MicroExpNet: An Extremely Small and Fast Model For Expression Recognition From Frontal Face Images

İlke Çuğu, Eren Şener, Emre Akbaş
Department of Computer Engineering, Middle East Technical University
06800 Ankara, Turkey
{cugu.ilke, sener.eren}@metu.edu.tr, emre@ceng.metu.edu.tr

Abstract

This paper is aimed at creating extremely small and fast convolutional neural networks (CNN) for the problem of facial expression recognition (FER) from frontal face images. We show that, for this problem, translation invariance (achieved through max-pooling layers) degrades performance, especially when the network is small, and that the knowledge distillation method can be used to obtain extremely compressed CNNs. Extensive comparisons on two widely-used FER datasets, CK+ and Oulu-CASIA, demonstrate that our largest model sets the new state-of-the-art by yielding 1.8% *and* 12.7% *relative improvement over the previous best results, on CK+ and Oulu-CASIA datasets, respectively. In addition, our smallest model (MicroExpNet), obtained using knowledge distillation, is less than* 1*MB in size and works at* 1408 *frames per second on an Intel i7 CPU. Being slightly less accurate than our largest model, MicroExpNet still achieves a* 8.3% *relative improvement, on the Oulu-CASIA dataset, over the previous state-of-the-art, much larger network; and on the CK+ dataset, it performs on par with a previous state-of-the-art network but with* 154*x fewer parameters.*

1. Introduction

Expression recognition from frontal face images is an important aspect of human-computer interaction and has many potential applications, especially in mobile devices. Face detection models have long been deployed in mobile devices, and relatively recently, face recognition models are also being used, e.g. for face based authentication. Arguably, one of the next steps is the mobile deployment of facial expression recognition models. Therefore, creating small and fast models is an important goal. In order to have an idea about the current situation, we looked at the size and runtime speeds of two representative, currently state-of-the-art models, namely PPDN [41] and FN2EN [4]. In terms of the number of total parameters in the network, both models are in the order of millions (PPDN has 6M and FN2EN has 11M). In terms of runtime speed, both models run at 5ms per image on a GTX 1050 GPU, however, on an Intel i7 CPU, while PPDN takes 17.73 ms, FN2EN takes 294.66 ms. Further details on size and speed can be found in Tables 6 and 7.

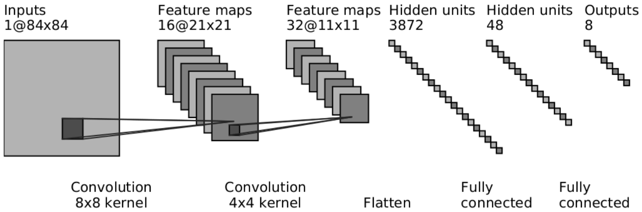


Figure 1. The architecture of MicroExpNet, our smallest (71K parameters, 0.93 MB in size) and fastest (1408 FPS on Intel i7 CPU) model. It performs on par with or outperforms the current state-of-the-art models which are much larger and slower.

The central question that motivated the present work was how much we could push the size and speed limits so that we end up with a compact expression recognition model that still works reasonably well. To this end, we first explored training large models for the problem and we were able to outperform the previous best results and set the new state-of-the-art on two widely used benchmark FER datasets, CK+ [26] and Oulu-Casia [40]. Then, using the "knowledge distillation" method [13], we were able to create a family of small and fast, yet reasonably accurate models. Our smallest model, which we dub MicroExpNet (Fig. 1), is extremely compact (less than 1MB in size) and fast (runs at 1408 frames per second, or 0.71 ms per image on Intel i7 CPU). Such a model would enable deployment in mobile devices. MicroExpNet, achieves slightly less accurate results than our large model. However, compared to the current state-of-the-art, it still yields better or on par results, using a much smaller number of parameters.

Overview. In this work, we focused only on frontal face images. First, we trained a large network (called the teacher or TeacherNet), and then used the "knowledge distillation" method [13] to train compact networks. In this method, the softmax output of the teacher network is used to "guide" the training of the student network via a hyperparameter called the "temperature." We conducted comprehensive temperature and network size experiments. Compared to the teacher network, MicroExpNet (i.e. the smallest student network) is 110x smaller in size, has 330x fewer parameters and is 296x faster. We also hypothesized that invariance to translation – typically achieved using max-pooling layers – would not be useful for the facial expression recognition problem as the expressions are sensitive to small, pixel-wise changes around the eye and the mouth. In our experiments, we found that, especially for small networks, max-pooling hurts the performance.

Contributions. Our contributions in this work are fourfold:

1. **We set the new state-of-the-art on CK+ and Oulu-CASIA datasets.** Our TeacherNet yields 1.8% and 12.7% relative improvement over the previous best results, on CK+ and Oulu-CASIA datasets, respectively. In addition, we achieve these results using a simpler training process compared to, for example, the multi-stage training of FN2EN [4] and the pairwise training scheme of PPDN [41].

2. **We show that it is possible to create extremely small and fast, yet accurate, models for the FER problem.** Our smallest model, MicroExpNet, has 71 thousand parameters (*cf.* millions in PPDN and FN2EN) and runs at 1408 frames per second on an Intel i7 CPU (*cf.* 56 for PPDN, 3.4 for FN2EN).

3. **We show that translation invariance degrades the performance in FER – especially when the network is small** (see Section 4.1). To the best of our knowledge, all current state-of-the-art methods use a mix of max and average pooling. Our finding could guide the design of future models on this problem.

4. **We showed that the effect of "knowlegde distillation" (compared to training from scratch) increases as the network size gets smaller.** To the best of our knowledge, this kind of analysis has not been done before. Whether this effect is specific to the FER problem is yet to be seen (left as future work).

In order to support our contributions, we provide standard classification performance comparison for CK+ and Oulu-CASIA datasets; parameter count comparison; runtime speed comparison; max pooling vs. no pooling analysis; extensive temperature and size combinations to provide answers to "what if" questions.

2. Related work

2.1. Facial expression recognition (FER)

We categorize the previous work as image (or frame) based and sequence based. While image-based method analyse individual images independently, sequence-based methods exploit the spatio-temporal information between frames.

Image based. There are three groups of work. Models that use 1) hand-crafted features (HCFs), 2) deep representations, and 3) both. Our work falls into the second group.

We do not focus on HCF models [2, 7, 35, 42] here because they are obsolete (with the emergence of deep models) and in general, they do not achieve competitive results. These works typically extract Local Binary Patterns (LBP) [29], SIFT [35] or Gabor features and use SVM [3] or AdaBoost [8] on top of these features, as the classifier.

Deep representations learned from face images are the main ingredients of [25, 27, 41, 4, 17]. Liu *et al.* [25] proposed a loopy boosted deep belief network framework which consists of a bottom-up unsupervised feature learning and a "boosted top-down supervised feature strengthen" process. Final features are combined to form an AdaBoost classifier. Our approach does not contain any feature selection or "strengthen" operation, and also we did not employ an Adaboost classifier. Mollahosseini *et al.* [27] introduced an inception network for expression recognition. They preprocessed face images via applying an affine transformation using facial landmark points. Their model is much larger compared to ours considering the two large fully connected layer at the end of their network. In addition, we did not use any inception layers or preprocessing step for face images. Zhao *et al.* [41] proposed a peak-piloted deep network architecture where they use both peak and non-peak expression images as pairs during training. The architecture used GoogLeNet [37] as basis, and employed the peak-piloted approach in the last two fully-connected layers. Training peak and non-peak images in pairs naturally requires their proposed back-propagation algorithm which adds complexity to implementation compared to our work. FaceNet2ExpNet [4] employs a multi staged model production for expression recognition. First, the authors trained five convolutional layers with the supervision of a pre-trained FaceNet's [30] pool5 layer outputs. Then, they append a fully connected layer to trained convolutional layers and start training to classify expressions. This is similar to the FitNet [31] approach. In our work, teacher net is trained in isolation and its guidance is applied only through its soft targets (see Section 3.1 for details). Recently, Kim *et al*. [17] introduced a deep generative contrastive model for facial expression recognition. They combined encoder-decoder networks and CNNs into a unified network that simultaneously learns to generate, compare, and classify samples on a dataset.

Finally, Levi and Hassner [20] used both HCFs and deep representations to form a hybrid approach. They trained CNNs with both the original input images and 3D mappings of local binary patterns (LBP) [29] derived from the input images. Then, they finalized their model via fine-tuning.

Sequence-based. We can categorize sequence based facial expression classifiers in the same three groups as in the case of image-based classifiers.

We do not focus on HCF based sequence models [11, 9, 33, 34] for the following reasons: 1) we do not use any HCFs, 2) HCF based models became obsolete with the emergence of deep models and they do not yield competitive results.

Deep representations are the core ingredient of [24, 5]. Liu *et al*. [24] identified facial expression recognition task as a combination of temporal alignment and semantics-aware dynamic representation problems, and proposed a manifold modeling of videos based on mid-level deep representations (expressionlet). These representations gathered via learned spatio-temporal filters. Kahou *et al*. [5] fused CNNs with recurrent neural networks (RNNs) to build a hybrid model working on videos. CNN is used on static images to gather high-level representations, then the final classification is done by RNN trained with those representations. They also used an SVM to process audios which is beyond the scope of this paper.

In a study done by Jung *et al*. [16] both HCFs and deep representations are used to form a hybrid approach. They presented two deep network models. First a 3D-CNN to extract the temporal appearance features from image sequences. Second, a fully connected deep neural network which captures geometrical information about the motion of the facial landmark points. By combining these two networks, they built a model which uses both landmark points and image sequences.

2.2. Model size reduction

FitNets. Romero *et al*. [31] built their FitNets using the “knowledge distillation” method to produce deep and thin student networks with comparable or better performance compared to the teacher. They built student networks that are thinner but deeper than their teacher by training some layers of the student beforehand with the teacher’s supervision for better initialization. They trained the whole student network using knowledge distillation to finalize their model. They applied their model to object recognition, handwriting recognition and face recognition where the FitNet failed to outperform the state-of-the-art solutions, but achieved superior performance against its teacher. To the best of our knowledge, we are the first to apply knowledge distillation to the facial expression recognition (FER) problem. In addition, we choose a model that is much shallower than the teacher and avoid any pre-training of the student to prevent increasing the complexity of the overall training procedure. Another important point is that, Romero *et al*. did not give much information on the selection of the temperature parameter, in which we do a systematic analysis.

SqueezeNets. Iandola *et al*. [14] proposed a CNN with no fully connected layers to reduce the model size, and preserved the classification performance via their fire modules. Like FitNets, they also did not test their model on FER.

3. Methodology

3.1. Knowledge distillation

Knowledge distillation was introduced by Hinton et al. [13] in 2015. The main idea is to have a cumbersome network called *the teacher* to supervise the training of a much smaller network called *the student* via soft outputs. The algorithm is as follows: first, a large teacher network is trained for the task using an empirical loss calculated with respect to one-hot vector of true labels. Then, a much smaller student network is trained using both one-hot vectors of the true labels and the softmax outputs (Eq. 2) of the teacher network. The aim is to increase the information about the target classes by introducing uncertainty into probability distributions. Since these distributions contain similarity information on different classes, Hinton *et al*. further used this similarity information coming from the teacher to correctly classify a target class intentionally removed from the training set of the student. Additionally, in order to prevent the teacher’s strong predictions to dominate the similarity information, softmax logits(z_i) of the teacher are softened using a hyperparameter called temperature denoted as T in Eq. 1.

Formally, let p_t be the softened output of the teacher’s softmax, z_i be the logits of the teacher, p_s be the hard and p_s' be the soft output of the student’s softmax, v_i be the logits of the student, λ be the weight of distillation, y be the ground truth labels, N be the batch size and function $\mathcal{H}$ refers to the cross-entropy. Then:

$$p_t = \frac{e^{z_i/T}}{\sum_j e^{z_j/T}}, \quad p_s' = \frac{e^{v_i/T}}{\sum_j e^{v_j/T}}, \quad p_s = \frac{e^{v_i}}{\sum_j e^{v_j}} \quad (1)$$

and the loss becomes

$$\mathcal{L} = \lambda(\frac{1}{N}\sum_{n=1}^{N}\mathcal{H}(p_t, p_s')) + (1-\lambda)(\frac{1}{N}\sum_{n=1}^{N}\mathcal{H}(y, p_s)). \quad (2)$$

3.2. Network architectures

In this work, we use two convolutional networks, namely the teacher and the student. The teacher is deep and large whereas the student is shallow and small. There are several versions of the student network having different number of parameters. We call our smallest network as MicroExpNet.

Student network. Our student network has a very simple architecture: two convolutional layers and a fully connected layer with rectified linear unit (ReLU) [28] as the activation function and a final fully connected layer as a bridge to the softmax. We hypothesized that translation invariance would not be useful to the FER problem as facial expressions seem to be sensitive to pixel-wise location changes. We report detailed examination about using max-pooling vs no-pooling, in Section 4.1. After this analysis, we decided not to have any pooling layers. Next, we squeezed the student network by reducing the size of its last fully connected layer to have a fairly compact CNN, and we used the knowledge distillation method [13] to keep the state-of-the-art performance. We created four student models, from largest to smallest: M, S, XS and XXS, to determine the most suitable size-performance balance for our final proposal. Table 1 presents the architectures of these four models. We compare their classification performances in sections 4.2 and 4.3, speeds in Section 4.6, and memory requirements in Section 4.5.

Teacher network. We choose to use the Deep Residual Learning [12] network (ResNet) as the teacher network for its proven record of success on classification tasks. ResNets won the 1st places in: ImageNet classification, ImageNet detection, ImageNet localization [32], COCO detection, and COCO segmentation [21]. Specifically, we employed ResNet with 50 layers due to memory constraints. Although ResNet-50 is far from being the largest of its genre, it meets our requirement for a large and deep teacher network.It is denoted as TeacherNet throughout the paper. We present its classification performance in sections 4.2 and 4.3, speed in Section 4.6, and memory usage in Section 4.5.

3.3. Implementation

CK+ & Oulu-CASIA. For each image in CK+, we apply the Viola Jones [38] face detector, and for each image in Oulu-CASIA we use the already cropped versions. All images are converted to grayscale. Then, in order to augment the data, we extract 8 crops (4 from each corner and 4 from each side) from an image with dimensions of 84x84 for students and 256x256 for the teacher. There is no difference on hyperparameter selections for the trainings on these two datasets. Therefore, the settings explained below apply for both CK+ and Oulu-CASIA experiments.

Model	# of Parameters	Architecture
M	1002808	conv1 - kernel: 8x8 , stride: 4 conv2 - kernel: 4x4 , stride: 2 fc1 - in: 3872 fc2 - in: 256 softmax - 8
S	257656	conv1 - kernel: 8x8 , stride: 4 conv2 - kernel: 4x4 , stride: 2 fc1 - in: 3872 fc2 - in: 64 softmax - 8
XS	133464	conv1 - kernel: 8x8 , stride: 4 conv2 - kernel: 4x4 , stride: 2 fc1 - in: 3872 fc2 - in: 32 softmax - 8
XXS	71368	conv1 - kernel: 8x8 , stride: 4 conv2 - kernel: 4x4 , stride: 2 fc1 - in: 3872 fc2 - in: 16 softmax - 8

Table 1. Architectures of the student networks from largest to smallest.

Teacher Network. We employ a ResNet trained on the 1.28 million training images of ImageNet [32] with 50 layers, and fine-tune it on a FER dataset. The base learning rate is set as 10^{-3} and remained constant through iterations. Weight decay is $5 * 10^{-4}$, momentum is 0.9, and mini-batch size is 24. All fine-tuning operations on ResNet-50 finalized after 3000 epochs. As previously done in the literature, the results we report for this model are obtained by averaging the 10-fold cross validation performances.

Vanilla & Student Networks. We have the same hyper-parameters across all of the different model sizes for both vanilla and student trainings. “Vanilla” training means that the network is trained from scratch without any teacher guidance. Weights and biases are initialized using Xavier initialization [10]. Network architectures are implemented via Tensorflow [1]. Adam [18] optimizer is adopted as the optimization algorithm. The base learning rate is set as 10^{-4}, dropout [36] is 0.5, mini-batch is 64 and the weight of the distillation λ is 0.5 (see Section 3.1) for all student models. Selected model sizes are 1M, 257K, 133K and 71K parameters respectively, which are produced by decreasing the size of the $FC1$ layer (see Table 1). Training operations are finalized after 3000 epochs for all models and an additional XXS student model is saved after 50000 epochs as our proposed model which is denoted as **MicroExpNet** which can be considered as a graduated student model. Em-

Model	CK+	Oulu-CASIA
CandidateExpNet_v_M	97.93%	97.68%
CandidateExpNet_$p1_M$	**97.99%**	**97.79%**
CandidateExpNet_$p2_M$	97.41%	96.64%
CandidateExpNet_$p12_M$	97.39%	97.47%
CandidateExpNet_v_S	96.65%	92.95%
CandidateExpNet_$p1_S$	**96.73%**	**93.22%**
CandidateExpNet_$p2_S$	94.09%	88.61%
CandidateExpNet_$p12_S$	94.39%	88.72%
CandidateExpNet_v_{XS}	**93.41%**	**88.73%**
CandidateExpNet_$p1_{XS}$	91.85%	80.16%
CandidateExpNet_$p2_{XS}$	86.84%	77.88%
CandidateExpNet_$p12_{XS}$	88.07%	77.04%
CandidateExpNet_v_{XXS}	**81.91%**	**73.64%**
CandidateExpNet_$p1_{XXS}$	69.05%	52.99%
CandidateExpNet_$p2_{XXS}$	77.74%	66.84%
CandidateExpNet_$p12_{XXS}$	78.52%	61.71%

Table 2. The effect of max-pooling. Classification performances of StudentExpNet candidate models for 1000 epochs of training. $p1$ indicates that there is only one max pooling layer after conv1, $p2$ indicates that there is only one max pooling layer after conv2, $p12$ indicates that each conv layer is followed by a max pooling layer, and v indicated that there is no pooling layer at all. The smaller the network, the more max-pooling degrades the performance.

pirical results are given in Table 4 and Table 5, note that for student networks we only put the best performers across different temperatures (selected using cross-validation). Furthermore, student models are used in temperature selection tests (for detailed explanation see Section 4.4). As previously done in the literature, the results we report for these models are obtained by averaging the 10-fold cross validation performances.

4. Experiments

4.1. Max. Pooling vs. No Pooling Analysis

Facial expressions are located mostly on eyes and mouth [6], and they form only a small fraction of a frontal face image. The idea is to capture these subtle indicators of an emotion by preserving the pixel information across layers. Therefore, our starting point was a CNN with no pooling layers. However, in order to validate our intuition, we build three variations containing max pooling layers for each student. All pooling layers have 2x2 filters with stride 2. All hyperparameters mentioned at Section 3.3 apply to these variations as well. We call them candidate expression networks. These candidates are explained in Table 2.

From the results in Table 2, we draw the following conclusions. When models are large enough, the provided capacity for learning dominates pooling effects. For instance, for the size M, classification performances of candidates are very close to each other. For size S, poolings in later layers drops the performance but early pooling is still the most profitable. After this point (XS and XXS), we begin to see the advantage of not having any pooling layers with significant gains in performance. Combining this observation with our intention to reduce the model size, we decided to employ the architecture with no pooling layer as the foundation of our student networks.

Note that adding a pooling layer drops the number of parameters, thus prevents a proper performance comparison. Therefore, we did two modifications to increase the model size, in order to make it a fair comparison. First, when we add a pooling layer after the first convolutional layer, we decrease the stride of the first conv layer from 4 to 2. This directly recovers all parameters that has been lost. Second, when we add a pooling layers after the second convolutional layer, we increase the number of outputs of the first fully connected layer by 3-fold. This results in having slightly more parameters than the original one (CandidateExpNet$_v$).

4.2. The CK+ dataset

CK+ is a widely used benchmark database for facial expression recognition. This database is composed of 327 image sequences with eight emotion labels: anger, contempt, disgust, fear, happiness, sadness, surprise and neutral. There are 123 subjects. As done in previous work, we extract last three and the first frames of each expression sequence when images are labeled. When unlabeled, we only extracted first frames as neutral. The total number of images is 1574 (see at Table 3), which is split into 10 folds. Each fold contains equal number of frames for each emotion except the last fold in which there are also leftover frames resulted from division to 10. Viola and Jones' [38] face detection algorithm is used to crop the smallest rectangular area containing the face from the image.

Training in Isolation. We evaluate the pre-trained ResNet-50 via fine-tuning on CK+. Then, we train four models, namely VanillaExpNet$_M$, VanillaExpNet$_S$, VanillaExpNet$_{XS}$, and VanillaExpNet$_{XXS}$, from scratch. At this stage, we did not employ knowledge distillation. For all models, we used 3000 epochs for training, and the classification performances are shown in Table 4. According to the table, VanillaExpNet$_M$ and VanillaExpNet$_S$ achieve comparable performance with the state-of-the-art solutions. On the other hand, ResNet-50 outperforms them on CK+. Although Zhao *et al*. [41] seem to achieve better performance than ResNet-50, they use only 6 emotion categories, whereas our model (ResNet-50) uses all of the 8 emotion categories. In the light of these results, we choose ResNet-

	Anger	Contempt	Disgust	Fear	Happy	Sad	Surprise	Neutral	All
CK+	135	54	177	75	207	84	249	593	1574
Oulu-CASIA	240	-	240	240	240	240	240	-	1440

Table 3. The number of images per expression classes in Ck+ and Oulu-CASIA.

50 as the teacher for the knowledge distillation stage.

Training with Supervision. We evaluate four students, namely StudentExpNet$_M$, StudentExpNet$_S$, StudentExpNet$_{XS}$, and StudentExpNet$_{XXS}$, via knowledge distillation on CK+. At this stage, we use the teacher's supervision to improve the learning. As explained in Section 3.3, we need to tune the *temperature* for each student since it is regarded as correlated with model size. Therefore, we conducted an extensive experiment on classification performances for a wide range of $temperatures$. The results are reported in Figure 2. According to these results, fluctuations between performances are increased while models are getting smaller. Consequently, it suggests that large networks are more tolerant to the changes in the temperature. Best performers, regarding their average classification performances for 10-fold cross validation, across different temperatures are then used for performance comparison in Table 4. Our findings (see Fig. 3) show that knowledge distillation can be used to gain back some of the performance lost by decreasing the model size.

4.3. The Oulu-CASIA dataset

Oulu-CASIA has 480 image sequences taken under *dark, strong, weak* illumination conditions. In this experiment, as also done in previous work, we used only videos with *strong* condition captured by a VIS camera. In total, there are 80 subjects and six expressions: anger, disgust, fear, happiness, sadness, and surprise. Similar to CK+, the first frame is always neutral while the last frame has the peak expression. All studies we have encountered on Oulu-CASIA database use only the last three frames of the sequences, so we also use the same frames to be able to compare our method with theirs. Therefore, the total number of images is 1440. As in the earlier studies, a 10 fold cross validation is performed, and the split is subject independent.

Training in isolation. We fine-tune the pre-trained ResNet-50 on the Oulu-CASIA dataset. Then, we train four models, namely VanillaExpNet$_M$, VanillaExpNet$_S$, VanillaExpNet$_{XS}$, and VanillaExpNet$_{XXS}$, from scratch. At this stage, we did not employ knowledge distillation. For all models, training lasted 3000 epochs, and the classification performances are shown in Table 5. According to the table, ResNet-50, VanillaExpNet$_M$, VanillaExpNet$_S$, and VanillaExpNet$_{XS}$ outperform the state-of-the-art solutions for Oulu-CASIA with improvements of 11.12%, 10.21%, 7.95%, and 3.73% respectively. However, since we want the best performer for supervision, we again choose ResNet-50 as the teacher for the knowledge distillation stage.

Method	Accuracy	# of Classes
CSPL [42]	89.9%	Six Emotions
3DCNN-DAP [23]	92.4%	
Inception [27]	93.2%	
AdaGabor [2]	93.3%	
STM-ExpLet [24]	94.2%	
LOMo [34]	95.1%	
LBPSVM [7]	95.1%	
BDBN [25]	96.7%	
DTAGN [16]	97.3%	
FN2EN [4]	98.6%	
DCN [39]	98.9%	
PPDN [41]	99.3%	
AU-Aware [22]	92.1%	Eight Emotions
FN2EN [4]	96.8%	
GCNet [17]	97.3%	
TeacherNet	**99.1%**	
VanillaExpNet$_M$	98.0%	
VanillaExpNet$_S$	97.5%	
VanillaExpNet$_{XS}$	95.2%	
VanillaExpNet$_{XXS}$	85.4%	
StudentExpNet$_M$	98.3%	
StudentExpNet$_S$	98.0%	
StudentExpNet$_{XS}$	97.6%	
StudentExpNet$_{XXS}$	91.4%	
MicroExpNet	**96.9%**	

Table 4. Average classification performances (over 10-folds) of different methods on the CK+ dataset.

Training with supervision. We evaluate four students, namely StudentExpNet$_M$, StudentExpNet$_S$, StudentExpNet$_{XS}$, and StudentExpNet$_{XXS}$, via knowledge distillation on Oulu-CASIA. At this stage, we use the teacher's supervision to improve the learning. As explained for CK+, we again need to tune the *temperature* for each student. Therefore, we also conducted an extensive *temperature* selection experiment for Oulu-CASIA dataset. The results are reported in Figure 4 from which, we can

Method	Accuracy
HOG 3D [19]	70.63%
AdaLBP [40]	73.54%
STM-ExpLet [24]	74.59%
Atlases [11]	75.52%
DTAGN [16]	81.46%
LOMo [34]	82.10%
PPDN [41]	84.59%
GCNet [17]	86.39%
FN2EN [4]	87.71%
TeacherNet	**98.83%**
VanillaExpNet$_{M}$	97.92%
VanillaExpNet$_{S}$	95.66%
VanillaExpNet$_{XS}$	91.44%
VanillaExpNet$_{XXS}$	75.47%
StudentExpNet$_{M}$	98.21%
StudentExpNet$_{S}$	97.63%
StudentExpNet$_{XS}$	94.90%
StudentExpNet$_{XXS}$	80.92%
MicroExpNet	**95.02%**

Table 5. Average classification performances (over 10-folds) of different methods on the Oulu-CASIA dataset.

observe a similar fluctuating behavior as seen in the CK+ experiments. Once again, we can see that large networks are more tolerant to the changes in the temperature than the smaller ones. Best performers across different temperatures are then used for performance comparison in Table 5. We can still observe that the student models perform better than vanilla models (which are trained from scratch without any teacher supervision) for facial expression recognition.

4.4. Temperature analysis

Temperature is a tool to enforce the uncertainty of the teacher network to emerge. This uncertainty may be used as similarity information between different classes to enhance the training. However, there is no formulation for selecting the most effective temperature; it is set empirically. We did a grid search for temperatures of [2, 4, 8, 16, 20, 32, 64, 128] with 10-fold cross validation across all of our student networks using both CK+ (see Figure 2) and Oulu-CASIA (see Figure 4) datasets. According to the results, smaller models are more prone to temperature changes in general, and performances for a given temperature seem rather stochastic. Nevertheless, when calibrated adequately, it improves the overall classification performance as it can be seen at figures 3 and 5.

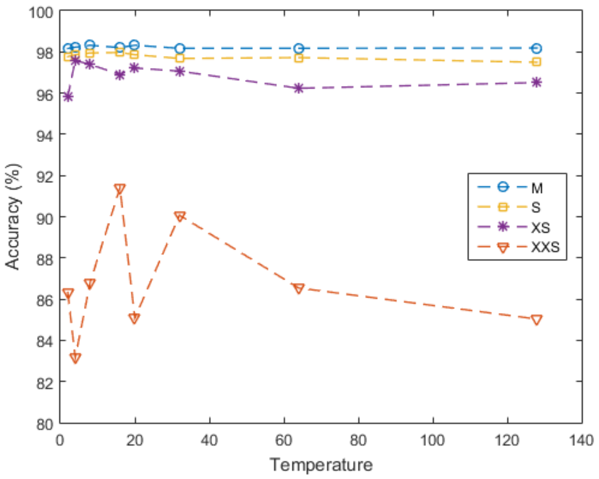


Figure 2. Classification performances of the student networks across different temperatures on the CK+ dataset.

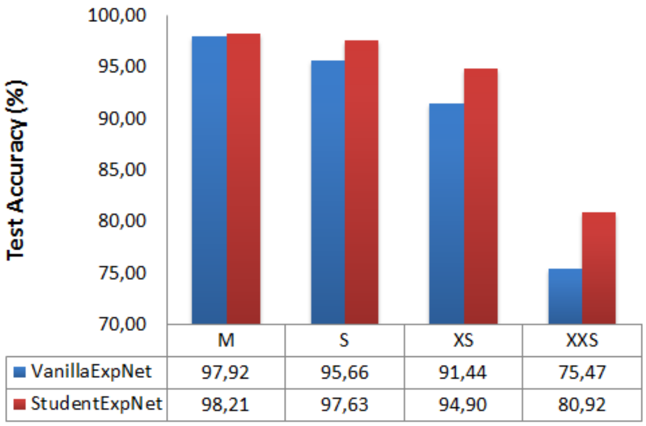


Figure 3. The effect of supervision on CK+ for 3000 epochs of training.

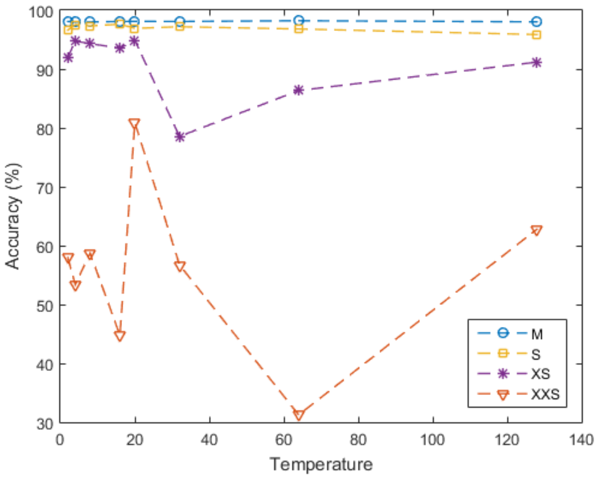
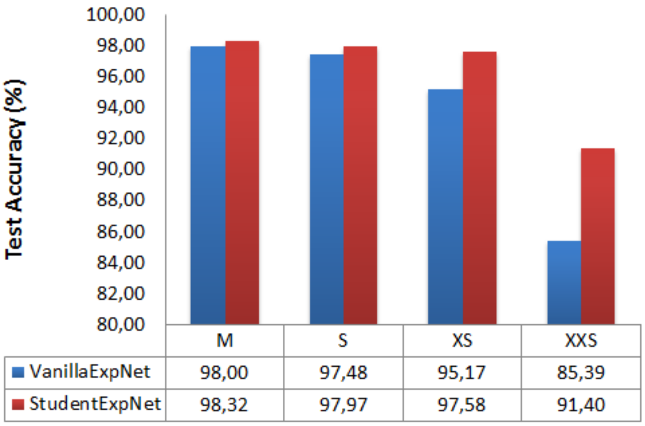


Figure 4. Classification performances of the student networks across different temperatures on the Oulu-CASIA dataset



	M	S	XS	XXS
🟦 VanillaExpNet	98,00	97,48	95,17	85,39
🟥 StudentExpNet	98,32	97,97	97,58	91,40

Figure 5. The effect of supervision on Oulu-CASIA for 3000 epochs of training.

Model	# of Parameters	Size
TeacherNet	23.5M	102.46 MB
FN2EN [4]	11M	35.32 MB
PPDN [41]	6M	50.13 MB
StudentExpNet$_M$	1M	12.08 MB
StudentExpNet$_S$	257K	3.17 MB
StudentExpNet$_{XS}$	133K	1.67 MB
MicroExpNet	**71K**	**0.93 MB**

Table 6. Memory requirements of different FER models.

4.5. Model size analysis

One of the most important benefits of a small neural network is its modest need for memory space. Table 6 shows the comparison of the model sizes in megabytes. Our ultimate facial expression recognition model MicroExpNet takes less than 1 MB to store which is 110x smaller than our teacher network (ResNet-50). In addition, MicroExpNet has 330x fewer parameters than the teacher.

Note that, the results reported for PPDN [41] are our own estimations as they were not available. Moreover, since FN2EN [4] used Caffe [15] framework to deploy their model, although they have more parameters than PPDN, their model has smaller memory requirement.

4.6. Model speed analysis

Another important benefit of a small neural network is its speed. In order to measure the speed, we ran each model for 1000 times with single input image and measure the average run time. Table 7 shows the comparison of the elapsed times to process one image in milliseconds. According to the table, MicroExpNet achieves the best performance by classifying the facial expression in an image in less than 1 ms on an Intel i7-7700HQ CPU. Also, it can be seen that all of the students achieved speeds that are well above the requirements of real-time processing. Ultimately, our final facial expression recognition model, when compared to our teacher network ResNet-50, MicroExpNet is 296x faster on Intel i7-7700HQ CPU, 256x faster on GTX1050 GPU, and 57x faster on Tesla K40 GPU.

Model	i7-7700HQ	GTX1050	Tesla K40
TeacherNet	210.44 ms	197.58 ms	93.29 ms
FN2EN [4]	294.66 ms	4.96 ms	6.28 ms
PPDN [41]	17.73 ms	5.55 ms	-
StudentExpNet$_M$	0.99 ms	0.88 ms	1.74 ms
StudentExpNet$_S$	0.88 ms	0.78 ms	1.69 ms
StudentExpNet$_{XS}$	0.73 ms	0.77 ms	1.63 ms
MicroExpNet	**0.71 ms**	**0.77 ms**	**1.63 ms**

Table 7. Average per-image runtimes of different FER models.

Note that, the numbers reported for PPDN [41] and FN2EN [4] are obtained from our own experiments (as they were not available). We used a custom Tensorflow implementation for PPDN and a Caffe based implementation of the authors for FN2EN.

5. Conclusion

We presented an extremely small (less than 1MB in size) and fast (1408 frames per second) model, called the MicroExpNet, for facial expression recognition (FER) from frontal face images. Our models set the new state-of-the-art on the CK+ and Oulu-CASIA datasets.

From our experimental work, we have drawn the following conclusions. (1) Translation invariance – achieved via max-pooling – degrades performance especially when the network is small. (2) We showed that the knowledge distillation methods works very well the FER problem. (3) “Knowledge distillation”’s effect (compared to training from scratch) gets more prominent as the network size decreases. If this effect is generalizable to other problems/datasets (this is yet to be seen in future work), knowledge distillation can become a mainstream training method when the goal is to produce small networks. (4) The temperature hyperparameter (in knowledge distillation) should be tuned carefully for optimal performance. Especially when the network is small, the final performance fluctuates with temperature.

We are curious about (1) how would incorporating spatio-temporal information (i.e. sequence-based modeling) change our results, (2) how would our model do for non-frontal face images. We left these two items as future work.

Availability

Our codes are available at GitHub.

Acknowledgement

We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Tesla K40 GPU used for this research.

References

[1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016. 4

[2] M. S. Bartlett, G. Littlewort, M. Frank, C. Lainscsek, I. Fasel, and J. Movellan. Recognizing facial expression: machine learning and application to spontaneous behavior. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 568–573. IEEE, 2005. 2, 6

[3] C. Cortes and V. Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995. 2

[4] H. Ding, S. K. Zhou, and R. Chellappa. Facenet2expnet: Regularizing a deep face recognition net for expression recognition. In *Automatic Face & Gesture Recognition (FG 2017), 2017 12th IEEE International Conference on*, pages 118–126. IEEE, 2017. 1, 2, 6, 7, 8

[5] S. Ebrahimi Kahou, V. Michalski, K. Konda, R. Memisevic, and C. Pal. Recurrent neural networks for emotion recognition in video. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, pages 467–474. ACM, 2015. 3

[6] P. Ekman and E. L. Rosenberg. *What the face reveals: Basic and applied studies of spontaneous expression using the Facial Action Coding System (FACS)*. Oxford University Press, USA, 1997. 5

[7] X. Feng, M. Pietikäinen, and A. Hadid. Facial expression recognition based on local binary patterns. *Pattern Recognition and Image Analysis*, 17(4):592–598, 2007. 2, 6

[8] Y. Freund and R. E. Schapire. A desicion-theoretic generalization of on-line learning and an application to boosting. In *European conference on computational learning theory*, pages 23–37. Springer, 1995. 2

[9] D. Ghimire and J. Lee. Geometric feature-based facial expression recognition in image sequences using multi-class adaboost and support vector machines. *Sensors*, 13(6):7714–7734, 2013. 3

[10] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 249–256, 2010. 4

[11] Y. Guo, G. Zhao, and M. Pietikäinen. Dynamic facial expression recognition using longitudinal facial expression atlases. In *Computer Vision–ECCV 2012*, pages 631–644. Springer, 2012. 3, 7

[12] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4

[13] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 1, 2, 3, 4

[14] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and¡ 0.5 mb model size. *arXiv preprint arXiv:1602.07360*, 2016. 3

[15] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 675–678. ACM, 2014. 8

[16] H. Jung, S. Lee, J. Yim, S. Park, and J. Kim. Joint fine-tuning in deep neural networks for facial expression recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2983–2991, 2015. 3, 6, 7

[17] Y. Kim, B. Yoo, Y. Kwak, C. Choi, and J. Kim. Deep generative-contrastive networks for facial expression recognition. *arXiv preprint arXiv:1703.07140*, 2017. 2, 6, 7

[18] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 4

[19] A. Klaser, M. Marszałek, and C. Schmid. A spatio-temporal descriptor based on 3d-gradients. In *BMVC 2008-19th British Machine Vision Conference*, pages 275–1. British Machine Vision Association, 2008. 7

[20] G. Levi and T. Hassner. Emotion recognition in the wild via convolutional neural networks and mapped binary patterns. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, pages 503–510. ACM, 2015. 3

[21] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 4

[22] M. Liu, S. Li, S. Shan, and X. Chen. Au-aware deep networks for facial expression recognition. In *Automatic Face and Gesture Recognition (FG), 2013 10th IEEE International Conference and Workshops on*, pages 1–6. IEEE, 2013. 6

[23] M. Liu, S. Li, S. Shan, R. Wang, and X. Chen. Deeply learning deformable facial action parts model for dynamic expression analysis. In *Asian Conference on Computer Vision*, pages 143–157. Springer, 2014. 6

[24] M. Liu, S. Shan, R. Wang, and X. Chen. Learning expressionlets on spatio-temporal manifold for dynamic facial expression recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1749–1756, 2014. 3, 6, 7

[25] 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, 6

[26] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews. The extended cohn-kanade dataset (ck+): A

complete dataset for action unit and emotion-specified expression. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, pages 94–101. IEEE, 2010. 1

[27] A. Mollahosseini, D. Chan, and M. H. Mahoor. Going deeper in facial expression recognition using deep neural networks. In *Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on*, pages 1–10. IEEE, 2016. 2, 6

[28] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814, 2010. 4

[29] T. Ojala, M. Pietikäinen, and D. Harwood. A comparative study of texture measures with classification based on featured distributions. *Pattern recognition*, 29(1):51–59, 1996. 2, 3

[30] O. M. Parkhi, A. Vedaldi, A. Zisserman, et al. Deep face recognition. In *BMVC*, volume 1, page 6, 2015. 2

[31] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio. Fitnets: Hints for thin deep nets. *arXiv preprint arXiv:1412.6550*, 2014. 2, 3

[32] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015. 4

[33] A. Sanin, C. Sanderson, M. T. Harandi, and B. C. Lovell. Spatio-temporal covariance descriptors for action and gesture recognition. In *Applications of Computer Vision (WACV), 2013 IEEE Workshop on*, pages 103–110. IEEE, 2013. 3

[34] K. Sikka, G. Sharma, and M. Bartlett. Lomo: Latent ordinal model for facial analysis in videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5580–5589, 2016. 3, 6, 7

[35] K. Sikka, T. Wu, J. Susskind, and M. Bartlett. Exploring bag of words architectures in the facial expression domain. In *Computer Vision–ECCV 2012. Workshops and Demonstrations*, pages 250–259. Springer, 2012. 2

[36] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *Journal of machine learning research*, 15(1):1929–1958, 2014. 4

[37] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015. 2

[38] P. Viola and M. J. Jones. Robust real-time face detection. *International journal of computer vision*, 57(2):137–154, 2004. 4, 5

[39] Z. Zhang, P. Luo, C. C. Loy, and X. Tang. From facial expression recognition to interpersonal relation prediction. *arXiv preprint arXiv:1609.06426*, 2016. 6

[40] G. Zhao, X. Huang, M. Taini, S. Z. Li, and M. Pietikäinen. Facial expression recognition from near-infrared videos. *Image and Vision Computing*, 29(9):607–619, 2011. 1, 7

[41] X. Zhao, X. Liang, L. Liu, T. Li, Y. Han, N. Vasconcelos, and S. Yan. Peak-piloted deep network for facial expression recognition. In *European Conference on Computer Vision*, pages 425–442. Springer, 2016. 1, 2, 5, 6, 7, 8

[42] L. Zhong, Q. Liu, P. Yang, B. Liu, J. Huang, and D. N. Metaxas. Learning active facial patches for expression analysis. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2562–2569. IEEE, 2012. 2, 6