Spatial Co-Training for Semi-Supervised Image Classification

Yi Hong
YHong@walmartlabs.com

@WalmartLabs
San Bruno, CA 94066, USA

Abstract

Co-training is a famous learning algorithm used when there are small amounts of labeled data and large amounts of unlabeled data, but it has limited applications in image classification due to the unavailability of two independent and sufficient representations of the images. In this paper, we propose a novel co-training algorithm in which these two independent and sufficient representations are automatically learned from the data. We call it as the spatial co-training algorithm (SCT). The main idea of SCT is to divide an image into two subregions and consider each of them as an independent representation. In SCT, the division of the image is firstly learned by an EM style algorithm on small amounts of labeled images, and finally relearned by a co-training style algorithm on many confident unlabeled images; while the classification of the image is performed jointly with the division of the image. We validate the proposed method by experimental results on several image datasets.

1 Introduction

Recent years have witnessed increasing interest in image classification. This interest resulted in many effective approaches that progressed the computer vision field very fast [4, 5, 9, 19, 22]. However most of these approaches are sensitive to their amounts of labeled images. A typical example is the experimental result on the Caltech-101 dataset [9]: accuracies of kNN [4], SVM [19] and Random Forests [5] are around 75%, 78% and 88% respectively if 30 images per category are labeled; but their performance degrades a lot, just around 65%, 70% and 70% if this number reduces to 15. Several effective methods have been proposed so far to improve the accuracy of the classifiers by the way of making use of large amounts of unlabeled data [3, 11, 13, 23]. Among them, the co-training algorithm assumes that each example has two different independent representations with sufficient discriminative ability for a good classification [3, 12, 21, 24]. It has been widely applied in the fields of email classification [14], web page mining [3] and visual tracking [15, 20]. But we have not observed many of its applications in the field of image classification, apart from [10] where the content of the image and its tags are used as two independent representations for web image classification and [1] where contour and skeleton are considered as two complementary representations for shape retrieval. As pointed out in [8], this is because it is usually hard to obtain such two independent and sufficient representations of a single image. Nevertheless, several recent studies have shown that the above assumption of the co-training algorithm is too strong and can be relaxed a lot [2, 6, 7, 17]. In [2], the authors proved that a weaker expanding property on the data is enough for the success of the co-training algorithm. They suggested that the co-training algorithm should work well if there are at least some cases when the classifier on one representation makes confident decisions while the classifier on the other representation does not have much confidence in its own decision. The above weaker expanding property of the co-training algorithm has been well demonstrated in [6, 17, 18] where the authors showed that a random split of a nature single feature set usually makes the co-training algorithm success and in [7] where the authors proposed an elegant algorithm to automatically decompose a single feature set into two complementary subsets as inputs of the co-training algorithm.

Inspired by the above weaker expanding property, in this paper we propose a novel co-training algorithm in which these two independent and sufficient representations are roughly learned from the image. Another inspiration of the proposed algorithm comes from the property of the image classification task itself. As for the image classification task, the information contained in a whole image is usually redundant. In this case, a subregion of the whole image is usually sufficient enough for the classifier to make a confident prediction on it. For example, if our task is to do human vs. non-human image classification, and if we have already known there is a face in the image, we can make sure that the image belongs to the human image, without further studying if there are two legs or not.

2 The spatial co-training algorithm

We start the discussion from notations of traditional co-training algorithm [3]. Let $L = \{(x_i^1, x_i^2, y_i), i = 1, ..., l\}$ denote the set of labeled examples and $U = \{(x_i^1, x_i^2), i = l+1, ..., l+u\}$ denote the set of unlabeled examples, where x_i^1 and x_i^2 are two different representations of the same example x_i. In traditional co-training algorithm, these two different representations are assumed independent and both of them are strong enough to make a good classification.

2.1 Problem definition

Our proposed algorithm is based on the bag-of-words model [9] that represents each image as the histogram of its local image patches. In particular, the bag-of-words model performs to: (1) extract a collection of local descriptors such as sift [16] from the images; (2) quantize them as indexes; (3) and represent each image as the histogram of indexes of its local image patches. A lot of visual codebook learning algorithms have been proposed so far, and in this paper we employ the k-means clustering algorithm because of its simplicity and wide applications.

Given a set of labeled images $\{I_1, ..., I_l\}$, we extract local sift descriptors densely from each image and express them as a matrix F:

$$F = \{f_{ij} | i = 1, ..., w; j = 1, ..., v\}, \tag{1}$$

where f_{ij} is the sift descriptor of the local image patch (i, j), w and v are the height and width of the matrix of the sift descriptor [16]. To simplify our notations, in the following paragraphs we assume that all images have the same size and thus have the same values of w and v of the matrix F. Then we apply the k-means clustering algorithm to quantize these local sift descriptors into indexes and rearrange indexes from one image as a matrix C:

$$C = \{c_{ij} | i = 1, ..., w; j = 1, ..., v\}, \tag{2}$$

where $c_{ij} \in \{1,...,K\}$ is the visual code of the local image patch (i,j) and K is the codebook size. As illustrated in the introduction section, the basic idea of the proposed algorithm is to partition an image into two subregions and consider each of them as an independent representation. Now suppose that I_1^s and I_2^s are a partition of the image I that corresponds to a partition C_1^s and C_2^s of all visual codes in C:

$$C_1^s \bigcap C_2^s = \emptyset; \quad C_1^s \bigcup C_2^s = C, \tag{3}$$

we calculate the histograms of visual codes in both I_1^s and I_2^s as:

$$h_{1,k} = \frac{\sum_{(i,j)\in C_1^s} \delta(c_{ij},k)}{\sum_{(i,j)\in C_1^s} \mathbf{1}}; \quad h_{2,k} = \frac{\sum_{(i,j)\in C_2^s} \delta(c_{ij},k)}{\sum_{(i,j)\in C_2^s} \mathbf{1}}, \tag{4}$$

after normalized for $k = 1,...,K$; and $\delta(a,b) = 1$ if $a = b$, and $\delta(a,b) = 0$ if $a \neq b$. According to (4), each division (C_1^s, C_2^s) of the image will lead to a representation pair (h_1, h_2) of the image:

$$(C_1^s, C_2^s) \Rightarrow (h_1, h_2), \tag{5}$$

where both h_1 and h_2 are the histograms with K bins. Consider the overall number of divisions of each image is around wv that is too large to be processed, in this paper we restrict the division of the image as a vertical line. In this case, the partition of C at the position d is $(C_{1:w,1:d}, C_{1:w,d+1:v})$, and we simplify it as $(C_{1:d}, C_{d+1:v})$ together with their histogram representation pair as $(h_{1:d}, h_{d+1:v})$. Therefore, the candidate pool of all possible representation pairs is:

$$H = \{(h_{1:d}, h_{d+1:v})| \ d = 1,...,v-1\}. \tag{6}$$

Based on the above notations, the proposed algorithm in this paper aims to learn good divisions $\{d_1,...,d_\ell,d_{\ell+1},...,d_{\ell+u}\}$ of both labeled and unlabeled images such that the co-training algorithm using $\{(h_{i,1:d_i}, h_{i,d_i+1:v})|i = 1,...,\ell;\ell+1,...,\ell+u\}$ as inputs can succeed. Two points should be mentioned about the above definition: (1) different images may have different good values of d; (2) the above restriction of the division of the image may not be optimal, but our experimental results indicate that it works quite well.

2.2 Solution

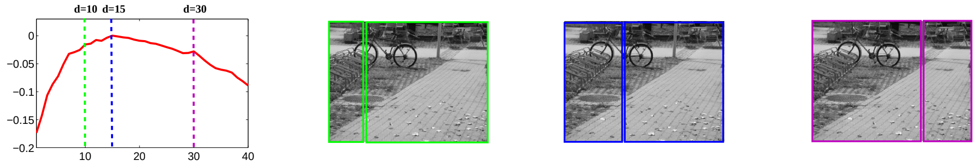
Our solution to obtaining $\{d_1,...,d_\ell,d_{\ell+1},...,d_{\ell+u}\}$ includes two folds: (1) first we learn divisions of labeled images $\{d_1,...,d_\ell\}$ by the EM style algorithm; (2) then we learn divisions of unlabeled images $\{d_{\ell+1},...,d_{\ell+u}\}$ by the co-training style algorithm. In the following paragraphs, we will go into details of describing the above two approaches.

2.2.1 Learning divisions of labeled images

The proposed algorithm works to divide each image into two subregions and consider each of them as an independent representation. Let $\{I_1,...,I_\gamma\}$ denote the set of labeled positive images and $\{d_1,...,d_\gamma\}$ denote their current divisions, based on the bag-of-words model we get the following representation pairs of these positive training images with respect to their current divisions:

$$\{(h_{1,1:d_1}, h_{1,d_1+1:v}),...,(h_{\gamma,1:d_\gamma}, h_{\gamma,d_\gamma+1:v})\}, \tag{7}$$

where $\{h_{1,1:d_1},...,h_{\gamma,1:d_\gamma}\}$ and $\{h_{1,d_1+1:v},...,h_{\gamma,d_\gamma+1:v}\}$ are their first and second representations respectively. Note like the whole image, both of these two subregions have their own



(a) (b) (c) (d)

Figure 1: An example of $f(d_i) = \min\left\{\frac{w^L\cdot\phi(h_{i,1:d_i})-\rho^L}{\sum_{v'}|w^L\cdot\phi(h_{i,1:v'})-\rho^L|}, \frac{w^R\cdot\phi(h_{i,d_i+1:v})-\rho^R}{\sum_{v'}|w^R\cdot\phi(h_{i,v'+1:v})-\rho^R|}\right\}$ where $d_i = 1,...,40$. Values of $f(d_i)$ with three different values of $d_i = \{10,15,30\}$ are highlighted in: (a), and their corresponding divisions of the image are shown in: (b) $d_i = 10$; (c) $d_i = 15$; (d) $d_i = 30$. It is observed that a higher value of $f(d_i)$ leads to a better division of the image.

meanings of a particular part of an object or a frequently appearing object in a scene, therefore we apply the one-class SVM to describe them as svm^L and svm^R:

$$\begin{aligned} &svm^L \\ &\min \quad \tfrac{1}{2}\|w^L\| + \tfrac{1}{\nu^L\gamma^L}\textstyle\sum_i \xi_i^L - \rho^L, \qquad s.t.\;\; w^L\cdot\phi(h_{i,1:d_i}) \geq \rho^L - \xi_i^L, \quad \xi_i^L \geq 0 \end{aligned} \tag{8}$$

$$\begin{aligned} &svm^R \\ &\min \quad \tfrac{1}{2}\|w^R\| + \tfrac{1}{\nu^R\gamma^R}\textstyle\sum_i \xi_i^R - \rho^R, \qquad s.t.\;\; w^R\cdot\phi(h_{i,1+d_i:v}) \geq \rho^R - \xi_i^R, \quad \xi_i^R \geq 0 \end{aligned} \tag{9}$$

where ρ^L and ρ^R are offsets of each example from the origin, ν^L and ν^R are shared trade-off parameters, and ξ_i^L and ξ_i^R are slack variables for the example. Now suppose the above two one-class SVMs are known and fixed, given an image I_i with the division position d_i, we calculate the decision functions:

$$(w^L\cdot\phi(h_{i,1:d_i}) - \rho^L, \qquad w^R\cdot\phi(h_{i,1+d_i:v}) - \rho^R), \tag{10}$$

that measures how well does the representation pair $(h_{i,1:d_i}, h_{i,1+d_i:v})$ fit the current model (svm^L, svm^R), then the division of the image that fits the model best can be found as:

$$d_i = \arg\max_d \;\; \min\left\{\frac{w^L\cdot\phi(h_{i,1:d}) - \rho^L}{\sum_{v'}|w^L\cdot\phi(h_{i,1:v'}) - \rho^L|}, \frac{w^R\cdot\phi(h_{i,d+1:v}) - \rho^R}{\sum_{v'}|w^R\cdot\phi(h_{i,v'+1:v}) - \rho^R|}\right\}. \tag{11}$$

Therefore we have the following EM style algorithm to learn divisions of labeled images:

```
• Input
    – A set $\{I_1,...,I_\gamma\}$ of labeled positive training images
    – A set $\{I_{\gamma+1},...,I_\ell\}$ of labeled negative training images
• Process
  (1) Set initial divisions $\{d_1,...,d_\gamma\}$ of the images
  (2) Loop:
      (a) $(svm^L, svm^R)$ ←Fix $(d_1,...,d_\gamma)$ and train two one-class SVMs
      (b) $(d_1,...,d_\gamma)$ ←Fix $(svm^L, svm^R)$ and update divisions of the images
      End loop
  (3) Determine divisions $\{d_{\gamma+1},...,d_\ell\}$ of negative training images
```

Figure 2: The EM style algorithm to learning divisions of labeled images.

Note the above algorithm is unstable due to different initial values of d_i. In this paper, we set:

$$d_i = \frac{v}{2}, \tag{12}$$

in order to make an initial division without bias, for $i = 1, ..., \gamma$. Fig. 1 gives an example that shows $\min\left\{ \frac{w^L \cdot \phi(h_{i,1:d_i}) - \rho^L}{\sum_{v'} |w^L \cdot \phi(h_{i,1:v'}) - \rho^L|}, \frac{w^R \cdot \phi(h_{i,d_i+1:v}) - \rho^R}{\sum_{v'} |w^R \cdot \phi(h_{i,v'+1:v}) - \rho^R|} \right\}$ with all possible values of d_i on a typical image after the EM algorithm converges.

2.2.2 Learning divisions of unlabeled images:

Let (svm^L, svm^R) denote two one-class SVMs learned by the EM style algorithm, we get divisions $\{d_{\ell+1}, ..., d_{\ell+u}\}$ of all unlabeled images by (11) and their corresponding representation pairs $\{(h_{i,1:d_i}, h_{i,d_i+1:v}) | i = \ell+1, ..., \ell+u\}$ by (4). Then we perform to classify unlabeled images by the way of considering these representation pairs as inputs of the co-training algorithm. Fig. 3 shows the whole framework of our proposed algorithm, which we call the spatial co-training algorithm (SCT).

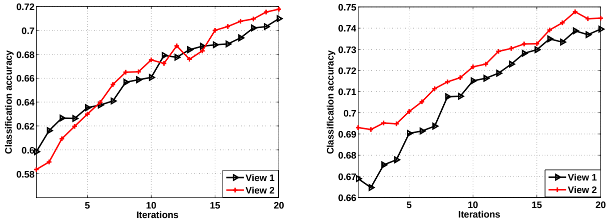
```
• Input
    – A set $L$ of labeled training images
    – A set $U$ of unlabeled images
• Process
  (1) Learn divisions $D = \{d_1, ..., d_\ell\}$ of labeled images in $L$ by EM style algorithm
  (2) Create a pool $U'$ of examples by choosing $u'$ examples at random from $U$
  (3) Loop for $k$ iterations
      (a) Learn divisions $\{d'_1, ..., d'_{u'}\}$ of unlabeled images in $U'$
      (b) Train a classifier $F_1$ on $\{h_{1,1:d_1}, ..., h_{\ell,1:d_\ell}\}$
      (c) Train a classifier $F_2$ on $\{h_{1,d_1+1:v}, ..., h_{\ell,d_\ell+1:v}\}$
      (d) Label $p$ positive and $n$ negative on which $F_1$ is most confident
      (e) Label $p$ positive and $n$ negative on which $F_2$ is most confident
      (f) Add these self-labeled images to $L$ and their divisions to $D$
      (g) Update $(svm^L, svm^R)$ by training two one-class SVMs on augmented $D$
      (h) $\ell = \ell + 2p + 2n$
      (i) Randomly choose $2p + 2n$ examples from $U$ to replenish $U'$
     End loop
  (4) Train a single view classifier $F$ on the augmented training dataset $L$
  (5) Label all the rest of unlabeled images by $F$
```

Figure 3: The spatial co-training algorithm (SCT).

In Fig. 3, confident unlabeled images are not only used for updating the classifiers for image classification, but also used for re-training (svm^L, svm^R) for the division of the image. In addition, the co-training process terminates after a certain number of iterations, followed by a single view image classification scheme that represents each image by the histogram of its local sift descriptors in the whole image instead of its subregions. As for classifiers of F, F_1 and F_2, there are many different types to choose from, and in this paper we employ the linear SVM with probability output.

Table 1: Classification accuracies(%). SVM$_S$: the support vector machine in which only small amount of images are labeled; SVM$_F$: the support vector machine in which large amount of images are labeled; CT: the co-training algorithm in which the division of the image is fixed; ST: the self-training algorithm.

Datasets	SVM$_S$	SVM$_F$	CT	ST	SCT
Caltech-4 Face	90.5 ± 1.6	94.2 ± 1.0	91.6 ± 3.5	89.0 ± 2.6	94.6 ± 2.0
Caltech-4 Motorbike	82.7 ± 3.3	89.6 ± 1.4	85.8 ± 3.2	83.6 ± 1.8	86.5 ± 2.9
Caltech-4 Car	95.5 ± 1.3	97.2 ± 1.7	97.3 ± 3.5	95.3 ± 2.6	97.3 ± 2.5
Caltech-4 Airplane	90.1 ± 0.9	95.3 ± 1.9	93.3 ± 1.9	94.0 ± 1.5	96.1 ± 0.3
Graz02 Bike	63.9 ± 3.1	72.6 ± 2.4	72.5 ± 8.9	67.3 ± 3.5	75.9 ± 3.1
Graz02 Person	70.0 ± 4.3	75.3 ± 2.2	72.4 ± 4.9	71.6 ± 3.6	76.4 ± 2.5
Scene-15 Mountain	85.8 ± 1.8	91.7 ± 1.6	87.2 ± 1.0	86.8 ± 2.3	90.0 ± 1.8
Scene-15 Store	80.4 ± 0.9	82.6 ± 0.1	81.6 ± 1.0	81.7 ± 1.2	82.6 ± 1.2
Scene-15 Office	80.5 ± 3.6	87.0 ± 1.4	89.7 ± 1.7	83.0 ± 2.1	89.8 ± 1.9
Scene-15 Building	85.6 ± 2.3	90.9 ± 2.1	90.4 ± 2.1	85.4 ± 2.7	91.7 ± 1.9



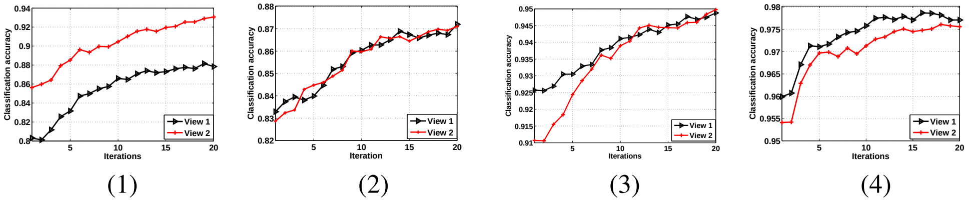
(1) (2)

Figure 4: Classification accuracies of the SCT algorithm at different iterations on: (1) Graz02-Bike; (2) Graz02-Person.

3 Experimental results and analysis

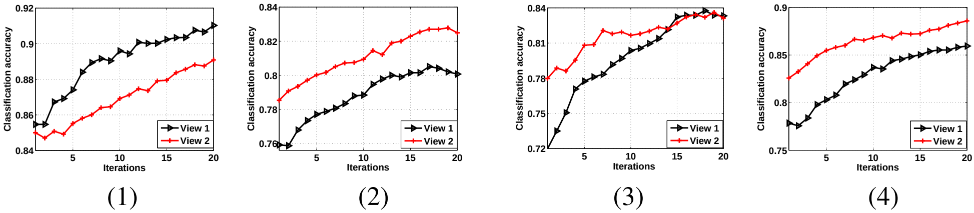
The SCT algorithm was tested to classify between positive and negative images on three widely used image datasets: Caltech-4, Graz02 and Scene-15. Since there are no negative images in the Scene-15 image dataset, we consider images with a specified label as positive images and images with other labels as negative images. Parameter settings in our experiments are given as follows: sift descriptors are extracted densely from each image with local image patch size 32×32 and step size 8; the codebook size K is fixed to 500; among all images, 20 positive and 20 negative images are randomly selected as labeled examples and others are used as unlabeled examples; at each iteration of the co-training algorithm, $p=2$ most confident positive and $n=2$ most confident negative examples are labeled, and the number of iterations is fixed to 20. All experiments were repeated for 10 independent runs and their results were averaged.

First, we compared the SCT algorithm with the following four approaches with respect to their classification accuracies: (1) SVM$_S$ that represents the support vector machine in which only small amounts of images are labeled; (2) SVM$_F$ that represents the support vector



(1) (2) (3) (4)

Figure 5: Classification accuracies of the SCT algorithm at different iterations on: (1) Caltech-4 Human face; (2) Caltech-4 Motorbike; (3) Caltech-4 Airplane; (4) Caltech-4 Car.



(1) (2) (3) (4)

Figure 6: Classification accuracies of the SCT algorithm at different iterations on: (1) Scene-15 Mountain; (2) Scene-15 Store; (3)Scene-15 Office; (4) Scene-15 High building.

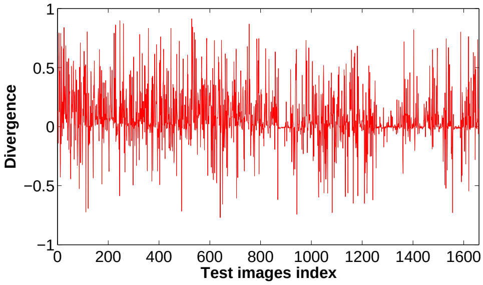


Figure 7: Divergence of SVM's outputs using two representations learned by the SCT algorithm on Caltech-4 Motorbike dataset.

machine in which large amounts of images are labeled; (3) CT that represents the co-training algorithm in which the division of the image is fixed; and (4) ST that represents the self-training algorithm. To guarantee a fair comparison, we set the number of labeled positive and negative images in the SVM$_\mathrm{F}$ algorithm as 100[1], that is much larger than 20 used in other four approaches; in addition, we set the division d_i of the image as $\frac{v}{2}$ for $i = \{1, ..., \ell, \ell+1, ..., \ell+\mu\}$ in the CT algorithm and considered these two resulted subregions as inputs of the co-training algorithm; furthermore, in the ST algorithm we trained a single view support vector machine and again made use of confident unlabeled images for re-training the support vector machine iteratively. Table (1) shows classification accuracies obtained by SVM$_\mathrm{S}$, SVM$_\mathrm{F}$, ST, CT and SCT on the Caltech-4, Graz02 and Scene-15 image datasets. We observe from Table

[1] we set the number of labeled positive and negative images as 100, because the number of positive and negative images in the augmented image dataset is 100 after the co-training process terminates in the SCT algorithm.

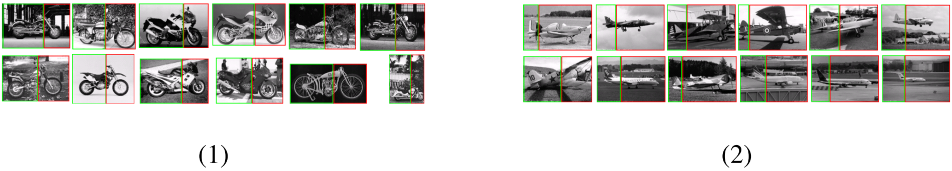


Figure 8: Divisions of images learned by the SCT algorithm on (1) Caltech-4 Motorbikes; (2) Caltech-4 Airplanes.

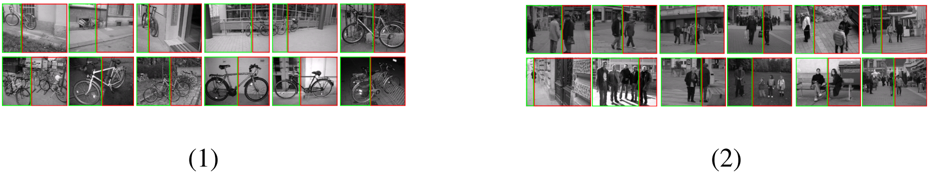


Figure 9: Divisions of images learned by the SCT algorithm on (1) Graz02 Bike; (2) Graz02 Person.

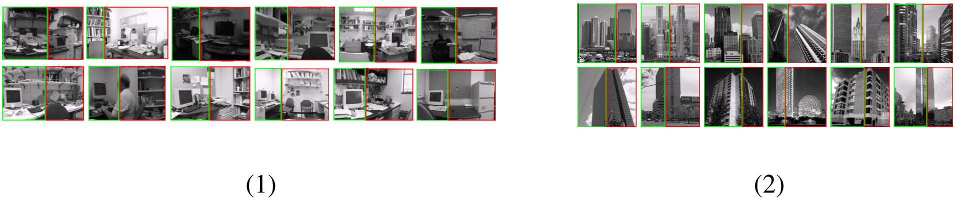


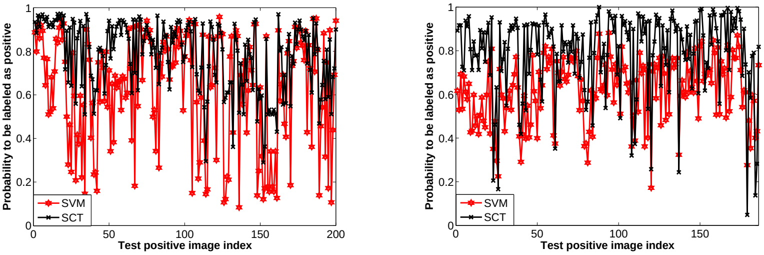
Figure 10: Divisions of images learned by the SCT algorithm on (1) Office; (2) High building.

1 that SCT performs better than CT and ST on all image datasets. It significantly outperforms SVM_S with classification accuracy gains ranging from around 2% on Scene-15 Store dataset to almost 12% on Graz02 Bike dataset. The above experimental results demonstrate the feasibility of the SCT algorithm for image classification in which only small amounts of images are labeled. Another interesting observation is the SCT algorithm achieves higher accuracies than those obtained by the SVM_F algorithm on several datasets. The reason is perhaps the SCT algorithm selects images to be labeled as training examples according to SVM's confidences on them, but which images will be labeled are randomly selected in SVM_F.

Second, we studied classification accuracies on each representation at different iterations of the co-training process. Fig. 4, Fig.5 and Fig. 6 show the experimental results. We observe from these figures that classification accuracies on both representations are increasing with small perturbations during the co-training process. The above experimental results tell us that these two representations learned by the SCT algorithm benefit each other a lot for image classification, and the co-training algorithm succeeds if these two representations are used as its inputs. In addition, we calculated the divergence of SVM's outputs on these two

(1) (2)

Figure 11: Divisions of some typical images learned by the SCT algorithm: (1) before the co-training process; and (2) after the co-training process. It tells us that unlabeled images can help the SCT algorithm learn a better division of the image.



(1) (2)

Figure 12: Probability outputs of SVM and SCT on: (1) Caltech-4 Motorbike; (2) Graz02 Bike.

representations measured as:

$$\chi(I) = F_1(y|I) - F_2(y|I),$$

that is an important factor for the success of the co-training algorithm, where $F_1(y|I)$ and $F_2(y|I)$ are probability outputs on the first and second representation respectively. Fig.(7) shows the experimental result that further demonstrates the feasibility of the SCT algorithm for learning two complementary representations automatically from the image.

Apart from classification accuracies, another target of the SCT algorithm is to learn a meaningful division of the image. Fig. 8, Fig. 9 and Fig. 10 are the division of the image learned by the SCT algorithm. We find that each image is divided into two subregions that are highlighted by a green rectangle and a red rectangle; and these two subregions are usually two parts of an object or two same objects or two different objects of the same scene. To test the usefulness of unlabeled images for the division of the image, we compared the division of the image learned by the SCT algorithm before and after the co-training process. Fig. 11 gives the experimental result. It is observed that the SCT algorithm using unlabeled images achieves better divisions than those obtained by the SCT algorithm without unlabeled images.

Finally, we observed an interesting phenomenon by comparing probability outputs of SVM and SCT on test positive images. Note their only difference is in SVM, the classifier is trained on original labeled images; while in SCT, the classifier is trained on augmented labeled images. The experimental result is given in Fig. 12. We observed from Fig. 12 that the co-training algorithm significantly increases the confidence of the classifier on unlabeled data. We claim that this may be another important reason why can the co-training algorithm succeed in many data classification problems.

4 Conclusions

In this paper, we have introduced a novel co-training algorithm, which we called the spatial co-training algorithm (SCT). The SCT algorithm overcomes the main limitation of traditional co-training algorithm, as it automatically learns two independent and sufficient representations from the data. Its basic idea is to divide an image into two subregions and consider each of them as an independent representation; and the division of the image and the classification of the image are learned jointly by a co-training style algorithm. We have tested the SCT algorithm on several image datasets with only small amounts of labeled images, and very good results were achieved.

References

[1] X. Bai, B. Wang, and Z. Tu. Co-transduction for shape retrieval. *ECCV*, 2010.

[2] M.F. Balcan, A. Blum, and K. Yang. Co-training and expansion: Towards bridging theory and practice. *NIPS*, 2004.

[3] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. *COLT*, 1998.

[4] O. Boiman, E. Shechtman, and M. Irani. In defense of nearest-neighbor based image classification. *CVPR*, 2008.

[5] A. Bosch, A. Zisserman, and Xavier Munoz. Image classification using random forests and ferns. *CVPR*, 2007.

[6] J. Chan, I. Koprinska, and J. Poon. Co-training with a single natural feature set applied to email classification. *IEEE/WIC/ACM Conference on Web Intelligence*, 2004.

[7] M. Chen, K.Q. Weinberger, and Y. Chen. Automatic feature decomposition for single view co-training. *ICML*, 2011.

[8] C. M. Christoudias, R. Urtasun, A. Kapoor, and T. Darrell. Co-training with noisy perceptual observations. *CVPR*, 2009.

[9] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: an incremental bayesian approach tested on 101 object categories. *Workshop of CVPR*, 2004.

[10] H. Feng, R. Shi, and T. Chua. A bootstrapping framework for annotating and retrieving www images. *ACM MM*, 2004.

[11] R. Fergus, Y. Weiss, A. Torralba, and Z. Ghahramani. Semi-supervised learning in gigantic image collections. *NIPS*, 2009.

[12] R. Ghani. Combining labeled and unlabeled data for text classification with a large number of categories. *ICDM*, 2001.

[13] M. Guillaumin, J. Verbeek, and C. Schmid. Multimodal semi-supervised learning for image classification. *CVPR*, 2010.

[14] S. Kiritchenko and S. Matwin. Email classification with co-training. *CASCON*, 2001.

[15] A. Levin, P. Viola, and Y. Freund. Unsupervised improvement of visual detectors using cotraining. *ICCV*, 2003.

[16] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004.

[17] K. Nigam and R. Ghani. Analyzing the effectiveness of cotraining. *CIKM*, 2000.

[18] A. Sharma, G. Hua, Z. Liu, and Z. Zhang. Meta-tag propagation by co-training an ensemble classifier for improving image search relevance. *NIPS*, 2007.

[19] A. Vedaldi, V. Gulshan, M. Varma, and A. Zisserman. Multiple kernels for object detection. *ICCV*, 2009.

[20] Q. Yu, T. Dinh, and G. Medioni. Online tracking and reacquisition using co-trained generative and discriminative trackers. *ECCV*, 2008.

[21] S. Yu, B. Krishnapuram, R. Rosales, H. Steck, and R. Rao. Bayesian co-training. *NIPS*, 2007.

[22] H. Zhang, A.C. Berg, M. Maire, and J. Malik. Svm-knn: Discriminative nearest neighbor classification for visual category recognition. *CVPR*, 2006.

[23] X. Zhu and Z. Ghahramani. Learning from labeled and unlabeled data with label propagation. *ICML*, 2002.

[24] X. Zhu, Z. Ghahramani, and J. Lafferty. Semi-supervised learning using gaussian fields and harmonic functions. *ICML*, 2003.