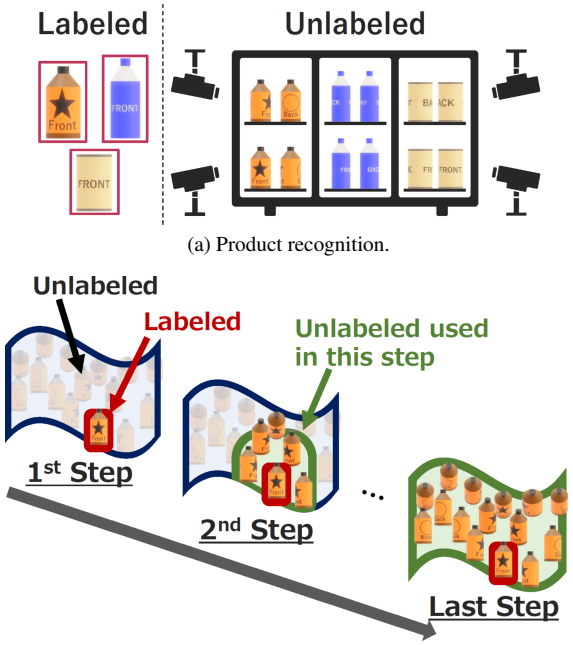
Appearance-Based Curriculum for Semi-Supervised Learning with Multi-Angle Unlabeled Data

Yuki Tanaka[1], Shuhei M. Yoshida[1], Takashi Shibata[1], Makoto Terao[1],
Takayuki Okatani[2,3], Masashi Sugiyama[2,4]

[1]NEC Corporation, [2]RIKEN AIP, [3]Tohoku University, [4]The University of Tokyo

Abstract

We propose an appearance-based curriculum (ABC) for a semi-supervised learning scenario where labeled images taken from limited angles and unlabeled ones taken from various angles are available for training. A common approach to semi-supervised learning relies on pseudo-labeling and data augmentation, but it struggles with large visual variations that cannot be covered by data augmentation. To solve this problem, ABC incrementally expands the pool of unlabeled images fed to a base semi-supervised learner so that newly added data are the ones most similar to those already in the pool. This way, the learner can assign pseudo-labels to the new data with high accuracy, keeping the quality of pseudo-labels higher than that when all the unlabeled data are processed at once, as customarily done in existing semi-supervised learning methods. We conducted extensive experiments and confirmed that our method outperforms the state-of-the-art semi-supervised learning methods in our scenario.



(a) Product recognition.

(b) Schematic figure of appearance-based curriculum.

Figure 1. Overview of the problem setting and our method. (a) Product image recognition as an example of our problem settings. The labeled images in the catalogs are taken from the front, and unlabeled images from security cameras in stores are taken from various angles. (b) Our method. Unlike conventional semi-supervised methods that process all the unlabeled data at once, unlabeled data are fed into the training step by step in the order of similarity.

1. Introduction

We study a scenario where labeled training images taken from limited angles and unlabeled ones taken from various angles are available for training. Such settings are significant in real-world applications. For example, consider product recognition (Fig. 1 (a)). One can obtain annotated images from a digital catalog, typically one or two images per product taken from the front. On the other hand, unlabeled images can easily be collected in shops or stores using security cameras from unconstrained viewpoints. If such data can be used to train a product recognition model, annotation costs are greatly reduced.

Such a combination of labeled and unlabeled data is naturally handled using semi-supervised learning [7, 37]. It is a subfield of machine learning much older than deep learning, but in the era of big data, its importance has become greater than ever because nowadays datasets that are too large to annotate fully are widely available. In the aforementioned example of product recognition, a large retail chain might be able to collect images from stores located across the country, and the dataset size could easily exceed the limit of the annotation budget. Motivated by such problems, researchers have developed numerous semi-supervised learn-

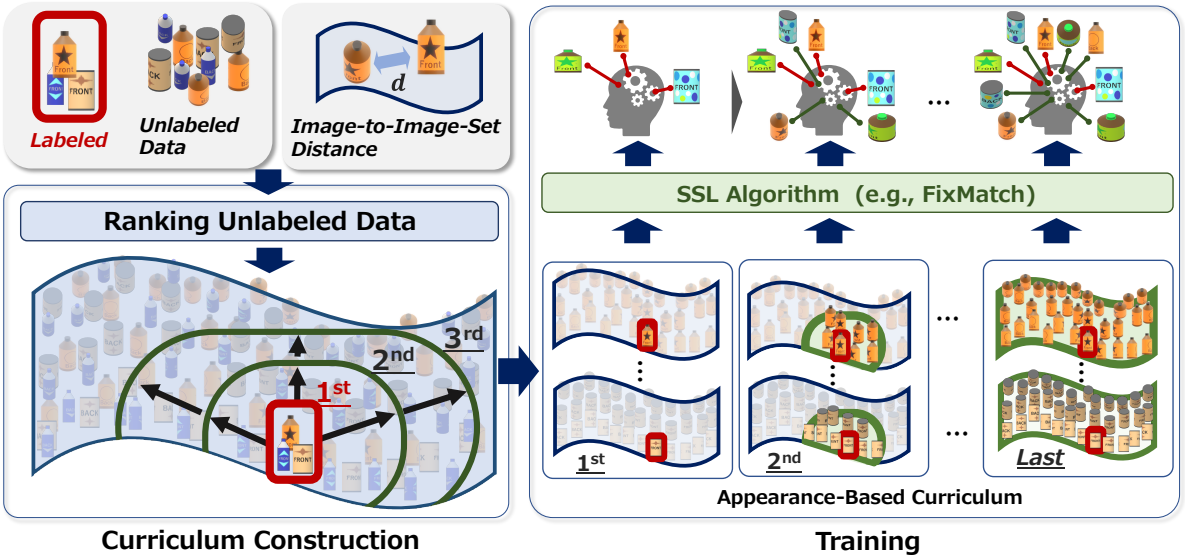


Figure 2. Overview of ABC learning. In the first phase, it constructs a curriculum using the ranking algorithm, which determines the schedule along which the unlabeled data are fed to a semi-supervised learning algorithm. The pool of unlabeled images used in the training is gradually expanded by incrementally adding new images that are most similar to those already in the pool. Then, in the second phase, the semi-supervised learning algorithm is executed step by step following the constructed curriculum.

ing algorithms for deep learning [32, 34].

Recent approaches to deep semi-supervised learning have combined pseudo-labeling strategies with data augmentation. For example, FixMatch [27], one of the state-of-the-art semi-supervised learning algorithms, is designed by combining data augmentation-based consistency regularization and a pseudo-labeling strategy. Such approaches implicitly assume that most of the unlabeled data can be covered by data augmentation.

Although the assumption is helpful for studying semi-supervised learning algorithms, it is too idealized for our purpose. In reality, the labeled data are so sparse in the support of the data distribution that data augmentation is not enough to cover the distribution. Indeed, our scenario is hard for conventional algorithms because widely available data augmentation cannot transform a photo taken from the front into, *e.g.*, the one taken from the back. Therefore, we need a novel approach that does not rely on such an assumption.

In this paper, we propose Appearance-Based Curriculum (ABC) learning to tackle this problem. The key insight behind ABC learning is that even the conventional semi-supervised methods can handle unlabeled images with shooting angles close enough to those of labeled ones because they have similar appearances. Therefore, by gradually expanding the camera angle range of unlabeled data consumed by the model in training, pseudo-labels can be assigned with high accuracy, even to images with distant angles (Fig. 1 (b)). A downside of this method is that it requires the camera angle information. To circumvent this limitation, we also propose two surrogate measures of camera angle similarity, local-descriptor-based and global-feature-based, that can be used when angle information is unavailable. Extensive experiments we conducted using the public and our privately collected datasets demonstrate that our method outperforms the state-of-the-art semi-supervised learning methods in our scenario.

2. ABC learning

In this section, we introduce ABC learning for semi-supervised scenarios in which unlabeled images taken from various angles are available. The core hypothesis is that a model trained on labeled data can generate accurate pseudo-labels of unlabeled instances "similar" to the labeled ones. In our scenario, we adopt resemblance in the appearance and the camera angles as the similarity. To substantiate this idea, we propose to expand gradually the pool of unlabeled data used in semi-supervised learning so that newly added data are the ones most similar to the labeled data or those already in the pool. This way, the quality of pseudo-labels should be higher than that when we use all the unlabeled data simultaneously.

Our framework consists of two phases, as shown in Fig. 2. The first is to construct a curriculum by a ranking algorithm that determines the order in which unlabeled data are fed to a given semi-supervised learning algorithm (Sec. 2.1). The second is the training phase to execute semi-supervised learning following the schedule set by the first phase (Sec. 2.2). This is a generic framework that relies on

Algorithm 1 Ranking algorithm of unlabeled data

Input: Labeled data $D_L = \{(x_i, y_i)\}_{i=1}^{N_L}$, unlabeled data $D_U = \{x_i\}_{i=1}^{N_U}$, image-to-image-set distance dist, the number of object categories K
Output: Sorted list of unlabeled data W

$U \leftarrow D_U$
$V_c \leftarrow \{(x, y) \in D_L \mid y = c\}$ for each class c
$W \leftarrow [\,]$
repeat
  for $c = 1, \ldots, K$ **do**
    $x \leftarrow \mathrm{argmin}_{x \in U} \mathrm{dist}(x, V_c)$
    $U \leftarrow U \setminus \{x\}$
    $V_c \leftarrow V_c \cup \{x\}$
    $W \leftarrow W + [x]$
  end for
until U is empty
return W

the similarity between an image and a set of images, or the distance function, which the ranking algorithm utilizes in sorting unlabeled images from easy to hard. The design of the distance function is the key to the success of this framework. In Sec. 2.3, leveraging the current scenario, we propose three appearance-based distance functions between object images that quantify visual (dis)similarity.

2.1. Constructing a curriculum

Algorithm 1 shows the ranking algorithm that determines how to expand the pool of unlabeled images given to a semi-supervised learning algorithm. The algorithm first initializes K bags of images with the labeled ones, where K is the number of object categories. Then, it expands the bag by adding the unlabeled image closest to the bag by using the given distance function, dist, that quantifies the closeness between an image and a set of images. This process is repeated until all the unlabeled images are added to the bags. The algorithm returns the order in which the unlabeled data are picked in this iteration.

This process is similar to how the Dijkstra-Jarník-Prim (DJP) algorithm [9, 14, 24] selects a new node in finding a minimum spanning tree in a weighted undirected graph. However, our method is different from the DJP algorithm in two ways. First, our method has several separate pools for every category, which are initially filled with labeled instances. This is to balance the distribution of unlabeled data over categories, especially in the earlier stages. The second difference resides in the distance function, dist, that quantifies the (dis)similarity between an unused instance and an existing pool of used data. In the DJP algorithm, dist, the node-to-set distance, is defined as the minimum of node-to-node distances (or weights, in the context of graph algorithms), but we choose to use a function that does not conform to this convention, as discussed in Sec. 2.3.

2.2. Training phase

Once the unlabeled data are sorted, we repeatedly run the base algorithm following the determined ranking. Specifically, let SSL be the base semi-supervised learning algorithm, n be the maximum number of steps, and W be the ordered list of unlabeled data. Suppose that SSL has a stopping criterion so that once it is satisfied, the execution is suspended. Then, we can write the update rule of the model parameters Θ and the checkpoint C as

$$\Theta, C \leftarrow \mathrm{SSL}(D_L, W[:iN_U/n], \Theta, C), \tag{1}$$

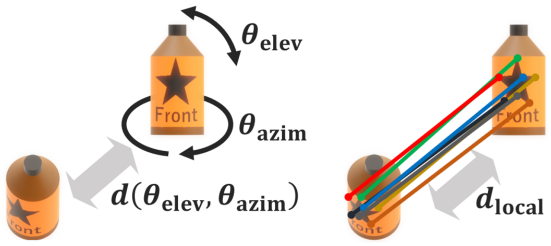
where D_L is the labeled data, N_U is the number of unlabeled images, and $W[:iN_U/n]$ represents the first $\frac{iN_U}{n}$ entries in W. The checkpoint C holds SSL's internal state so that the execution can be resumed with additional data from the point of suspension. We can use an arbitrary semi-supervised method for SSL if we equip it with some stopping criterion. The initial value of Θ can be a random value or that of some pretrained models, including the one fine-tuned on the labeled data.

In this study, we used FixMatch [27] as SSL in Eq. (1) because it is one of the state-of-the-art deep semi-supervised methods and yet is simple and flexible enough to accept add-ons like ABC. FixMatch assigns a pseudo-label to an unlabeled image if the confidence of the model's prediction is higher than a preset threshold, so the number of pseudo-labeled images varies along the course of training. This variation is used to determine when to proceed to the next step. Specifically, we employ the Feed-on-Plateau (FoP) criterion, which suspends the algorithm when the proportion of pseudo-labeled images in the unlabeled set has stayed within a preset width w_{th} for a patience period t_{p}. We allow SSL to be completed before consuming all the unlabeled images, which can be the case with adaptive stopping criteria like this one, depending on the progress of training. We set n as the number of outer repetitions in Algorithm 1, which is approximately N_U/K.

2.3. Distance function

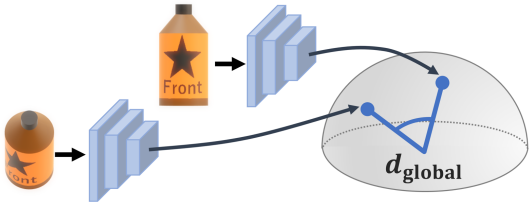
So far, we have discussed the general framework that can potentially be applied to various scenarios by designing an appropriate distance function dist. In this section, we focus on the situation in which labeled images taken from limited angles and unlabeled ones taken from various angles are available for semi-supervised learning.

The discussion can be simplified by utilizing dist, the distance between an instance and a pool of instances, which is defined with a distance function d that quantifies the



(a) d_{angle}

(b) d_{local}



(c) d_{global}

Figure 3. Schematic images of the three distance functions. (a) d_{angle} measures the difference in the camera angles between the two images. (b) d_{local} utilizes key-point matching based on the local descriptors like AKAZE [2] to quantify the similarity. (c) d_{global} is calculated using the cosine similarity between the two feature vectors extracted by a learned feature extractor.

(dis)similarity between two instances. We specifically use as dist the following form:

$$\text{dist}(u, V) = \frac{1}{|V|} \sum_{v \in V} d(u, v). \tag{2}$$

One may instead utilize the following form:

$$\text{dist}(u, V) = \min_{v \in V} d(u, v). \tag{3}$$

Intuitively, the latter selects a point closest to the edge of the set, while the former prefers the one closest to the center of the set. We choose the form of Eq. (2) because it is more stable and robust against outliers. The following paragraphs discuss the design of the instance-wise distance function d.

Under the current scenario, we hypothesize that two images of an object taken from similar viewpoints are similar and that if one in such a pair is in a pool of labeled data, the prediction on the other tends to be accurate. In this case, an appropriate distance function can measure the similarity of viewpoints. Because a viewpoint is characterized by the spherical coordinates, i.e., the elevation angle and the azimuthal angle, a straightforward way to quantify the similarity of two viewpoints is to use the following **angle-based** distance function (Fig. 3 (a)):

$$d(\Delta\theta_{\text{azim}}, \Delta\theta_{\text{elev}}) = \sqrt{(\Delta\theta_{\text{azim}})^2 + (\Delta\theta_{\text{elev}})^2}, \tag{4}$$

where $\Delta\theta_{\text{elev}}$ is the difference of the elevation angles and $\Delta\theta_{\text{azim}}$ is that of the azimuthal angles. This function can be used in our framework when labeled and unlabeled data have viewpoint information.

Because collecting information of camera angles is unrealistic in practice, we also propose using two distance functions, d_{local} and d_{global}, that require no additional annotation besides a small number of ground-truth labels as in regular semi-supervised learning. Instead of relying on extra manual efforts to annotate images, they use local and global appearance features that are either hand-crafted or learning-based.

The **local-descriptor-based** distance d_{local} utilizes off-the-shelf hand-crafted local descriptors, such as a scale-invariant feature transform (SIFT) [20], speeded up robust features (SURF) [3], KAZE [1], accelerated KAZE (AKAZE) [2], and oriented FAST and rotated BRIEF (ORB) [26] (Fig. 3 (b)). They were at the center of image registration, object detection, and tracking until learning-based approaches prevailed. They are still valuable for applications where collecting training data is infeasible. We utilize them as d_{local} because they are good at finding images of an object taken from similar angles without using any training data.

Specifically, we use AKAZE [2] to extract keypoints and local features and brute-force matching to match them across two images. Then, the distance function d_{local} is calculated as the Hamming distance between local descriptors for the compared images averaged over matched keypoints. See Appendix A for an OpenCV implementation of d_{local}.

The **global-feature-based** distance d_{global}, on the other hand, uses a pretrained model to extract global image features (Fig. 3 (c)). Specifically, we first train a model on the labeled data of the current task and then use its feature extractor f_Θ to convert an image into a fixed-length feature vector. The distance function d_{global} between two image instances u and v is calculated as one minus the cosine similarity between the extracted feature vectors:

$$d_{\text{global}}(u, v) = 1 - \frac{f_\Theta(u) \cdot f_\Theta(v)}{\|f_\Theta(u)\| \|f_\Theta(v)\|}, \tag{5}$$

where $\|f_\Theta(u)\|$ is the L_2 norm of $f_\Theta(u)$.

3. Related work

Semi-supervised learning is a learning paradigm with labeled and unlabeled data, and its performance has improved substantially, especially in image classification tasks. One commonly used approach for deep semi-supervised learning is pseudo-labeling [19]; it assigns pseudo-labels to unlabeled images for which the model's prediction score exceeds a threshold value. Another one is consistency regularization, which enforces consistent model predictions with different augmented images (e.g., Π-model [25], Mean Teachers [31], and VAT [21]). Methods combining these two approaches have also been

proposed (*e.g.* ReMixMatch [5], UDA [33], and FixMatch [27]) and have achieved high accuracy on public datasets such as CIFAR-10/100 [17] and ImageNet [8].

Label propagation (LP) [36] is a semi-supervised learning method that has existed since before the advent of deep learning. LP constructs a graph representing each image as a node and the similarity between images as an edge weight and propagates labels from labeled to unlabeled nodes. In contrast with other methods, LP leverages knowledge about how to measure similarity between instances. This is similar in spirit to ABC learning, which uses similarity metrics to construct a curriculum. However, as we will empirically show, our method performs much better than LP because ours can incorporate such knowledge into state-of-the-art algorithms.

Curriculum learning [4] is a method of machine learning that consumes training data in the order of "easy" to "hard." It reportedly can improve model performance in many kinds of tasks [28]. Various data difficulty criteria, such as annotation time [13] and classification loss [15, 18, 30], have been studied in image classification tasks. Some previous studies also incorporate the curriculum idea into semi-supervised learning [6, 35]. They use prediction confidence to determine the curriculum, but the confidence scores in deep learning are often misleading [11]. In contrast, ABC learning is based on the appearance of objects and does not rely on the model's confidence, thereby circumventing problems caused by over- or under-confidence.

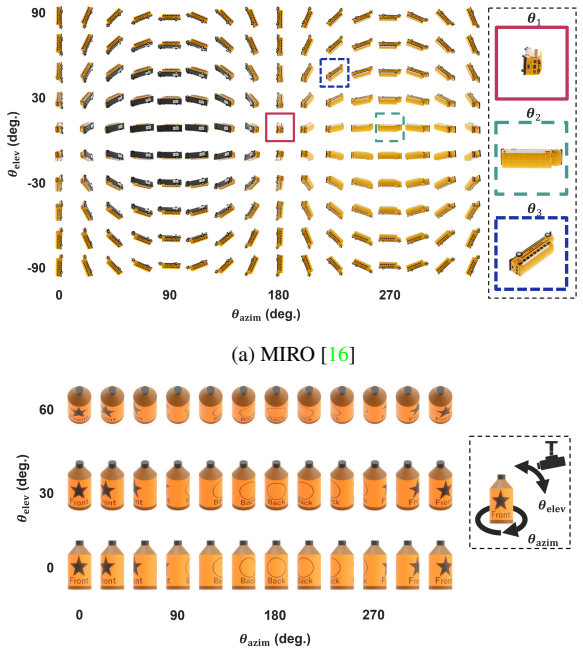
4. Experiments

This section presents our evaluation of our method (ABC) on three object recognition benchmarks. We compared ABC with three deep semi-supervised learning algorithms, Π-model [25], VAT [21], and FixMatch [27], as well as LP [10, 36] and supervised learning.

4.1. Experimental setup

Datasets. We evaluated our methods on three datasets, the MIRO dataset [16], DRINK dataset, and COIL-100 dataset [22]. They contains images of objects from various angles. Moreover, the information of camera angles is also included, which makes them useful for the purpose of evaluating our method. Some basic statistics about these datasets are summarized in Tab. 1.

MIRO [16] is a public dataset of commodity images from various viewpoints. It contains 120 objects (12 objects per class $\times$ 10 classes), and images of each object is taken from 160 angles (16 angles, ranging from 0 to 337.5° every 22.5° in azimuth, and ten angles, ranging from -90 to 90° every 20° in elevation) (Fig. 4 (a)). In this study, we treated this dataset as a 120-class object classification task. The 160 images of each object were split into the training, validation, and test splits containing 80, 40, and 40 images,



(a) MIRO [16]

(b) DRINK

Figure 4. Illutration of MIRO and DRINK. (a) Images of the `Bus_1` class in the MIRO dataset. The ones taken from the camera angles $\boldsymbol{\theta}_1, \boldsymbol{\theta}_2$, and $\boldsymbol{\theta}_3$ are boxed in the red, green, and blue boxes, respectively. (b) Concept images of the DRINK dataset.

respectively. Of the 80 training images, one image was selected as labeled, and the remaining 79 were selected unlabeled images. We chose three camera angles for labeled images: $(\theta_{\text{azim}}, \theta_{\text{elev}}) = (180, 10), (270, 10)$, and $(225, 50)$, denoted by $\boldsymbol{\theta}_1, \boldsymbol{\theta}_2$, and $\boldsymbol{\theta}_3$, respectively. We also evaluated the case in which a labeled angle is randomly picked from the three angles for each class; we denote this setting by $\boldsymbol{\theta}_{\text{r}}$.

We also privately collected DRINK, a real-world beverage product dataset, which contains images of 65 brands available in Japan (Fig. 4 (b)). This dataset requires fine-grained recognition based on patterns on the package surface than general object recognition because the beverages have a small variation in shape. Each product was placed on a rotating stage with the representative side facing front ($\theta_{\text{azim}} = 0°$) and shot at 0°, 30°, and 60° elevation angles while rotating the stage in 5° increments. Of the 216 images for each product, 108 were used as training data, 54 as validation data, and 54 as test data. Of the 108 train data, we chose as labeled images the ones taken from the camera angle $(\theta_{\text{azim}}, \theta_{\text{elev}}) = (0, 0)$, while the others were chosen as unlabeled images.

COIL-100 [22] is also a public dataset of commodity images from various viewpoints. COIL-100 contains images of 100 objects. Each object was placed on a rotating stage, and images were taken at every 5 degrees in a 360-degree

	# of classes	Train	Val	Test	Azimuth	Elevation
MIRO [16]	120	80	40	40	$0^\circ : 22.5^\circ : 337.5^\circ$	$-90^\circ : 20^\circ : 90^\circ$
DRINK	65	108	54	54	$0^\circ : 5^\circ : 355^\circ$	$0^\circ : 30^\circ : 60^\circ$
COIL-100 [22]	100	36	18	18	$0^\circ : 5^\circ : 355^\circ$	Fixed

Table 1. Dataset summary. Train, Val, and Test show the number of images per object class in the respective splits. Azimuth and Elevation show the ranges of the azimuthal and elevation angles, respectively. The notation $S^\circ : I^\circ : E^\circ$ represents the regularly spaced angles from S° to E° with the increments of I°, including both ends. The elevation angle of COIL-100 is fixed at about 25°.

rotation. The camera elevation angle was fixed. In the 72 images for each product, 36 were used as training data, 18 as validation data, and 18 as test data. In the 36 train data, only the front image taken from $\theta_{\text{azim}} = 0^\circ$ was considered labeled, while images taken from other angles were considered unlabeled.

With all the datasets, the three splits were made in a way in which the angular distribution of each split was nearly uniform. This ensures that all the splits have similar distributions of camera angles.

Implementation details. We used ResNet50 [12] as a classification model. In the experiments on MIRO and COIL-100, it was initialized with the parameters of the model pretrained on ImageNet [8], while in DRINK, we randomly initialized the model. The evaluation of semi-supervised learning algorithms uses the exponential moving average of model parameters over a training trajectory as in [27, 31]. We applied random flip and random crop as the data augmentation unless otherwise noted. The parameters of the entire model were optimized using SGD with Nesterov momentum [23, 29] of 0.9. For each setting, the learning rate, the total number of training iterations, and some of the algorithm-specific hyperparameters were tuned using grid search. The complete list of hyperparameters is given in Appendix B. We also note that we adopted $\text{dist}(u, D_L)$ instead of $\text{dist}(u, V)$ in ABC with d_{angle}.

We utilized the framework in [10] as the LP-based semi-supervised learning. This framework uses transductive LP to assign pseudo-labels to unlabeled data and feed them to supervised learning. We used the method of [36] as a transductive method.

We also conducted experiments with three supervised settings: Oracle, Supervised, and Supervised (w/ rot. aug.). "Oracle" indicates fully supervised learning, *i.e.*, using all the training images as annotated data, while "Supervised" indicates only one annotated image per object. "Supervised (w/ rot. aug.)" is similar to Supervised but with the additional inclusion of rotational data augmentation.

4.2. Experimental results

Table 2 shows the results on the MIRO dataset. With d_{global} as a distance function, ABC outperformed all the baseline methods by a large margin. Although ABC with d_{global} achieved better accuracy than that with d_{angle} or d_{local}, the latter still outperformed the deep semi-supervised learning methods. For example, ABC with d_{global} provided an accuracy gain of more than $20\,\%$ over FixMatch. Even ABC with d_{local} achieved $50.9\,\%$ for $\boldsymbol{\theta}_2$, compared with $43.5\,\%$ for FixMatch. In fact, deep semi-supervised methods including FixMatch struggled with this dataset and even failed to beat "Supervised," depending on the camera angle of a labeled image. On the other hand, LP showed strong performance with the edge weights determined by the global feature similarity. These results indicate the utility and importance of incorporating appearance similarity into semi-supervised learning in this scenario. We also note that rotation data augmentation can boost supervised learning, but its utility depends severely on the camera angle of labeled images. This reflects the fact that perspective projection and rotation transformations can mimic viewpoint changes to some extent but not completely.

	Accuracy [%]			
	$\boldsymbol{\theta}_1$	$\boldsymbol{\theta}_2$	$\boldsymbol{\theta}_3$	$\boldsymbol{\theta}_{\text{r}}$
Oracle	99.9			
Supervised	45.2	39.8	65.8	50.3
Supervised (w/ rot. aug.)	45.1	44.3	70.7	57.6
LP (local desc.)	41.2	37.6	68.7	48.2
LP (global feat.)	49.8	52.4	72.4	59.8
Π-model	36.6	37.0	66.9	47.8
VAT	42.6	40.4	71.0	51.1
FixMatch	39.3	43.5	65.4	50.7
ABC (d_{angle})	47.7	54.9	75.9	60.8
ABC (d_{local})	48.6	50.9	78.4	60.4
ABC (d_{global})	**64.9**	**67.6**	**86.5**	**75.3**

Table 2. Accuracy on the MIRO dataset. Each column shows the results with labeled images taken from the given angle (see Sec. 4.1). ABC (d_{global}) outperformed the other methods. ABC (d_{angle}) and ABC (d_{local}) also outperformed deep semi-supervised methods.

	Accuracy
Oracle	99.9
Supervised	46.4
Supervised (w/ rot. aug.)	36.1
LP (local desc.)	64.4
LP (global feat.)	50.1
Π-model	38.1
VAT	48.8
FixMatch	48.3
ABC (d_{angle})	79.9
ABC (d_{local})	**81.4**
ABC (d_{global})	71.5

Table 3. Accuracy on the DRINK dataset. ABC outperformed the other methods. Among them, ABC (d_{local}) and ABC (d_{angle}) achieved much higher accuracy than that of ABC (d_{global}).

	Accuracy
Oracle	100.0
Supervised	84.8
FixMatch	99.4
ABC (d_{angle})	**100.0**
ABC (d_{local})	98.7
ABC (d_{global})	99.4

Table 4. Accuracy on the COIL-100 dataset. ABC and the baseline FixMatch both performed nearly perfectly.

Table 3 shows the results on the DRINK dataset. Again, ABC outperformed all the other methods, but ABC with d_{local} and d_{angle} this time performed much better than ABC with d_{global}. A similar trend was observed with LP, *i.e.*, LP using the AKAZE descriptor achieved higher accuracy with DRINK, while the global features were preferred in the MIRO experiment. We will discuss this point in Sec. 4.3.

Table 4 shows the results on the COIL-100 dataset, which consists of images taken at a fixed elevation angle. Because no change occured in elevation angle, supervised learning with only labeled data achieved high accuracy than other datasets. Furthermore, FixMatch also achieved 99.4% accuracy. Our method achieved almost the same accuracy as FixMatch, demonstrating no adverse effects.

4.3. Analysis and ablation study

Quantity and quality of pseudo-labels. Here, we analyze the effect of ABC on pseudo-labeling. Figure 5 visualizes the accuracy of pseudo-labels and the proportion of pseudo-labeled images in the bag of unlabeled data over training iterations in the MIRO experiments.

With FixMatch, the number of pseudo-labels quickly increased and most of the unlabeled images got pseudo-labeled in the first half of training. At the same time, their accuracy degraded fast from nearly $100\,\%$ until it reached a plateau at about 15K iterations. The last-iteration accuracy was $65.8\,\%$. This was approximately equal to the test accuracy and reasonable because the unlabeled and test data had similar distributions.

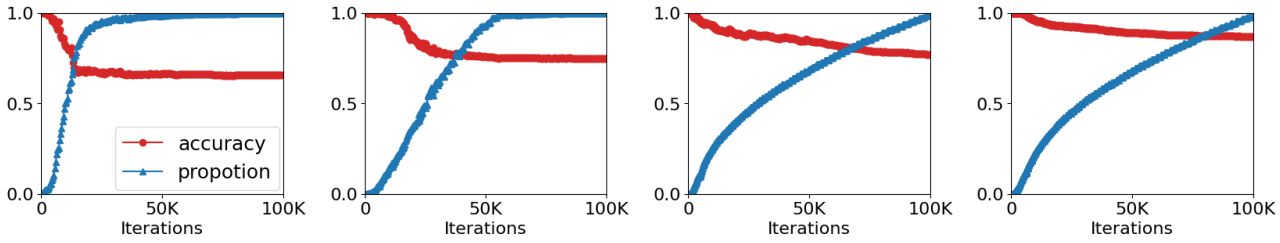
In contrast, ABC slowed the generation and deterioration of pseudo-labels. With d_{angle}, almost all the images were pseudo-labeled at about 50K iterations, but the rate of pseudo-labeling was much slower in the early phase of training. The accuracy stayed near $100\,\%$ for about 15K iterations and then started to lower gradually to $74.9\,\%$. With the other two distance functions, it took the whole training to assign pseudo-labels to the entire image pool. The accuracy curves did not have large plateaus like the ones observed in FixMatch and d_{angle}. Instead, they steadily but slowly degraded to $77.0\,\%$ (d_{local}) and $86.6\,\%$ (d_{global}).

These observations indicate that by controlling the ordering and speed of feeding unlabeled data, ABC exposes the underlying learner to cleaner pseudo-labeled data. This, in turn, leads to higher test accuracy.

When to feed additional unlabeled data. One important factor determining the performance of curriculum learning is the timing to add new data or the stopping criterion of SSL in Eq. (1). As explained in Sec. 2.2, we utilized the **Feed-on-Plateau** (FoP) criterion, which suspends SSL when the proportion of pseudo-labeled images in the unlabeled set reaches a plateau. Here, we also present an examination of the simpler, **Even-Interval** (EI) criterion, which feeds new unlabeled images at regular intervals. Specifically, if the curriculum has N steps and the total duration of training is T iterations, then this schedule brings additional data every $\lfloor T/N \rfloor$ iterations.

We evaluated the schedules on the MIRO $\boldsymbol{\theta}_3$ and DRINK datasets. The results are shown in Tab. 5. Regardless of the schedules, ABC outperformed the baseline FixMatch and is robust to a change in the stopping criteria. In both cases, ABC + FoP performed slightly better than ABC + EI. This suggests that FoP is a sensible default that does not require severe tuning.

Value of curriculum design. To quantify the importance of good curricula, we examined a curriculum based on the random ordering of unlabeled data. Specifically, we applied the update rule Eq. (1) with a randomly ordered W. As Tab. 6 shows, the randomized curriculum deteriorated the accuracy to $62.3\,\%$, which is even worse than $65.4\,\%$ of the baseline FixMatch. On the other hand, ABC reached $75.9\,\%$ to $86.5\,\%$, depending on the distance function used to construct the curricula. This suggests that the careful design of our curriculum is a crucial factor leading to the success of



(a) FixMatch (b) ABC (d_{angle}) (c) ABC (d_{local}) (d) ABC (d_{global})

Figure 5. Analysis of pseudo-labels during training on the MIRO $\boldsymbol{\theta}_3$ dataset. The red marks represent the accuracy of pseudo-labels and the blue marks represent the proportion of pseudo-labeled images in the set of unlabeled data. The number of pseudo-labels in ABC grows much slower than that in FixMatch. Also, the accuracy of pseudo-labels stays higher in ABC than in FixMatch.

	Accuracy	
	MIRO $\boldsymbol{\theta}_3$	DRINK
FixMatch	65.4	48.3
ABC (d_{angle}) + EI	74.4	79.5
ABC (d_{angle}) + FoP	**75.9**	**79.9**

Table 5. Effect of timing of feeding the next unlabeled dataset on the MIRO $\boldsymbol{\theta}_3$ and DRINK datasets. The FoP criterion achieved a slightly better accuracy than that of EI.

	Accuracy
FixMatch	65.4
ABC (d_{angle})	75.9
ABC (d_{local})	78.4
ABC (d_{global})	86.5
Random order	62.3

Table 6. Effect of ranking strategies on the MIRO $\boldsymbol{\theta}_3$ dataset. With the random order, the test accuracy became lower than the baseline FixMatch, indicating the importance of designing a good distance function for ABC to perform well.

our method.

The number of AKAZE descriptors. Because d_{local} relies on the matching of AKAZE descriptors between images, having the sufficient number of key points is crucial for the method to work successfully. In particular, the distance d_{local} between two images cannot be defined if one of them does not have any key point detected, and such an image is not utilized in constructing a curriculum. An unlabeled image with no key point detected is ranked the lowest in ABC, while a labeled image without any key points is not involved in the curriculum construction.

To understand the implication of this effect on the current experiments, we counted the number of key points found. Figure 6 shows the distribution of the number of AKAZE descriptors per image. This revealed that MIRO and COIL-100 had many images in which zero or very few key points were detected. On the other hand, most of the images in DRINK had 100 or more key points detected. This difference explains why ABC with d_{local} outperformed ABC with d_{global} on this dataset. It also suggests that the distribution of the AKAZE key points can be a good indicator of which distance function to use with a given dataset.

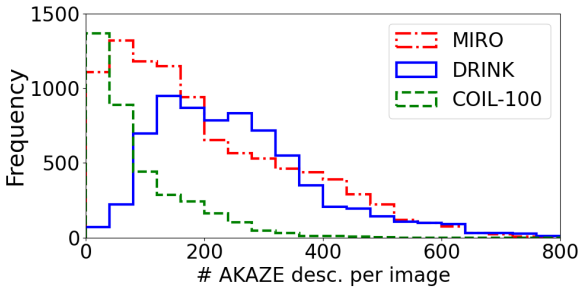


Figure 6. Histogram of the number of the detected AKAZE descriptors per image for each dataset. MIRO and COIL-100 have the modes close to zero key points per image, while more than 100 key points are detected in images in the DRINK dataset.

5. Conclusion

In this paper, we proposed an appearance-based curriculum (ABC) for a semi-supervised learning scenario where labeled images taken from limited angles and unlabeled ones taken from various angles are available for training. ABC incrementally expands the pool of unlabeled images fed to a base semi-supervised learner so that newly added data are the ones most similar to those already in the pool and that the quality of pseudo-labels is kept high during training. Extensive experiments showed that our method outperformed the state-of-the-art semi-supervised learning methods in our scenario. Analyses suggested that ABC and the proposed distance functions together succeeded in assigning correct pseudo-labels with high probability.

References

[1] Pablo Fernández Alcantarilla, Adrien Bartoli, and Andrew J. Davison. KAZE Features. In *European Conference on Computer Vision*, pages 214–227, 2012. 4

[2] Pablo Fernández Alcantarilla, Jesús Nuevo, and Adrien Bartoli. Fast Explicit Diffusion for Accelerated Features in Nonlinear Scale Spaces. In *British Machine Vision Conference*, 2013. 4

[3] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-Up Robust Features (SURF). *Computer Vision and Image Understanding*, 110(3):346–359, 2008. 4

[4] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum Learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 41–48, 2009. 5

[5] David Berthelot, Nicholas Carlini, Ekin D. Cubuk, Alex Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. ReMixMatch: Semi-Supervised Learning with Distribution Matching and Augmentation Anchoring. In *International Conference on Learning Representations*, 2020. 5

[6] Paola Cascante-Bonilla, Fuwen Tan, Yanjun Qi, and Vicente Ordonez. Curriculum Labeling: Revisiting Pseudo-Labeling for Semi-Supervised Learning. In *Proceedings of the AAAI conference on artificial intelligence*, pages 6912–6920, 2021. 5

[7] Olivier Chapelle, Bernhard Schölkopf, and Alexander Zien. *Semi-Supervised Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2006. 1

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 5, 6

[9] Edsger W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959. 3

[10] Ismail Elezi, Alessandro Torcinovich, Sebastiano Vascon, and Marcello Pelillo. Transductive Label Augmentation for Improved Deep Network Learning. In *International Conference on Pattern Recognition*, pages 1432–1437, 2018. 5, 6

[11] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On Calibration of Modern Neural Networks. In *Proceedings of the 34th International Conference on Machine Learning*, pages 1321–1330, 2017. 5

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 6

[13] Radu Tudor Ionescu, Bogdan Alexe, Marius Leordeanu, Marius Popescu, Dim P. Papadopoulos, and Vittorio Ferrari. How Hard Can It Be? Estimating the Difficulty of Visual Search in an Image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2157–2166, 2016. 5

[14] Vojtěch Jarník. O jistém problému minimálním: (Z dopisu panu O. Bor°uskovi). *Práce Moravské přírodovědecké společnosti*, pages 57–63, 1930. 3

[15] Lu Jiang, Deyu Meng, Teruko Mitamura, and Alexander G. Hauptmann. Easy Samples First: Self-Paced Reranking for Zero-Example Multimedia Search. In *Proceedings of the 22nd ACM International Conference on Multimedia*, pages 547—-556, 2014. 5

[16] Asako Kanezaki, Yasuyuki Matsushita, and Yoshifumi Nishida. RotationNet: Joint Object Categorization and Pose Estimation Using Multiviews From Unsupervised Viewpoints. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5010–5019, 2018. 5, 6

[17] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009. 5

[18] M. Pawan Kumar, Benjamin Packer, and Daphne Koller. Self-Paced Learning for Latent Variable Models. In *Advances in Neural Information Processing Systems*, 2010. 5

[19] Dong-Hyun Lee. Pseudo-Label : The Simple and Efficient Semi-Supervised Learning Method for Deep Neural Networks. In *International Conference on Machine Learning Workshop*, page 896, 2013. 4

[20] David G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60:91–110, 2004. 4

[21] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual Adversarial Training: A Regularization Method for Supervised and Semi-Supervised Learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(8):1979–1993, 2018. 4, 5, 11

[22] Sameer A. Nene, Shree K. Nayar, and Hiroshi Murase. Columbia Object Image Library (COIL-100). Technical report, Department of Computer Science, Columbia University, 1996. 5, 6

[23] Yurii Nesterov. A method of solving a convex programming problem with convergence rate $\mathcal{O}(1/k^2)$. In *Doklady Akademii Nauk*, pages 543–547, 1983. 6

[24] Robert C. Prim. Shortest connection networks and some generalizations. *The Bell System Technical Journal*, 36(6):1389–1401, 1957. 3

[25] Antti Rasmus, Mathias Berglund, Mikko Honkala, Harri Valpola, and Tapani Raiko. Semi-supervised Learning with Ladder Networks. In *Advances in Neural Information Processing Systems*, 2015. 4, 5

[26] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. ORB: An efficient alternative to SIFT or SURF. In *International Conference on Computer Vision*, pages 2564–2571, 2011. 4

[27] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A. Raffel, Ekin D. Cubuk, Alexey Kurakin, and Chun-Liang Li. FixMatch: Simplifying Semi-Supervised Learning with Consistency and Confidence. In *Advances in Neural Information Processing Systems*, pages 596–608, 2020. 2, 3, 5, 6, 11

[28] Petru Soviany, Radu Tudor Ionescu, Paolo Rota, and Nicu Sebe. Curriculum Learning: A Survey. *International Journal of Computer Vision*, 130(6):1526–1565, 2022. 5

[29] Ilya Sutskever, James Martens, George Dahl, and Geoffrey Hinton. On the importance of initialization and momentum

in deep learning. In *Proceedings of the 30th International Conference on Machine Learning*, volume 28, pages 1139–1147, 2013. 6

[30] Kevin Tang, Vignesh Ramanathan, Li Fei-fei, and Daphne Koller. Shifting Weights: Adapting Object Detectors from Image to Video. In *Advances in Neural Information Processing Systems*, 2012. 5

[31] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in Neural Information Processing Systems*, 2017. 4, 6

[32] Jesper E. Van Engelen and Holger H. Hoos. A survey on semi-supervised learning. *Machine learning*, 109(2):373–440, 2020. 2

[33] Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. Unsupervised Data Augmentation for Consistency Training. In *Advances in Neural Information Processing Systems*, pages 6256–6268, 2020. 5

[34] Xiangli Yang, Zixing Song, Irwin King, and Zenglin Xu. A Survey on Deep Semi-Supervised Learning. *IEEE Transactions on Knowledge and Data Engineering*, 35(09):8934–8954, 2023. 2

[35] Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. FlexMatch: Boosting Semi-Supervised Learning with Curriculum Pseudo Labeling. In *Advances in Neural Information Processing Systems*, pages 18408–18419, 2021. 5

[36] Xiaojin Zhu and Zoubin Ghahramani. Learning from Labeled and Unlabeled Data with Label Propagation. Technical report, CMU CALD, 2002. 5, 6, 11

[37] Xiaojin Zhu and Andrew B. Goldberg. *Introduction to Semi-Supervised Learning*. Morgan & Claypool Publishers, 2009. 1