Local Learning to Improve Bag of Visual Words Model for Facial Expression Recognition

Radu Tudor Ionescu RADUCU.IONESCU@GMAIL.COM
Department of Computer Science, University of Bucharest, 14 Academiei Street, Bucharest, Romania

Marius Popescu POPESCUNMARIUS@GMAIL.COM
Department of Computer Science, University of Bucharest, 14 Academiei Street, Bucharest, Romania

Cristian Grozea CRISTIAN.GROZEA@BRAINSIGNALS.DE
VISCOM, Fraunhofer FOKUS, Kaiserin-Augusta-Allee 31, 10589 Berlin, Germany

Abstract

In this paper we propose a novel computer vision method for classifying human facial expression from low resolution images. Our method uses the bag of words representation. It extracts dense SIFT descriptors either from the whole image or from a spatial pyramid that divides the image into increasingly fine sub-regions. Then, it represents images as normalized (spatial) presence vectors of visual words from a codebook obtained through clustering image descriptors. Linear kernels are built for several choices of spatial presence vectors, and combined into weighted sums for multiple kernel learning (MKL). For machine learning, the method makes use of multi-class one-versus-all SVM on the MKL kernel computed using this representation, but with an important twist, the learning is local, as opposed to global – in the sense that, for each face with an unknown label, a set of neighbors is selected to build a local classification model, which is eventually used to classify only that particular face.

Empirical results indicate that the use of presence vectors, local learning and spatial information improve recognition performance together by more than 5%. Finally, the proposed model ranked fourth in the Facial Expression Recognition Challenge, with an accuracy of 67.484% on the final test set.

ICML 2013 Workshop on Representation Learning, Atlanta, Georgia, USA, 2013.

1. Introduction

History shows that understanding facial expressions can be challenging and insightful, and a good illustration of this fact is Darwin's book *The Expression of the Emotions in Man and Animals* (Darwin, 1872), which tries to link human movements with emotional states. Facial expression recognition is still challenging for computer vision researchers, and a good illustration of this fact is the The Facial Expression Recognition Challenge (FER) of the ICML 2013 Workshop in Challenges in Representation Learning (WREPL). This paper presents our approach to this competition. Although presented as a competition in *representation learning*, this challenge, as organizers state, "does not explicitly require that entries use representation learning ... [but] this contest will see which methods are the easiest to get quickly working on new data". Our strategy was of the second type, we started from a good but general approach of image classification and adapted it to the particularities of the provided dataset and task.

Among the state of the art models in computer vision are discriminative classifiers using the bag of words (BoW) representation (Zhang et al., 2007; Sivic et al., 2005) and spatial pyramid matching (Lazebnik et al., 2006), generative models (Fei-Fei et al., 2007) or part-based models (Lazebnik et al., 2005). Our approach is based on the BoW model. Such models, which represent an image as a histogram of local features, have demonstrated impressive levels of performance for image categorization (Zhang et al., 2007), image retrieval (Philbin et al., 2007), or related tasks. In BoW models, a vocabulary (or codebook) of visual words is obtained by clustering local image descriptors extracted from images. An image is then represented as a bag of visual words, which is a sparse vector of oc-

currence counts of the visual words in the vocabulary. Next, kernel methods are used to compare such histograms. This is a rather general approach for image categorization, because it does not use any particular characteristics of the image. More precisely, this approach treats images representing faces, objects, or textures in the same manner. Our method developed for the FER Challenge stems from this generic approach. The model had to be adapted for the dataset provided by the organizers. First, histograms of visual words were replaced with normalized presence vectors, to eliminate noise introduced by word frequencies. For facial expression recognition, the presence of a visual word is more important than its frequency. Second, local multiple kernel learning was used to predict class labels of test images, in order to reduce both the image variation and the labeling noise in the resulting training sets.

Preliminary experiments were performed to validate our approach. Empirical results shown that presence vectors improve accuracy by almost 1%, while local learning improves performance by almost $2-3\%$. Several kernel methods were also evaluated in the experiments. The SVM classifier performs better than the Kernel Linear Discriminant Analysis (KLDA) and the Kernel Ridge Regression (KRR). Experiments show that spatial information also helps to improve recognition performance by almost $2-3\%$. Presence vectors that record different spatial information are combined to improve accuracy even further. The method we propose here was fairly successful, it ranked fourth in the FER Challenge, with an accuracy of 67.484% on the final (private) test.

The paper is organized as follows. Section 2 presents the learning framework used for image retrieval, image categorization and related tasks. The local learning approach is presented in section 3. Experiments conducted on the Facial Expression Recognition Challenge dataset are presented in section 4. Finally, the conclusions are drawn in section 5.

2. Learning Model

In computer vision, the BoW model can be applied to image classification and related tasks, by treating image descriptors as words. A bag of visual words is a sparse vector of occurrence counts of a vocabulary of local image features. This representation can also be described as a histogram of visual words. The vocabulary is usually obtained by vector quantizing image features into visual words. For facial expression recognition, the presence of a visual word is more important than its frequency. For example, it should be enough to detect the presence of a cheek dimple to recognize a smiling face. Thus, instead of recording occurrence counts in a histogram, it is enough to record visual words presence in a presence vector. The presence vector is normalized not to favor faces with more visual words.

The proposed framework has two stages, one for training and one for testing. Each stage is divided in two different steps. The first step in both stages is for feature detection and representation. The second step is to train a kernel method (in the training stage) in order to predict the class label of a new image (in the testing stage). For each test image, a local classification problem is constructed by selecting only the nearest neighbors from the kernel feature space. Local learning is further described in section 3. The entire process, that involves both training and testing stages, is summarized in figure 1.

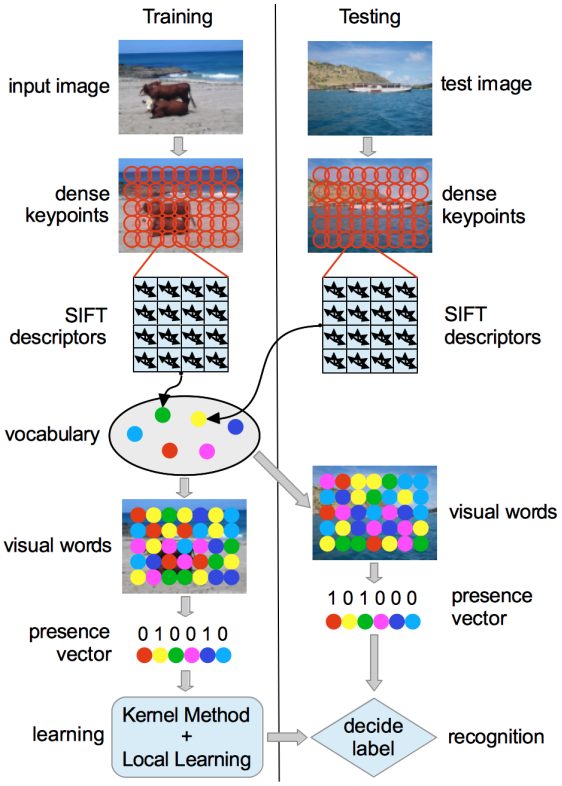

Figure 1. The BoW learning model for object class recognition. The feature vector consists of SIFT features computed on a regular grid across the image (dense SIFT) and vector quantized into visual words. The presence of each visual word is then recorded in a presence vector. Normalized presence vectors enter the training stage. Learning is done by a kernel method.

The feature detection and representation step in the

training stage works as follows. Features are detected using a regular grid across the input image. Although most of the state of the art approaches are based on sparse descriptors, others have used dense descriptors (Fei-Fei & Perona, 2005; Winn et al., 2005). The approach known as dense SIFT (Dalal & Triggs, 2005; Bosch et al., 2007) was used, which computes a SIFT feature (Lowe, 1999) at each interest point. Next, SIFT descriptors are vector quantized into visual words and a codebook of visual words is obtained. The vector quantization process is done by k-means clustering (Leung & Malik, 2001), and visual words are stored in a randomized forest of k-d trees (Philbin et al., 2007) to reduce search cost. It is interesting to note that our method is semi-supervised in the sense that it can leverage the SIFT descriptors from the unlabeled test set by including them in the clustering process as well, together with those from the training dataset. For increased test sets this has led to a better representation of the faces manifold and to increased classification performance. The presence of each visual word is recorded in a presence vector which represents the final feature vector for the image. Normalized presence vectors of visual words can now enter the learning step. Figure 2 presents a sample of 30 SIFT descriptors extracted from two images of the FER dataset.

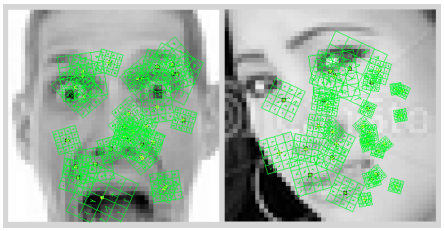

Figure 2. An example of SIFT features extracted from two images representing distinct emotions: fear (left) and disgust (right).

Feature detection and representation is similar during the testing stage. The presence vector of visual words that represents the test image is compared with the training presence vectors, through the implicit distance defined by the kernel, to select a number of nearest neighbors. For a certain test image, only its nearest neighbors actually enter the learning step. In other words, a local recognition problem is built for each test image. A kernel method is employed to learn the local recognition problem and finally predict a class label for the test image. Classifiers such as the SVM, the KLDA or the KRR are good choices to perform the local learning task.

The model described so far ignores spatial relationships between image features. Despite ignoring spatial information, visual words showed a high discriminative power and have been used for region or image level classification (Csurka et al., 2004; Fei-Fei & Perona, 2005; Zhang et al., 2007). A good way to improve performance is to include spatial information. This can be done by dividing the image into spatial bins. The presence of each visual word is then recorded in a presence vector for each bin. The final feature vector for the image is a concatenation of these presence vectors. A more robust approach is to use a spatial pyramid (Lazebnik et al., 2006). The spatial pyramid is usually obtained by dividing the image into increasingly fine sub-regions (bins) and computing histograms of visual words found inside each bin. Our framework makes use of this spatial information by computing a spatial pyramid from presence vectors. It is reasonable to think that dividing an image representing a face into bins is a good choice, since most features, such as the contraction of the muscles at the corner of the eyes, are only visible in a certain region of the face. In other words, one does not expect to find raised eyebrows on the cheek, or cheek dimples on the forehead.

3. Local Learning

The development of unconventional (or nonstandard) learning formulations and non-inductive types of inference was studied in (Vapnik, 2006). The author argues in favor of introducing and developing unconventional learning methods, as an alternative to algorithmic improvements of existing learning methods. This view is consistent with the main principle of VC theory (Vapnik & Chervonenkis, 1971), suggesting that one should always use direct learning formulations for finite sample estimation problems, rather than more general settings (such as density estimation). In (Bottou & Vapnik, 1992) the idea of local algorithms for pattern recognition was used, where local linear rules (instead of local constant rules) and VC bounds (Vapnik & Chervonenkis, 1971) (instead of the distance to the k-th nearest neighbor) were utilized. The local linear rules demonstrated an improvement in accuracy on the popular MNIST dataset.

Local learning methods attempt to locally adjust the performance of the training system to the properties of the training set in each area of the input space. A simple local learning algorithm works as follows: for each testing example, select a few training examples located in the vicinity of the testing example, train a classifier with only these few examples and apply the resulting classifier to the testing example.

The learning step of our framework is based on a local learning algorithm that uses the presence kernel

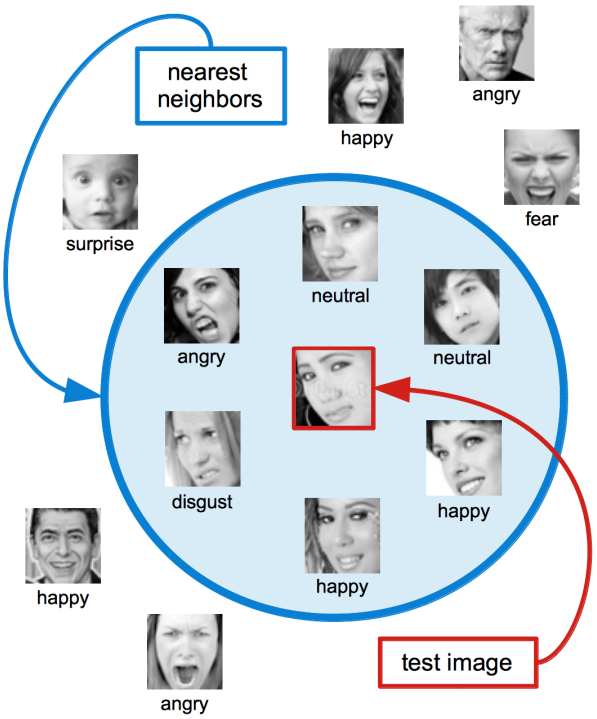

Figure 3. The six nearest neighbors selected with the presence kernel from the vicinity of the test image are visually more similar than the other six image randomly selected from the training set. Despite this fact, the nearest neighbors do not adequately indicate the test label. Thus, a learning method needs to be trained on the selected neighbors to accurately predict the label of the test image.

to select nearest neighbors in the vicinity of a test image. Local learning has a few advantages over standard learning methods. Firstly, it divides a hard classification problem into more simple sub-problems. Secondly, it reduces the variety of images in the training set, by selecting images that are most similar to the test one. Thirdly, it improves accuracy for datasets affected by labeling noise. Considering all these advantages, a local learning algorithm is indeed suitable for the FER dataset.

Figure 3 shows that the nearest neighbors selected from the vicinity of a particular test image are visually more relevant than a random selection of training images. It also gives a hint that local learning should perform better than a standard learning formulation.

4. Experiments

4.1. Dataset Description

The dataset of the FER Challenge consists of 48×48 pixel gray-scale images of faces representing 7 categories of facial expressions: anger, disgust, fear, happiness, sadness, surprise, and neutral. There are 28709 examples for training, 3589 examples for testing, and another 3589 examples for private testing. The task is to categorize each face based on the emotion shown in the facial expression in to one of 7 categories. Images were collected from the web using a semi-automatic procedure. Therefore, the dataset may contain images that do not represent faces. Another issue is that the training data may also contain labeling noise, meaning that the labels of some faces do not indicate the right facial expression.

4.2. Implementation

The framework described in section 2 is used for facial expression recognition. Details about the implementation of the model are given next. In the feature detection and representation step, a variant of dense SIFT descriptors extracted at multiple scales, called PHOW features (Bosch et al., 2007), are used. the PHOW features were extracted using a grid step of 1 pixel at scales ranging from 2 to 8 pixels. These features are extracted from the entire FER dataset. A random sample of 240000 features is selected to compute the vocabulary. The number of visual words used in the experiments ranges from 5000 to 20000. A slight improvement in accuracy can be observed when the vocabulary dimension grows. Both histograms of visual words and presence vectors were tested. Empirical results show that presence vectors are able to improve accuracy, by eliminating some of the noise encoded by the histogram representation.

Different spatial presence vectors were combined to obtain several spatial pyramid versions. Images were divided into 2×2 bins, 3×3 bins, and 4×4 bins to obtain spatial presence vectors. The basic presence vectors are also used in the computation of spatial pyramids.

The kernel trick is implied to obtain spatial pyramids. Kernel matrices are computed for each (spatial) presence vector representation. Then, kernel matrices are summed up to obtain the kernel matrix of a certain spatial pyramid. Some of the proposed models are based on a weighted sum of kernel matrices, with weights adjusted to match the accuracy level of each (spatial) presence vector. Summing up kernel matrices is equivalent to presence vector concatenation. But presence vectors are actually high-dimensional sparse vectors, and the concatenation of such vectors is not a viable solution in terms of space and time.

Several state of the art kernel methods are used to perform the local learning tasks, namely the SVM, the KLDA and the KRR. Results show that the SVM per-

Table 1. Accuracy levels for several models obtained on the validation, test, and private test sets.

Model	Neighbours	Validation	Global SVM	Test	Private
8000 1×1	1000	59.07%			
8000 2×2	1000	62.22%			
8000 3×3	1000	62.27%			
8000 4×4	1000	62.93%			
8000 sum	1000	63.27%		65.73%	66.73%
17000 1×1	1000	60.86%			
14000 2×2	1000	62.69%			
11000 3×3	1000	62.36%			
mix1	1000	63.03%		65.89%	
mix2	1000	63.74%		66.42%	
mix3	1000	63.61%		66.65%	
Models re-built using the new test data					
mix1	1000	62.91%	59.35%	66.59%	66.73%
mix2	1000	63.99%	60.95%	67.01%	67.31%
mix3	1000	64.30%	62.27%	67.32%	67.48%
mix3	3000	64.23%	62.27%		
20k	500	63.59%	61.82%		
20k	1000	63.90%	61.82%		
20k	1500	64.10%	61.82%	66.53%	66.98%
20k	3000	64.45%	61.82%		
20k	5000	64.35%	61.82%		

forms slightly better than the KLDA, and much better than the KRR. The number of nearest neighbors selected to enter the local learning phase for each test image is $1,000$. However, an experiment is conducted to show the accuracy level as the number of nearest neighbors varies. The regularization parameter C of the SVM was set to 1000. This choice is motivated by the fact that the dataset is separable since there is a small number of training examples (1000 neighbors), in a high-dimensional feature space. Thus, the best working SVM is a hard margin SVM that can be obtained setting the C parameter of the SVM to a high value (Shawe-Taylor & Cristianini, 2004).

4.3. Parameter Tuning and Results

For parameter tuning and validation, the training set was randomly split in two thirds kept for training and one third for validation. Preliminary experiments using different models were performed on the validation set to asses the performance levels of the kernel methods. Obtained results pointed that the SVM is about $1-2\%$ better than the KRR, and about 0.5% better than KLDA. We also obtained that presence vectors are $0.5-1\%$ better than histograms. In the experiments presented in Table 1 only results obtained with various SVM models based on presence vectors are included. The model names of the form "8000 3×3" specify the size of the vocabulary, followed by the size of the grid used to partition the image into spatial bins. The kernel of "8000 SUM" is the mean of the kernels 8000 1×1 to 4×4. It has performed on the validation set better than each of its terms. The kernel "MIX3" is the mean of 17000 1×1, 14000 2×2, 11000 3×3, and 8000 4×4. Again, it has performed better than each of its terms. The kernel "20K" is a variant of "MIX3" built on even larger vocabularies, as is the mean of 20000 1×1, 20000 2×2, 12000 3×3, and 8000 4×4. It did not perform better than "MIX3" neither on the validation dataset nor on the actual test set. The kernel "MIX1" is the weighted mean of 7000 1×1, 7000 2×2, 7000 3×3, and 5000 4×4, with the weights 0.1, 0.2, 0.4, 0.3, respectively. The kernel "MIX2" is the weighted mean of 11000 1×1, 9000 2×2, 7000 3×3, and 5000 4×4, with the weights 0.2, 0.3, 0.3, 0.2, respectively.

In Table 1 one can observe that the performance of the global one-versus-all SVM is at least 2% lower than that obtained with the local learning based on one-vs-all SVM with the same parameters. Another interesting behavior that can seen in this table is the effect on accuracy of dividing the image area into spatial bins: the accuracy increases as the image is divided in finer sub-regions. This table also shows the effect of the number of neighbors, another parameter we have adjusted. Although we did not submit anything computed with more than 1500 neighbors, it may well be that using 3000 neighbors could have led to somewhat higher scores. Our tuning was limited both by the

amount of RAM available in our machines (24 GB for the largest one) and by the speed of the CPUs (4-core Xeon E5540 at 2.53 GHz in the fastest one). Test cycles took therefore up to 9 hours.

5. Conclusion and Further Work

This paper presented a bag of visual words model adapted for the FER Challenge dataset. Histograms of visual words were replaced with normalized presence vectors, then local learning was used to predict class labels of test images. The proposed model also includes spatial information in the form of spatial pyramids computed from presence vectors.

Experiments were performed to validate the proposed model. By reserving one third of the training dataset as validation set we have been able to tune our method's parameters without over-fitting, as can be seen in Table 1. Empirical results showed that the proposed model has an almost 5% improvement in accuracy over a classical bag-of-words model. Also, using multiple kernel learning (with sum or weighted sum kernels) led to accuracy levels higher than that of the individual kernels involved. Finally, the proposed model ranked fourth in the FER Challenge, with an accuracy of 67.484% on the final test. A different approach to local learning, that of clustering train images to divide the learning task on each cluster separately, can be studied in future work.

References

Bosch, Anna, Zisserman, Andrew, and Munoz, Xavier. Image Classification using Random Forests and Ferns. *Proceedings of ICCV*, pp. 1–8, 2007.

Bottou, Leon and Vapnik, Vladimir. Local Learning Algorithms. *Neural Computation*, 4:888–900, 1992.

Csurka, Gabriella, Dance, Christopher R., Fan, Lixin, Willamowski, Jutta, and Bray, Cdric. Visual categorization with bags of keypoints. *In Workshop on Statistical Learning in Computer Vision, ECCV*, pp. 1–22, 2004.

Dalal, Navneet and Triggs, Bill. Histograms of Oriented Gradients for Human Detection. *Proceedings of CVPR*, 1:886–893, 2005.

Darwin, Charles. *The Expression of the Emotions in Man and Animals*. London: John Murray (1st edition), 1872.

Fei-Fei, Li and Perona, Pietro. A Bayesian Hierarchical Model for Learning Natural Scene Categories. *Proceedings of CVPR*, 2:524–531, 2005.

Fei-Fei, Li, Fergus, Rob, and Perona, Pietro. Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories. *CVIU*, 106(1):59–70, April 2007.

Lazebnik, Svetlana, Schmid, Cordelia, and Ponce, Jean. A Maximum Entropy Framework for Part-Based Texture and Object Recognition. *Proceedings of ICCV*, 1:832–838, 2005.

Lazebnik, Svetlana, Schmid, Cordelia, and Ponce, Jean. Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories. *Proceedings of CVPR*, 2:2169–2178, 2006.

Leung, Thomas and Malik, Jitendra. Representing and Recognizing the Visual Appearance of Materials using Three-dimensional Textons. *IJCV*, 43(1):29–44, June 2001.

Lowe, David G. Object Recognition from Local Scale-Invariant Features. *Proceedings of ICCV*, 2:1150–1157, 1999.

Philbin, James, Chum, Ondrej, Isard, Michael, Sivic, Josef, and Zisserman, Andrew. Object retrieval with large vocabularies and fast spatial matching. *Proceedings of CVPR*, pp. 1–8, 2007.

Shawe-Taylor, John and Cristianini, Nello. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004. ISBN 978-0-521-81397-6.

Sivic, Josef, Russell, Bryan C., Efros, Alexei A., Zisserman, Andrew, and Freeman, William T. Discovering Objects and their Localization in Images. *Proceedings of ICCV*, pp. 370–377, 2005.

Vapnik, Vladimir. Estimation of dependencies based on empirical data (Information Science and Statistics). *SpringerVerlag*, 2nd edition, 2006.

Vapnik, Vladimir and Chervonenkis, Alexey. On the Uniform Convergence of Relative Frequencies of Events to Their Probabilities. *Theory of Probability and its Applications*, 16(2):264–280, 1971. doi: 10.1137/1116025.

Winn, J., Criminisi, A., and Minka, T. Object Categorization by Learned Universal Visual Dictionary. *Proceedings of ICCV*, 2:1800–1807, 2005.

Zhang, Jian, Marszalek, Marcin, Lazebnik, Svetlana, and Schmid, Cordelia. Local Features and Kernels for Classification of Texture and Object Categories: A Comprehensive Study. *IJCV*, 73(2):213–238, June 2007.