Simple Method for High-Performance Digit Recognition Based on Sparse Coding

Kai Labusch
Institute for Neuro- and Bioinformatics
University of Lübeck
D-23538 Lübeck
labusch@inb.uni-luebeck.de
Erhardt Barth
Institute for Neuro- and Bioinformatics
University of Lübeck
D-23538 Lübeck
barth@inb.uni-luebeck.de
Thomas Martinetz (Senior Member IEEE)
Institute for Neuro- and Bioinformatics
University of Lübeck
D-23538 Lübeck
martinetz@informatik.uni-luebeck.de
keywords: digit recognition, sparse coding, feature extraction, SVM

***Abstract*— We propose a method of feature extraction for digit recognition that is inspired by vision research: a sparse-coding strategy and a local maximum operation. We show that our method, despite its simplicity, yields state-of-the-art classification results on a highly competitive digit-recognition benchmark. We first employ the unsupervised Sparsenet algorithm to learn a basis for representing patches of handwritten digit images. We then use this basis to extract local coefficients. In a second step, we apply a local maximum operation in order to implement local shift invariance. Finally, we train a Support-Vector-Machine on the resulting feature vectors and obtain state-of-the-art classification performance in the digit recognition task defined by the MNIST benchmark. We compare the different classification performances obtained with sparse coding, Gabor wavelets, and principle component analysis. We conclude that the learning of a sparse representation of local image patches combined with a local maximum operation for feature extraction can significantly improve recognition performance.**

I. Introduction

A common approach to solve a visual pattern recognition problem such as digit recognition is to divide the solution into the two parts of feature extraction and classification. A preliminary preprocessing step may be regarded as part of the feature extraction. In general it is not clear how learning methods can be used to obtain features that are optimal for a given task. Hence, methods for feature extraction are often selected according to heuristic principles based on experience and problem-specific knowledge.

In order to tackle the problem of object recognition one can match the unknown object against some reference [1]. The matching result is then used to perform the recognition task. Though matching methods perform well in a number of tasks they are often complex and associated with a number of difficulties. For example they require to solve the computationally expensive correspondence problem [1].

Principal-Component-Analysis (PCA) [2] or Gabor wavelets [3] belong to another group of feature extraction methods. These methods do not perform an explicit matching but provide a new representation of the data. It is assumed that the new representation is advantageous with respect to recognition tasks. Based on the new representation a classifier is trained. However, it is not clear that the new representation is advantageous, since it is not guaranteed that it provides invariances adapted to the recognition problem.

In vision research the *optimal-coding hypothesis* proposes that the human visual system has adapted to the statistical properties of natural images [4], [5], [6], [7]. Different statistical models of image synthesis have been proposed, such as the independent component analysis (ICA) [8], [9] and Sparse Coding [10], [11]. These models have been successfully employed to mimic properties of simple cells in the primary visual cortex [12], [13]. In addition, advanced models of the visual system postulate that the output of simple cells is fed to a class of neurons which exhibit a maximum-selection behaviour [14], [15].

Is there a convenient, i.e, simple, way for practitioners to employ findings from vision research to actually solve a technical problem? An example of a complex multi-stage recognition system being inspired by the neurosciences can be found in [16].

Though the problem of digit recognition has been intensively investigated [17], [18], [19], the improvement of digit recognition performance is still a major issue in a number of

industrial applications, e.g. parcel sorting. We here describe a novel method for digit recognition that employs biologically inspired principles, i.e. a learned sparse representation and a local maximum operation. We evaluate the performance of our method for handwritten-digit recognition on the MNIST data set, being a very competitive benchmark for which many different methods already have been evaluated [20]. In the same framework, we compare our results using a sparse code that was learned by the Sparsenet algorithm [13] against those obtained with more common feature-extraction methods such as PCA and Gabor wavelets. The goal of this paper is to show that our way of combining unsupervised learning of a sparse code with a local maximum operation leads to features that allow for highly competitive recognition performance without the need for heuristic features and problem specific knowledge.

The paper is structured as follows: In section II we first describe how to obtain the preprocessed image patch vectors the feature extraction method operates on. How a new representation of the image patch vectors in terms of coefficients of basis functions can be learned from the data is then explained in section II-A. In section II-B Gabor wavelets are defined. In section II-C we describe how to obtain a set of coefficient images from the preprocessed image patch vectors. A local minmax-operation that is used to obtain the final feature vectors for the classifiers is described in section II-D. The experiments, i.e. learning of the basis functions and training SVM-classifiers on the final feature vectors can be found in section III. The results together with references to related work are presented in section IV followed by a discussion in section V.

II. Feature extraction

Often vision systems that are inspired by the neurosciences tend to become quite complex. We here propose a simple two-stage model of feature extraction. Firstly, a number of coefficient images are computed by using a learned basis, and secondly, a local maximum operation is performed (see figure 1). We do not operate on the raw pixel values but apply a preprocessing to the image patches. This is required for the PCA and to improve the convergence of the Sparsenet algorithm. In the following, I denotes an entire image, whereas for an odd number N $\widehat{P}(x,y)$ denotes an vector containing all N^2 pixel values of an image patch of size $N \times N$ centered at position (x,y) arranged in an appropriate scheme. For a $\widehat{P}(x,y)$ the mean value of the pixel entries of this vector is denoted by $\overline{\widehat{P}(x,y)}$. Removing the mean pixel value of the patch vector leads to

$$\widetilde{P}(x,y) = \widehat{P}(x,y) - \overline{\widehat{P}(x,y)} \; . \tag{1}$$

With $\overline{P}$ we refer to the mean of a large number of such $\widetilde{P}(x,y)$ that were obtained from many training images with $\widetilde{P}(x,y)$ placed at random positions. By removing $\overline{P}$ we finally obtain the centered vectors $P(x,y)$:

$$P(x,y) = \widetilde{P}(x,y) - \overline{P} \; . \tag{2}$$

The first stage of the feature extraction is based on a set of basis functions $\vec{w}_j$ of size N^2 which are applied to each patch vector $P(x,y)$ of a given image I.

A. Learning the basis functions

We want to learn basis functions that represent the centered patch vectors $P(x,y)$ such that these can be reconstructed from the basis functions. This leads to the interpretation of PCA and sparse coding as generative image-patch models. The underlying model defines which basis functions are selected and either postulates perfect reconstruction or allows for reconstruction errors (noise). Applying PCA and sparse coding to a large number of random image-patch vectors can be interpreted as parameter determination for the underlying model. In case of PCA and sparse coding the underlying models are closely related. In both cases, a patch vector $P(x,y)$ is obtained from a linear combination of the basis functions $\vec{w}_j$ with coefficients $\vec{a}$ that lead to the features we are going to use for classification. An additive error term $\vec{\epsilon}$ may be allowed, which corresponds to assuming a certain amount of noise. K, the number of basis functions, is a free model parameter.

$$P(x,y) = \sum_{j=1}^{K} \vec{w}_j a_j + \vec{\epsilon} = W\vec{a} + \vec{\epsilon} \; . \tag{3}$$

1) PCA: Equation (3) can be seen as a generative model using principal components. If the number of $\vec{w}_j$, i.e. K, equals N^2, $\|\vec{\epsilon}\| = 0$ is assumed. The a_j are pairwise uncorrelated, i.e. the $\vec{w}_j$ form an orthogonal basis of the $P(x,y)$. The $\vec{w}$ can be obtained as the eigenvectors of the covariance matrix $C = E\left(P(x,y)P(x,y)^T\right)$ of the distribution of the $P(x,y)$. (here $E()$ denotes the expectation which is approximated by averaging in practice)

2) Sparse Coding: Within the Sparse Coding approach, equation (3) postulates an image-patch generation model where the a_j stem from sparse (leptocurtic) distributions. Hence, the primary goal of sparse coding is the maximization of the sparsity of the coefficients a_j. The reconstruction error (noise) is assumed to be Gaussian. The model now allows for balancing the reconstruction error $\|\vec{\epsilon}\|$ against the sparsity of the coefficients a_j. There are different sparse-coding approaches available, see for example [13], [21], [22]. Here we use the Sparsenet algorithm [13] which solves the following optimization problem:

$$\min_{W} E\left(\min_{\vec{a}} \left(\|P(x,y) - W\vec{a}\| + \lambda S(\vec{a})\right)\right) \; . \tag{4}$$

The additive regularization term $S(\vec{a})$ favors model parameters W that lead to sparse coefficients $\vec{a}$. The parameter λ allows to balance the reconstruction error $\vec{\epsilon}$ against the sparseness of the coefficients.

B. Gabor wavelets as basis functions

Many models of the visual system employ Gabor wavelets to model the receptive fields of simple cells in area V1. We include Gabor wavelets in our experiments to compare the performance of this well-known feature extraction method against our Sparsenet approach, a V1-like representation that was obtained by unsupervised learning from the data.

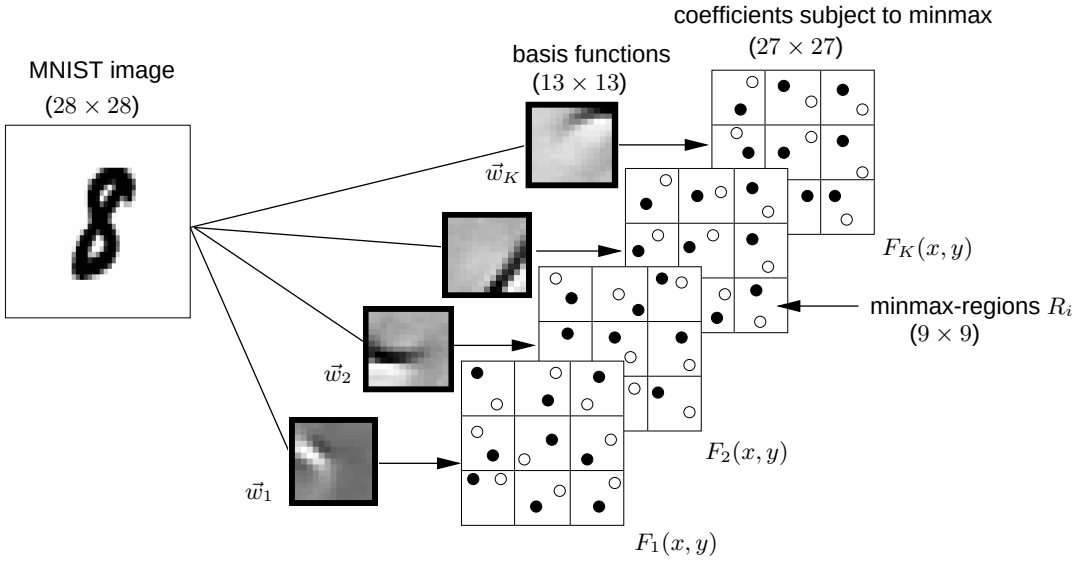

Fig. 1. A schematic view of the feature extraction method that is proposed in this paper. First, the coefficients are extracted either by convolution of the input image with the basis functions or by minimization of the objective function of the Sparsenet algorithm. Coefficients are computed for each pixel and each basis function. Second, each coefficient image is divided into regular non-overlapping regions, and for each region the minimal and maximal entry (indicated by the circles) are selected. These extrema are the features for the subsequent classification step.

A two-dimensional Gabor wavelet $\vec{w}_j$ is determined by its orientation α_j, wavelength λ_j, bandwidth b_j, phase ϕ_j, and center $\vec{c}_j$:

$$\vec{w}_j = e^{-\frac{\|R_{\alpha_j}(\vec{x}-\vec{c}_j)\|}{2*\left(\frac{\lambda_j}{\pi}\sqrt{\left(\frac{log(2)}{2}\right)}\frac{2^{b_j}+1}{2^{b_j}-1}\right)^2}} * cos\left(\frac{2*\pi\left(R_{\alpha_j}(\vec{x}-\vec{c}_j)\right)}{\lambda_j}+\phi_j\pi\right). \quad (5)$$

With R_{α_j} we refer to a two-dimensional rotation of degree α_j.

C. Obtaining the coefficient images

We now describe how the system works with given basis functions $\vec{w}_j$ that either were obtained from the training data by the Sparsenet algorithm (section II-A.2) or PCA (section II-A.1), or that were given by the Gabor wavelets (section II-B).

The coefficient images F_j contain the coefficients a_j of the basis functions $\vec{w}_j$ for all patches of the input image. In order to obtain coefficient images that have the same size as the initial image I, it is is required to enlarge I by setting the value of pixels outside the image to a fixed value. The method of obtaining these coefficient images differs depending on the method that was used to learn the corresponding basis functions. In case of PCA and Gabor wavelets the coefficient images are obtained by a convolution operation, i.e. for each centered patch vector $P(x,y)$ of the input image I we compute

$$F_j(x,y) = \frac{P(x,y)^T\vec{w}_j}{\|\vec{w}_j\|}, j = 1,\ldots,K. \quad (6)$$

The Sparsenet coefficients are obtained by minimizing the objective function that was used to learn the basis:

$$\vec{F}(x,y) = (F_1(x,y),\ldots,F_K(x,y)) \quad (7)$$

$$= \arg\min_{\vec{a}}(\|P(x,y)-\sum_{j=1}^{K}\vec{w}_j a_j\| + \lambda S(\vec{a})) \quad (8)$$

As in the Sparsenet algorithm the minimization of the objective function is performed via gradient descent.

D. Local maximum operation

The basis functions represent relevant attributes of the image patches they were learned from since it is possible to reconstruct the image patches by a linear combination of only few basis functions. In case of sparse linear combinations only few basis functions are required to "explain" a certain image patch. A certain attribute is present at a certain location if the coefficient of the basis function that represents the attribute has a large absolute value. The absolute value of the coefficient can be interpreted as the similarity of the image at a certain position with respect to the basis function. Due to the nature of the digit images some uncertainty with respect to the exact localisation of important attributes in the image remains. Hence, we would like to allow for some spatial uncertainty to obtain local shift invariance. Assuming that those basis functions that are highly expressed, i.e. have large absolute coefficient values, are important, computing the maximum, as well as the minimum, in a local region localises the important attributes and achieves the desired local shift invariance. The attributes are considered independent, i.e. the positions where the minimum and maximum values are obtained differ for each basis function.

We implement this principle in a very simple way. Thereby we divide the input image into a set of regular, non-overlapping regions $R_i, i = 1,...,M^2$ and take as local

features the maximum and minimum of each region with respect to each coefficient image(see Figure 1):

$$F_j^{max}(R_i) = \max_{x,y \in R_i} F_j(x,y) \ . \tag{9}$$

$$F_j^{min}(R_i) = \min_{x,y \in R_i} F_j(x,y) \ . \tag{10}$$

There is some experimental evidence that the behaviour of complex cells in the visual cortex can be described by a local maximum operation [15], and that human observers might account for position uncertainty by using the same principle [23]. The principle has been used recently in technical applications [16].

The final feature vector (that is given as input to the classifier) of each input image consists of the maximum and minimum values of all regions with respect to all coefficient images:

$$\begin{aligned} f_I \quad = \quad & (F_1^{max}(R_1), \ldots, F_1^{max}(R_{M^2}), \ldots, \\ & F_K^{max}(R_1), \ldots, F_K^{max}(R_{M^2}), \\ & F_1^{min}(R_1), \ldots, F_1^{min}(R_{M^2}), \ldots, \\ & F_K^{min}(R_1), \ldots, F_K^{min}(R_{M^2})) \ . \end{aligned} \tag{11}$$

Note, that in general the final feature vector f_I is not sparse. The coefficients of each patch are sparse but due to the maximum and minimum operation large positive or negative values are accumulated in the final feature vector.

III. Experiments

We test Sparsenet, PCA, and Gabor basis functions on the well-known MNIST benchmark of handwritten digit images. It consists of 60000 training and 10000 test images of handwritten digits of size 28×28 pixels.

The PCA and Sparsenet basis functions $\vec{w}_j$ are obtained by determining the parameters of the underlying image patch generation model with respect to the image-patch vectors $P(x,y)$. The patch vectors are extracted at random positions from randomly chosen training images. The noise level parameter (λ) of the Sparsenet algorithm is chosen such that the best mean validation error is obtained. Additionally, as mentioned before, the performance of a simple set of Gabor wavelets as basis functions is evaluated. We do not optimize the parameters of the Gabors explicitly but take the best parameters out of a limited set we experimented with.

We obtain the features for the classifier as described in section II-C and II-D. The size of the basis functions is 13×13 pixels. Though the number of basis functions may be optimized with respect to recognition performance, for simplicity we choose as many basis functions as the number of pixels in the image patches considered, i.e. 169. The set of Gabor wavelets that provided the best results on the digit benchmark in our experiments consists of 160 filters, which is close to the number of basis functions used in the Sparsenet and PCA setting (We also tried larger Gabor sets). We use a layout of 9 minmax-operator regions of size 9×9 pixels as shown in Figure (1). Since the size of the coefficient images F_j equals the size of the input images, we do not select entries from the bottom row and the last column of the coefficient images, which corresponds to dropping the last row and column of each digit image.

As classifier we use a standard 2-norm soft margin Support-Vector-Machine (SVM) with Gaussian kernels [24], [25]. In order to train the SVM the SoftDoubleMinOver learning algorithm which was introduced in [26] is used. We normalize the training data such that the mean norm of the feature vectors f_I is set to one. The hyperparameters of the SVM are optimised using a validation scheme where seven realisations of test and training data are used. In each realisation the training and test set are disjoint and consist of 10000 samples that were randomly chosen from the entire training set. We take the hyperparameters providing the best mean classification error on the validation test sets in a grid search over the trade-off parameter C and the Gaussian kernel parameter γ. The search uses a logarithmic grid and proceeds recursively from a coarse meshed grid to a small meshed grid. The start range was $[1, 10^4]$ for C and $[1, 10^3]$ for the parameter γ of the gaussian kernel $K(x,z) = exp(-\gamma \|x-z\|^2)$. The best hyperparameters are shown in table II. Using the best hyperparameters, the final classifier is trained on the entire set of feature vectors of all training samples.

	γ	C
raw data	21.5	1291.5
PCA	5.5	31.5
Gabors	5.5	75
Sparsenet	2.5	93

TABLE II
THE BEST SVM HYPERPARAMETERS .

Due to the multiple minmax-regions and basis functions the dimensionality of the data increases from 784 to $9 \times 169 \times 2 = 3042$ resp. $9 \times 160 \times 2 = 2880$ feature dimensions. We need to solve a ten-class problem since we have to differentiate ten digit classes (0-9). To accomplish this task using a SVM we trained 45 two-class classifiers, each of which separates two different digits (one against one). The decisions of all the two-class classifiers are then counted and finally the class with the majority of votes is selected. We used the same hyperparameters for the 45 two-class classifiers.

IV. Results

The data set is quite popular and results of many state-of-the-art methods are available for comparison [17], [18], [19]. Currently, the best results reported on the MNIST data set were obtained with convolutional neural networks plus elastic distortions (0.4% error rate [19]) and Virtual SVM with deskewing and jittering preprocessing (0.56% error rate [18]). A recent approach that uses sparse representations and elastic distortions obtains an error rate of 0.39% [27]. In [28] a method is proposed where a LeNet5 is used to train a feature extraction layer that is fed to a set of SVMs. Using elastic distortions these authors report an error rate of 0.54%.

We consider the use of an extended training set that was generated using a problem specific distortion model as data specific knowledge. Our method belongs to the class of methods that do not use such additional knowledge. In an evaluation

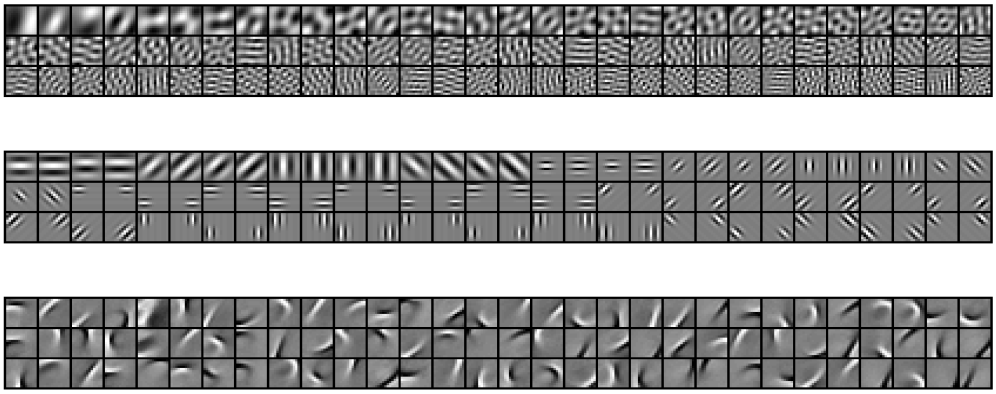

Fig. 2. Subset of basis functions used on the digit benchmark. From top to bottom: PCA basis functions, Gabor wavelets, sparse basis functions trained on digit images.

	mean validation error rate	#SVs per digit class										error rate on MNIST test set
		0	1	2	3	4	5	6	7	8	9	
raw data	2.95%($\pm$0.17)	9785	5994	13241	13244	10976	12989	9842	10429	14015	12033	1.42%
PCA	1.29%($\pm$0.08)	3293	2293	4580	4731	4041	4223	3676	3747	5246	4930	0.80%
Gabors	1.24%($\pm$0.10)	4298	2970	5959	5690	4647	5371	4527	4986	6582	5630	0.71%
Sparsenet	1.00%($\pm$0.09)	2698	1812	3667	3650	2833	3505	2770	3089	4216	3610	0.59%

TABLE I

SVM RESULTS: THE SECOND COLUMN OF THE TABLE SHOWS MEAN AND STANDARD DEVIATION OF THE TEST ERROR OF THE BEST HYPERPARAMETERS DETERMINED BY VALIDATION ON THE 7 REALISATIONS OF 10000 TRAINING AND 10000 TEST SAMPLES. THE REMAINING COLUMNS REFER TO THE CLASSIFIERS THAT WERE OBTAINED BY USING THESE HYPERPARAMETERS TO TRAIN A SVM ON THE COMPLETE MNIST TRAINING SET CONSISTING OF 60000 SAMPLES. THE NUMBER OF SVS OF ALL CLASSIFIERS OF A RESPECTIVE DIGIT CLASS AS WELL AS THE ERROR ON MNIST TEST SET ARE SHOWN.

of several matching methods that also belong to this class of methods an error rate of 0.52% is obtained [1]. In [29] an error rate of 0.63% is reported using a shape matching approach. In [30] the authors report a positive influence of sparseness to recognition performance on the MNIST set though they cannot obtain state-of-the-art performance.

The different types of basis functions obtained from and used on the MNIST set of handwritten digits are shown in figure 2. Table I shows the mean validation error of the best hyperparameter combination. Referring to the final result obtained by using the best hyperparameters on the entire MNIST training set, it shows the number of support vectors (SVs) that are used by the classifiers of each digit class as well as the error rate of the SVM on the MNIST test set (note that we use a one-against-one scheme, therefore we have 9 classifiers per digit class and the number of support vectors is the sum over all 9 SVMs). The mean validation error is worse than the final test error, since only 10000 instead of 60000 training samples were used for training.

All methods significantly outperform the direct classification of the raw data. Gabor Wavelets clearly outperform PCA. In the PCA experiment all basis functions were used. This means that for each image-patch vector an error free representation is obtained. The PCA result shows on the one hand that some performance gain can be attributed to the minimum and maximum operation since without it error free PCA yields the same result as on raw data. On the other hand the Gabor and Sparsenet results show that a sparse representation further improves performance.

The result using a learned sparse code is significantly better than the results obtained with Gabor wavelets and PCA. The number of support vectors of the best method using a learned sparse code is reduced by about a factor of three compared to the result on raw data, indicating that the feature extraction we perform successfully implements invariances of the problem of handwritten digit recognition. Also compared to earlier SVM results, for instance, the virtual SVM reported in [18], our method uses significantly less support vectors (about a factor four).

In case of sparse coding, the proposed feature extraction requires to solve the optimization problem of equation (8) via gradient descent for each patch of a given input image. Therefore PCA and Gabor wavelet feature extraction is more efficient from a computational complexity point of view. However, a number of more recent algorithms that learn sparse codes are available, see for example [21], [22]. New results also indicate that an efficient computation of the coefficients via orthogonal matching pursuit [31] is possible under certain conditions [32]. Here, we do not aim to evaluate the properties of different sparse coding approaches but want to demonstrate that the principle of sparse coding provides significant performance improvements in a real world problem on a competitive benchmark.

V. Conclusions

We proposed a method for digit recognition based on unsupervised learning of sparse basis functions. From the sparse coefficients in that basis, a new representation of the digits is obtained by applying a minmax-operation to these coefficients. The new representation incorporates invariances of handwritten digits as can be seen from the reduced number of support vectors. The performance gain is significant even though the final feature vector used for classification is not sparse in general and the dimensionality of the data increases. We compared a representation based on a learned sparse code with more traditional representations based on PCA and Gabor wavelets. Gabor wavelets can be seen as sparse basis of natural images. For digits a better performance can be achieved by using a sparse code that is obtained by unsupervised learning from the data.

Despite its simplicity, our approach performs as well as state-of-the-art methods that do not use prior knowledge specific to the handwritten digit recognition problem (a comprehensive list of results can be found on the internet [20]). Our method provides comparable performance to the image matching methods as evaluated in [1]. For example, methods proposed in [18], [19], [27], [28] employ an elastic deformation modell for digits to boost their performance, while our method implicitly extracts deformation invariances by the unsupervised learning of a sparse code.

We have shown that a sparse feature representation, combined with biologically plausible max-operations, leads to highly competitive classification performance. We suggest that the method, being quite general, simple, and straightforward, may be applied to a broad range of visual pattern recognition problems. In cases where a specific image distribution is given that deviates significantly from natural images, the specific learned sparse feature representation which arises may lead to a significant increase in classification performance.

References

[1] D. Keysers, C. Gollan, T. Deselaers, and H. Ney, "Deformation models for image recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 8, pp. 1422–1435, 2007.

[2] K. Fukunaga, *Introduction to statistical pattern recognition (2nd ed.)*. San Diego, CA, USA: Academic Press Professional, Inc., 1990.

[3] J. G. Daugman, "Complete discrete 2-D gabor transforms by neural networks for image analysis and compression," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 36, no. 7, pp. 1169–1179, 1988.

[4] C. Zetzsche, E. Barth, and B. Wegmann, "The importance of intrinsically two-dimensional image features in biological vision and picture coding," in *Digital Images and Human Vision*, A. B. Watson, Ed. MIT Press, Oct. 1993, pp. 109–38.

[5] D. J. Field, "What is the goal of sensory coding?" *Neural Computation*, vol. 6, no. 4, pp. 559–601, 1994.

[6] B. A. Olshausen and D. J. Field, "Natural image statistics and efficient coding," *Network*, vol. 7, no. 2, pp. 333–339, 1996.

[7] E. P. Simoncelli and B. A. Olshausen, "Natural image statistics and neural representation," *Annual Review of Neuroscience*, vol. 24, pp. 1193–1216, 2001.

[8] A. J. Bell and T. J. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," *Neural Computation*, vol. 7, no. 6, pp. 1129–1159, 1995.

[9] A. Hyvärinen, "Survey on independent component analysis," *Neural Computing Surveys*, vol. 2, pp. 94–128, 1999.

[10] B. A. Olshausen and D. J. Field, "Sparse coding with an overcomplete basis set: a strategy employed by V1?" *Vision Research*, vol. 37, no. 23, pp. 3311–3325, 1997.

[11] M. S. Lewicki and T. J. Sejnowski, "Learning nonlinear overcomplete representations for efficient coding," in *NIPS '97: Proceedings of the 1997 conference on Advances in neural information processing systems 10*. Cambridge, MA, USA: MIT Press, 1998, pp. 556–562.

[12] A. J. Bell and T. J. Sejnowski, "The "independent components" of natural scenes are edge filters." *Vision Res*, vol. 37, no. 23, pp. 3327–3338, December 1997.

[13] B. A. Olshausen and D. J. Field, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, no. 381, pp. 607–609, 1996.

[14] M. Riesenhuber and T. Poggio, "Hierarchical models of object recognition in cortex," *Nature Neuroscience*, vol. 2, no. 11, pp. 119–125, 1999.

[15] I. Lampl, D. Ferster, T. Poggio, and M. Riesenhuber, "Intracellular measurements of spatial integration and the max operation in complex cells of the cat primary visual cortex," *J Neurophysiol*, vol. 92, no. 5, pp. 2704–2713, 2004.

[16] T. Serre, L. Wolf, S. Bileschi, M. Riesenhuber, and T. Poggio, "Robust object recognition with cortex-like mechanisms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 3, pp. 411–426, 2007.

[17] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-Based Learning Applied to Document Recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, November 1998.

[18] D. Decoste and B. Schölkopf, "Training Invariant Support Vector Machines," *Mach. Learn.*, vol. 46, no. 1-3, pp. 161–190, 2002.

[19] P. Y. Simard, D. Steinkraus, and J. C. Platt, "Best Practices for Convolutional Neural Networks Applied to Visual Document Analysis," in *ICDAR '03: Proceedings of the Seventh International Conference on Document Analysis and Recognition*. Washington, DC, USA: IEEE Computer Society, 2003, p. 958.

[20] Y. LeCun, "MNIST handwritten digit database, NEC research institute," http://yann.lecun.com/exdb/mnist/.

[21] M. S. Lewicki and T. J. Sejnowski, "Learning Overcomplete Representations." *Neural Computation*, vol. 12, no. 2, pp. 337–365, 2000.

[22] M. Aharon, M. Elad, and A. Bruckstein, "K-svd: An algorithm for designing overcomplete dictionaries for sparse representation," *Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and Signal Processing, IEEE Transactions on]*, vol. 54, no. 11, pp. 4311–4322, 2006.

[23] E. Barth, B. L. Beard, and A. J. Ahumada, "Nonlinear features in vernier acuity," in *IS&T/SPIE Symposium on Human Vision and Electronic Imaging*, vol. 3644, no. 8, 1999.

[24] V. N. Vapnik, *The nature of statistical learning theory*. New York, NY, USA: Springer-Verlag New York, Inc., 1995.

[25] N. Christianini and J. Shawe-Tylor, *Support Vector Machines*. Cambridge University Press, 2003.

[26] T. Martinetz, K. Labusch, and D. Schneegaß, "SoftDoubleMinOver: A Simple Procedure for Maximum Margin Classification." in *Artificial Neural Networks: Biological Inspirations. ICANN 2005: 15th International Conference. Proceedings, Part II*, W. Duch, J. Kacprzyk, E. Oja, and S. Zadrozny, Eds., 2005, pp. 301–306.

[27] M. Ranzato, C. Poultney, S. Chopra, and Y. LeCun, "Efficient learning of sparse representations with an energy-based model," in *Advances in Neural Information Processing Systems 19*, B. Schölkopf, J. Platt, and T. Hoffman, Eds. Cambridge, MA: MIT Press, 2007, pp. 1137–1144.

[28] F. Lauer, C. Y. Suen, and G. Bloch, "A trainable feature extractor for handwritten digit recognition," *Pattern Recogn.*, vol. 40, no. 6, pp. 1816–1824, 2007.

[29] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 4, pp. 509–522, 2002.

[30] R. Steinert, M. Rehn, and A. Lansner, "Recognition of handwritten digits using sparse codes generated by local feature extraction methods," in *ESANN*, 2006, pp. 161–166.

[31] Y. Pati, R. Rezaiifar, and P. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," *Proceedings of the 27 th Annual Asilomar Conference on Signals, Systems,*, November 1993.

[32] D. L. Donoho, M. Elad, and V. N. Temlyakov, "Stable recovery of sparse overcomplete representations in the presence of noise." *IEEE Transactions on Information Theory*, vol. 52, no. 1, pp. 6–18, 2006.