TMLAI TRANSACTIONS ON MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE
SOCIETY FOR SCIENCE AND EDUCATION UNITED KINGDOM
VOLUME 2, ISSUE 1
ISSN: 2054 - 7390

Hierarchical sparse representation for object recognition

Toru Nakashika, Takeshi Okumura, Tetsuya Takiguchi, Yasuo Ariki
Graduate School of System Informatics, Kobe University, Japan;
nakashika@me.cs.scitec.kobe-u.ac.jp, {takigu,ariki}@kobe-u.ac.jp

ABSTRACT

Recently, generic object recognition that achieves human-like vision has being looked to for use in robot vision, automatic categorization of images, and image retrieval. In object recognition, semi-supervised learning, which incorporates a large amount of unsupervised training data (unlabeled data) along with a small amount of supervised data (labeled data), is regarded as an effective tool to reduce the burden of manual annotation. However, some unlabeled data in semi-supervised models contain outliers that negatively affect the parameter estimation during the training stage. Such outliers often cause an over-fitting problem especially when a small amount of training data is used. Furthermore, another problem that occurs when using the conventional methods is that when labeling an image based on super-pixel representation, the lack of discrimination of the image features and the scale variance of the objects decreases the recognition accuracy because the feature extraction is based on the mono-scale segmentation. In this paper, we propose an object recognition method for solving both problems. For the former problem, our method prevents the over-fitting associated with the semi-supervised based approach by using sparse representation to suppress existing outliers in the data. For the latter problem, we employ Tree Conditional Random Field to construct the hierarchical structure of an image. Experiment results using two datasets confirm the effectiveness of our method.

Keywords: Object recognition, Automatic image annotation, Sparse representation, Semi-supervised learning, Hierarchical representation, Tree Conditional Random Field.

1. INTRODUCTION

Generic object recognition (automatic image annotation), in which the system automatically assigns labels to an image, is one of the most significant tasks in computer vision. Most of the conventional methods are based on a supervised labeling approach in order to achieve an exact classification. However, it has been pointed out that with this approach the training cost is extremely high because an enormous amount of training data must be labeled manually. To reduce the amount of such a troublesome work, a semi-supervised approach has recently

DOI: 10.14738/tmlai.21.95
Publication Date:10th February 2014
URL: http://dx.doi.org/10.14738/tmlai.21.95

attracted considerable attention in machine learning [1][2][3][4][5]. The semi-supervised approach inputs a large amount of non-labeled data (unsupervised data) for the training as well as not so much labeled data. Hence, it helps to improve the training accuracy without using a lot of labeled data.

Descriptions of popular methods using semi-supervised learning in text classification can be found in [2], which introduces TSVM (transductive support vector machine) as a classification model, and in [3], which introduces SemiNB (semi-supervised naive Bayes classifier) as a generative model. TSVM extends the well-known SVM so that it can be trained not only with a few labeled data but also with a large volume of unlabeled data. During the training, labeled data first determine the margin, which classifies unlabeled data. The former SemiNB is a semi-supervised version of Naive Bayes (NB). Both methods, especially in SemiNB, are adversely affected by the influence of outliers in large amounts of unsupervised data, because they both take whole the unlabeled data as well as labeled data in the training process. The TSVM limits the influence of the outliers only to data around the margin. Therefore, the TSVM is not influenced as much by outliers as SemiNB is, though it is inevitable that the outliers negatively affect the margin estimation. Furthermore, the TSVM is a computationally expensive algorithm. Given a large number of training data, it needs to take an approximate approach that causes weak estimation of the margin.

In consideration of the drawbacks in a semi-supervised approach, we propose an automatic image annotation method where an effective semi-supervised tool, semi-supervised canonical correlation analysis (semi-CCA) [4], and sparse representation [6] collaboratively suppress the influence of outliers. First, subspaces that maximize the correlation between image features and label features are generated by semi-CCA, using a small amount of labeled data and much unlabeled data. Semi-CCA extends canonical correlation analysis (CCA), so as to avoid over-fitting when it has a few (labeled) training data. Given a large amount of unlabeled data as well as the labeled data, it grabs a global distribution. Since the trained distribution is affected by outliers somewhat, we adopt Regularized Orthogonal Matching Pursuit (ROMP) [7], one of the handy sparsing algorithms. Using sparse representation, it is possible to achieve the automatic annotation that utilizes an abundance of unlabeled data for the semi-supervised learning that is robust to the influence of outliers.

Our approach is based on super-pixel representation [8,9], where low-level features, such as the color feature and the texture feature, are extracted from the local region (super-pixel), and the class of each region is recognized based on such features. However, there is a problem in the super-pixel-based methods in that they are not robust to the scale variance due to their inability to discriminate the features extracted from the local regions. Therefore, we further employ hierarchical representation in this paper. This hierarchical structured model, which is called Tree Conditional Random Field (TCRF) [10], is robust to the scale variance since it accounts for multi-scale hierarchization.

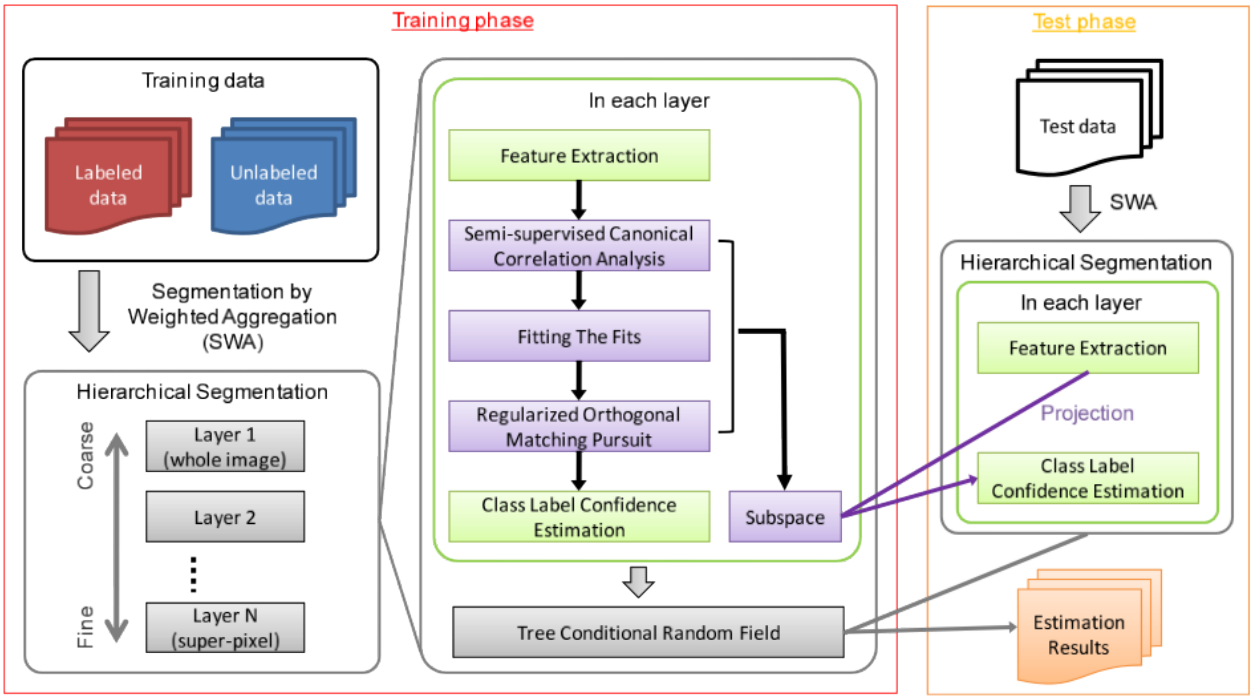

Figure 1: System flowchart of proposed method.

Figure 1 shows the flow of the proposed method. In the training stage, we first divide all training images including labeled and unlabeled data into hierarchical subregions using SWA (Segmentation by Weighted Aggregation) [11]. For each region in each layer, we extract features and apply sparse-representation in semi-supervised learning to estimate class label confidence. Then, we train TCRF using the confidence and the label data in all layers to estimate the probability of co-occurrences within classes in a hierarchical structure. In the test stage as well, we first apply SWA to obtain hierarchical subregions, project the regions into sub-space in each layer, and finally estimate the class label for each region (pixel) using TCRF.

The rest of the paper is organized as follows. In Section 2, we present the way of applying sparse representation in semi-supervised learning, and we explain the final annotation using TCRF in Section 3. The performance of the proposed method is evaluated in Section 4, and we conclude in Section 5.

2. SUBSPACE GENERATION AND SPARSE REPRESENTATION

Due to the high cost of preparing correct labels as training data in automatic image annotation, it is desirable to employ a semi-supervised approach which uses unlabeled data instead of some of labeled data. However, there exist outliers in the unlabeled data. In this section, we discuss a method that generates subspaces using the semi-supervised approach called Semi-supervised Canonical Correlation Analysis (semi-CCA) [4], and suppresses such outliers using Regularized Orthogonal Matching Pursuit (ROMP) [7] as shown in Figure 2.

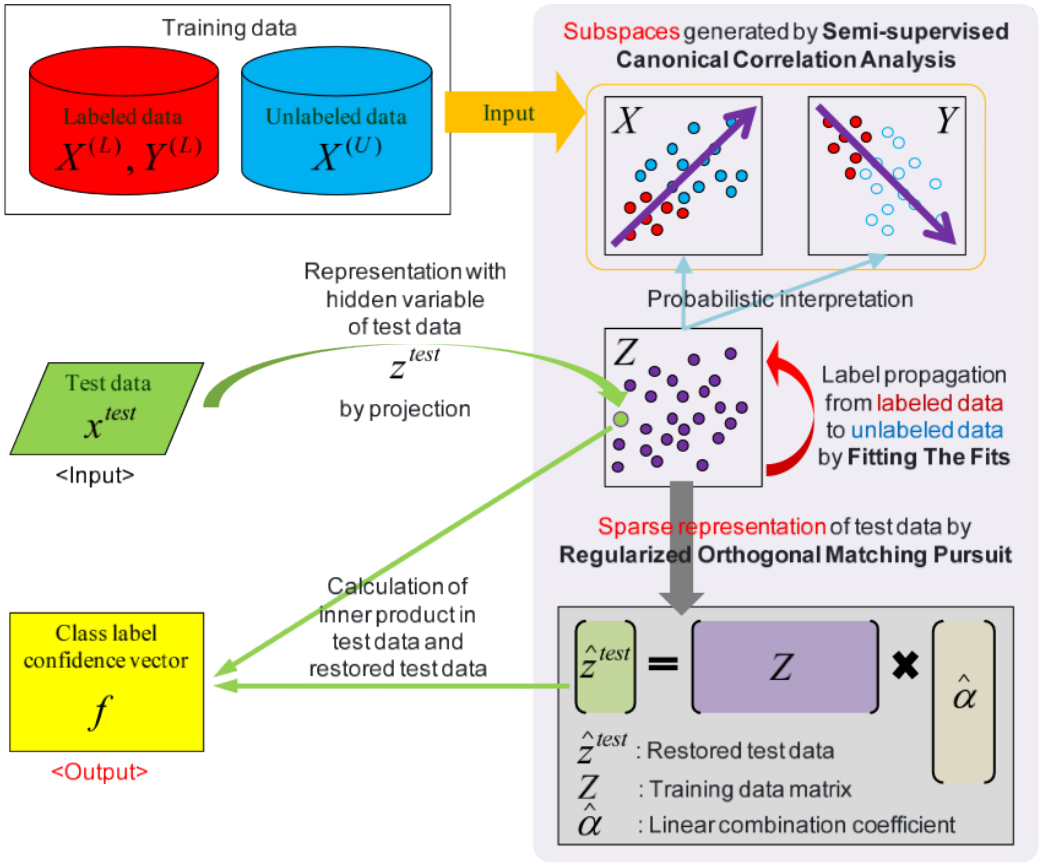

Figure 2: Flow of outlier suppression in semi-supervised learning using sparse representation.

2.1 Semi-CCA

The semi-CCA is an extended version of Canonical Correlation Analysis (CCA) so that it substitutes unlabeled data for some labeled data. Both methods find the subspace that maximizes a correlation between two different types of features. In this paper, the relationship (correlation) between an image and the accompanying labels are obtained.

Let $\{X^{(L)}, Y^{(L)}, X^{(U)}\}$ be a training data set, where $X^{(L)} = \{x_n\}_{n=1}^{N}$ and $Y^{(L)} = \{y_n\}_{n=1}^{N}$ are N labeled data, and $X^{(U)} = \{x_m\}_{m=1}^{M}$ is M unlabeled data. x and y indicate image feature and label feature, respectively (See 2.2). The aim of Semi-CCA or CCA is to find the optimum subspace that maximizes a correlation between projected x and y:

$$r(w_x, w_y) = \frac{w_x^T S_{xy}^{(L)} w_y}{\sqrt{w_x^T S_{xx}^{(L)} w_x}\sqrt{w_y^T S_{yy}^{(L)} w_y}} \tag{1}$$

where w_x and w_y are projection vectors to the subspace from the original feature space x and y, respectively. $S_{ii}^{(L)}$ indicates each variance-covariance matrix within the labeled data. For example, $S_{xy}^{(L)} = N^{-1}\sum_{n=1}^{N} x_n y_n^T$.

If unlabeled data $X^{(U)}$ is not given as the training (in other words, in the case of normal CCA), all that can be done is just to formulate the maximum problem in which the optimum w_x and w_y are found to maximize Eq. (1) using a Lagrange multiplier. In that case, the formulation boils down to an eigen-value problem.

When the amount of labeled data is not adequate, the obtained subspace is inefficiently overfitted to the training data. Hence, unlabeled data are added for correcting a global structure of data distribution in the subspace. In order to do that, PCA is employed using the concept of semi-CCA [4]. In a similar way to CCA, a projection matrix of the PCA can be calculated by solving an eigenvalue problem, in which a variance-covariance matrix of the data is maximized under a normalized orthogonal constraint.

As mentioned above, semi-CCA can be expressed as the combination of two factors: CCA with labeled data and PCA with all data including unlabeled data. Therefore, the semi-CCA formulation is also obtained by combination of the two eigenvalue problems, as in Eq. (2). A projection matrix can ultimately be obtained from the upper eigenvalues using semi-CCA.

$$B \begin{bmatrix} w_x \\ w_y \end{bmatrix} = \lambda C \begin{bmatrix} w_x \\ w_y \end{bmatrix} \tag{2}$$

where,

$$B = \beta \begin{bmatrix} 0 & S_{xy}^{(L)} \\ S_{yx}^{(L)} & 0 \end{bmatrix} + (1-\beta) \begin{bmatrix} S_{xx} & 0 \\ 0 & S_{yy}^{(L)} \end{bmatrix} \tag{3}$$

$$C = \beta \begin{bmatrix} S_{xx}^{(L)} & 0 \\ 0 & S_{yy}^{(L)} \end{bmatrix} + (1-\beta) \begin{bmatrix} I_{D_x} & 0 \\ 0 & I_{D_y} \end{bmatrix} \tag{4}$$

and $S_{xx} = (N+M)^{-1} \sum_{n=1}^{N+M} x x^T$ is a variance-covariance matrix of all image feature vectors including unlabeled images. I_{D_x} and I_{D_y} are identity matrices with the size $D_x \times D_x$ and $D_y \times D_y$, respectively. Note that the first term and the second term in Eq. (3) and (4) indicate the terms related to eigenvalue problems of CCA and PCA, respectively. β is a trade-off parameter which determines the effects of CCA and PCA.

The image feature and the label feature are connected via latent variables z in the subspace. These variables can be calculated by applying the conditional Gaussian model (For more details, see [4]). After this, we can rewrite the training and test data with the latent variables z for the sake of consideration in the subspace.

2.2 Image Feature and Label Feature

Each image is first divided into small subregions using Normalized Cuts [12]. In each subregion, the following features are extracted:

- Color : Statistics of RGB, HSV, Lab, and YCbCr
- Gabor : Gabor filter and Laplacian-of-Gaussian
- Position : Center position of a region
- Geometric : Area of a region

An image feature vector x is defined as a super-vector, where all these features are included. A label feature vector y is a binary vector, where each label is assigned in the subregion or not.

2.3 Annotation Using Sparse Representation

Recently, classification methods based on sparse representation, in which test data are represented as a linear combination of sparse bases, have been drawing attention in image processing [13][14]. It was reported in these papers that classification results showed favorable robustness of sparse representation against outliers. In this paper, we set out to suppress the effect of outliers that unintentionally appear when there is a large amount of unlabeled data, by employing sparse representation.

If a sufficient amount of training data is prepared, an input image $\mathbf{z}^{\text{test}}$ in the subspace can be represented as a linear combination of the training data. Our aim is to find sparse coefficients associated with each training data. Those entries are mostly zero, except for a few elements. This can be formulated as a minimizing problem with respect to a coefficient vector α in Eq. (5).

$$\min_{\boldsymbol{\alpha}} ||\boldsymbol{\alpha}||_{\epsilon} \quad s.t. \quad \mathbf{z}^{\text{test}} = \sum_{n=1}^{N+M} \alpha_n \mathbf{z}_n = \mathbf{Z}\boldsymbol{\alpha} \tag{5}$$

where $\mathbf{Z} \in \mathbb{R}^{D_z \times (N+M)}$ is a training data matrix (D_z is a dimension of subspace feature). $||\boldsymbol{\alpha}||_{\epsilon}$ indicates l_{ϵ} norm, which is the number of almost-zero elements in α, given by $||\boldsymbol{\alpha}||_{\epsilon} = (N+M)^{-1} \# \{n | \alpha_n \leq \epsilon\}$ with an experimentally-determined small value ϵ. However, it is computationally difficult to find the optimum vector in Eq. (5) because $||\boldsymbol{\alpha}||_{\epsilon}$ is indifferentiable. In this paper, we consequently adopt one of the popular greedy algorithms, Regularized Orthogonal Matching Pursuit (ROMP) [7] to solve this optimum problem.

At the end, the test data can be restored by multiplying a training data and the obtained vector $\hat{\boldsymbol{\alpha}}$ as $\hat{\mathbf{z}}^{test} = \mathbf{Z}\hat{\boldsymbol{\alpha}}$. By taking an inner production between the test data and the restored data, a restoration ratio f_c of the label class c can be calculated as

$$f_c = \frac{\mathbf{z}^{test^T}\hat{\mathbf{z}}_c^{test}}{||\mathbf{z}^{test}||_2||\hat{\mathbf{z}}^{test}||_2} = \frac{\mathbf{z}^{test^T}\mathbf{Z}_c\hat{\boldsymbol{\alpha}}_c}{||\mathbf{z}^{test}||_2||\hat{\mathbf{z}}^{test}||_2} \quad (6)$$

where Zc is a training data matrix that only contains the data given the label c, and $\ddot{\boldsymbol{\alpha}}_c$ is a coefficient vector associated with the training data in Zc. The restoration ratio f_c implies a confidence of the class c. Therefore, multi-label classification can be realized by calculating all the confidences $f_c(c = 1, \cdots, C)$ (C is the number of label classes). However, we use the best class label f_c in all classes by taking the maximum for the following procedure.

3. HIERARCHICAL REPRESENTATION

From Eq. (6), we obtain the class label for one sub-pixel in a certain layer. After obtaining all the class labels for each sub-pixel in each layer, we finally estimate the class label for each region (pixel), considering the hierarchical and spatial co-occurrence in adjacent subregions using Tree Conditional Random Field (TCRF) [10].

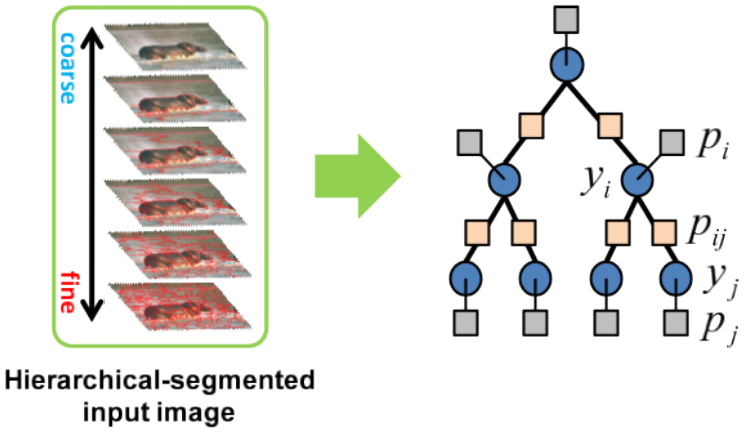

Figure 3: Graphical representation of the image using Tree Conditional Random Field.

Conditional Random Field (CRF) [15] was originally introduced in linguistic processing to represent a graphical and discriminative model. This model is used for estimating the class of the structured observation. When we apply the CRF to the hierarchical-segmented image as shown in Figure 3 *left*, each segment is represented as a node, and all segments that have the relation between layers are connected by an edge (Figure 3 *right*). Such a graphical model is called a TCRF.

Let $i \in T$ denote each segment in a hierarchical-segmented image, $\pi(i)$ be the set of the child nodes for the parent node i, $\mathbf{F} = \{f_i\}_{i \in T}$ describe the class reliability in each segment obtained from the previous section, and $\mathbf{Y} = \{y_i\}_{i \in T}$ indicate the class labels estimated in each node. Then, the model formula of the TCRF is written as the following conditional distribution $P(\mathbf{Y}|\mathbf{F};\theta)$.

$$P(\mathbf{Y}|\mathbf{F};\theta) = \frac{1}{Z}\exp\left\{\sum_{i\in T} p_i(y_i|\mathbf{f}_i;\alpha) + \sum_{i\in T}\sum_{j\in\pi(i)} p_{ij}(y_i,y_j;\beta)\right\} \quad (7)$$

where Z is called partition for regularization. $\theta = \{\alpha, \beta\}$ represents the model parameters of TCRF estimated based on the following Maximum A Posteriori (MAP) by using all the training images with ground truth.

$$\theta^* = \arg\max_{\theta}\left\{\sum_{t=1}^{T}\log P(\mathbf{y}^t|\mathbf{X}^t;\theta) - \frac{R}{2}||\theta||^2\right\} \quad (8)$$

where T is the number of the training images, and R is the parameter for preventing over-fitting. θ^* is computed analytically by an L-BFGS method [9]. The first term in Eq. (7) $p_i(y_i|\mathbf{x}_i;\alpha)$ is the class reliability distribution in each node, which is defined as follows.

$$p_i(y_i|\mathbf{f}_i;\alpha) = \sum_{k=1}^{C}\alpha_{ky_i}f_{ik} \qquad \left(\sum_{k=1}^{C} f_{ik} = 1\right) \quad (9)$$

The second term in Eq. (7) $p_{ij}(y_i,y_j;\beta)$ is the class co-occurrence between the adjacent nodes defined as:

$$p_{ij}(y_i,y_j;\beta) = \beta_{y_iy_j} \qquad (y_i \neq y_j) \quad (10)$$

For the final class estimation, we need to find the class of each node that maximizes the conditional distribution shown in Eq. (7), given the test image $I^{test} = \mathbf{F}^{test}$. For this purpose, we use Maximizer of Posterior Marginal (MPM) estimation.

$$y_i^* = \arg\max_{y_i\in\mathcal{C}} P(y_i|\mathbf{F}^{test};\theta^*) = \arg\max_{y_i\in\mathcal{C}}\sum_{\mathbf{Y}\setminus y_i} P(\mathbf{Y}|\mathbf{F}^{test};\theta^*), \quad i\in T \quad (9)$$

where y_i^* is the class maximizing the posterior marginal distribution, and $\mathcal{C}$ is a collection of the labels. Since the graph structure is one of the tree structures, the global optimal estimation can be done by Belief Propagation [16].

By decreasing the segmentation error, we regard the estimation result in the bottom layer as the final estimation result. Since this estimation considers all estimation results in all layers of the hierarchy and is a global optimum, the proposed method is robust to the scale variance of objects.

4. EXPERIMENTAL EVALUATION

In Section 4.1, we first see the effectiveness of the labeling method only using sparse representation in semi-supervised learning described in Section 2. Then, we evaluate our method that combines the sparse representation and hierarchical representation in Section 4.2.

4.1 Object Recognition Using Sparse Representation in Semi-supervised Learning

4.1.1 Experimental Conditions

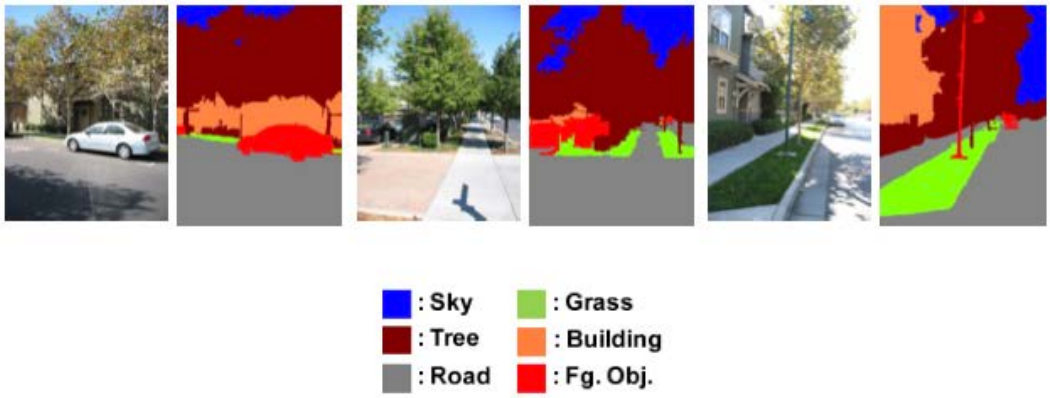

Figure 4: Examples from STAIR image dataset

For this image annotation experiments, we used a STAIR image data set [17], which contains 534 images along with pixel-wise 5 labels (“Sky”, “Tree”, “Road”, “Grass” and “Building”) as their examples are shown in Figure 4. In our experiments, the training and test are conducted using features in each subregion, which is divided by Normalized Cuts. Labeling accuracies for each class and their average are calculated by accumulating the subregion results. The accuracy was evaluated with 3-fold cross validation. Images for the training and the test were randomly selected (400 images for the training and 134 images for the test) three times for each validation.

We conducted two experiments in this section. In the first experiment, we compared with conventional semi-supervised methods: “SemiNB” and “TSVM”. Secondly, we examined these methods in a supervised manner; a supervised variation of our method (without hierarchical representation) was compared with “NB” and “SVM,” just to see the effectiveness of semi-supervised approach. Here we employed CCA instead of semi-CCA, given a full set of labeled data.

4.1.2 Results and Discussion

The results of semi-supervised and supervised approaches are shown in Table 1 and Table 2, respectively. As shown in these tables, the recognition accuracy of our method is higher than not only the other semi-supervised approaches but also the supervised approaches, such as SVM. The other methods, SVM and NB, suffer decreased accuracy in the semi-supervised case. This is, in general, because conventional approaches make extensive use of unsupervised data, and their classifiers were consequently affected by unsupervised factors, especially outliers. On the other hand, our approach increases the accuracy in the semi-supervised case. This is

considered to be due to the benefit of semi-supervised learning, which helps the classifier to catch the global structure of data distribution in the case where there is a very small amount of labeled data for the training (due to the effective suppression of outliers using sparse representation).

Table 1: Recognition accuracies of non-hierarchical methods (semi-supervised approaches) [%]

Label	Sky	Tree	Road	Grass	Building	Average
SemiNB	25.3	30.5	70.6	47.5	31.3	41.0
TSVM	82.8	62.0	83.3	74.4	68.6	74.2
Our method	87.2	66.0	84.4	80.1	75.9	78.7

Table 2: Recognition accuracies for comparison of supervised methods [%]

Label	Sky	Tree	Road	Grass	Building	Average
NB	43.2	26.7	73.8	54.3	33.1	46.2
SVM	87.8	57.1	85.5	76.9	65.2	74.5
ROMP	54.5	39.2	51.5	44.5	39.9	45.9
Our method (sp)	85.0	63.7	89.6	75.4	73.2	77.4

4.2 Object Recognition Using Hierarchical Sparse Representation

4.2.1 Experimental Conditions

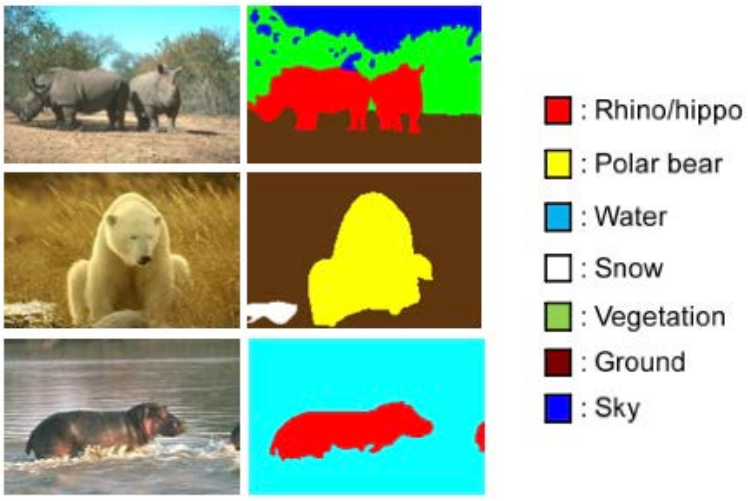

Figure 5: Examples from STAIR image dataset

In this section, we show the object recognition results of our proposed method, where we first estimate class labels for each of the subregions in each layer using the sparse representation method described in Section 2, and then adjust the labels using hierarchical representation as described in Section 3. For the experiments, we used a Corel image dataset

[18], in addition to a STAIR image dataset, which consists of 100 images with 7 labels ("Rhino/Hippo", "Polar bear", "Water", "Snow", "Vegetation", "Ground", and "Sky") as shown in Figure 5. For this dataset, all images have the same size of 180x120. We divided all images into hierarchical subregions using SWA (Segmentation by Weighted Aggregation) [11], and extracted local features from the lowest layer where we set the number of subregions to 200 per image. Each method was evaluated using leave-one-out cross validation. The experimental conditions related to the STAIR image dataset are the same as in the previous section.

4.2.2 The Number of Layers

First, we investigated our hierarchical method to see how the accuracy changes as the number of layers increases. We changed the number of layers from 1 to 6, and set the number of subregions in each case as shown in Table 3. For example, we divide an image into 200 subregions, 100 subregions, and 50 subregions for the first layer, the second layer, and the third layer, respectively, when we use the four-layer structure.

The results are shown in Figure 6. In most of the cases, the recognition accuracy increases as the number of layers increases. We obtained the best performance when there were 5 layers. Therefore, we used the five-layer structure in the remaining experiments.

Table 3: The number of subregions in each layer

	Layer index					
Number of layers	1	2	3	4	5	6
1	200	-	-	-	-	-
2	200	1	-	-	-	-
3	200	100	1	-	-	-
4	200	100	50	1	-	-
5	200	100	50	25	1	-
6	200	100	50	25	12	1

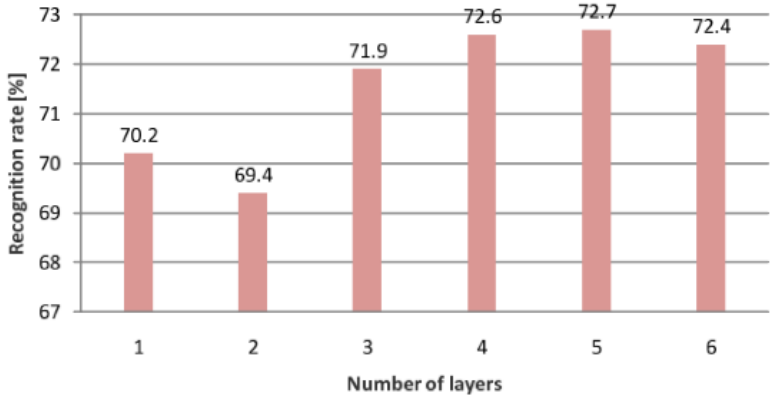

Figure 6: Change in accuracy due to increasing the number of layers.

4.2.3 Experimental Results and Discussion

The experimental results using a Corel image dataset are shown in Table 4. The results of the comparison method "CRF" were obtained from a non-hierarchical structure [19,20]; i.e. the same case as when the number of layers was 1 in the previous section. "LR" has the same conditions as "CRF", except for using logistic regression for the clustering method instead of CRF. As shown in Table 4, our proposed method improved the accuracy by 2.5 points due to the hierarchical structure.

Table 5 shows the results of our proposed method using the STAIR image dataset. The difference between "Our method(a)" and "Our method(b)" is that "Our method(a)" does not have a hierarchical structure but has sparse representation (in the same case as shown in Section 4.1.2). On the other hand, "Our method(b)" has a hierarchical structure. We obtained better performance with "Our method(b)".

Table 4: Recognition accuracies using Corel image dataset [%]

Label	Rhino	P. bear	Water	Snow	Vege.	Ground	Sky	Average
LR	73.5	65.1	70.3	68.2	75.3	71.0	56.6	68.6
CRF	71.8	71.0	82.6	70.6	78.9	74.7	41.7	70.2
Our method	76.2	74.1	80.4	73.0	80.9	74.1	50.4	72.7

Table 5: Recognition accuracies using STAIR image dataset [%]

Label	Sky	Tree	Road	Grass	Building	Average
Our method(a)	87.2	66.0	84.4	80.1	75.9	78.7
Our method(b)	92.1	63.2	85.1	84.9	74.9	80.0

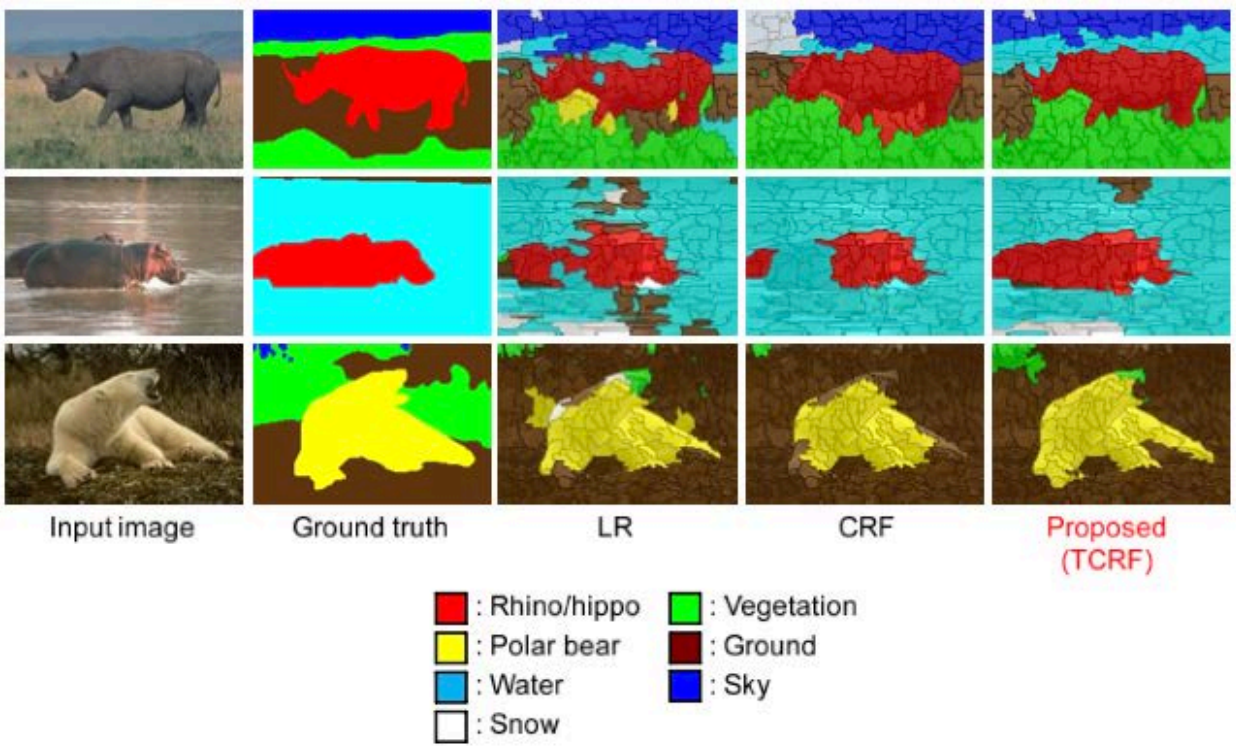

Figure 7: Examples of recognition results from each method

From Table 4 and Table 5, we see the effectiveness of the method that is based on sparse and hierarchical representation, regardless of the dataset. To show the details of the results, we provide some examples of the recognition results in Figure 7. First we compare the conventional methods "LR" and "CRF". The local errors are improved with "CRF" due to considering the co-occurrence of the labels in adjacent regions, resulting in better performance. Our method further considers the hierarchical structure, and comes up with labeling results that are more natural and closer to the ground truth.

5. CONCLUSION

In this paper, we proposed an effective object recognition method, which suitably combines a semi-supervised approach (Semi-supervised Canonical Correlation Analysis; semi-CCA), sparse representation (Regularized Orthogonal Matching Pursuit; ROMP), and hierarchical representation (Tree Conditional Random Field; TCRF). Semi-supervised learning has the advantage of being able to capture a global structure of the true data distribution even when given only a small amount of labeled training data. However, outliers included in unsupervised data often negatively affect the classifier construction. Our approach suppresses such outliers in terms of sparse representation in the subspace that is created using semi-CCA. Furthermore, we adopt TCRF in order to be robust to scale variance, which often causes errors in super-pixel-based object recognition.

The experimental results using two datasets showed the effectiveness of our proposed method satisfactorily. While conventional semi-supervised approaches decreased the labeling accuracy compared with their supervised methods, our sparse-representation-based approach, on the contrary, increased the accuracy, taking full advantage of semi-supervised learning. When we also applied hierarchical representation, we further obtained better results than when the non-hierarchical structured approach is used.

In our method, we used sparse representation and hierarchical representation in a tandem manner. We would like to investigate the integration of these methods into one framework in the future.

REFERENCES

[1]. X. Zhu, Semi-supervised learning literature survey. Proc. IEEE International Conference on Machine Learning (ICML), tutorial, 2007.

[2]. T. Joachims, Transductive inference for text classification using support vector machines. Proc. IEEE Inter6national Conference on Machine Learning (ICML), pp. 200–209, 1999.

[3]. K. Nigam, A. Mccallum, and T. Mitchell, Semi-supervised text classification using EM. In Semi- supervised Learning, pp. 33–56, 2006.

[4]. A. Kimura, et al., Semicca: Efficient semi-supervised learning of canonical correlations. Proc. IEEE International Conference on Pattern Recognition (ICPR), pp. 2933–2936, 2010.

[5]. M. Culp and G. Michailidis, An iterative algorithm for extending learners to a semi-supervised setting. Journal of Computational and Graphical Statistics, pp. 545– 571, 2008.

[6]. M. Elad, M. A. T. Figueiredo, and M. Yi, On the role of sparse and redundant representations in image processing. Proc. IEEE Special Issue on Applications of Sparse Representation and Compressive Sensing, pp. 972–982, 2010.

[7]. D. Needell and R. Vershynin, Uniform uncertainty principle and signal recovery via regularized orthogonal matching pursuit. Foundations of Computational Mathematic, pp. 317–334, 2009.

[8]. X. Ren, J. Malik, Learning a classification model for segmentation. Proc. IEEE International Conference on Computer Vision (ICCV), pp. 10-17, 2003.

[9]. P. F. Felzenszwalb, D. P. Hutenlocher, Efficient graph-based image segmentation. International Journal of Computer Vision (IJCV), vol. 59, no. 2, pp. 167-181, 2004.

[10]. Takeshi Okumura, Tetsuya Takiguchi, Yasuo Ariki, Generic Object Recognition by Tree Conditional Random Field Based on Hierarchical Segmentation. ICPR2010, pp. 3025-3028, 2010.

[11]. E. Sharon, A. Brandt, and R. Basri, Fast multiscale image segmentation. Proc. IEEE Computer Vision and Pattern Recognition, pp 70-77, 2000.

[12]. J. Shi and J. Malik, Normalized cuts and image segmentation. Proc. IEEE Computer Vision and Pattern Recognition, pp. 731–737, 1997.

[13]. J. Wright, A. Ganesh, S. Rao, and Y. Ma, Exact recovery of corrupted low-rank matrices by convex optimization. Proc. IEEE, 2009.

[14]. A. Y. Yang, R. Jafari, S. S. Sastry, and R. Bajcsy, Distributed recognition of human actions using wearable motion sensor networks. Journal of Ambient Intelli- gence and Smart Environments, pp. 103–115, 2009.

[15]. J. D. Lafferty, A. McCallum, and F. C. N. Pereira, Conditional random fields: Probabilistic models for segmenting and labeling sequence data. Proc. International Conference on Machine Learning, 2001.

[16]. C. M. Bishop, Pattern Recognition and Machine Learning. Springer, Chapter. 8, 2006.

[17]. Stanford artificial intelligence robot (stair) image dataset. http://cs.stanford.edu/group/stair/

[18]. P. Duygulu, K. Barnard, J. de Freitas, and D. Forsyth, Object recognition as machine translation: learning a lexicon for a fixed image vocabulary. ECCV, pp. 97-112, 2002.

[19]. J. Shotton, J. Winn, C. Rother, and A. Criminisi, Textonboost: joint appearance, shape and context modeling for multi-class object recognition and segmentation., Proc. IEEE European Conference on Computer Vision, pp. l-15, 2006.

[20]. S. Gould, J. Rodgers, D. Cohen, G. Elidan and D. Koller, Multi-Class segmentation with relative location prior. International Journal of Computer Vision, pp. 300-316, 2008.