# **Relational Contrastive Learning for Scene Text Recognition**

Jinglei Zhang<sup>\*</sup>, Tiancheng Lin<sup>\*</sup>, Yi Xu<sup>†</sup>, Kai Chen, Rui Zhang

MoE Key Lab of Artificial Intelligence, AI Institute, Shanghai Jiao Tong University

Shanghai, China

{zhangjinglei168,ltc19940819,xuyi,kchen,zhang\_rui}@sjtu.edu.cn

#### ABSTRACT

Context-aware methods achieved great success in supervised scene text recognition via incorporating semantic priors from words. We argue that such prior contextual information can be interpreted as the relations of textual primitives due to the heterogeneous text and background, which can provide effective self-supervised labels for representation learning. However, textual relations are restricted to the finite size of dataset due to lexical dependencies, which causes the problem of over-fitting and compromises representation robustness. To this end, we propose to enrich the textual relations via rearrangement, hierarchy and interaction, and design a unified framework called RCLSTR: Relational Contrastive Learning for Scene Text Recognition. Based on causality, we theoretically explain that three modules suppress the bias caused by the contextual prior and thus guarantee representation robustness. Experiments on representation quality show that our method outperforms state-of-the-art self-supervised STR methods. Code is available at https://github.com/ThunderVVV/RCLSTR.

### **1 INTRODUCTION**

Self-supervised learning (SSL), especially contrastive learning methods [5, 7-9, 15, 17, 41, 47], has achieved great success in computer vision tasks for natural images. An excellent visual representation learned from unlabeled data is attractive for scene text recognition (STR). Otherwise, a mass of labeled data is usually needed for training to decode the contained text from images [12, 25, 45]. Directly transferring the contrastive learning methods of natural images to scene text images is sub-optimal since the characteristics of scene text images are quite different from natural images. We argue that text images mainly have the following essential characteristics. First, foreground (i.e., text) and background are heterogeneous in text images, and text recognition relies primarily on text rather than the background. Second, text images are known to have a left-toright structure. Third, besides the whole image, text images contain the sequence of characters and structure of multi-granularity. Significantly these text characteristics should be fully explored and accordingly propose a new framework of SSL on scene text images.

Some pioneering works [1, 27, 50] have explored how to construct variants of contrastive learning for text recognition. Seq-CLR [1] considers scene text images as a sequence of subwords and thus proposes an instance-mapping function, which makes the atoms of contrastive learning to be sequential frames (*i.e.*, the subwords) rather than images (*i.e.*, the whole image). PerSec [27] conducts contrastive learning on low-level and high-level features, aiming to simultaneously learn the representations from stroke and



(c) Cross-Hierarchy Similarity

Figure 1: Textual Relations. (a) In the dataset, the context is limited. By rearrangement (shuffling and concatenating), we can create a richer context for regularization. (b) Text images naturally have hierarchical features on multiple levels. The most granular level is character. Multiple characters form a subword, and multiple subwords form a word. We use "CL" to denote contrastive learning. (c) For cross-hierarchy relations, the character presents higher similarity with the subword from the same region than the subwords in other regions. Similarly, the subword shows higher similarity with the word from the same image than the words in other images.

semantic context. DiG [50] directly integrates contrastive learning and masked image modeling (MIM) into a unified model for text recognition, while the gains mainly come from the powerful ViT [11] and MIM [49]. These above self-supervised methods are mainly transferred from natural images and only partially explore the text characteristics. Unlike them, our idea is rooted in the supervised text recognition in our community, aiming at fully exploring the characteristics of the text. In particular, context-aware

<sup>\*</sup>Both authors contributed equally to this research.

<sup>&</sup>lt;sup>†</sup>Corresponding author.

methods [13, 36, 51] achieved great success by incorporating semantic priors from words in a supervised fashion. We argue that such contextual information can be interpreted as the relations of textual primitives, thus, can be utilized in an unsupervised way. Unfortunately, textual relations are restricted to the finite size of the dataset, which usually causes the problem of over-fitting due to lexical dependencies [42]. To address this problem, we propose to enrich the textual relations via rearrangement, hierarchy and interaction, resulting in a more complete contrastive mechanism. For "rearrangement", text images can be divided and rearranged into new context relations. For "hierarchy", there are multi-level relations in text images, such as words, subwords and characters. For "interaction", we can leverage the interactions among the objects of different levels, *e.g.*, character-subword and subword-word similarities.

Correspondingly, we propose RCLSTR: Relational Contrastive Learning for Scene Text Recognition, with three novel modules to fully explore the relations in texts. First, we design a relational regularization module to generate new word images, enriching the variety and diversity of relations. Instead of enumerating all possible relations in texts, which is impractical, we turn to creating new images on-the-fly. The rearrangement of images creates richer context relations. For example, as shown in Figure 1 (a), words (e.q., "justify" and "notice") can be broken up into subwords of roots and affixes and new words of "justice" and "notify" can be achieved by rearrangement. In practice, we generate new permuted images by horizontal division and concatenation, since text images usually have left-to-right direction. Note that the position labels of roots and affixes are unavailable in SSL, so the ideal image division is not attainable. Our experiments further study multiple strategies for image division and find that ideal division is not necessary for this module. Second, a hierarchical structure is proposed to conduct representation learning at multiple levels of primitives, which is motivated by the fact that texts have multiple objects with different granularities. As shown in Figure 1 (b), at the highest level, one image is taken as a whole to learn the representations of words. The words can be divided into subwords (e.g., roots and affixes) in the middle level, and they work as functional language units from the linguistic perspective [38]. The lowest level of characters is the atomic elements of texts. We hypothesize that mining multilevel relations in a hierarchical structure could enrich semantic information and enhance representation learning. Third, besides the intra-hierarchical relations, we further propose consistency constraints to explore the inter-hierarchical relations. As shown in Figure 1 (c), the characters (at the lowest level) and subwords (at the middle level) from the same locations (in the same images) share similar attributes in color and stroke, thus showing higher similarity in the feature space. The same phenomenon is also found across the levels of subwords and words. Therefore, we are motivated to explicitly constrain the consistency of semantic similarity across the hierarchical levels of text images. We hypothesize that enabling interaction across multiple levels can facilitate the learning of highquality representations in a more effective manner.

We summarize the contributions of this work as follows:

• We propose to explore the relations in text images for selfsupervised learning. Text images encode rich contextual information in the relations among textual primitives, which are essential for contrastive learning.

- We propose a novel framework RCLSTR: Relational Contrastive Learning for Scene Text Recognition, which includes three novel modules for exploring relational regularization, hierarchical relations and inter-hierarchy relational consistency.
- Our RCLSTR achieves superior performance over the stateof-the-art self-supervised STR methods on representation quality. Moreover, the effectiveness of key model components is verified by the ablation study.

#### 2 RELATED WORK

#### 2.1 Self-Supervised Learning

For natural image self-supervised learning, contrastive learning methods [5, 7-9, 15, 17, 41, 47] show great success, which performs the instance discrimination task to classify different dataaugmentation views from the same image into a class. In NPID [47], the task of instance discrimination is proposed, and noise contrastive estimation (NCE) is used for contrastive learning, which is further replaced by InfoNCE [41]. MoCo [17] and SimCLR [7] improve the quality of learned representations, which proposes the momentum encoder and uses a single network with a large batch size, respectively. SwAV [5] constrains the consistency of cluster allocation of different data-augmentation views. Recently, BYOL [15] and Simsiam [9] further propose asymmetric frameworks which do not need negative samples. More recently, some works [33, 34, 46, 55] propose to use KL divergence to constrain relative consistency in the form of similarity distribution. However, these self-supervised methods are designed for natural images, which is quite different from text images. Considering the characteristics of text, we need to specially design the self-supervised method for text images.

#### 2.2 Self-Supervised Text Recognition

Some pioneering works [1, 3, 27, 29, 50] explored self-supervised methods in text recognition and have achieved promising results. We summarize these methods into three main categories. The first approach is based on contrastive learning. SeqCLR [1] maps sequence features of words to instances as atomic elements of contrast learning. They only consider the text sequence structure. Per-Sec [27] proposed to conduct contrastive learning on the low-level stroke and the high-level semantic features of text images corresponding to visual and semantic information. The second is based on mask image model. DiG [50] proposed a self-supervised framework for text recognition that combines contrastive learning and masked image models(MIM). Concurrent work [30] also uses MIM for STR. However, these MIM methods are directly transferred from natural image methods. The third is based on generative learning, and a representative work is SimAN [29], which proposes to reconstruct the images from the decoupled content and style information. In sum, the above methods have not fully explored the characteristics of text images. Our approach takes into account the heterogeneity of texts, the left-to-right structure and the hierarchical structure of the sequence to fully explore the text characteristics. We propose a novel contrastive learning framework to enrich the textual relations via rearrangement, hierarchy and interaction.



Figure 2: Block diagram. Each image in a batch is augmented twice and then fed separately into the online branch (top) and the momentum branch (bottom) of the encoder and projector to create pairs of representation maps. In relational regularization module, we randomly permute the image patches and undo permutation on their features. Next, for the hierarchical contrastive learning of these representations, we apply three predictors that transform them into frames, subwords and words, respectively. In relational consistency module, the corresponding positions on three circles represent the same spatial positions across different hierarchical levels. And we take the corresponding frame & subword or subword & word as positive pairs. The diagram uses the corresponding colors to represent the three levels.

#### 3 METHOD

Based on the structure of MoCo [17], an efficient and effective baseline, we propose the relational contrastive learning framework for text recognition (RCLSTR). As shown in Figure 2, we introduce a novel permutation stage in the online branch (upper branch) to yield horizontal permuted images from the original, which is denoted as relational regularization module (Sect. 3.2). In addition, we design a hierarchical structure to learn relations at each level, which is called hierarchical relation module (Sect. 3.3). Meanwhile, we propose a cross-hierarchy relational consistency module (Sect. 3.4) so that the network learns the relation between hierarchies.

#### 3.1 Preliminaries

**Text recognition framework.** As we focus on general contrastive learning for text images, we follow SeqCLR [1] and use a general text recognition framework in [2]. This framework is the foundation of many text recognizers, which consists of an encoder and a decoder. In the encoder, we use a Thin Plate Spline (TPS) transformation [39] and a feature extraction network. The decoder can be a CTC-based decoder [14] or attention-based decoder [10]. Note that there are other kinds of text recognition architectures [13, 25, 45, 48] in recent research, and it is expected our RCLSTR can also be applied to them.

**Contrastive learning.** Contrastive learning methods [6–8, 17] perform an instance discrimination pretext task in the pre-training phase. This pretext task trains the model to discriminate the positive view from the negative views. The query view  $X_i^q$  and positive view  $X_i^p$  are encoded as **q** and **p**. To avoid the need for large batchsize, we follow MoCo to maintain a queue of size K, and there are K negative features  $\{\mathbf{n}_k\}_{k=1}^K$  from other images. Then, the contrastive loss of InfoNCE is written as:

$$\mathcal{L}_{info}(\mathbf{q}, \mathbf{p}, \mathbf{n}) = -\log \frac{\exp(\mathbf{q} \cdot \mathbf{p} / \tau_{info})}{\sum_{\mathbf{u} \in \{\mathbf{n}_k\}_{k=1}^K \cup \{\mathbf{p}\}} \exp(\mathbf{q} \cdot \mathbf{u} / \tau_{info})}, \quad (1)$$

where  $\tau_{info}$  is a temperature hyper-parameter. This loss function aims to pull closer together features of positive pairs and to push all the other negative examples farther apart.

**Naive relational contrastive learning.** Relational contrastive learning aims at learning not only the relation between query views and positive views, but also the relation between query views and negative views. Inspired by [44, 46, 55], we calculate the similarity between the positive and the negatives (*i.e. P*) and that between the query and the negatives (*i.e. Q*). We encourage the agreement of two similarity distributions. Formally, we use symmetric Kullback-Leibler (KL) Divergence as the measure of disagreement, imposing consistency between *P* and *Q*:

$$Q_{i}(\mathbf{q}, \mathbf{n}) = \frac{\exp(\mathbf{q} \cdot \mathbf{n}_{i}/\tau_{kl})}{\sum_{k=1}^{K} \exp(\mathbf{q} \cdot \mathbf{n}_{k}/\tau_{kl})},$$

$$P_{i}(\mathbf{p}, \mathbf{n}) = \frac{\exp(\mathbf{p} \cdot \mathbf{n}_{i}/\tau_{kl})}{\sum_{k=1}^{K} \exp(\mathbf{p} \cdot \mathbf{n}_{k}/\tau_{kl})},$$

$$\mathcal{L}_{kl}(\mathbf{q}, \mathbf{p}, \mathbf{n}) = \frac{1}{2}D_{\mathrm{KL}}(P||Q) + \frac{1}{2}D_{\mathrm{KL}}(Q||P),$$
(2)

where  $\tau_{kl}$  is also a temperature hyper-parameter. The total relational loss is a weighted average of the InfoNCE loss term and the KL loss term:

$$\mathcal{L}_{re}(\mathbf{q}, \mathbf{p}, \mathbf{n}) = \mathcal{L}_{info}(\mathbf{q}, \mathbf{p}, \mathbf{n}) + \alpha \mathcal{L}_{kl}(\mathbf{q}, \mathbf{p}, \mathbf{n}), \tag{3}$$

where  $\alpha$  denotes the coefficient to balance the two terms. The first term is the absolute similarity constraint between **q** and **p**. The second term is the relative similarity constraint, which aims to keep the similarity distribution consistency of **q** and **p** with the negatives.

However, due to the finite size of the dataset, the textual relations are restricted, and the performance of naive relational contrastive learning is limited. Therefore, we propose relational regularization, hierarchical relation and cross-hierarchy relation modules to learn richer textual relations, building a more complete relational contrastive learning framework.



Figure 3: An illustration of the random permutation operation, which generates features for relational regularization.

### 3.2 Relational Regularization

STR model usually treats each feature in a sequence as the atom for prediction. Previous works [42, 54] have pointed out that text recognizers are prone to over-dependence on context. It should be noted that the previous methods are used for supervised learning, while our method is proposed for unsupervised learning. In order to alleviate this context-dependent problem, we propose a permutation module to generate new text images. The generated images contain more diversity of context relations, encouraging the encoder not to over-fitting finite contexts in the dataset. Generally, the process goes like 1) dividing text images horizontally into several patches, 2) randomly shuffling and concatenating patches to generate permuted images, and 3) adding a regularization loss term corresponding to these permuted images.

Specifically, the permutation operation is performed directly on the input images, as shown in Figure 3. Firstly, we divide each image horizontally into N patches, where the default N is 2. Next, we take M images as a group to randomly shuffle the NM patches in each group, where the default M is 2. Then, every N patches are concatenated horizontally to make new images. Therefore, we produce shuffled images, denoted as { $x^{reg}$ }.

We only feed  $\mathbf{x}^{reg}$  into the online encoder and projector to get frame features, and such an implementation empirically performs better, which shares the same spirit of a multi-cropping strategy [5]. To align all features of permuted images, we unshuffle them (inverting the random shuffle operation) to put the features back in their original position. We denote the resulting features of regularization as  $\mathbf{q}^{reg}$ . The relational contrastive loss with regularization of the permuted images can be written as:

$$\mathcal{L}_{req}(\mathbf{q}, \mathbf{p}, \mathbf{n}) = \mathcal{L}_{re}(\mathbf{q}, \mathbf{p}, \mathbf{n}) + \mathcal{L}_{re}(\mathbf{q}^{reg}, \mathbf{p}, \mathbf{n}), \tag{4}$$

where  $\mathcal{L}_{re}$  is from Equation 3.  $\mathcal{L}_{reg}$  constrains the invariance of the relation under random permutations. The regularization comes from the fact that this constraint on both original and permuted images can force the model not to over-fit the existing contexts.

For the step of horizontal division, since no character position information is available for SSL, we choose equal division as our default setting, which may generate partial characters. And we further study multiple image division strategies (illustrated in Figure 4). 1) Default direct cutting for equal division. 2) Cutting and dropping



Figure 4: Multiple image division strategies.

boundary features. Since the boundary features may correspond to the characters that are cut, we drop these features when calculating the contrastive loss. 3) Using vertical projection to cut. The vertical projection method can cut from the character gap to avoid cutting the character itself.

### 3.3 Hierarchical Relation

Since text images are encoded as sequence features, contrastive learning is applied to the individual elements of the sequence. Considering text words have different granularities in the horizontal direction, we propose a novel hierarchical structure, which maps the features to three levels of frame, subword and word. Thus we conduct hierarchical relational contrastive learning to learn about relations at each level.

To this end, we use three mapping functions to map the feature sequence into three levels, where representations of different objects are encoded. Specifically, the most fine-grained level is called the frame, which usually contains stroke information of only a portion of the letter. We use the identity function as the frame mapping function. The middle level is called the subword, and it usually contains one or more letters, like roots and affixes. We use an avgpooling layer as the subword mapping function, which map features to *T* subwords with T = 4. The highest level is called word, i.e., contains a whole word. And an average function is used as the word mapping function. At each level, we maintain a separate queue of negative features, respectively. We calculate the relational contrastive losses at each level and sum them up:

$$\mathcal{L}_{hier} = \sum_{h \in H} \mathcal{L}_{reg}(\mathbf{q}^h, \mathbf{p}^h, \mathbf{n}^h), \tag{5}$$

where  $H = \{frame, subword, word\}$ . With the proposed hierarchical relational contrastive learning, the model can learn the frameframe, subword-subword and word-word relations simultaneously. At each level, we also perform the regularization as in the Equation 4.

### 3.4 Cross-Hierarchy Relational Consistency

In the previous section, we obtained the features of multiple levels and performed relational contrastive learning within each level.



Figure 5: An illustration of causal graph for our framework.

However, there are semantic relations between features across different levels, which are unexplored. Therefore, we propose consistency constraints to learn the relation between neighboring levels. Our default implementation performs frame-subword and subword-word consistency constraints, and we provide the results of other settings in the experiments. As shown in Figure 2, for frame-subword relations, since the frame and subword features from the same spatial locations (in the same images) show higher similarity in the feature space, we treat them as positives and treat features in other locations as negatives. And the subword-word positives and negatives are determined in the same way. For the frame-subword and subword-word relation, we impose the consistency between the similarity distributions by KL loss as the measure of disagreement:

$$\mathcal{L}_{f2s} = \mathcal{L}_{kl}(\mathbf{q}^{f}, \mathbf{p}^{s}, \mathbf{n}^{s}),$$
  
$$\mathcal{L}_{s2w} = \mathcal{L}_{kl}(\mathbf{q}^{s}, \mathbf{p}^{w}, \mathbf{n}^{w}),$$
 (6)

where superscripts of  $\{f, s, w\}$  denote  $\{frame, subword, word\}$ , respectively. This loss constrains the relation between each feature and its neighboring upper-level feature. Here we take the positives and negatives from the upper level because we consider that upper-level features contain more semantic information than lowerlevel. Finally, the total loss of our relational contrastive learning is formulated as:

$$\mathcal{L}_{total} = \underbrace{\sum_{h \in H} \mathcal{L}_{reg}(\mathbf{q}^h, \mathbf{p}^h, \mathbf{n}^h)}_{\underbrace{h \in H}} + \underbrace{\mathcal{L}_{f2s} + \mathcal{L}_{s2w}}_{\underbrace{\mu \in H}}$$
(7)

Regularized hierarchical relation Cross-hierarchy consistency

#### 3.5 Justification

As shown in Figure 5, we formulate the SSL framework as a causal graph, which contains three nodes: *X*: scene text images, *Y*: robust representations, and *C*: context information.

 $X \rightarrow Y$ : This path indicates that the SSL model can learn robust representation for downstream tasks. For example, The success of recent SeqCLR [1] proves that unsupervised pre-training benefits text representation.

 $C \rightarrow X$ : This path indicates the generation of scene text images – combining specific text under some scenes. Some synthetic dataset, like SynthText [16], is generated in this way.

 $C \rightarrow Y$ : This path indicates that context information prior in the training dataset can help the learning of robust representations. For example, these context-based STR methods usually utilize that for the prediction of occluded text [13, 36, 51].

In the causal graph, *C* confounds *X* and *Y* via the back-door path  $X \rightarrow C \rightarrow Y$ , *i.e.*, learning the representations for STR recognition

only based on the dataset prior. The useful context information can be harmful when the data is out-of-distribution, *i.e.*, with a different context prior. The existence of a back-door path causes a spurious correlation between *X* and *Y*, which prevents the learning of representations. An ideal SSL method should capture the true causality between *X* an *Y*, but the conventional correlation of P(Y|X) fails to do so, as such a spurious correlation is inevitable. Therefore, we instead seek to use the causal intervention P(Y|do(X)), where  $do(\cdot)$ is the random controlled trails. As enumerating all textual relations is impossible, we propose the three modules act as the physical intervention, cutting off the confounding effect. In particular, relational regularization module creates new contexts, hierarchical module breaks the context, and cross-hierarchy module learns the correspondence of local-to-global context. They achieve the  $do(\cdot)$ operation exactly.

### **4 EXPERIMENTS**

Datasets. We conduct our experiments on public datasets of scene text recognition. We train on the synthetic dataset SynthText [16]. SynthText [16] is a synthetically generated dataset. It has 5.5M training data once the word boxes are cropped and filtered for non-alphanumeric characters. For evaluation, we use seven realscene text datasets. IIIT5K(IIIT5K-Words) [32], IC03(ICDAR 2003) [28], IC13(ICDAR 2013) [22] and SVT(Street View Text) [43] are regular text images, which are nearly horizontal. They contain 3000, 867, 1015 and 647 word images for evaluation, respectively. IC15(ICDAR 2015) [21], SVTP(Street View Text Perspective) [35] and CUTE80 [37] are irregular text images. They are mostly perspective text images, and some are blurry or curved. They contain 2077, 645 and 288 word images for evaluation, respectively. Metrics. To evaluate performance, we adopt the metrics of wordlevel accuracy (Acc). Word-level accuracy is the number of correctly predicted words divided by the total number of words.

Network configurations. For the network illustrated in Figure 2, data augmentation module transforms a given image  $X_i$  in a batch of images into two augmented images  $X_i^a, X_i^b \in \mathbb{R}^{C \times H \times W}$ , where we set H as 32, W as 100 and C as 3. We take blocks of transformation (TPS [39]) and feature extraction (ResNet) as the encoder and a two-layer Bidirectional-LSTM (BiLSTM) with 256 hidden units as the projector. For augmented images of both branches, these components extract sequential representations,  $R_i^a, R_i^b \in R^{F \times T}$ , where F is the feature dimension, and T is the number of columns (frames). In our network, we set F = 256 and T = 26 by default. The predictor is a mapping function followed by a fully connected (FC) layer. For the frame level, the mapping function is an identity function. For the subword level, it is an adaptive avgpooling layer that maps frames to *T* subwords with T = 4. For the word level, it is an average function. Finally, the feature dimension for contrastive learning is set to 128.

**Self-Supervised Pre-Training.** The synthetic dataset SynthText [16] without labels is used for pre-training. We employ an SGD optimizer [4] with a constant learning rate scheduler and train the models for 5 epochs. The training hyperparameters are: the batch size as 32, base learning rate as 1.5e-3, weight decay as 1e-4, momentum for SGD optimizer as 0.9. The pre-training experiments are conducted with 4 GPUs.

Table 1: Representation quality. Accuracy(%) is used to evaluate the quality of representation from encoder, and we train a decoder with labeled data on top of frozen encoder which was pretrained on unlabeled images. Our method with different modules added is compared with the previous methods, where "reg" denotes the relational regularization module, "hier" denotes the hierarchical relation module and "con" denotes the cross-hierarchy relational consistency module.

Decodor	Mathad				Scene-Te	ext Datas	set		
Decouel	Methou	IIIT5K	IC03	IC13	SVT	IC15	SVTP	CUTE80	Avg
	SeqCLR [1]	35.70	43.60	43.50	-	-	-	-	-
	PerSec-CNN [27]	37.90	45.70	46.40	-	-	-	-	-
	SeqMoCo w/o KL Loss	41.63	48.21	46.50	25.35	22.05	19.53	22.22	32.21
CTC	SeqMoCo	42.97	51.44	48.37	25.35	23.01	20.62	23.26	33.57
	Ours based on SeqMoCo								
	w/ reg	48.43	58.94	54.98	35.09	26.43	26.82	29.17	39.98
	w/ reg & hier	51.90	61.36	59.01	38.79	30.62	30.08	30.21	43.14
	w/ reg & hier & con	54.83	64.82	60.89	41.58	32.60	34.26	32.64	45.95
	SeqCLR [1]	49.20	63.90	59.30	-	-	-	-	-
	PerSec-CNN [27]	50.70	65.70	61.10	-	-	-	-	-
	SeqMoCo w/o KL Loss	50.97	58.36	55.86	35.55	29.42	28.53	30.56	41.32
Atten	SeqMoCo	51.83	59.75	59.90	37.40	31.73	28.99	32.29	43.13
	Ours based on SeqMoCo								
	w/ reg	56.30	67.70	63.25	41.27	35.05	36.9	37.15	48.23
	w/ reg & hier	59.03	71.51	67.29	46.37	38.32	36.90	36.81	50.89
	w/ reg & hier & con	61.07	72.90	68.77	50.54	40.30	40.16	39.24	53.28

**Feature Representation Evaluation.** For CTC-based and Attentionbased decoders, we inherit the configurations from SeqCLR [1] and PerSec [27]. Following the decoder evaluation [1, 27], during the training for feature representation evaluation, the base encoder is frozen, and we only train a decoder to evaluate the feature representation quality. We employ an Adam optimizer [23] and the one-cycle learning rate scheduler [40] with a maximum learning rate of 5e-4. The training hyperparameters are: the batch size as 256, the number of iterations as 200K, gradient clipping magnitude as 5.

**Fine-Tuning Evaluation**. During the training of fine-tuning evaluation, the base encoder is not frozen, and we fine-tune the whole network. Following [2], we used ST [16] and MJ [20] as the fine-tuning training datasets. An AdaDelta optimizer [53] and constant learning rate scheduler are employed. The training hyperparameters are: the batch size as 192, the number of iterations as 50K, the base learning rate as 1.0, the decay rate of AdaDelta optimizer as 0.95, gradient clipping magnitude as 5.

### 4.1 Representation Quality

For the study of representation quality, the base encoder is unsupervised pre-trained and then frozen. Following SeqCLR [1], We only train a decoder with labeled data on top of it. We compare our results with other CNN-based SSL methods, and the results are shown in Table 1. Because SeqCLR requires a large batchsize, considering the limitations of hardware, we replace the baseline with MoCo. Based on MoCo [17], we add the mapping function of SeqCLR [1] and implement the sequential relational contrastive learning method, denoted as the baseline SeqMoCo. Based on SeqMoCo, we add relational regularization module, hierarchical relation module and cross-hierarchy relational consistency module in turn. The results of representation quality are shown in Table 1. Compared with SeqMoCo without KL loss, SeqMoCo with naive relational contrastive learning achieves limited performance gain. This performance gain is limited by finite dataset relations and suffers from over-fitting due to the lexical dependencies. Compared with SeqMoCo, our method equipped with all three modules further gains an improvement of +12.38% on average for the CTC-based decoder and +10.15% for the Attention-based decoder. Also, the effectiveness of three key modules is verified in this table. It should be noted that SeqMoCo is not a stronger baseline (especially in attention-based decoder). Our performance superiority is from the three proposed modules and not from the KL loss or baseline.

#### 4.2 Fine-tuning

We further unfreeze the parameters of the encoder and fine-tune it with the decoder. Table 2 shows the performance comparison between our RCLSTR and other methods. "Supervised baseline" does not perform self-supervised pre-training, in which parameters are randomly initialized. Compared with SeqMoCo, our method gains an improvement of average performance. Compared with SeqCLR [1] and PerSec [27], our RCLSTR can outperform them in most datasets. These results demonstrate that the image encoder learned by RCLSTR benefits downstream recognition fine-tuning.

#### 4.3 Semi-Supervised Learning

We further evaluate our method by considering semi-supervised settings. We use the same encoders as before, which were pretrained on the unlabeled data, and let the whole network be finetuned using 1% or 10% of the labeled dataset. We use the same randomly selected data for all the experiments.

Decoder	Method	IIIT5K	IC03	IC13	SVT	IC15	SVTP	CUTE80	Avg
	Supervised baseline	84.40	91.81	89.16	83.62	68.05	73.33	71.08	80.21
	SeqCLR [1]	82.90	92.20	87.90	-	-	-	-	-
Atten	PerSec-CNN [27]	84.20	-	88.90	82.40	68.20	73.60	68.40	-
	SeqMoCo	84.40	92.73	89.85	84.54	69.30	74.88	64.81	80.08
	RCLSTR(Ours)	86.03	92.73	91.13	83.15	69.15	74.88	67.94	80.72

Table 2: Fine-tuning results. Accuracy(%) of fine-tuning a pretrained model with labeled data.

Table 3: Semi-supervised results. Accuracy(%) of fine-tuning a pre-training model with 10% and 1% of the labeled data.

Fraction	Method	IIIT5K	IC03	IC13	SVT	IC15	SVTP	CUTE80	Avg
	Supervised baseline	70.90	83.85	79.02	66.46	49.74	50.70	47.04	63.96
10%	SeqMoCo	75.20	87.77	81.87	71.41	54.48	57.98	48.78	68.21
	RCLSTR(Ours)	76.80	87.31	82.86	72.64	55.31	60.16	54.01	69.87
	Supervised baseline	64.57	80.05	74.09	60.59	42.92	45.12	37.28	57.80
1%	SeqMoCo	65.57	81.55	74.98	62.91	48.86	53.95	37.98	60.83
	RCLSTR(Ours)	73.73	86.51	81.77	72.80	51.35	58.61	45.99	67.25

Table 4: The decoder evaluation performance with ViT-based encoder-decoder architecture. Our RCLSTR method with different modules added is compared with the SeqMoCo, where "reg" denotes the relational regularization module, "hier" denotes the hierarchical relation module and "con" denotes the cross-hierarchy relational consistency module.

En ester Deseter	M. (1 1	M - 1-1			Sc	ene-Tex	t Dataset	ts		
Encoder-Decoder	Method	Modules	IIIT5K	IC03	IC13	SVT	IC15	SVTP	CT80	Avg
SATRN <sub>small</sub> (ViT-Based)	SeqMoCo	-	62.57	75.55	70.44	63.83	44.1	46.05	35.76	56.90
		w/ reg	74.43	83.51	77.83	70.48	53.83	54.73	46.53	65.91
	RCLSTR	w/ reg & hier	78.23	86.16	80.89	75.43	58.59	61.24	54.51	70.72
		w/ reg & hier & con	78.10	87.31	82.46	74.34	59.65	60.16	55.21	71.03

Table 5: The decoder evaluation performance on Chinese and handwritten datasets.

Mathad	Chinaga Datagat	Handwritten Dataset			
Method	Chillese Dataset	IAM	CVL		
SeqMoCo	47.56	56.16	77.80		
RCLSTR	55.70	62.88	85.92		

Table 3 compares our method with SeqMoCo and supervised baseline training. As can be seen, RCLSTR achieves better performance on average under different amount of labeled data. Our method succeeds in significantly improving the results of SeqMoCo. Compared with SeqMoCo, our method gains an improvement of +1.66% on average for 10% labeled data and +6.42% for 1% labeled data. These results verify that the representations learned by RCLSTR benefit the learning from insufficient data.

#### 4.4 Results on ViT-based Encoder-Decoder

We evaluate our RCLSTR method on the ViT-based encoder-decoder architecture, and the results are summarized in Table 4. We choose the small version of SATRN [26] to verify the effectiveness of our method on the ViT encoder. We perform SeqMoCo and RCLSTR pretraining, and freeze encoder to perform decoder evaluation. And we add three modules in turn. The performance gain verifies that our three modules can also be effective for the ViT-based encoder.

### 4.5 Results on More Languages and Types of Text Image Datasets

The condition of using our method only assumes that the text is horizontal, so it is also useful for other languages (*e.g.*, Chinese) and other fonts (*e.g.*, handwritten). For the Chinese document dataset [52], we perform SeqMoCo and RCLSTR pre-training, and freeze encoder to perform decoder evaluation, the accuracies are summarized in Table 5. RCLSTR achieved superior performance on Chinese datasets. Since Chinese Text Images have left-to-right structures and horizontal multi-grained hierarchies, RCLSTR can also facilitate self-supervised learning of their features.

We further evaluate our RCLSTR method on the handwritten datasets, comparing its performance with SeqMoCo. We consider the English handwritten datasets IAM [31] and CVL [24], and the results are summarized in Table 5. Compared with SeqMoCo, RCLSTR achieves better performance on these two datasets and gains an improvement of +6.72% for IAM and +8.12% for CVL. Although handwritten fonts have a certain irregularity, our RCLSTR can also utilize their horizontal and multi-hierarchical structure to facilitate feature learning.

Table 6: Ablations. (a) Analysis of the setting for hierarchies. Without adding other modules, we try the settings of different combinations for hierarchies. (b) Effect of image division strategies. \*: Direct cutting is the default setting of RCLSTR. (c) Effect of consistency constraints. (d) Ablation on hierarchical consistency loss functions.

	Hierarchies	IIIT5K	IC03	IC13	SVT	IC15	SVTP	CUTE80	Avg
(a)	w/ subword & frame	50.07	62.17	58.82	36.01	30.52	27.29	28.82	41.96
	w/ subword & word	56.30	66.90	64.63	43.59	35.19	35.50	35.07	48.17
	w/ subword & word & frame	58.80	68.51	66.21	47.45	37.65	37.36	39.24	50.75
	Image Division								
(b)	Direct cutting*	61.07	72.90	68.77	50.54	40.30	40.16	39.24	53.28
	Dropping boundary features	61.33	74.51	69.56	49.00	38.85	38.14	38.54	52.85
	Vertical projection	60.97	72.66	68.08	53.17	40.25	40.31	39.24	53.53
	Cross-Hierarchy Consistency								
(c)	w/ subword-word	59.47	70.24	66.60	47.30	36.11	38.14	32.29	50.02
	w/ frame-subword	59.60	71.28	67.00	50.08	38.81	41.24	38.19	52.31
	w/ subword-word & frame-subword	61.07	72.90	68.77	50.54	40.30	40.16	39.24	53.28
	Cross-Hierarchy Consistency Loss Fu	nction							
(d)	InfoNCE	60.60	73.82	70.15	48.38	39.38	40.16	37.15	52.81
	KL	61.07	72.90	68.77	50.54	40.30	40.16	39.24	53.28
	RE	59.93	71.05	67.39	50.08	38.42	39.69	36.46	51.86





#### 4.6 Visualization

In Figure 6, we use t-SNE [19] to visualize the final features of IIIT5K [32] images corresponding to two methods, *i.e.*, SeqMoCo (baseline) and the proposed RCLSTR, in which features for attentionbased decoder are visualized by attaching character labels to frame features. As can be seen, our method mines intra-class relation to cluster characters of the same class. Besides, our method also mines the inter-class relations, where the clusters with similar-looking characters (*e.g.*, I&J and D&O) are close.

### **5 ABLATION STUDY**

First, we analyze the setting for hierarchies. Without adding other modules, we try the settings of different combinations for hierarchies, and the results are shown in Table 6 (a). We can observe that the setting of subword and word gains higher performance than subword and frame. The best performance is achieved by learning at all three levels. In the following ablation, we used all three levels as the default setting to explore variants of other modules.

**Effect of image division strategies.** We study how different image division strategies affect the effectiveness of relational regularization. The results under different image division strategies (as shown in Figure 4) are summarized in Table 6 (b). Under the condition of no available character positions for SSL, the process of direct division and concatenation creates richer context, but meaningless

images (such as partial characters from non-ideal image division) are generated. There is a trade-off between context diversity and non-ideal image division. Since the goal of relational regularization is to avoid over-fitting the original context, the effectiveness mainly comes from more diversity of contexts, and it is insensitive to the non-ideal image division boundaries. We find that dropping the boundary features has a similar performance. And the vertical projection method to avoid cutting character also has no significant performance gap.

**Effect of consistency constraints.** In Table 6 (c), we impose subword-word and frame-subword consistency constraints separately, and we find that frame-subword consistency brings the most performance gain, which indicates that the more granular consistency is more useful for learning text representations.

**Ablation on hierarchical consistency loss functions.** By default, we use KL-divergence loss to constrain the consistency between hierarchies. As shown in Table 6 (d), we test other consistency loss functions, such as InfoNCE loss in Equation 1 and relational loss (RE) in Equation 3. For cross-hierarchy relations, local and global features do not have absolute consistency and only have relative similarities. Thus KL loss is better than InfoNCE and RE (*i.e.* InfoNCE + KL) loss.

#### **6** CONCLUSIONS

This work proposes a novel framework, Relational Contrastive Learning for Scene Text Recognition (RCLSTR). To take advantage of contextual priors in STR, we argue that contextual information can be interpreted as the relations of textual primitives and utilized in an unsupervised way. In this framework, the relations in text images are fully explored by three modules. The relational regularization module is proposed to enrich the intra- and inter-image context relations. The hierarchical relational module for relational contrastive learning can capture multi-granularity representations. Additionally, the cross-hierarchy relational consistency module is designed for the interactions across different hierarchical levels in scene text images. Experiments on representation quality verify the superiority of our RCLSTR method.

#### ACKNOWLEDGMENTS

This work was supported in part by NSFC 62171282, Shanghai Municipal Science and Technology Major Project (2021SHZDZX0102), 111 project BP0719010, STCSM 22DZ2229005.

#### REFERENCES

- [1] Aviad Aberdam, Ron Litman, Shahar Tsiper, Oron Anschel, Ron Slossberg, Shai Mazor, R Manmatha, and Pietro Perona. 2021. Sequence-to-sequence contrastive learning for text recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, Piscataway, NJ, USA, 15302–15312.
- [2] Jeonghun Baek, Geewook Kim, Junyeop Lee, Sungrae Park, Dongyoon Han, Sangdoo Yun, Seong Joon Oh, and Hwalsuk Lee. 2019. What is wrong with scene text recognition model comparisons? dataset and model analysis. In Proceedings of the IEEE/CVF international conference on computer vision. IEEE Computer Society, Los Alamitos, CA, USA, 4715–4723.
- [3] Jeonghun Baek, Yusuke Matsui, and Kiyoharu Aizawa. 2021. What if we only use real datasets for scene text recognition? toward scene text recognition with fewer labels. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. IEEE COMPUTER SOC, 10662 LOS VAQUEROS CIRCLE, PO BOX 3014, LOS ALAMITOS, CA 90720-1264 USA, 3113–3122.
- [4] Léon Bottou. 2010. Large-scale machine learning with stochastic gradient descent. In Proceedings of COMPSTAT'2010. Springer, Berlin, German, 177–186.
- [5] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. 2020. Unsupervised learning of visual features by contrasting cluster assignments. Advances in Neural Information Processing Systems 33 (2020), 9912–9924.
- [6] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. 2020. Unsupervised learning of visual features by contrasting cluster assignments. Advances in Neural Information Processing Systems 33 (2020), 9912–9924.
- [7] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*. PMLR, JMLR, 1269 LAW ST, SAN DIEGO, CA, UNITED STATES, 1597–1607.
- [8] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. 2020. Improved Baselines with Momentum Contrastive Learning. arXiv:2003.04297 [cs.CV]
- [9] Xinlei Chen and Kaiming He. 2021. Exploring simple siamese representation learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. IEEE COMPUTER SOC, 10662 LOS VAQUEROS CIRCLE, PO BOX 3014, LOS ALAMITOS, CA 90720-1264 USA, 15750–15758.
- [10] Zhanzhan Cheng, Fan Bai, Yunlu Xu, Gang Zheng, Shiliang Pu, and Shuigeng Zhou. 2017. Focusing attention: Towards accurate text recognition in natural images. In Proceedings of the IEEE international conference on computer vision. IEEE, 345 E 47TH ST, NEW YORK, NY 10017 USA, 5076–5084.
- [11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. arXiv:2010.11929 [cs.CV]
- [12] Shancheng Fang, Hongtao Xie, Yuxin Wang, Zhendong Mao, and Yongdong Zhang. 2021. Read like humans: Autonomous, bidirectional and iterative language modeling for scene text recognition. In *Proceedings of the IEEE/CVF Conference* on Computer Vision and Pattern Recognition. IEEE COMPUTER SOC, 10662 LOS VAQUEROS CIRCLE, PO BOX 3014, LOS ALAMITOS, CA 90720-1264 USA, 7098– 7107.
- [13] Shancheng Fang, Hongtao Xie, Yuxin Wang, Zhendong Mao, and Yongdong Zhang. 2021. Read like humans: Autonomous, bidirectional and iterative language modeling for scene text recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. IEEE COMPUTER SOC, 10662 LOS VAQUEROS CIRCLE, PO BOX 3014, LOS ALAMITOS, CA 90720-1264 USA, 7098– 7107.
- [14] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*. ACM, New York, NY, United States, 369–376.
- [15] Jean-Bastien Grill, Florian Štrub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. 2020. Bootstrap your own latent-a new approach to self-supervised learning. Advances in neural information processing systems 33 (2020), 21271–21284.
- [16] Ankush Gupta, Andrea Vedaldi, and Andrew Zisserman. 2016. Synthetic data for text localisation in natural images. In Proceedings of the IEEE conference on computer vision and pattern recognition. IEEE, 345 E 47TH ST, NEW YORK, NY 10017 USA, 2315–2324.

- [17] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. IEEE Computer Society, Los Alamitos, CA, USA, 9729–9738.
- [18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer* vision and pattern recognition. IEEE, 345 E 47TH ST, NEW YORK, NY 10017 USA, 770–778.
- [19] Geoffrey Hinton and Sam Roweis. 2002. Stochastic Neighbor Embedding. In Proceedings of the 15th International Conference on Neural Information Processing Systems (NIPS'02). MIT Press, Cambridge, MA, USA, 857–864.
- [20] Max Jaderberg, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. 2014. Synthetic Data and Artificial Neural Networks for Natural Scene Text Recognition. arXiv:1406.2227 [cs.CV]
- [21] Dimosthenis Karatzas, Lluis Gomez-Bigorda, Anguelos Nicolaou, Suman Ghosh, Andrew Bagdanov, Masakazu Iwamura, Jiri Matas, Lukas Neumann, Vijay Ramaseshan Chandrasekhar, Shijian Lu, et al. 2015. ICDAR 2015 competition on robust reading. In 2015 13th international conference on document analysis and recognition (ICDAR). IEEE, IEEE, 345 E 47TH ST, NEW YORK, NY 10017 USA, 1156–1160.
- [22] Dimosthenis Karatzas, Faisal Shafait, Seiichi Uchida, Masakazu Iwamura, Lluis Gomez i Bigorda, Sergi Robles Mestre, Joan Mas, David Fernandez Mota, Jon Almazan Almazan, and Lluis Pere De Las Heras. 2013. ICDAR 2013 robust reading competition. In 2013 12th international conference on document analysis and recognition. IEEE, IEEE, 345 E 47TH ST, NEW YORK, NY 10017 USA, 1848–1493.
- [23] Diederik P. Kingma and Jimmy Ba. 2017. Adam: A Method for Stochastic Optimization. arXiv:1412.6980 [cs.LG]
- [24] Florian Kleber, Stefan Fiel, Markus Diem, and Robert Sablatnig. 2013. Cvldatabase: An off-line database for writer retrieval, writer identification and word spotting. In 2013 12th international conference on document analysis and recognition. IEEE, IEEE, 345 E 47TH ST, NEW YORK, NY 10017 USA, 560–564.
- [25] Junyeop Lee, Sungrae Park, Jeonghun Baek, Seong Joon Oh, Seonghyeon Kim, and Hwalsuk Lee. 2020. On recognizing texts of arbitrary shapes with 2D selfattention. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. IEEE COMPUTER SOC, 10662 LOS VAQUEROS CIRCLE, PO BOX 3014, LOS ALAMITOS, CA 90720-1264 USA, 546–547.
- [26] Junyeop Lee, Sungrae Park, Jeonghun Baek, Seong Joon Oh, Seonghyeon Kim, and Hwalsuk Lee. 2020. On recognizing texts of arbitrary shapes with 2D selfattention. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. IEEE COMPUTER SOC, 10662 LOS VAQUEROS CIRCLE, PO BOX 3014, LOS ALAMITOS, CA 90720-1264 USA, 546–547.
- [27] Hao Liu, Bin Wang, Zhimin Bao, Mobai Xue, Sheng Kang, Deqiang Jiang, Yinsong Liu, and Bo Ren. 2022. Perceiving Stroke-Semantic Context: Hierarchical Contrastive Learning for Robust Scene Text Recognition. In AAAI. AAAI, 2275 E BAYSHORE RD, STE 160, PALO ALTO, CA 94303 USA, 1702–1710.
- [28] Simon M Lucas, Alex Panaretos, Luis Sosa, Anthony Tang, Shirley Wong, Robert Young, Kazuki Ashida, Hiroki Nagai, Masayuki Okamoto, Hiroaki Yamamoto, et al. 2005. ICDAR 2003 robust reading competitions: entries, results, and future directions. International Journal of Document Analysis and Recognition (IJDAR) 7, 2 (2005), 105–122.
- [29] Canjie Luo, Lianwen Jin, and Jingdong Chen. 2022. SimAN: Exploring Self-Supervised Representation Learning of Scene Text via Similarity-Aware Normalization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. IEEE COMPUTER SOC, 10662 LOS VAQUEROS CIRCLE, PO BOX 3014, LOS ALAMITOS, CA 90720-1264 USA, 1039–1048.
- [30] Pengyuan Lyu, Chengquan Zhang, Shanshan Liu, Meina Qiao, Yangliu Xu, Liang Wu, Kun Yao, Junyu Han, Errui Ding, and Jingdong Wang. 2022. MaskOCR: Text Recognition with Masked Encoder-Decoder Pretraining. arXiv:2206.00311 [cs.CV]
- [31] U-V Marti and Horst Bunke. 2002. The IAM-database: an English sentence database for offline handwriting recognition. *International Journal on Document Analysis and Recognition* 5 (2002), 39–46.
- [32] Anand Mishra, Karteek Alahari, and CV Jawahar. 2012. Scene text recognition using higher order language priors. In *BMVC-British machine vision conference*. BMVA, B M V A PRESS, 49A ELMSIDE ONSLOW VILLAGE, GUILDFORD, SUR-REY GU2 5SX, ENGLAND, 127.1–127.11.
- [33] Jovana Mitrovic, Brian McWilliams, Jacob Walker, Lars Buesing, and Charles Blundell. 2020. Representation Learning via Invariant Causal Mechanisms. arXiv:2010.07922 [cs.LG]
- [34] Massimiliano Patacchiola and Amos J Storkey. 2020. Self-supervised relational reasoning for representation learning. Advances in Neural Information Processing Systems 33 (2020), 4003–4014.
- [35] Trung Quy Phan, Palaiahnakote Shivakumara, Shangxuan Tian, and Chew Lim Tan. 2013. Recognizing text with perspective distortion in natural scenes. In Proceedings of the IEEE International Conference on Computer Vision. IEEE, 345 E 47TH ST, NEW YORK, NY 10017 USA, 569–576.

- [36] Zhi Qiao, Yu Zhou, Dongbao Yang, Yucan Zhou, and Weiping Wang. 2020. Seed: Semantics enhanced encoder-decoder framework for scene text recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. IEEE Computer Society, Los Alamitos, CA, USA, 13528–13537.
- [37] Anhar Risnumawan, Palaiahankote Shivakumara, Chee Seng Chan, and Chew Lim Tan. 2014. A robust arbitrary text detection system for natural scene images. *Expert Systems with Applications* 41, 18 (2014), 8027–8048.
- [38] Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural Machine Translation of Rare Words with Subword Units. arXiv:1508.07909 [cs.CL]
- [39] Baoguang Shi, Xinggang Wang, Pengyuan Lyu, Cong Yao, and Xiang Bai. 2016. Robust scene text recognition with automatic rectification. In Proceedings of the IEEE conference on computer vision and pattern recognition. IEEE, 345 E 47TH ST, NEW YORK, NY 10017 USA, 4168–4176.
- [40] Leslie N Smith and Nicholay Topin. 2019. Super-convergence: Very fast training of neural networks using large learning rates. In Artificial intelligence and machine learning for multi-domain operations applications, Vol. 11006. SPIE, SPIE-INT SOC OPTICAL ENGINEERING, 1000 20TH ST, PO BOX 10, BELLINGHAM, WA 98227-0010 USA, 369–386.
- [41] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2019. Representation Learning with Contrastive Predictive Coding. arXiv:1807.03748 [cs.LG]
- [42] Zhaoyi Wan, Jielei Zhang, Liang Zhang, Jiebo Luo, and Cong Yao. 2020. On vocabulary reliance in scene text recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE Computer Society, Los Alamitos, CA, USA, 11425–11434.
- [43] Kai Wang, Boris Babenko, and Serge Belongie. 2011. End-to-end scene text recognition. In 2011 International conference on computer vision. IEEE, IEEE, 345 E 47TH ST, NEW YORK, NY 10017 USA, 1457–1464.
- [44] Xiao Wang and Guo-Jun Qi. 2022. Contrastive learning with stronger augmentations. IEEE Transactions on Pattern Analysis and Machine Intelligence 45 (2022), 5549–5560.
- [45] Yuxin Wang, Hongtao Xie, Shancheng Fang, Jing Wang, Shenggao Zhu, and Yongdong Zhang. 2021. From two to one: A new scene text recognizer with visual language modeling network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. IEEE, 345 E 47TH ST, NEW YORK, NY 10017 USA, 14194–14203.
- [46] Chen Wei, Huiyu Wang, Wei Shen, and Alan Yuille. 2020. CO2: Consistent Contrast for Unsupervised Visual Representation Learning. arXiv:2010.02217 [cs.CV]
- [47] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. 2018. Unsupervised feature learning via non-parametric instance discrimination. In Proceedings of the IEEE conference on computer vision and pattern recognition. IEEE, 345 E 47TH ST, NEW YORK, NY 10017 USA, 3733–3742.
- [48] Xudong Xie, Ling Fu, Zhifei Zhang, Zhaowen Wang, and Xiang Bai. 2022. Toward Understanding WordArt: Corner-Guided Transformer for Scene Text Recognition. In Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVIII. Springer, SPRINGER INTERNATIONAL PUBLISHING AG, GEWERBESTRASSE 11, CHAM, CH-6330, SWITZERLAND, 303–321.
- [49] Zhenda Xie, Zheng Zhang, Yue Cao, Yutong Lin, Jianmin Bao, Zhuliang Yao, Qi Dai, and Han Hu. 2022. Simmim: A simple framework for masked image modeling. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. IEEE COMPUTER SOC, 10662 LOS VAQUEROS CIRCLE, PO BOX 3014, LOS ALAMITOS, CA 90720-1264 USA, 9653–9663.
- [50] Mingkun Yang, Minghui Liao, Pu Lu, Jing Wang, Shenggao Zhu, Hualin Luo, Qi Tian, and Xiang Bai. 2022. Reading and writing: Discriminative and generative modeling for self-supervised text recognition. In *Proceedings of the 30th ACM International Conference on Multimedia*. ACM, New York, NY, USA, 4214–4223.
- [51] Deli Yu, Xuan Li, Chengquan Zhang, Tao Liu, Junyu Han, Jingtuo Liu, and Errui Ding. 2020. Towards accurate scene text recognition with semantic reasoning networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. IEEE Computer Society, Los Alamitos, CA, USA, 12113– 12122.
- [52] Haiyang Yu, Jingye Chen, Bin Li, Jianqi Ma, Mengnan Guan, Xixi Xu, Xiaocong Wang, Shaobo Qu, and Xiangyang Xue. 2022. Benchmarking Chinese Text Recognition: Datasets, Baselines, and an Empirical Study. arXiv:2112.15093 [cs.CV]
- [53] Matthew D. Zeiler. 2012. ADADELTA: An Adaptive Learning Rate Method. arXiv:1212.5701 [cs.LG]
- [54] Xinyun Zhang, Binwu Zhu, Xufeng Yao, Qi Sun, Ruiyu Li, and Bei Yu. 2022. Context-based contrastive learning for scene text recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. AAAI, 2275 E BAYSHORE RD, STE 160, PALO ALTO, CA 94303 USA, 3353–3361.
- [55] Mingkai Zheng, Shan You, Fei Wang, Chen Qian, Changshui Zhang, Xiaogang Wang, and Chang Xu. 2021. Ressl: Relational self-supervised learning with weak augmentation. Advances in Neural Information Processing Systems 34 (2021), 2543-2555.

## A VISUALIZATION RESULTS OF REGULARIZED SAMPLES

In Figure 7, we show some samples of randomly permuted images in the relational regularization module. The permutation generates new word images with new contexts. The meaningful generated images are an extension of the contexts. However, non-ideal image divisions (such as partial characters or unaligned boundaries) are generated. As we analyzed in the ablation section, there is a trade-off between context diversity and non-ideal image division. Since relational regularization aims to avoid over-fitting the original context, the effectiveness mainly comes from more diversity of contexts, and our ablation finds that ideal division is not necessary for the relational regularization module.

# B MORE RESULTS ON CROSS-HIERARCHY CONSISTENCY

In Table 7, we provide the performance of adding frame-word consistency constraints. We find that there is no significant performance improvement with adding frame-word consistency constraint. Considering the semantic similarity between the frame and word levels is relatively distant, the consistency between the two might be not so important.

### C MEMORY BANK SIZE

In Table 8, we compare the average accuracy(%) of decoder evaluation under different settings of memory bank size (number of negatives). As can be seen, the performance is positively correlated to memory bank size, which is similar to MoCo. The default memory size setting of RCLSTR is 65536, and all other experiments are performed with this setting.

### D COMPUTATIONAL COST & MODEL SIZE

**Computational Cost.** In Table 9, we compare the per-iteration forwarding time(s) for pre-training SeqMoCo and RCLSTR on the same training machine. The results show that RCLSTR needs only a tiny amount of extra computations. It should be noted that this time may be different on other kinds of hardware, and our purpose is to illustrate the relative computational cost.

**Model Size.** We build the encoder and decoder following SeqCLR, so the model sizes of SeqCLR and ours are fair. The Persec decoder is the same as ours. Because some parameter nums are unreleased, Persec-CNN has an unknown size for the encoder. It should be noted that our model also exceeds Persec-ViT(as shown in Table 10), which uses a stronger encoder than Persec-CNN. So our performance is better than PerSec. To sum up, RCLSTR is relatively fair in terms of model size, and our performance improvement is not from a larger model size.

## **E** MORE IMPLEMENTATION DETAILS

## E.1 Text Recognition Scheme

We use the "encoder-decoder" text recognition network. In the encoder, we use a transformation and a feature extraction network. The decoder can be a CTC-based decoder or attention-based decoder. The transformation stage transforms a cropped image into a normalized image. Because the input image may contain text in a non-axis aligned layout, as often occurs in scene text images, the transformation is necessary. We follow [1] and utilize the Thin Plate Spine (TPS) transformation [39], which is a variant of the spatial transformer network [39]. This transformation first uses a CNN of 4 layers to detect a pre-defined number of fiducial points at the top and bottom of the text region. Then, a smooth spline interpolation is applied between the obtained points to map the predicted textual region to a constant pre-defined size.

In the feature extraction stage, we use a ResNet [18] of 29 layers, which is the same as [1]. For Bidirectional-LSTM (BiLSTM), we follow [1] to use 2 layers of BiLSTM, and the hidden size is 256. We also follow [1] to build Connectionist Temporal Classification (CTC) based and attention-based decoder to decode the predictions from the sequential features.

### E.2 Data Augmentation

For the data augmentation, we follow SeqCLR [1], and our augmentation consists of a random subset of the linear contrast, blur, sharpen, crop, perspective transform and piecewise affine. The augmentation procedure is implemented using the imgaug augmentation package, which is used to augment each image twice for self-supervised learning. The pseudo-code for augmentation written with imgaug package is as shown in Algorithm 1.

# E.3 Self-Supervised Pre-Training

The goal of self-supervised pre-training is to pre-train the weights of the feature encoder. We use TPS [39] and ResNet [18] of 29 layers as the feature encoder. And we use BiLSTM as a projector, and the hidden size is 256. The projector is followed by mapping functions and FC layers that act as predictors. In the pre-training stage, we use a pre-trained TPS module and freeze its weight. The projector and predictors are auxiliary networks that are discarded entirely after the pre-training stage. After pre-training, we only use the pre-trained weights of the feature encoder.

## E.4 Feature Representation Evaluation

During the training for feature representation evaluation, the base encoder is frozen, and we only train a decoder. For CTC-based and Attention-based decoders, we inherit the configurations from Seq-CLR [1]. The paper has illustrated the training and testing settings.

## E.5 Fine-Tuning Evaluation

For the training of fine-tuning evaluation, the base encoder is not frozen, and we fine-tune the whole network. We inherit the configurations of CTC-based and Attention-based decoders from SeqCLR [1]. The training and testing settings are illustrated in the paper.

## E.6 Semi-Supervised Evaluation

During the training of semi-supervised evaluation, the base encoder is not frozen, and we fine-tune the whole network. Following [2], we used ST [16] and MJ [20] as the fine-tuning training data sets and used 1% or 10% of labeled data of them. An AdaDelta optimizer [53] and constant learning rate scheduler are employed. The training



Figure 7: Visualization results of the regularized samples. In relational regularization module, we randomly permute the image patches, which generates images with new contexts.

Table 7: More results of cross-hierarchy consistency. "Cross-Hierarchy Consistency" represents different combinations of consistency constraints between hierarchies. Here we supplement the result with adding frame-word consistency constraint. All results are on representation qualities (decoder evaluation accuracy).

Cross-Hierarchy Consistency	IIIT5K	IC03	IC13	SVT	IC15	SVTP	CUTE80	Avg
w/ subword-word & frame-subword	61.07	72.90	68.77	50.54	40.30	40.16	39.24	53.28
w/ subword-word & frame-subword & frame-word	61.40	74.39	68.57	50.85	39.09	40.62	39.24	53.45

	Table 8: Con	iparison (	of different	memory	<sup>7</sup> bank siz
--	--------------	------------	--------------	--------	-----------------------

-	Size	4096	16384	65536	
-	Avg Acc	49.65	52.13	53.28	
Tabl	e 9: Per-ite	eration	forware	ding tim	e(s).
	Method	SeqMo	oCo RC	CLSTR	
	Time(s)	0.14	8 0	.178	
Table 10	Addition	al comp	parison	with Per	sec-ViT.
Decoder	Method		IIIT5K	IC03	IC13
Atten	Persec-V	ïT [27]	52.30	66.60	62.30

hyperparameters are: the batch size as 192, the number of iterations as 5K, the base learning rate as 1.0, the decay rate of AdaDelta optimizer as 0.95, gradient clipping magnitude as 5.

61.07

72.90

68.77

#### E.7 Algorithm Pseudocode

RCLSTR

Algorithm 2 provides the pseudo-code of RCLSTR for the pretraining task. As shown in the pseudo-code, we use random permutation for relational regularization. The regularized relational contrastive losses at the frame, subword and word level are calculated separately. Besides, the cross-hierarchy consistency losses for frame-to-subword and subword-to-word are proposed. The final loss is the sum of the regularized relational loss at each level and the relational consistency losses across levels.

#### E.8 Evaluation Variance

We find that the evaluation results may have some tiny variance in the experiment. The evaluation variance may be due to incomplete evaluation protocols. We look forward to future community work to complete the evaluation protocol of this field.

#### Algorithm 1 Pseudocode of data augmentation.

from imgaug import augmenters as iaa iaa.Sequential([iaa.SomeOf((1, 5), Ε iaa.LinearContrast((0.5, 1.0)), iaa.GaussianBlur((0.5, 1.5)), iaa.Crop(percent=((0, 0.4), (0, 0), (0, 0.4) (0, 0.0)), keep\_size=True), iaa.Crop(percent=((0, 0.0), (0, 0.02), (0, 0), (0, 0.02)), keep\_size=True), iaa.Sharpen(alpha=(0.0, 0.5), lightness=(0.0, 0.5)), iaa.PiecewiseAffine(scale=(0.02, 0.03), mode='edge'), iaa.PerspectiveTransform( scale=(0.01, 0.02)), ], random\_order=True)])

```
# f_q, f_k: online and momentum networks for query and key
# queue: dictionary as a queue of K keys (CxK)
# m: momentum; t: temperature
f_k.params = f_q.params # initialize
for x in loader:
   x_q = aug(x) # a randomly augmented version
   x_k = aug(x) # another randomly augmented version
   # relational regularization of random permutation
   x_q_reg = aug_pm(x_q)
   frame_q, subword_q, word_q = f_q.forward(x_q)
   frame_k, subword_k, word_k = f_k.forward(x_k)
   frame_q_reg, subword_q_reg, word_q_reg = f_q.forward(x_q_reg) # queries of regularized samples
   # calculate relational contrastive loss for each level
   L_frame = relational_loss(frame_q, frame_k, frame_queue)
   L_frame_reg = relational_loss(frame_q_reg, frame_k, frame_queue)
   L_subword = relational_loss(subword_q, subword_k, subword_queue)
   L_subword_reg = relational_loss(subword_q_reg, subword_k, subword_queue)
   L_word = relational_loss(word_q, word_k, word_queue)
   L_word_reg = relational_loss(word_q_reg, word_k, word_queue)
   # calculate cross-hierarchy consistency loss
   L_f2s = relational_loss(frame_q, subword_k, subword_queue)
   L_s2w = relational_loss(subword_q, word_k, word_queue)
   # total loss
   loss = L_frame + L_frame_reg + L_subword + L_subword_reg + L_word + L_word_reg + L_f2s + L_s2w
   # SGD update: guery network
   loss.backward()
   update(f_q.params)
   # momentum update: key network
   f_k.params = m*f_k.params+(1-m)*f_q.params
   # update dictionary, enqueue and dequeue the current minibatch
   enqueue_dequeue(frame_queue, frame_k)
   enqueue_dequeue(subword_queue, subword_k)
   enqueue_dequeue(word_queue, word_k)
def relational_loss(q, k, queue):
   # positive logits: Nx1
   l_pos = bmm(q.view(N,1,C), k.view(N,C,1))
   # negative logits: NxK
   l_neg = mm(q.view(N,C), queue.view(C,K))
   # logits: Nx(1+K)
   logits = cat([l_pos, l_neg], dim=1)
   # InfoNCE contrastive loss
   labels = zeros(N) # positives are the 0-th
   InfoNCE_loss = CrossEntropyLoss(logits/t, labels)
   # KL contrastive loss
   similarity_q = mm(q.view(N,C), queue.view(C,K))
   similarity_k = mm(k.view(N,C), queue.view(C,K))
   KL_loss = 0.5 * kl(similarity_q, similarity_k) + 0.5 * kl(similarity_k, similarity_q)
   return InfoNCE_loss + KL_loss
```