Multi-Label Classification for Image Annotation via Sparse Similarity Voting

Tomoya Sakai[1], Hayato Itoh[2], and Atsushi Imiya[3]

[1] Faculty of Engineering, Nagasaki University, Japan
tsakai@ieee.org
[2] Graduate School of Science and Technology, Chiba University, Japan
hayato-itoh@graduate.chiba-u.jp
[3] Institute of Media and Information Technology, Chiba University, Japan
imiya@faculty.chiba-u.jp

Abstract. We present a supervised multi-label classification method for automatic image annotation. Our method estimates the annotation labels for a test image by accumulating similarities between the test image and labeled training images. The similarities are measured on the basis of sparse representation of the test image by the training images, which avoids similarity votes for irrelevant classes. Besides, our sparse representation-based multi-label classification can estimate a suitable combination of labels even if the combination is unlearned. Experimental results using the PASCAL dataset suggest effectiveness for image annotation compared to the existing SVM-based multi-labeling methods. Nonlinear mapping of the image representation using the kernel trick is also shown to enhance the annotation performance.

1 Introduction

This paper addresses multi-label classification for annotating images of multiple objects. Multi-labeling is a fundamental functionality of a multi-class classifier for the automatic image annotation. The classifier is required to assign multiple labels of objects to an image of those objects.

Prior Work on Multi-class Classification and Multi-Labeling A popular approach to the image-based object recognition and annotation is to employ a discriminative model using bag-of-features image representation [1] in learning and labeling phases. One-vs-rest SVM [2, 3] and one-vs-one SVM [4] consist of two-class SVM classifiers, each of which learns a margin between object classes. A test image to be annotated, however, has mixture of features of multiple objects in it. The two-class classifiers have to be able to discriminate the individual objects by the mixture. Multi-label ranking (MLR) [5] fixes this problem by simultaneously learning from multi-label data so as to minimize the classification error for all classes in total. MLR is shown to outperform the state-of-the-art multi-labeling SVM algorithms in the bag-of-features image classification task, but its performance for test images with unlearned combinations of labels is not guaranteed.

The image annotation based on multi-label classification is essentially a problem of finding a combination of learned objects whose features can synthesize the mixture of features of a test image. An important fact is that among the learned classes a few of them are relevant to a test image. Sparse representation-based classification (SRC) [6] takes advantage of this fact by representing a test image as a sparse linear combination of training images. The SRC achieves robust single-labeling for face recognition. For the image annotation task, Wang *et al.* [7] proposed multi-label sparse coding (MSC) in the same manner as the SRC together with linear embedding into a discriminative space learned from the training images and their sparse labels. Hsu *et al.* [8] have exploited the sparsity of the classifier output by the compressed sensing technique [9–13] for reducing computational expense of multi-label classification with linear regression.

Our Method In this paper, we propose a substantial method of multi-labeling on the basis of the sparse representation and accumulation of similarities. Our method consists of the following steps:

Sparse representation: explain concisely the test image by the training images, i.e., find sparse coefficients $\hat{\alpha}_j$ such that

$$\boldsymbol{\phi}(\text{test image}) \approx \textstyle\sum_j \hat{\alpha}_j \boldsymbol{\phi}(j\text{-th training image})$$

where $\boldsymbol{\phi}$ indicates a high dimensional representation of the input image, e.g., a histogram of visual words.

Similarity measurement: compute similarities

$$w_j \sim \hat{\alpha}_j \, \kappa \, (j\text{-th training image}, \text{test image})$$

where κ calculates an inner product.

Voting: classes indicated by the labels of the j-th training image receive the votes of w_j.

Preliminary details of the sparse representation are provided in Section 2. Differing from the existing multi-label methods exploiting sparsity, our method does not use the labels of training images for the computation of the sparse coefficients $\hat{\alpha}_j$. While the use of the labels in the training phase would refine the classification performance for a test image to give a learned combination of labels, it could degrade the generalization capabilities of the sparse representation for most of the label combinations unlearned in practice. After the sparse representation, our method measures the similarities because we must not assemble the output labels by directly using the coefficients $\hat{\alpha}_j$ as done in the MSC. We also introduce the kernel trick to improve the classification performance. Our algorithms and the kernelization are described in Section 3. We experimentally show the ability to find unlearned label combinations as well as the outperformance of our method in Section 4.

2 Sparse Representation for Multi-Labeling

2.1 Multi-Class Classification and Multi-Labeling

Multi-label classification is a task of assigning a suitable number of class labels to unlabeled test data. A training dataset $S \subset \mathbb{R}^d$ with a collection of labels $Y \subset \{0,1\}^l$ is available for the classification. The labels of a training data $\boldsymbol{s}_j \in \mathcal{S}$ are represented as a binary vector $\boldsymbol{y}_j = [y_1, \ldots, y_l]^\top$ where $y_i \in \{0,1\}$.

The binary classification is the case of $l = 1$, and the case of $l > 1$ is known as the multi-class classification. In the prediction of a label $\hat{\boldsymbol{y}} \in \{0,1\}^l$ for a given test data $\boldsymbol{x} \in \mathbb{R}^d$, the multi-class classification under the constraint $||\hat{\boldsymbol{y}}||_0 \leq 1$ is called the single-labeling. Here, $|| \cdot ||_0$ denotes the l^0 norm, which counts the nonzero components. The multi-class classification without the constraint is the multi-labeling. There are possibly 2^l combinations of labels.

2.2 Sparse Representation of Test Data

Let $\mathbf{S} \in \mathcal{R}^{d \times n}$ be a matrix of d-dimensional n column vectors of training data $\boldsymbol{s}_j$, and let $\mathbf{Y} \in \{0,1\}^{l \times n}$ be the matrix with corresponding label vectors $\boldsymbol{y}_j$ in its columns. Supposing the linear vector space model and given an enough number of training data, one can represent a test data $\boldsymbol{x} \in \mathbb{R}^d$ as a linear combination of the vectors of training data.

$$\boldsymbol{x} = \sum_{j=1}^{n} \alpha_j \boldsymbol{s}_j = \mathbf{S}\boldsymbol{\alpha} \tag{1}$$

Here, $\boldsymbol{\alpha} \in \mathbb{R}^n$ is the vector of n combination coefficients α_j to be estimated.

The solution $\boldsymbol{\alpha}$ to Equ. (1) exists if the test data $\boldsymbol{x}$ lies in $\operatorname{span} \mathbf{S}$, i.e., the subspace spanned by the training data. We would like to assign labels to the test data according to the solution to Equ. (1). If no solution exists, one should not assign any label, i.e., $\hat{\boldsymbol{y}} = \mathbf{0}$. This is the case where the training dataset is insufficient for representing the test data. If a sufficient number of training data are given, Equation (1) has non-unique solutions. We require regularization to select a unique solution. From the viewpoint of classification, a test data should be concisely explained by relevant training data. A sparse solution whose nonzero components indicate a few relevant classes to the test data would be preferable.

Finding a sparse solution is formulated as a l^0-minimization problem:

$$\min ||\boldsymbol{\alpha}||_0 \quad \text{subject to} \quad \boldsymbol{x} = \mathbf{S}\boldsymbol{\alpha}. \tag{2}$$

The l^0-minimization is a NP-hard problem, which is often relaxed to a convex problem:

$$\min ||\boldsymbol{\alpha}||_1 \quad \text{subject to} \quad \boldsymbol{x} = \mathbf{S}\boldsymbol{\alpha}. \tag{3}$$

One can find literature on the uniqueness of the sparse solution and on the equivalence between the l^0- and l^1-minimization problems [12, 14, 15]. The uniqueness of the solution, for example, is guaranteed under the condition called the restricted isometry property (RIP). The RIP condition with parameters (m, δ) for a matrix $\mathbf{\Theta}$ is described as

$$(1-\delta)||\boldsymbol{\beta}||_2 \leq ||\mathbf{\Theta}\boldsymbol{\beta}||_2 \leq (1+\delta)||\boldsymbol{\beta}||_2 \quad \forall \boldsymbol{\beta} \in \left\{ \boldsymbol{b} \,\middle|\, ||\boldsymbol{b}||_0 \leq m \right\} .$$

A vector $\boldsymbol{b}$ is called m-sparse if $||\boldsymbol{b}||_0 \leq m$. It is known that the l^0-minimization problem (2) has a unique m-sparse solution if the matrix $\mathbf{S}$ satisfies the RIP condition with $(2m, \delta < 1)$. The m-sparse solution is equivalent to the l^1-minimizer for (3) if $\mathbf{S}$ satisfies the RIP condition with $(2m, \delta < \sqrt{2}-1)$ [12].

2.3 Dimensionality Reduction

One can reduce the computational cost of dealing with high-dimensional training and test data by linear projection. The compressed sensing methodology shows that a small number of projections of a high-dimensional vector can contain salient information about its sparse representation enough to recover it with regularization that promotes sparsity [9, 11, 16]. Random projection is known to be a universal way of dimensionality reduction.

Let $\mathbf{R}$ be a $d_c \times d$ random matrix. A training dataset $\mathbf{S}$ and a test data $\boldsymbol{x}$ are compressed by random projection as $\boldsymbol{x}_c = \mathbf{R}\boldsymbol{x} \in \mathbb{R}^{d_c}$ and $\mathbf{S}_c = \mathbf{R}\mathbf{S} \in \mathbb{R}^{d_c \times n}$. Equation (1) is rewritten as $\boldsymbol{x}_c = \mathbf{S}_c\boldsymbol{\alpha}$. It is known that the m-sparse vector $\boldsymbol{\alpha}$ can be reconstructed from $\boldsymbol{x}_c$ with probability $1 - e^{-\mathcal{O}(d_c)}$ by the sparse regularization if $d_c \geq d_0 = \mathcal{O}(m \log(d/m))$ [17, 18].

2.4 Multi-Label Estimation by Similarity Voting

We describe how to assign labels to a test data via sparse representation. Let $\hat{\boldsymbol{x}}$ be a reconstructed test data using the training data matrix $\mathbf{S}$ and a sparse solution $\boldsymbol{\alpha} = \hat{\boldsymbol{\alpha}}$.

$$\hat{\boldsymbol{x}} = \mathbf{S}\hat{\boldsymbol{\alpha}}$$

We measure the similarity between the test data $\boldsymbol{x}$ and its reconstruction $\hat{\boldsymbol{x}}$ as

$$\cos\theta = \frac{\boldsymbol{x}^\top \hat{\boldsymbol{x}}}{||\boldsymbol{x}||_2 ||\hat{\boldsymbol{x}}||_2} = \frac{\boldsymbol{x}^\top \mathbf{S}\hat{\boldsymbol{\alpha}}}{||\boldsymbol{x}||_2 ||\hat{\boldsymbol{x}}||_2} = \sum_{j=1}^{n} w_j .$$

Here,

$$w_j = \frac{\hat{\alpha}_j \boldsymbol{s}_j^\top \boldsymbol{x}}{||\boldsymbol{x}||_2 ||\hat{\boldsymbol{x}}||_2} \tag{4}$$

is the similarity between the test data and the j-th component of the reconstructed test data on the basis of training data. Note that $\boldsymbol{w} = [w_1, \ldots, w_n]^\top$ is as sparse as $\hat{\boldsymbol{\alpha}}$. Regarding w_j as the partial membership value for a combination of classes labeled as $\boldsymbol{y}_j$, we estimate the multi-label $\hat{\boldsymbol{y}}$ for the test data by accumulating the labels as

$$\hat{\boldsymbol{y}} = \sum_{j=1}^{n} w_j \boldsymbol{y}_j = \mathbf{Y}\boldsymbol{w} .$$

This accumulation is interpreted as label voting with the weight w_j. One can determine the labels for the test data by thresholding or ranking the magnitudes of the vector components of $\hat{\boldsymbol{y}}$.

3 Algorithms

3.1 Multi-Label Classification

Our multi-labeling algorithm is summarized in Algorithm 1.

```
Algorithm 1 Multi-label classification (main algorithm in linear case)
Input: $\boldsymbol{x} \in \mathbb{R}^d$: test data,  $\mathbf{S} \in \mathbb{R}^{d \times n}$: matrix of training data,  $\mathbf{Y} \in \{0,1\}^{l \times n}$: matrix
    of labels;
1   normalize the columns of $\mathbf{S}$ to have unit $l^2$ norm;
2   perform dimensionality reduction of $\mathbf{S}$ and $\boldsymbol{x}$ if the dimensionality $d$ is intractably
    high;
3   decompose $\boldsymbol{x}$ with respect to $\mathbf{S}$ under sparse regularization to obtain the sparse
    solution $\hat{\boldsymbol{\alpha}}$;
4   compute the similarities $\boldsymbol{w} = [w_1, \dots, w_n]^\top$;
Output: $\hat{\boldsymbol{y}} \leftarrow \mathbf{Y}\boldsymbol{w}$: label estimates.
```

The classification does not involve any expensive computation for training. We do not have to solve a quadratic programming problem like support vector machines or an eigenvalue problem for subspace methods. Algorithm 1 can start testing soon after loading the training data. It is therefore easy to append and remove the data before testing if necessary. We would also remark that Algorithm 1 can answer unlearned combinations of labels when the relevant training data can sparsely represent the test data.

3.2 Sparse Decomposition

There are basically two types of algorithms for solving the minimization problem (2). One is called the basis pursuit (BP) [19], which relaxes the l^0 to l^1 minimization problem. Linear programming can solve the l^1 minimization problem in (3). One can find some algorithms [20–23] for the related convex problems

$$\min ||\boldsymbol{x} - \mathbf{S}\boldsymbol{\alpha}||_2 \text{ subject to } ||\boldsymbol{x}||_1 \leq \tau \tag{5}$$

$$\min ||\boldsymbol{\alpha}||_1 \quad \text{subject to } ||\boldsymbol{x} - \mathbf{S}\boldsymbol{\alpha}||_2 \leq \varepsilon \tag{6}$$

to obtain robust solution against noise.

The other type is the greedy algorithms [24–27], which greedily seek for the nonzero components. Matching pursuit (MP) [28] selects a column vector $\boldsymbol{s}_j$ in $\mathbf{S}$ which is most coherent to the residual of $\mathbf{x}$, and removes from the residual the component in the direction of $\boldsymbol{s}_j$, iteratively. Orthogonal matching pursuit (OMP) [24] instead removes the component in the subspace spanned by previously selected column vectors. Regularized orthogonal matching pursuit (ROMP) [26] is guaranteed to recover any m-sparse solution for a matrix satisfying the RIP condition with $(2m, 0.03/\sqrt{\log m})$. The greedy algorithms are very simple to implement and faster than BP. In this paper, we employ ROMP.

3.3 Kernelization

The above formulation assumes the linear relationship as in Equ. (1). Although Algorithm 1 can benefit from the sparsity of the linear representation, we would like to translate our framework into a nonlinear version hoping to improve the classification performance. We map the data in the nonlinear input space $\mathbb{R}^d$ to an Affine space using a nonlinear function $\boldsymbol{\phi}$, assuming the linear relationship between training data and test data as

$$\boldsymbol{\phi}(\boldsymbol{x}) = \sum_{j=1}^{n} \alpha_j \boldsymbol{\phi}(\boldsymbol{s}_j). \tag{7}$$

We apply the kernel trick using a kernel function $\kappa(\mathbf{x}, \boldsymbol{y}) = \boldsymbol{\phi}(\boldsymbol{x})^\top \boldsymbol{\phi}(\boldsymbol{y})$ and kernel matrix $\mathbf{K}(\mathbf{X}_1, \mathbf{X}_2) \in \mathbb{R}^{n_1 \times n_2}$ whose ij-th entry is the inner product of the i-th and j-th column vectors of the matrices $\mathbf{X}_1 \in \mathbb{R}^{d \times n_1}$ and $\mathbf{X}_2 \in \mathbb{R}^{d \times n_2}$, respectively.

Algorithm 2 Kernelized ROMP

```
Input: $\boldsymbol{x} \in \mathbb{R}^d$: test data, $\mathbf{S} \in \mathbb{R}^{d \times n}$: matrix of training data, $m_0$: sparsity level, $\varepsilon_0$:
  tolerance;
1  initialize $\mathcal{I} \leftarrow \emptyset$ and $\hat{\boldsymbol{\alpha}} \leftarrow \mathbf{0}$;
2  repeat
3     $\boldsymbol{u} \leftarrow \mathbf{K}(\mathbf{S}, \boldsymbol{x}) - \mathbf{K}(\mathbf{S}, \mathbf{S}_{\mathcal{I}})\hat{\boldsymbol{\alpha}}_{\mathcal{I}}$;
4     $\boldsymbol{\gamma} \leftarrow [|u_1|, \ldots, |u_n|]^\top$;
5     let $\mathcal{J}$ be a set of indices of the $m_0$ biggest components of $\boldsymbol{\gamma}$, or all of its nonzero
      components, whichever set is smaller;
6     sort $\mathcal{J}$ in descending order of the components $\boldsymbol{\gamma}$;
7     among all subsets $\mathcal{J}_0 \subset \mathcal{J}$ such that $\gamma_i \leq 2\gamma_j$ for all $i < j \in \mathcal{J}_0$, choose $\mathcal{J}_0$ with
      the maximal energy $||\gamma_{\mathcal{J}_0}||_2^2 = \sum_{k \in \mathcal{J}_0} \gamma_k^2$;
8     $\mathcal{I} \leftarrow \mathcal{I} \cup \mathcal{J}_0$;
9     $\hat{\boldsymbol{\alpha}}_{\mathcal{I}} \leftarrow \arg\min_{\boldsymbol{\alpha}_{\mathcal{I}}} ||\boldsymbol{r}(\boldsymbol{\alpha}_{\mathcal{I}})||_2^2$;
10 until $||\boldsymbol{r}(\hat{\boldsymbol{\alpha}_{\mathcal{I}}})||^2/||\boldsymbol{x}||_2 \leq \varepsilon_0$ or card$\mathcal{I} \geq 2m_0$;
Output: $\hat{\boldsymbol{\alpha}}$: sparse solution.
```

We present a kernelized version of ROMP for nonlinear structure of the input space. The kernelized ROMP is described as Algorithm 2. The vector $\boldsymbol{\alpha}_{\mathcal{I}}$ indicates a vector with the components of $\boldsymbol{\alpha}$ specified by $\mathcal{I}$. At Step 9, one can easily obtain $\hat{\boldsymbol{\alpha}}_{\mathcal{I}}$ by solving least squares problem without explicitly computing the residual vector $\boldsymbol{r}$, since the squared norm is a quadratic form

$$||\boldsymbol{r}(\boldsymbol{\alpha}_{\mathcal{I}})||_2^2 = \kappa(\boldsymbol{x}, \boldsymbol{x}) - 2\boldsymbol{\alpha}_{\mathcal{I}}^\top \mathbf{K}(\mathbf{S}_{\mathcal{I}}, \boldsymbol{x}) + \boldsymbol{\alpha}_{\mathcal{I}}^\top \mathbf{K}(\mathbf{S}_{\mathcal{I}}, \mathbf{S}_{\mathcal{I}})\boldsymbol{\alpha}_{\mathcal{I}}. \qquad (8)$$

As the ROMP works in linear time with respect to n and d [26], our kernelized ROMP also works in linear time.

After running the kernelized ROMP, the similarities are measured as

$$w_j = \frac{\hat{\alpha}_j \kappa(\boldsymbol{s}_j, \boldsymbol{x})}{\sqrt{\kappa(\boldsymbol{x}, \boldsymbol{x})\hat{\boldsymbol{\alpha}}_{\mathcal{I}}^\top \mathbf{K}(\mathbf{S}_{\mathcal{I}}, \mathbf{S}_{\mathcal{I}})\hat{\boldsymbol{\alpha}}_{\mathcal{I}}}}. \qquad (9)$$

Equation (9) coincides with Equ. (4) if one utilizes the linear kernel $\kappa(\boldsymbol{x}, \boldsymbol{x}) = \boldsymbol{x}^\top \boldsymbol{x}$ and $\mathbf{K}(\mathbf{X}_1, \mathbf{X}_2) = \mathbf{X}_1^\top \mathbf{X}_2$. Algorithm 2 with the linear kernel is also equivalent to the original ROMP.

4 Experiment

Data We apply our multi-label method to image annotation. We used PASCAL VOC 2009 dataset [29]. The VOC 2009 dataset has 3,473 training images and 3,581 validation images of twenty object classes. Each image is annotated by one or more object class labels. We chose the 2,236 training images with single labels as the training data in order to assess the ability to find suitable combinations of labels without using multi-label training data. We randomly selected half of the validation images for tuning the classifier parameters and the other half for testing. A standard bag-of-features model [1] was used to represent the images in this experiment. We extracted SIFT descriptors [30] from every training image in grayscale, and clustered these features into 1,000 clusters by the k-means clustering. Each image was represented as a tf-idf vector.

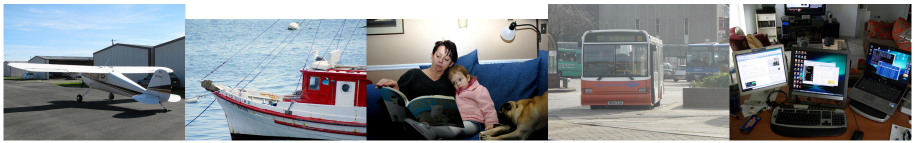
Evaluation and Procedure We characterize the performance of multi-label classification as receiver operating characteristic (ROC) curve and the area under the curve (AUC). Our ROC evaluates the ranking performance: how high the correct labels are ranked. We calculate the true positive ratio (TPR) and false positive ratio (FPR) by changing the number of top labels indicated by the label estimates $\hat{\boldsymbol{y}}$. The same evaluation metric is used for MLR [5]. We did not invoke the dimensionality reduction in Algorithm 1. The input parameters of Algorithm 2 were tuned and set as $m_0 = 35$ and $\varepsilon_0 = 10^{-2}$.

Results Table 1 shows the AUC of rank ROC. Our method provides a comparative AUC to MLR with the linear kernel. The AUC is improved by the kernelization in both methods. Our method with a Gaussian kernel achieves slightly better performance than MLR. MLR has been shown to outperform the existing multi-label SVMs [5]. We deduce from these results that our method is highly effective for the image annotation tasks.

Figure 1 shows some examples of multiply annotated images and annotations by our method with the Gaussian kernel. Note that we used only single-label images for training. We could observe that the relevant object labels are ranked high. Algorithm 1 with MATLAB implementation took about 0.1 (linear) and 0.5 (kernelized) seconds per test image using a CPU single core.

Table 1. AUC of rank ROC for PASCAL VOC 2009.

Kernel	Proposed	MLR
Linear	74.0%	74.1%
Nonlinear	78.1%	76.3%



aeroplane, car	bird, boat	dog, person, sofa	bus, car, person	chair, person, sofa, tvmonitor
aeroplane, car	**bird, boat**	**person**, cat, **sofa**	**bus, car**, train, **person**	**person, tvmonitor, chair**

Fig. 1. Multi-labeling results. First row: test images, second row: ground-truth labels, third row: labels by our method. The true positive labels are in bold.

5 Concluding Remarks

Assigning multiple labels of objects to an unlabeled test image is a problem of finding a combination of learned objects which can synthesize the mixture of features of objects in the test image. We casted this problem as a sparse decomposition of image representation. Our method decomposes the bag-of-features representation of a test image into those of labeled training images as concisely as possible via sparse regularization. This enables us to detect the relevant training images even if all the combinations of objects are not learned from the training images. As suggested in Section 3.1, our method does not have any intensive computation in training. Of course the sparse decomposition for testing requires considerable time, but we have many advantages: easy update of training data, capability to answer unlearned label combinations, and robustness against noise or clutter. We should investigate the performance of our method on large-scale dataset. The performance would be further improved by incorporating co-occurrence statistics of objects and features.

Acknowledgments The first author was partially supported by the Grant-in-Aid for Young Scientists, from the Ministry of Education, Culture, Sports, Science and Technology of Japan under MEXT KAKEN 22700163.

References

1. Csurka, G., Dance, C.R., Fan, L., Willamowski, J., Bray, C.: Visual categorization with bags of keypoints. In: In Workshop on Statistical Learning in Computer Vision, ECCV. (2004)
2. Joachims, T.: Text categorization with support vector machines: learning with many relevant features. In Nédellec, C., Rouveirol, C., eds.: Proceedings of ECML-98, 10th European Conference on Machine Learning, Heidelberg et al., Springer (1998) 137–142
3. Hsu, C.W., Lin, C.J.: A comparison of methods for multiclass support vector machines. Neural Networks, IEEE Transactions on **13** (2002) 415–425
4. Kressel, U.H.G.: Pairwise classification and support vector machines. MIT Press, Cambridge, MA, USA (1999)
5. Bucak, S.S., Mallapragada, P.K., Jin, R., Jain, A.K.: Efficient multi-label ranking for multi-class learning: approach to object recognition. In: In International Conference on Computer Vision. (2009)
6. Wright, J., Yang, A.Y., Ganesh, A., Sastry, S.S., Ma, Y.: Robust face recognition via sparse representation. IEEE Transactions on Pattern Analysis and Machine Intelligence **31** (2009) 210–227
7. Wang, C., Yan, S., Zhang, L., Zhang, H.J.: Multi-label sparse coding for automatic image annotation. Computer Vision and Pattern Recognition, IEEE Computer Society Conference on **0** (2009) 1643–1650
8. Hsu, D., Kakade, S., Langford, J., Zhang, T.: Multi-label prediction via compressed sensing. In: In 23rd Annual Conference on Neural Information Processing Systems. (2009)
9. Donoho, D.: Compressed sensing. IEEE Trans. Information Theory **52** (2006) 1289–1306
10. Candès, E.J., Romberg, J., Tao, T.: Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. IEEE Transactions on Information Theory **52** (2006) 489–509
11. Candès, E.J., Romberg, J., Tao, T.: Stable signal recovery from incomplete and inaccurate measurements. Comm. on Pure and Applied Math **59** (2006) 1207–1223
12. Candès, E.J.: The restricted isometry property and its implications for compressed sensing. Comptes Rendus Mathematique **346** (2008) 589–592
13. Candès, E.J., Wakin, M.B.: An introduction to compressive sampling. IEEE Signal Processing Magazine (March, 2008) 21–30
14. Gribonval, R., Nielsen, M.: Sparse representations in unions of bases. IEEE Transactions on Information Theory **49** (2003) 3320–3325
15. Donoho, D., Elad, M.: Optimally sparse representation in general (non-orthogonal) dictionaries via l^1 minimization. In: Proc. the National Academy of Sciences of the United States of America. (2003) 2197–2202
16. Candès, E.J., Tao, T.: Near-optimal signal recovery from random projections: Universal encoding strategies? IEEE Transactions on Information Theory **52** (2006) 5406–5425

17. Candès, E.J., Tao, T.: Decoding by linear programming. IEEE Transactions on Information Theory **51** (2005) 4203–4215
18. Rudelson, M., Vershynin, R., Rudelson, M., Vershynin, R.: Geometric approach to error correcting codes and reconstruction of signals. Int. Math. Res. Not **64** (2005) 4019–4041
19. Chen, S.S., Donoho, D.L., Saunders, M.A.: Atomic decomposition by basis pursuit. SIAM J. Sci. Comput. **20** (1998) 33–61
20. Tibshirani, R.: Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society, Series B **58** (1996) 267–288
21. Kim, S.J., Koh, K., Lustig, M., Boyd, S., Gorinevsky, D.: An interior-point method for large-scale l_1-regularized least squares. IEEE Journal on Selected Topics in Signal Processing **1** (2007) 606–617
22. Figueiredo, M.A.T., Nowak, R.D., Wright, S.J.: Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems. Selected Topics in Signal Processing, IEEE Journal of **1** (2007) 586–597
23. Tomioka, R., Sugiyama, M.: Dual augmented lagrangian method for efficient sparse reconstruction. Technical report, arXiv:0904.0584 (preprint) (2009)
24. Pati, Y.C., Rezaiifar, R., Rezaiifar, Y.C.P.R., Krishnaprasad, P.S.: Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In: Proceedings of the 27 th Annual Asilomar Conference on Signals, Systems, and Computers. (1993) 40–44
25. Tropp, J.A., Anna, Gilbert, C.: Signal recovery from random measurements via orthogonal matching pursuit. IEEE Trans. Information Theory **53** (2007) 4655–4666
26. Needell, D., Vershynin, R.: Uniform uncertainty principle and signal recovery via regularized orthogonal matching pursuit. Foundations of Computational Mathematic **9** (2009) 317–334
27. Needell, D., Tropp, J.A.: CoSaMP: Iterative signal recovery from incomplete and inaccurate samples. Applied and Computational Harmonic Analysis **26** (2009) 301–321
28. Mallat, S., Zhang, Z.: Matching pursuit with time-frequency dictionaries. IEEE Transactions on Signal Processing **41** (1993) 3397–3415
29. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The Pascal Visual Object Classes (VOC) Challenge. International Journal of Computer Vision **88** (2010) 303–338
30. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision **60** (2004) 91–110