Hybrid Transfer Learning for Efficient Learning in Object Detection

Masamitsu Tsuchiya, Yuji Yamauchi, Hironobu Fujiyoshi
Dept. of Computer Science
Chubu University
Aichi, Japan
tsuchiya@vision.cs.chubu.ac.jp, yuu@vision.cs.chubu.ac.jp, hf@cs.chubu.ac.jp

Takayoshi Yamashita
OMRON Corporation
Shiga, Japan
takayosi@omm.ncl.omron.co.jp

***Abstract*—In the detection of human from image using statistical learning methods, the labor cost of collecting training samples and the time cost for retraining to match the target scene are major issues. One method to reduce the work involved in sample collection is transfer learning based on boosting. However, if there is a large change between the auxiliary scene and target scene, it is difficult to apply the transfer learning. We therefore propose a hybrid transfer learning method in which two feature spaces are prepared, one of feature obtained by transfer and another of full feature search that is the same as retraining. The feature space is selectively switched on the basis of the defined training efficiency. The proposed method improving accuracy up to 8.35% compared to conventional transfer learning while accelerating training time by 3.2 times faster compared to retraining.**

I. Introduction

There has been much research to date on human detection from images based on statistical learning methods, and its application to intelligent transport systems (ITS) and other fields has already begun. Human detection techniques based on statistical learning methods prepare large quantities of human and background images to train classifiers such as support vector machines (SVMs) and boosting algorithms [1] for classifying such images. There are techniques that extract human silhouettes from gradients using Histograms of Oriented Gradients (HOG) [2] as features as proposed by Dalal. It is also common to create a learning database by collecting human images as positive samples on the order of several thousand and background images as negative samples on the order of tens of thousand. All samples of human images are required to have no human position shifts and to be uniform in terms of human size, aspect ratio, etc. This is because local-area gradients are used as features, which means that the occurrence of human position shifts or differences in human size in the images could prevent the selection of common human features by statistical learning methods.

In response to this problem, the generation of sample images by computer-graphic means has been researched as a method for simplifying the collection of a large volume of high-quality samples [3], [4]. In these studies, it has been possible to obtain many uniformly positioned positive samples, which means that good-quality training samples with no position shifts can be prepared. However, in the case that the environment used for collecting the training samples is different from the target scene in which the human-detection system is being applied, the appearance of people will differ and human-detection performance will drop. For example, if the camera's angle of depression differs between the two environments, the way in which people look in terms of aspect ratio, proportion with respect to body parts, etc., may differ greatly even for the same person. Solving this problem requires that human images be collected from the environment targeted by the human-detection system and that the classifier be retrained. Here, however, creating a dataset to train the human detector for every target scene and performing the training incurs high labor and time costs. One approach to solving this problem is to use transfer learning [5], which has been proposed as a technique for reducing the labor involved in collecting samples and retraining in response to such variations in appearance. Pang et al. performed pre-training using samples obtained from many existing databases [8]. The method they proposed reduces the labor involved in collecting samples by transferring the classifier obtained by pre-training to the target environment and by adapting it to a small number of target samples collected from that environment. The method also reduces the labor involved in retraining by transferring the features deemed effective in the classifier previously trained with existing databases thereby limiting the feature search. However, significant differences between scenes can significantly degrade performance compared to retraining. This is "negative transfer" in the field of transfer learning. Rosenstein et al. have shown that negative transfer can occur when attempting to adapt to very different scenes by transfer learning [10]. Against the above background, we propose a hybrid transfer learning method that prepares two feature spaces –one consisting of features obtained by transfer and the other a full-feature space the same as retraining– and that selectively switches between these feature spaces based on a defined training efficiency. This method makes it possible to create a classifier that minimizes the effects of negative transfer. As a result, it achieves high accuracy even between significantly disparate scenes that have been difficult to handle in conventional transfer learning and speeds up training time compared with retraining.

II. Transfer Learning by Covariate-shift

Transfer learning is a type of learning technique used in the field of machine learning. Although the term “transfer learning” can be interpreted in a number of ways, it has been defined in the call-for-participation announcement of the NIPS 2005 Workshop–Inductive Transfer: 10 Years Later [5] as the “problem of retaining and applying the knowledge learned in one or more tasks to efficiently develop an effective hypothesis for a new task.” Research using transfer learning has been increasing in recent years as reported in papers like SOINN [11], TRAdaBoost [6], and CovBoost [8]. In the study presented in this paper, we use covariate-shift boost (CovBoost) that introduces boosting to transfer learning based on the covariate shift.

A. Covariate-shift Boost (CovBoost)

Covariate-shift boost (CovBoost) is a boosting technique that can use a small quantity of training samples in a new scene and yet maintain the same level of detection accuracy as when using a large quantity of training samples and performing a full-feature search. This is accomplished by applying information on weak classifiers obtained by pre-training and training samples previously used for training to training for the target scene. The CovBoost technique has been proposed by Pang et al. as a means of reducing the labor cost of retraining when appearance differs between standard training samples and training samples for specific scenes targeted for detection, and has been extended for semi-supervised online learning as well [8], [9]. In general, a boosting technique aims to determine the strong classifier $H(x)$ that minimizes the loss function shown by

$$L = \sum_{\Omega} e^{-yH(x)}. \tag{1}$$

Here, Ω is the total number of training samples while x and y correspond to feature and its class label in a training sample. In CovBoost, input samples consist of training samples used for pre-training (auxiliary domain) and training samples used for retraining (target domain). Here, probability density distribution $p_a(x)$ of the auxiliary domain observed in terms of features is generally different than probability density distribution $p_t(x)$ of the target domain, or in other words, $p_a(y|x) \neq p_t(y|x)$. Thus, samples effective for training in the target domain can be selected by weighting samples in the auxiliary domain by $\frac{p_t(y|x)}{p_a(y|x)}$, which is called a covariate loss. This value can be used to insert samples of the auxiliary domain into the target domain, which is a process called transfer leaning by covariate shift. The covariate loss by $\frac{p_t(y|x)}{p_a(y|x)}$ is denoted by the symbol λ. The target function of transfer learning using the covariate shift is given by

$$\tilde{L} = \sum_{(x_i,y_i)\in T_t} e^{-y_i H_t(x_i)} + \sum_{(x_j,y_j)\in T_a} \lambda_j e^{-y_j H_t(x_j)}. \tag{2}$$

Here, $(x_i, y_i) \in T_t$ and $(x_j, y_j) \in T_a$ denote target-sample feature and its class label of target domain t and auxiliary domain a, respectively. The value λ can be calculated as shown in Eq. (3).

$$\lambda = \frac{1 + e^{-yH_a(x)}}{1 + e^{-yH_t(x)}} \tag{3}$$

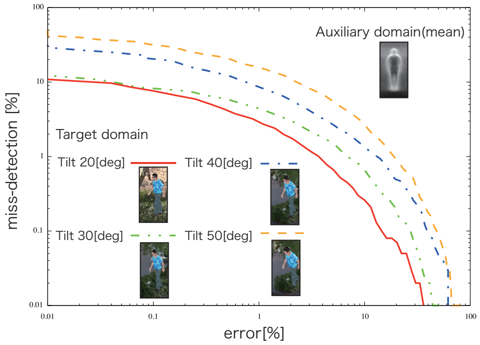

Fig. 1. Variation of appearance of training samples by an angle of depression.

The value λ therefore expresses the adaptiveness of auxiliary-domain samples to the target domain in terms of the classifiers H_a and H_t. A larger value means greater adaptiveness to the target domain.

B. Problems with CovBoost

Pang and others have achieved equivalent performance for environments with different viewpoints by transferring classifiers by CovBoost despite reducing the number of newly collected training samples to one-third the usual amount. However, in the case that camera angle of depression differs significantly, appearance in those images will vary significantly with change in the angle of camera-tilt. As a result, the transfer of features becomes difficult and performance deteriorates. Change in classification performance by transfer learning when changing the camera's angle of depression is shown in Fig.1 as detection error trade-off (DET). Here, we used the INRIA person dataset for pre-training and HOG as features. These results show that performance deteriorates when making a big change in the angle of depression since the appearance of training samples changes. In short, it can be seen that a transferred feature in itself cannot adapt well to a greatly changed target domain.

III. Hybrid Transfer Learning

Transfer learning can be used to achieve high-accuracy classification even when collecting only a small quantity of target training samples, but it cannot adapt if auxiliary scenes and target scenes differ greatly. In response to this problem, we prepare two feature spaces–one consisting of features obtained by transfer and the other of features obtained by retraining–and selectively switch between the transfer-feature space and full-feature space based on training efficiency (Fig.2). We propose this method as a form of “hybrid transfer learning” with the aim of creating a classifier that is faster than retraining and more accurate than conventional transfer learning.

A. Defining the Problem

In this study, we define learning by standard data as the auxiliary domain and data of specific scenes in an actual installation environment as the target domain.

Auxiliary domain

Since learning in the auxiliary domain can be done by offline processing, it uses a large amount of standard data. For the auxiliary domain of this study, we used 2,416 human

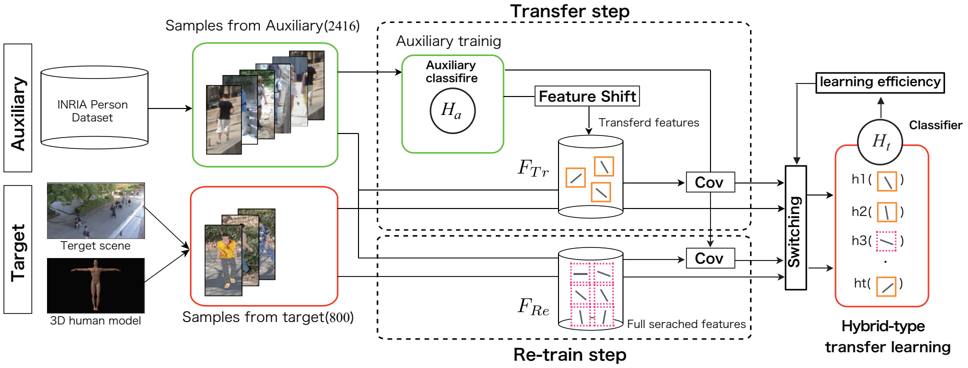

Fig. 2. Hybrid transfer learning.

images from the INRIA person dataset [12]–a standard dataset for human-detection purposes–and use a classifier trained by AdaBoost.The INRIA person dataset is widely known as a benchmark for pedestrian detection and as effective data for detecting upright humans at a low angle of depression.

- Number of positive/negative samples: 2,416/12,180
- Scenes: Camera tilt = 0; front-facing human images

Target domain

For the target domain, we used specific scenes having different camera angles of depression. In contrast to auxiliary-domain data, target-domain data must be newly obtained, which means that getting by with as small a number of samples as possible is desirable. In this study, we saved on labor by obtaining samples through the creation of human images by computer-graphic means as described in [4]. Specifically, we generated 800 specific scenes for each angle of depression used, namely 20, 30, 40, and 50, and took these scenes to be the target domain.

- Number of positive/negative samples: 800/12,180
- Scenes: Camera tilt = 20, 30, 40, 50

In this study, we train a classifier at high speed while maintaining accuracy using a small amount of new samples (one-third that of the auxiliary domain in [8]) by selectively switching to the feature space used for retraining when performing transfer learning for the problem defined above.

B. Feature Shift

CovBoost performs feature transfer as a preparation to learning. First, as shown in Fig. 3(1), it determines the center coordinates of the local feature of a weak classifier selected by pre-training. It then generates L local candidate areas with these coordinates as center by normal random numbers as shown in Fig. 3(2). According to [8], L = 50 is an appropriate number of candidates. At this point, the proposed method determines histograms of local features from the candidate areas and compares each of them with the histogram of the local feature of the weak classifier selected in pre-training to assess their similarity (Fig. 3(3)). The Bhattacharyya coefficient shown by the following equation is used to calculate histogram similarity.

$$Bhattacharyya = \sum_{i=1}^{n} \sqrt{p(x)q(x)} \quad (4)$$

Here, $p(x)$ and $q(x)$ denote the probability density distributions of different domains. Finally, we treat the transfer candidate having the highest similarity to the weak classifier selected in pre-training as a feature to be transferred and define the set of all such transferred features as transfer-feature space F_{Tr}. We also define all features extracted from images the same as in retraining as full-feature space F_{Re}. Average similarity for each camera-tilt is listed in TABLE I. As touched upon in section 2.2, similarity drops the more that target data departs from training data in the auxiliary domain. This difference can be said to be a factor in the degradation of classifier performance shown in Fig. 1.

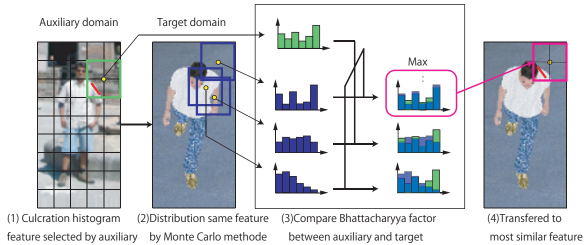
Auxiliary domain
Target domain
Max
(1) Culcration histogram feature selected by auxiliary
(2)Distribution same feature by Monte Carlo methode
(3)Compare Bhattacharyya factor between auxiliary and target
(4)Transfered to most similar feature

Fig. 3. Transferring HOG feature for Pedestrian detection.

TABLE I
VARIATION OF THE SIMILARITY BY CAMERA TILT.

Camera tilt[deg]	20	30	40	50
Bhattacharyya coefficient	0.975	0.974	0.970	0.967

C. Learning by the Hybrid Transfer Learning Method

Hybrid-type learning uses a group of samples extracted from both auxiliary domain T_a and target domain T_t. All of these samples have a class label set to $+1$ for positive samples and -1 for negative samples. The next step is to initialize sample weights. Here, the normalized values of the target domain and auxiliary domain are taken be the initial values of those domains with the weights of each denoted as $D_t(x_i)$ and $D_a(x_j)$. The selection of a weak classifier is achieved by determining $h(x)$ so as to minimize Eq. (2). Here, we get the approximate formula of Eq. (5) by subjecting Eq. (2) to a first-order Taylor expansion with $h(x) = 0$. Determining $h(x)$ so as to minimize Eq. (5) selects weak classifier $h_m(x)$.

$$\begin{aligned} h_m = \arg\min_{h_t}(& \sum_{(x_i,y_i)\in T_t} e^{-2y_i D_t(x_i)} y_i h_t(x_i) \quad (5) \\ & + \sum_{(x_j,y_j)\in T_a} \lambda_j e^{-2y_j D_a(x_j)} y_j h_t(x_j)) \end{aligned}$$

In this case, we determine each $h()$ by searching through transfer-feature space F_{Tr} and full-feature space F_{Re}. We next

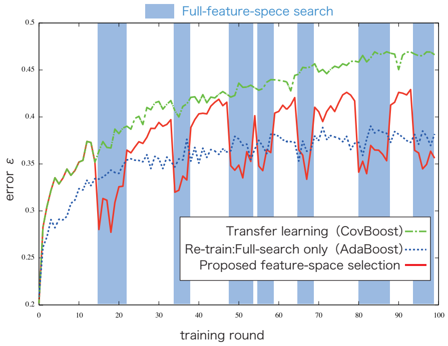

Fig. 4. Feature space selection. (green is error of transfer, blue is error of full-search and red is proposed: Error greatly improves whenever moving to a full search and that the system moves back when error has dropped.

calculate error ϵ_m by

$$\epsilon_m = \frac{\displaystyle\sum_{h(x_i)\neq y_i} e^{-2y_i D_t(x_i)} + \sum_{h(x_j)\neq y_j} \lambda_j e^{-2y_j D_a(x_j)}}{\displaystyle\sum_i e^{-2y_i D_t(x_i)} + \sum_j \lambda_j e^{-2y_j D_a(x_j)}}. \quad (6)$$

At this point, we calculate training efficiency ζ and perform re-selection of weak classifiers in full-feature space F_{Re} if the value of ζ is equal to or less than a certain threshold. The method for calculating ζ is explained in the next section. Next, we calculate weight α_m for the selected weak classifier by

$$\alpha_m = \frac{1}{4}\ln\frac{1-\epsilon_m}{\epsilon_m}. \quad (7)$$

Next, we update the weight of the training samples as

$$D(x) = D(x)e^{-2y\alpha_t h_m(x)} \quad (8)$$

The above process is repeated the same number of times as the number of training rounds in pre-training. Finally, by weighting all weak classifiers and taking a majority vote, we create a strong classifier for detection purposes. This strong classifier is given by

$$H_t(x) = sign[\sum_{t=1}^{M}\alpha_t h_t(x) - th]. \quad (9)$$

Here, th denotes the threshold value and M the number of training rounds.

D. Feature Space Selection based on Training Efficiency

In feature transfer as described in the previous section, transfer learning takes place in transfer-feature space F_{Tr} consisting of features with high transfer likelihood. This approach can reduce training time (lower search cost), but if there are significant changes between pre-training data and target training data, it may happen that there are no similar features to be observed in the first place. It is for this reason why switching is performed between the proposed transfer-feature space based on likelihood and the full-feature space the same as that of retraining. In short, we use high-speed transfer learning based on likelihood in the case that transferring is effective and learning by full-feature space in the case that transferring is difficult. These feature spaces are defined as follows.

Transfer-feature space

- Feature dimensions: 100 (selected by pretraining)
- Computational cost: low
- Degraded performance if difference between domains is large

Full-feature space

- Feature dimensions: 3,780×100(dim×threshold)
- Computational cost: high
- Optimized for target domain

Here, the index needed for switching is a value that judges whether transfer learning is sufficient enough for classification purposes. Change in weak-classifier error in both transfer learning and retraining boosting is shown in Fig.4. Here, error ϵ can be computed by Eq. (6). It increases in value as learning proceeds in both transfer learning and retraining. This is because the weight of adaptively difficult samples increases in value even though learning is progressing. Error rises easily, in particular, if scenes in transfer learning are greatly different since the limit of classification performance is low in this case. If the variation in this error value is low, learning will converge to some extent and there will be no significant improvement. We therefore define the gradient of this error value as training efficiency, which we use as an index for switching. Specifically, we observe the slope of this error, and if it turns out to be a gentle slope as transfer learning proceeds and if its absolute value falls below the threshold, we apply a full search. In this study, we calculate this gradient by a least squares approximation using the most recent five points. Since error ϵ drops significantly if effective features can be discovered by a full search, the gradient will increase and the system will move back to transfer learning. Change in error when switching between feature spaces using training efficiency is shown in Fig.4. These results show that error greatly improves whenever moving to a full search and that the system moves back to transfer learning when error has sufficiently dropped.

IV. Experiment

We performed an experiment to assess the effectiveness of the proposed method on the basis of classification accuracy and training speed.

A. Overview of Experiment

To begin with, we pre-trained a classifier based on HOG features and AdaBoost. For HOG features, cell size was set to 8 and block size to 2 and the total number of dimensions was 3780. For pre-training samples, we used 2,416 human images from the INRIA person dataset as positive samples and 12,180 non-human images as negative samples. For target training samples, we used 800 computer-generated human images as positive samples for angles of depression of $20 \sim 50$[deg] and 12,180 background images as negative samples. Furthermore, we used 2,416 computer-generated human images for retraining as a comparison technique without performing any pre-training.Finally, as evaluation samples, we used 10,000 computer-generated Human images and 10,000 background images. We evaluated performance in terms of equal error rate (EER), which is the value at which the false-positive rate is equal to the false-negative rate. A low EER signifies high accuracy.

TABLE II
ACCURACY EVALUATION BY EER.

Camera tilt[deg]	20	30	40	50
Proposed method[%]	2.26	6.18	8.37	6.37
Transfer learning[%]	3.85	10.56	16.72	15.61
Re-train[%]	0.08	1.07	1.45	1.02

TABLE III
COMPARISON OF TRAINING COST.

Camera tilt[deg]	20	30	40	50
Proposed method [min]	15.0	13.8	18.6	14.4
Re-train [min]	60			

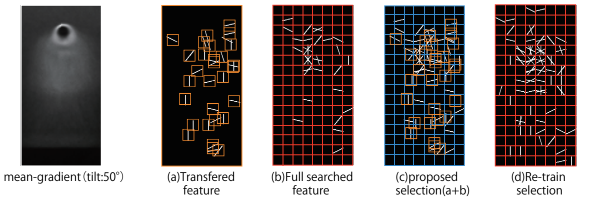

Fig. 5. Selected HOG Feature : (a)selected transfer-step, (b)selected retrain-step, (c)(a)+(b) selected hybrid transfer learning, (d)selected retrain

B. Change in Accuracy by the Feature Space Selection Method

To assess the effectiveness of the proposed hybrid transfer learning method, we compared it with the existing transfer learning and retraining methods focusing on target scenes having a disparity with pre-training scenes. The classification performances of these methods for four types of target scenes corresponding to different camera angles of depression are compared in TABLE II.

These results show that the performance of the existing transfer learning method dropped significantly with change in scenes. In contrast, the proposed method, while inferior to retraining, demonstrated greatly improved performance of 1.59% $\sim$ 8.35% compared with transfer learning even for significant scene changes as the ones here. Retraining had the highest performance since it could be applied to the target domain with an ample number of samples without relying on the auxiliary domain.

C. Comparison of Training Speeds

The proposed method demonstrated a level of accuracy near that of retraining for a small number of samples the same as that of transfer learning. Retraining, however, is a method applied to target scenes, which means that sample collection cost and computational training cost must be taken into account. We therefore compared the proposed method and retraining method with this in mind. Their respective training costs are listed in TABLEIII. The proposed method achieved training times 3.2 $\sim$ 4.1 times faster than that of re-training.These results suggest that the proposed method could be easily applied to raising classification accuracy without incurring a large training cost that would normally be expected when adding data, creating cascade structures, etc.

D. Discussion

The proposed method maintains high accuracy by supplementing the classification process with features from a full search in response to large changes in scenes that transfer learning cannot deal with. Among the features selected by the proposed method, Fig.5 visualizes those selected transfer features (a) and full search (b). It can be seen from Fig.5(a) that standard shoulder edges and vertically oriented edges of legs could be transferred. On the other hand, it can be seen from the full search of Fig.5(b) that horizontal edges are conspicuous and that features that adapted to changes in the appearance of upper body parts owing to changes in the angle of depression were selected. The overlaying of (a) and (b) as the entire proposed method is shown in Fig.5(c) and the features selected by retraining are shown in Fig.5(d). Comparing the two methods, it can be seen that the positional relationships and gradient directions of the features are similar, which means that the proposed method of Fig.5(c) can obtain a configuration of features nearly the same as the retraining method of Fig.5(d) by combining transfer features with a full search.

V. CONCLUSION

The method proposed in the paper prepares two feature spaces–a transfer-feature space obtained by transferring features in transfer learning and a full-feature space the same as retraining–and adaptively switches between them. By selecting space according to training efficiency based on the gradient of weak-classifier error ϵ, we have improved training performance by 1.59% $\sim$ 8.35% compared with conventional transfer learning and increased training speed by more than 3.2 times compared with retraining. Looking forward, we plan to expand our method beyond covariate-shift-type transfer learning and to develop high-accuracy classifiers through Real AdaBoost and other approaches.

REFERENCES

[1] Y. Freund and R.E. Schapire, “Experiments with a new boosting algorithm,” ICML, pp.148–156, 1996.

[2] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” In CVPR, pp.886–893, 2005.

[3] J. Marn, D. Vzquez, D. Gernimo, and A.M. Lpez, “Learning appearance in virtual scenarios for pedestrian detection,” CVPR, pp.137–144, IEEE, 2010.

[4] Y. Yamauchi, and H. Fujiyoshi, “Automatic Generation of Training Samples and a Learning Method Based on Advanced MILBoost for Human Detection,” Asian Conference on Pattern Recognition, pp. 603-607, 2011.

[5] NIPS 2005 Workshop - Inductive Transfer: 10 years later.
http://iitrl.acadiau.ca/itws05/.

[6] W. Dai, Q. Yang, G. rongXue, and Y. Yu, “Boosting for transfer learning,” In ICML, 2007.

[7] W. Li, M. Wang, and X. Wang, “Transferring a generic pedestrian detector towards specific scenes,” 2012 IEEE Conference on Computer Vision and Pattern Recognition, vol.0, pp.3274–3281, 2012.

[8] J. Pang, Q. Huang, S. Yan, S. Jiang, and L. Qin, “Transferring boosted detectors towards viewpoint and scene adaptiveness,” Trans. Img. Proc., vol.20, no.5, pp.1388–1400, May 2011.

[9] G. Li, L. Qin, Q. Huang, J. Pang, and S. Jiang, “Treat samples differently: Object tracking with semi-supervised online covboost,” Proceedings of the 2011 International Conference on Computer Vision, pp.627–634, ICCV ’11, IEEE Computer Society, Washington, DC, USA, 2011.

[10] M.T. Rosenstein, Z. Marx, L.P. Kaelbling, and T.G. Dietterich, “To transfer or not to transfer,” In NIPS05 Workshop, Inductive Transfer: 10 Years Later, 2005.

[11] F. Shen, H. Yu, K. Sakurai, and O. Hasegawa, “An incremental online semi-supervised active learning algorithm based on self-organizing incremental neural network,” Neural Comput. Appl., vol.20, no.7, pp.1061–1074, Oct. 2011.

[12] INRIA Person Dataset:
“http://pascal.inrialpes.fr/data/human/”.