










# Incremental Teacher Model with Mixed Augmentations and Scheduled Pseudo-label Loss for Handwritten Text Recognition

Masayuki Honda<sup>1</sup> , Hung Tuan Nguyen<sup>1</sup> , Cuong Tuan Nguyen<sup>1</sup> ,  
Cong Kha Nguyen<sup>2</sup> , Ryosuke Odate<sup>2</sup> , Takashi Kanemaru<sup>2</sup> ,  
and Masaki Nakagawa<sup>1</sup> 

<sup>1</sup> Tokyo University of Agriculture and Technology, 2-24-16 Naka-Cho, Koganei, Tokyo, Japan  
s183611u@st.go.tuat.ac.jp, {fx7297, fx4102}@go.tuat.ac.jp,  
nakagawa@cc.tuat.ac.jp

<sup>2</sup> Hitachi Ltd., Tokyo, Japan  
{cong\_kha.nguyen.zz, ryosuke.odate.qs,  
takashi.kanemaru.kf}@hitachi.com

**Abstract.** We propose a training framework for deep neural network-based handwritten text recognizers using both labeled and unlabeled data. The proposed framework is a semi-supervised learning (SSL) framework based on Mixed Augmentations and Scheduled Pseudo-Label loss. Mixed Augmentations provide weakly and strongly transformed variants from each original sample so that the pseudo-label loss is computed between these two variants. The Scheduled Pseudo-Label loss is used to gradually include the pseudo-label loss into the optimizer to avoid the negative effect of incorrect pseudo labels. First, a student model is pre-trained by labeled samples and used to initiate a teacher model. Subsequently, the teacher model predicts a pseudo label from every weakly transformed variant. On the other hand, the student model is trained using the Scheduled Pseudo-Label loss. Next, the teacher model is incrementally updated using the student model. Finally, it is used to evaluate. We term the framework Incremental Teacher Model. The proposed framework was applied to four architectures of distinct handwriting recognizers. For almost every architecture, the recognizer trained by our method outperforms those trained by well-known SSL methods, namely Mean Teacher, Pseudo-Labeling, and FixMatch, evaluated using different ratios of labeled training samples on the IAM handwriting database.

**Keywords:** Semi-Supervised Learning · Mixed augmentations · Scheduled Pseudo-Label loss · Training framework · Handwriting recognition

## 1 Introduction

Deep neural networks (DNNs) have been extensively studied in the past few decades and employed in multiple pattern recognition tasks owing to their high performance when large labeled datasets are available [1–3]. For handwriting recognition, DNNs

have achieved increasing recognition accuracy [4–6] on many benchmark databases [7–11] of Latin, Arabic, Chinese, Indic, and Japanese scripts. These models require more labeled samples for training when the number of parameters is high [12, 13]. On the other hand, they do not take advantage of unlabeled samples. Unlabeled samples are easier to collect in large quantities and at a lower cost than labeled samples. For example, the two new databases of handwritten answers, namely SCUT-EPT [14] and NCUEE-HJA [15], have 40,000 labeled sentences and more than 190,000 unlabeled sentences, respectively. Only a few studies have utilized unlabeled samples for handwritten text recognition [16, 17]. Thus, we aim to create a generalized learning framework for any handwriting recognizer that satisfies two criteria (i) Trainable with as less labeled data as possible; (ii) Utilizable for unlabeled and labeled data.

Thus far, semi-supervised learning (SSL) methods have been established and developed to address the use of unlabeled data. Since the early deep learning era, Pseudo-Labeling has been proposed and extended for image classification tasks [18]. In the Pseudo-Labeling method, a pre-trained model is initialized using a small, labeled subset and is then used to predict the pseudo labels of a large unlabeled subset. Next, the unlabeled subset with the corresponding pseudo labels is used to re-train the model. Generally, Pseudo-Labeling is similar to the teacher-student training framework, where the initialized supervised pre-trained model is a teacher model while the training model is a student model. The teacher model provides pseudo labels for training a student model with unlabeled input samples. Thus, the handwriting recognizer is optimized on both the labeled and unlabeled samples using features from the unlabeled samples.

In fact, the Pseudo-Labeling method depends on the quality of the pseudo labels, as erroneous predictions often appear early in the training process [19]. Handwritten text recognition (HTR) is considered a sequential labeling task requiring a sequence of character predictions. It is difficult to employ Pseudo-Labeling for training HTR because misrecognized labels might lead to incorrect predictions in the rest of the sequence. Hence, we propose a framework, termed the Incremental Teacher Model, to gradually extend the effect of pseudo labels during the training process. The teacher model is incrementally updated after each epoch by its student model.

We have not focused on developing a novel handwriting recognizer in this work. Instead, we employ the proposed framework to train existing handwriting recognition architectures: Convolutional Recurrent Neural Network (CRNN) with connectionist temporal classification (CTC) [20], Attention-based Encoder-Decoder (AED) [21], and Self-Attention-based CRNN with CTC [22]. These handwriting recognition architectures utilize unlabeled data using the proposed SSL framework.

The rest of this paper is organized as follows: Sect. 2 reviews related studies on SSL methods. Section 3 presents our proposed framework with Mixed Augmentations and Scheduled Pseudo-Label loss. Section 4 presents the experiments and results of the proposed framework applied to different HTR architectures. In Sect. 5, we draw conclusions.

## 2 Related Works

Although DNNs have been continuously improved for higher performance, they strongly depend on large-scale labeled datasets for training. In fact, it is difficult to efficiently adapt them to new tasks, such as recognizing unseen or seen characters written in a new writing style. During the last few years, meta-learning has been widely studied to make DNNs to learn new patterns with a few training samples [23]. It is a wide field of machine learning that includes few-shot learning, one-shot learning, and domain adaptation [17, 24, 25]. Among them, the domain adaptation (DA) methods, particularly methods following the SSL approach, are promising to generalize a handwriting recognizer using both labeled and unlabeled data. Specifically, we focus on the inner-domain handwriting recognition task where training and testing sets have the same categories.

Two main approaches are studied based on these assumptions: consistency regularization and entropy minimization. Consistency regularization is mainly based on data augmentation and weight noise by dropout, as small changes should not significantly affect the prediction made by the network. The consistency loss measures the distance between the network predictions, with and without augmentations for input samples. Some well-known methods in this approach are the  $\Pi$ -Model [26], Temporal Ensembling [26], Mean Teacher [27], and Virtual Adversarial Training (VAT) [28].

The  $\Pi$ -Model employs stochastic augmentation to provide minor changes in each input sample. It also applies dropout to make noise on the weights of a given DNN model. The distance between the predictions of the original sample (without either augmentation or dropout) and its variant (with both augmentation and dropout) is then minimized. While the  $\Pi$ -Model requires two executions of the network for every sample, Temporal Ensembling keeps and updates the ensembled prediction of every sample during the training process; thus, its computation cost is lower than that of the  $\Pi$ -Model. Mean Teacher focuses on updating the ensembled model instead of tracing the ensembled patterns so that it helps converge faster than Temporal Ensembling. On the other hand, VAT approximates how augmentations to be employed on each input sample affect the output class distribution most significantly.

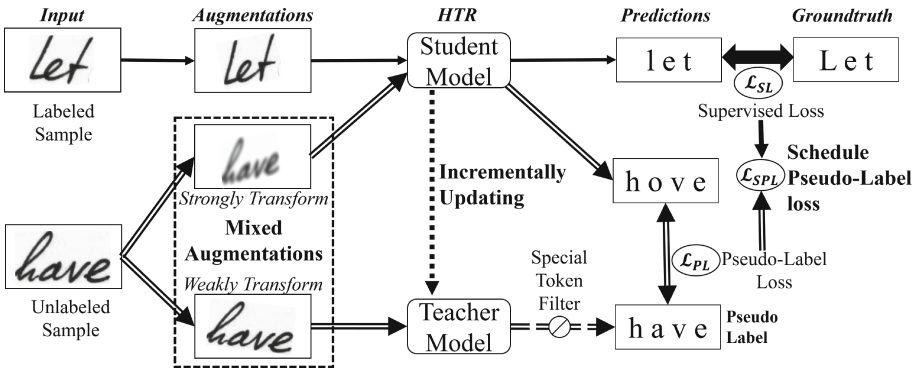
Entropy minimization prevents the decision boundary from lying near the low-confidence prediction region in the feature space. A simple loss term is commonly used to minimize the entropy for unlabeled data with all the classes. Two well-known methods based on entropy minimization are Pseudo-Labeling [18] and Label Propagation [29]. Pseudo-Labeling trains a student model based on a teacher model's predictions or pseudo labels, in which the teacher model is pre-trained using supervised learning. On the other hand, Label Propagation is to diffuse from labeled samples to unlabeled ones according to the propagation weights computed from pairwise similarity scores.

Recent studies have combined consistency regularization and entropy minimization, such as MixMatch [30] and FixMatch [31]. These methods apply multiple augmentations on a single unlabeled sample and force the model to predict these augmented input data similarly. By combining numerous augmentations, the trained model extracts invariant features to improve the overall performance even using a small number of labeled samples.

### 3 Methodology

By extending the Pseudo-Labeling method, we propose an SSL framework integrated with mixed augmentations and multiple losses, as shown in Fig. 1. First, an initial handwriting recognizer as a student model is prepared using labeled data by supervised learning. Second, mixed augmentations are applied to generate a weakly transformed variant and a strongly transformed variant from each original sample. Third, the teacher model produces a pseudo label from the weakly augmented variant and then computes a pseudo-label loss on the strongly augmented variant. For the prediction from the teacher model, the special tokens of padding or blank [PAD], start of sequence [SOS], and unknown [UNK] should not exist. These tokens are eliminated from the predictions to maintain the quality of the pseudo labels. Fourth, the student model is trained by minimizing both the supervised and pseudo-label losses with a flexible ratio. The ratio depends on the rate between labeled and unlabeled samples in a single training minibatch and the number of trained epochs. Note that the pseudo-label loss is gradually used to update the handwriting recognizer to avoid the negative effect of incorrect pseudo labels, termed the Scheduled Pseudo-Label loss. Finally, the teacher model was incrementally updated using the student model and used for evaluation.

Although the Mean Teacher and Pseudo-Labeling methods are the basis of this study, they follow different training schemes. Thus, we modified their training schemes similar to our model to achieve a fair comparison with the proposed framework in this study.



**Fig. 1.** Workflow of our proposed Incremental Teacher Model with Mixed Augmentations and Scheduled Pseudo-Label loss. The single-line arrows illustrate supervised learning using labeled samples, whereas the double-line arrows represent SSL with unlabeled samples.

#### 3.1 Incremental Teacher Model

Updating of the models that generate pseudo labels is handled differently depending on the research and application. In [18], the teacher model is commonly pre-trained and fixed; therefore, the predicted pseudo labels are stable for training the student model. This approach is good in the case where the teacher model is sufficiently trained on

labeled data. In practice, however, many labeled samples are not always available. On the other hand, methods that compute consistency regularization, such as Mean Teacher, can simultaneously train the student model and the teacher model that generates the pseudo labels in the training process. However, it might update the teacher model with a worse student model in the early stage of the training process. Thus, we propose to update the teacher model with the student model whenever the validation accuracy is improved at the end of each training epoch. The teacher model is updated by copying the weighted parameters from the student model. Finally, the teacher model was used for evaluation. To the best of our knowledge, this is the first work applying incremental updates of the teacher model for handwriting recognition using pseudo labels.

A well-initialized pre-trained model is essential to prepare a good teacher model to enhance the performance of the student model later. Because RotNet has been demonstrated to be effective for general images with complex background [32], we expected that it would be suitable for HTR with simple background. Moreover, the handwritten word image ratio was in range of general image ratio. Therefore, we employed RotNet, a self-supervised learning method for predicting the rotation of images, as a pretext task. This initialization method provides more general network weights to achieve a higher accuracy using supervised learning or SSL in the later training process.

### 3.2 Mixed Augmentations

In recent years, augmentation has played an important role in avoiding overfitting during the DNN training process [33] since it provides a large number of variants from a small number of samples. With more variants, a well-trained DNN model with augmentation tends to perform better extraction and focus on the invariant features. Since augmentation does not require newly collected data, it is commonly employed as an efficient method to improve the DNN performance. On the other hand, sequence-to-sequence contrastive learning (SeqCLR) has been proposed to employ stochastic image augmentation to generate two different variants from a single input sample [16]. Subsequently, the mapping between two extracted feature sequences is computed and considered the contrastive loss for optimization. In addition, augmentations are employed to generate multiple variants of a single sample for training based on prediction consistency [30].

In this study, we used multiple augmentation methods to generate two variants from a sample, which was named as “Mixed Augmentations”. One variant used smaller deformations to obtain a pseudo label, while the other had larger deformations. Note that the stochastic image augmentation in SeqCLR randomly generates two variants of an original sample using a single transforming pipeline repeatedly. Owing to the asymmetry of the proposed framework, two generated variants in our method are normally generated by two different transforming pipelines (weak and strong).

Augmentations used in general image recognition, such as FixMatch [31], are composed of geometric transforms for weak and multiple mixed transformations for strong transforms. For handwriting recognition, however, geometric transforms are limited to maintain the readability of the augmented handwritten images. Thus, we use four augmentations, namely rotation, crop, perspective, and Gaussian blur, which are commonly employed in handwriting recognition studies, as shown in Table 1. These settings are

based on comparative experiments and applied consistently in experiments with many HTR architectures and in different labeled ratio scenarios.

**Table 1.** Details of Mixed Augmentations.

Augmenta-tion	Description	Main parameter	Weak transformation	Strong transformation
Rotation	Randomly rotates the input text image between 0 and a parameter value	Rotation degree (deg)	15	15
Crop	Crops and enlarges a random area of the image by a specified percentage. Note that the aspect ratio is maintained	Crop percentage (%)	-	80
Perspective	Generates a perspective image with randomly transformed vertex positions according to the specified distortion ratio	Distortion percentage (%)	-	30
Gaussian Blur	Blurs an image by applying a Gaussian filter. The blur strength is specified by standard deviation	Sigma	-	2

### 3.3 Scheduled Pseudo-Label Loss

For training samples  $X$  with corresponding labels  $Y$ , the supervised loss is based on the negative log-likelihood as follows:

$$\mathcal{L}_{SL} = \sum_{(X,Y)} -\log p(Y|X) \quad (1)$$

The pseudo-label loss for the unlabeled training samples  $X$  is defined as follows:

$$\mathcal{L}_{PL} = \sum_{(X)} -\log p(\bar{Y}|\bar{X}) \text{ with } \bar{Y} = \text{teacher}(\bar{X}) \quad (2)$$

Here,  $\bar{X}$  and  $\overline{\bar{X}}$  are the weakly and strongly transformed variants from  $X$ , respectively. The pseudo labels  $\bar{Y}$  are predicted by the teacher model on  $\bar{X}$ . Thus, the pseudo-label loss is based on the conditional probabilities of the pseudo-label  $\bar{Y}$  for the strongly transformed variants  $\overline{\bar{X}}$ .

We introduce scheduling of the loss calculations for the pseudo labels of the unlabeled samples. It is aimed to avoid the problem that the target model does not converge due to the generation of incorrect pseudo labels in the early stages of training. Label scheduling has been proposed besides Pseudo-Labeling, and several derivations have been considered in other related studies. In this study, we applied the Scheduled Pseudo-Label loss as follows:

$$\mathcal{L}_{SPL} = \frac{1}{n} \mathcal{L}_{SL} + \alpha(t) \frac{1}{n'} \mathcal{L}_{PL} \quad (3)$$

where  $n$  is the total number of labeled samples,  $n'$  is the total number of unlabeled samples,  $t$  is the training epoch and  $\alpha(t)$  is the scheduled weight for  $\mathcal{L}_{PL}$  that depends on  $T_1$ ,  $T_2$ , and  $A$  as shown below:

$$\alpha(t) = \begin{cases} 0 & t < T_1 \\ \frac{t-T_1}{T_2-T_1} A & T_1 \leq t < T_2 \\ A & T_2 \leq t \end{cases} \quad (4)$$

Thus,  $\mathcal{L}_{PL}$  begins to affect  $\mathcal{L}_{SPL}$  when the number of epochs crosses  $T_1$  and monotonically increases until it reaches  $T_2$ ; then,  $A$  is the highest weight of  $\mathcal{L}_{PL}$ . In this study, we applied  $T_1$  of 50,  $T_2$  of 250, and  $A$  of 1, so that  $\mathcal{L}_{PL}$  is used from the midpoint of learning on the labeled data. Note that the current hyperparameters of the scheduled pseudo-label loss were experimentally chosen.

## 4 Experiments

### 4.1 IAM Handwriting Database and Scenarios for SSL

We used handwritten English word-level patterns of the IAM database for evaluation because they have been used as the benchmark for many HTR studies [7]. Although the SSL methods have been employed for many recognition tasks, they have not been widely applied in handwriting recognition as mentioned in the review section. For handwriting recognition, a sequence of characters is required for prediction instead of single characters. Thus, preliminary experiments at the word level are the most straightforward HTR task.

Table 2 shows four splitting scenarios derived from the RWTH Aachen University split<sup>1</sup> of the IAM handwriting database, where Words, Pages, and Writers denote the numbers of labeled and unlabeled samples in the training set, the number of samples

<sup>1</sup> <https://www.openslr.org/56/>.

in validation set, and that in the testing set, respectively. There is no writer duplication between the labeled and unlabeled samples. These scenarios are prepared to evaluate the SSL methods with our handwriting recognizers. These splitting scenarios satisfy the writer-independent requirement, which is commonly used to benchmark the handwritten English text recognizers.

Scenario 1 is the same as the supervised learning configuration without unlabeled samples. Scenarios 2, 3, and 4 are prepared to randomly select 50%, 10%, and 1% of the training set as the labeled training sets, respectively, while the rest is used as unlabeled training sets. Note that the labeled training set of Scenario 4 (1% labeled) does not include the eight character categories, which is over 10% of all character categories (8/79). Thus, Scenario 4 is the most challenging with unseen categories and writing styles.

**Table 2.** Details of SSL scenarios on IAM handwriting database.

Scenarios for SSL		IAM Subsets			
		Training set		Validation set	Testing set
		Labeled	Unlabeled		
Scenario 1 (100% labeled samples)	Words	55,081	0	8,895	25,920
	Pages	747	0	116	336
	Writers	283	0	56	161
Scenario 2 (50% labeled samples)	Words	27,727	27,354	Same as above	Same as above
	Pages	373	374		
	Writers	139	144		
Scenario 3 (10% labeled samples)	Words	5,364	49,717	Same as above	Same as above
	Pages	72	675		
	Writers	27	256		
Scenario 4 (1% labeled samples)	Words	551	54,530	Same as above	Same as above
	Pages	8	739		
	Writers	2	281		

## 4.2 Handwritten Text Recognition Architectures

As recognition models tested in the experiments, we used four architectures of handwriting recognizers. The first is a CRNN using ResNet as a feature extractor and Bidirectional Long Short-Term Memory (BLSTM) with CTC [20]. The second is another general encoder–decoder architecture, where an attention layer guides the decoder (AED) [21]. The third is a Deep Convolutional Recurrent Neural Network (DCRN) derived from AED with a simple Convolutional Neural Network (CNN) and a stacked BLSTM that provides a deeper sequential encoder [22]. The fourth is a CRNN using multiple Self-Attention layers for the sequential encoder (SelfAttn) [22]. These are listed in Table 3 with each major component.



**Table 3.** Main components of four HTR architectures.

Components	HTR Architectures			
	CRNN	AED	DCRN	SelfAttn
Feature Extractor (Local Encoder)	ResNet	ResNet	CNN	CNN
Sequential Encoder	BLSTM	BLSTM	Stacked BLSTM	BLSTM +SelfAttn
Sequential Decoder	CTC	LSTM +Attention	LSTM +Attention	CTC

### 4.3 Results of Different Recognition Architectures

To the best of our knowledge, no related research applied similar techniques to the HTR problem. The related studies were proposed for general image classification. For comparison, we experimented using Mean Teacher [27] and Pseudo-Labeling [18] because the proposed method is derived from them. Furthermore, we experimented using Fix-Match [31] as this is one of the most efficient SSL methods. Note that we modified these SSL methods to match with the training scheme used for our method.

Table 4 reports the results of four HTR architectures trained by different frameworks in each scenario. The baseline column shows the character accuracy rate (CAR) of the HTR architectures trained by only labeled samples, while the other columns show the CARs of trained HTR architectures using Mean Teacher, Pseudo-Labeling, FixMatch, and Incremental Teacher Model. For Pseudo-Labeling, we followed the default setting of scheduling parameters reported in [18]. Note that these reported results are on the IAM word-level testing set. The recognition rates shown here seem inferior to the state-of-the-art results [34] since these rates are obtained without word dictionaries and language models.

Overall, AED produced the best results in all scenarios with any training framework (bold), while CRNN typically produced the second-best results (underline). These results suggest that using a ResNet-based feature extractor seems to be better than the simple CNN. Moreover, the high complex sequential encoders of DCRN and SelfAttn did not achieve an accuracy as high as that of the simple sequential encoders of AED and CRNN. The performance of all the HTR architectures decreased significantly in Scenario 4 since the labeled training set did not cover the character set.

For the related SSL methods, Pseudo-Labeling outperformed Mean Teacher and FixMatch in almost all scenarios with all the HTR architectures. Note that in the case of the Mean Teacher and FixMatch methods, the performance of the HTR architecture is deteriorated in some cases, which is shown by  $\downarrow$  in Table 4. Mean Teacher and FixMatch mainly rely on the loss calculated from the distribution comparison between pseudo labels and output, as the consistency cost is unsuitable for text line recognition. It is considered difficult to capture the consistency because the output before decoding is a time series of classification, which varies significantly depending on the augmentation

**Table 4.** Character accuracy rate (%) of HTR architectures trained by Supervised Learning, Mean Teacher, Pseudo-Labeling, FixMatch, and Incremental Teacher Model in four SSL scenarios.

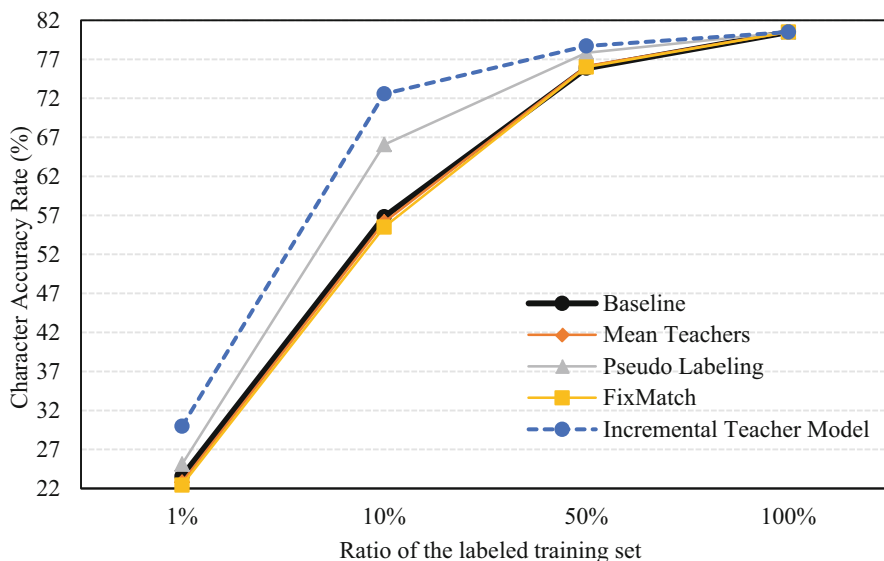
Data Split	HTR Architecture	Supervised Learning (Baseline)	Mean Teacher [27]	Pseudo-Labeling [18]	Fix Match [31]	Incremental Teacher Model (Our)
Scenario 1 (100%)	<u>CRNN</u>	<u>77.45</u>				
	<b>AED</b>	<b>80.49</b>				
	DCRN	74.72				
	SelfAttn	74.19				
Scenario 2 (50%)	<u>CRNN</u>	72.29	74.36	76.05	73.68	<u>76.22</u>
	<b>AED</b>	75.86	76.14	77.84	76.00	<b>78.71</b>
	DCRN	71.30	71.19↓	74.54	72.25	75.18
	SelfAttn	68.73	68.70↓	70.83	72.00	71.22
Scenario 3 (10%)	<u>CRNN</u>	53.83	57.93	62.44	54.40	<u>70.26</u>
	<b>AED</b>	56.81	56.14↓	66.06	55.49↓	<b>72.57</b>
	DCRN	48.76	55.76	58.29	51.10	61.62
	SelfAttn	48.62	51.60	55.37	51.33	60.88
Scenario 4 (1%)	<u>CRNN</u>	20.42	21.10	22.21	19.52↓	<u>24.99</u>
	<b>AED</b>	23.55	22.80↓	25.10	22.43↓	<b>29.96</b>
	DCRN	21.33	20.85↓	22.11	21.82	24.13
	SelfAttn	20.89	21.33	22.78	21.98	21.70

with positional information. Therefore, a method that expands on the pseudo labels is effective, and additional study is required to introduce consistency costs.

For every architecture except SelfAttn, the recognizer trained by the Incremental Teacher Model outperforms the recognizers trained by the well-known SSL methods: Mean Teacher, Pseudo-Labeling, and FixMatch in every scenario using only 50%, 10%, or 1% labeled training samples on the IAM handwriting database, respectively. The SelfAttn architecture with a simple feature extractor and a complex sequential encoder does not perform well in Scenarios 2 and 4. Mixed Augmentations seem to be helpful for the feature extractor rather than the sequential encoder.

Figure 2 illustrates the changes in the recognition accuracy with the increase in the ratio of labeled data in the training set. The Incremental Teacher Model increases the accuracy of the AED architecture by at most 15.7 percentage points (p.p.) in Scenario 3. Despite using the 1% labeled samples for training, the accuracy of AED is increased by 6.4 p.p. Compared to Pseudo-Labeling, it improves the HTR accuracy by at least 0.9 and at most 6.5 p.p. in Scenarios 2 and 3, respectively. These results show that the proposed framework could leverage unlabeled data to improve the HTR efficiency. Moreover, they give a clue about the possibility of applying HTR in practice on an unlabeled dataset by labeling only a small portion of the dataset.

Table 5 lists six word-level samples from the IAM handwriting database with the predictions from four architectures trained by Incremental Teacher Model. For short words such as “of”, “the” and “friend”, CRNN and AED correctly predicted while DCRN and SelfAttn produced misrecognitions. For longer words, even CRNN and AED did not perform correctly. The predictions by AED differed from the ground truth by one to two



**Fig. 2.** Character accuracy rate (%) of AED trained by different methods in four SSL scenarios.

characters while those by CRNN had more differences. The predictions by DCRN were shorter than the ground truth which might suggest that the DCRN capability is limited in the length of its output sequences. The SelfAttn architecture performed well with its predictions being different from the ground truth by only one to two characters.

**Table 5.** IAM word-level samples with predictions from four architectures trained using Incremental Teacher Model in Scenario 3.

Samples	<i>of</i>	<i>the</i>	<i>friend</i>	<i>original</i>	<i>natural</i>	<i>respectability</i>
Ground truth	of	the	friend	original	natural	respectability
CRNN	of	the	friend	<u>original</u>	<u>malural</u>	<u>eppectabet</u>
AED	of	the	friend	original	<u>natural</u>	<u>expectability</u>
DCRN	of	<u>He</u>	<u>find</u>	<u>logial</u>	<u>what</u>	<u>repathet</u>
SelfAttn	of	<u>he</u>	friend	<u>original</u>	<u>matual</u>	<u>nespectability</u>

#### 4.4 Results of Different Augmentation Configurations

Table 6 shows our search for weak/strong transformation settings, where we trained the AED architecture on Scenario 3 (10% of the training samples have been labeled). The most basic augmentation is rotation by at most 15 degrees (Rot15). Thus, we conducted a series of experiments with Rot15 as weak and strong transformations and inserted

other augmentations into the strong transformation, such as Crop80 (randomly removed at most 20% of an image), Blur2 (randomly applied Gaussian blur with the highest value of sigma of 2), and Per30 (randomly and vertically distorted an image by at most 30%). By employing more augmentations on the strong transformation, the AED performance increases from *R1* to *R5*. Moreover, we tried to eliminate Rot15 from weak transformation; however, *R6* performs worse than *R5* at 2.3 p.p. Next, we modified the parameters used for augmentations from the settings of *R5* to make *R7*. The small changes in the parameters might reduce the final recognition accuracy. Moreover, we tested to include more augmentations in the weak transformation. As shown in the *R8* and *R9* rows, the recognition accuracy declines when more augmentations are applied.

**Table 6.** Ablation studies for different configurations of Mixed Augmentations in Scenario 3.

Weak transformation	Strong transformation	Character accuracy (%)	Result IDs
Rot15	Rot15	67.79	<i>R1</i>
Rot15	Rot15+Crop80	69.10	<i>R2</i>
Rot15	Rot15+Crop80+Blur2	69.88	<i>R3</i>
Rot15	Rot15+Crop80+Per30	70.15	<i>R4</i>
<b>Rot15</b>	<b>Rot15+Crop80+Per30+Blur2</b>	<b>72.57</b>	<i>R5</i>
–	Rot15+Crop80+Per30+Blur2	70.25	<i>R6</i>
Rot15	Rot30+Crop70+Per40+Blur2	70.70	<i>R7</i>
Rot15+Crop90	Rot15+Crop80+Per30+Blur2	68.55	<i>R8</i>
Rot15+Crop80+Per30+Blur2	Rot15+Crop80+Per30+Blur2	67.41	<i>R9</i>

Thus, we might assume that simple augmentations are suitable for weak transformations. Moreover, we still need to search for the optimal parameters of Mixed Augmentations.

#### 4.5 Discussions

Based on the experiments, the AED model outperformed other models, which may be owing to its components of a ResNet-based feature extractor and an LSTM-based decoder with attention. These components are large and deep to extract useful features for recognition and correctly focus on character regions. Thus, they are commonly used to build handwriting recognizers. Because these experiments were on word-level patterns only, further experiments on sentence-level are required to verify the efficacy of the proposed framework. We believe that designing the consistency cost for long handwritten text is challenging. As it is impractical to investigate all types of augmentation in this study, we selected and applied the augmentations commonly used with better performance on HTR. However, we expect that other augmentations are also possible to be employed in the proposed framework.

## 5 Conclusions

We proposed Incremental Teacher Model and demonstrated its effectiveness. It produces a high recognition accuracy for handwritten text recognition even when only a part of the training set is labeled. It comprises Mixed Augmentations and Scheduled Pseudo-Label loss for handwritten text recognition. Instead of using a fixed pre-trained handwritten text recognition (HTR) model as a teacher model to generate pseudo labels, the proposed framework incrementally updates the teacher model using the latest recognizer. We applied the proposed framework to four DNN architectures for handwriting recognition and compared it with well-known semi-supervised learning methods: Mean Teacher, Pseudo-Labeling, and FixMatch. For almost every architecture, the recognizer trained by the Incremental Teacher Model outperforms the recognizers trained by other well-known SSL methods in every scenario when using only 50%, 10%, or 1% labeled training samples on the IAM handwriting database. However, we only confirmed the effectiveness of our framework for word-level English, so we plan to examine the framework for text-line-level English as well as for other languages in future works.

**Acknowledgement.** We thank anonymous reviewers for helpful comments on the manuscript. This work is partially supported by the joint research budget from Hitachi, Ltd. and Kakenhi (S) 18H05221.

## References

1. Krizhevsky, A., Sutskever, I., Hinton, G.: ImageNet classification with deep convolutional neural networks. In: The 25th Neural Information Processing Systems, pp. 1106–1114 (2012). <https://doi.org/10.1145/3065386>
2. van den Oord, A., et al.: WaveNet: a generative model for raw audio. In: The 9th ISCA Speech Synthesis Workshop, p. 125 (2016). <https://doi.org/10.1109/ICASSP.2009.4960364>
3. Vinyals, O., Toshev, A., Bengio, S., Erhan, D.: Show and tell: lessons learned from the 2015 MSCOCO image captioning challenge. *IEEE Trans. Pattern Anal. Mach. Intell.* **39**, 652–663 (2017). <https://doi.org/10.1109/TPAMI.2016.2587640>
4. Graves, A., Schmidhuber, J.J.: Offline handwriting recognition with multidimensional recurrent neural networks. In: The 21st International Conference on Neural Information Processing Systems, pp. 545–552 (2008). <https://doi.org/10.1007/978-1-4471-4072-6>
5. Puigcerver, J.: Are multidimensional recurrent layers really necessary for handwritten text recognition? In: The 14th International Conference on Document Analysis and Recognition, pp. 67–72 (2017). <https://doi.org/10.1109/ICDAR.2017.20>
6. Bluche, T.: Joint line segmentation and transcription for end-to-end handwritten paragraph recognition. In: The 30th International Conference on Neural Information Processing Systems, pp. 838–846 (2016). <https://doi.org/10.5555/3157096.3157190>
7. Marti, U.V., Bunke, H.: The IAM-database: an English sentence database for offline handwriting recognition. *Int. J. Doc. Anal. Recognit.* **5**, 39–46 (2003). <https://doi.org/10.1007/s100320200071>
8. Shivram, A., Ramaiah, C., Setlur, S., Govindaraju, V.: IBM-UB-1: a dual mode unconstrained english handwriting dataset. In: The 12th International Conference on Document Analysis and Recognition, pp. 13–17 (2013). <https://doi.org/10.1109/ICDAR.2013.12>

9. Mahmoud, S.A., et al.: KHATT: an open Arabic offline handwritten text database. *Pattern Recognit.* **47**, 1096–1112 (2014). <https://doi.org/10.1016/j.patcog.2013.08.009>
10. Liu, C.L., Yin, F., Wang, D.H., Wang, Q.F.: CASIA online and offline Chinese handwriting databases. In: *The 11th International Conference on Document Analysis and Recognition*, pp. 37–41 (2011). <https://doi.org/10.1109/ICDAR.2011.17>
11. Kumar Bhunia, A., et al.: Handwriting trajectory recovery using end-to-end deep encoder-decoder network. In: *The 24th International Conference on Pattern Recognition*, pp. 3639–3644 (2018). <https://doi.org/10.1109/ICPR.2018.8546093>
12. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: *The 29th IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778 (2016). <https://doi.org/10.1109/CVPR.2016.90>
13. Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: *The 30th IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2261–2269 (2017). <https://doi.org/10.1109/CVPR.2017.243>
14. Zhu, Y., Xie, Z., Jin, L., Chen, X., Huang, Y., Zhang, M.: SCUT-EPT: new dataset and benchmark for offline Chinese text recognition in examination paper. *IEEE Access.* **7**, 370–382 (2019). <https://doi.org/10.1109/ACCESS.2018.2885398>
15. Nguyen, H.T., Nguyen, C.T., Oka, H., Ishioka, T., Nakagawa, M.: Handwriting recognition and automatic scoring for descriptive answers in Japanese language tests. In: Porwal, U., Fornés, A., Shafait, F. (eds.) *ICFHR 2022. LNCS*, vol. 13639, pp. 274–284. Springer, Cham (2022). [https://doi.org/10.1007/978-3-031-21648-0\\_19](https://doi.org/10.1007/978-3-031-21648-0_19)
16. Aberdam, A., et al.: Sequence-to-sequence contrastive learning for text recognition. In: *The 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15297–15307 (2021). <https://doi.org/10.1109/CVPR46437.2021.01505>
17. Kang, L., Rusiñol, M., Fornés, A., Riba, P., Villegas, M.: Unsupervised adaptation for synthetic-to-real handwritten word recognition. In: *The IEEE/CVF Winter Conference on Applications of Computer Vision* (2020). <https://doi.org/10.1109/WACV45572.2020.9093392>
18. Lee, D.-H.: Pseudo-label: the simple and efficient semi-supervised learning method for deep neural networks. In: *ICML 2013 Workshop: Challenges in Representation Learning*, pp. 1–6 (2013)
19. Rizve, M.N., Duarte, K., Rawat, Y.S., Shah, M.: In defense of pseudo-labeling: an uncertainty-aware pseudo-label selection framework for semi-supervised learning. In: *The 9th International Conference on Learning Representations* (2022). <https://doi.org/10.48550/arXiv.2101.06329>
20. Xie, Z., Sun, Z., Jin, L., Feng, Z., Zhang, S.: Fully convolutional recurrent network for handwritten Chinese text recognition. In: *The 23rd International Conference on Pattern Recognition*, pp. 4011–4016 (2016). <https://doi.org/10.1109/ICPR.2016.7900261>
21. Sueiras, J., Ruiz, V., Sanchez, A., Velez, J.F.: Offline continuous handwriting recognition using sequence to sequence neural networks. *Neurocomputing* **289**, 119–128 (2018). <https://doi.org/10.1016/J.NEUCOM.2018.02.008>
22. Ly, N.T., Ngo, T.T., Nakagawa, M.: A self-attention based model for offline handwritten text recognition. In: Wallraven, C., Liu, Q., Nagahara, H. (eds.) *ACPR 2022. LNCS*, vol. 13189, pp. 356–369. Springer, Cham (2022). [https://doi.org/10.1007/978-3-031-02444-3\\_27](https://doi.org/10.1007/978-3-031-02444-3_27)
23. Munkhdalai, T., Yu, H.: Meta networks. In: *The 34th International Conference on Machine Learning*, pp. 2554–2563 (2017). <https://doi.org/10.48550/arXiv.1703.00837>
24. Souibgui, M.A., Fornés, A., Kessentini, Y., Megyesi, B.: Few shots are all you need: a progressive learning approach for low resource handwritten text recognition. *Pattern Recogn. Lett.* **160**, 43–49 (2022). <https://doi.org/10.1016/J.PATREC.2022.06.003>

25. Chakrapani Gv, A., Chanda, S., Pal, U., Doermann, D.: One-shot learning-based handwritten word recognition. In: Palaiiahnakote, S., Sanniti di Baja, G., Wang, L., Yan, W.Q. (eds.) ACPR 2019. LNCS, vol. 12047, pp. 210–223. Springer, Cham (2020). [https://doi.org/10.1007/978-3-030-41299-9\\_17](https://doi.org/10.1007/978-3-030-41299-9_17)
26. Laine, S., Aila, T.: Temporal ensembling for semi-supervised learning. In: The 5th International Conference on Learning Representations (2016). <https://doi.org/10.48550/arXiv.1610.02242>
27. Tarvainen, A., Valpola, H.: Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In: The 31st International Conference on Neural Information Processing Systems, pp. 1195–1204 (2017). <https://doi.org/10.48550/arxiv.1703.01780>
28. Miyato, T., Maeda, S.I., Koyama, M., Ishii, S.: Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE Trans. Pattern Anal. Mach. Intell.* **41**, 1979–1993 (2017). <https://doi.org/10.48550/arxiv.1704.03976>
29. Iscen, A., Tolia, G., Avrithis, Y., Chum, O.: Label propagation for deep semi-supervised learning. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5065–5074 (2019). <https://doi.org/10.1109/CVPR.2019.00521>
30. Berthelot, D., Carlini, N., Goodfellow, I., Oliver, A., Papernot, N., Raffel, C.: MixMatch: a holistic approach to semi-supervised learning. In: The 33rd International Conference on Neural Information Processing Systems, pp. 5049–5059 (2019). <https://doi.org/10.48550/arXiv.1905.02249>
31. Sohn, K., et al.: FixMatch: simplifying semi-supervised learning with consistency and confidence. In: The 34th International Conference on Neural Information Processing Systems, pp. 596–608 (2020). <https://doi.org/10.5555/3495724.3495775>
32. Gidaris, S., Singh, P., Komodakis, N.: Unsupervised representation learning by predicting image rotations. In: The 6th International Conference on Learning Representations (2018). <https://doi.org/10.48550/arXiv.1803.07728>
33. Shorten, C., Khoshgoftaar, T.M.: A survey on image data augmentation for deep learning. *J. Big Data* **6**(1), 1–48 (2019). <https://doi.org/10.1186/s40537-019-0197-0>
34. Bhunia, A.K., Das, A., Bhunia, A.K., Kishore, P.S.R., Roy, P.P.: Handwriting recognition in low-resource scripts using adversarial learning. In: The IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 4762–4771 (2019). <https://doi.org/10.1109/CVPR.2019.00490>