Temporal Stochastic Softmax for 3D CNNs – Supplementary Material –

Théo Ayral[1], Marco Pedersoli[1], Simon Bacon[2], and Eric Granger[1]
[1] LIVIA, Dept. of Systems Engineering, École de technologie supérieure, Montreal, Canada
[2] Dept. of Health, Kinesiology & Applied Physiology, Concordia University, Montreal, Canada
theo.ayral.1@ens.etsmtl.ca, {marco.pedersoli, eric.granger}@etsmtl.ca
simon.bacon@concordia.ca

In this document, we present a more detailed description for the implementation of stochastic softmax sampling. We also provide experiments with decoupled sampling and pooling temperatures and additional visualizations of distributions obtained during training with softmax sampling and REINFORCE. Finally we discuss complementary experiments and results.

S.1. Stochastic Softmax – Implementation

Clip Sampling. The sampler S, extracts a clip of F contiguous frames at temporal position t from a video x of arbitrary length L. The sampling mechanism can be formulated as:

$S : R^{L\times 3\times H\times W} \mapsto R^{F\times 3\times H\times W}$, with $F \leqslant L$, $H \times W$ is the spatial resolution of the data and we consider three colour channels. At every epoch of training, we construct batches of training clips. One clip is sampled from each training video. There are $N = L - F + 1$ possible clips to extract from a given dataset example. Videos are padded to contain at least F frames. With weighted training, the temporal sampling probability distribution of each video is computed from its classification scores. In the context of a deep-learning classifier, we consider that inference class scores represent a good measure of a clip's informativeness and relevance to the task [73, 75]. Specifically, for a given training clip, we use the score corresponding to the target label. In this sense, our method is similar to using the Oracle Sampler conceptualized in [68] at training time. This strategy minimizes the training loss by selecting the best scoring clips. The aim is to also improve the validation accuracy and reduce training time by learning from informative clips, without irrelevant and noisy frames. Let w_x be the temporal sequence of N classification scores estimates corresponding to the temporal responses of the classifier convolved over x. Then, $w_{x,t}$ is the classification score for the training clip x_t. This score will be the base of our clip weighting.

Temporal softmax sampling follows the formula:

$$p(S(x) = x_t) = \frac{\exp(\gamma w_{x,t})}{\sum_{n=1}^{N} \exp(\gamma w_{x,n})} \; \forall x_t \subset x. \qquad \text{(S.1)}$$

Distribution updates. At every epoch, a single clip is selected from each video in the training dataset. We apply inference on the sampled clip without data-augmentation (as this would introduce noise in the score distribution), and separately use a copy with data-augmentation to train the model. The importance of evaluating scores from "clean" clips is further discussed in Section S.3.

We employ a propagation mechanism to update several clip probabilities from a single clip evaluation. After training with clip x_t, obtaining classification score $f(x_t)$, we update the estimate of score $w_{x,t+i}$ for all i in $[-F; F]$ with linear interpolation:

$$w_{x,t+i} = w_{x,t+i} + \frac{F - |i|}{F}(f(x_t) - w_{x,t+i}). \qquad \text{(S.2)}$$

Training phases. Efficient training-clip sampling is highly dependant on the accuracy of the temporal distributions. Since the estimate w is built iteratively and the model is trained simultaneously, it could take a long time for clip sampling to become interesting. Waiting for all clips to be evaluated before starting to sample them efficiently would require an unreasonable number of epochs. Also, we do not want to update distributions with classification scores obtained from an untrained model. Therefore, we implemented several mechanisms to bootstrap the distributions, to make them representative of the informativeness of clips as early as possible during training, without introducing heavy computational overhead. We decompose the training process into three simple steps relative to the sampling mechanism: first, warm-up with uniform sampling and no distribution updates, then, exploration with deterministic sampling and initialization of distributions, and

finally exploitation with softmax sampling and distribution updates as described above. The number of epochs associated to each of these phases can be adapted to the task, based on the total duration of training, the average video length and the clip size.

During the first 3 epochs, uniform sampling is used without updating the distributions. This warm-up time allows the model to jump from 14% to 25% validation accuracy on AFEW, which is about half of the final accuracy. After the 3rd epoch, the classification scores are far more reliable to update sampling distributions.

Before starting to exploit the sampling mechanism, we need to build entire temporal distributions of the videos in order to have information on the relative importance of each clip. One solution is to compute classification scores on the entire training videos with an inference step. This would be very expensive as a single training video can have hundreds of frames, and the number of possible F-frame overlapping clips to evaluate is $L-F+1$. Also, the model's temporal receptive field and the duration of training clips are not equal, so using score vectors computed convolutionally from long videos might not be reliable and wouldn't be coherent with the scores obtained from short clips in the exploitation step. Instead, we propose a lighter exploration step that integrates smoothly into our framework. For 5 epochs, we sample training clips deterministically from uniformly spaced temporal locations. The selected clips are used for training and provide classification scores to initialize the corresponding $\frac{1}{5}$ of the distributions. In our implementation, the score is obtained before the back-propagation. Evaluating the clip right after training on it would bias the distributions toward rewarding most fitted clips, while we aim at evaluating their informativeness. Overall, this exploration step enforces a diversity of clips in the early stages, which is important for building representative distributions.

Then, for the main part of the training, training clips are sampled stochastically, based on the softmax probabilities computed from w. We keep updating the distributions throughout training. As clips share a lot of frames with their neighbours, we update the distribution around the sampled temporal location. We consider this particularly important in our experiments because the number of clips in videos is generally greater than the number of training epochs. We use linear interpolation centered on the selected clip and propagate to 16 frames on each side, with decreasing update weight further from the center. The settings for the number of steps in each step and the method for smoothing distributions can be optimized and adapted to the task at hand, and require more attention in future work.

S.2. Discussion on sampling temperatures

To evaluate the benefits of temporal softmax weighting, we study the impact of the joint sampling and pooling temperature parameter on the classification performance and training time. Results on AFEW are reported in Table S.1. The baseline is the uniform training, with average pooling during inference. This is equivalent to the temperature $\gamma = 0$ in the softmax framework. However, our method comprises warm-up and exploration as described in the implementation discussion, so we report both the results with uniform sampling and with $\gamma = 0$. We can see that deterministic exploration seems to be beneficial as it provides more variations in training data, but consequently delays convergence by acting as a regularizer. Weighted sampling is effective after the 8^{th} epoch (3 epochs of warm-up and 5 epochs of exploration). With large γ temperature parameters, the best clips are exploited more often, leading to lower training data variation and more informative clips. We observe a shortening in the duration (epochs) of the training process. The best classification performance (47.35%) is obtained with softmax temperature $\gamma = 1$, which provides a compromise between uniform and maximum temperatures, effectively focusing on relevant clips while maintaining diversity in selection. Results with softmax temperatures during training and average pooling during testing demonstrate the effect of sampling temperature on the learning phase. Interestingly, we observe that having different temperature during training (clip sampling) and during testing (temporal pooling) can lead to even better performance. When considering this possibility, the best accuracy (47.55%) is obtained with $\gamma_s = 1$ for training-clip sampling, and $\gamma_p = 10^6$ for softmax pooling. The *Sampling only* experiment on UNBC-McMaster reported in Table S.4 also supports this hypothesis. It shows that weighted sampling improves training quality on its own (more details below). Although here, average pooling at test time seems more efficient than max pooling, probably due to differences between the categorical emotion recognition task and the binary pain detection setup (the benefits of temporal softmax being limited for No Pain videos). As we designed this method as a unifying framework, we do not extensively study the effect of decoupling the two softmax temperatures.

S.3. Additional Results and Discussions

Clean sample scoring. In order to update the sampling distributions, it is straightforward to use the scores obtained during training. However, these scores are subject to data-augmentation and dropout. This introduces noise in the estimation of temporal distributions. Table S.2 shows this phenomenon. Using "clean" copies of the clips to evaluate their score makes temporal sampling more efficient. In our experiments, using softmax temperature $\gamma = 10$, accuracy was 46.65% with data-augmented samples and 46.91% with clean samples (with a uniform sampling baseline of 45.87%). The computational overhead is very limited as it consists in adding only an inference step on small clips with

Training γ_s	**Accuracy (%)**	**Epochs**	**Test $\gamma_p = 0$**	**Test $\gamma_p = 10^6$**
uniform	45.66 $\pm$0.21	24.55 $\pm$2.75	45.66	46.91
$\gamma_s = 0$	46.07 $\pm$0.20	25.66 $\pm$3.14	46.07	46.86
$\gamma_s = 0.5$	46.07 $\pm$0.27	23.56 $\pm$3.09	45.61	47.00
$\gamma_s = 1$	47.35 $\pm$0.27	20.33 $\pm$1.72	46.59	**47.55**
$\gamma_s = 10$	46.65 $\pm$0.40	17.22 $\pm$2.20	45.84	46.76

Table S.1. Results obtained by decoupling training (sampling γ_s) and testing (pooling γ_p) softmax temperatures on the AFEW dataset. Models are trained with the clip-sampling strategy indicated in the left column, and results are provided for tests with average ($\gamma_p = 0$) and max ($\gamma_p = 10^6$) video-level temporal pooling. The *uniform* entry and $\gamma_s = 0$ differ because of the deterministic exploration at the beginning of training.

Training method	**Acc. (%)**
Uniform training $\gamma = 0$	45.87
Clean scoring, $\gamma = 10$	46.91
Data-augmented scoring, $\gamma = 10$	46.65

Table S.2. Influence of scoring from clean samples compared to directly using the data-augmented training samples, on AFEW.

	Clip duration (frames)		
γ **Temp.**	**8**	**16**	**32**
0	45.17	45.78	47.09
1	46.39	47.00	47.43
10	46.04	45.85	47.17

Table S.3. Influence of training-clip duration for classification accuracy (%) with 3D-CNN stochastic softmax on AFEW.

Method	**EER Acc. (%)**	**Epoch**
Our baseline 3D VGG (unif.)	86.58	43.0
Stochastic Softmax ($\gamma = 2$)	87.21	37.4
Sampling only ($\gamma_s = 2$)	87.63	35.2
PSPI sampling ($\gamma_s = 0.8$)	87.84	25.8

Table S.4. Additional results of a 3D CNN on UNBC-McMaster, we compare the baseline (uniform training and average pooling) with temporal stochastic softmax as proposed in the paper ($\gamma_s = \gamma_p = 2$), a decoupled version of temporal stochastic softmax ($\gamma_s = 2$ and $\gamma_p = 0$) and an experiment involving expert frame-level labels to guide sampling.

no back-propagation. We can note that even when taking the readily available scores to update the distributions, our sampling method performs better than uniform sampling.

Clip duration. We study the effect of training-clip duration on classification accuracy, for different temperatures of sampling. We perform very limited hyper-parameter search for this study, so performances could probably be improved for large clip duration. Results presented in Table S.3 show that accuracy improves with clip size, but the impact of stochastic softmax is greater for smaller clips. Uniform sampling ($\gamma = 0$) particularly benefits from larger clips, as they will reduce noise in gradients and training inputs will be closer to those in inference mode (long videos). Weighted sampling on the contrary has more impact with small clips, as they allow for more precise focus and avoiding of irrelevant clips. Note that all clips become similar when their size is large, with more overlapping frames.

Sampling with frame-level labels. As the UNBC-McMaster dataset provides expert-annotated PSPI scores, measuring pain intensity at each frame, it can constitute an alternative to our estimated sampling distributions. Table S.4 reports the performance of a model trained with short-clips sampled with the PSPI distributions. As the PSPI range is 0-16, much higher than the classification scores produced by the model, we use a temperature $\gamma_s = 0.8$. With an accuracy of 87.84%, this model performs better than the weakly-supervised model. The improvement is quite limited, confirming that stochastic softmax is able to estimate meaningful distributions from sequence-level labels only.

Figure S.1 provides more details to compare PSPI and weakly-supervised sampling. The distributions estimated from sequence-level classification scores have clear similarities with the PSPI curves. We see that a data-oriented sampling strategy can replace the need for more labels. Theoretically the proposed method could also learn distributions for No Pain, while PSPI scores are generally zero for this class, but this doesnt seem to be relevant in our experiments.

Also, Werner *et al*. [74] discussed limitations of the Prkaching and Solomon Pain Intensity (PSPI) scores, how they do not always correspond to pain expressions, and how their high temporal resolution might be misleading. This suggests that using expert annotations are not necessarily the best approach, even when they are available. However, a clear advantage of the PSPI-based sampling is the possibility to train with high intensity sampling directly, without exploration. In our experiments, this translates into a reduction of training time from 35.2 to 25.8 epochs in average.

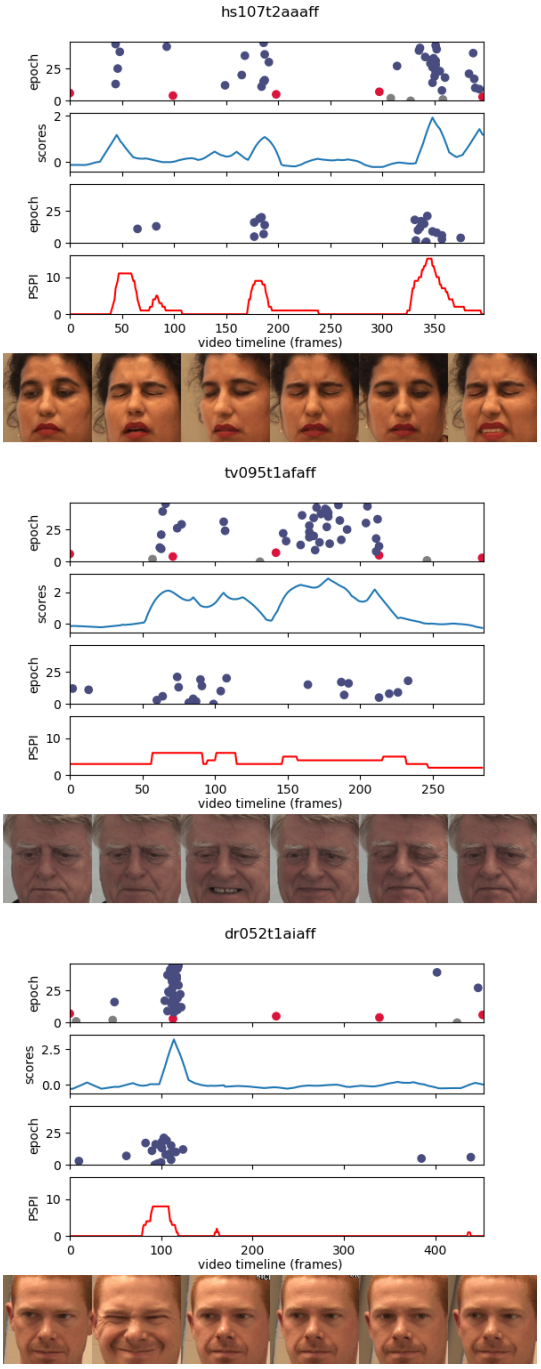

Figure S.1. Visualizations of stochastic softmax training for three Pain samples of the UNBC-McMaster dataset. For each sample, the figures describe, from top to bottom, the sampling maps (temporal location for each epoch) and corresponding temporal sampling distributions, for stochastic softmax training from sequence-level labels (OPI) versus frame-level annotations (PSPI). The sequence-level label of each video is OPI 3 for *hs107t2aaaff* and *tv095t1afaff*, OPI 5 for *dr052t1aiaff*.

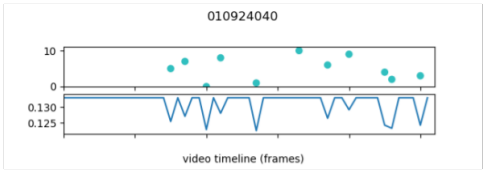

Figure S.2. Distribution updates obtained with REINFORCE are small and localized. The distributions take too much time to fit for the needs of training.

S.4. Additional visualizations of training

Figure S.2 displays an example of training distribution obtain with the noisy updates of REINFORCE.

Figures S.3 to S.10 provides illustrative examples, for AFEW and BioVid datasets, of the temporal stochastic softmax training process. They report sampling distributions and the temporal positions of the sampled clips at each epoch of training for a specific training video. They show that the model is able to estimate meaningful distributions that correspond to the observable emotion or pain level. These distributions are built from evaluating short training clips online, iteratively through the training process.

We can note the general difference in the prediction score (logits) intensities between the datasets. This is probably due to the weight initialization of the model, which involves pretraining on 2D emotion recognition. The temperature parameter can be adapted to task-specific distributions of logits to obtain the desired sampling strategy.

On BioVid, Figure S.9 shows that the model is able to learn different expressions of pain. We also visualize distributions obtained for No Pain samples of BioVid (BL1) in Figure S.10. It is clear that the logits are too small to provide any real advantage over uniform sampling and average pooling for these neutral states.

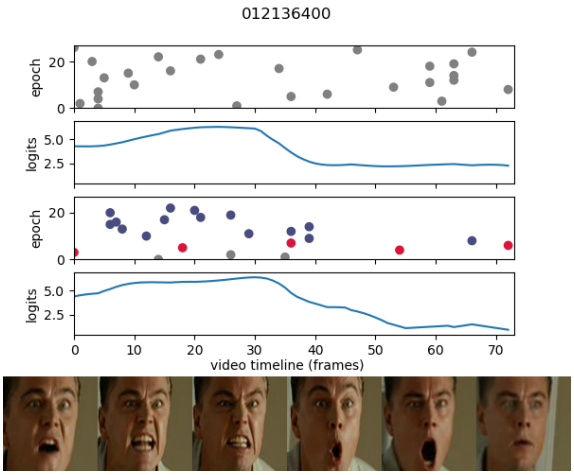

Figure S.3. Visualization of sampling distributions for uniform training (above) and softmax temperature 1 (bellow), for sample 012136400 of the Angry category.

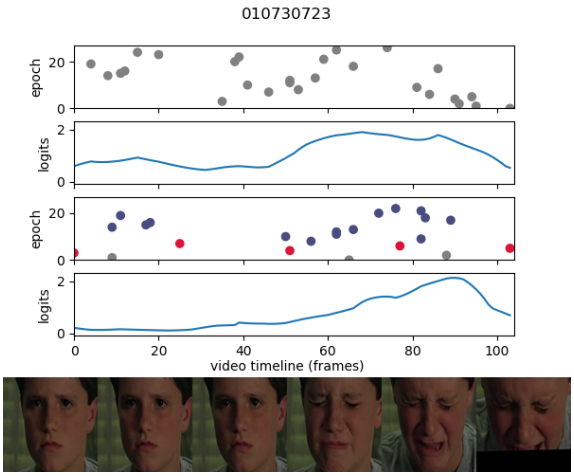

Figure S.4. Visualization of training for sample 010730723 of the Sad category, with a clear emotional progression from Neutral to Sad, and occlusion in the final frames.

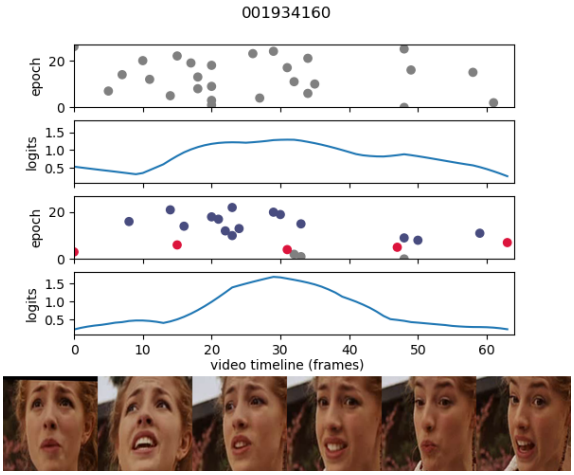

Figure S.5. Visualization of training for sample 001934160 of the Disgust category, focusing on the least ambiguous expressions.

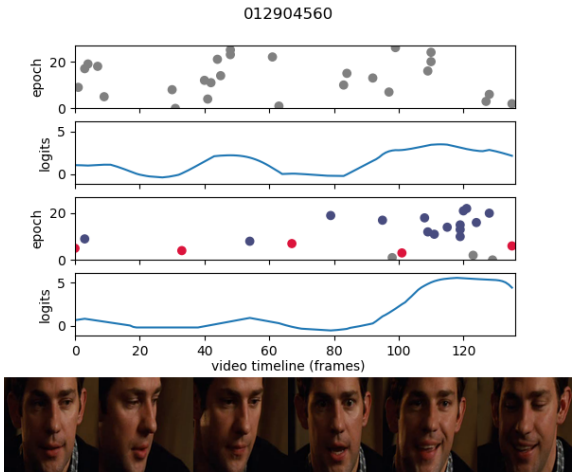

Figure S.6. Visualization of training for sample 012904560 of the Happy category, avoiding Neutral and Surprise expressions to focus on the Happy frames at the end of the video.

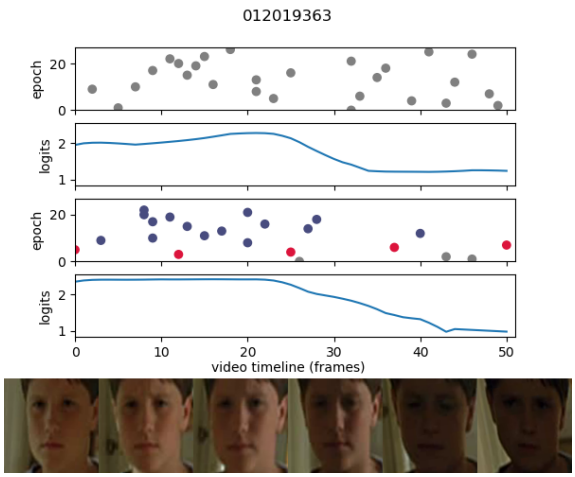

Figure S.7. Visualization of training for sample 012019363 of the Neutral category, with lightning variations rendering the end of the video uninformative.

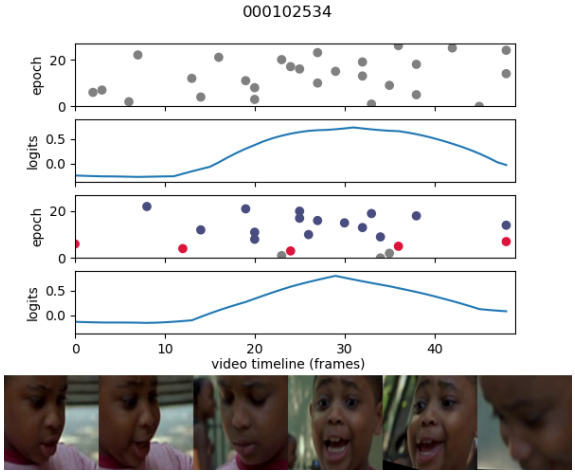

Figure S.8. Visualization of training for sample 000102534 of the Surprise category, with clear apex, and head-pose variations.

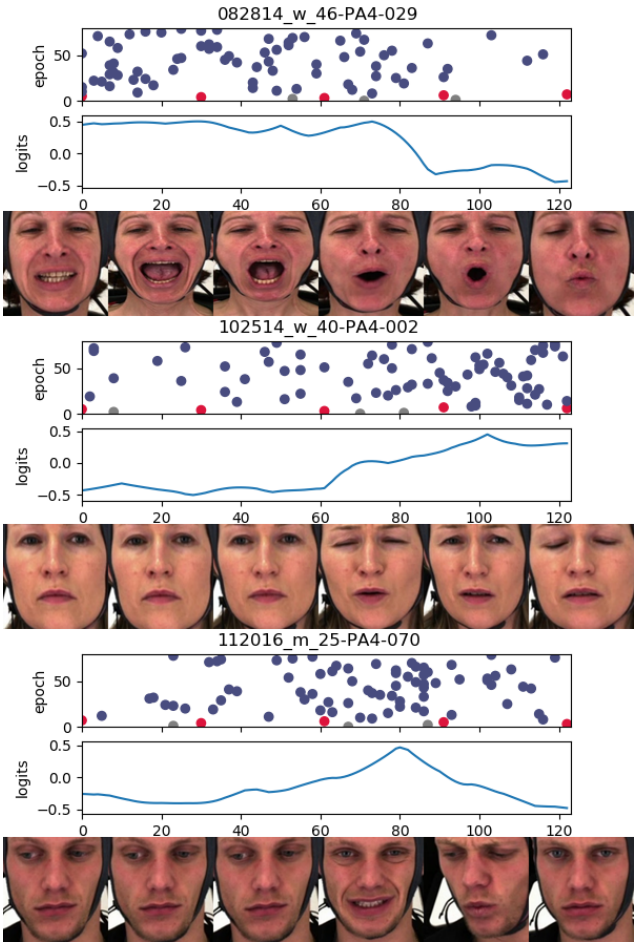

Figure S.9. Visualizations of stochastic softmax training for three Pain (PA4) samples of the BioVid dataset. The inverse temperature parameter is set to $\gamma = 2$.

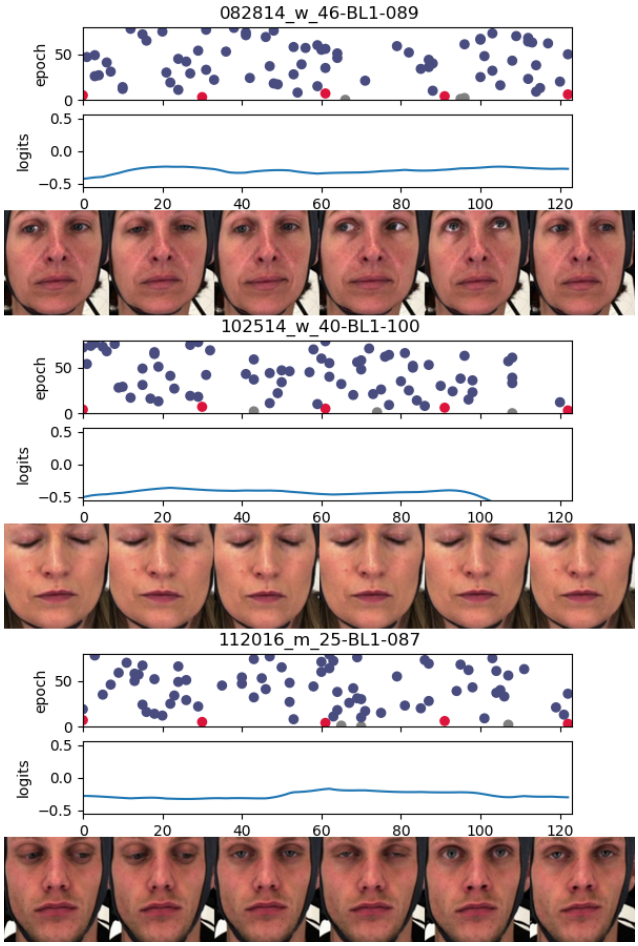

Figure S.10. Visualizations of stochastic softmax training for three No Pain (BL1) samples of the BioVid dataset. The logits are small and negative, resulting in almost uniform sampling, even with high values of γ.

Reference	Architecture	Accuracy
Li et al., 2019 [69]	ResNet-18 + BLSTM	43.34
2nd place AV EMOTIW 2019	DenseNet-121 + BLSTM	49.35
Lu et al. 2018 [71]	3D VGG-16	39.36
3rd place AV EMOTIW 2018	VGG BLSTM	53.91
Liu et al. 2018 [70]	4CNNs + LSTM	56.13
1st place AV EMOTIW 2018	3D landmarks + SVM	39.95
Fan et al. 2018 [66]	VGG-Face	45.16
2nd place AV EMOTIW 2018	FG-Net	47.00
Vielzeuf et al., 2018 [72]	ResNet-18 av. pool.	49.7
3rd place AV EMOTIW 2018	weighted av. pool	50.2
Vielzeuf et al., 2017 [73]	LSTM C3D	43.2
4th place AV EMOTIW 2017	Weighted C3D	42.1
Fan et al. 2016 [67]	C3D	39.69
Bargal et al., 2016 [65]	VGG-13	57.07

Table S.5. Results reported on the AFEW dataset from the literature. Differences in methodology, testing sets, use of extra-data and other factors make any comparison of these results hazardous. We provide this table as an overview of approaches and performances reported in the literature.

References

[65] S. A. Bargal, E. Barsoum, C. Canton-Ferrer, and C. Zhang. Emotion recognition in the wild from videos using images. In *ICMI*, pages 433–436, 2016.

[66] Y. Fan, J. C. K. Lam, and V. O. K. Li. Video-based emotion recognition using deeply-supervised neural networks. In *ICMI*, pages 584–588, 2018.

[67] Y. Fan, X. Lu, D. Li, and Y. Liu. Video-based emotion recognition using CNN-RNN and C3D hybrid networks. In *ICMI*, pages 445–450, 2016.

[68] B. Korbar, D. Tran, and L. Torresani. Scsampler: Sampling salient clips from video for efficient action recognition. In *ICCV*, pages 6231–6241, 2019.

[69] S. Li, W. Zheng, Y. Zong, C. Lu, C. Tang, X. Jiang, J. Liu, and W. Xia. Bi-modality fusion for emotion recognition in the wild. In *ICMI*, pages 589–594, 2019.

[70] C. Liu, T. Tang, K. Lv, and M. Wang. Multi-feature based emotion recognition for video clips. In *ICMI*, pages 630–634, 2018.

[71] C. Lu, W. Zheng, C. Li, C. Tang, S. Liu, S. Yan, and Y. Zong. Multiple spatio-temporal feature learning for video-based emotion recognition in the wild. In *ICMI*, pages 646–652, 2018.

[72] V. Vielzeuf, C. Kervadec, S. Pateux, A. Lechervy, and F. Jurie. An occam's razor view on learning audiovisual emotion recognition with small training sets. In *ICMI*, pages 589–593, 2018.

[73] V. Vielzeuf, S. Pateux, and F. Jurie. Temporal multimodal fusion for video emotion classification in the wild. In *ICMI*, pages 569–576, 2017.

[74] P. Werner, A. Al-Hamadi, K. Limbrecht-Ecklundt, S. Walter, S. Gruss, and H. C. Traue. Automatic pain assessment with facial activity descriptors. *IEEE Trans. Affect. Comput.*, 8(3):286–299, 2017.

[75] W. Zhu, J. Hu, G. Sun, X. Cao, and Y. Qiao. A key volume mining deep framework for action recognition. In *CVPR*, pages 1991–1999, 2016.