ADAPTIVE FEATURE SPLIT SELECTION FOR CO-TRAINING: APPLICATION TO TIRE IRREGULAR WEAR CLASSIFICATION

Wei Du, Ronald Phlypo, and Tülay Adalı

Department of CSEE, University of Maryland, Baltimore County, MD 21250, USA

ABSTRACT

Co-training is a practical and powerful semi-supervised learning method. It yields high classification accuracy with a training data set containing only a small set of labeled data. Successful performance in co-training requires two important conditions on the features: diversity and sufficiency. In this paper, we propose a novel mutual information (MI) based approach inspired by the idea of dependent component analysis (DCA) to achieve feature splits that are maximally independent between-subsets (diversity) or within-subsets (sufficiency). We evaluate the relationship between the classification performance and the relative importance of the two conditions. Experimental results on actual tire data indicate that compared to diversity, sufficiency has a more significant impact on their classification accuracy. Further results show that co-training with feature splits obtained by the MI-based approach yields higher accuracy than supervised classification and significantly higher when using a small set of labeled training data.

Index Terms— Co-training, semi-supervised classification, feature splits, DCA, LTM tire data

1. INTRODUCTION

Semi-supervised learning has recently attracted much attention in the machine learning field. It is designed to achieve high classification accuracy with reduced effort from experienced human annotators, since only a small size of labeled training data is required.

Among several semi-supervised learning methods [1–3], co-training, as a data-driven method, provides a practical and powerful approach for real-world problems. Co-training is based on the training of two classifiers, each using a subset of the features. First, both classifiers are trained on the available labeled data. Then, the unlabeled data samples with the most confident predictions in one classifier are cross-fed to the other classifier as newly labeled samples on which the training stages can re-iterate. Success in co-training is guaranteed under two important conditions [4]:

Diversity: Features can be split into two sets that are conditionally independent given the class;

Sufficiency: Each subset of features attributed to a single classifier is sufficient to train a good classifier.

Generally, we could achieve high classification accuracy when one of the aforementioned conditions is met while the other is only weakly satisfied [5, 6]. However, in many cases, it is difficult to find sufficiently powerful features that naturally split into two sets. Hence, features are often split into subsets based on heuristic criteria [7–9]. To the best of our knowledge, this is the first time feature splits are obtained based on either the diversity or the sufficiency condition for co-training, allowing to investigate their respective influence on the performance.

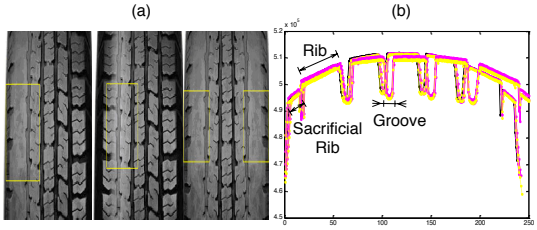

Fig. 1. (a) Three tires with varying degrees of IW patterns. (b) Sample mean sections of different tire types along with the definition of the terminology. A section of a tire is a set of samples across ribs for each scanning point. Each tire type can be characterized by the positions of its grooves and ribs.

We propose a novel mutual information (MI) based approach inspired by the idea of dependent component analysis (DCA) [10, 11] to successfully address the problem of feature split selection. The MI-based method automatically constructs hierarchical clusters and achieves feature splits with maximal between- or within- classifier independence that satisfies diversity or sufficiency. Experimental results on laser tread mapping (LTM) tire data indicate that among the two conditions, sufficiency has a more significant impact on the classification accuracy than diversity. Further results show that co-training with feature splits obtained by the MI-based approach yields significantly higher classification accuracy than supervised learning when only few labeled training data are available.

2. DATA AND PREPROCESSING

2.1. Data set

We analyze LTM data obtained from 22 tires at specific mileages in their service life. Laser mapping is used to measure the progress of the tread wear. To obtain the LTM data, a single point conical laser is used to scan the surface of the tire at 1mm lateral spacing and 4140 points per single wheel revolution (360 degrees) and measures the distance to the tire surface along the normal to the tread surface. Tires with irregular wear (IW)—i.e., non-uniform or uneven wear patterns, resulting in locally depressed regions—should be labeled as a bad tire. Example tire images exhibiting IW are shown in

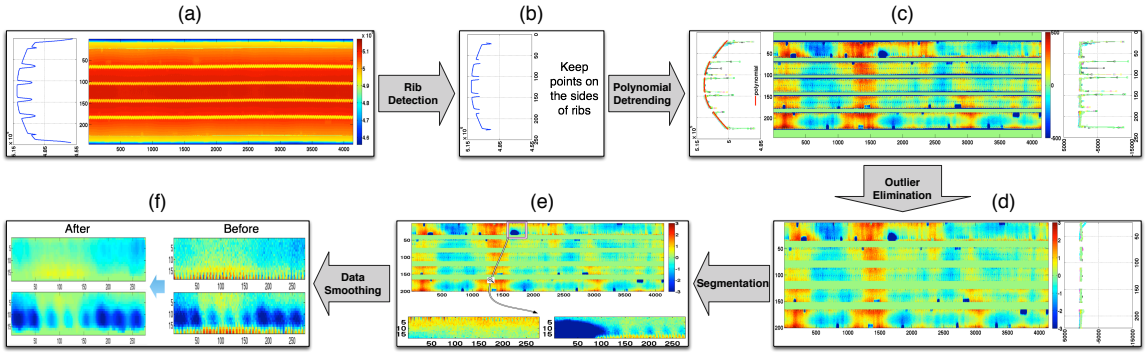

Fig. 2. The preprocessing steps for the LTM data: (a) The mean section of a tire, mean values of 4140 scanning points along the circumference, and the image of the raw data; (b) the mean section after the rib detection step where we keep as many points as possible on the sides of each rib to preserve IW; (c) five second-order polynomials in red and several sections from the tire on left, the image of the tire data after detrending in the middle and the same sections on right; (d) the image and sections of the tire data after the outlier elimination step; (e) images of the whole tire on top and the small samples after segmentation on bottom; and (f) the small samples on right and the corresponding classification samples after the data smoothing step on left.

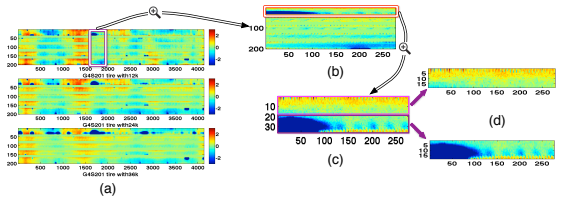

Fig. 3. The figure shows how we cut a tire into small samples: (a) Images of the same tire for specific mileages; (b) one patch from that tire measured in the earliest mileage; (c) one rib from the patch; and (d) the two halves of a rib.

Figure 1 (a). Our goal is to predict the label (IW or non-IW) of tires based on data acquired at the lowest available mileage under the hypothesis that IW can be expressed as a function of the tread depth measurements.

2.2. Data preprocessing

We implement five preprocessing steps. We pay special attention not to introduce any bias to the subsequent analysis stages, thus preserving the information as much as possible. The preprocessing steps are shown in Figure 2, which are rib detection, polynomial detrending, outlier elimination, segmentation, and data smoothing. (1) *Rib detection*: We are interested in studying the IW as it manifests itself on the ribs. Each tire has different positions of grooves and ribs as shown in Figure 1 (b). We retain samples on ribs and discard samples corresponding to grooves. For further analysis, we also combine sacrificial ribs with their neighboring ribs to be able to analyze the former for possible IW. (2) *Polynomial detrending*: For each rib, we select points within two times the standard deviation (std) of samples and compute the regression polynomials for the mean section. We use a second-order polynomial as the trend of a given tire rib to avoid matching points in grooves and IW. We then subtract the trends from each section of the given tire data and obtain flattened tire data. (3) *Outlier elimination*: We detect and remove outliers from the flattened data. After we locate several ribs in the first step, there still exist outliers belonging to grooves. Those typical outliers on the edges of a rib are eliminated by a Grubbs test [12]. (4) *Segmentation*: To obtain more homogeneous samples, we are interested in classifying small units of tires as shown in Figure 3 (d). Thus, we cut each tire into patches according to the general contact length between a tire and the road, then each patch into several ribs and each rib into two halves. (5) *Data smoothing*: We smooth sample images by 2D median filtering. Since IW is expected to be smooth and to have a reasonably large area, we apply 2D median filtering using a window size of 7-by-7 to reduce noise and preserve edges of IW. The classification samples obtained from the 4th step are normalized to have zero mean and unit variance. Thus, IW—usually observed as a depression of the tire—now is associated mainly with negative values.

3. FEATURE EXTRACTION

The aim of the classification task is to detect samples with IW patterns. For each sample, we extract and select 14 relevant features to distinguish between the two groups. Some of our features are straightforward, such as the minimum and mean of negative values for a given sample, and the Euclidean distance from the test sample to the mean of good training samples. Other features are defined as follows.

3.1. KPCA-LDA

Kernel principal component analysis (KPCA) maps the input data into a higher dimension space, called the feature space, by using a non-linear mapping and then applies linear PCA in this feature space [13]. First, for the training matrix $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]^T$, we define an n-by-n matrix $\mathbf{K}$ with entries $k(\mathbf{x}_i, \mathbf{x}_j)$, where $i, j = 1, \ldots, n$ and $k(\mathbf{x}_i, \mathbf{x}_j)$ is the kernel representation. In this paper, we use a Gaussian kernel. Second, we obtain the m largest positive eigenvalues and the corresponding normalized eigenvectors through the eigen-decomposition of the kernel matrix $\mathbf{K}$. The dimension m is selected automatically according to the gap in the eigen spectrum instead of using a fixed number [14]. Third, the KPCA transformed feature is calculated by $\mathbf{y} = \sum_{i=1}^{n} \boldsymbol{\beta}_i k(\mathbf{x}_i, \mathbf{x})$, where $\boldsymbol{\beta}_i = [\beta_{i1}, \beta_{i2}, \ldots, \beta_{im}]^T$ and β_{im} is the i-th entry of the m-th largest eigenvector.

Linear discriminant analysis (LDA) aims to achieve an optimal linear dimensionality reduction [15, 16]. According to Fisher's criterion $J(\mathbf{w}) = \mathbf{w}^T \mathbf{S}_B \mathbf{w} (\mathbf{w}^T \mathbf{S}_W \mathbf{w})^{-1}$, where $\mathbf{S}_B$

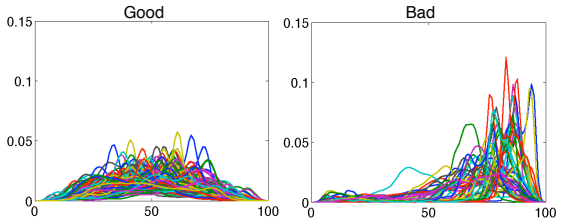

Fig. 4. Estimated probability distributions of good and bad samples by kernel density estimate. The x-axis represents 100 points covering the range of the data and y-axis shows the corresponding density values.

is the between-class covariance matrix and $\mathbf{S}_W$ is the total within-class covariance matrix, we can find a linear combination $\mathbf{w}$ of features that provides a balance between maximum class compactness and class separability by maximizing $J(\mathbf{w})$, where $\mathbf{w}$ is the generalized eigenvector of $(\mathbf{S}_W, \mathbf{S}_B)$ corresponding to the largest generalized eigenvalue. Then the KPCA-LDA feature is obtained by $f = \mathbf{w}^T\mathbf{y}$, where $\mathbf{y}$ is the output of KPCA.

3.2. KL divergence and global statistics

The Kullback-Leibler (KL) divergence is a non-symmetric measure of the information-theoretic distance between two probability distributions [17, 18]. For probability mass functions p and q of a discrete random variable, their KL divergence is defined as $D_{\mathrm{KL}}(p \parallel q) = \sum_i p(i)\log\frac{p(i)}{q(i)}$ which is non-negative and $D_{\mathrm{KL}}(p \parallel q) = 0$ if and only if $p = q$.

From visual inspection, good and bad samples have significant differences between their probability distributions as shown in Figure 4. Hence, we calculate KL divergence between histograms of the test sample and the selected good/bad sample in the training set as a feature. The selected sample is the center of a subset consisting of training samples in the 40th-60th percentile of their skewness values, which are representative for each class.

In addition, we calculate several other statistical measures of the LTM data as features, including std (σ), skewness and kurtosis. Thus, we obtain two features based on KL divergence and three features based on statistics of the LTM data.

3.3. DCT coefficients

A discrete cosine transform (DCT) expresses a sequence of finitely many data points in terms of a sum of cosine functions oscillating at different frequencies [19]. It is defined as

$$y(k) = \varpi(k)\sum_{n=1}^{N} x(n)\cos\left(\frac{\pi(2n-1)(k-1)}{2N}\right), k = 1, 2, \ldots, N$$

where $\varpi(k) = 1/\sqrt{N}$, if $k = 1$, otherwise $\sqrt{2}/\sqrt{N}$ and N is the length of input vector $\mathbf{x}$. DCT, as a simple and efficient frequency domain analysis method, can be used to capture the IW patterns present in high frequencies for the bad sample shown in Figure 5 (b). Thus, we calculate the mean and std of DCT coefficients (1st to 100th that contain enough discriminative power) to define another feature used for classification.

3.4. Magnitude of the 2D gradient

An image gradient is a directional change in the intensity or color in an image [20]. The coordinates of the gradient are given by the formula $\nabla g = [\partial g/\partial x, \partial g/\partial y]^T$, where x represents the horizontal direction and y is the vertical direction.

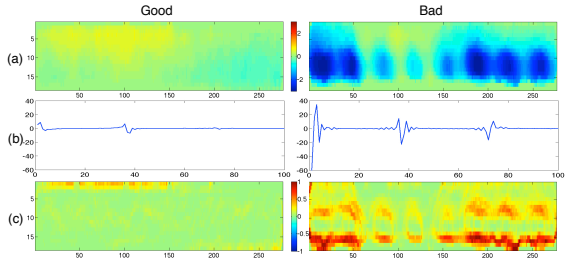

Fig. 5. Comparison of a good and a bad sample using three methods: (a) images of the good/bad sample after preprocessing steps; (b) 1st to 100th DCT coefficients; (c) images of 2D gradient magnitude.

Since IW corresponds to a local depression of the tire surface, we expect 2D gradient magnitude to be higher for IW samples, see Figure 5 (c). Additionally, the std, mean and maximum value of 2D gradient magnitude are calculated as features to be used in classification.

4. CO-TRAINING

4.1. Co-training algorithm

The goal of co-training is to learn a classification mapping from the training set including labeled and unlabeled data. Each classifier is initialized using only the typically few available labeled examples. At every iteration of co-training, each classifier chooses a set of unlabeled examples to add to the training set. The selected set includes those with the highest classification confidence provided by the other classifier. Then, each classifier learns from their augmented labeled set, and the process repeats. The intuition behind the co-training algorithm is that one classifier adds examples to the labeled set that the other classifier will then be able to use successfully for learning [21].

We implement several classifiers to perform co-training, including Naive Bayes (NB), probabilistic multilayer perceptron (MLP) [22, 23] and a support vector machine (SVM) classifier. If we apply NB and MLP in co-training, the confident examples are computed based on the posterior probabilities which are the classifier outputs. The probability of a test example belonging to one class is obtained as the product of the outputs of two classifiers. If we apply SVM, the confident examples are selected based on the distance to the decision boundary and a test example belongs to the group predicted by the classifier producing higher confidence.

4.2. Feature splits based on MI

Co-training requires constructing and splitting two sets of features from original data to perform successful classification. However, it is not easy to construct feature sets that satisfy both diversity and sufficiency. Hence, we propose an MI-based approach inspired by the idea of DCA for the task.

DCA model relaxes the independence assumption by decomposing the data into independent subsets where within each subset, the components are dependent. A practical and effective way to obtain DCA decomposition is by first performing independent component analysis (ICA) [24, 25] and then grouping the independent components into clusters by using MI as the metric between components [26, 27]. We propose to use the grouping part of DCA to split features by max-

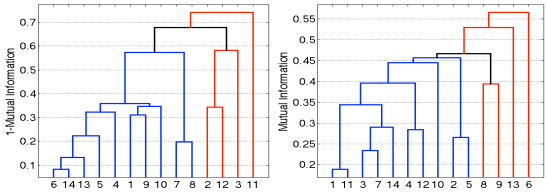

Fig. 6. Two feature splits obtained by the MI-based approach. The x-axis represents indices of features with the order of description in Section 3. Features shown as the same color are grouped in one classifier in co-training.

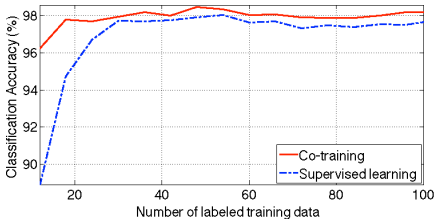

Fig. 7. Comparison between co-training and supervised learning. For both, we use NB classifier.

imization of the MI between two sub-feature sets denoted as $\mathcal{I}(\{f_i, i \in S\}, \{f_j, j \in \{1, 2, \ldots, F\} - S\})$, where F is the total number of features, involving several stages:

1. Select and extract feature $f_i[k] = f_i(\mathbf{x}_k)$ from each of labeled LTM samples $\mathbf{x}_k$, $1 \le k \le L$;
2. Calculate $\mathcal{I}(f_i, f_j), i, j = 1, \ldots, F$ (normalized through $\sqrt{1 - e^{-2\mathcal{I}(f_i,f_j)}} \in [0, 1)$);
3. Construct $F \times F$ MI matrices $\mathbf{M}^{[g]}$ and $\mathbf{M}^{[b]}$ using L_1 good and L_2 bad labeled samples, respectively;
4. Calculate the MI matrix given class information, defined as $\mathbf{M} = (\mathbf{M}^{[g]} \times L_1 + \mathbf{M}^{[b]} \times L_2)/L$;
5. Generate dendrograms using hierarchical clustering based on the distance measure $\mathbf{1} - \mathbf{M}$ and $\mathbf{M}$, respectively, where $\mathbf{1}$ is the $F \times F$ matrix with all entries equal to 1.

After applying this MI-based approach, we perform classification using the co-training algorithm with the feature splits, we thus obtain.

5. EXPERIMENTAL RESULTS

Labels for LTM samples are assigned by an expert. Then we select 270 samples for which we have confidence in their labels including 47 bad and 223 good half-ribs from a total of 1320 samples. For each experiment, we take the average of 100 runs as the final classification accuracy and report the std. In the co-training procedure, we select labeled training and test data randomly from 270 samples and allow others to be unlabeled data for each run. Also, we keep the same proportion (1/5) of bad/good samples in the training set of each run to make sure co-training yields consistent results.

5.1. Evaluation of feature splits

We obtain two feature splits that satisfy two conditions of co-training from the MI-based method. To investigate the significance of these two splits for co-training, we randomly select 11 features from 14 features to construct feature splits and perform co-training with NB classifier using 36 labeled training data. For each split, we require that each classifier has at least three features. Then we analyze the experimental

Table 1. Results of two splits in co-training(%)

	NB	SVM			MLP
		Linear	RBF ($\sigma = 2.5$)	Poly (2)	
Split 1	97.7 ± 2.1	97.7 ± 2.0	98.5 ± 1.9	96.2 ± 2.9	98.2 ± 1.7
Split 2	98.4 ± 2.0	97.8 ± 2.4	98.1 ± 1.8	97.1 ± 2.5	98.4 ± 1.5

Table 2. Evaluation of two feature splits

	Between-class	Within-class	Split 1 − Split 2
t-values	2.6	−42.8	−1.1
p-values	0.06	0.18×10^{-5}	0.28

results of 364 feature combinations, each combination containing two splits. The results include MI between-classifier, the average MI within-classifier and classification accuracy. In the analysis, we perform (1) a paired t-test between classification accuracy of two splits (Split 1 represents the split based on $\mathbf{1} - \mathbf{M}$; Split 2 represents the split based on $\mathbf{M}$), and (2) a permutation test on multiple regression coefficients. The multiple regression is defined as $\mathbf{m} = a\mathbf{n}_1 + b\mathbf{n}_2$, where $\mathbf{m}$ represents a random subset of obtained classification accuracy, $\mathbf{n}_1$ and $\mathbf{n}_2$ denote the corresponding subsets of between- and within- classifier MI, and a and b are the coefficients. We also perform co-training on two splits of 14 features using several classifiers, including NB, MLP and SVM with different kernels, and give the comparison in Figure 6 and Table 1.

The results in Table 2 show that within-class MI has significantly negative correlation with the classification accuracy. In other words, the more powerful the retained features within a classifier, the higher is the obtained classification accuracy. The between-class independence is also important since the t-value of between-class MI is not significant and it is very small compared to the result of within-class. Even though the classification rates yielded by Split 2 are not significantly higher than the results of Split 1, our results indicate that sufficiency is more important than diversity. Thus, in the following analysis, we apply Split 2 in co-training to evaluate the classification performance.

5.2. Performance of co-training

One of the advantages of co-training is that even a few labeled training data may lead to high classification accuracy. We thus evaluate the performance of co-training with increasing number of initially available labeled training data and compare the results with supervised learning as shown in Figure 7. The results indicate that co-training has great power using a few labeled training samples compared to supervised learning with the NB classifier.

6. CONCLUSION

In this work, we propose a novel MI-based approach to split features for co-training. These features are extracted from LTM data using several feature extraction methods. We introduce an efficient method to perform co-training when features are not naturally separated into two subsets. In earlier studies, few methods of feature splits have been proposed for co-training [7–9]. In these methods, best splits are evaluated or selected among a huge amount of random feature splits according to their criteria. Additionally, our experimental result indicates that sufficiency has a more significant contribution to classification accuracy compared to diversity, which clarifies the dependence of co-training performance on two conditions diversity and sufficiency.

7. REFERENCES

[1] X. Zhu, "Semi-supervised learning literature survey," Computer Sciences, University of Wisconsin-Madison, Tech. Rep. 1530, 2005.

[2] G. Tur, "Co-adaptation: Adaptive co-training for semi-supervised learning," in *Acoustics, Speech and Signal Processing (ICASSP), 2009 IEEE International Conference on*, 2009, pp. 3721–3724.

[3] X. Cui, J. Huang, and J.-T. Chien, "Multi-view and multi-objective semi-supervised learning for large vocabulary continuous speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, 2011, pp. 4668–4671.

[4] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," in *Proceedings of the 11th annual conference on Computational Learning Theory*, New York, NY, 1998, pp. 92–100.

[5] S. Abney, "Bootstrapping," in *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 2002, pp. 360–367.

[6] M. F. Balcan, A. Blum, and K. Yang, "Co-training and expansion: Towards bridging theory and practice," in *Neural Information Processing Systems (NIPS)*, 2004.

[7] M. Terabe and K. Hashimoto, "Evaluation criteria of feature splits for co-training," in *Proceedings of the International Multi-Conference of Engineers and Computer Scientists*, 2008, pp. 540–545.

[8] A. Salaheldin and N. El-Gayar, "New feature splitting criteria for co-training using genetic algorithm optimization," in *Multiple Classifier Systems*, 2010, vol. 5997, pp. 22–32.

[9] ——, "Complementary feature splits for co-training," in *Information Science, Signal Processing and their Applications (ISSPA)*, 2012, pp. 1303–1308.

[10] A. Hyvärinen and P. Hoyer, "Emergence of phase- and shift-invariant features by decomposition of natural images into independent feature subspaces," *Neural Comput.*, vol. 12, no. 7, pp. 1705–1720, 2000.

[11] F. R. Bach and M. I. Jordan, "Beyond independent components: Trees and clusters," *J. Mach. Learn. Res.*, vol. 4, pp. 1205–1233, 2003.

[12] F. E. Grubbs, "Procedures for detecting outlying observations in samples," *Technometrics*, vol. 11, no. 1, pp. 1–21, 1969.

[13] B. Schölkopf, A. Smola, and K.-R. Müller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Computation*, vol. 10, no. 5, pp. 1299–1319, 1998.

[14] W. Du, V. D. Calhoun, H. Li, S. Ma, T. Eichele, K. A. Kiehl, G. D. Pearlson, and T. Adalı, "High classification accuracy for schizophrenia with rest and task fMRI data," *Frontiers in Human Neuroscience*, vol. 6, no. 145, 2012.

[15] R. Fisher, "The use of multiple measures in taxonomic problems," *Ann Eugenics*, vol. 7, pp. 179–188, 1936.

[16] C. M. Bishop, *Neural Neworks for Pattern Recognition*. New York, NY: Oxford Univerity Press, 1995.

[17] S. Kullback and R. A. Leibler, "On information and sufficiency," *Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[18] S. Kullback, K. P. Burnham, N. F. Laubscher, G. E. Dallal, L. Wilkinson, D. F. Morrison, M. W. Loyer, B. Eisenberg, S. Ghosh, I. T. Jolliffe, and J. S. Simonoff, "Letters to the editor: The kullback-leibler distance," *The American Statistician*, vol. 41, no. 4, pp. 340–341, 1987.

[19] N. Ahmed, T. Natarajan, and K. Rao, "Discrete cosine transform," *Computers, IEEE Transactions on*, vol. C-23, no. 1, pp. 90–93, 1974.

[20] C. R. Gonzalez and R. E. Woods, *Digital Image Processing*, 3rd ed. Pearson Prentice Hall, 2007.

[21] K. Nigam and R. Ghani, "Analyzing the effectiveness and applicability of co-training," in *Proceedings of the 9th international conference on Information and Knowledge Management*, New York, NY, 2000, pp. 86–93.

[22] D. MacKay, "The evidence framework applied to classification networks," *Neural Computation*, vol. 4, no. 5, pp. 720–736, 1992.

[23] ——, "A practical Bayesian framework for backpropagation networks," *Neural Computation*, vol. 4, pp. 448–472, 1992.

[24] P. Comon, "Independent component analysis, a new concept?" *Signal Processing*, vol. 36, no. 3, pp. 287–314, 1994.

[25] A. Hyvärinen and E. Oja, "Independent component analysis: Algorithms and applications," *Neural Networks*, vol. 13, pp. 411–430, 2000.

[26] J. F. Cardoso, "Multidimensional independent component analysis," in *Acoustics, Speech and Signal Processing (ICASSP), 1998 IEEE International Conference on*, vol. 4, 1998, pp. 1941–1944.

[27] S. Ma, X.-L. Li, N. Correa, T. Adalı, and V. Calhoun, "Independent subspace analysis with prior information for fMRI data," in *Acoustics, Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, 2010, pp. 1922–1925.