In-Bed Posture Classification Based on Sparse Representation in Redundant Dictionaries

O. Mihálik * **T. Sýkora** ** **M. Husák** *** **P. Fiedler** ****

* *Department of Control and Instrumentation, Faculty of Electrical Engineering, Brno University of Technology, Brno, Czech republic, (e-mail: ondrej.mihalik1@vut.cz).*
** *(e-mail: tomas.sykora1@vut.cz)*
*** *(e-mail: xhusak08@vut.cz)*
**** *(e-mail: fiedlerp@vut.cz)*

Abstract: Non-orthogonal signal representation using redundant dictionaries gradually gained popularity over the last decades. Sparse methods find major application in signal denoising, audio declipping, time-frequency analysis, and classification, to name a few. This paper is inspired by the exceptional results of sparse representation classification originally suggested for face recognition. We compare the method to other common classifiers using simulated as well as real datasets. In the latter the proposed method is tested with real pressure data from a bed equipped with a matrix of 30×11 pressure sensors. Here the method outperforms standard classification methods (surpassing 91 % accuracy) without need of parameter selection or special user's skills. Furthermore it offers a means of dealing with occlusions, whose results are presented as well.

Keywords: sparse representation, redundant dictionary, SRC, posture classification, denoising

1. INTRODUCTION

Over the last decades, research of signal-processing methods has shifted from orthogonal bases, such as Fourier basis or Wavelet bases, towards the non-orthogonal signal representations relying on redundant dictionaries, see Mallat (2010). Sparse signal representation is a modern approach to signal modelling which enables us to express a discrete signal as a superposition of a small number of vectors, called *atoms*. Atoms are drawn from a *redundant dictionary*, which may be constructed analytically or learned from the data. Hence, compared to bases, redundant dictionaries offer much better adaptability to the class of signals which are expected in a particular application. Sparse representation of signals was successfully applied in a broad range of domains, such as audio declipping, Zaviska (2020); time-frequency analysis, Mallat (2010); denoising, Condat (2013); classification, Wright (2009); etc. The focus of this article is on the last of the listed applications, the so called *sparse representation classification* (SRC), which has been successfully used for face recognition in the above cited paper.

The aim of this article is to classify in-bed posture of a person via sparse approximation of the two-dimensional pressure field. The paper is organised as follows. Section 2 gives an overview of the mathematical framework needed for sparse representation of signals. Section 3 deals with the details of the SRC procedure and illustrates its performance on a simulated waveform example. The method is compared with standard classification methods. Section 4 discusses a means of dealing with occlusions and demonstrates application to real data. SRC is compared to some of the most common classifiers. Their accuracies are assessed in terms of misclassification rates for normal and partially corrupted data obtained from a bed equipped with a matrix of pressure sensors. Immunity to occlusions is crucial for our case, because the application lies stress on robustness even during a partial failure of the measuring matrix.

2. PRELIMINARIES

In this paper we expect that the signal being processed had already been sampled at K discrete time instants. Hence it can be represented by the column vector

$$\mathbf{f} = [f_1 \; f_2 \; f_3 \; \dots \; f_K]^{\mathsf{T}}, \tag{1}$$

where the superscript 'T' stands for the transpose and $\{f_k\}_{k=1}^{K}$ are the signal samples.

2.1 Signal representation in orthogonal bases

In many scientific fields, discrete signals (1) are conveniently represented using orthonormal bases, i.e., as a superposition of basis vectors stored in a square matrix $\mathbf{B}$ and a vector $\mathbf{c}$ containing K spectral coefficients,

$$\mathbf{f} = \mathbf{B}\mathbf{c}. \tag{2}$$

Perhaps the most common orthogonal representation is Discrete Fourier basis, for which $\mathbf{B}$ contains samples of K complex exponentials and $\mathbf{c}$ stores K complex numbers, the coefficients of the frequency spectrum.

2.2 Sparse representation of signals

In sparse signal representation we represent the signal $\mathbf{f}$ using a redundant dictionary $\mathbf{D}$, which typically contains P columns referred to as *atoms*, and a vector $\mathbf{a}$ containing P coefficients of the representation

$$\mathbf{f} = \mathbf{D}\mathbf{a}. \tag{3}$$

Typically the number of atoms P is larger than the number of their samples K, hence the term *redundant dictionary.* The intention here is that making P sufficiently large may secure that there will be atoms similar to the processed signal. Therefore a small number of these atoms should be sufficient to form a good signal approximation. The representation (3) is referred to as *sparse*, because the vector $\mathbf{a}$ contains only a small number on non-zero elements.

Before we proceed to the methods of obtaining sparse signal representation, we need to recall that a discrete signal (1) is associated with its ℓ^p norm

$$\|\boldsymbol{f}\|_p = \left(\sum_{k=1}^{K} |f_k|^p \right)^{1/p}, \qquad p \in \langle 0, \infty) \,. \tag{4}$$

The cases $p = 0, 1, 2$ are essential for this work. The ℓ^2 norm is perhaps the most common; it is the Euclidean length of the vector. The ℓ^1 norm is the sum of absolute values of the vector elements. And the ℓ^0 "norm" is the number of non-zero vector elements, although it is not considered to be a proper norm in the mathematical sense.

Practical digital signal processing has to deal with different forms of noise. To do so, a small approximation error $E > 0$ is admitted so that

$$\mathbf{f} \approx \mathbf{D}\mathbf{a}, \qquad \|\mathbf{f} - \mathbf{D}\mathbf{a}\|_2 \leq E. \tag{5}$$

To obtain sparse signal representation, we seek such vector $\mathbf{a}$ which contains as few non-zero coefficients as possible and, at the same time, satisfies the condition (5). In other words, we are aiming to minimize $\|\boldsymbol{a}\|_0$ while keeping the approximation error below a pre-defined positive constant E.

Contemporary techniques cannot find the exact solution of above stated problem if the problem is too large, for we would have to test too many combinations of atoms. Nevertheless, there are greedy algorithms for finding a suboptimal solution, which is often not far from the true solution, see Mallat (2010).

2.3 Orthogonal matching pursuit

At the outset of our research, we relied on the *orthogonal matching pursuit* (OMP), an iterative approach which starts signal approximation with one atom (i.e. only one non-zero element in $\mathbf{a}$) and iteratively admits new atoms to the approximation. During each iteration, the coefficients of included atoms are updated via least-squares fit, i.e., using matrix inversion. The error E gradually decreases with iteration and the algorithm is terminated once the condition (5) is satisfied.

OMP often works well for a small number of atoms, but once an atom is admitted into the representation, it remains there even if the addition of new atoms and the orthogonal update would render its amplitude negligible. In some applications OMP may be prone to poor performance and numerical instability, which occurred during our research. Once we admitted too many atoms into the representation, their amplitude 'blew up' during the inversion process. This may be attributed to the greediness of the OMP.

2.4 Basis pursuit

The numerical instability of the OMP (which is an ℓ^0 approach) may be mitigated via the so called ℓ^1 *relaxation*, see Mallat (2010). *Basis pursuit* minimizes the modified functional

$$Q_{\mathrm{B}}(\mathbf{a}) = \|\mathbf{D}\mathbf{a} - \mathbf{f}\|_2^2 + \lambda_{\mathrm{B}} \|\mathbf{a}\|_1 \tag{6}$$

instead of the original ℓ^0 problem

$$Q(\mathbf{a}) = \|\mathbf{D}\mathbf{a} - \mathbf{f}\|_2^2 + \lambda \|\mathbf{a}\|_0 \,. \tag{7}$$

In many cases minimization of (6) and (7) leads to the same solutions, but minimization of the former is much simpler for the contemporary techniques. There has been an extensive research of ℓ^1 algorithms. For instance, MATLAB supports this form of minimization through the `lasso` function, which is based on the *coordinate descent method* of Friedman (2010). This frees the user from the study of the abstruse implementation details of available ℓ^1 algorithms.

3. SPARSE REPRESENTATION CLASSIFICATION

The reader may wonder how to construct a good dictionary. One may merge several bases, e.g., Fourier basis $\mathbf{F}$ and Haar basis $\mathbf{H}$, to form a dictionary by appending them into one larger matrix

$$\mathbf{D} = [\mathbf{F}\ \mathbf{H}] \,. \tag{8}$$

Unfortunately, such analytical construction of dictionaries requires a great deal of experience.

When we have a dataset representing the typical signals which occur in our application, we may use these data to construct the dictionary. There are a number of *dictionary learning algorithms*, see Ramirez (2010). Various dictionary-learning processes typically involve three common parameters: the regularisation constant γ, the number of atoms P and the number of learning iterations. Their appropriate selection normally requires a time-consuming crossvalidation process.

Perhaps the simplest approach to dictionary learning is the SRC method presented by Wright (2009). Assume we have classes labelled $1, 2, \ldots, N$ associated with training observations stacked into matrices $\mathbf{D}_1, \mathbf{D}_2, \ldots, \mathbf{D}_N$, respectively. (We assume that all training observations had already been normalised so that each column of these matrices has unit ℓ^2 norm.) In the learning phase, all available observations are simply concatenated to form the dictionary

$$\mathbf{D} = [\mathbf{D}_1\ \mathbf{D}_2\ \cdots\ \mathbf{D}_N] \,. \tag{9}$$

In the classification phase, SRC uses this dictionary to express a new observation $\mathbf{f}$ in a sparse way; in our case through the minimization of (6). The coefficients thus obtained may be split into shorter vectors

$$\mathbf{a} = \begin{bmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \\ \vdots \\ \mathbf{a}_N \end{bmatrix} \tag{10}$$

whose heights match the widths of the corresponding matrices $\mathbf{D}_1, \mathbf{D}_2, \ldots, \mathbf{D}_N$. These are then used to calculate approximation errors e_n induced when the coefficients of the corresponding class only are kept in the approximation.

$$e_n = \|\mathbf{f} - \mathbf{D}_n \mathbf{a}_n\|_2 \,, \qquad n = 1, 2, \ldots, N \tag{11}$$

The observation is classified into the class with the smallest error e_n, in other words, into the class whose coefficients, used alone, provide the most accurate representation of the observation $\mathbf{f}$. (We will illustrate these statements in the subsequent sections using two different classification examples.)

The reader may be concerned about the computational burden of the ℓ^1 optimization in practice. It is true that searching through a large dictionary would correspond to long classification times, but the datasets used in this work comprised only of hundreds of training samples and the whole bed pressure field in Section 4 was sampled once per five seconds. Therefore there were no issues with the feasibility of real time classification.

SRC approach seems especially advantageous in situations when the training database is small. In such cases the computational complexity of the ℓ^1 optimization remains amiable, yet we can do without the selection of dictionary learning parameters since SRC involves no iterative training process.

3.1 Simulated example: Waveform data

We will examine a synthetic one-dimensional classification example before we proceed to the practical application in the next section. We decided to use the example originally proposed by Breiman (1984) and later popularized by Hastie (2010), as it is labelled as a 'difficult pattern recognition problem' by the latter author. There are three classes and the predictors are defined as

$$X_j = \begin{cases} Uh_1(j) + (1-U)h_2(j) + \epsilon_j, & \text{Class 1,} \\ Uh_1(j) + (1-U)h_3(j) + \epsilon_j, & \text{Class 2,} \\ Uh_2(j) + (1-U)h_3(j) + \epsilon_j, & \text{Class 3.} \end{cases} \tag{12}$$

where $j = 1, 2, 3, \ldots, 21$, U is a random variate with uniform distribution on $(0;1)$, ϵ_j are standard normal variates, and the h_l are the shifted triangular waveforms

$$\begin{aligned} h_1(j) &= \max(6 - |j - 11| \,, 0), \\ h_2(j) &= \max(6 - |j - 15| \,, 0), \\ h_3(j) &= \max(6 - |j - 7| \,, 0). \end{aligned} \tag{13}$$

There are 100 training observations per class, which accounts for a total of 300 training observations.

To illuminate this rather elaborate mathematical description, we provide Figure 1, which displays three random observations for each of the three classes.

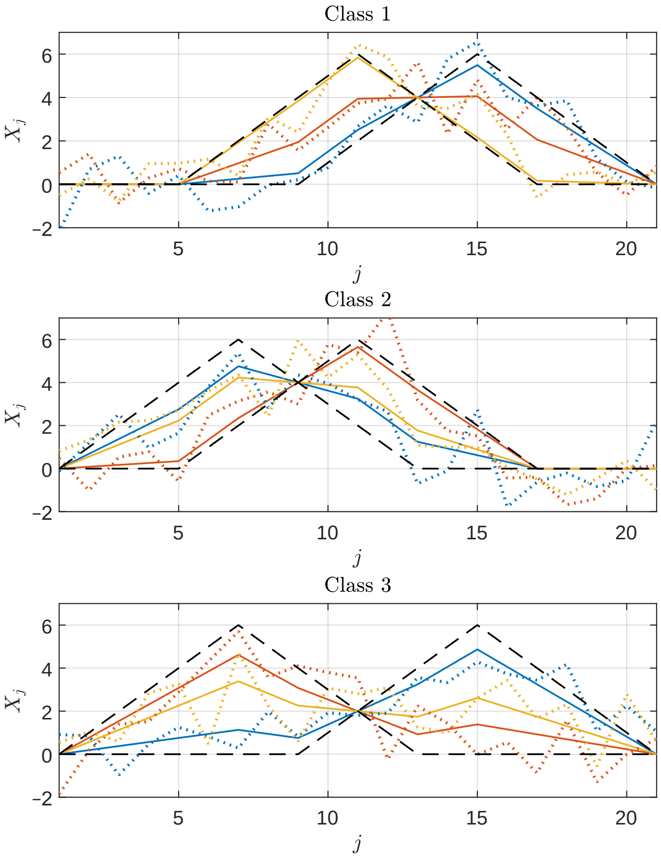

Fig. 1. The three classes of the waveform example. Generating triangular functions (dashed) which are summed to produce random instances of noiseless observations (solid lines). These are eventually corrupted by noise to produce the simulated data (dotted lines). Particular observations are distinguished by colours.

A classifier is trained using the 300 observation. Then it is assessed using 500 new test observations. The generation of datasets, training of the classifier and its test are repeated ten times, so that we get an estimate of the mean accuracy for each classifier. Hastie (2010) provides the following table of error rates for some of the most popular classifiers. The Bayes prediction rate is also included.

Table 1. Error rates for waveform data, Hastie (2010) p. 454. The values are averages over ten simulations, with the standard error of the average in the parentheses.

Technique	Error rate	
	Training	Test
LDA	0.121(0.006)	0.191(0.006)
QDA	0.039(0.004)	0.205(0.006)
CART	0.072(0.003)	0.289(0.004)
FDA/MARS (degree = 1)	0.100(0.006)	0.191(0.006)
FDA/MARS (degree = 1)	0.068(0.004)	0.215(0.002)
MDA (3 subclasses)	0.087(0.005)	0.169(0.006)
MDA (3 subclasses, 4 df)	0.137(0.006)	0.157(0.005)
PDA (4 df)	0.150(0.005)	0.171(0.005)
SRC	0 (0)	0.225(0.006)
Bayes		0.140

The error rates of SRC (listed directly above the Bayes rate) are worse than the average of the other methods. Nevertheless, the example is excellent for demonstration of the details of the SRC algorithm.

All 300 training observations, such as those in Figure 1, were used for construction of the dictionary (9). Perfect accuracy on training data is the expected behaviour of the method, because this situation corresponds to solving (5) with one of the columns in the dictionary $\mathbf{D}$ being equal to the signal $\mathbf{f}$. We get the sparsest representation possible, since only one non-zero coefficient in $\mathbf{a}$ is needed.

We will now move to the behaviour on test dataset. Of course, the situation is less favourable in this setting. We need multiple coefficients, but they are still relatively sparse, as we can see in Figure 2 (top).

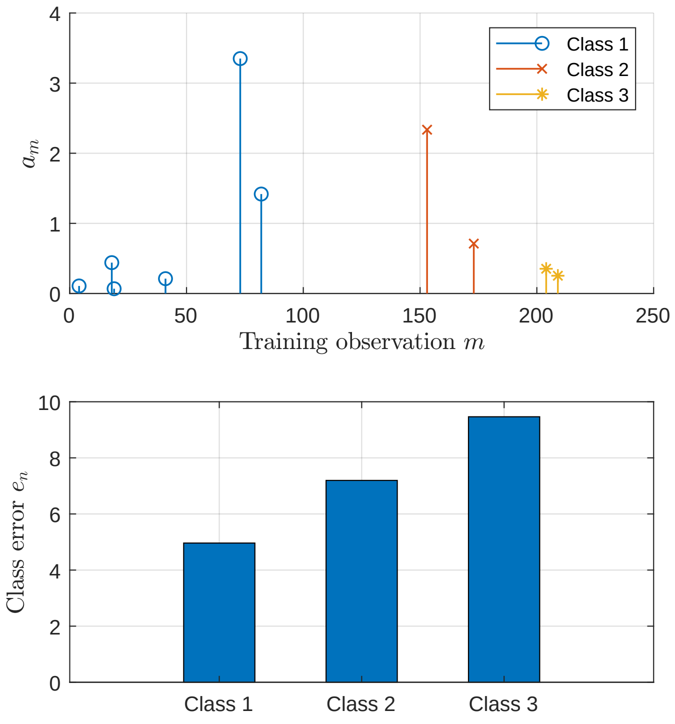

Fig. 2. Sparse coefficients, $\mathbf{a}$, of an observation correctly classified into Class 1 (top) and class errors e_n corresponding to the Classes 1–3 (bottom).

There are only 10 non-zero coefficients in $\mathbf{a}$ so the representation can be considered quite sparse. The observation $\mathbf{f}$ which is being analysed corresponds to Class 1 and, obviously, coefficients of this class preponderate. Some coefficients of the Classes 2 and 3 are present as well, yet only those of the former attain pronounced magnitudes. This signal $\mathbf{f}$ is shown in the Figure 3 so that the reader may compare it with the typical training examples in Figure 1.

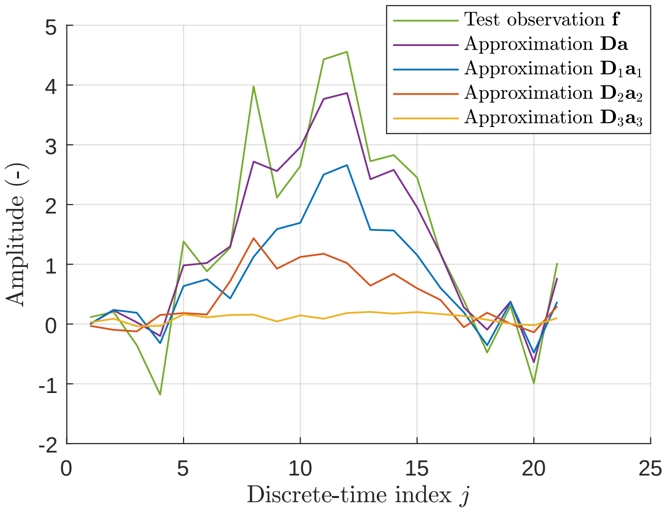

Fig. 3. Signal and its approximations calculated using coefficients of its sparse representation.

Coefficients of the classes may be used to compute the approximations of the observed signal $\mathbf{f}$. The approximation $\mathbf{Da}$ uses all coefficients, thereby exhibits the best accuracy. Contributions from different classes, $\mathbf{D}_1\mathbf{a}_1$, $\mathbf{D}_2\mathbf{a}_2$, and $\mathbf{D}_3\mathbf{a}_3$, are shown in Figure 3 as well.

A glance at the figure may assures us that the Class 1 is the most likely. But the computer cannot perform such visual analysis of the waveforms. Instead, it may compute the ℓ^2 errors between the signal $\mathbf{f}$ and each of these approximations according to (11). The results are displayed in Figure 2 (bottom). The error of Class 1 is the smallest, which is in accordance with our visual analysis.

4. DEALING WITH OCCLUSIONS

One of the desirable features of SRC is its ability to deal with concentrated noise, such as extremely dark or bright spots on photographs. We will describe a different application where this feature is also required.

4.1 Bed pressure measurement

There are automated systems in healthcare facilities which require a means of classification of a patient's posture when staying in bed. Examples may be sleep monitoring or decubitus prevention. We will not venture to describe their details here. Instead we will confine our discussion to the fact that a special mattress may be equipped with a matrix of pressure sensors which measure the pressure field. In our case, the bed was covered with a matrix of 30×11 sensors. Figure 4 depicts a few examples of such pressure fields. Each column corresponds to a different body positions and rows are associated with different test subjects.

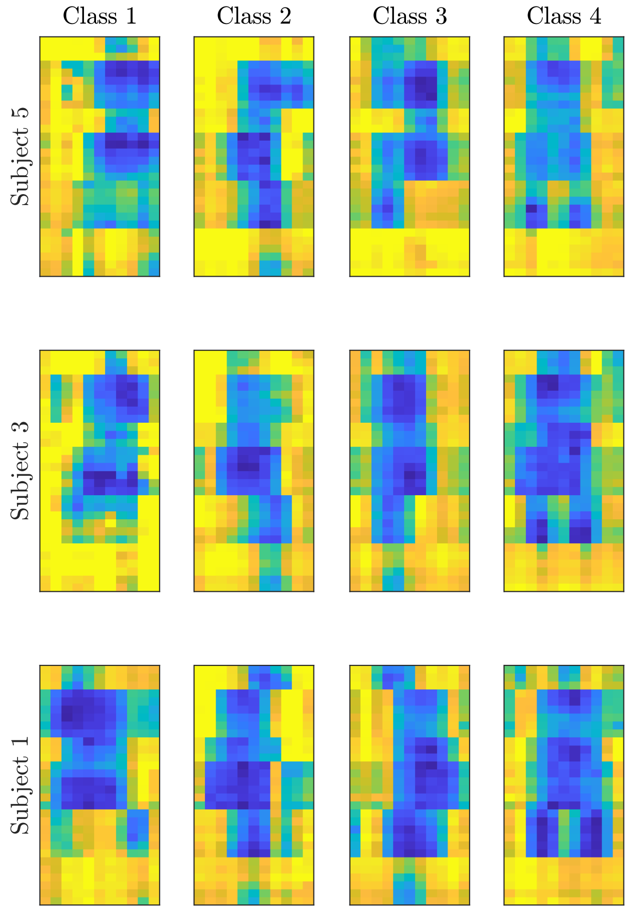

Fig. 4. Signal examples from the bed pressure dataset.

The Classes 1–4 correspond to the subjects measured in

(1) supine,
(2) left lateral,
(3) right lateral, and
(4) prone position.

There were 6 test subjects of various weights and heights. The dataset contains 736 different images of the pressure field and the data are, naturally, noisy. This classification task is commonly solved via more involved methods of feature extraction, such as Hung (2015), but authors such as Wright (2009) or Liu (2013) have shown that, using SRC, we may dispense with these methods.

4.2 Adapting to occlusions

Wright (2009) described a means of enlarging dictionary so as to include the anticipated occlusions into the dictionary. In its simplest form this is achieved by appending an identity matrix $\mathbf{I}$ to the dictionary. Therefore, instead of using (9), we concatenate the matrices

$$\mathbf{D} = [\mathbf{D}_1 \ \mathbf{D}_2 \ \cdots \ \mathbf{D}_N \ \mathbf{I}]\,. \tag{14}$$

When the user has some additional information concerning the occlusions, such as their typical sizes or most probable locations, other matrices may be appended in place of the identity matrix. For instance, we know that if a fault occurs, it disables sensors in the whole row of the mattress. There are 30 such rows in the mattress. Therefore it is more appropriate to append a matrix which contains all 30 instances of admissible errors.

Note that we do not need to add all possible combinations of failed rows because this task is, indeed, solved by the ℓ^1 sparse representation.

4.3 Numerical experiment

SRC and some of the most common classifiers were assessed using the bed pressure dataset. Their error rates were estimated using the 6-fold crossvalidation procedure. In each fold, one of the subjects was left out for testing, and remaining five subjects were used for training. The results are listed in the first numerical column of the following table.

Table 2. Error rates for bed pressure data. The error rates were obtained using crossvalidation.

Technique	Test error rate		
	Without occlusions	Occluded by zeroes	Occluded by ones
LDA	0.240	0.595	0.486
LDA*	0.190	0.199	0.199
QDA*	0.385	0.374	0.375
SVM	0.140	0.236	0.187
SVM*	0.148	0.162	0.159
KNN, $K=1$	0.149	0.207	0.165
KNN*, $K=1$	0.151	0.148	0.148
CART	0.486	0.511	0.496
CART*	0.455	0.451	0.451
Bagged tree	0.242	0.263	0.251
Bagged tree*	0.217	0.218	0.213
SRC	0.087	0.086	0.090

Two more separate tests were performed. During the crossvalidation test fold was enlarged by artificially generated occlusions to test the robustness of the classifiers. Error rates soared for most of the classifiers, while the accuracy of SRC remained virtually unaltered.

To make the comparison fair, training sets of the common classifiers marked with asterisk were enhanced. They were trained using an enlarged dataset, which was created by occluding different rows of the training observations. Since there are 30 rows in the bed, and the data may be occluded either by zeros or ones, this corresponds to a 61-fold enlargement of the training dataset. (The signals without occlusions were left in the training dataset as well.) Training of QDA was made possible by this enriched dataset, since with the dataset without occlusions it failed.

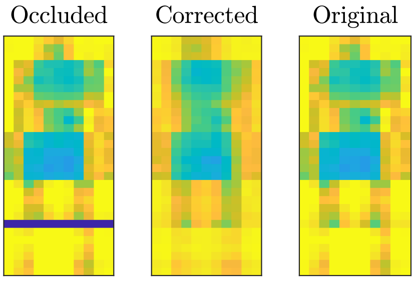

Fig. 5. Observed data, data corrected by SRC, and original image.

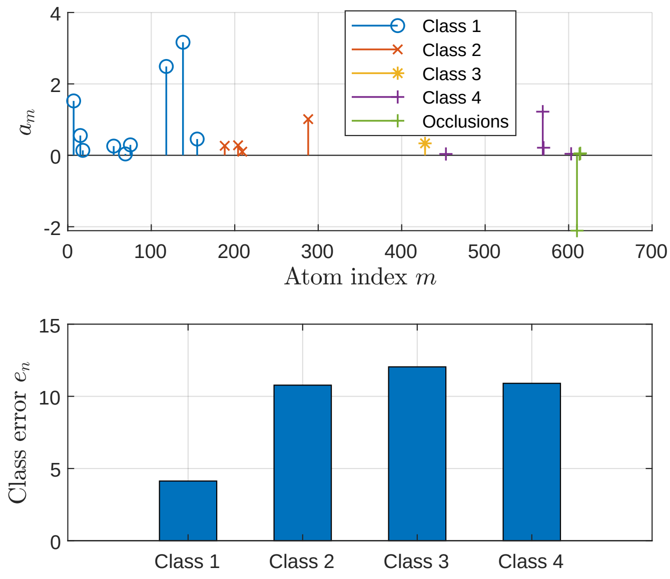

Fig. 6. Sparse coefficients, **a**, of an observation correctly classified into Class 1 (top) and class errors e_n corresponding to the Classes 1–4 and occlusions (bottom).

5. CONCLUSION

This paper shows how to implement a robust algorithm capable of practical classification of patients' posture in bed. SRC requires no feature extraction, thereby greatly simplifies the design of the classifier by avoiding selection of parameters, which would require estimation trough a crossvalidation process. The proposed method is relatively simple, can be implemented in MATLAB and, as it was demonstrated, it is suitable for processing of measured real-world signals. It was shown that the method inherently deals with occlusions exceptionally well.

We would like to focus our future research on the D-KSVD algorithm introduced by Aharon (2006). Its may be more attractive for the classification in embedded devices. Compared to SRC, D-KSVD is reported to be successful at reducing computational burden by compacting the dictionaries, albeit at the cost of sacrificing some accuracy, see Jiang (2016).

ACKNOWLEDGEMENTS

The completion of this paper was made possible by the following projects:

- FV40196—Research and development of the monitoring of immobile persons who are tethered to the bed in terms of risk of suffocation – decubitus, Ministry of Industry and Trade, Czech Republic.
- FW03010100—Research and development of detection and monitoring of people with dense coverage in shopping centers in order to optimize shopping flows and, in the event of crises and epidemics, access control to these areas.
- FV40247—Cooperative robotic platforms for automotive and industrial applications
- FEKT-S-20-6205—Research in Automation, Cybernetics and Artificial Intelligence within Industry 4.0, financially supported by the Internal science fund of Brno University of Technology.

REFERENCES

Aharon, M. et al. (2006). K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE transactions on signal processing.* PISCATAWAY: IEEE, **54**(11), pp. 4311–4322. ISSN 1053-587X. doi: 10.1109/TSP.2006.881199.

Breiman, L. and Ihaka, R. (1984). Nonlinear discriminant analysis via scaling and ACE, *Technical report*, University of California, Berkeley.

Condat, L. (2013). A Direct Algorithm for 1-D Total Variation Denoising. *IEEE signal processing letters.* IEEE, **20**(11), pp. 1054–1057. ISSN 1070-9908. doi: 10.1109/LSP.2013.2278339

Friedman, J. et al. (2010). Regularization Paths for Generalized Linear Models via Coordinate Descent. *Journal of Statistical Software.* ASA, **33**(1), pp. 1–22. ISSN 1548-7660. doi: 10.18637/jss.v033.i01

Hastie, T. et al. (2017). *The Elements of Statistical Learning*, 2nd ed. Springer, New York. ISBN 978-0-387-84857-0.

Hung, Y. et al. (2015). Bed posture classification based on artificial neural network using fuzzy c-means and latent semantic analysis. *Journal of the Chinese Institute of Engineers.* TAIPEI: Taylor & Francis, **38**(4), pp. 415–425. ISSN 0253-3839. doi: 10.1080/02533839.2014.981212

Jiang, W. et al. (2016). Joint Label Consistent Dictionary Learning and Adaptive Label Prediction for Semisupervised Machine Fault Classification. *IEEE transactions on industrial informatics.* PISCATAWAY: IEEE, **12**(1), pp. 248–256. ISSN 1551-3203. doi: 10.1109/TII.2015.2496272

Liu, J.J. et al. (2014). Sleep posture analysis using a dense pressure sensitive bedsheet. *Pervasive and mobile computing.* Elsevier B.V, **10**, 34–50. ISSN 1574-1192. doi: 10.1016/j.pmcj.2013.10.008

Mallat, S. (2009). *A wavelet tour of signal processing: the sparse way*, 3rd ed. Academic Press, Elsevier, Boston. ISBN 978-0-12-374370-1.

Ramirez, I. et al. (2010). Classification and Clustering via Dictionary Learning with Structured Incoherence and Shared Features, *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition.* IEEE, pp. 3501–3508. ISSN 1063-6919. doi: 10.1109/CVPR.2010.5539964

Wright, J. et al. (2009). Robust Face Recognition via Sparse Representation. *IEEE transactions on pattern analysis and machine intelligence.* LOS ALAMITOS: IEEE, **31**(2), 210-227. ISSN 0162-8828. doi: 10.1109/TPAMI.2008.79

Zaviska, P. et al. (2021). A survey and an extensive evaluation of popular audio declipping methods. *IEEE journal of selected topics in signal processing.* IEEE, **15**(1), pp 5–24. ISSN 1932-4553. doi: 10.1109/JSTSP.2020.3042071

Zhang, B. and Li, Q. (2010). Discriminative K-SVD for dictionary learning in face recognition. *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition.* IEEE, pp. 2691–2698. ISSN 1063-6919. doi: 10.1109/CVPR.2010.5539989