Object Detection in Aerial Imagery Based on Enhanced Semi-Supervised Learning

Jian Yao and Zhongfei (Mark) Zhang
Computer Science Department
Binghamton University
PO Box 6000, Binghamton, NY 13902
{jyao,zzhang}@binghamton.edu

Abstract

Object detection in aerial imagery has been well studied in computer vision for years. However, given the complexity of large variations of the appearance of the object and the background in a typical aerial image, a robust and efficient detection is still considered as an open and challenging problem. In this paper, we present the Enhanced Semi-Supervised Learning (ESL) framework and apply this framework to revising an object detection methodology we have developed in a previous effort. Theoretic analysis and experimental evaluation using the UCI machine learning repository clearly indicate the superiority of the ESL framework. The performance evaluations of the revised object detection methodology against the original one clearly demonstrate the superiority of this approach.

1 Introduction

Object detection in aerial imagery has been well studied for years in computer vision [5, 9, 20]. Concerning with the object detection methods reported in the literature, objects may be either detected as a boundary delineation or as a bounding box extraction. The former [9, 11, 12, 15] is usually achieved by perceptual grouping while the latter [8, 10, 17] is typically accomplished by classification.

The classification based object detection problem is typically solved in two stages: *candidate generation* and *candidate classification* [19]. The majority of the classification models used in the detection proposed in the literature are based on the supervised learning, including boosting models [13], cascade models [10, 17], neural networks [7], Bayesian networks [20], generative models [15], and statistical models [14]. Typically, manually ground truthing is tedious and error-prone. Consequently, the semi-supervised learning (SSL) algorithms [1, 3, 6] may be used to relieve this since they only need a small set of labelled training samples. Typically, an SSL is achieved by iteratively applying supervised learning.

In the previous effort [19], we have developed an SSL theory in which we have presented a novel labelling strategy for unlabelled training samples to maximize the learning accuracy for the supervised classifier at each iteration. We have applied this theory to aerial imagery object detection problem and have developed a context based object detection methodology, called CONTEXT. However, this theory, together with other existing SSL algorithms, cannot guarantee that the accuracy increases when the number of iterations increases. In this paper, we present the *Enhanced Semi-Supervised Learning* (ESL) framework in which we prove in theory that an SSL algorithm under this framework is probabilistically guaranteed to have the accuracy increased when the number of iterations increases. An SSL algorithm under this framework is called an ESL algorithm and the SSL algorithm per se is called the original SSL algorithm. We have applied this framework to revising the CONTEXT to show the substantially improved learning efficiency in aerial imagery object detection.

The rest of the paper is organized as follows. In Section 2, we present the ESL framework. In Section 3, we report the experimental evaluations of two ESL algorithms using the UCI Machine Learning Repository [2]. In Section 4, we present the revised CONTEXT and report the evaluation performance. Finally, the conclusion is given in Section 5.

2 ESL Framework

In this section, we first identify a fundamental problem with the existing SSL algorithms in the literature. We then develop the ESL framework for the 2-class SSL algorithms. Finally, we extend the framework to the general K-class SSL algorithms. The whole framework is based on the following assumption: each sample is identically and independently generated from an unknown distribution (i.i.d.).

2.1 Problem with Existing SSL Algorithms

The input to an SSL algorithm includes a labelled training sample set L and an unlabelled training sample set U. A typical SSL method is achieved by iteratively labelling the unlabelled training samples, which are called the *tentative labels*, and subsequently training a supervised classifier using L, U, and the tentative labels.

The supervised classifier used in an SSL procedure is called the *base classifier*. Due to the existence of unlabelled training samples, there are two interpretations for an unlabelled training sample to be correctly classified in each iteration. The first is that the unlabelled sample has a classified label equal to its ground truth label. The second is that the unlabelled sample has a classified label equal to its current tentative label. The two interpretations are called, respectively, the *ground truth correct* and the *perceived correct* interpretations. The accuracies determined using the two interpretations are called, respectively, the *ground truth accuracy* and the *perceived accuracy*.

As stated earlier, existing SSL algorithms cannot guarantee that the ground truth accuracy of the base classifier at an iteration increases when the number of iterations increases. To show this observation, we run two representative SSL algorithms on four databases from the UCI Machine Learning Repository. The four databases are denoted as, respectively, PD, WD, LR, and DR. The four databases are explained in detail later. The first SSL algorithm is denoted as SEM [19], which uses the EM algorithm [4] to estimate class probabilities, the probabilities for each unlabelled sample to belong to each class. L, U, the tentative labels, and the class probabilities are used to learn the base classifier at each iteration. The second SSL algorithm is denoted as SSVM [1], which uses the SVM [16] as the base classifier and does not include probability in the learning. In the experiments, we randomly select 5% percent training samples as the labelled training samples and consider the remaining training samples as the unlabelled training samples. Three stop criteria are selected: #1—the perceived accuracy stops increasing; #2—the average value of the class probability difference between the current and the previous iteration for all the unlabelled training samples is less than 1%; #3—the percentage of the unlabelled training samples which change their labels from all the unlabelled training samples is less than 1%. For each algorithm and each database, 20 times of learning with different randomly selected training samples are used to generate 20 classifiers. Table 1 reports the number of classifiers which maintain the increased ground truth accuracy during the learning. Table 2 reports the average value of the unnecessary iterations taken during the learning. It is clear from Tables 1 and 2 that for all the algorithms on all the databases, the increase of the ground truth accuracy cannot be guaranteed. Besides, the learning can be stopped earlier without affecting the final ground truth accuracy.

Table 1. Accuracy increase test results

Stop Criterion	Algorithm	PD	WD	LR	DR
# 1	SEM	4	4	3	1
	SSVM	4	5	3	2
# 2	SEM	6	5	4	2
	SSVM	N/A	N/A	N/A	N/A
# 3	SEM	7	7	4	2
	SSVM	6	7	4	3

Each value represents the number of the classifiers (out of 20) which maintain the increased ground truth accuracies using the specified classifier and stop criterion.

Table 2. Learning efficiency results

Stop Criterion	Algorithm	PD	WD	LR	DR
# 1	SEM	1.4	1.9	5	3.9
	SSVM	1.8	2.4	5.9	4.8
# 2	SEM	1.3	1.5	3.7	3.1
	SSVM	N/A	N/A	N/A	N/A
# 3	SEM	1.2	1.4	3.9	3.4
	SSVM	1.3	2.0	4.8	4.2

Each value represents the average number of iterations which do not lead to the ground truth accuracy increase using the specified algorithm and stop criterion.

2.2 2-class ESL Framework

For a 2-class classification problem, assume that L includes a positive training sample set P and a negative training sample set N. Let the number of the samples in a set X be $|X|$. Assume that U is divided into U_P and U_N, which are, respectively, the ground truth positive training sample set in U and the ground truth negative training sample set in U. We first assume that $|U_N|$ and $|U_P|$ are known and will discuss the case when the assumption is relaxed later. Denote η^i_{pg} and η^i_{ng} as the ground truth accuracies at iteration i for positive training samples and negative training samples, respectively. Similarly, denote η^i_{pp} and η^i_{np} as the perceived accuracies at iteration i for positive training samples and negative training samples, respectively. Following the i.i.d. property of the training samples, it is not difficult to derive:

$$\eta^i_{pg} = \frac{|P| \times \eta^i_{pp} + |U_P| \times (\eta^{i-1}_{pg}\eta^i_{pp} + (1-\eta^{i-1}_{pg})(1-\eta^i_{np}))}{|P| + |U_P|} \quad (1)$$

$$\eta^i_{ng} = \frac{|N| \times \eta^i_{np} + |U_N| \times (\eta^{i-1}_{ng}\eta^i_{np} + (1-\eta^{i-1}_{ng})(1-\eta^i_{pp}))}{|N| + |U_N|} \quad (2)$$

For different applications, there may be different emphases on the accuracy of positive samples or the accuracy of negative samples. In order to make the learning adaptive to different applications, we define the overall ground truth ac-

curacy of the classifier at iteration i, which is denoted as η_g^i, as a linear combination of the two ground truth accuracies:

$$\eta_g^i = \alpha \times \eta_{pg}^i + (1-\alpha) \times \eta_{ng}^i \quad (3)$$

Where α is an application dependent parameter which specifies the relative emphasis on the accuracy of positive samples. Substituting (1) and (2) into (3), we finally have:

$$\begin{aligned}\eta_g^i = \alpha \times \frac{|P| \times \eta_{pp}^i}{|P|+|U_P|} + (1-\alpha) \times \frac{|N| \times \eta_{np}^i}{|N|+|U_N|} \\ + \alpha \times \frac{|U_P| \times (1-\eta_{np}^i + \eta_{pg}^{i-1}(\eta_{pp}^i + \eta_{np}^i - 1))}{|P|+|U_P|} \\ + (1-\alpha) \times \frac{|U_N| \times (1-\eta_{pp}^i + \eta_{ng}^{i-1}(\eta_{pp}^i + \eta_{np}^i - 1))}{|N|+|U_N|} \quad (4)\end{aligned}$$

In (4), the known parameters are $|P|$, $|N|$, $|U_P|$, $|U_N|$, and α; the unknown parameters, which can be reliably estimated using the method presented later, are η_{pg}^{i-1} and η_{ng}^{i-1}; the remaining parameters are η_{pp}^i and η_{np}^i. After the base classifier at iteration i is learned, we can consider these two parameters as known parameters. Then we can estimate η_g^i using (4) and compare it with η_g^{i-1}. If η_g^i is higher, we move to the next iteration. Otherwise, the learning stops.

Now the problem becomes how to estimate η_{pg}^0 and η_{ng}^0. We first randomly generate some sample sets from $U \bigcup L$ and determine η_{pg}^0 and η_{ng}^0 for each sample set. Based on the Sampling Theory and the Central Limit Theorem [18], when the number of sample sets, which is called the *sample size*, is sufficiently large (> 30), the distribution of the average η_{pg}^0 of all the groups is approximately normal with the mean equal to η_{pg}^0 estimated using $U \bigcup L$ and the standard deviation equal to the standard deviation estimated using $U \bigcup L$ divided by the square root of the sample size. Similar results can be derived for η_{ng}^0. Consequently, Algorithm 1 is presented to estimate η_{pg}^0 and η_{ng}^0 and Algorithm 2 is the 2-class ESL framework.

Algorithm 1 Initial Parameter Estimation

1.Train the base classifier using L.
2.Randomly select M samples from $U \bigcup L$ and ground truth them. These samples are called the *seed samples*.
3.Classify the seed samples using the base classifier learned in Step 1.
4.Divide the seed samples into 30 groups evenly. Determine the mean and the standard deviation of the positive accuracies and the negative accuracies from all the groups, and denote them as, respectively, (m_{pg},σ_{pg}) and (m_{ng},σ_{ng}).
5.For confidence level $1-\beta$ ($\beta \in [0,1]$), let η_{pg}^0 be $m_{pg} + z_{\beta/2} \times \frac{\sigma_{pg}}{\sqrt{30}}$ and let η_{ng}^0 be $m_{ng} + z_{\beta/2} \times \frac{\sigma_{ng}}{\sqrt{30}}$. z_x satisfies $G_{0,1}(-\infty < y < z_x) = x$, where $G_{0,1}$ is the cumulative distribution function of the standard Gaussian.

Algorithm 2 2-class ESL Framework

1. Train an initial classifier using L.
2. Use Algorithm 1 to estimate η_{pg}^0 and η_{ng}^0. Set $i = 1$.
3. Classify U using the trained classifier at Iteration $i-1$ and assign tentative labels to unlabelled samples.
4. Re-train the classifier using L, U, and the tentative labels of U. Determine η_{pp}^i and η_{np}^i.
5. Determine η_{pg}^i and η_{ng}^i using Equations (1) and (2). η_g^i is determined using Equation (3).
6. If $\eta_g^i > \eta_g^{i-1}$ then $i = i+1$ and goto step 3;
else output the classifier at Iteration $i-1$ and stop.

We will address the issue of how to determine an appropriate value for M later. The following theory is the theoretic foundation of Algorithm 2:

Theorem 1. *Given an arbitrary β, assuming that Algorithm 2 stops at iteration G, the probability of the event that the ground truth accuracies increase over iterations is at least* $(1-\beta/2)^2$*,i.e.,*$P(\eta_g^{\hat{G-1}} > ... > \hat{\eta_g^1} > \hat{\eta_g^0}) > (1-\beta/2)^2$.

Proof. First, we use the mathematical induction to prove that if $\eta_{pg}^0 > \hat{\eta_{pg}^0}$ and $\eta_{ng}^0 > \hat{\eta_{ng}^0}$, then we have $\eta_{pg}^i > \hat{\eta_{pg}^i}$, $\eta_{ng}^i > \hat{\eta_{ng}^i}$, and $\eta_g^i > \hat{\eta_g^i}$ for any $i < G$.
Initial step: $\eta_{pg}^0 > \hat{\eta_{pg}^0}$ and $\eta_{ng}^0 > \hat{\eta_{ng}^0}$ are correct from the assumption. Consequently, $\eta_g^0 > \hat{\eta_g^0}$ is correct from (3).
Induction step: Assume that $\eta_{pg}^{i-1} > \hat{\eta_{pg}^{i-1}}$, $\eta_{ng}^{i-1} > \hat{\eta_{ng}^{i-1}}$, and $\eta_g^{i-1} > \hat{\eta_g^{i-1}}$. From (1), we have:

$$\frac{\partial \eta_{pg}^i}{\partial \eta_{pg}^{i-1}} = \frac{|U_P|}{|P|+|U_P|} \times (\eta_{pp}^i + \eta_{np}^i - 1) \quad (5)$$

For a learned classifier, it must tentatively correctly classify at least half of the training samples, i.e., $\eta_{pp}^i + \eta_{np}^i > 1$. Therefore, $\frac{\partial \eta_{pg}^i}{\partial \eta_{pg}^{i-1}} > 0$. Since $\eta_{pg}^{i-1} > \hat{\eta_{pg}^{i-1}}$, we have:

$$\begin{aligned}\frac{|P| \times \eta_{pp}^i + |U_P| \times (\eta_{pg}^{i-1}\eta_{pp}^i + (1-\eta_{pg}^{i-1})(1-\eta_{np}^i))}{|P|+|U_P|} > \\ \frac{|P| \times \eta_{pp}^i + |U_P| \times (\hat{\eta_{pg}^{i-1}}\eta_{pp}^i + (1-\hat{\eta_{pg}^{i-1}})(1-\eta_{np}^i))}{|P|+|U_P|} \quad (6)\end{aligned}$$

which leads to $\eta_{pg}^i > \hat{\eta_{pg}^i}$. Similarly, we can derive $\eta_{ng}^i > \hat{\eta_{ng}^i}$ from (2). By (3), $\eta_g^i > \hat{\eta_g^i}$ is also correct.

Now we prove that for any $i < G$, if $\eta_{pg}^{i-1} > \hat{\eta_{pg}^{i-1}}$ and $\eta_{ng}^{i-1} > \hat{\eta_{ng}^{i-1}}$, then $\hat{\eta_g^i} > \hat{\eta_g^{i-1}}$. Since the learning procedure

does not stop at iteration i, we have $\eta_g^i > \eta_g^{i-1}$, i.e.,

$$\alpha \times \frac{|P| \times \eta_{pp}^i + |U_P| \times (\eta_{pg}^{i-1}\eta_{pp}^i + (1-\eta_{pg}^{i-1})(1-\eta_{np}^i))}{|P| + |U_P|} + (1-\alpha) \times \frac{|N| \times \eta_{np}^i + |U_N| \times (\eta_{ng}^{i-1}\eta_{np}^i + (1-\eta_{ng}^{i-1})(1-\eta_{pp}^i))}{|N| + |U_N|} > \alpha\eta_{pg}^{i-1} + (1-\alpha)\eta_{ng}^{i-1} \quad (7)$$

Move the left hand side of (7) to the right hand side and denote the resulting right hand side as $\phi(\eta_{ng}^{i-1}, \eta_{pg}^{i-1})$. Then we have:

$$\frac{\partial \phi}{\partial \eta_{pg}^{i-1}} = \alpha \times (1 - \frac{|U_P|}{|P| + |U_P|} \times (\eta_{np}^i + \eta_{pp}^i - 1)) \quad (8)$$

Since $\alpha > 0$, $0 < \frac{|U_P|}{|P|+|U_P|} < 1$, and $\eta_{np}^i + \eta_{pp}^i < 1 + 1 = 2$, we have $\frac{\partial \phi}{\partial \eta_{pg}^{i-1}} > 0$. Similarly, we also have $\frac{\partial \phi}{\partial \eta_{ng}^{i-1}} > 0$. Since $\eta_{pg}^{i-1} > \hat{\eta_{pg}^{i-1}}$ and $\eta_{ng}^{i-1} > \hat{\eta_{ng}^{i-1}}$, we have $\phi(\hat{\eta_{ng}^{i-1}}, \hat{\eta_{pg}^{i-1}}) < \phi(\eta_{ng}^{i-1}, \eta_{pg}^{i-1}) < 0$. Reorder the inequality $\phi(\hat{\eta_{ng}^{i-1}}, \hat{\eta_{pg}^{i-1}}) < 0$ and we have:

$$\hat{\eta_g^i} > \hat{\eta_g^{i-1}} \quad (9)$$

Combining the two results, it is clear that if $\eta_{pg}^0 > \hat{\eta_{pg}^0}$ and $\eta_{ng}^0 > \hat{\eta_{ng}^0}$, then $\hat{\eta_g^i} > \hat{\eta_g^{i-1}}$ is correct for any $i < G$. Consequently, we have:

$$P(\eta_g^{\hat{G-1}} > \eta_g^{\hat{G-2}} > ... > \hat{\eta_g^1} > \hat{\eta_g^0}) > P(\eta_{pg}^0 > \hat{\eta_{pg}^0}) \times P(\eta_{ng}^0 > \hat{\eta_{ng}^0}) = (1-\beta/2)^2 \quad (10)$$

□

In Algorithm 2, if we set η_{pg}^0 as $m_{pg} - z_{\beta/2} \times \frac{\sigma_{pg}}{\sqrt{30}}$ and set η_{ng}^0 as $m_{ng} - z_{\beta/2} \times \frac{\sigma_{ng}}{\sqrt{30}}$, we have $\eta_g^i < \hat{\eta_g^i}$ for any $i < G$. Consequently, we have the following corollary:

Corollary 1. *Given an arbitrary $1-\beta$, denote the η_g^i generated by selecting $\eta_{pg}^0 = m_{pg} + z_{\beta/2} \times \frac{\sigma_{pg}}{\sqrt{30}}$ and $\eta_{ng}^0 = m_{ng} + z_{\beta/2} \times \frac{\sigma_{ng}}{\sqrt{30}}$ as $\acute{\eta_g^i}$ and the η_g^i generated by selecting $\eta_{pg}^0 = m_{pg} - z_{\beta/2} \times \frac{\sigma_{pg}}{\sqrt{30}}$ and $\eta_{ng}^0 = m_{ng} - z_{\beta/2} \times \frac{\sigma_{ng}}{\sqrt{30}}$ as $\grave{\eta_g^i}$. Then we have: $P(\forall i, \acute{\eta_g^i} > \hat{\eta_g^i} > \grave{\eta_g^i}) > (1-\beta)^2$.*

It is clear that the above theory is based on the assumption that $|U_P|$ and $|U_N|$ are known. In case there is no such prior knowledge, we could also use the parameter estimation method to estimate them. Not only the seed samples, but also the labelled training samples and testing samples can be used to estimate them. Experimental results indicate that those samples are sufficient to reliably estimate $|U_N|$ and $|U_P|$.

2.3 K-class ESL Framework

For the K-class classification, let U_{ig} be the ground truth class i sample set in U; let $\eta_{ijg}^k(\eta_{ijp}^k)$ be the probability of a ground truth (tentative) class i sample to be classified as a ground truth (tentative) class j sample at iteration k. Assume the overall accuracy is a linear combination of the accuracies for each class, i.e, $\eta_g^k = \sum_j \alpha_j \times \eta_{jjg}^k$,where

$$\eta_{iig}^k = \frac{|P| \times \eta_{iip}^k + \sum_j U_{ig} \times \eta_{ijg}^{(k-1)} \times \eta_{jip}^k}{|P| + |U_{ig}|} \quad (11)$$

Similar to Algorithm 1 and Algorithm 2, an initial parameter estimation algorithm for η_{ijg}^0 for any i and j and the K-class ESL framework can be derived. Denote the independent η_{ijg}^0 as the *free parameters* and the number of the free parameters as F. Note that for the K-class problem $F \leq K \times (K-1)$. Due to the correlations between different η_{ijg}^0, the actual F is far less than the upper bound of F, i.e., $K \times (K-1)$. For example, in the 2-class problem, the positive accuracy and the negative accuracy can be combined into one parameter—overall accuracy if these two have little difference. The following method is used to determine F value when no prior knowledge is available:
1. Let F be a small value. Divide all the η_{ijg}^0 into F groups, where all the η_{ijg}^0 in one group are considered the same.
2. Estimate the F free parameters.
3. Use the estimated η_{ijg}^0 to estimate η_{ijg}^1. If those of η_{ijg}^1 which are in one group are not the same actually, increase F by 1, modify the group correspondingly, and go to Step 2; otherwise, stop the procedure and output the current F value as the final F.

Experimental results show that when each group contains $4F$ samples, i.e., M equals to $4F \times 30 = 120 \times F$, the estimated η_g^k is accurate, i.e., the difference between the upper bound and the lower bound of η_g^k is small. Similar to the proof of Theorem 1, we have the following theorem as the theoretic foundation for the K-class ESL framework:

Theorem 2. *Given an arbitrary $1-\beta$, assuming that the K-class ESL procedure stops at iteration G, the probability of the event that the ground truth accuracies increase over iterations is at least $(1-\beta/2)^F$,i.e.,$P(\eta_g^{\hat{G-1}} > \eta_g^{\hat{G-2}} > ... > \hat{\eta_g^1} > \hat{\eta_g^0}) > (1-\beta/2)^F$.*

3 Evaluation Using the Public Data

We use the UCI machine learning repository [2] to evaluate the ESL algorithms against the original SSL algorithms to demonstrate the strength and the superiority of the ESL framework. We select the Pima Indians Diabetes Database, the Wisconsin Diagnostic Database, the Letter Recognition Database, and the Optical Recognition of Handwritten Digits Database, which are denoted as PD, WD, LR, and DR,

Table 3. Training Accuracy Comparisons

Database	SEM	ESEM	FEM	SSVM	ESSVM	FSVM
PD	(76.3%,75.2%)	(79.5%,78.2%)	(81.4%,79.7%)	(76.3%,76.1%)	(76.7%,77.1%)	(76.7%,79.9%)
WD	(93.9%,92.3%)	(93.7%,94.2%)	(95.3%,93.1%)	(92.3%,92.3%)	(95.1%,93.7%)	(95.7%,92.7%)
LR	(83.2%,81.1%)	(85.7%,84.3%)	(86.3%,86.1%)	(83.0%,81.2%)	(83.2%,83.2%)	(85.7%,85.7%)
DR	(93.7%,93.4%)	(97.1%,96.8%)	(99.2%,93.1%)	(94.9%,93.4%)	(97.3%,97.1%)	(99.1%,95.3%)
AVE	(86.8%,85.6%)	(89.0%,88.4%)	(90.6%,88.0%)	(86.6%,85.8%)	(88.1%,87.8%)	(89.3%,88.4%)

Each value pair represents the average learning accuracy and test accuracy using the specified algorithm and database.

respectively, as the test data sets. All the databases contain only numeric attribute values and no missing attribute.

For those databases which only have training samples, we randomly select 60% training samples as the training samples for the corresponding classifiers and use the remaining samples as test samples. Besides, β_l percent training samples are randomly selected as the labelled training samples for the SSL classifiers. All the remaining training samples are considered as unlabelled training samples. Without an explicit notice, β_l is set to 5%. All the results are the average values of 20 times of learning based on different randomly selected labelled training samples. We select the SSVM [1] and SEM [19] as the original SSL algorithms; after applying the ESL framework to them, we call the corresponding ESL algorithms as ESSVM and ESEM, respectively, for the reference propose; to further compare the performance between the SSL classifiers and the corresponding supervised learning algorithms, we also use all the training samples to train the corresponding supervised classifier using SVM and EM, respectively, and refer to them as FSVM and FEM, respectively.

First, we compare the final accuracies of the six classifiers over the four databases. Table 3 reports the results. Each entry in the table contains two values. The first is the accuracy for training samples and the second is the accuracy for test samples. It is clear that in most cases, the ESL algorithms have higher accuracies than those of the correspondingly original SSL algorithms. In addition, the accuracies of the SSL algorithms are typically lower than those of the correspondingly supervised learning algorithms. For the learning on LR and DR, which are multi-class classifications, there are 77.5% times of learning which have strictly increased accuracies for ESEM and ESSVM, compared with only 11.3% times of learning which have strictly increased accuracy for SEM and SSVM. Those values for PD and WD are 96.3% and 21.3%. The fact that the ESL framework for multi-class classification contributes less to maintaining the increased accuracy than the ESL framework for 2-class classification is consistent with the theory we have developed in Section 2.

Second, we compare the number of iterations taken by the SSL algorithms. Table 4 documents this experiment. It is clear that in most cases, the ESL algorithms need much fewer iterations than the corresponding original SSL algorithms. The reason for this is that the framework we have developed imposes a strong constraint on the perceived accuracy. If the perceived accuracy does not meet the condition at a specific iteration, even if there is a perceived accuracy increase w.r.t. the perceived accuracy at the previous iteration, the learning stops. On the other hand, this is not the case for the correspondingly original SSL. As we have shown already, an increase of the perceived accuracy does not sufficiently lead to an increase of the ground truth accuracy. Consequently, this also explains the result of the previous experiment: why the accuracy of an ESL algorithm is typically higher than that of the correspondingly original SSL algorithm. The differences between the number of iterations taken by an ESL algorithm and that by the correspondingly original SSL algorithm for PD and WD databases are small while those for LR and DR databases are large. The reason may be that the number of samples in LR and DR databases are relatively large and it costs more iterations for the original SSL algorithms to converge.

Table 4. Learning efficiency comparison

Database	SEM	ESEM	SSVM	ESSVM
PD	3.1	1.7	3.2	1.4
WD	3.5	1.6	4.1	1.7
LR	7.9	2.9	8.5	2.6
DR	6.4	2.5	7.2	2.4
AVE	5.2	2.2	5.8	2.0

Each value represents the number of iterations taken during the learning using the specified algorithm and database.

4 Object Detection Based on the ESL Framework

In this section, we apply the ESL framework to revising the CONTEXT methodology we have developed earlier for aerial imagery object detection [19]. The basic idea of CONTEXT is to use SSL algorithms to achieve an effective and efficient detection through thoroughly exploiting the context information. The CONTEXT contains mainly the following steps:

1. An aerial image is first segmented and the background is then identified.
2. The disconnected background regions are generated.
3. An SSL classifier is used to classify the background regions which may surround an object.
4. Another SSL classifier is used to verify whether there exist the objects which are surrounded by the background regions that have passed the first SSL classifier.

The revised CONTEXT is called RCON, which is exactly the same as CONTEXT except that the two SSL classifiers in CONTEXT are replaced with two corresponding ESL classifiers here. For the first ESL classifier, since it is more important for the background regions which actually surround an object to be correctly classified, i.e., to have a high accuracy for positive samples, we select a high α value. For the second ESL classifier, we use the same penalty for missing an object and for incorrectly detecting a non-existing object. Consequently, we let α be 0.5.

In order to facilitate a fair comparison, we evaluate RCON by focusing on aircraft detection, as was reported in [19]. The evaluation data set, the parameter selection, and the ground truthing procedure of RCON are exactly the same as those in [19]. The detection effectiveness is measured in terms of the *detection rate*, which is the percentage of the number of correctly detected objects from the ground truth number of objects in the data set, and the *false alarm rate*, which is the percentage of the number of incorrectly detected objects from the number of detected objects in the data set. Table 5 reports the comparison. Clearly, RCON improves CONTEXT slightly in detection efficiency and substantially in learning efficiency.

Table 5. Performance comparison

Metric	RCON	CONTEXT
Detection rate	95.8%	94.7%
False alarm rate	6.5%	7.3%
Detection time (s)	0.27	0.27
Training time (h)	23	74

5 Conclusions

In this paper, we present the Enhanced Semi-Supervised Learning (ESL) framework and apply this framework to revising an object detection methodology we have developed in a previous effort. Theoretic analysis and experimental evaluation using the UCI machine learning repository clearly indicate the superiority of the ESL framework. The performance evaluations of the revised object detection methodology using the ESL algorithms against the original one clearly demonstrate the promise and the superiority of this approach.

References

[1] K. P. Bennett and A. Demiriz. Semi-supervised support vector machines. *Advances in Neural Information Processing Systems*, 12, 1999.

[2] C. L. Blake and C. J. Merz. UCI repository of machine learning databases, 1998.

[3] I. Cohen, F. G. Cozman, N. Sebe, M. C. Cirelo, and T. S. Huang. Semisupervised learning of classifiers: Theory, algorithms, and their application to human-computer interaction. *PAMI*, 26(12):1553–1567, 2004.

[4] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum-likelihood from incomplete data via the em algorithm. *J.Royal Statist. Soc.*, 39:1–38, 1977.

[5] A. Filippidis, L. C. Jain, and N. Martin. Fusion of intelligent agents for the detection of aircraft in sar images. *PAMI*, 22(4):378–383, 2000.

[6] S. A. Goldman and Y. Zhou. Enhancing supervised learning with unlabeled data. In *ICML*, 2000.

[7] B. Kamgar-Parsi, B. Kamgar-Parsi, A. K. Jain, and J. E. Dayhoff. Aircraft detection: A case study in using human similarity measure. *PAMI*, 23(12):1404–1414, 2001.

[8] Z. Kim and J. Malik. Fast vehicle detection with probabilistic feature grouping and its application to vehicle tracking. In *ICCV*, pages 524–531, 2003.

[9] J. Li, R. Nevatia, and S. Nornoha. User assisted modeling of buildings from aerial images. In *CVPR*, 1999.

[10] H. Schneiderman. Feature-centric evaluation for efficient cascaded object detection. In *CVPR*, pages 29–36, 2004.

[11] E. Sharon, A. Brandt, and R. Basri. Segmentation and boundary detection using multiscale intensity measurements. In *CVPR*, pages 469–476, 2001.

[12] J. Shi and J. Malik. Normalized cuts and image segmentation. *PAMI*, 22(8):888–905, 2000.

[13] A. Torralba, K. P. Murphy, and W. T. Freeman. Sharing features: efficient boosting procedures for multiclass object detection. In *CVPR*, pages 762–769, 2004.

[14] A. Torralba and P. Sinha. Statistical context priming for object detection. In *ICCV*, pages 763–770, 2001.

[15] Z. Tu, X. Chen, A. L. Yuille, and S.-C. Zhu. Image parsing: unifying segmentation, detection, and recognition. In *ICCV*, pages 18–25, 2003.

[16] V. Vapnik. *Statistical Learning Theory*. John Wiley and Sons, 1998.

[17] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *CVPR*, 2001.

[18] D. Wackerly, W. Mendenhall, and R. L. Scheaffer. *Mathematical Statistics with Applications*. 2002.

[19] J. Yao and Z. Zhang. Semi-supervised learning based object detection in aerial imagery. In *CVPR*, 2005.

[20] T. Zhao and R. Nevatia. Car detection in low resolution aerial image. In *ICCV*, pages 710–717, 2001.