symmetry

MDPI

Article

Hierarchical Open-Set Object Detection in Unseen Data

Yeong Hyeon Kim [1], Dong Kyun Shin [2], Minhaz Uddin Ahmed [1] and Phill Kyu Rhee [1,*]

[1] Intelligence Technology Lab, Inha University, 100 Inha-Ro, Nam Gu, Incheon 22212, Korea; ohpely@gmail.com (Y.H.K.); minhaz.ahmed@gmail.com (M.U.A.)

[2] KT R&D Center, 151 Taebong-ro, Seocho-gu, Seoul 06763, Korea; tlsgb_1@naver.com

* Correspondence: pkrhee@inha.ac.kr

Received: 22 September 2019; Accepted: 4 October 2019; Published: 11 October 2019

Abstract: In this paper, we propose an open-set object detection framework based on a dynamic hierarchical structure with incremental learning capabilities for unseen object classes. We were motivated by the observation that deep features extracted from visual objects show a strong hierarchical clustering property. The hierarchical feature model (HFM) was used to learn a new object class by using collaborative sampling (CS), and open-set-aware active semi-supervised learning (ASSL) algorithms. We divided object proposals into superclasses by using the agglomerative clustering algorithm. Data samples in each superclass node were classified into multiple augmented class nodes instead of directly associating with regular object classes. One or more augmented class nodes are related to a regular object class, and each augmented class has only one superclass. Object proposals from inexperienced data distribution are assigned to an augmented class node. Dynamic HFM nodes in the decision path are assembled to constitute an ensemble prediction, and the new augmented object is associated with a new regular object class. Our experimental results showed that the proposed method uses standard benchmark datasets such as PASCAL VOC, MS COCO, ILSVRC DET, and local datasets to perform better than state-of-the-art techniques.

Keywords: object detection; deep learning; convolutional neural network; active learning

1. Introduction

There have been many advances in object detection technology since the deep-learning breakthrough achieved by Krizhevsky et al. in 2012 [1]. However, collecting and labeling training samples are necessary and exhaustive tasks. Furthermore, all object classes to be detected should be decided in advance, but this is not achievable in many real-world applications. In such applications, object detection systems trained with limited object classes tend to be impractical because a new object class in an unseen data distribution cannot be properly handled and can result in false predictions [2–4]. Few researchers have focused on open-set object detection in unseen data distributions, even though there are strong requirements in many application fields. The unseen data distribution of a new object poses a difficult problem in real-world object detection systems. A larger dataset with extended classes and a more complex neural network can be adopted, but this requires a complicated process of network redesign or remodeling tasks. However, the need for unseen object class detection is very common in many real-world applications.

State-of-the-art detection methods such as OverFeat [5], Faster RCNN [6], Spatial Pyramid Pooling [7], the YOLO series [8–10], and RetinaNet [11] still cannot satisfy real-world requirements. Even though they employ high-dimensional deep-feature spaces, performance degradation is unavoidable in object detection, especially when it is due to the imperfect quality of training samples and the diversity of real-world image capturing qualities. In practice, completely reliable human-labeling requirements are not acceptable because the cost of a high-quality labeling method is very expensive. The tree structure approach is very effective for solving problems in a flat architecture and supporting open-set classification. The pros and cons of hierarchical architecture in classification have been extensively investigated in the past few decades [12]. In hierarchical architecture, the label information in higher level tree nodes generally captures more discriminable, relevant label concepts and can be inherited by lower level nodes that are more difficult to distinguish [13]. Most previous approaches have relied on handcrafted features and annotations for hierarchical classifier training, but they required much effort and time, and an error prone annotation process is used in current classification in diverse application environments.

Two related issues with regard to open-set object detection errors include misclassified data points and out-of-distribution data points. Most object detectors tend to fail due to noisy environments such as cluttered backgrounds, pose variances, and illumination changes. Furthermore, some input data samples do not belong to a closed set, (i.e., the in-distribution class of the training dataset), but in an open set (i.e., outlier data). Both misclassification and outlier data samples are not consistent with trained data samples in deep-feature space, and detection results based on maximum likelihood are unreliable and incorrect. Many open-set algorithms distinguish between in-distribution and outlier data classes relying on classification scheme thresholds [3,14,15]. They show that discrimination against unseen classes from regular representative classes effectively minimize open-space risk. However, the information about unseen classes, such as which feature space and object class the test samples would have, is not available in advance. Both training and testing in classifiers are processed in deep-feature space rather than in semantic object label space [12].

In this paper, we reduce ambiguities from the mixture within regular object classification by employing the concept of augmented object classes based on a physical feature in a tree structure, not relying on the semantic regular object class hierarchy. Augmented object class hierarchies in the deep-feature space are organized by considering the between-class deep feature correlations and the within class variation criteria. We propose a hierarchical open-set object detection framework with the learning capabilities of unseen data distribution for a new object class. The proposed method employs a dynamic hierarchical structure, each node of which keeps track of related data, features, and the deep model, and evolves itself in accordance with changing data distributions. It adopts outlier detection for open-set active semi-supervised learning. The dynamic hierarchical feature model (HFM) can increase its visual taxonomy in the flexible hierarchical structure based on the novel concept of the augmented object class. An augmented object class is a portion of the regular class, the mixture complexity of which is less complex than or equal to that of the regular object class. The proposed framework can adaptively learn both closed-set classes for performance improvement and open-set object classes using unlabeled data samples. Our method combines the incremental open-set aware active semi-supervised learning (ASSL) [16] and the dynamic hierarchical feature model (HFM) update algorithm for effectively grouping unseen objects together. In real-world scenarios, an object detection system should handle noisy and open-set data. Static world assumptions adopted by most detection methods [5–10] are no longer valid in practice. We tackled this problem by leveraging the discriminative capability of the dynamic HFM embedded by the outlier detection algorithm and the collaborative sample selection-based open-set ASSL. We propose efficient open-set object detection using a flexible hierarchical structure that provides informative and nonredundant sample selection and the open-set-aware ASSL algorithm.

2. Related Works

2.1. Multi-Object Detection

Advanced object detection techniques depend primarily on the availability of large, properly labeled datasets for lengthy training [17–19]. Researchers have employed high-dimensional deep feature spaces to reduce performance degradation in detecting an object due to the imperfect quality of training samples and unseen data distributions, compared with varied and changing real-world environments. Most object detection schemes [5–10] assume that labeled training data samples that are independent and identically distributed (IID), are available in a static environment. Such a static IID assumption is not valid in many real-world object detection applications such as pedestrian detection [20], visual surveillance [21], activity recognition [15,22,23], and pose estimation [24]. Junwei et al. analyzed and compared the recent progress in a variety of state-of-the-art object detection studies using benchmark datasets [25]. They also proposed a two-stage co-segmentation algorithm with the assumption that dependable human labeling is not valid. This is because the cost of a high-quality labeling process becomes too expensive and it is unacceptable in real-time applications where an even background is necessary to reduce a disturbed background [26]. The quality of new data points in unseen distribution creates a challenging problem in object detection.

2.2. Open-Set Recognition

The open-set recognition problem is that the test sample is associated with an unseen class during training [4]. Most classification algorithms are based on the closed-world assumption, whereby a test sample will belong to one of K classes used during training. In many real-world applications, however, a testing sample may come from a class from the unseen distribution. Open-set recognition has incomplete knowledge of the world at training time, but unknown classes can be submitted to the algorithm during testing [4]. The open-set recognition issue has seldom been presented until now due to its strong generalization requirement. Most of the early works were based on hand engineered features that are not suitable for incremental learning. One of the works that is closely related to our approach is the Hasan et al. [27] method, where a deep neural network is applied in order to select the features incorporated with semi-supervised learning in a hybrid approach. Felzenszwalb et al. [22] proposed a deformable part model for employing latent information, which forms invariance in local transformations, leading to better localization. Tang et al. [28] presented a similarity-based knowledge transfer model and investigated how knowledge about object similarities can be transferred to adapt an object classifier to an object detector.

Common machine learning research work is based on closed sets. In [4], the nature of open-set recognition was discussed and formalized as a constrained minimization problem, which is similar to our work. The authors applied their method to face verification whereas we apply our proposed method to object detection. Neural networks show very strong generalization capability, as the data distribution of training and testing are the same.

In the real world, object detection performance usually decreases since new unseen object classes cannot be properly dealt with. In this context, we propose an open-set object framework based on a dynamic hierarchical structure with incremental learning capabilities for unseen object classes.

Neural networks tend to make high confidence predictions, even for completely unrecognizable [29] or irrelevant inputs [30–32], but we often have very little control over testing the data distribution in real-world applications. The correct detection of out-of-distribution samples is important for object detection/classification tasks [33]. It is important to be aware of the uncertainty of new types of input data in terms of in- and out-of-distribution samples, or those at their boundary [34]. One general weakness of open-set recognition is selecting the right candidates through collaborative sampling strategies, which is not easy where uncertainty, diversity, and confidence criteria play a major role. Often depending on the training data size, labeling time also varies, which is very tedious work. Moreover, after a number of iterative incremental learning the final model reaches saturation, and then the classification performance does not vary with more training. On the positive side, our method performs much better compared to state-of-the-art object detector approaches. The ASSL combined approach can effectively select the dataset that reduces the training time and labeling effort.

2.3. Active Learning and Semi-Supervised Learning Combination

Active learning (AL) and semi-supervised learning (SSL) try to solve the same problem based on different theorems, and they have the common goal of achieving high classification accuracy with minimum human labeling (Settles, 2010; Zhu, 2008). AL selects the most informative samples that are beneficial to the process of classification training by leveraging known information in the test data in accordance to the oracle's decisions [35,36]. The sampling approach for uncertainty is adopted to pick samples nearest to the decision boundary [37,38]. The famous AL sampling strategy, query by committee (QBC), is an ensemble learning method that relies on the different hypotheses of a committee, whereas the most informative samples are considered those of the maximal disagreement between classifiers [36]. While AL allows for human intervention to some extent, SSL directly uses unlabeled data in the training process without any human labeling [39]. The AL and SSL techniques can be mixed to handle labeled and tentatively labeled samples for classification practices, while mixed techniques investigate fresh samples manually labeled with minimal effort. The ensemble methods of AL and SSL are categorized into the sequential combination, with SSL embedding into AL, and collaborative samplings. The sequential combination emphasizes the fact that the initial training set is important for SSL convergence to objective performance. Muslea et al. adopted this strategy by employing multiple views for both AL and SSL [35]. The AL and SSL ensemble method is based on several different architectures [36,39]. Wan et al. showed AL-based verification for low-confidence pseudo-labeled samples labeled by SSL [40]. The collaborative combination of AL and SSL using the confidence score from a boosting algorithm was applied to a spoken-language classification problem [41]. Several methods of AL and SSL collaboration were studied in [36,39].

3. System Overview

We present an open-set object detector framework that learns a new object class and retrains a regular one in unseen data distribution. The framework begins with a very small number of labeled data samples and incrementally learns by using unlabeled data samples in an open-set setting. The main components of the proposed framework, shown in Figure 1 are the dynamic HFM, the outlier-detection algorithm, the collaborative sampling (CS) algorithm [16], and incremental ASSL [14]. We used the initial CNN model, which was pretrained by using labeled data samples, i.e., PASCAL VOC 2007 and the 2012 trainval dataset [17].

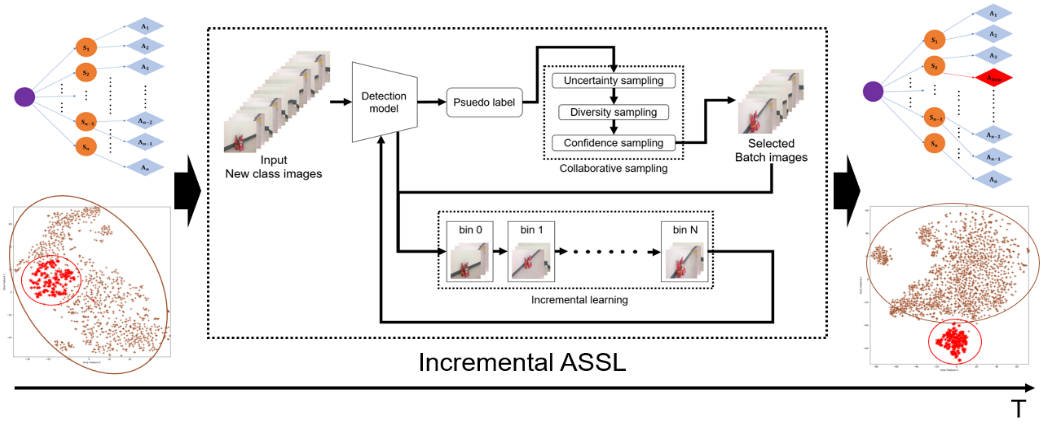

Figure 1. Application of incremental active semi-supervised learning (ASSL) on dynamic HFM. Data distributions (lower left, large brown and small red circles) before and (lower right) after application of incremental ASSL. Since the red circle was included in the brown circle, the detector could not distinguish between two object classes, i.e., the brown circle represents 'bottle', the red circle represents 'fire extinguisher'. The data distribution change is shown in the lower right where the brown and red circles were separated. Input dynamic HFM was trained by incremental ASSL and then updated to a new augmented class node.

The dynamic HFM builds the initial CNN model and improves the detection accuracy by improving model performance and increasing confident pseudo-labeled samples step by step. The hierarchical structure of labeled samples was modeled in terms of the super- and augmented classes using the agglomerative clustering algorithm. Data samples in each superclass node were related to multiple augmented class nodes. An augmented object class is a portion of a regular object, having distinctive data distribution from another portion/other portions of the regular object class. i.e., a regular object class consists of one or more augmented class nodes. Discrimination information is used in the dynamic HFM for open-set learning for an unseen object class when a new augmented object class is added. We considered object proposals generated from unseen data distribution. The object proposal sequence was partitioned into bins, which is dealt with at the same time, instead of one image sample at a time, as in many other approaches. This means that the learning/update for the dynamic HFM is not sensitive to noisy images, even in a new cluttered environment. The open-set ASSL performs collaborative sampling and analyzes object proposal distribution using the outlier-detection algorithm. Based on the results of the outlier detection, the ASSL retrains or creates an augmented class for an unseen regular class in the dynamic HFM. In the dynamic HFM, object proposals are filtered by the CS algorithm, combining criteria of uncertainty and diversity for AL and the criterion of the confidence for SSL. The collaboration between SSL and AL makes it possible to obtain pseudo-labeled training samples that are more confident and informative from unlabeled object proposals in the partition of the image. The outlier-detection algorithm is used to discriminate in- and out-of-distribution object proposals in the current deep-feature space.

In the dynamic HFM, learning in the ASSL is divided into the AL cycle using the confident dataset, and incremental SSL using unlabeled data divided into batches and bin sequences. In the AL cycle, a batch of object samples is split into several bins, and bin-based incremental SSL cycles are performed. If unseen object proposals are detected by the outlier-detection algorithm, the dynamic HFM is updated using the unseen data samples. The detected outliers are accumulated and clustered. If the volume of a cluster exceeds a threshold value, a corresponding augmented-object-class predictor model is built and added to the associated superclass node in the dynamic HFM. The open-set ASSL trains the CNN by using the confidently marked samples and continuously retrains the next profound model for the CNN by placing the chosen batch of samples using the present object detector until convergence. The suggested technique thus offers methods for both exploration and exploitation by combining informative and reliable (well-found) approaches to sampling in an open environment. The decision path for an object prediction ensemble is built by combining current object models and the object prediction model created for the new augmented class. Finally, the dynamic HFM is updated.

4. Dynamic Hierarchical Feature Model

We present a hierarchical deep-feature structure for open-set object detection that extends the HFM in [1] with the capability of incremental learning for regular objects and open-set learning for unseen objects. The dynamic HFM consists of two different levels: the superclass and augmented-class levels, as shown in Figure 2b. We employed the concept of an augmented class, which is defined as a distinctive portion of a regular class in data distribution. An augmented class shares common between-class characteristics with the superclass level, and closer within-class characteristics than the associated regular object class (see Figure 3).

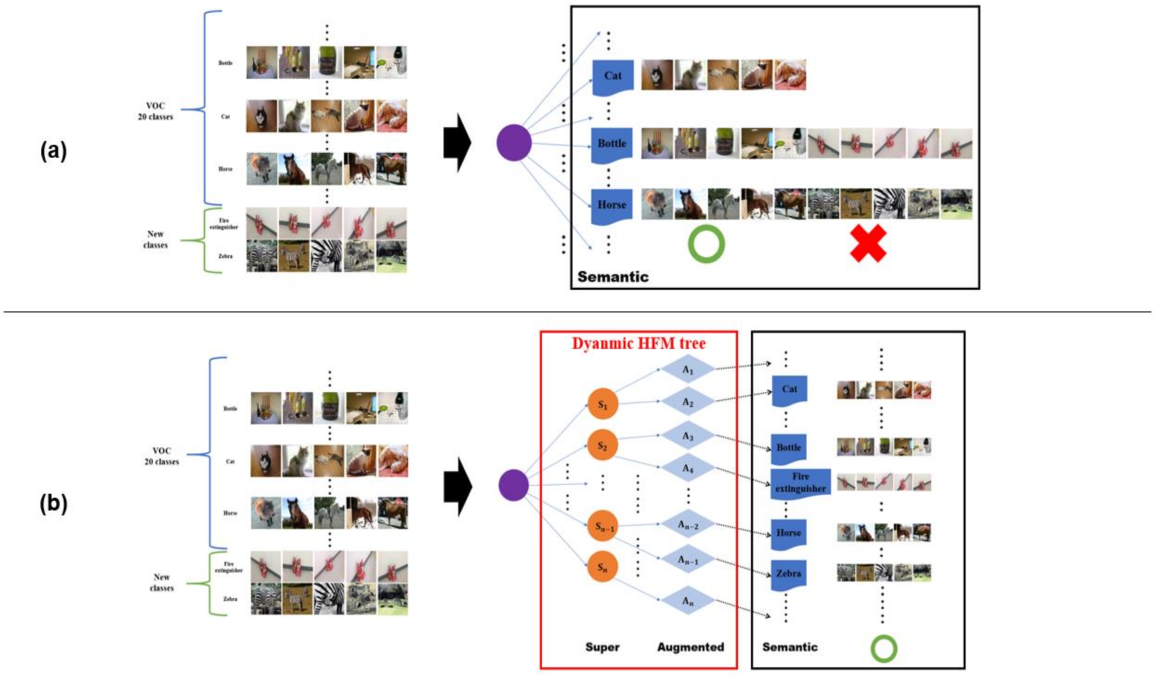

Figure 2. (**a**) General object detection flat model, which has good performance on trained class images. However, if the model meets untrained class data, there are many errors. (**b**) Our dynamic hierarchical feature model (HFM) tree framework, constructed by superclass and augmented class level. When data reach the augmented class node, the augmented class node is matched to a semantic class.

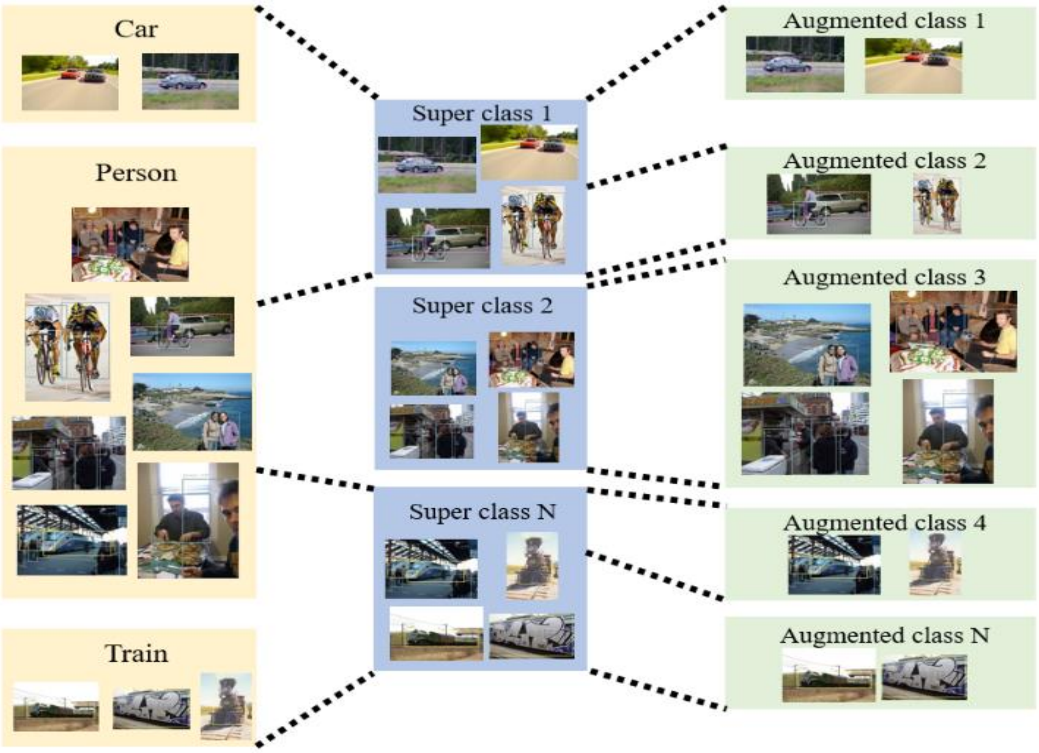

Figure 3. Illustration of proposed superclass and augmented class concept.

5. Outlier Detection

Unseen object classes cannot be correctly classified into a current augmented class and cannot produce regular object output. However, there exist superclass objects that share common between-class attributes with high probability, even though the new augmented-class prediction capability does not exist in the dynamic HFM. We adopted outlier detection methods [42] for our outlier detection. Object proposals were inputted to the dynamic HFM at time t. For object proposals $\{x_i\}$ extracted from the unlabeled dataset, the causes of object-detection errors are discriminated into a particular data point that is hard to classify and has not been defined yet in the dynamic HFM, i.e., an unseen data point (a particular data point belonging to a new class). The discrimination of the above types of errors is hard since it is deeply related to the semantic goal of a detector [28]. We investigated the distribution of object proposals and updated the dynamic HFM using the current prediction models. Following [42], we chose centroids $\left\{\mu_j\right\}_{j=1}^{J}$ and the cluster membership weights to optimize the within-cluster sum of squares (WCSS). Let $w_{ij} \in [0,\ 1]$ denote the cluster membership weight of deep-feature vector FV_i in cluster j. Let us define outlier compensation vector CV_i, which is an M-dimensional zero vector (i.e., 0 if CV_i is an inlier), or nonzero vector (when CV_i an outlier). Let us define $\boldsymbol{CV} = [CV_1 \cdots CV_N]$ and $\boldsymbol{\Lambda} = \left[\mu_1 \cdots \mu_J\right]$, and membership weight matrix $\boldsymbol{W} \in \mathbb{R}^{N\times J}$, where each element is represented by cluster membership weight $w_{ij} \in [0,\ 1]$. The soft K-means algorithm leverages the sparsity of the CV_is and defines the outlier-compensated version $(FV_i - CV_i)$ that replaces FV_i, and w_{ij}. Outlier aware soft K-means clustering is defined by

$$\min_{\boldsymbol{CV},\ \boldsymbol{\Lambda},\boldsymbol{W}} \sum\nolimits_{i=1}^{N} \sum\nolimits_{j=1}^{J} w_{ij}^{\tau}\left(\left\|FV_i - \mu_j - CV_i\right\|^2 +\ \lambda\left\|CV_i\right\|\right) \tag{1}$$

where τ is tuning parameter $\tau > 1$. The soft k-means algorithm solves based on the block coordinate descent (BCD) algorithm [42] that iteratively optimizes cost function, focusing on one variable at a time, while the other variables remain fixed. If CV_i was greater than threshold ρ, we defined CV_i as an outlier.

For a given bin of object proposals, the dynamic HFM algorithm updates the current node attributes. The dynamic HFM executes the tasks of open-set learning in each augmented class node. The dynamic HFM continuously updates its node attributes composed by the node dataset, node feature vectors, and node prediction model. Those are updated using the open-set ASSL algorithm discussed in the following section.

6. Open-Set-Aware Incremental ASSL

The proposed incremental open-set-aware ASSL combines the AL paradigm's uncertainty and diversity characteristics with the incremental SSL paradigm's confidence property. Taking into account AL's uncertainty criterion, most uncertain samples are considered as the most helpful training samples to be added, as they are anticipated to be wrongly classified with high probability by the present detection model. However, the uncertainty criterion may cause noisy or redundant samples to be selected. We adjusted a pool-based (batch or bin) AL structure coupled with incremental SSL philosophy based on AL and SSL's collaborative sampling method in terms of uncertainty, variety, and trust criteria that are expected to select more informative and low-redundancy training samples.

We used an AL batch cycle similar to [43], and added a bin-based cycle for incremental SSL. In the AL batch cycle, a training dataset was divided into well-defined labeled training samples D_{well}, and weakly and unlabeled training samples $D_{tentative}$. Open-set aware incremental ASSL dealt with them to increase the volume of D_{well} above $D_{tentative}$, and update the dynamic HFM. First, the original model learns from the prelabeled dataset used to build the CNN. Then, the batch of samples is chosen considering the distribution of the trained models and category balancing. The present detector assigns confidence scores to the pseudo-labeled samples. Depending on the confidence score, which are measured and ranked by the present detector, confident and well-defined samples are chosen from weak samples. A subset of pseudo-labeled samples is chosen using a collaborative sampling approach, whereby the present detector reassigns fresh labels or assigns elevated ratings to labels;

some ambiguous samples that have previously been identified are removed or relabeled by the oracle after filtering by the criteria of uncertainty and diversity.

Open-set aware incremental ASSL reduces training time by making a pool of D_{Δ} based on the uncertainty, diversity, and confidence criteria.

Candidate sample set $D_{diversity}$ that covers more revealing samples within rank ϑ

$$\begin{gathered} D_{diversity} = \{X \big| X \in D_{tentative}, 0 \leq f(X) \leq 1\} \\ \text{s.t. } Rank(x) < \eta \end{gathered} \tag{2}$$

where $Rank(x)$ denotes decreasing order of $f(x)$. Next, we initialized D_{Δ} with sample $X_{top} = argmax_{x \in D_{diversity}} f(x), X_{top} \in D_{diversity}$ using confidence criterion parameter γ. A sample from $D_{diversity}$ adds to D_{Δ}. $D_{diversity}$ becomes the most similar sample in D_{Δ} in terms of confidence score, i.e.,

$$X_{top} = argmax_{x \in D_{diversity}} \left\{ max_{x_i,\, x_j \in D_{\Delta}} d\left(x_i, x_j\right) \right\} \tag{3}$$

In this equation, we used Euclidian distance between two features to calculate $d\left(x_i, x_j\right)$.

When the cardinality of D_{Δ} becomes γ, the sample selection process is stopped, and the final sample set is D_{Δ}. We retrained the CNN using the pool of samples, and the process was repeated until a convergence criterion was satisfied. The entire process and parameters are summarized in Algorithm 1 and [44].

Algorithm 1. Open-set aware incremental active semi-supervised learning using outlier detection

```
Input: Confident labeled dataset $D_{well}$,
tentatively labeled dataset $D_{tentative}$, and dynamic HFM.
Output: Optimal dynamic HFM.
while $D_{tentative} \neq \varnothing$ do
    Train initial CNN model $f$ using $D_{well}$.
      while $f$ not convergence, do
        Select batch pool of candidate samples from $D_{tentative}$.
        Select $D_{\Delta}$ tentatively labeled samples
        filtered by $\eta$, $\vartheta$, $\gamma$ parameters using (2) and (3).
        and each selection criteria.
        Assign pseudo-label and score to each unlabeled $D_{\Delta}$.
        Sort pseudo-labeled tentative samples $D_{\Delta}$ in decreasing order.
        Divide $j$ bins sorted tentative samples in decreasing order.
        $i^{th}$ bin has samples in range of $(i-1)/\,|D_{\Delta}|$ to $i/\,|D_{\Delta}|$.
        Generate bin sequence $BSeq = [bin_i]_{i=0}^{j}$ by partitioning $D_{\Delta}$.
        while $i < j$ do
          $bin_i = BSeq[i]$, train $f^{(i+1)}$
          using $D_{train}^{(i)} \cup bin_i$ and calculate $Acc_{bin_i}^{(i)}$.
          If $Acc^{(i+1)} \geq Acc^{(i)}$, $bin^* = \underset{bin_i}{\text{argmax}} \left\{Acc^{(i+1)}\right\}$;
            $D_{train}^{(i+1)} = D_{train}^{(i)} \cup bin^*; f^{(i+1)} = f_{bin^*}^{(i+1)}$.
        Else if $Acc^{(i+1)} < Acc^{(i)}$ or outlier detected by (1),
            oracle labels incorrectly labeled data in $bin^*$ and return $f^{(i+1)} = f^{(i)}$.
          $i$++
          end
          Retrain $f$ using $D_{well} = D_{well} \cup D_{\Delta}$; $D_{tentative} = D_{tentative} - D_{\Delta}$.
      end
    Update dynamic HFM with $f$.
end
```

The computational complexity of Algorithm 1 depends on the number of bins and the augmented classes used during our experiment, which we discuss in the next section. In some cases where many unseen object classes were present or the number of bins increased as a result, the training time also increased but the performance of the proposed method improved. In our experiment, we commonly considered a smaller number of bins to optimize performance, that is, around ten.

7. Experiment

The main goal of our experiment was to identify the efficiency of our dynamic HFM tree model framework using ASSL [44]. To achieve this goal, we conducted several experiments on benchmark datasets, such as PASCAL VOC, MS COCO, and ILSVRC. We then compared the results with advanced detectors such as Faster RCNN. We used the evaluation metric in the VOC development kit. All implementations were on a single server with a single NVIDIA TITAN X and Tensorflow [45].

7.1. Dataset Overview

PASCAL VOC dataset. The PASCAL VOC 2007 dataset [17] has 20 classes with 9963 images (train/validation/test) containing 24,640 annotated objects. PASCAL VOC 2012 also has 20 classes and consists of 11,540 images (train/validation/test) containing 27,450 annotated objects. We used other unseen object datasets to add to the PASCAL VOC 2007 test dataset for evaluating our method. Our experimental settings for training and testing were as follows: we used the Darknet-19 model [9], where the base detector is YOLOv2, which is trained by the PASCAL VOC 2007 and 2012 trainval datasets, and the resolution of the input images was 416 × 416. Training parameters were the same as [27,37].

Local dataset. In our experiment, we used the fire-extinguisher class and hog class dataset for input and to train new classes to our model. This dataset [5–10] was trained with the existing PASCAL VOC dataset. The fire-extinguisher class had 110 images that included 100 training-data images and 10 testing-data images. The hog class had a total of 110 images (100 training images and 10 testing images).

MS COCO dataset. The MS COCO dataset has 80 object-detection classes. We used MS COCO 2017 as a training and validation dataset [18], which has 118,287 training images and 5000 validation images. For ASSL training and evaluation, we used unseen training and validation dataset classes of PASCAL VOC in MS COCO animal classes (bear, elephant, giraffe, zebra).

ILSVRC DET dataset. The ILSVRC DET dataset has 200 classes for object detection training. We used the ILSVRC DET 2017 training and validation dataset [19], which contains 456,567 training images, 20,121 validation images, and 40,152 testing images. For ASSL training and evaluation, we used unseen training and validation dataset classes of PASCAL VOC in the ILSVRC vehicle classes (golf cart, snowmobile, snowplow, unicycle, watercraft).

7.2. Results

Local dataset. In Figure 4a1, it can be seen that the fire-extinguisher class was neighboring the augmented class 9, and the hog class was similar to augmented classes 8, 11, 14, 17, and 22. Therefore, it was possible to define the fire-extinguisher class and hog class, which had similar distributions to the existing augmented class, as a new augmented class. When we applied dynamic HFM, we could see that applying ASSL (Figure 4a2–a4 and VOC 2007 test + local dataset graph in Figure 5) could increase the area and average precision of the precision–recall curve over time. In Figure 4a5, the fire-extinguisher class was well separated from the existing augmented class 9, and the hog class was also different from the existing augmented class.

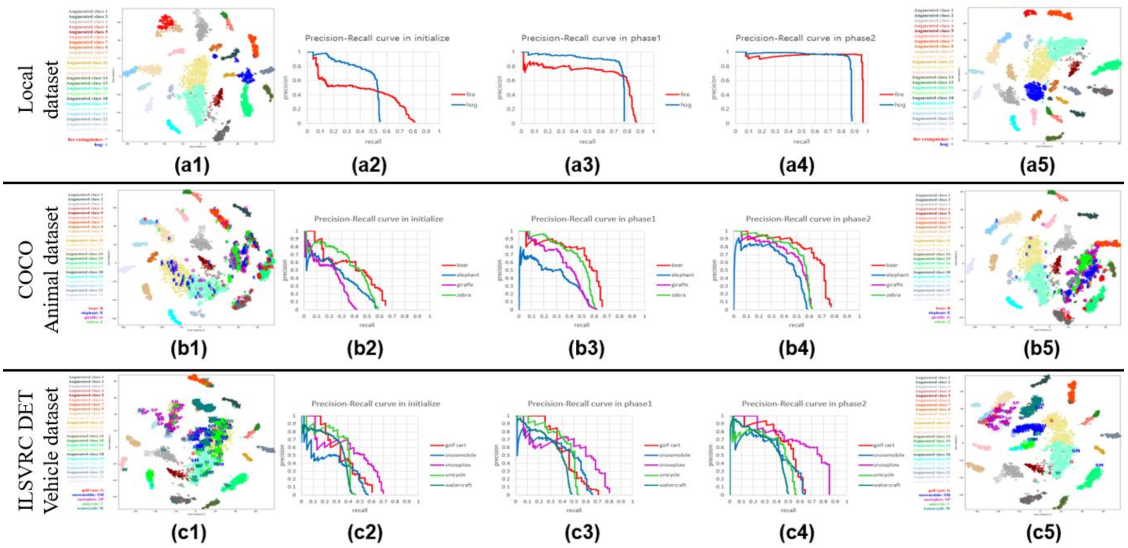

Figure 4. Result of applying dynamic HFM to other datasets. The visualization method was t-Stochastic Neighbor Embedding (t-SNE). (**a**) Series in local dataset, (**b**) series from COCO animal dataset, and (**c**) series from ILSVRC vehicle dataset. (**a1**) Input data distribution and (**a5**) output data distribution applied to our method. (**a2**–**a4**) Precision–recall curve in the training phase included in our ASSL method. (**a2**) Initialized model result, (**a3**) training phase 1 result, and (**a4**) training phase 2 result. COCO dataset and ILSVRC DET dataset results were similar to the local dataset.

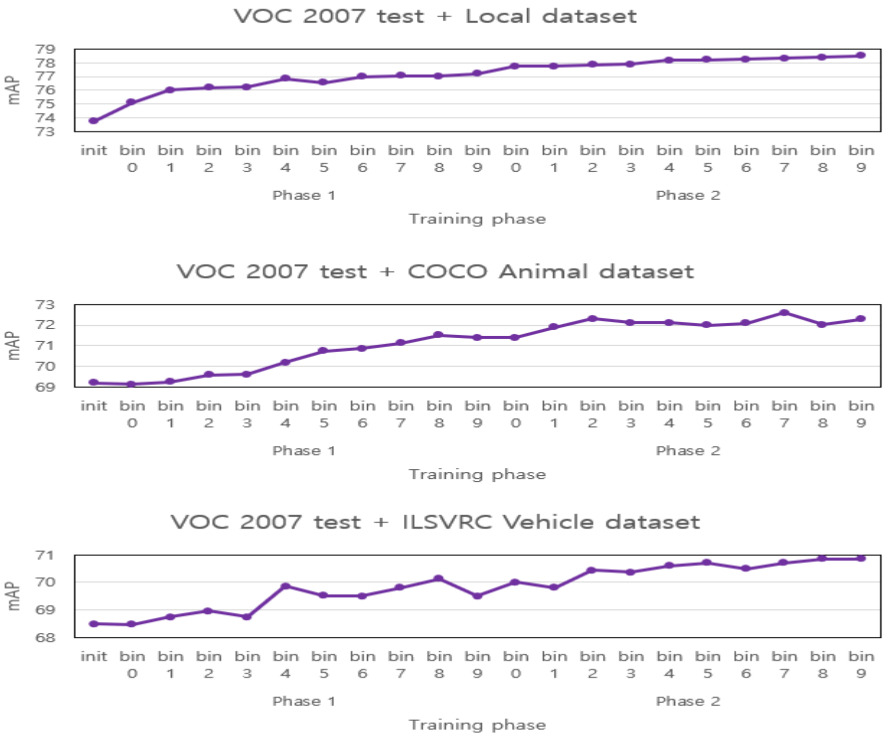

Figure 5. Results of increasing performance over time in incremental ASSL training of each dataset. Y-axis shows mean average precision (mAP), and x-axis shows number of bins and training phase.

MS COCO dataset. Figure 4b1 shows animal classes (bear, elephant, giraffe, zebra) neighboring augmented classes 2, 12, 14, 16, 17. It was possible to define the COCO animal classes that had similar distribution to existing augmented classes as a new augmented class. When we used our method, we could see that the time-dependent change with ASSL (Figure 4b2–b4 and VOC 2007 test + COCO animal dataset graph in Figure 5) could increase the area and average precision of the precision–recall curve over time. In Figure 4b5, it can be seen that the COCO animal classes were well independent from existing augmented classes such as augmented classes 12 and 16.

ILSVRC DET dataset. In Figure 4c1, we can see that the ILSVRC DET vehicle classes were very similar to augmented classes 8, 10, 12, 16, 21, and 23. It was possible to define vehicle classes that had similar distributions to existing augmented classes as a new augmented class. When we applied dynamic HFM, we could see that change in time sequence with ASSL (Figure 4c2–c4 and VOC 2007 test + ILSVRC vehicle dataset graph in Figure 5) could increase the area and average precision of the precision–recall curve over time. In Figure 4c5, we can see that the vehicle classes were well separated from existing augmented classes like augmented classes 12 and 16.

Comparison with other detection methods. We experimented with other detection methods. Table 1 shows that, in the case of other detection methods, we confirmed that performance was significantly reduced because other methods did not know the class for the environment in which the unseen object class appeared. However, dynamic HFM proved to be superior to other detection methods such as Faster RCNN, SSD300, and YOLOv2 for unseen object classes. The number of augmented classes in the dataset affects the performance of our method. As there are new augmented classes, the performance of our method improves compared to the state-of-the-art methods.

Table 1. Comparison of other detection methods in data-mixture environments.

w	Backbone	mAP			
		07 Test + Local Data (Fire Extinguisher)	07 Test + Local Data (Hog)	07 Test + COCO 2017 val Data (Animal)	07 Test + ILSVRC DET 2017 val Data (Vehicle)
Faster RCNN	VGG16	69.3	67.1	59.8	52.9
Faster RCNN	Resnet101	75.5	75.8	59.7	55.8
SSD 300	VGG16	73.3	73.6	64.1	54.2
YOLOv2	Darknet19	71.5	72.3	61.4	57.9
YOLOv2	Resnet50	67.3	67.6	57.0	54.1
YOLOv2	Resnet152	69.9	70.4	59.3	56.4
YOLOv2	Densenet201	71.9	72.5	61.4	58.1
Ours	Darknet19	77.9	77.5	72.3	70.8

8. Conclusions

In this paper, we proposed an open-set object detection framework called dynamic HFM, which provides incremental learning capabilities for unseen object classes. Data samples were clustered into superclasses according to deep-feature hierarchy attributes using the agglomerative clustering algorithm, and each superclass node was built to have multiple augmented class nodes instead of directly associating with regular object classes as in many other hierarchical approaches. The dynamic HFM discovers more informative deep-feature information with low mixture complexity by learning an augmented class instead of learning a regular class with high mixture complexity. The dynamic HFM was used to learn new object classes by imbedding outlier detection and a collaborative sampling method based on incremental ASSL algorithms. Dynamic HFM nodes in the decision path were assembled to constitute a prediction ensemble for associating to a regular object class. Finally, it adds an unseen object class as a new regular class. Our suggested model delivers greater efficiency with fewer mistakes, greater precision, and requires less human effort compared to other methods of pure object detection. These achievements encourage further improvement to our proposed model. Future research directions include finding ways to deal with huge datasets.

Author Contributions: Conceptualization, Y.H.K. and D.K.S.; writing—original-draft preparation, P.K.R.; writing—review and editing, M.U.A. and Y.H.K.; supervision, P.K.R.; project administration, Y.H.K.

Funding: This research received no external funding.

Acknowledgments: This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2016R1D1A1B03935440). The GPUs used in this research were generously donated by NVIDIA.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–6 December 2012; pp. 1097–1105.
2. Bendale, A.; Boult, T. Towards Open Set Deep Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 July 2015. [CrossRef]
3. Geng, C.; Chen, S. Hierarchical Dirichlet Process-based Open Set Recognition. *arXiv* **2018**, arXiv:1806.11258.
4. Scheirer, W.J.; Rocha, A.D.R. Toward Open Set Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *35*, 1757–1772.
5. Sermanet, P.; Eigen, D.; Zhang, X.; Mathieu, M.; Fergus, R.; LeCun, Y. OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks. *arXiv* **2013**, arXiv:1312.6229. [CrossRef]
6. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *Advances in Neural Information Processing Systems*; Nips: Grenada, Spain, 2015; pp. 1–10. [CrossRef]
7. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1904–1916. [CrossRef] [PubMed]
8. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2015. [CrossRef]
9. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017.
10. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
11. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollar, P. Focal Loss for Dense Object Detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2999–3007. [CrossRef]
12. Fan, J.; Zhao, T.; Kuang, Z.; Zheng, Y.; Zhang, J.; Yu, J.; Peng, J. HD-MTL: Hierarchical Deep Multi-Task Learning for Large-Scale Visual Recognition. *IEEE Trans. Image Process.* **2017**, *26*, 1923–1938. [CrossRef] [PubMed]
13. Wu, Q.; Tan, M.; Song, H.; Chen, J.; Ng, M.K. ML-FOREST: A multi-label tree ensemble method for multi-label classification. *IEEE Trans. Knowl. Data Eng.* **2016**, *28*, 2665–2680. [CrossRef]
14. Zhang, H.; Patel, V.M. Sparse representation-based open set recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *39*, 1690–1696. [CrossRef]
15. Rudd, E.M.; Jain, L.P.; Scheirer, W.J.; Boult, T.E. The Extreme Value Machine. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 762–768. [CrossRef]
16. Rhee, P.K.; Erdenee, E.; Kyun, S.D.; Ahmed, M.U.; Jin, S. Active and semi-supervised learning for object detection with imperfect data. *Cognit. Syst. Res.* **2017**, *45*, 109–123. [CrossRef]
17. Everingham, M.; Van Gool, L.; Williams, C.K.I.; Winn, J.; Zisserman, A. The pascal visual object classes (VOC) challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338. [CrossRef]
18. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Zitnick, C.L. Microsoft COCO: Common objects in context. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 8693 LNCS(PART 5). In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2014; pp. 740–755. [CrossRef]
19. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Li, F. ImageNet Large Scale Visual Recognition Challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211–252. [CrossRef]

20. Uçar, A.; Demir, Y.; Güzeliş, C. Object recognition and detection with deep learning for autonomous driving applications. *Simulation* **2017**, *93*, 759–769. [CrossRef]
21. Bengio, Y.; Courville, A.; Vincent, P. Representation Learning: A Review and New Perspectives. Pattern Analysis and Machine Intelligence. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 1798–1828. [CrossRef] [PubMed]
22. Felzenszwalb, P.F.; Girshick, R.B.; Mcallester, D.; Ramanan, D. Object Detection with Discriminatively Trained Part Based Models. *IEEE Trans. Pattern Anal. Mach. Intell.* **2009**, *32*, 1–20. [CrossRef] [PubMed]
23. Makantasis, K.; Doulamis, A.; Doulamis, N.; Psychas, K. Deep learning based human behavior recognition in industrial workflows. In Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 25–28 September 2016; pp. 1609–1613. [CrossRef]
24. Cao, Z.; Simon, T.; Wei, S.E.; Sheikh, Y. Realtime multi-person 2D pose estimation using part affinity fields. In Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2017), Honolulu, HI, USA, 21–26 January 2017; pp. 1302–1310. [CrossRef]
25. Han, J.; Zhang, D.; Cheng, G.; Liu, N.; Xu, D. Advanced Deep-Learning Techniques for Salient and Category-Specific Object Detection: A Survey. *IEEE Signal Process. Mag.* **2018**. [CrossRef]
26. Han, J.; Quan, R.; Zhang, D.; Nie, F. Robust Object Co-Segmentation Using Background Prior. *IEEE Trans. Image Process.* **2018**, *27*, 1639–1651. [CrossRef]
27. Hasan, M.; Roy-Chowdhury, A.K. A Continuous Learning Framework for Activity Recognition Using Deep Hybrid Feature Models. *IEEE Trans. Multimed.* **2015**, *17*, 1909–1922. [CrossRef]
28. Tang, Y.; Wang, J.; Gao, B.; Dellandrea, E.; Gaizauskas, R.; Chen, L. Large Scale Semi-Supervised Object Detection Using Visual and Semantic Knowledge Transfer. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 2119–2128. [CrossRef]
29. Nguyen, A.; Yosinski, J.; Clune, J. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 427–436. [CrossRef]
30. Moosavi-Dezfooli, S.M.; Fawzi, A.; Fawzi, O.; Frossard, P. Universal adversarial perturbations. In Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2017), Honolulu, HI, USA, 21–26January 2017; pp. 86–94. [CrossRef]
31. Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; Fergus, R. Intriguing properties of neural networks. *arXiv* **2013**, arXiv:1312.6199.
32. Yang, J.; Guan, Y.; Dong, X. Lymphocyte-style word representations. In Proceedings of the 2014 IEEE International Conference on Information and Automation (ICIA 2014), Hailar, Chian, 28–30 July 2014; pp. 920–925. [CrossRef]
33. Ji, S.; Yang, M.; Yu, K. 3D convolutional neural networks for human action recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 221–231. [CrossRef]
34. Mahalakshmi, R.; Praba, V.L. ENHANCING THE LABELLING TECHNIQUE OF SUFFIX TREE CLUSTERING ALGORITHM. *Int. J. Data Min. Knowl. Manag. Process* **2014**, *4*, 41.
35. Muslea, I.; Minton, S.N.; Knoblock, C.A. Active learning with strong and weak views: A case study on wrapper induction. In Proceedings of the IJCAI International Joint Conference on Artificial Intelligence, Acapulco, Mexico, 9–15 August 2003; pp. 415–420.
36. Settles, B. Active Learning Literature Survey. *Mach. Learn.* **2010**, *15*, 201–221.
37. Gordon, J.; Hernández-Lobato, J.M. Bayesian Semisupervised Learning with Deep Generative Models. *arXiv* **2017**, arXiv:1706.09751.
38. Zhang, K.; Zhang, Z.; Li, Z.; Member, S.; Qiao, Y.; Member, S. Joint Face Detection and Alignment using Multitask Cascaded Convolutional Networks. *IEEE Signal Process. Lett.* **2016**, *23*, 1499–1503. [CrossRef]
39. Zhu, X. Semi-Supervised Learning Literature Survey Contents. *Sciences* **2008**, *10*, 10.
40. Wan, L.; Tang, K.; Li, M.; Zhong, Y.; Qin, A.K.; Lunjun, W.; Qin, A.K. Collaborative Active and Semisupervised Learning for Hyperspectral Remote Sensing Image Classification. Geoscience and Remote Sensing. *IEEE Trans. Geosci. Remote Sens.* **2015**, *53*, 2384–2396. [CrossRef]
41. Sorokin, A.; Forsyth, D. Utility data annotation with Amazon Mechanical Turk. In Proceedings of the 2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPR Workshops), Anchorage, AK, USA, 23–28 June 2008; pp. 1–8. [CrossRef]

42. Forero, P.A.; Kekatos, V.; Giannakis, G.B. Robust clustering using outlier-sparsity regularization. *IEEE Trans. Signal Process.* **2012**, *60*, 4163–4177. [CrossRef]
43. Settles, B.; Craven, M. An analysis of active learning strategies for sequence labeling tasks. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP '08), Honolulu, Hawaii, 25–27 October 2008; p. 1070. [CrossRef]
44. Shin, D.K.; Ahmed, M.U.; Rhee, P.K. Incremental Deep Learning for Robust Object Detection in Unknown Cluttered Environments. *IEEE Access* **2018**, *6*, 2169–3536. [CrossRef]
45. Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Zheng, X. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. *arXiv* **2016**, arXiv:1603.04467.