_i^N \left[\sum_{q\in V} P_q(\mathbf{y}^q_i|\mathbf{x}_i) + \sum_{r,s\in E} P_{rs}(\mathbf{y}^r_i, \mathbf{y}^s_i|\mathbf{x}_i)\right]. \tag{11}$$

Where λ defines the strength of the regularized. However, as we show empirically, minimizing the negative log-likelihood of Eq.(11) using all training data leads easily to model overfitting and, thus, poor performance. This is also due to the inherent hierarchical structure of our model (deep layers, CRF marginals and edge dependencies).

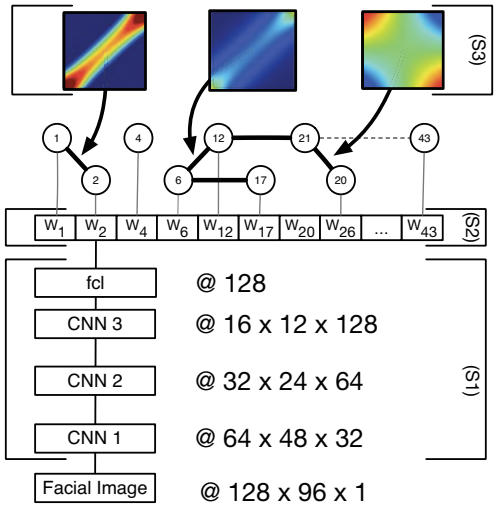


Figure 2: Three step parameter learning of the CCNN model. The input to the network is a facial image and the global negative log likelihood is optimized in an iteratively manner. First, the wights of the CNN are optimized in step 1 (S1). In the second step (S2), we optimize the parameter of the unary potentials and finally, in step 3 (S3), we optimize the parameter of the pairwise potentials. The frank copula density function is shown for a strongly correlated pair of AUs (1&2), for a weakly correlated pair of AUs (6&12) and for a negatively correlated pair (21&20).

Iterative Balanced Batch Learning. To tackle the challenges mentioned above, we introduce an iterative balanced batch (IBB) learning approach to deal with the data imbalance during optimization of our deep structured CRF. This imbalance is highly pronounced in the number of images per training subject, average number of examples per intensity level, as well as number of different label combinations, adversely affecting the learning of the CNN weights (W) and the unary (ϕ) and pairwise (θ) potential parameters, respectively. The main idea behind our IBB is to update each set of parameters with batches that are most representative of the target structure and, more importantly, balanced for that structure. To this end, when optimizing CNN weights, we generate batches (bb_n) that are balanced with respect to subjects in the dataset. This ensures that the learned network is not biased toward a specific subject. We adopt the same approach when creating batches for learning the marginals (balanced AU levels – bb_m) and copula parameters (balanced AU co-occurrences – bb_e). The learning algorithm for our network is shown in Alg.1. We optimize different areas of the network in each step of the al-

gorithm. We also compute the batches in each iteration of the algorithm by sampling from the target distribution function. We apply the three step optimization iteratively, where we update the parameters of one network region and fix the remaining parameters. All updates are made with respect to the global objective defined in Eq.(12) and we tune the validation parameter λ_r separately for each AU. Finally, we used Stochastic GradientDescent (SGD) with a batch size of 128, learning rate of 0.001 and momentum of 0.9.

Input: Training data: $\mathcal{D} = \{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^{N}$
Model parameters: $\Omega = \{W, \phi, \theta, U, V\}$

while *Eq.11 not converged* **do**

Step 1: *train W with balanced batches:*
$W \leftarrow \underset{W}{\text{argmax}} \sum_i^{N(bb_n)} P(\mathbf{y}_i|\mathbf{x}_i, W), i \in bb_n$

Step 2: *train ϕ with balanced batches:*
$\forall q \in U : \phi^q \leftarrow \underset{\phi^q}{\text{argmax}} \sum_i^{N(bb_m)} \Pr(\mathbf{y}_\text{i}^\text{q}|\mathbf{x}_\text{i}, \phi^\text{q}) + \lambda^\text{q}||\phi^\text{q}||_2^2 , i \in bb_m$

Step 3: *train θ with balanced batches:*
$\forall (rs) \in V : \theta^{rs} \leftarrow \underset{\theta^{rs}}{\text{argmax}} \sum_i^{N(bb_e)} \Pr(\mathbf{y}_\text{i}^\text{r}, \mathbf{y}_\text{i}^\text{s}|\mathbf{x}_\text{i}, \theta^\text{rs}) , i \in bb_e$

end

Output: Model parameters: $\Omega^{opt} = \{W, \phi, \theta\}$

Algorithm 1: Structured CNN Learning with balanced batches

Augmented Learning from Multiple Datasets. As discussed in Sec.1, leveraging data from multiple datasets efficiently is expected to further improve the AU estimation performance. To achieve this, we assume we are given K datasets $\mathcal{D} \in \{D_1, D_2, \ldots, D_K\}$. We then generalize the objective function of our deep structured CRFs:

$$\begin{aligned} P(\mathbf{Y}|\mathbf{X}) &= P_{D_1} \cdot P_{D_2} \cdot \cdots \cdot P_{D_K} \\ &= \prod_{u \in \mathcal{D}} \prod_{q \in V_u} P(\mathbf{y}^q|x) \\ &\times \prod_{h \in \mathcal{D}} \prod_{r,s \in E_h} P_h(\mathbf{y}^r, \mathbf{y}^s|x). \end{aligned} \tag{12}$$

The key property of these sets is that they may have different AUs annotated, different dependency distributions but also contain overlapping AUs. To handle this in a principled manner, we account for the shared marginals $P(\mathbf{y}^q|x)$ – the common AUs, while preserving the context-specific AU dependencies – $P_v(\mathbf{y}^r, \mathbf{y}^s|x)$ – data-specific joints. This joint modeling is expected to result in (i) improved feature representations, and (ii) more robust learning of the (shared) marginals. To avoid bias due to the dataset order during optimization, we combined the balanced batches from all datasets, in the same manner as in the proposed IBB learning, resulting in $bb_c \in \{bb_c^1, bb_c^2, , ..., bb_c^K\}$, where $c \in \{n, m, e\}$.

3.3.1 Joint Inference

The resulting CRF graph is an undirected graphical model that can contain loops and its potentials are not submodular. The inference of test data in this model is in general an np-hard problem due to the need to evaluate all possible label configurations. Because of this, we resort to approximate decoders based on the message-passing and dual decomposition algorithms. Specifically, we employed the AD3 decomposition algorithm [7].

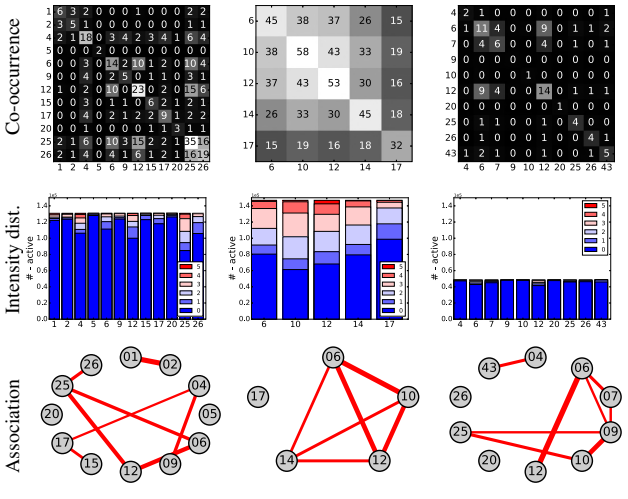
4. Experiments

Datasets. We evaluate the proposed model on two major benchmark datasets – DISFA [28] and on the subset of the BinghamtonPittsburgh 4D (FERA2015) [41]. These databases include acted and spontaneous expressions and vary in context eliciting facial expressions. The DISFA dataset contains video recordings of 27 subjects while watching YouTube videos. We performed the experiments in a subject independent setting (dividing data in training and test partion. For DISFA, we used 18 subjects for training and 9 for testing. In FERA2015 we used the official Training/Development splits. We also include the UNBC-McMaster Shoulder-Pain dataset for learning of the deep models[25][2]. In these datasets, each frame is coded in terms of the AU intensity on a six-point ordinal scale. We use the Intra-class Correlation ICC(3,1), which is commonly used in behavioral sciences to measure agreement between annotators (in our case, the AU intensity levels). We also report the Mean Absolute Error (MAE), commonly used for ordinal prediction tasks [18, 33].

Pre-Processing. To do the basic image normalization, we used the openCV eye detector [13] to extract the locations of the eyes from facial images in each dataset. We then registered the 2 facial points to a reference frame (average points in each dataset) using an affine transform. We then normalized each image using per-image contrast normalization, which increases the robustness against illumination changes. Lastly, we cropped a random bounding box to 85% of the original image size for data augmentation and robustness against displacements (as usually done in deep models).

Models: Baselines. We fist conducted experiments using standard CNN architectures employed in previous works ([14, 17]). The **CNN [14]** model is a standard CNN

[2]We use this dataset only to improve learning of our deep model; however, the evaluation results are in the supplementary materials.

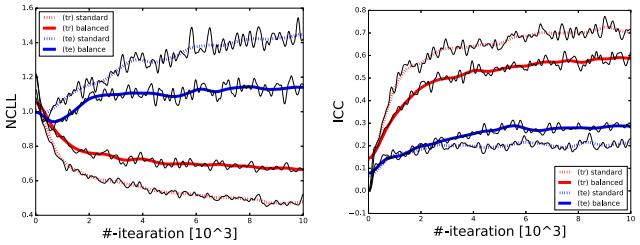


(a) DISFA (b) FERA (c) PAIN

Figure 3: Different types of dependences of AU intensity levels in the used datasets. The thickness of the edges corresponds to the magnitude of the θ parameter. Connections with a low association ($\theta > 0.05$) were removed.

with fully connected layer and softmax layer for multi-output classification. The **OCNN** is the same network but with an ordinal classification layer. This is a special case of the proposed CCNN where the pairwise potentials are ignored. For both methods, the weights are jointly learned and the predictions are computed independently.We conducted our experiments on a relative shallow CNN (see Fig. 2). We used 3 CNN-Layer for all our experiments but we performed cross validation to find the optimal filter size and number of channels per layer (see supplementary). Finally, the network parameters were optimized until the cost converged (see Fig.4).

Models: CNNs. The **R-CNN [44]** was introduced for AU detection tasks. It combines a basic CNN with a customized conditional layer that has region specific weights. This feature makes the model flexible and robust by allowing the weights to be different for background and face regions, for example. The **OR-CNN [30]** is another ordinal CNN. This network was introduced for the task of age estimation from images but can be readily applied to AU intensity estimation. **VGG16 [37]** is a widely used very deep CNN for object detection. In order to adapt it for out task, we used the pre-trained model and fine-tuned the last 3 layers for the task of AU intensity estimation. **SCNN [22]** is a structured CNN introduced for object detection. The linear pairwise potentials build a fully connected CRF, which is trained using piecewise [39] optimization. Since this model only performs binary detection, we extended it to multi-class classification by replacing its unary potentials with the



(a) Learning Curves (b) ICC per iteration

Figure 4: The plots show the learning curves of the CCNN model on the DISFA dataset when balanced batches is active and inactive.

ordinal classifier. Finally, the **CCNN** is the proposed copula conditional neural network where the pairwise potentials are defined by the copula function. The proposed model with the stepwise balanced batch optimization is named **CCNN-IT**, and with the data augmentation **CCNN-IT** (*). Likewise, the compared model is **SCNN-IT**.

4.1. Results

Ordinal vs. Softmax Unary Potentials. Table 1 shows the comparative results for the different models evaluated. On average, ordinal models largely outperform the output softmax models including related CNNs, across both measures on most of the AUs. This is particularly evident in the ICC scores, where the average difference is 7% on the DISFA database, and 3% on the FERA2015. We attribute this to modeling not just the different classes of intensity but also their ordinal relationship. **Independent vs. Structured CNNs.** Both, OCNN and R-CNN achieve an ICC of 29% on the DISFA dataset, which is the highest performance among the independent models. The CCNN model is equivalent to OCNN but with additional copula output for structured prediction. This results in average improvement of 4%. This is in particular visible on AU1&2, which are strongly correlated (see dependencies in Fig. 3). This correlation is modeled through the copula functions with high associations parameter. **The comparison with Related Deep Models.** OR-CNN [30] performs poorly in our experiments. This model learns one binary classifier for each AU intensity level, resulting in a large number of parameters and overfitting. We achieved better results with the standard VGG16 [37] network. However, also this model does not reach comparative results with our proposed model as it does not account for ordinal intensity levels. The same applies for the R-CNN [44]. While both models have a significant improvement over the standard CNN [14], they fail to accurately predict ordinal intensities.

Effect of batch balancing. Next, we observe that the average performance of the CCNN-IT model is another 3% higher than that achieved by the CCNN model (directly op-

Table 1: The intensity estimation results on the DISFA & FERA2015 datasets for different AUs. The best results are shown in bold and in brackets. The second best results are highlighted bold. We also highlight the methods where data augmentation with multiple datasets was used with a asterix.

		FERA2015						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 (*)	**[.75]**	**.69**	**[.86]**	**[.40]**	**[.45]**	**[.63]**	.18	**[.15]**	**[.61]**	.07	**[.65]**	**[.55]**	**[.82]**	**[.44]**	.37	**[.28]**	**[.77]**	**[.54]**	**[.45]**
	SCNN [22] (*)	.75	.67	**[.86]**	.39	.42	.62	.16	.12	.43	.06	**.62**	**.54**	**[.82]**	**.43**	.37	**[.28]**	**[.77]**	**.53**	**.43**
	CCNN-IT	**[.75]**	**.69**	**[.86]**	**[.40]**	**[.45]**	**[.63]**	**[.20]**	.12	**.46**	**[.08]**	.48	.44	.73	.29	**[.45]**	.21	.60	.46	.38
	OCNN-IT	**[.75]**	.68	**[.86]**	.40	.44	.62	**[.20]**	.07	**.46**	.08	.48	.41	.73	.29	.41	.21	.60	.44	.36
	CCNN	.74	.67	.85	**[.40]**	**[.45]**	.62	.14	.12	.37	**[.08]**	.46	.44	.64	.25	.37	.09	.58	.31	.32
	OCNN	.73	.63	.81	**[.40]**	.43	.60	.04	.05	.41	.01	.35	.19	.72	.23	**[.45]**	.06	.53	.44	.29
	CNN [14]	.67	**[.69]**	.77	.35	.33	.56	.05	.04	.36	.02	.44	.27	.67	.25	.08	.03	.46	.22	.23
	R-CNN [44]	.62	.64	.74	.31	.32	.52	.05	.06	.32	.02	.36	.39	.77	.29	.19	.04	.65	.35	.29
	VGG16 [37]	.63	.61	.73	.25	.31	.51	.19	**.14**	.19	.02	.39	.33	.68	.14	.27	.03	.59	.38	.28
	OR-CNN [30]	.60	.61	.59	.25	.31	.47	.03	.07	.01	.00	.29	.08	.67	.13	.27	.00	.59	.33	.20
MAE	CCNN-IT (*)	**[1.14]**	1.30	.99	1.65	**[1.08]**	**[1.23]**	.87	.63	**[.86]**	.26	.73	.57	**[.55]**	**[.38]**	.57	.45	**[.81]**	**[.64]**	**[.61]**
	SCNN [22] (*)	1.17	1.30	**[.97]**	**1.60**	1.18	**1.25**	.93	.84	1.05	**.17**	**[.71]**	.52	.59	**.39**	.51	.45	**[.81]**	.71	**.64**
	CCNN-IT	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
	OCNN-IT	**1.15**	**1.28**	1.05	1.62	1.19	1.26	.73	**.55**	**1.03**	.34	.72	.60	.72	.43	**[.47]**	**.45**	1.16	.70	.66
	CCNN	1.17	1.43	**[.97]**	1.65	**[1.08]**	1.26	**.69**	.72	1.19	.21	**.72**	**[.51]**	.74	.44	**.48**	.47	1.28	.73	.68
	OCNN	1.16	1.32	1.11	1.65	1.15	1.28	1.07	.82	1.16	.19	.87	.89	.72	.56	.50	.48	1.47	**.69**	.79
	CNN [14]	1.30	1.35	1.28	1.80	1.14	1.37	1.62	1.09	1.44	.23	.86	.71	.83	.50	.63	.47	1.71	.84	.91
	R-CNN [44]	1.37	**[1.25]**	1.13	**[1.59]**	1.16	1.30	.85	.70	1.07	.20	.75	.58	**.59**	.47	.57	.48	1.36	.77	.70
	VGG16 [37]	1.24	1.39	1.14	1.80	1.19	1.35	**[.68]**	**[.52]**	1.31	**[.16]**	.76	.59	.67	.43	.59	.47	1.33	.76	.69
	OR-CNN [30]	1.37	1.39	1.37	1.80	1.19	1.42	1.05	.87	1.47	.17	.79	.70	.69	.44	.59	.50	1.33	.86	.79

timized without the data balancing). The IBB learning that we applied in CCNN-IT yields a better performance on the DISFA dataset. Note that the highest improvement is made on DISFA and, in particular, on those AUs that occur infrequent (AU 1, 5, 17 and 20). As expected, we could not make this observation on the FERA database, since the labels there are relatively balanced.

Data Augmentation. Finally, we analyse the contribution of additional databases for training. To study this for FERA, we augment the training data with data from DISFA and PAIN. Similarly, for DISFA we augment the training data with FERA and PAIN. Results obtained in this setting are indicated with (*) in Tab. 1. With augmentation, the ICC for CCNN-IT increases by 7% on DISFA. The largest improvement is made on AU6 and AU12, as these AUs are shared among all datasets. There is also a significant improvement on some AUs shared with only one other database (e.g., AU 4, 9, 20, and 25 that are part of PAIN data but not FERA). Lastly, we noticed a strong increase of the ICC for AU15, present only in DISFA, and a decrease on AU17 common to DISFA and FERA. This behavior is somewhat counter-intuitive but could potentially be explained by different contexts of AU17 in the two datasets (dependent in DISFA, independent in FERA). Improvements in AU15 may be the result of refined but shared feature representation. To compare with SCNN [22], we use our implementation in which we minimize the same objective as in the CCNN-IT model but with linear pairwise potentials instead of the copula functions. By also applying our definition of the loss for multiple datasets, we make it possible to jointly train this model on multiple datasets and compare it with the CCNN-IT. We can see that, using copulas gives an improvement especially on the strongly correlated AU pairs (AU1-AU2 and AU6-AU12). This is expected, since the purpose of the Frank copula is to model pairwise correlations. On average, we obtain the highest performance with the CCNN-IT model where an ICC of 0.45% was reached.

5. Conclusion

We proposed a novel Copula CNN deep learning approach for joint estimation of facial intensity for multiple, dependent AUs. Specifically, we show that the "end-to-end" pipeline, coupled with key components for robust dependency modeling (copulas), balanced training (balanced-batch iterative), and data augmentation (across cross-context datasets) improves the performance achieved by existing structured deep models and models for estimation of facial expression intensity that fail to model non-linear dependencies in the output and also ordinal relations in the intensity levels.

Acknowledgements

This work has been funded by the European Community Horizon 2020 [H2020/2014-2020] under grant agreement no. 645094 (SEWA). and no. 688835 (DE-ENIGMA). The work of O. Rudovic is funded by European Union H2020, Marie Curie Action - Individual Fellowship (EngageMe 701236), and was funded earlier by the European Community Horizon 2020 [H2020/2014-2020] under grant agreement no. 688835 (DE-ENIGMA). The work of Vladimir Pavlovic is funded by the National Science Foundation under Grant no. IIS0916812.

References

[1] A. Agresti. Analysis of ordinal categorical data. *Wiley Series in Prob. and Stat.*, pages 1–287, 1984. 4

[2] M. S. Bartlett, G. C. Littlewort, M. G. Frank, C. Lainscsek, I. R. Fasel, and J. R. Movellan. Automatic recognition of facial actions in spontaneous expressions. *Journal of multimedia*, 1(6):22–35, 2006. 2

[3] P. Berkes, F. Wood, and J. W. Pillow. Characterizing neural dependencies with copula models. In *NIPS*, pages 129–136. 2009. 4

[4] J. Braeken, F. Tuerlinckx, and P. De Boeck. Copula functions for residual dependency. *Psychometrika*, pages 393–411, 2007. 2

[5] L.-C. Chen, A. G. Schwing, A. L. Yuille, and R. Urtasun. Learning deep structured models. In *ICML*, 2015. 3, 4

[6] J. Dai, K. He, and J. Sun. Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *CVPR*, pages 1635–1643, 2015. 2

[7] D. Das, A. F. Martins, and N. A. Smith. An exact dual decomposition algorithm for shallow semantic parsing with constraints. In *Proc. of the 1st Joint Conf. on Lexical and Computational Semantics-Volume 1*, pages 209–217. Association for Computational Linguistics, 2012. 6

[8] C. Dong, C. C. Loy, K. He, and X. Tang. Learning a deep convolutional network for image super-resolution. In *ECCV*, pages 184–199. Springer, 2014. 3, 4

[9] D. Eigen and R. Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *CVPR*, pages 2650–2658, 2015. 3

[10] D. Eigen, D. Krishnan, and R. Fergus. Restoring an image taken through a window covered with dirt or rain. In *CVPR*, pages 633–640, 2013. 3, 4

[11] P. Ekman, W. V. Friesen, and J. C. Hager. Facial action coding system. *Manual: A Human Face*, 2002. 1

[12] C. Genest. Frank's family of bivariate distributions. *Biometrika*, pages 549–555, 1987. 5

[13] A. Gorbenko and V. Popov. On face detection from compressed video streams. volume 6, pages 4763–4766, 2012. 6

[14] A. Gudi, H. E. Tasli, T. M. den Uyl, and A. Maroulis. Deep learning based facs action unit occurrence and intensity estimation. In *FG'W*, 2015. 1, 2, 6, 7, 8

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

[16] A. Kapoor and R. W. Picard. Multimodal affect recognition in learning environments. In *ACM*, pages 677–682. ACM, 2005. 2

[17] P. Khorrami, T. Paine, and T. Huang. Do deep neural networks learn facial action units when doing expression recognition? In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 19–27, 2015. 1, 6

[18] M. Kim and V. Pavlovic. Structured output ordinal regression for dynamic facial emotion intensity prediction. *ECCV*, pages 649–662, 2010. 6

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. 1

[20] J. D. Lafferty, A. McCallum, and F. C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*, pages 282–289, 2001. 4

[21] Y. Li, S. M. Mavadati, M. H. Mahoor, and Q. Ji. A unified probabilistic framework for measuring the intensity of spontaneous facial action units. In *FG*, pages 1–7, 2013. 5

[22] G. Lin, C. Shen, A. van den Hengel, and I. Reid. Efficient piecewise training of deep structured models for semantic segmentation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 2, 3, 5, 7, 8

[23] M. Liu, S. Li, S. Shan, and X. Chen. Au-aware deep networks for facial expression recognition. In *FG*, pages 1–6, 2013. 1, 2

[24] P. Liu, S. Han, Z. Meng, and Y. Tong. Facial expression recognition via a boosted deep belief network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1805–1812, 2014. 2

[25] P. Lucey, J. F. Cohn, K. M. Prkachin, P. E. Solomon, and I. Matthews. Painful data: The unbc-mcmaster shoulder pain expression archive database. In *FG*, pages 57–64, 2011. 6

[26] S. Lucey, A. B. Ashraf, and J. F. Cohn. *Investigating spontaneous facial action recognition through aam representations of the face*. INTECH Open Access Publisher, 2007. 2

[27] M. Mahoor, S. Cadavid, D. Messinger, and J. Cohn. A framework for automated measurement of the intensity of non-posed facial action units. *CVPR*, pages 74–80, 2009. 1

[28] S. M. Mavadati, M. H. Mahoor, K. Bartlett, P. Trinh, and J. F. Cohn. Disfa: A spontaneous facial action intensity database. *TAC*, pages 151–160, 2013. 6

[29] J. Nicolle, K. Bailly, and M. Chetouani. Facial action unit intensity prediction via hard multi-task metric learning for kernel regression. In *FG*, volume 6, pages 1–6. IEEE, 2015. 2

[30] Z. Niu, M. Zhou, L. Wang, X. Gao, and G. Hua. Ordinal regression with multiple output cnn for age estimation. In *CVPR*, June 2016. 3, 7, 8

[31] S. Nowozin and C. H. Lampert. Structured learning and prediction in computer vision. *Foundations and Trends in Computer Graphics and Vision*, 6(3–4):185–365, 2011. 3

[32] O. M. Parkhi, A. Vedaldi, and A. Zisserman. Deep face recognition. In *British Machine Vision Conference*, volume 1, page 6, 2015. 1

[33] O. Rudovic, V. Pavlovic, and M. Pantic. Context-sensitive dynamic ordinal regression for intensity estimation of facial action units. *TPAMI*, pages 944–958, 2014. 6

[34] G. Sandbach, S. Zafeiriou, and M. Pantic. Markov random field structures for facial action unit intensity estimation. In *ICCV*, 2013. 2, 5

[35] L. Shen, T. Wee Chua, and K. Leman. Shadow optimization from structured deep edge detection. In *CVPR*, pages 2067–2074, 2015. 2, 3, 4

[36] J. H. Shih and T. A. Louis. Inferences on the association parameter in copula models for bivariate survival data. *Biometrics*, pages 1384–1399, 1995. 4

[37] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014. 7, 8

[38] Y. Sun, X. Wang, and X. Tang. Deep learning face representation from predicting 10,000 classes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1891–1898, 2014. 1

[39] C. Sutton and A. McCallum. Piecewise pseudolikelihood for efficient training of conditional random fields. In *Proceedings of the 24th international conference on Machine learning*, pages 863–870. ACM, 2007. 5, 7

[40] C. Szegedy, A. Toshev, and D. Erhan. Deep neural networks for object detection. In *Advances in Neural Information Processing Systems*, pages 2553–2561, 2013. 1

[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 *Automatic Face and Gesture Recognition (FG), 2015 11th IEEE International Conference and Workshops on*, volume 6, pages 1–8, 2015. 6

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

[43] R. Winkelmann. *Econometric analysis of count data*. Springer Science & Business Media, 2003. 4

[44] K. Zhao, W.-S. Chu, and H. Zhang. Deep region and multi-label learning for facial action unit detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3391–3399, 2016. 1, 2, 7, 8

[45] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. H. Torr. Conditional random fields as recurrent neural networks. In *CVPR*, pages 1529–1537, 2015. 2