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

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

1. Gradient Computation

In the paper, we show the gradient of the objective in Section 3.4. We present more details as follows. The loss of a tuple is computed as

$$\ell(T) = \ell_{lb}(T) + \lambda_1\ell_{ord}(T) + \lambda_2\ell_{rel}(T) + \lambda_3\ell_{sym}(T) + \lambda_4\ell_{con}(T). \quad (1)$$

It is the summation of a set of basic terms, *i.e.*,

$$\ell_1 = \max(d(\mathbf{x},\mathbf{y}) - d(\mathbf{x},\mathbf{z}) + \alpha, 0), \quad (2)$$
$$\ell_2 = \max(\alpha - d(\mathbf{x},\mathbf{y}), 0), \quad (3)$$
$$\ell_3 = d(\mathbf{x},\mathbf{y}), \quad (4)$$

where $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{z}$ are vectors and α represents the margin. $d(\mathbf{x},\mathbf{y}) = ||\mathbf{x}-\mathbf{y}||^2$. ℓ_1 is the triplet loss, ℓ_2 is the max function, and ℓ_3 is the square loss.

Gradient of ℓ_1 The gradients of ℓ_1 with respect to $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{z}$ can be written as

$$\frac{\partial\ell_1}{\partial\mathbf{x}} = \begin{cases} -2(\mathbf{y}-\mathbf{z}), & \text{if } d(\mathbf{x},\mathbf{y}) - d(\mathbf{x},\mathbf{z}) + \alpha > 0 \\ 0, & \text{otherwise} \end{cases},$$

$$\frac{\partial\ell_1}{\partial\mathbf{y}} = \begin{cases} -2(\mathbf{x}-\mathbf{y}), & \text{if } d(\mathbf{x},\mathbf{y}) - d(\mathbf{x},\mathbf{z}) + \alpha > 0 \\ 0, & \text{otherwise} \end{cases},$$

$$\frac{\partial\ell_1}{\partial\mathbf{z}} = \begin{cases} 2(\mathbf{x}-\mathbf{z}), & \text{if } d(\mathbf{x},\mathbf{y}) - d(\mathbf{x},\mathbf{z}) + \alpha > 0 \\ 0, & \text{otherwise} \end{cases}.$$

Gradient of ℓ_2 The gradients of ℓ_2 with respect to $\mathbf{x}$ and $\mathbf{y}$ can be written as

$$\frac{\partial\ell_2}{\partial\mathbf{x}} = \begin{cases} -2(\mathbf{x}-\mathbf{y}), & \text{if } \alpha - d(\mathbf{x},\mathbf{y}) > 0 \\ 0, & \text{otherwise} \end{cases},$$

$$\frac{\partial\ell_2}{\partial\mathbf{y}} = \begin{cases} 2(\mathbf{x}-\mathbf{y}), & \text{if } \alpha - d(\mathbf{x},\mathbf{y}) > 0 \\ 0, & \text{otherwise} \end{cases},$$

Gradient of ℓ_3 The gradients of ℓ_3 with respect to $\mathbf{x}$ and $\mathbf{y}$ can be written as

$$\frac{\partial\ell_3}{\partial\mathbf{x}} = 2(\mathbf{x}-\mathbf{y}),$$

$$\frac{\partial\ell_3}{\partial\mathbf{y}} = 2(\mathbf{y}-\mathbf{x}).$$

2. Data Preparation

Our method is a semi-supervised method. It requires only the AU intensity annotations of peak and valley frames in training sequences. To verify the effectiveness of the proposed method, we need to evaluate on databases that provide only intensity annotations for peak and valley frames in the training set.

FERA 2015 [10] and DISFA [4] provide frame-level intensity annotations. To evaluate our method, we select the peak of valley frames according to the definitions of peaks and valleys in [3] and the provided annotations in the training set. We assume only annotations of peak and valley frames are given while annotations of other frames are unknown. After obtaining peak and valley frames, sequences can be split into segments. For each segment, we sample a set of training tuples for model learning.

3. Convergence

In Section 3.4 of the main script, we present the learning and inference of our model. Here, we present the convergence of our method. We use AU 12 as an illustration. The performance of AU12 at different iterations is shown in Fig. 2. As the iteration proceeds, the algorithm converges.

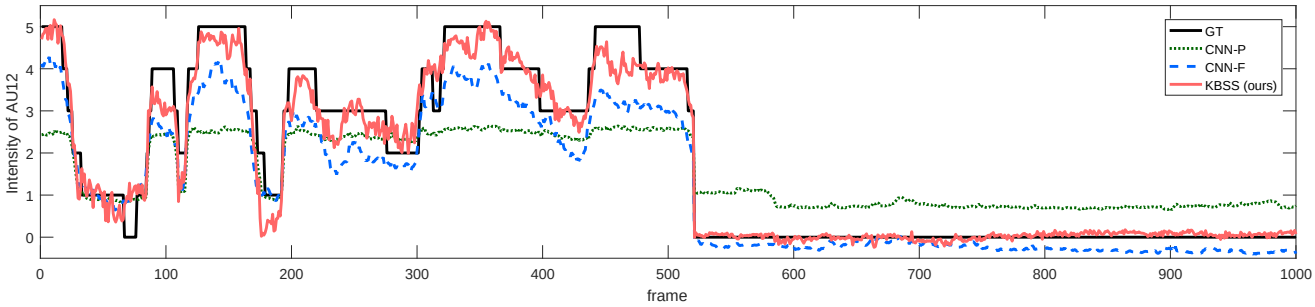

Figure 1. An illustration of the prediction of AU12 on a sequence.

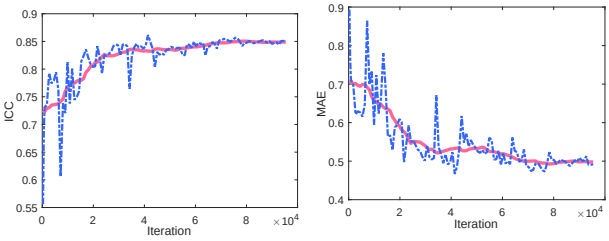

Figure 2. The performance of AU12 at different iterations. The blue line is the performance while the red line is the moving average.

We present the prediction of AU12 on a sequence in Fig. 1. We compare the performance of our method with CNN-P and CNN-F. CNN-P uses only annotated frames while CNN-F uses the annotations of all the frames. As shown in the figure, our method gives more accurate predictions than CNN-P and CNN-F.

4. Detailed Comparison with CNN-F

In Table 2 of the main script, our semi-supervised model outperforms CNN-F because CNN-F tends to overfit the training set. On FERA2015, the average training performance of CNN-F is (ICC:0.98, MAE:0.09) and the testing performance is (ICC:0.62, MAE:0.73). However, the training performance of our method is (ICC:0.87, MAE:0.40) and the testing performance is (ICC:0.67, MAE:0.66).

5. The Study of Increasing Annotations

In Section 4.2 of the main script, we compare with the baseline methods. Here, we study our method when increasing the number of AU intensity annotations. We perform an experiment on FERA 2015 by adding the intensity annotations of 20% of subjects each time. The results on FERA2015 are shown in Table 1. Before 60%, the performance gets better than without using additional annotations. Since the whole database is small, when using the annotations of more than 60% of subjects, the annotation dominates the learning and the model fits the training set better. However, the effect of the knowledge becomes weak and the model generalizes worse. The performance of our method when using all the annotations is better than CNN-F (Table 2 in the main script) because the knowledge helps improve the generalization ability to some extent.

Table 1. Performance under different annotation rates

subjects	0%	20%	40%	60%	80%	100%
ICC	0.67	0.67	0.68	0.66	0.65	0.65
MAE	0.66	0.63	0.64	0.68	0.68	0.70

6. Temporal Order on Features

In Section 3.1 of the main script, the assumption of temporal order on features comes from human heuristics. As shown in Fig. 3, A, B, C and D are four frames in a segment. Since the appearance of A is more similar to B than C, the features of A are encouraged to be more similar to B than C during learning. The right figure shows the loss of the temporal feature order on the testing set. The learned features satisfy the assumption more as the learning proceeds. Dropping such knowledge leads to the drop of performance as shown in Table 2 in the manuscript.

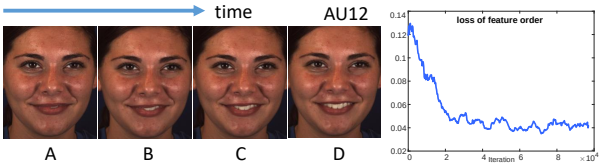

Figure 3. Left: appearance changes. Right: loss of feature order

7. Evaluation by Continuous Prediction

In the paper, we discretize the continuous prediction into discrete AU intensity for evaluation in Section 4.2. Here we also present the performance of different methods by using the continuous AU intensity prediction. Table 2 shows the comparison between our method and the baseline methods. Table 3 shows the comparison with the state-of-the-art AU intensity estimation methods. Note that only our method is a semi-supervised method while others are fully supervised methods. Table 4 shows the comparison with the state-of-the-art semi-supervised learning methods. We can draw the same conclusion from the evaluations with using the continuous prediction.

Table 2. Comparison to the baseline methods. Bracketed and bold numbers indicate the best performance; bold numbers 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	[**.78**]	.70	.84	.29	[**.55**]	.63	.02	.08	.44	.06	**.54**	.23	.76	.13	.18	[**.11**]	.80	.30	.30
	CNN-P	.73	.49	.81	.18	.23	.49	.07	.03	.12	.02	.53	.06	.73	.01	.05	.00	.71	.29	.22
	CNN-K	.27	.26	.31	.16	-.09	.18	.02	.05	.01	.06	-.30	.05	.40	.02	.05	.02	-.30	-.14	-.01
	KBSS-Pair	.74	**.74**	.83	.41	.45	.63	**.15**	.07	**.51**	.22	.48	.23	**.77**	[**.26**]	.23	.06	.78	.38	.34
	KBSS-Tri	.71	.69	.85	.40	.49	.63	.07	.10	.43	.27	.46	.25	.74	[**.26**]	.25	.06	.83	.34	.34
	KBSS-NO	.70	.70	.85	.39	.49	.62	.12	[**.12**]	.44	.25	.50	.22	.70	.13	.15	.03	.82	[**.43**]	.33
	KBSS-NR	.76	.67	.84	.39	.50	.63	.11	.04	.41	.25	.43	**.26**	[**.78**]	.16	.23	**.09**	.83	.19	.32
	KBSS-NS	.75	.73	[**.86**]	.42	.49	**.65**	.12	.09	[**.53**]	.26	.51	[**.28**]	.72	.21	[**.27**]	.07	[**.84**]	.30	**.35**
	KBSS-NC	.74	.72	.84	**.43**	.51	**.65**	**.15**	**.11**	.33	**.27**	[**.56**]	.24	.71	.20	.21	.02	.83	[**.43**]	.34
	KBSS	**.77**	[**.75**]	[**.86**]	[**.47**]	**.52**	[**.67**]	**.25**	**.11**	.48	[**.28**]	.52	.25	.72	.24	**.26**	.05	[**.84**]	.42	[**.37**]
MAE	CNN-F	**.64**	.80	.61	1.11	.76	.78	.68	**.48**	**.82**	.20	.41	.44	.41	.31	.56	.31	.60	.72	.50
	CNN-P	.70	1.01	.69	1.13	.70	.85	[**.51**]	[**.36**]	.99	.17	.40	.38	.43	.23	.46	**.25**	.73	.55	.45
	CNN-K	.98	1.39	1.16	[**.94**]	1.14	1.12	.58	.78	.97	.50	.53	.39	.67	.83	.44	.48	1.23	.81	.68
	KBSS-Pair	.70	.79	.59	1.16	.92	.83	.94	.63	**.77**	[**.11**]	.37	[**.23**]	.38	[**.19**]	.55	**.25**	.62	.47	.46
	KBSS-Tri	.67	.82	.60	**.98**	**.69**	.75	.58	.56	.78	.18	[**.31**]	**.26**	[**.36**]	.21	[**.40**]	[**.20**]	**.50**	**.44**	**.40**
	KBSS-NO	.74	.76	.56	1.11	.70	.77	1.05	.91	1.02	.18	.55	.42	.58	.43	.93	.77	.55	.63	.67
	KBSS-NR	.67	.82	.60	1.09	.80	.80	1.27	1.13	1.32	.24	.53	.28	.39	.36	.55	.46	.55	.69	.65
	KBSS-NS	.65	**.74**	[**.55**]	1.05	.72	**.74**	1.02	.79	.82	.20	.54	.30	.43	.30	.50	.41	.55	.51	.53
	KBSS-NC	.66	.78	.56	1.13	.78	.78	1.16	.68	1.02	.16	.34	.27	.50	.30	.73	.49	.52	.52	.56
	KBSS	[**.62**]	[**.72**]	[**.55**]	.99	[**.69**]	[**.71**]	**.53**	.54	[**.60**]	[**.11**]	[**.31**]	.28	[**.36**]	[**.19**]	.46	.27	[**.49**]	[**.40**]	[**.38**]

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 [11]*	.75	.69	[**.86**]	.40	.45	.63	.20	.12	.46	.08	.48	**.44**	.73	**.29**	[**.45**]	[**.21**]	.60	.46	[**.38**]
	2DC [9]*	**.76**	**.71**	.85	**.45**	[**.53**]	**.66**	[**.70**]	[**.55**]	[**.69**]	.05	[**.59**]	[**.57**]	[**.88**]	[**.32**]	.10	.08	[**.90**]	**.50**	[**.50**]
	CNN [1]	.74	.65	.83	.22	[**.53**]	.60	.06	.04	.38	**.16**	.48	.33	**.77**	.21	.20	**.12**	.76	.44	.33
	VGG [8]	.69	.64	.76	.35	.38	.56	**.32**	**.32**	.41	.15	.39	.14	.57	.05	.19	.04	.63	.24	.29
	OR-CNN [5]	.74	.70	.85	.34	.51	.63	-.01	.02	.21	.10	.47	.30	.76	.14	.21	.07	**.84**	[**.59**]	.31
	KBSS (ours)	[**.77**]	[**.75**]	[**.86**]	[**.47**]	.52	[**.67**]	.25	.11	**.48**	[**.28**]	**.52**	.25	.72	.24	**.26**	.05	**.84**	.42	.37
MAE	CCNN-IT [11]*	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 [9]*	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
	CNN [1]	.71	.87	.64	1.14	.72	.82	.53	.49	.67	.18	.35	.29	.39	.24	.47	.27	.67	.53	.42
	VGG [8]	.68	.84	.71	[**.94**]	[**.67**]	.77	[**.32**]	[**.28**]	[**.58**]	**.11**	.34	[**.20**]	.46	.20	[**.46**]	**.24**	.70	.53	**.37**
	OR-CNN [5]	[**.56**]	[**.72**]	[**.49**]	**.95**	**.69**	[**.68**]	**.48**	**.45**	.95	[**.04**]	[**.28**]	**.23**	[**.27**]	[**.12**]	.47	[**.12**]	[**.40**]	[**.32**]	[**.34**]
	KBSS (ours)	**.62**	[**.72**]	**.55**	.99	**.69**	**.71**	.53	.54	**.60**	**.11**	**.31**	.28	**.36**	**.19**	[**.46**]	.27	**.49**	**.40**	.38

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 [6]	.65	.63	.79	.24	.45	.55	-.01	.03	.16	.01	.50	.10	.64	-.01	.06	.00	.57	.22	.19
	RSTP [7]	.68	.63	.77	.24	.48	.56	-.00	.05	.20	.05	.42	.11	.58	.09	.13	.05	.68	.38	.23
	LBA [2]	.71	.65	.80	.28	.50	.59	.04	.06	.39	.01	.41	.12	**.73**	.13	**.27**	**.10**	.82	**.43**	.29
	KBSS (ours)	**.77**	**.75**	**.86**	**.47**	**.52**	**.67**	**.25**	**.11**	**.48**	**.28**	**.52**	**.25**	.72	**.24**	.26	.05	**.84**	.42	**.37**
MAE	Ladder [6]	.72	.82	.62	1.15	.66	.79	.68	.39	.94	.14	**.26**	.29	.34	.17	**.26**	**.13**	.78	.52	.41
	RSTP [7]	.73	.92	.70	1.24	**.61**	.84	1.17	.80	1.23	.25	.34	.38	.42	.23	.66	.39	.59	**.39**	.57
	LBA [2]	.63	.79	.60	1.07	.62	.74	**.43**	**.29**	**.51**	**.10**	.30	**.19**	**.30**	**.11**	.31	.14	**.40**	**.39**	**.29**
	KBSS (ours)	**.62**	**.72**	**.55**	**.99**	.69	**.71**	.53	.54	.60	.11	.31	.28	.36	.19	.46	.27	.49	.40	.38

References

[1] 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. 3

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

[3] M. Mavadati, P. Sanger, and M. H. Mahoor. Extended disfa dataset: Investigating posed and spontaneous facial expressions. In *CVPRW*, 2016. 1

[4] 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. 1

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

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

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

[8] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 3

[9] D. L. Tran, R. Walecki, S. Eleftheriadis, B. Schuller, M. Pantic, et al. Deepcoder: Semi-parametric variational autoencoders for facial action unit intensity estimation. *arXiv preprint arXiv:1704.02206*, 2017. 3

[10] 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. 1

[11] R. Walecki, V. Pavlovic, B. Schuller, M. Pantic, et al. Deep structured learning for facial action unit intensity estimation. 2017. 3