Weighted Collaborative Representation and Classification of Images

Radu Timofte[1] and Luc Van Gool[1,2]
[1]*ESAT-PSI/IBBT, Catholic University of Leuven, Belgium*
[2]*D-ITET, ETH Zurich, Switzerland*
{*Radu.Timofte, Luc.VanGool*}*@esat.kuleuven.be*

Abstract

Recently a collaborative representation (CR) based classification with regularized least squares (CRC-RLS) has been proposed for the classification of faces. CRC-RLS is a simple yet fast alternative to sparse representation (SR) based classification (SRC). While SR is the solution to an l_1-regularized least square decomposition, CR starts from an l_2-regularized least square formulation. In this paper we extend the CRC-RLS approach to the case where the samples are weighted based on classification confidence and/or the feature channels are weighted using variance. The weighted collaborative representation classifier (WCRC) improves classification performance over that of the original formulation, while keeping the simplicity and the speed of the original CRC-RLS formulation.

1 Introduction

Expressing new observations as linear combinations of previously recorded data is a powerful method for classification [8]. Usually the problem is defined as an optimization aiming at lowering the residual error between a new query and its obtained reconstruction. Commonly, the residual error is measured in the sense of least squares. Having an algebraic solution of the fitting problem is desirable, which is the case for the least squares formulation. When all samples contribute to the solution, the method is coined Collaborative Representation (CR) [10]. Aside from the basic least squares criterion for best fitting new observations, other constraints are considered in the literature as well. For stabilizing the coefficients of the least squares decomposition, one could add minimization over the l_2-norm of the solution coefficients, thus still admitting an algebraic solution. Enforcing sparsity of the least squares solution leads to l_0-regularization, which, in practice, boils down to an l_1-regularized least squares problem known as *lasso* and bringing a Sparse Representation (SR) [8]. Sparsity is a key insight in compressed sensing – most signals admit a decomposition over a reduced set of signals from the same class. Unfortunately, there is no known algebraic solution to such an l_1-regularized least squares formulation. A combined l_1/l_2 regularization tends to robustify/group the coefficients of the solution while enforcing sparsity [11]. The obtained representation, seen as a linear decomposition over a pool of samples, has a structural meaning in that the residuals and the solution coefficients reveal the importance of each sample (or group of) for the new input query. This information is used in the classification (or assignment) of the query sample to the best class of samples (the one with minimum residual error or largest coefficient impact).

In this paper, we investigate the weighted variants of the aforementioned representations, focusing on the l_2-regularized least squares formulations with algebraic solutions. How useful are these representations for classification? In which scenarios? Does weighting improve the performance? How to introduce weights? – are the main questions we aim to answer.

Section 2 briefly reviews the least squares based formulations. Section 3 presents weighted variants. Section 4 describes the classifiers based on these representations. Section 5 shows experimental results, while Section 6 concludes the paper.

2 Least Squares formulations

The basic Ordinary Least Squares (OLS) problem aims at optimizing:

$$\hat{\beta}_{OLS} = \arg\min_{\beta} \|y - X\beta\|^2 \tag{1}$$

where $X \in \mathbb{R}^{n \times m}$ is the data matrix with $m \in \mathbb{N}$ n-dimensional samples and $\beta \in \mathbb{R}^m$ is the vector of coefficients from the representation of the query $y \in \mathbb{R}^n$. If $(X^T X)^{-1}$ exists the algebraic solution is given by:

$$\hat{\beta}_{OLS} = (X^T X)^{-1} X^T y \tag{2}$$

The Collaborative Representation with Regularized Least Squares [10], here shortly called CR, solves:

$$\hat{\beta}_{CR} = \arg\min_{\beta} \|y - X\beta\|^2 + \lambda_{CR}\|\beta\|^2 \quad (3)$$

where $\lambda_{CR} \in \mathbb{R}$ is a regulatory parameter. The algebraic solution becomes:

$$\hat{\beta}_{CR} = (X^T X + \lambda_{CR} I)^{-1} X^T y \quad (4)$$

where I is the $m \times m$ identity matrix.

If instead of l_2-regularization we enforce the sparsity by means of l_1-regularization we obtain a Sparse Representation (SR) [8] and solve:

$$\hat{\beta}_{SR} = \arg\min_{\beta} \|y - X\beta\|^2 + \lambda_{SR}\|\beta\|_1 \quad (5)$$

where $\lambda_{SR} \in \mathbb{R}$ is the Lagrangian regulatory parameter. For this problem, also known as *lasso*, we do not know an algebraic solution.

Combining sparsity and robustness by means of l_1 and l_2 regularization we have to solve:

$$\hat{\beta}_{EN} = \arg\min_{\beta} \|y - X\beta\|^2 + \lambda_2\|\beta\|^2 + \lambda_1\|\beta\|_1 \quad (6)$$

a problem also known as Elastic Net (EN) [11], where λ_1 and λ_2 are regulatory parameters.

3 Weighted Representations

OLS generalizes to the heteroscedasticity cases (unequal variances of the observations) or cases with correlated observations under the form of Generalized Least Squares (GLS). Assuming that $y = X\beta_{GLS} + \epsilon$, $E[\epsilon|X] = 0$ and $Var[\epsilon|X] = \Omega$, GLS estimates β_{GLS} by minimizing the squared Mahalanobis length of the residuals:[1]

$$\hat{\beta}_{GLS} = \arg\min_{\beta} (y - X\beta)^T \Omega^{-1/2} (y - X\beta) \quad (7)$$

which leads to the explicit formula:

$$\hat{\beta}_{GLS} = (X^T \Omega^{-1/2} X)^{-1} X^T \Omega^{-1/2} y \quad (8)$$

When Ω is a diagonal matrix, GLS boils down to Weighted Least Squares (WLS).

When Ω is not directly known, it can be estimated as in Feasible Generalized Least Squares (FGLS) [2]. First, use OLS and obtain the residuals u, and take for Ω the diagonal matrix of squared residuals:

$$\hat{u}_{OLS} = y - X\hat{\beta}_{OLS},\ \Omega_{OLS} = diag(\hat{u}_{OLS})^2 \quad (9)$$

Then $\hat{\beta}_{FGLS_1}$ is estimated:

$$\hat{\beta}_{FGLS_1} = (X^T \Omega_{OLS}^{-1/2} X)^{-1} X^T \Omega_{OLS}^{-1/2} y \quad (10)$$

[1] Ω^{-1} from the ICPR published version is corrected to $\Omega^{-1/2}$.

leading to

$$\hat{u}_{FGLS_1} = y - X\hat{\beta}_{FGLS_1},\ \Omega_{FGLS_1} = diag(\hat{u}_{FGLS_1})^2$$

$$\hat{\beta}_{FGLS_2} = (X^T \Omega_{FGLS_1}^{-1/2} X)^{-1} X^T \Omega_{FGLS_1}^{-1/2} y \quad (11)$$

Under certain assumptions, Ω_{FGLS} will converge after a number of iterations.

Ridge Regression (RR), also known as Tikhonov regularization, solves:

$$\hat{\beta}_{RR} = \arg\min_{\beta} \|y - X\beta\|^2 + \|\Gamma_{RR}\beta\|^2$$

$$\hat{\beta}_{RR} = (X^T X + \Gamma_{RR}^T \Gamma_{RR})^{-1} X^T y \quad (12)$$

where $\Gamma_{RR} \in \mathbb{R}^{m \times m}$ is the Tikhonov matrix, suitably chosen to alleviate ill-posed problems. If Γ_{RR} is null or a scaled identity matrix then RR boils down to OLS or CR, respectively.

A Generalized Weighted Collaborative Representation (WCR) can be defined as:

$$\hat{\beta}_{WCR} = \arg\min_{\beta} [(y - X\beta)^T \Omega_{WCR}^{-1/2} (y - X\beta) + \|\Gamma_{WCR}\beta\|^2]$$

$$\hat{\beta}_{WCR} = (X^T \Omega_{WCR}^{-1/2} X + \Gamma_{WCR}^T \Gamma_{WCR})^{-1} X^T \Omega_{WCR}^{-1/2} y \quad (13)$$

Thus, for WCR, Ω_{WCR} modulates the importance of each dimension similarly to FGLS, and Γ_{WCR} encodes the importance of each sample in the solution similarly to RR.

Adding sparsity regularization to WCR brings us to a Generalized Weighted Elastic Net (WEN) formulation:

$$\hat{\beta}_{WEN} = \arg\min_{\beta} (y - X\beta)^T \Omega_{WEN}^{-1/2} (y - X\beta) + \|\Gamma_{WEN}\beta\|^2 + \|\Lambda_{WEN}\beta\|_1 \quad (14)$$

where Λ_{WEN} mitigates the importance of each sample.

4 Classification

The information used for classification usually is the residual corresponding to each class c [8]:

$$r_c(y) = \|y - X_c\hat{\beta}_c\| \quad (15)$$

where $\hat{\beta}_c$ and X_c are the coefficients and samples corresponding to class c from the full representation of y defined by the coefficients $\hat{\beta}$ and the training samples X. And the classification decision is taken using:

$$class(y) = \arg\min_{c} r_c(y). \quad (16)$$

If in eq. (15) $\hat{\beta} = \hat{\beta}_{SR}$, then the resulting decision is the Sparse Representation-based Classifier (SRC) decision.

Another, faster, approach is to directly use the weights in absolute values as deciding information [6]. Thus,

$$w_c(y) = \|\hat{\beta}_c\|_1,\ class(y) = \arg\max_c w_c \quad (17)$$

If in eq. (17) $\hat{\beta} = \hat{\beta}_{SR}$, then the resulting decision is the SRC$_w$ decision based on coefficients.

For a Collaborative Representation Classifier with Regularized Least Squares (CRC) [10] we use eq. (4)

$$\hat{\beta}_{CR} = Py,\ P = (X^T X + \lambda_{CR} I)^{-1} X^T \quad (18)$$

and the regularized residuals are taken as:

$$r_c(y) = \|y - X_c\hat{\beta}_c\| / \|\hat{\beta}_c\| \quad (19)$$

The CRC decision is taken as in eq. (16). P does not depend on the query y and can be precomputed. This brings a large computational advantage of CRC over SRC which runs a query-dependent optimization.

When the number of data samples (X) exceeds data dimensionality, the computation of P can be troublesome. A solution comes from the Moore-Penrose pseudoinverse [4]: one can work on the transposed data in order to compute the pseudoinverse

$$P = ((XX^T + \lambda_{CR} I)^{-1} X)^T \quad (20)$$

Adapting the computation of P using eq. (20) or eq. (18), allows CRC to scale well with either very large datasets or a very high dimensionality of the data.

For WCR based Classification (WCRC) we use the regularized residuals approach as in CRC. The P is:[2]

$$P = (X^T \Omega_{WCR}^{-1/2} X + \lambda_{WCR}(\kappa_1 I + \kappa_2 \Gamma_{WCR}^T \Gamma_{WCR}))^{-1} X^T \Omega_{WCR}^{-1/2} \quad (21)$$

Ω_{WCR} can be estimated using the procedure of FGLS, where Γ_{WCR} is fixed. If there is sufficient data, then Ω_{WCR} is estimated as the variance from the training data, as we do for our experiments. Γ_{WCR}, in our case, is learned using the training data and cumulating the evidence per each sample for correct and incorrect classification participation. Thus, the weights are correlated with the WCRC classifier decision.

Let the class of the i-th training sample/column of X be t_i. For each training sample $\mathbf{x}_i$ we cumulate the corresponding i-th coefficients $\hat{\beta}_i(\mathbf{x}_j)$ of the representations for all training samples $\mathbf{x}_j$, computed using eq. (21) with $\Gamma_{WCR} = I, \kappa_2 = 0$. For the samples sharing the same class with $\mathbf{x}_i$ we cumulate in θ_i^+, otherwise in θ_i^-:

$$\theta_i^+ = \sum_{j=1}^{m} [t_j = t_i] \hat{\beta}_i^2(\mathbf{x}_j),\ \theta_i^- = \sum_{j=1}^{m} [t_j \neq t_i] \hat{\beta}_i^2(\mathbf{x}_j) \quad (22)$$

[2] P is corrected from the ICPR published version.

The weights are taken as:[3]

$$\Gamma_{WCR} = diag([\frac{\theta_1^+}{\theta_1^+ + \theta_1^-}, \cdots, \frac{\theta_m^+}{\theta_m^+ + \theta_m^-}]^{\frac{1}{2}}) \quad (23)$$

We reduce the working parameters to a single one, λ_{WCR}, by empirically setting κ_2 as the mean value of $X^T \Omega_{WCR}^{-1/2} X$, and κ_1 to 0.1 from this value, respectively.

5 Experimental results

5.1 Benchmark setup

In our experiments we use the AR face database [3] with the same settings as in [10, 7]. There are 100 individuals for a total of 700 training and 700 testing face images of size 60×43. Complementary experiments are conducted on GTSRB traffic signs database [5]. This is much larger with its 43 classes, 39209 training and 12630 testing images. All the features are l_2 normalized before and after projections in all experiments.

SRC uses either the Feature Sign (FeSg) [1], Homotopy (Hmtp) or L1LS algorithm for solving the l_1 minimization.

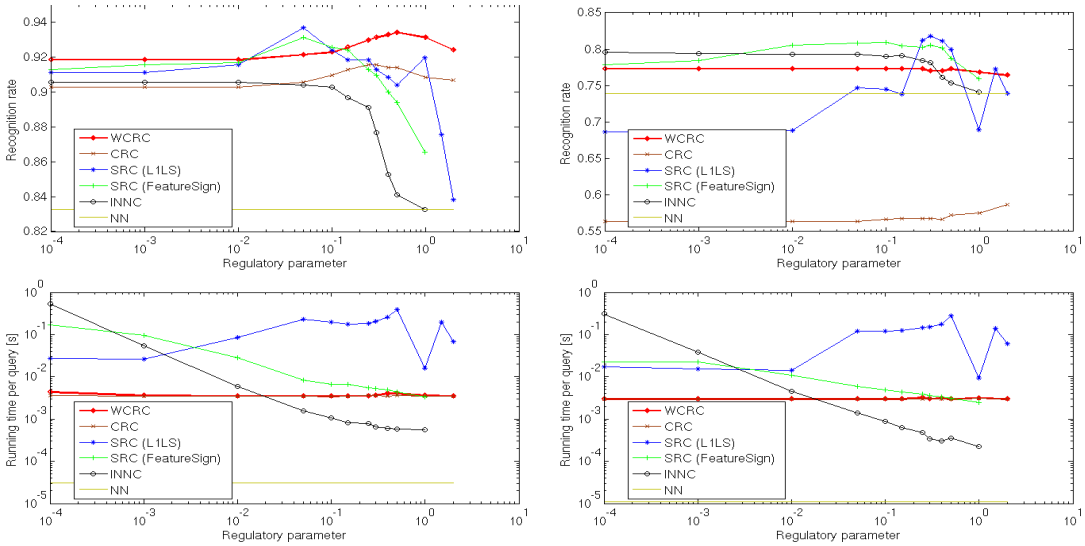
5.2 l_1, l_2 and data dimensionality

First, we study the role of l_1 and l_2 regularization versus the dimensionality of the data and regulatory parameter. We use the AR face database [3] as in [10, 7] and apply regularized ($\lambda = 0.001$) LDA projections. Fig. 1 depicts results for a low, 20-dimensional LDA embedding and a higher, 70-dimensional one versus the regulatory parameter for all the considered classifiers, *i.e.* λ_{WCR} is the parameter of WCRC.

The l_2 regularization works very well for high-dimensional data, while for low-dimensional data the l_1 regularization is much more effective. The weighting (of both samples and channels) usually improves the results over those of the flat formulations (initial formulations corresponding to equal uniform weights). Fig. 2 gives an overview of such improvements for the AR dataset where the features are the grayscale values of downsampled images. The effect of the combined weighting is biggest for low dimensions where the contribution of sample weighting (when $\Omega_{WCR} = I$) is dominant. At higher dimensions, improvement via channel weighting (when $k_2 = 0$) takes over, but is small. WCRC reaches 96%, 2% better than the best CRC results.

The WCRC result on AR is similar to the one recently reported by [9] for the Relaxed Collaborative Representation (RCR) Classifier (RCRC), 96.0%-WCRC vs. 95.9%-RCRC. Instead of moving from CR, eq. (4), to a WCR weighted formulation as we do in

[3] $\Gamma_{WCR}, \kappa_1, \kappa_2$ are corrected from the ICPR published version.



a) 70 dim. LDA projection

b) 20 dim. LDA projection

Figure 1. Parameter influence (on AR).

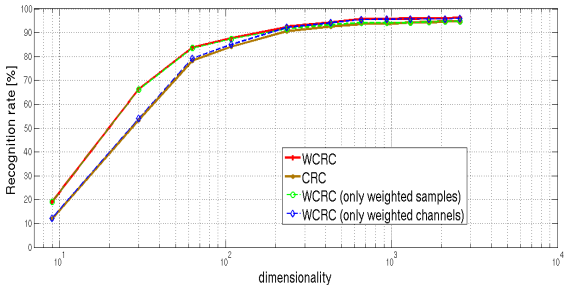


Figure 2. Dimensionality influence (AR).

eq. (21), RCRC introduces an extra regularization term individually weighting each coefficient. The solution for RCR is obtained through numeric iterations, while in our case, WCR still has a direct algebraic solution. Under the same conditions, using Matlab scripts and tested on the AR dataset, starting from 60×43 grayscale pixel values, RCRC is more than 2.5 times slower than our WCRC when we run the scripts provided by the authors.

5.3 Classification

The classification performance of the least squares based classifiers was tested on the AR face database [3] and the GTSRB traffic signs database [5], for different types of projections, as in [7]. We keep the settings from [7]. For the GTSRB results from Table 2, we use 300-dimensional eigenfaces, 99-dimensional Sparse Representation based Linear Projections (SRLP) [6] and the Iterative Nearest Neighbors based Linear Projections (INNLP) [7], and 42-dimensional regularized LDA projections ($\lambda = 0.05$). We compare also with Nearest Neighbor (NN) and Linear Support Vector Machines (LSVM).

As shown for AR, in Table 1, WCRC gets on par top performance with SRC(L1LS) for 99- up to 300-dimensional eigenfaces on AR. However for lower dimensionalities, while better than CRC, WCRC is still below the performance of Iterative Nearest Neighbors Classifier (INNC) [7], SRC with FeSg or Hmtp solvers, and even below NN for very low-dimensional eigenface embeddings. Similar WCRC behavior is observed for the discriminant projections (LDA, SRLP, INNLP): WCRC improves over CRC and gets on par top performance with SRC(FeSg) for high-dimensional embeddings, while in the lower range WCRC consistently performs below INNC, SRC(FeSg) and even NN.

When we run on GTSRB (see Table 2), WCRC is ahead of CRC, but both methods are below the top SRC(FeSg) and INNC classifiers, except in the case where we operate on high-dimensional eigenfaces. GTSRB is a database 2 orders of magnitude larger than AR. This fact combined with the relatively low-dimensional embeddings of the LDA, SRLP and INNLP projections as used here, seem to not accommodate W/CRC well. When looking at AR vs. GTSRB, side by side, we see the importance of the sparsity regularization (l_1) that helps in obtaining least squares decompositions that are meaningful at class level when there is a large pool of data (GTSRB), while for smaller pools, l_2 is sufficient. The WENC strikes a balance between sparse and collaborative solutions and is expected to be more robust than WCRC and SRC, but at the price of increased time complexity. However, while theoretically better, here we do not investigate the WENC classifier, which is particularly slow.

WCRC generally outperforms the original CRC for-

Table 1. Face recognition [%] on AR.

	Dim	5	10	30	54	99	120	300
eigenfaces	WCRC	08.2	27.5	68.8	**83.1**	**89.5**	**91.0**	**93.7**
	CRC	06.3	19.5	64.2	80.3	89.3	90.1	**93.8**
	SRC(L1LS)	06.2	19.6	64.7	81.0	**90.0**	**91.4**	**93.4**
	SRC(FeSg)	23.3	46.1	**70.8**	76.7	80.4	81.0	82.7
	SRC(Hmtp)	23.3	**48.6**	69.7	75.1	77.4	78.1	79.8
	INNC	**24.1**	**48.4**	68.8	74.1	77.1	77.7	79.4
	NN	23.5	43.4	59.1	68.1	69.8	70.4	71.4
	LSVM[10]				69.4		74.7	75.4
	Dim	5	10	30	54	99	120	300
LDA	WCRC	20.6	46.5	**86.8**	**92.3**	93.1		
	CRC	09.6	26.0	75.5	90.7	91.0		
	SRC(L1LS)	20.0	47.9	86.0	**92.4**	**94.9**		
	SRC(FeSg)	34.5	56.8	**87.1**	**92.1**	**94.4**		
	INNC	**37.5**	**60.8**	**86.6**	88.8	92.6		
	NN	**37.2**	58.8	76.7	81.8	87.0		
	Dim	5	10	30	54	99	120	300
SRLP_{FeSg}	WCRC	21.1	47.6	**86.9**	91.0	**93.6**	**94.3**	**94.6**
	CRC	09.9	27.3	77.0	90.0	91.8	92.4	88.4
	SRC(L1LS)	18.7	40.9	81.8	91.3	93.1	93.7	**94.4**
	SRC(FeSg)	34.9	61.4	86.3	**92.0**	**94.0**	**94.3**	**94.4**
	INNC	**39.4**	**62.3**	85.3	88.6	92.4	93.0	93.1
	NN	37.8	59.8	77.3	82.8	86.0	86.4	86.7
	Dim	5	10	30	54	99	120	300
INNLP	WCRC	21.6	49.6	**87.1**	91.1	**93.7**	**94.1**	**94.3**
	CRC	09.7	27.6	76.3	89.6	91.7	92.6	88.0
	SRC(L1LS)	18.3	39.8	81.4	90.7	93.0	**93.7**	**94.4**
	SRC(FeSg)	34.9	60.0	86.1	**92.3**	**94.1**	**94.0**	**94.1**
	INNC	**37.3**	**60.8**	85.1	89.0	92.7	93.1	93.0
	NN	**37.3**	59.7	77.4	82.7	86.0	86.1	86.9

Table 2. Recognition [%] on GTSRB.

classifier	eigenf.	SRLP_{FeSg}	INNLP	SRLP_{Hmtp}	LDA
NN	66.05	89.54	89.35	89.27	91.84
WCRC	**86.12**	87.17	86.99	86.86	89.23
CRC	84.34	83.43	83.56	82.91	84.08
INNC	77.70	**93.64**	**93.61**	**93.61**	**93.64**
SRC(FgSg)	85.31	**93.94**	**93.74**	92.93	92.91
SRC(L1LS)	74.78	79.34	79.41	93.13	93.01
LSVM		87.87		87.38	87.00
IKSVM		89.51		89.66	86.30
RBFSVM		92.43		92.28	92.46

mulations for all the settings. Out of these top methods WCRC (and CRC) admits an algebraic solution and is faster than the SRC variants, but slower than NN and INNC, see Tables 1, 2 and 3.

5.4 Running time

We now compare the running times for these methods. The recognition rates and times per query are listed in Table 3 for the AR face database with 300-dimensional eigenface embeddings and for the GTSRB traffic sign database with 42-dimensional LDA embeddings. The influence of the data dimensionality and regulatory parameter is depicted also in Figs. 1 and 2. W/CRC is very fast, orders of magnitude faster than the SRC formulation. Moreover, INNC [7] is faster than W/CRC but has a poorer performance for high-dimensional data.

6 Conclusions

This paper reviewed the current least squares based representations and investigated the impact of adding weights. Thus, we investigated the Weighted Collaborative Representation (WCR), revealing strong points and weaknesses for the task of image classification. WCR inherits the simplicity and the effectiveness of the CR scheme, while also having an algebraic solution.

The methods have been validated on face and traffic sign datasets. Out of these experimental results emerges the following picture: for the lowest dimensions one can best use INNC, for somewhat higher dimensions SRC (Feature Sign) is the best performer, then to be replaced for high dimensions by WCRC as the method of choice.

Table 3. Running times on AR and GTSRB.

	AR w/ Eigenfaces(300)		GTSRB w/ LDA(42)	
classifier	Recog.[%]	Time (s)	Recog.[%]	Time (s)
NN	71.43	0.0001	91.84	0.0009
INNC	79.51	0.0023	**93.64**	**0.0081**
SRC(L1LS)	93.41	0.8187	93.01	2.8524
SRC(FeSg)	93.56	0.0927	92.91	0.1679
CRC	93.76	0.0044	84.08	0.0381
WCRC	**93.72**	**0.0044**	89.23	0.0392

Acknowledgments. This work was partly supported by the European Commission FP7 ICT-269980 AXES project and the IWT/SBO ALAMIRE project.

References

[1] H. Lee, A. Battle, R. Raina, and A. Y. Ng. Efficient sparse coding algorithms. In *NIPS*, 2006.

[2] R. J. A. Little and D. B. Rubin. *Statistical Analysis with Missing Data (2nd ed.)*. John Wiley & Sons, Inc., 2002.

[3] A. Martinez and R. Benavente. The AR face database. Technical report, CVC Tech. Report No. 24, 1998.

[4] R. Penrose. A generalized inverse for matrices. *Mathematical Proceedings of the Cambridge Philosophical Society*, 51:406–413, 1955.

[5] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel. The German Traffic Sign Recognition Benchmark: A multi-class classification competition. In *IJCNN*, 2011.

[6] R. Timofte and L. Van Gool. Sparse representation based projections. In *BMVC*, 2011.

[7] R. Timofte and L. Van Gool. Iterative nearest neighbors for classification and dimensionality reduction. In *CVPR*, 2012.

[8] J. Wright, A. Y. Yang, A. Ganesh, S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *PAMI*, 31(2), February 2009.

[9] M. Yang, L. Zhang, D. Zhang, and S. Wang. Relaxed collaborative representation for pattern classification. In *CVPR*, 2012.

[10] L. Zhang, M. Yang, and X. Feng. Sparse representation or collaborative representation: Which helps face recognition? In *ICCV*, 2011.

[11] H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *JRSSB*, 67(2):301–320, 2005.