Margin-Mix: Semi–Supervised Learning for Face Expression Recognition

Corneliu Florea[1][0000−0001−9754−6795], Mihai Badea[1], Laura Florea[1], Andrei Racoviteanu[1], and Constantin Vertan[1]

Image Processing and Analysis Laboratory, Unversity Politehnica of Bucharest
{corneliu.florea; mihai_sorin.badea; laura.florea; andrei.racoviteanu; constantin.vertan }@upb.ro

Abstract. In this paper, as we aim to construct a semi-supervised learning algorithm, we exploit the characteristics of the Deep Convolutional Networks to provide, for an input image, both an embedding descriptor and a prediction. The unlabeled data is combined with the labeled one in order to provide synthetic data, which describes better the input space. The network is asked to provide a large margin between clusters, while new data is self-labeled by the distance to class centroids, in the embedding space. The method is tested on standard benchmarks for semi–supervised learning, where it matches state of the art performance and on the problem of face expression recognition where it increases the accuracy by a noticeable margin.

Keywords: margin loss; semi–supervised learning; data mixup; face expression recognition

1 Introduction

In the latest period, deep learning techniques acknowledged great advance. One of the ingredients that favored this advance is the collection and annotation of large data corpora [20]. Ordinarily, the data comes in two variants: labeled, when each instance $\mathbf{x_i}$ has the related label y_i and unlabeled, when the instances miss their labels (i.e. there are no y_i). Learning within the case of labeled information is less demanding and it is favored as it has been thoroughly explored. However there are circumstances when labeling is either costly (for instance locating boxes around particular objects in images), or it requires highly trained personnel (encountered, for instance, in the case of medical imaging). In such situations, only a portion of the data is annotated and Semi–Supervised Learning (SSL) algorithms that produce robust solutions using only the limited amount of available annotated data are used to annotate the large volumes of unlabeled data [6].

Our SSL proposal is built upon two principles. The first principle refers to the deep convolutional networks (DCN) characteristic to provide simultaneously decision layers and feature descriptors of the input image [11]. The second principle is that SSL favors the borders through a low density area [6]. Our algorithm seeks to cluster data and create low density areas between borders.

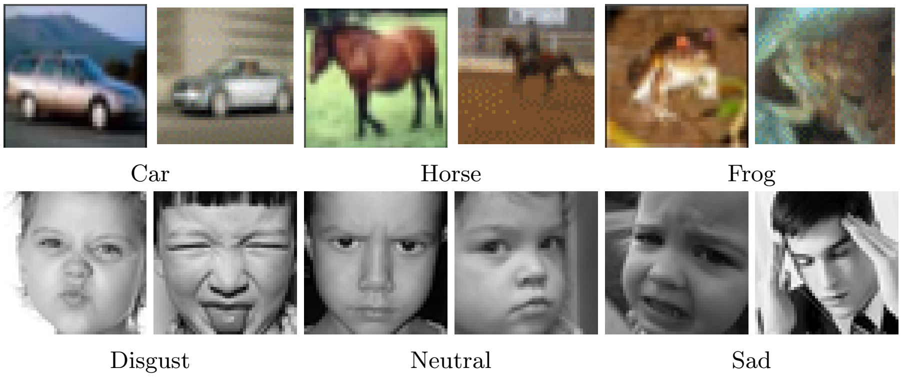


Fig. 1. Examples of the two problems approached. The images from different classed in CIFAR-like databases (top rows) are more different than are face expressions (bottom row).

A particular field that may greatly benefit from semi-supervised learning algorithms is face expression recognition (FER). The topic contains many previous development and areas of interest. For a thorough introduction we kindly refer the reader to the reviews on the topic [33,8]. In this paper, we concentrate our efforts towards the expression categorization into fundamental classes as defined by Ekman *et al.*. [12]: "neutral", "anger", "fear","disgust", "happy","sad", "surprise"; sometimes "contempt" is also included.

With respect to the FER problem, a particular characteristic is the fact that human annotation of such data is hard and costly. "Hard" refers to the fact that the average person has difficulties in differentiating between expressions. In this direction, Susskind et al. [35] showed that an experienced observer (psychology student) reached 89.2%accuracy in a 6 expressions experiment. Alternatively, Bartlett et al. [3] and Ekman et al. [12] noted that more than 100 hours of training are needed for a person in order to get 70% accuracy in recognizing face movements relevant for expressions. To give a reference for comparison, we recall that the average, untrained user achieves ≈ 94% accuracy for image classes on CIFAR-10 [16], reaching 100% on 90% of images [30]. Thus, due to the difficulty in annotating images, problems related to face expression analysis welcome methods and strategies that use additional unlabeled data as a substitute to more annotations, in order to augment the performance.

This contrast can be intuitively associated with the structure and density of the data of the two above mentioned domains. Seemingly, classes from CIFAR-like databases have high intra-class variability, but also low density areas between classes. Classes from any face expression database differentiate between themselves by small and subtle differences (as illustrated in figure 1). Thus, the FER domain has low variance both inter-class and intra-class.

Contribution and paper structure. We propose a semi–supervised learning method, oriented toward classification, that is derived from the classical self-labeling paradigm [6]. The used learner, a deep convolutional neural network, simultaneously clusters and classifies the data. For an unlabeled point, the distance to the class centroids is used for self-labeling. We integrated this idea in the MixUp arrangement [43]. MixUp ensures that the input data space is thoroughly interpolated and it corresponds to a mirror distribution in the prediction space. In our proposal, the intermediate feature space is also thoroughly investigated and the correspondence with input and prediction space is preserved.

More precisely, this paper contributes with: (i) a novel semi-supervised learning algorithm that classifies unlabeled data based on distance to class centroids in a feature space; (ii) the method is showed to be comparable with state of the art in problems with low density areas and to have significant improvements for problems with dense areas between classes, as is the FER problem.

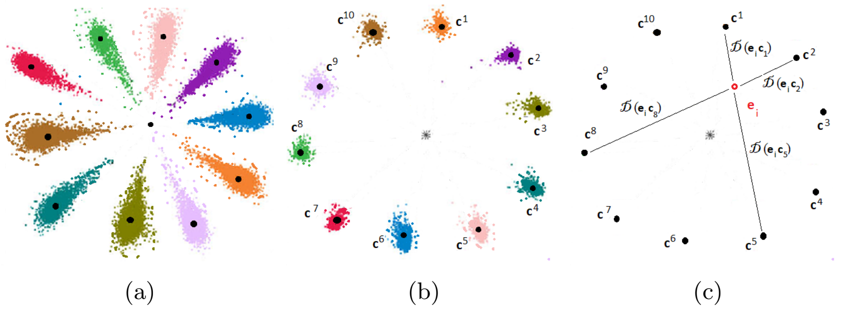
2 Related work

The proposed method seeks discriminative embeddings (features) in DCN while implementing a semi-supervised learning strategy, that is effective for face expression recognition. In this section we provide a short summary over these three directions (discriminative features, SSL and FER).

Loss function for better deep features discrimination. Major contributions in this direction originated in approaches to the face recognition problem. In conjunction to deep learning, several different types of loss function were proposed in the last years. Wen *et al.*. [40] proposed the *center loss* function, to minimize the intra-class distances between the deep features; Liu *et al.*. [24] learned angular discriminative features with the angular softmax loss in order to achieve smaller maximal intra–class distance than minimal inter-class distance; Zhang *et al.*. [44] developed a loss function for long tailed distributions; Zheng *et al.*. [47] showed that normalizing the deep features with the so-called Ring Loss leads to improved accuracy. All these methods were shown to give good results on face recognition tasks, where very large annotated datasets like MegaFace are available. To our best knowledge the strategy of computing discriminative embeddings using the class centroids to annotate new data has not yet been used in general, nor in the context of SSL. A similar concept, but adapted to clustering, may be found in the work of Ren et al. [32]; yet they did not seek consistent data space and prediction in the manner we do here.

From the many existing variants we have relied on a development of the center loss as it uses the Euclidean distance, making this step consistent with data interpolation in the original space due to the MixUp arrangement.

Semi-Supervised Learning. While the problem of SSL has been in the attention of the community for a long period, the appearance of the data hungry deep learning methods brought increased interest. In the context of the deep learning, many initial results were based on generative models such as denoising [31], variational autoencoders [17], or generative adversarial networks[27]. The concept of self labeling has been used in the form of entropy regularization [21].



(a) (b) (c)

Fig. 2. Structuring embeddings on a 2 dimensional layer when the DCN was trained with different losses : (a) softmax dominated and (b) center loss dominated. (c) Our proposal: given a new unlabeled point $\mathbf{e}_i$, one-hot encoding values $\mathbf{y}_i$ are determined by distances to class centroids

In the later period, improved results were obtain by adding consistency regularization losses while processing unlabeled data. The consistency regularization uses the discrepancy between predictions on unlabeled data points and predictions on labeled examples to correct weights. Practical solutions improved performance by smoothing the weight correction before measuring the discrepancy. In this category, one might count Π-Model with Temporal Ensembling [19], Mean Teacher [36] and Virtual Adversarial Training [25], fast-SWA [1] or consistent embedding description in associative domain transfer [15]. More recently, build upon the MixUp strategy [43], models have been constrained to showed consistency with respect to perturbation of the input examples in the MixMatch algorithm [5] or Interpolation Consistency Training - ICT [38]. In summary, it has been showed that consistency between labeled and unlabeled data is helpful; however the consistency has not yet been quantified within a Euclidean space metric onto the intermediate embedding layer, as it is in our proposal.

Face Expression Recognition. This theme has been dominated in the later period by deep learning methods too. For instance, several solutions [46,37] trained a single network or an ensemble of networks and adapted the predictions onto a single independent image or onto a video sequence containing a face expression. The problem of delicate labeling has been addressed by Barsoum et al. [2], who noted the presence of noisy labels in the FER database and thus re–annotated the database by crowdsourcing, showing much improved results; yet the solution was database-specific and overfitting could have appeared. More recently, multiple databases, and thus better generalization, are envisaged in a series of purely supervised methods that augments the baseline performance by the usage of a modified center loss [22]. Others [46], have found that better results can be achieved by specifically selecting some of the layers from the network. Attention mechanisms have also been envisaged for expression recognition with good results [23].

In the later years, the restricted amount of annotated data has been noted and solutions sought to use the power of semi–supervised learning or of the domain transfer to alleviate the limitation. Zhang et al. [45] used a strategy that re-evaluates self labels predictions over randomly selected data instances from the unlabeled data at each iteration. Zeng et al. [42] adopted the self labeling strategy based on bottom-up propagation in a relational graph. Recently, Florea et al. [13] regularized the contribution of unlabeled data with injection of random quantities in the gradient.

Overall, the methods have evaluated the minimal amount of labels required in a database, given a SSL framework, to obtain accuracy values comparable with the supervised case, but more often pushed the supervised performance with database particular choices such as pre–training on specific subsets.

3 Method

From a technical point of view, we propose a methodology to train a deep network in a semi–supervised manner for a classification problem with mutually exclusive categories. In this scenario, we ask the network to include a layer that acts as a discriminative embedding or as a feature descriptor. As discussed in the implementation subsection, we use a WideResNet [41]; in this case, the embedding is the last layer before the decision one, but after flattening. A intuitive view of the system behavior is in figure 2.

For classification problems, initially the label is a scalar y_i indicating a categorical value, but later we will switch to one-hot encoding $\mathbf{y}_i$.

3.1 Large Margin Embedding

Given an image $\mathbf{x}_i$, its associate embedding $\mathbf{e}_i$ and its prediction y_i, one may ask the learner to cluster embedding by incorporating a specific loss. Wen *et al.*. [40] introduced the center loss that explicitly reduces the intra-class variations by encouraging embedding samples to move towards their corresponding class centers in the feature space (embeddings) during training. The center loss is [40]:

$$\mathcal{L}_C = \sum_{i=1}^{N} \mathcal{D}(\mathbf{e}_i, \mathbf{c}^c); \quad \mathbf{c}^c = \frac{\sum_{i=1}^{N} \mu_i^c \mathbf{e}_i}{\sum_{i=1}^{N} \mu_i^c}; \quad \mu_i^c = \begin{cases} 1 \text{ , } y_i = c \\ 0 \text{ , } y_i \neq c \end{cases} \tag{1}$$

where $\mathbf{x}_i$ is a data from class c ($y^i = c$), $\mathbf{e}_i$, its embedding, $\mathbf{c}^c$ is the centroid of the class c and μ_i^c is the membership of the data i to class c. In supervised learning, the membership is binary and provided by the labels.

The standard center loss assumes an Euclidean distance: $\mathcal{D}(\mathbf{e}_i, \mathbf{c}^c) = \|\mathbf{e}_i - \mathbf{c}^c\|_2$; also that choice is conditioned by the necessity to compute the position of the centroids as the (weighted) arithmetic mean for the vectors. The centers are updated in each iteration, based on latest batches using Stochastic Gradient Descent (SGD) derived optimization. Later developments of this method [24,44] sought ways to enforce also large distances between class centroids using the cosine derived distances for $\mathcal{D}()$.

A limitation of these methods is that in the absence of an explicit intervention over the other class centroids, there is an optimum where all data is tightly grouped in a large cluster with centroids overlapped and small distances for each point to its cluster. A second problem is data scaling, as the network could learn some biases that will simply downscale data.

To alleviate such potential behaviors, we propose to use the normalized embedding and to modify the loss by favoring small distance to the belonging class centroid and large distances to other centroids. Thus, large margins are imposed between different classes clusters. Formally a large margin loss , $\mathcal{L}_{\mathcal{M}}$ can be written as:

$$\mathcal{L}_{\mathcal{M}} = \sum_{i=1}^{N} \left(\mathcal{D}\left(\frac{\mathbf{e}_i}{\|\mathbf{e}_i\|_2}, \frac{\mathbf{c}^c}{\|\mathbf{c}^c\|_2} \right) - \frac{1}{C-1} \sum_{j=1, j\neq c}^{C} \mathcal{D}\left(\frac{\mathbf{e}_i}{\|\mathbf{e}_i\|_2}, \frac{\mathbf{c}^j}{\|\mathbf{c}^j\|_2} \right) \right) \tag{2}$$

If the normalized embedding is $\hat{\mathbf{e}}_i = \frac{\mathbf{e}}{\|\mathbf{e}\|_2}$ the loss can be rewritten as:

$$\mathcal{L}_{\mathcal{M}} = \sum_{i=1}^{N} \left(\mathcal{D}\left(\hat{\mathbf{e}_i}, \hat{\mathbf{c}^c} \right) - \frac{1}{C-1} \sum_{j=1, j\neq c}^{C} \mathcal{D}\left(\hat{\mathbf{e}_i}, \hat{\mathbf{c}^j} \right) \right) \tag{3}$$

where C is the number of classes. Normalization in Eq. (3) limits the space, while the subtraction imposes that one instance should be near to its class center and far from the other centers. Again, the centers can be determined after every batch, using Eq.(1), conditioned by an Euclidean choice for the distance.

Such behavior is illustrated in figure 3. The normalization of the data ensures that the loss $\mathcal{L}_{\mathcal{M}}$ is bounded and it prevents numerical instability.

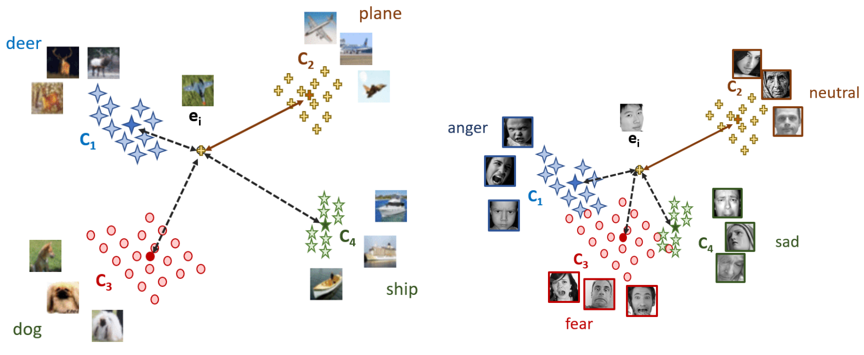
3.2 Distance generalization

The margin loss and the embedding system is inspired by the classical K-means algorithm. While the solution presented in the results section concentrates solely on the Euclidean distance, thus retrieving the classical K-means algorithm, one may extend the algorithm based on non-Euclidean distances [10] and other margin based losses [24,44,9] can be used.

The generalization assumes the following: given N vectors, $\mathbf{b}_i$ are a set of standard basis vectors of the space, a set of membership values μ_i and let us denote by $\mathbf{s}_c = \left(\sum_{i=1}^{N} \mu_i^c \mathbf{b}_i\right) \Big/ \left(\sum_{i=1}^{N} \mu_i^c\right)$. The sought centroids are $\mathbf{c}^c$.

Given a generalized squared distance matrix, $A \in \mathbf{R}^{n\times n}$, with $a_{ij} = \mathcal{D}(\mathbf{e}_i, \mathbf{e}_j)$, then the non-euclidean distance between points and centroids can be developed with respect to a vector $\mathbf{w}$ as:

$$\mathcal{D}(\mathbf{e}_i, \mathbf{c}^c) = \mathbf{e}_i{\cdot}\mathbf{w} = \sum_{j=1}^{N} w_j e_{ij}; \;\; \text{s.t.} \;\; \sum_{j=1}^{N} w_j = 0 \Rightarrow \left(\mathcal{D}(\mathbf{e}_i, \mathbf{c}^c)\right)^2 = -\frac{1}{2}\mathbf{w}^T A \mathbf{w} \tag{4}$$



(a) (b)

Fig. 3. Large margin behavior represented for a 2-dimensional embeding: new data $\mathbf{e}_i$ is from class 2, thus the distance to its class centroid (marked by continuous line) should be made shorter, while the distances to the other classes (marked with dashed lines) should be made longer. (a) Structuring in CIFAR is more sparse, while in (b) FER is with higher density since classes are easier to be confused

In this case the centroids can be computed, given $m_c = \sum_{i=1}^{N} \mu_i^c$, by :

$$\mathbf{c}^c = \frac{1}{m_c} \sum_{j=1}^{N} \mu_j^c \mathbf{e}_j; \quad \text{while } \mathbf{w} = \mathbf{s}_c - \mathbf{b}_j \tag{5}$$

3.3 Self-labeling

Given an unlabeled data $\mathbf{x}_i^u$, its embedding $\mathbf{e}_i^u$, the pseudo-label in one-hot encoding form $\mathbf{y}_i^u = [y_i^1, y_i^2, \dots y_i^C]$ is found based on distances to centroids with a method inspired from Fuzzy C-means algorithm:

$$y_i^c = \frac{1}{\sum_{j=1}^{C} \left(\frac{\|\mathbf{e}_i - \mathbf{c}^c\|_2}{\|\mathbf{e}_i - \mathbf{c}^j\|_2} \right)^2} \tag{6}$$

The process is illustrated in figures 2 (c) and 3, where this time the center position is set and the relative size of distances (arrows) form label probabilities (i.e. class memberships).

3.4 Augmentative processing

To prevent the network to memorize data we regularize the training weight decay (i.e. penalization of the L_2 norm of the model parameters) [25]. Additionally, in the last period, several techniques to improve efficiency have been proposed:

- *Classical data augmentation*: flipping, cropping, Gaussian noise addition, small rotations for face images. Both labeled and unlabeled data has been augmented. Each unlabeled data $\mathbf{x}_b^u$ in a batch is augmented independently N_{aug} times (algorithm 1, line 4).
- *Label guessing*. Berthelot et al. [5] showed that training is more stable if an entire set of N_{aug} variants of unlabeled data have the same labels. In the initial version the labels are retrieved by relative position of the embedding with respect to class centroids as defined by eq. (6). Now, the overall pseudo-label may be found by summing over all N_{aug}:

$$y_i^c = \sum_{k=1}^{N_{aug}} y_k^c = \sum_{k=1}^{N_{aug}} \frac{1}{\sum_{j=1}^{C} \left(\frac{\|\mathbf{e}_i^k - \mathbf{c}^c\|_2}{\|\mathbf{e}_i^k - \mathbf{c}^j\|_2} \right)^2} \tag{7}$$

  where $\mathbf{e}_i^k$ is the embedding of the k-augmentation of the unlabeled data $\mathbf{x}_i^u$.
- *Sharpening* - It has been showed [5] higher non-uniformity of the weights improves the robustness. This is implemented injecting a non linear transform guided by the temperature T hyperparameter, together with normalization from previous step:

$$y_i^c = \frac{y_i^{\frac{1}{T}}}{\psi_j}; \;\; \psi_j = \sum_{c=1}^{C} p_c^{\frac{1}{T}} \tag{8}$$

  One might notice that the combination of sharpening and large margin based on euclidean distance makes the solution close to the soft max procedure.
- *MixUp* [43] - assumes building synthetic new data instances by considering convex combination with random weight of existing data. It is applied on both labeled examples and margin-self-labeled examples:

$$\begin{aligned} \mathbf{x}' &= \lambda \mathbf{x}_i + (1-\lambda)\mathbf{x}_j \\ \mathbf{y}' &= \lambda \mathbf{y}_i + (1-\lambda)\mathbf{y}_j \end{aligned} \tag{9}$$

  where λ is a small random quantity extracted from $Beta(\alpha, \alpha)$ distribution, while α is a hyperparameter. If the second contributor originates in unlabeled data $\mathbf{x}^j = \mathbf{x}_j^u$, λ has to be small such that, the new data is closer to labeled example.
  Considering convex combinations between data points according to the MixUp paradigm, the input space is thoroughly investigated.

3.5 Total Loss

Overall, the network is trained using the loss computed as a weighted sum:

$$\mathcal{L} = \mathcal{L}_{\mathcal{S}} + \lambda_M \mathcal{L}_{\mathcal{M}} = \mathcal{L}_{\mathcal{S}} + \lambda_M (\mathcal{L}_{\mathcal{M}1} + \lambda_u \mathcal{L}_{\mathcal{M}2}) \tag{10}$$

where λ_M and λ_u are weighting hyperparameters, $\mathcal{L}_{\mathcal{S}}$ is the cross entropy decision loss with L2 weight decaying regularization. $\mathcal{L}_{\mathcal{M}1}$ is the large margin loss computed on labeled data, while $\mathcal{L}_{\mathcal{M}2}$ is computed on unlabeled data.

Algorithm 1: The MarginMix algorithm takes as input a batch of labeled data $\mathcal{X}$ and one without labels $\mathcal{U}$ and produces densely sampled input examples $\mathcal{X}'$ respectively self-labeled densely sample examples $\mathcal{U}'$. Self-labeling is based on clustering in the embedding space. The purpose is to adjust the weights of learner ψ

```
Data: : Batch of $b$ labeled instances with embeddings and one–hot labels
      $\mathcal{X} = \{\dots, (\mathbf{x}_i, \mathbf{e}_i, \mathbf{y}_i), \dots\}$, $i = 1 \dots b$, batch of $b$ unlabeled instances
      $\mathcal{X}^u = \{\dots, (\mathbf{x}_i^u), \dots\}$, sharpening temperature $T$ , number of
      augmentations $N_{Aug}$, $\beta$ distribution parameter $\alpha$ for MixUp.
for $b$ = 1 : $N_{batch}$ do
    Compute embeddings for labeled samples $\mathbf{e}_b = \psi(\mathbf{x}_b)$ ;
    Update centroids using eq. (1);
    $\tilde{\mathbf{x}_b}$ = Augment($\mathbf{x}_b$ ) # data augmentation to $\mathbf{x}_b$ ;
    for $k$ = 1 to $N_{Aug}$ do
        $\tilde{\mathbf{x}_k^u}$ = Augment($\mathbf{x}_k^u$ ) # one of the k-th data augmentation to $\mathbf{x}_k^u$ ;
        Self-label by large Margin using eq. (7)
    end
    Compute average, sharpen predictions across all $\tilde{\mathbf{x}_k^u}$ using eq. (7)
end
Collect augmented labeled data: $\tilde{\mathcal{X}} = (\mathbf{x}_b, \mathbf{y}_b); b \in \{1, \dots, N_{batch}\}$ ;
Collect augmented unlabeled data with their self predicted labels:
  $\tilde{\mathcal{X}^u} = (\mathbf{x}_b^u, \mathbf{y}_b^u); b \in \{1, \dots, N_{batch}\}$ ;
Concatenate $\tilde{\mathcal{W}} = (\tilde{\mathcal{X}}, \tilde{\mathcal{X}^u})$;
Use MixUp - eq. (9) for pairs of labeled and new data $\mathcal{X}' = MixUp(\tilde{\mathcal{X}}, \tilde{\mathcal{W}})$
  and pairs of unlabeled and new data $\mathcal{X}_u' = MixUp(\tilde{\mathcal{X}^u}, \tilde{\mathcal{W}})$;
Compute total loss with eq. (10) using $\mathcal{X}'$ $\mathcal{X}_u'$ ;
Update network weights;
```

In the backward propagation, the derivative of the margin loss with respect to the current d-th element of the D-dimensional embedding can be written as:

$$\frac{\partial \mathcal{L}_{\mathcal{M}}}{\partial e_d} = \left(2(\hat{\mathbf{e}}_i - \hat{\mathbf{c}^c}) - \frac{2}{C-1}\sum_{j=1,j\neq c}^{C}\left(\hat{\mathbf{e}}_i - \hat{\mathbf{c}^j}\right)\right)\cdot\frac{\partial \hat{\mathbf{e}}_i}{\partial e_d}; \quad \frac{\partial \hat{\mathbf{e}}_i}{\partial e_d} = \frac{1-\hat{\mathbf{e}_i}^2}{\|\mathbf{e}_i\|_2} \tag{11}$$

3.6 Margin–Mix algorithm

The purpose of the algorithm is to train a DCN using both labeled and unlabeled data. The proposed method is described by Algorithm 1. Intuitively, in a first step, a batch of labeled data passes to collect embedding and update centroid position. Then both labeled and unlabeled data is augmented using the MixUp procedure. Unlabeled is self-annotated by Large margin procedure and the network is asked to provide embeddings that are more discriminative.

3.7 Implementation

The implementation is developed from the tests and procedure described in [28] and [5] respectively[1] . The method has been implemented in Pytorch [29].

For fair comparisons with other SSL methods, we restrict our experiments to the "Wide ResNet-28-2" [41] as architecture. For training, we used SGD solver with a learning rate of 0.001. The margin loss (and subsequent parameter - centroids) has a learning rate of 0.5. We use a cosine scheduler for a learning rate decay from 0.1 to 0.0001. We also fix the weight decay rate to 5e-4. For all experiments, we use a batch size of 64 images. The number of training epochs is dependent on the distribution of the database: for database where the classes contribute uniformly, we used 1024 batches while, overall the model is trained for 1024 epochs.

4 SSL performance. Comparison with state of the art

First we evaluate the proposed algorithm on four standard benchmarks. To asses the proposed method, we perform semi-supervised tasks on four datasets: CIFAR-10 and CIFAR-100 [18], SVHN [26], and STL-10 [7]. The first three are fully annotated, but it is common for the SSL testing to consider as labeled only a subset of the training set and the remainder unlabeled. We emphasize that these databases have the classes perfectly balanced. The last one, was build specifically for SSL, with 5000 labeled images and 100000 unlabeled images. On a fast visual inspection, the unlabeled data is also highly *balanced* between the 10 classes. For the large margin, λ_M was set to 1 and λ_u was set to 0.4.

Achieved results and comparison with prior art [3] can be followed in tables 1 and respectively 2. One may notice that results are very close to the state of the art performance, sometimes even outmatching it. In general the method has similar performance with MixMatch algorithm with which shares several common traits. On direct comparison, for a first view, the MixMatch is lacking weights for margin loss, has fewer parameters, thus may be simple to be tuned; yet the influence of the two parameters was found to be less dramatic and variations around mentioned values (i.e. ±20%) produced similar errors (i.e. ±0.3).

5 Face Expression Recognition with Few Annotations

In this case, the tests are performed on two databases with images in the wild containing various face expressions. The databases are FER+ and RAF-DB.

[1] Code is developed from Pytorch implementation of MixMatch available at `https://github.com/YU1ut/MixMatch-pytorch`. Additional details may be retrieved from the project webpage[2]

[3] Very recently several SSL methods were made public, although not published yet [4],[39],[34] that report improved results. However, they propose augmentation techniques that complement the self-labeling procedure. Beyond very recent publication, they may be used together with the proposed method.

Table 1. Comparative errors (smaller is better) on CIFAR datasets obtained with WideResNet-28-2 . Top row lists the number of examples with labels (over all classes) considered

	CIFAR-10			**CIFAR-100**
Methods/Labels	250	1000	4000	10000
Supervised [38]	–	–	20.26	–
Π-Model [19]	53.02	31.53	17.41	39.19
PseudoLabel [21]	49.98	30.91	16.21	–
MixUp [43]	47.43	25.72	13.15	–
VAT [25]	36.03	18.68	11.05	–
MeanTeacher [36]	47.32	17.32	10.36	–
ICT [38]	–	–	7.66	–
MixMatch [5]	11.80	7.75	6.24	28.88
MarginMix	**10.76**	8.33	**6.17**	29.12

Table 2. Comparative error (smaller is better) on SVHN and STL datasets obtained with WideResNet-28-2. Some results are taken from [28]

	SVHN		**STL**	
Methods/Labels	1000	4000	1000	5000
Supervised [28]	–	12.84	–	–
Π-Model [19]	8.06	5.57	17.41	39.19
VAT [25]	5.63	18.68	11.05	–
MeanTeacher [36,28]	5.65	3.39	10.36	–
ICT [38,28]	3.53	–	7.66	–
MixMatch [5]	3.27	2.89	10.18	5.59
MarginMix	3.35	3.33	**9.85**	5.80

RAF-DB [22] contains facial color images in the wild, which are, often, larger than 300×300. The database is annotated by at least 40 trained annotators per image and divided into 12271 training images and 3078 testing images. It is labeled for the seven basic emotions.

FER+ is derived from FER2013 [14] and contains 28709 training images, 3589 validation (public test) and another 3589 (private) test images, in the wild. FER+ images have 48×48 pixels, are gray-scale and contain only the face. Barsoum et al. [2] noted the high noise in the original labels and performed some ”cleaning”, by removing the images with missing faces and providing labels by aggregating the opinion of 10 non-specialist annotators. Compared to RAF-DB, the images are small, gray and have been annotated less rigorously.

For FER experiments, prior SSL algorithms had trouble solving the task and often converged to a state where only the most populated class was predicted or it simply oscillated without converging. MixMatch often encountered such problems significantly reduce the sharpening temperature from 0.5 to 0.25. Performance for the two databases may be followed in tables 3 and respectively 4.

We report the baseline obtained when training in purely supervised manner but containing MixUp and temporal averaging. In this case the network has been randomly initialized, as it is in the case of SSL methods. For 4000 labeled images considered, a uniform distribution would have required 500 per class, yet three of them do not have so many, so the distribution is already uneven. For SSL methods reported, Mean teacher [36] and MixMatch [5], we have used the public code, tuned as mentioned. For 320 labeled images (i.e. 40 per class) we could not make the Mean Teacher to report multiple classes, but only the dominant one.

As one can notice in these experiments, the proposed method reaches better accuracy than similar solutions by a large margin. We claim that differences originate from two directions.

Firstly, the distribution of labels among classes is uneven. This fact is illustrated in figure 4; there one may see that the most populated class in FER+ database has 5 times more instances than the least populated one. As emphasized in the original MixUp work [43], this technique populates the space near existing examples. Given an uneven distribution, part of the space with sparse classes will become relatively even sparser. Simultaneously, the populated classes will tend to expand (in confidence) in the detriment of sparser ones. Also when parsing unlabeled data, MixMatch will label it more often with the dominant classes value. In our case, the centroid exists, and the relative distance is accepted.

Secondly, fully supervised performance in the case of FER databases is lower than for CIFAR like sets. This suggests that classes are spread in a more intricate manner, which again will favor the most populated classes. Enforcing an intermediate embedding with a large margin, we force the learner to make space for all classes, thus untangling the mixture from the initial data space. A measure of inter-class variance is offered by evaluation of the large margin as defined by eq. (3) in the first iterations of the training procedure, normalized by the number of data instances. The loss measures the quality of clustering: small loss means well defined clusters while large loss means blended clusters. The value is 4× larger in the case of FER+ database when compared to CIFAR-10, although the later has 10 classes compared to 8.

Comparison with softmax/center-loss. When we have performed tests with a solution trained with softmax/center-loss as defined in [40] we have find out that this version often did not converge as on the validation set it entered into oscillating performance or it ended in predicting always a single class. It converged in 50% cases for CIFAR like benchmarks and 20% for expression experiments when it often predicted the most populous class. Intuitively, the standard center loss asks only that instances are close to the class centroid and lets the cross-entropy distance the clusters. Yet, the cross entropy, which is more an angular distance, allows clusters to be close one to another in terms of Euclidean distance, thus on many unlabeled instances produces near uniform

Table 3. Comparative accuracy (larger is better) on FER+ dataset obtained with WideResNet-28-2. Top row lists the number of examples with labels (over all classes) considered. 'nc' stands for not converged

Methods/Labels	320	400	2000	4000	10000	All
Supervised WideResNet	nc	37.92	50.29	56.78	63.56	84.88
Supervised [2]	–	–	–	–	–	84.99
MeanTeacher [36]	–	45.56	50.84	58.28	68.36	–
MixMatch [5]	45.60	50.25	58.35	70.91	71.24	–
MarginMix	**50.76**	**56.75**	**60.83**	**75.18**	**81.25**	85.36

Table 4. Comparative accuracy (larger is better) on RAF–DB dataset obtained with WideResNet-28-2 . Top row lists the number of examples with labels (over all classes) considered

Methods/Labels	320	400	1000	4000	All
Supervised WideResNet	nc	26.75	35.25	55.66	85.58
Supervised [22]	–	–	–	–	84.13
MeanTeacher [36]	nc	28.23	36.53	60.36	–
MixMatch [5]	35.60	42.25	60.37	65.24	–
MarginMix	**40.55**	**45.75**	**66.47**	**70.68**	85.36

class probabilities. It is similar to consider supra-unitary sharpening (we have illustrated the effect of sharpening only up to 0.5, but the trend is obvious). The margin loss imposes that clusters distance themselves.

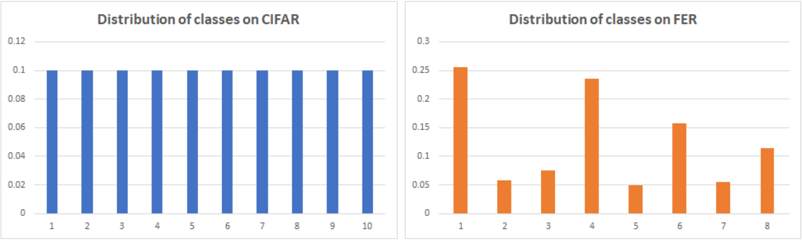
5.1 Parameter Ablation

Our method proved to be more robust in the case of Face Expression Recognition which have much lower inter-class variance. Various versions of the method have been tested on the FER+ database when 2000 examples, equally distributed among classes. The performance is presented in table 5.

The stochastic variance (i.e. variation of accuracy when running the same solution consecutive times) is 0.55. In this case, one may notice that only sharpening may have an impact larger than the stochastic effect. Dramatic decrease is found in the self-labeling if instead of soft probabilities, hard one (based on the nearest centroid) are used; this result is in line with test about sharpening. Otherwise, the solution is robust to slight variations of the parameters.

6 Conclusions

In this paper, we presented MarginMix, a novel framework that combines the capability of deep DCN to produce simultaneously predictions and discriminative embeddings with "the low density separation" principle, while building SSL



(a) (b)

Fig. 4. Distribution of classes on databases from the two categories of experiments: (a) CIFAR and (b) FER+.

Table 5. MarginMix Accuracy on FER dataset when 2000 images have labels that are equally distributed among classes when various versions have been considered

Methods - Parameters	Accuracy
Baseline ($T = 0.25$, $\lambda_M = 1$,$\lambda_u = 0.4$)	60.83
Sharpening $T = 0.5$	55.35
No Sharpening $T = 0$	57.29
$\lambda_M = 0.5$,$\lambda_u = 0.4$	60.44
$\lambda_M = 0.5$,$\lambda_u = 0.4$	60.74
N_{aug}=512 (instead of 1024)	60.68
without parameter EMA	59.85
with nearest centroid	51.87

models. It contains the MixUp paradigm which thoroughly investigates input space by considering convex combinations of the input data. Our proposal structures via embeddings and with the Euclidean distance an intermediate space, in preparation of the final space, where actual prediction takes place.

The experiments have been structured in two categories. The first refers to standard benchmarks such as CIFAR-10, CIFAR-100 and SVHN where a part of the training data is considered as unlabeled and STL-10 which was build specifically for the SSL systems. Here the data is evenly distributed, and the classes are rather easily separable, our method performed on par with previous similar works.

The second category is dedicated to face expression, which we argue that is truly a direction which should benefit from SSL learning since annotation is hard, noisy and costly. In this case, examples from different classes are more similar, and differences are more in details of the image. In this scenario, our proposal outperforms the state-of-the-art methods on all the datasets tested by a significant margin, while also improving the fully-supervised baseline.

References

1. Athiwaratkun, B., Finzi, M., Izmailov, P., Wilson, A.G.: There are many consistent explanations of unlabeled data: Why you should average. In: ICLR (2019) 4
2. Barsoum, E., Zhang, C., Ferrer, C., Zhang, Z.: Training deep networks for facial expression recognition with crowd-sourced label distribution. In: ICMI. pp. 279 – 283 (2016) 4, 11, 13
3. Bartlett, M., Hager, J., Ekman, P., Sejnowski, T.: Measuring facial expressions by computer image analysis. Psychophysiology **36**(2), 253 – 263 (1999) 2
4. Berthelot, D., Carlini, N., Cubuk, E.D., Kurakin, A., Sohn, K., Zhang, H., Raffel, C.: Remixmatch: Semi-supervised learning with distribution alignment and augmentation anchoring. arXiv preprint arXiv:1911.09785 (2019) 10
5. Berthelot, D., Carlini, N., Goodfellow, I., Papernot, N., Oliver, A., Raffel, C.A.: Mixmatch: A holistic approach to semi-supervised learning. In: NIPS. pp. 5050–5060 (2019) 4, 8, 10, 11, 12, 13
6. Chapelle, O., Schölkopf, B., Zien, A.: Semi-Supervised Learning. MIT Press (2006) 1, 3
7. Coates, A., Ng, A., Lee, H.: An analysis of single-layer networks in unsupervised feature learning. In: AISTATS. pp. 215–223 (2011) 10
8. Corneanu, C., Simón, M., Cohn, J., Escalera, S.: Survey on RGB, 3d, thermal, and multimodal approaches for facial expression recognition: History, trends, and affect-related applications. IEEE T. PAMI **38**(8), 1548 – 1568 (2016) 2
9. Deng, J., Guo, J., Xue, N., Zafeiriou, S.: Arcface: Additive angular margin loss for deep face recognition. In: CVPR (2019) 6
10. Dhillon, I.S., Guan, Y., Kulis, B.: Kernel k-means: spectral clustering and normalized cuts. In: ACM–SIGKDD. pp. 551–556 (2004) 6
11. Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., Darrell, T.: Decaf: A deep convolutional activation feature for generic visual recognition. In: Interantional Conference on Machine Learning (2014) 1
12. Ekman, P., Rosenberg, E.: What the Face Reveals: Basic and Applied Studies of Spontaneous Expression Using the FACS. Oxford Scholarship (2005) 2
13. Florea, C., Florea, L., Vertan, C., Badea, M., Racoviteanu, A.: Annealed label transfer for face expression recognition. In: BMVC. p. 12 pp (2019) 5
14. Goodfellow, I., Erhan, D., Carrier, P., Courville, A., Mirza, M., Hamner, B., Cukierski, W., Tang, Y., Thaler, D., et al.: Challenges in representation learning: A report on three machine learning contests. In: ICONIP. pp. 117 – 124 (2013) 11
15. Haeusser, P., Mordvintsev, A., Cremers, D.: Learning by association-a versatile semi-supervised training method for neural networks. In: CVPR. pp. 89–98 (2017) 4
16. Ho-Phuoc, T.: CIFAR10 to compare visual recognition performance between deep neural networks and humans. CoRR **abs/1811.07270** (2018) 2
17. Kingma, D.P., Mohamed, S., Rezende, D.J., Welling, M.: Semi-supervised learning with deep generative models. In: NIPS. pp. 3581–3589 (2014) 3
18. Krizhevsky, A.: Learning multiple layers of features from tiny images. Tech. rep., MIT (2009) 10
19. Laine, S., Aila, T.: Temporal ensembling for semi-supervised learning. In: ICLR (2016) 4, 11
20. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature **521**(7553), 436–444 (2015) 1

21. Lee, D.H.: Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In: ICML Workshops (2013) 3, 11
22. Li, S., Deng, W.: Reliable crowdsourcing and deep locality-preserving learning for unconstrained facial expression recognition. IEEE Trans. on Image Processing **28**(1), 356 – 370 (2019) 4, 11, 13
23. Li, Y., Zeng, J., Shan, S., Chen, X.: Occlusion aware facial expression recognition using cnn with attention mechanism. IEEE Transactions on Image Processing **28**(5), 2439–2450 (2018) 4
24. Liu, W., Wen, Y., Yu, Z., Li, M., Raj, B., Song, L.: Sphereface: Deep hypersphere embedding for face recognition. In: CVPR. pp. 212–220 (2017) 3, 5, 6
25. Miyato, T., Maeda, S.i., Koyama, M., Ishii, S.: Virtual adversarial training: a regularization method for supervised and semi-supervised learning. IEEE T. PAMI **41**(8), 1979–1993 (2018) 4, 7, 11
26. Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., Ng, A.Y.: Learning multiple layers of features from tiny images. Tech. rep., Stanford (2009) 10
27. Odena, A.: Semi-supervised learning with generative adversarial networks. In: ICML Workshop on Data-Efficient Machine Learning (2016) 3
28. Oliver, A., Odena, A., Raffel, C., Cubuk, E.D., Goodfellow, I.J.: Realistic evaluation of deep semi-supervised learning algorithms. In: ICLR (2018) 10, 11
29. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.: Pytorch: An imperative style, high-performance deep learning library. In: NIPS. pp. 8024–8035 (2019) 10
30. Peterson, J., Battleday, R., Griffiths, T., Russakovsky, O.: Human uncertainty makes classification more robust. In: ICCV. pp. 9617–9627 (2019) 2
31. Rasmus, A., Berglund, M., Honkala, M., Valpola, H., Raiko, T.: Semi-supervised learning with ladder networks. In: NIPS. pp. 3546–3554 (2015) 3
32. Ren, Y., Hu, K., Dai, X., Pan, L., Hoi, S.C., Xu, Z.: Semi-supervised deep embedded clustering. Neurocomputing **325**, 121–130 (2019) 3
33. Sariyanidi, E., Gunes, H., Cavallaro, A.: Automatic analysis of facial affect: A survey of registration, representation, and recognition. IEEE T. PAMI **37**(6), 1113 – 1133 (2015) 2
34. Sohn, K., Berthelot, D., Li, C.L., Zhang, Z., Carlini, N., Cubuk, E.D., Kurakin, A., Zhang, H., Raffel, C.: Fixmatch: Simplifying semi-supervised learning with consistency and confidence. arXiv preprint arXiv:2001.07685 (2020) 10
35. Susskind, J., Littlewort, G., Bartlett, M., Movellan, J., Anderson, A.: Human and computer recognition of facial expressions of emotion. Neuropsychologia **45**(1), 152 – 162 (2007) 2
36. Tarvainen, A., Valpola, H.: Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In: NIPS. pp. 1195–1204 (2017) 4, 11, 12, 13
37. Tran, E., Mayhew, M.B., Kim, H., Karande, P., Kaplan, A.D.: Facial expression recognition using a large out-of-context dataset. In: Proc. of IEEE Conf. on Winter Applications on Computer Vision. pp. 52 – 59 (2018) 4
38. Verma, V., Lamb, A., Kannala, J., Bengio, Y., Lopez-Paz, D.: Interpolation consistency training for semi-supervised learning. IJCAI (2019) 4, 11
39. Wang, X., Kihara, D., Luo, J., Qi, G.J.: Enaet: Self-trained ensemble autoencoding transformations for semi-supervised learning. arXiv preprint arXiv:1911.09265 (2019) 10

40. Wen, Y., Zhang, K., Li, Z., Qiao, Y.: A discriminative feature learning approach for deep face recognition. In: ECCV. pp. 499–515 (2016) 3, 5, 12
41. Zagoruyko, S., Komodakis, N.: Wide residual networks. In: BMVC (2016) 5, 10
42. Zeng, J., Shan, S., Chen, X.: Facial expression recognition with inconsistently annotated datasets. In: ECCV. pp. 222–237 (2018) 5
43. Zhang, H., Cisse, M., Dauphin, Y.N., Lopez-Paz, D.: mixup: Beyond empirical risk minimization. In: ICLR (2018) 3, 4, 8, 11, 12
44. Zhang, X., Fang, Z., Wen, Y., Li, Z., Qiao, Y.: Range loss for deep face recognition with long-tailed training data. In: CVPR. pp. 5409–5418 (2017) 3, 5, 6
45. Zhang, Z., Han, J., Deng, J., Xu, X., Ringeval, F., Schuller, B.: Leveraging unlabeled data for emotion recognition with enhanced collaborative semi-supervised learning. IEEE Access **6**, 22196–22209 (2018) 5
46. Zhao, S., Cai, H., Liu, H., Zhang, J., Chen, S.: Feature selection mechanism in CNNs for facial expression recognition. In: BMVC. p. 12 pg. (2018) 4
47. Zheng, Y., Pal, D.K., Savvides, M.: Ring loss: Convex feature normalization for face recognition. In: CVPR. pp. 5089–5097 (2018) 3