SCOB: Universal Text Understanding via Character-wise Supervised Contrastive Learning with Online Text Rendering for Bridging Domain Gap

Daehee Kim[1†], Yoonsik Kim[1†*], DongHyun Kim[1], Yumin Lim[2], Geewook Kim[1], Taeho Kil[1]
[1] NAVER Cloud AI [2] Seoul National University
{daehee.k,yoonsik.kim90}@navercorp.com

Abstract

Inspired by the great success of language model (LM)-based pre-training, recent studies in visual document understanding have explored LM-based pre-training methods for modeling text within document images. Among them, pre-training that reads all text from an image has shown promise, but often exhibits instability and even fails when applied to broader domains, such as those involving both visual documents and scene text images. This is a substantial limitation for real-world scenarios, where the processing of text image inputs in diverse domains is essential. In this paper, we investigate effective pre-training tasks in the broader domains and also propose a novel pre-training method called SCOB that leverages character-wise supervised contrastive learning with online text rendering to effectively pre-train document and scene text domains by bridging the domain gap. Moreover, SCOB enables weakly supervised learning, significantly reducing annotation costs. Extensive benchmarks demonstrate that SCOB generally improves vanilla pre-training methods and achieves comparable performance to state-of-the-art methods. Our findings suggest that SCOB can be served generally and effectively for read-type pre-training methods. The code will be available at https://github.com/naver-ai/scob.

1. Introduction

Visually-situated language, which encompasses a mixture of texts and visual objects such as documents, tables, infographics, and user interfaces, is now ubiquitous in modern human civilization. Accordingly, automatically reading and understanding visually-situated language with machine learning systems is considered commercially valuable and challenging. Considering the usability and training convenience for machine learning systems, Visual Document Understanding (VDU) and Scene Text Understanding (STU) tasks have been separately studied for visually-situated language. VDU mainly handles visually scanned or binarized document images, whereas STU takes images in real-world and dynamic environments as input, as shown in Figure 2.

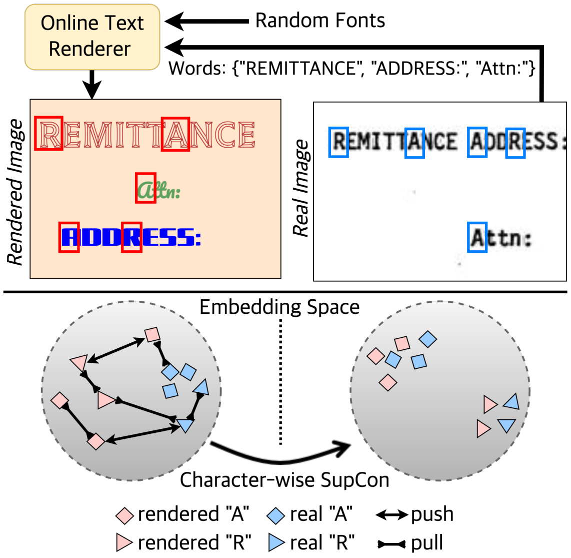

Figure 1. Our proposed SCOB is applicable to pre-training tasks of generative text understanding models, including *text-read* and *OCR-read*. (**Top**) Our renderer generates images at the word-level with diverse fonts and sizes. Shapes of "A"s and "R"s are little different respectively in the real image (blue box), but their shapes vary significantly in the rendered image (red box). (**Bottom**) By applying the character-wise supervised contrastive loss, "A"s and "R"s are clustered respectively and the clusters of "A" and "R" push away each other in the embedding space.

In the context of VDU, Donut [32] has been proposed as a sequence generation model, which pre-trains a *text-read* task of reading all texts in raster scan order from an image, as illustrated in Figure 2. Meanwhile, Pix2seq [7] is an image-to-sequence model that extends to the object detection task by gridding images and using coordinate tokens on the grid. These recent studies [7, 32, 42] suggest that prompt control in a sequence generation approach can successfully

[†] Equal contribution
[*] Corresponding author and Project leader
[2] This work was done while the author was at NAVER Cloud AI

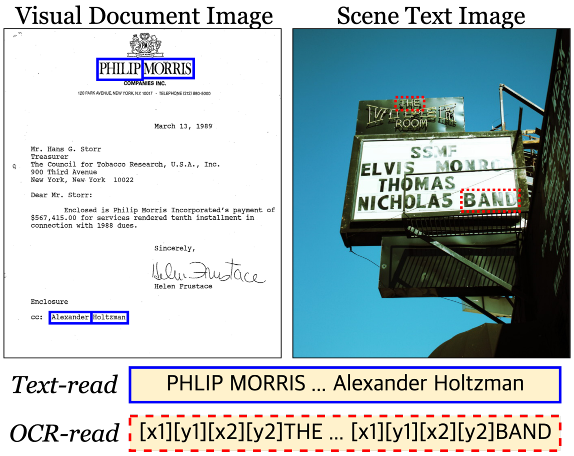
Visual Document Image
Scene Text Image
Text-read
PHLIP MORRIS ... Alexander Holtzman
OCR-read
[x1][y1][x2][y2]THE ... [x1][y1][x2][y2]BAND

Figure 2. *Text-read* reads the text present in an image in raster scan order (blue solid box), while *OCR-read* detects both the text and its corresponding coordinates from the image (red dashed box).

expand tasks or domains more easily. Inspired by these approaches, we investigate using VDU and STU data together for integrating text-related tasks.

This work aims to pre-train a universal text understanding model for both document and scene domains and extend its use to downstream tasks. However, we empirically observed that *text-read* using both VDU and STU data often fails and becomes unstable, likely due to the complex natural scene backgrounds in STU conflicting with the document images in VDU. To mitigate this issue, we explore *OCR-read* (i.e., optical character recognition) [48], which explicitly guides the model to recognize text in complex images by adding coordinate tokens from *text-read* in the sequence generation architecture. While *OCR-read* has successfully pre-trained both domains, it requires high-cost annotation due to the need of location information, unlike *text-read*. According to Yair *et al.* [34], adding box coordinate annotations for the OCR task increases annotation time by about 140% compared to text-only annotations.

To address the issues, we propose a novel pre-training method called **SCOB**, which stands for Character-wise **S**upervised **C**ontrastive Learning (SupCon) with **O**nline Text Rendering for **B**ridging Domain Gap. Our online text renderer serves as a more effective augmentation method than augmentation methods used in conventional representation learning [16, 20, 6] for character-wise SupCon, as shown in Figure 1. SCOB bridges the domain gap by learning recognition more easily through rendered images and attracting the projection of positive samples from synthetic, scene text, and document domains to each other. Applying SCOB to *text-read* provides learning stability, indicating that SCOB effectively bridges the domain gap. Moreover, pre-training *OCR-read* with SCOB requires image data with only text annotation, dramatically reducing annotation costs compared to traditional OCR training.

The proposed SCOB is applicable to a broad range of existing Transformer-based generative text understanding models, including Donut [32], Dessurt [12], and Pix2Struct [36]. From a generalized perspective, these image-to-text models can be interpreted as the same framework, which is a transformer-based encoder-decoder model with a “read” pre-training strategy (e.g., *text-read* and *OCR-read*). We refer to this framework as the **U**niversal Text **U**nderstanding (**W**) framework in this work. We conduct extensive experiments using the **W** framework on eleven benchmarks spanning both VDU and STU domains to observe the characteristics and effects of respective pre-training strategies with SCOB. Our experimental results and analysis demonstrate the efficacy and versatility of SCOB improving the overall model performance. We summarize our main contributions as follows:

- This paper investigates the effects of *text-read* and *OCR-read* pre-training on a total of eleven tasks, including those in the VDU and STU domains.
- We propose a novel pre-training method SCOB that utilizes character-wise contrastive learning with online text rendering to effectively bridges the domain gap between VDU and STU domains.
- SCOB enables weakly supervised OCR pre-training, significantly reducing annotation costs by using only text annotations.
- Experimental results show that read-based pre-training for table reconstruction achieves state-of-the-art performance, and our proposed SCOB generally enhances the performance of read-based pre-training on various text-related downstream tasks.

2. Related Work

2.1. Visual Document Understanding

Inspired by the great success of BERT [28] in natural language process tasks, Xu *et al.*[58] presented a powerful VDU model, LayoutLM, with an efficient pre-training task, named masked visual-language modeling. Recently, LayoutLMv3 [22] exploited masked image modeling with latent codes of a discrete VAE and achieved state-of-the-art. However, these approaches [58, 59, 38, 22, 21] require a specific architectural design for the output format of each downstream task. Moreover, since they use OCR results as input, they strongly depend on the OCR engine. In addition, OCR increases overall computational cost, and the errors of OCR often propagate to the final outputs [32].

In order to solve these challenges, Kim *et al.* [32] proposed Donut that does not require preprocessing as OCR. Donut is an end-to-end encoder-decoder model that autoregressively generates the desired type of output sequence. With a simple concept, Donut solves multiple VDU tasks with a single unified pipeline, and it showed state-of-the-art

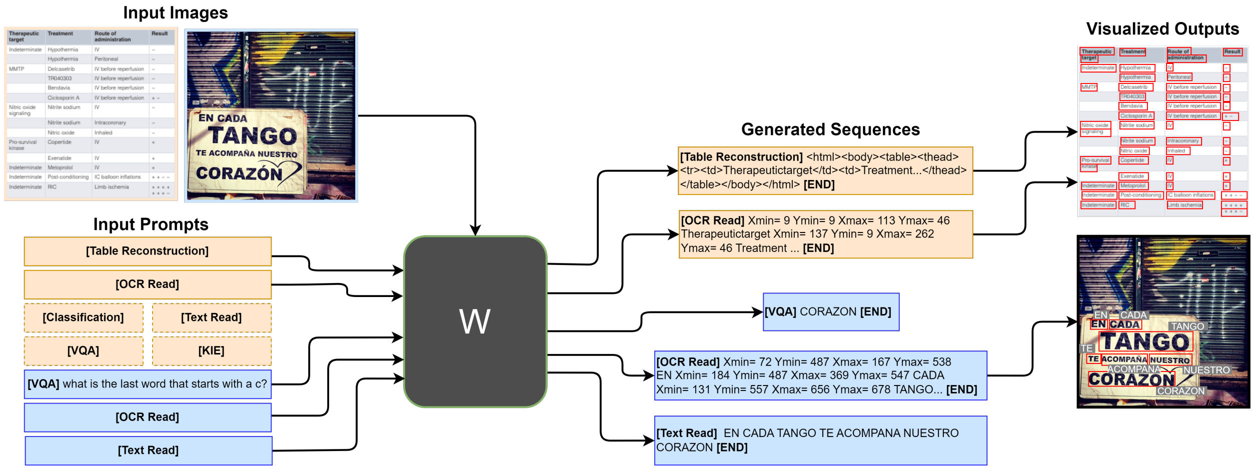
Input Images
TANGO
Input Prompts
[Table Reconstruction]
[OCR Read]
[Classification]
[Text Read]
[VQA]
[KIE]
[VQA] what is the last word that starts with a c?
[OCR Read]
[Text Read]
W
Generated Sequences
[Table Reconstruction] <html><body><table><thead><tr><td>Therapeutictarget</td><td>Treatment...</thead></table></body></html> [END]
[OCR Read] Xmin= 9 Ymin= 9 Xmax= 113 Ymax= 46 Therapeutictarget Xmin= 137 Ymin= 9 Xmax= 262 Ymax= 46 Treatment ... [END]
[VQA] CORAZON [END]
[OCR Read] Xmin= 72 Ymin= 487 Xmax= 167 Ymax= 538 EN Xmin= 184 Ymin= 487 Xmax= 369 Ymax= 547 CADA Xmin= 131 Ymin= 557 Xmax= 656 Ymax= 678 TANGO... [END]
[Text Read] EN CADA TANGO TE ACOMPANA NUESTRO CORAZON [END]
Visualized Outputs

Figure 3. The overview of the **U**niversal Text **U**nderstanding (**W**) framework. This framework provides a unified approach for various visual text-related tasks. Given an input text image, **W** generates output sequences for downstream tasks conditioned on task-prompts in the text decoder. We aim to train the **W** framework with pre-training tasks such as *text-read* or *OCR-read* to effectively handle both VDU (yellow box) and STU (blue box) domains. When the model is pre-trained without domain conflicts, it can be fine-tuned on various text understanding tasks spanning both domains, such as table reconstruction, OCR, classification, VQA, and KIE, with improved performance.

performances on various VDU tasks. Donut is pre-trained with a task, denoted as *text-read* task, which is simply reading all characters in the image with raster order [32]. Although Donut showed promising results on many VDU tasks, it has not been investigated yet how it performs in scene text-related tasks. In addition, we note that text localization is likely to be important for some text-related tasks, which have not been explored deeply in the previous works [32, 12]. In this paper, *text-read* and *OCR-read* are investigated as pre-training methods and their impact on VDU and STU downstream tasks.

2.2. Contrastive Learning for Visual Representation Learning

In the computer vision, unsupervised representation learning methods have succeeded with contrastive learning [16, 20, 6, 8, 29, 9]. The common idea of these methods is that image augmentation is performed on a single batch of images, where a pair of augmented images can be treated as a positive pair originally taken from the same image and a negative pair originally taken from two different images. The augmentation plays a critical role in contrastive learning, and the effect of its type and intensity has been extensively investigated [16, 20, 6, 8, 24]. Another critical factor is a large number of samples [6]; thus, dictionary-based methods [57, 20, 9] have been proposed to cache the negative samples. To fully leverage supervision, supervised contrastive learning methods have been proposed [29, 23] where the positives and negatives are constructed with their label information.

STU field, especially OCR, has also explored the application of contrastive learning to train image and text encoders [3, 1, 60]. Specifically, self-supervised learning was employed for text recognizer [3, 1] and these methods can be mainly categorized by the instance-level contrastive learning. Baek *et al.* [3] defined instance-level as an image and applied MoCo [20] for representation learning. On the other hand, Aberdam *et al.* [1] defined instance-level as a sub-image under an assumption that the placement of texts in positives would not be that different unless severe placement-related augmentations (e.g., flip) are applied. Recently, CLIP-based [49] contrastive learning was proposed to train both text and image encoders with label information. In this paper, we pre-train the auto-regressive text decoder as well as the image encoder with SCOB, which can be effectively transferred to downstream tasks.

3. Method

Inspired by Donut [32], we adopt the sequence generation model to process various downstream tasks with a single architecture. For pre-training the sequence generation model, we investigate two objectives: *text-read* and *OCR-read*. As shown in Figure 3, *text-read* is simply reading all characters in the image with raster order [32]. *OCR-read* incorporates *text-read* and text localization objectives that decodes the coordinates of bounding boxes and text transcriptions. We expect *OCR-read* can employ richer information packaging physical coordinates and sizes, as well as the relative distances between text instances [61]. Since text localization occupies part of the target sequence, shorter text transcriptions can be exploited for a learning language model because the decoder has a limited decoding max length. Thus, *text-read* learns a language model more com-

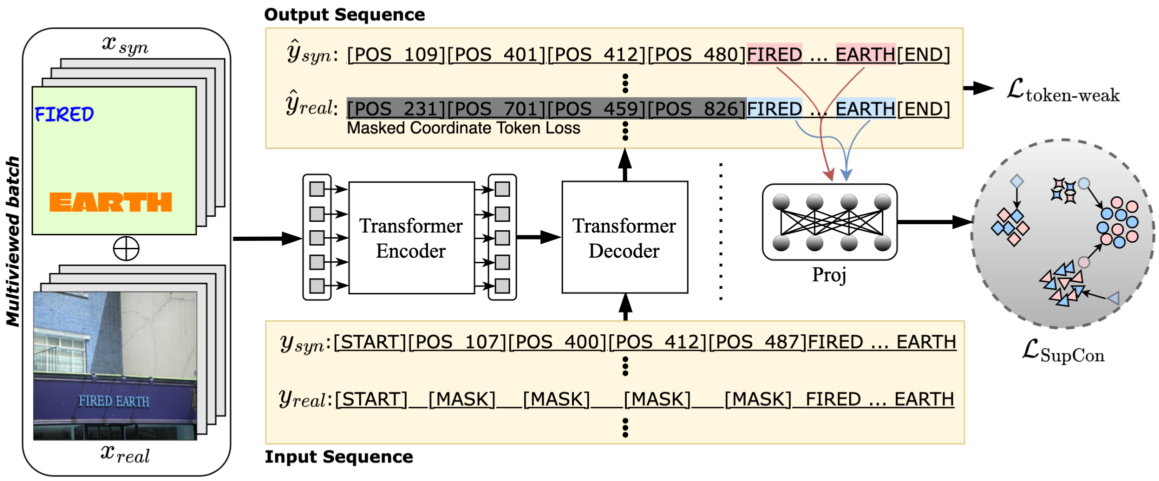
Output Sequence
$\hat{y}_{syn}$: [POS_109][POS_401][POS_412][POS_480]FIRED ... EARTH[END]
$\hat{y}_{real}$: [POS_231][POS_701][POS_459][POS_826]FIRED ... EARTH[END]
Masked Coordinate Token Loss
$\mathcal{L}_{\text{token-weak}}$
x_{syn}
FIRED
EARTH
Multiviewed batch
FIRED EARTH
x_{real}
Transformer Encoder
Transformer Decoder
Proj
$\mathcal{L}_{\text{SupCon}}$
y_{syn}:[START][POS_107][POS_400][POS_412][POS_487]FIRED ... EARTH
y_{real}:[START] [MASK] [MASK] [MASK] [MASK] FIRED ... EARTH
Input Sequence

Figure 4. An illustration of the proposed **SCOB**-applied *OCR-read* pre-training method for text understanding models. SCOB can also be applied to *text-read* by excluding the coordinate tokens. Given real images, $\mathbf{x}_{real}$, our renderer generates a corresponding synthetic batch, $\mathbf{x}_{syn}$, and the multiviewed batch is used to train the model. Under a teacher-forcing scheme [56], the model is trained using a cross-entropy loss, $\mathcal{L}_{\text{token-weak}}$, along with a contrastive loss, $\mathcal{L}_{\text{SupCon}}$. Notably, SCOB does not require coordinate annotations of $\mathbf{y}_{real}$. The coordinate labels of $\mathbf{y}_{real}$ are replaced with masked tokens ([MASK]), and the loss from the masked tokens is ignored (gray box). To compute the $\mathcal{L}_{\text{SupCon}}$, the predicted character embeddings are passed through an MLP projector (Proj). Note that the predicted character embedding is the last layer hidden embedding of the decoder, $\mathbf{d}$, not the character token. With SCOB, the same classes of characters are forced to be clustered in the embedding space, leading to improved model robustness.

prehensively when the image contains many text instances that the decoding length of *OCR-read* cannot cover entire text instances.

Furthermore, we propose SCOB, a novel pre-training approach that leverages character-wise SupCon and online text rendering to maximize their synergy. The online text renderer serves as a suitable augmentation method for general text-related contrastive learning. Document images [37] are usually well-scanned or binarized, making it easy for models to recognize text, whereas scene text images can be challenging due to the natural background. Also, we observe that *text-read* is not robust enough to pre-train document and scene text data together (see Section 4.3). SCOB overcomes this limitation by training the model on synthetic text images, which are easier to recognize, and transferring this knowledge to more challenging real-world scenarios. Specifically, SCOB pulls the feature of positive samples from synthetic, document, and scene text images, facilitating the learning of difficult samples. Moreover, SCOB supports *text-read* and *OCR-read* pre-training methods and can be trained with weak supervision for *OCR-read*, making it an ideal solution for scenarios where data is scarce.

In the next Section 3.1, we explain the architecture of $\mathbf{W}$ and vanilla pre-training objectives. Then, we discuss the character-wise supervised contrastive learning method pre-training in Section 3.2. Finally, Section 3.3 describes the detailed settings of the online text renderer and the weakly supervised pre-training method.

3.1. Read-based Pre-training

Recently, the proposal of Pix2Seq [7] made detection possible with sequence generation, which allows unifying the output of multiple tasks, including detection. Our *OCR-read* pre-training is also in the form of sequence generation, and the sequence is composed of coordinates (bounding box) and transcription as shown in Figures 3, and 4. To express the bounding box as a sequence, we uniformly discretize the height and width of the image into 1,000 bins following Pix2Seq. Therefore, the sequence of word instance consists of 4 coordinate tokens $[x_{min}, y_{min}, x_{max}, y_{max}]$, followed by n character tokens (transcription). In the case of *text-read* pre-training, the target sequence is composed of only character tokens, which can handle more words than *OCR-read*.

Architecture. The architecture of $\mathbf{W}$ follows the encoder-decoder framework. The image encoder converts the input image $\mathbf{x} \in \mathbb{R}^{H \times W \times C_{in}}$ into visual embedding $\mathbf{v}$ = Enc($\mathbf{x}$) where H, W, and C_{in} denote the height, width, and channel of the input image, respectively. The text decoder takes both $\mathbf{v}$ and previously generated token from the decoder as input and auto-regressively generates output sequence $(\hat{\mathbf{y}})_{i=1}^{N}$ where $\hat{\mathbf{y}}_i$ is the i-th generated token, and N denotes the sequence length of the decoder. We use Swin Transformer [39] and Transformer-based [54] decoder as an encoder and decoder, respectively.

Objective. The model learns to predict target tokens such as prompt, coordinate (only for *OCR-read*), and character tokens, using maximum likelihood:

$$\mathcal{L}_{\text{token}} = -\sum_{i=1}^{N} \log P(\mathbf{y}_i|\mathbf{x}, \hat{\mathbf{y}}_{1:i\text{-}1}), \tag{1}$$

where $\mathbf{y}_i$ denotes i-th target token.

3.2. Character-wise Supervised Contrastive Loss

In general image classification, supervised contrastive learning [29] has been mostly applied at image-level. On the other hand, in the image containing text, each character can be regarded as an instance and we propose character-wise contrastive learning. The proposed method enables stable learning without using a kind of memory bank (dynamic dictionary queue [20]) because character-wise SupCon allows obtaining abundant positive and negative samples even in a small batch where most OCR images have high-resolution sizes (larger than 768×768). Another major factor of contrastive learning is the augmentation for multiview images. We propose to generate multiview images using the online text renderer, which will be described in the following subsection.

Figure 4 shows the overview of the proposed character-wise SupCon. Our model takes original (real) and multiview (synthetic) images as the input and auto-regressively generates the token $\hat{y}_i = \text{MLP}(\mathbf{d}_i)$ where $\mathbf{d}_i$ denotes the last hidden embedding of the decoder at i-th generation index. At the same time, character-wise projections $\mathbf{z}_i = \text{Proj}(\mathbf{d}_i)$ are placed in a contrastive subspace. Here, we define a multiviewed batch as a union of original batch and rendered batch constructed by the renderer. Within a multiviewed batch, let $j \in \mathbf{C}$ be the index of a character where $\mathbf{C}$ denotes the set of all target characters in multiviewed batch. Then, the model is trained with character-wise supervised contrastive loss:

$$\mathcal{L}_{\text{SupCon}} = \sum_{j\in\mathbf{C}} \frac{-1}{|P(j)|} \sum_{p\in P(j)} \log \frac{\exp(\mathbf{z}_j \cdot \mathbf{z}_p/\tau)}{\sum_{a\in A(j)} \exp(\mathbf{z}_j \cdot \mathbf{z}_a/\tau)}, \tag{2}$$

where $A(j) = \mathbf{C} \setminus \{j\}$, and $P(j) = \{p \in A(j) : \mathbf{c}_p = \mathbf{c}_j\}$ is the set of indices that have same character label $\mathbf{c}$ in the multiviewed batch. $|P(j)|$, symbol $\cdot$ and τ denote cardinality of $P(j)$, dot product, and scalar temperature, respectively.

3.3. Online Text Renderer

To address the limitations of existing augmentation methods for OCR in contrastive learning [3, 1, 49], we propose the online text renderer. Existing methods cannot employ strong geometric augmentations like crop and flip, as the same word must be identically contained in multiview images, which weakens the variance between the positive views. In contrast, the renderer generates an image corresponding one-to-one with the original image using text transcription, outputting the synthetic image and the bounding box annotation together. This creates a more substantial variance between the same characters even without strong augmentation, such as cropping.

The renderer is designed specifically for the character-wise SupCon and differs from existing renderers [17, 32] in two significant ways: i) We adopt an online generation method to replace existing augmentation in contrastive learning. ii) To maximize the character variance with high speed, we randomly select various fonts and background colors. We also observed that the renderer could generate synthetic data in less time than it takes to load real data into memory. Overall, the proposed renderer presents a simple and efficient approach to text rendering that can improve the variance between positive views, leading to better contrastive learning results.

Online Rendering Engine. Our generation engine is implemented using the Python Pillow package [11]. To generate synthetic text images, we require only a font and a set of words. We leverage more than 3,000 fonts provided by Google* to maximize the character variance. Detailed settings can be adjusted, including image resolution range, background RGB range, font size range, and whether to generate character-level coordinates. The background color is chosen randomly within the specified range, and each word in the set is rendered at a randomly selected location and size within the specified range. Examples of synthetic images are shown in Figures 1 and 4.

Weakly Supervised Pre-training OCR. We propose a weakly supervised pre-training for *OCR-read* that eliminates the need for expensive coordinate annotation of real images. Specifically, the model learns coordinate information solely from rendered data. As shown in Figure 4, the proposed weakly supervised learning involves two steps: i) replacing the input coordinate tokens of real data with mask tokens in a teacher-forcing scheme [56], and ii) masking the coordinate token loss of the real data. As a result, the model learns localization and recognition on synthetic data while only learning text annotations on real data. Additionally, the renderer generates the same characters of input text annotations in multiview images, providing high-quality positive samples for character-wise SupCon. The proposed weakly supervised learning can be expressed as follows:

$$\mathcal{L}_{\text{token-weak}} = -\sum_{i=1}^{N} w_i \log P(\mathbf{y}_i|\mathbf{x}, \hat{\mathbf{y}}_{1:i\text{-}1}), \tag{3}$$

where w_i denotes a pre-assigned weight for coordinate tokens in the sequence. We set $w_i = 0$ for the coordinate

*https://fonts.google.com

tokens of real data and set $w_i = 1$ for other cases. In the case of *text-read*, $\mathcal{L}_{\text{token-weak}}$ is equivalent to $\mathcal{L}_{\text{token}}$ due to the absence of the coordinate tokens.

Loss Function. Finally, we present the following loss function $\mathcal{L}$ of SCOB that consists of token loss $\mathcal{L}_{\text{token-weak}}$ as well as our character-wise supervised contrastive loss $\mathcal{L}_{\text{SupCon}}$:

$$\mathcal{L} = \frac{1}{M} \sum_{m=1}^{M} \mathcal{L}^{m}_{\text{token-weak}} + \lambda \mathcal{L}_{\text{SupCon}}, \tag{4}$$

where M is the number of image-label pairs in multiviewed batch and λ denotes a scaling factor of $\mathcal{L}_{\text{SupCon}}$.

4. Experiments

4.1. Pre-training Details

Architecture Setup. Architecture of **W** has a few changes from Donut [32]. We use Swin-B [40] pre-trained with ImageNet-22K [13] as a visual encoder and set the layer numbers to [2, 2, 18, 2], window size to 12, and input resolution to 768×768. The decoder uses a 12-layer Transformer [54] initialized by BERT [28] with 12 heads and 768 hidden size, employing character tokenization with 512 maximum sequence length due to its empirical superiority in OCR. The model is trained for 1M steps using Adam [33] optimizer with a batch size of 32 distributed across 8 NVIDIA V100 GPUs. For SCOB training, we set $\lambda = 0.5$, $\tau = 0.07$, and a 2-layer MLP character-level projector with 128 hidden dimensions. Since SCOB for *OCR-read* is performed under the assumption that the coordinates of the real dataset are unavailable, it learns a sequence constructed in random order. The coordinate token is configured in the form of a bounding box for all datasets.

Dataset. For pre-training, we use the IIT-CDIP [37] and real scene text datasets [27, 26, 10, 35, 51, 41], common in VDU and scene text OCR, with batch ratios of 20% and 80%, respectively. IIT-CDIP is a dataset composed of 11M scanned English document images with abundant sentence-level texts and we achieve pseudo-OCR labels through the CLOVA OCR API following Donut [32]. We employ real scene text dataset such as ICDAR2013 [27], ICDAR2015 [26], TotalText [10], OpenImages v6 [35], TextOCR [51] and HierText [41] where the total amount of data is 857K. For OpenImages v6, we filter non-text images and obtain pseudo-OCR labels through CLOVA OCR API. These scene text datasets contain word instances and complex backgrounds, enabling the model to learn coordinate information and embed diverse visual features in contrastive subspace.

4.2. Fine-tuning Details on Downstream Tasks

To present a comprehensive investigation, we provide extensive benchmarks on 11 datasets as shown in Table 1. Although *OCR-read* has the advantage of the text localization objective, adding coordinate tokens can cause the maximum sequence length to be relatively insufficient. To compensate for this, we perform a *text-read* task of 50K as short intermediate training just before fine-tuning. Fine-tuned downstream tasks are briefly described as follows. We will provide fine-tuning details in the supplemental file.

Scene Text OCR. To evaluate the text localization objective, we fine-tune and evaluate the widely used scene text OCR datasets: ICDAR2013, ICDAR2015, and TotalText. The train set is the pre-training dataset excluding IIT-CDIP.

Table Reconstruction. PubTabNet is a dataset annotated with HTML format that contains 500K training, 9K validation, and 9K test samples. In this paper, our models decode contents in the cell as well as table structure from the input image. We employ TEDS [63] as an evaluation metric.

VQA for Scene Text and Document. For scene text and document VQA, we include additional datasets [45, 18, 15], following prior work [55], and fine-tune three models with different batch ratios: scene text VQA [50, 5], DocVQA [44], and InfoVQA [43]. Evaluation adheres to standard settings, with scene text VQA on the validation dataset [30], and document VQA on the test dataset.

Document Classification. RVL-CDIP [19], a subset of IIT-CDIP, is 400K scanned document images labeled into 16 categories. This dataset comprises 320K train images, 40K validation images, and 40K test images.

Key Information Extraction. CORD, the Consolidated Receipt Dataset, consists of 800 train, 100 validation, and 100 test receipt images. We construct the target sequence the same as Donut, and the performance is reported with a TED score between generated and ground-truth JSON files.

4.3. Performance Evaluation and Investigation

We investigate pre-training objectives and validate the effect of SCOB. Accordingly, we present four pre-trained models as shown in Table 1:

- $\mathbf{W}_{\text{OCR-read}}$: a *OCR-read* pre-trained model that learns a sequence composed of coordinate information and transcription in a raster scan order using Eq. 1.
- $\mathbf{W}_{\text{OCR-read}}$ w/ SCOB: a pre-trained model where SCOB is applied to *OCR-read*. It is trained by Eq. 4 with the coordinate token of rendered data.
- $\mathbf{W}_{\text{text-read}}$: a *text-read* pre-trained model that learns transcription-only sequences in raster scan order (pseudo-label order) using Eq. 1. This can be considered the previous *text-read* based method, such as Donut [32] and Dessurt [12].
- $\mathbf{W}_{\text{text-read}}$ w/ SCOB: a pre-trained model where SCOB is applied to *text-read*. It is trained by Eq. 4 without the coordinate token.

Method	#GPUs	Universal Text Understanding Downstream Tasks										
		Table Reconstruction	KIE	Document Classification	Document VQA		Layout Analysis	Scene Text OCR			Scene Text VQA	
		PubTabNet [63]	CORD [47]	RVL-CDIP [19]	DocVQA [44]	InfoVQA [43]	PubLayNet [64]	IC13 [27]	IC15 [26]	TotalText [10]	TextVQA [50]	ST-VQA [5]
$\mathrm{W}_{\text{OCR-read}}$	8×V100	96.0	88.2	94.2	56.1	22.7	93.8	95.8	89.6	84.9	55.4	**62.9**
$\mathrm{W}_{\text{OCR-read}}$ w/ SCOB	8×V100	95.9 (-0.1)	88.5 (+0.3)	94.6 (+0.4)	60.2 (+4.1)	**28.5** (+5.8)	93.9 (+0.1)	**96.6** (+0.8)	**90.9** (+1.3)	**86.0** (+1.1)	**56.2** (+0.8)	62.6 (-0.3)
$\mathrm{W}_{\text{text-read}}$	8×V100	**96.2**	85.5	94.4	57.0	25.2	93.6	96.0	87.2	83.7	49.3	57.2
$\mathrm{W}_{\text{text-read}}$ w/ SCOB	8×V100	96.0 (-0.2)	87.4 (+1.9)	94.3 (-0.1)	59.6 (+2.6)	27.5 (+2.3)	93.9 (+0.3)	96.0 (+0.0)	90.2 (+3.0)	85.3 (+1.6)	54.4 (+5.1)	61.2 (+4.0)
TableFormer [46]	n/a	93.7	-	-	-	-	-	-	-	-	-	-
$\mathrm{Donut}_{\text{proto}}$ [32]	8×V100	-	85.4	94.5	47.1	10.2*	-	-	-	-	-	-
Donut [32]	64×A100	-	**90.9**	95.3	67.5	24.4*	-	-	-	-	36.8*	61.5*
LayoutLMv3 [22]	32×V100	-	84.4*	**95.5**	**83.4**	-	**95.1**	-	-	-	-	-
SPTS [48]	32×V100	-	-	-	-	-	-	93.3	77.5	82.4	-	-
PreSTU [30]	n/a	-	-	-	-	-	-	-	-	-	54.5	62.6

Table 1. The extensive benchmarks for text-related downstream tasks. "#GPUs" denotes the total number of employed GPUs for pre-training. The left section is VDU tasks, and the right section is STU tasks. The best performance is represented in **bold**. Note that Donut was pre-trained on IIT-CDIP and SynthDoG, while $\mathrm{Donut}_{\text{proto}}$ was pre-trained on SynthDoG [32]. $*$ denotes the performance results of our fine-tuning, conducted following the author's guidelines.

During the pre-training of $\mathbf{W}_{\text{text-read}}$, we observed several instabilities. Thus, we employed the weights of $\mathbf{W}_{\text{text-read}}$ that achieved the highest score on the validation set. Additional results on the instability are in the supplemental file.

We also report scores of the comparison model. Specifically, TableFormer [46], SPTS [48], and PreSTU [30] focused on table reconstruction, scene text OCR, and scene text VQA tasks, respectively. Additionally, $\mathrm{Donut}_{\text{proto}}$ [32], Donut [32], and LayoutLMv3 [22] served as foundational models for VDU tasks. It is well-known that the total capacity of GPUs is crucial for pre-training. Unfortunately, we used only eight V100 GPUs, which are a relatively limited resources compared to other methods, resulting in relatively lower performance on few benchmarks. However, since the main goal of our paper is to present an investigation of pre-training methods and validate the effect of SCOB, our models still provide meaningful experimental results. In the supplemental material, an investigation is conducted into the ramifications stemming from variations in the batch size and image resolution during the pre-training phase. We believe this investigation can provide insights into the optimization of performance scalability through the harnessing of additional GPUs.

4.3.1 *Text-read* vs. *OCR-read*

We frequently observe that training $\mathbf{W}_{\text{text-read}}$ using both VDU and STU data leads to unstable learning, while $\mathbf{W}_{\text{OCR-read}}$ is trained stably. We suspect that the coordinate information in $\mathbf{W}_{\text{OCR-read}}$ alleviates domain conflict by guiding the text to be read explicitly even for complex scene images. Thus, $\mathbf{W}_{\text{text-read}}$ can be more vulnerable to training complex scene text images, which is represented in our experimental results. As shown in Table 1, $\mathbf{W}_{\text{OCR-read}}$ considerably outperforms $\mathbf{W}_{\text{text-read}}$ on all scene text benchmarks. On the other hand, $\mathbf{W}_{\text{text-read}}$ shows better performance on document VQA tasks. We think that comprehending the contextual information of the text within a well scanned or binarized image is a pivotal component of document VQA tasks. Thus, $\mathbf{W}_{\text{text-read}}$, which is trained on longer text sequences, can be beneficial for contextual comprehension.

4.3.2 The Effect of SCOB.

Experimental results validate the effect of SCOB on both *text-read* and *OCR-read*. We expect SCOB to facilitate stable pre-training by bridging the complex domain gap and we confirm $\mathbf{W}_{\text{text-read}}$ w/ SCOB is trained stably. Table 1 also shows that SCOB considerably improves the performance on scene text OCR, VQA, and KIE benchmarks with a large margin. In particular, it is quite notable that the improvements on TextVQA and ST-VQA are 5.1 and 4.0, respectively. $\mathbf{W}_{\text{OCR-read}}$ w/ SCOB also notably outperforms $\mathbf{W}_{\text{OCR-read}}$ on VQA and scene text OCR benchmarks, achieving the best performance among comparisons in infoVQA. We would emphasize that $\mathbf{W}_{\text{OCR-read}}$ w/ SCOB is trained under the weakly supervised setting. Its required annotation is equivalent to $\mathbf{W}_{\text{text-read}}$ and $\mathbf{W}_{\text{text-read}}$ w/ SCOB, which is much lower cost than that of $\mathbf{W}_{\text{OCR-read}}$.

Large improvements are generally achieved at the OCR and VQA tasks. This can be because better character recognition is a prerequisite for a better understanding of document or scene text. Moreover, VQA is closely related to OCR because most of the answers exist in the image containing text. Significant enhancements to KIE, which involves the task of reading and organizing word boxes, arise from analogous reasons.

4.3.3 Comparison with SoTA Methods

As shown in Table 1, the presented models achieve competitive or better performance across the VDU and STU

Method	#Params	CORD [47] (Acc)	RVL-CDIP [19] (Acc)	DocVQA [44] (ANLS)
BERT [28]	110M + α	65.5	89.8	63.6
LayoutLM [58]	113M + α	81.3	94.4	69.8
LayoutLMv2 [59]	200M + α	82.4	95.3	78.1
LayoutLMv3 [22]	133M + α	84.4	**95.4**	**78.8**
Dessurt [12]	127M	-	93.6	63.2
Donut$_{\text{proto}}$ [32]	143M	85.4	94.5	47.1
Donut [32]	143M	**90.9**	95.3	67.5
$\mathrm{W}_{\text{OCR-read}}$	202M	88.2	94.2	56.1
$\mathrm{W}_{\text{OCR-read}}$ w/ SCOB	202M	88.5	94.6	60.5
$\mathrm{W}_{\text{text-read}}$	202M	84.4	94.4	57.0
$\mathrm{W}_{\text{text-read}}$ w/ SCOB	202M	87.4	94.3	59.6

Table 2. The public benchmark on CORD [47], RVL-CDIP [19], and DocVQA [44]. α is represented for the requirement of an additional OCR model.

Method	#Param	TextVQA [50] (Acc.)	ST-VQA [5] (ANLS)
SA-M4C [25]	93M	45.4	51.2
TAP [62]	160M	54.7	59.8
GIT$_{Large}$ [55]	347M	37.5	44.6
PreSTU [30]	278M	54.5	62.6
$\mathrm{W}_{\text{OCR-read}}$	202M	55.4	**62.9**
$\mathrm{W}_{\text{OCR-read}}$ w/ SCOB	202M	**56.2**	62.6
$\mathrm{W}_{\text{text-read}}$	202M	49.3	57.2
$\mathrm{W}_{\text{text-read}}$ w/ SCOB	202M	54.4	61.2
Flamingo [2]	80B	57.1	-
GIT [55]	681M	59.9	69.1
LaTr [4] + Rosetta-en	n/a	48.4	-
LaTr [4] + Amazon-OCR	n/a	59.5	67.5

Table 3. The public benchmark on TextVQA [50] and ST-VQA [5] for scene text VQA. LaTr requires the result of OCR as the input. We report two results depending on the employed OCR models. The best performance among similar-sized models (#Param less than 400M) is represented in **bold**.

benchmarks. $\mathbf{W}_{\text{text-read}}$, regarded as a re-implementation of Donut using our framework, is faithfully reproduced given the number of GPUs used, batch size, and resolution. To validate the presence of the domain gap between VDU and STU data, we fine-tune Donut on scene text VQA benchmarks where Donut is mainly pre-trained on VDU data. While Donut achieved a competitive advantage on the DocVQA benchmark, it only managed to secure comparable scores on ST-VQA. Furthermore, its performance significantly dropped in the TextVQA. We also find that read-based pre-training is effective for table reconstruction and our models achieve state-of-the-art. In STU benchmarks, $\mathbf{W}_{\text{OCR-read}}$ w/ SCOB outperforms SPTS on all of the scene text OCR benchmarks and achieves better performance than PreSTU on TextVQA.

Input	Method	TEDS		
		Simple	Complex	All
PDF	Tabula	78.0	57.8	67.9
	Acrobat Pro	68.9	61.8	65.3
	TableFormer [46]	95.4	90.1	93.6
Image	Acrobat Pro	53.8	53.5	53.7
	WYGIWYS [14]	81.7	75.5	78.6
	EDD [63]	91.2	85.4	88.3
	$\mathrm{W}_{\text{OCR-read}}$	97.7	94.2	96.0
	$\mathrm{W}_{\text{OCR-read}}$ w/ SCOB	97.5	94.1	95.9
	$\mathrm{W}_{\text{text-read}}$	**97.9**	**94.5**	**96.2**
	$\mathrm{W}_{\text{text-read}}$ w/ SCOB	97.6	94.1	96.0

Table 4. The public benchmark on PubTabNet [63] for table reconstruction including content in the cell.

4.4. Detailed Performance Comparison

In this subsection, we compare our models with more diverse methods. Table 2 shows that our models present the second-best performance on CORD and relatively lower performance on DocVQA. LayoutLMv2 [59] and LayoutLMv3 [22] have notable performance on DocVQA. Unlike Donut, Dessurt, and our models, LayoutLMs take both image and text (OCR) modalities as input, which additionally requires OCR results as a pre-process. Accordingly, LayoutLMv3 (1.8 sec/img) takes longer inference time than our models (1.1 sec/img) due to acquiring OCR results. We measure the inference time with a V100 GPU on CORD dataset [47]. Kim *et al.* [32] also reported that Donut is 2 times faster than LayoutLMv2.

As illustrated in Table 3, our models show comparable performance on scene text VQA tasks. Specifically, our model achieves the best performance among the similar-sized models. Surprisingly, $\mathrm{W}_{\text{OCR-read}}$ w/ SCOB achieves comparable performance to Flamingo, which has extremely large parameters. Since our models are also scalable, like GIT, we expect the enlarged models to reach the performance of GIT and Flamingo.

For table reconstruction, taking the image input is more challenging than PDF because the contents in the cell should also be decoded. As can be seen in Table 4, the performance of Acrobat Pro on PDF has much higher than that on the image input. Our models substantially outperform previous methods despite using the image input and can be a strong baseline for the generation model. We discuss more comparisons on other tasks such as layout analysis, and scene text OCR, in a supplemental file.

5. Analysis

Ablation Study. We conduct an ablation study on proposed components: (A) *OCR-read*, (B) character-wise SupCon, (C) online text rendering, (D) SCOB, and (E) SCOB with

Components	KIE [47]	DocVQA [44]	OCR [27, 26, 10]	Scene VQA [50, 5]
A. *OCR-read*	88.2	55.1	81.3	57.3
B. A w/ SupCon	88.0	50.0	82.2	56.8
C. A w/ rendering	87.7	47.8	82.0	54.3
D. A w/ SCOB	**88.5**	**55.5**	**83.0**	59.4
E. D w/ full annotation	86.8	55.1	82.6	**59.6**

Table 5. Ablation study on the proposed components. E denotes that the model is trained by SCOB with full annotations of both rendered and real images. We report the performance averaged on scene text OCR and scene text VQA.

full annotations of rendered and real images. We pre-train each model with different components and fine-tune each model on several downstream tasks. As shown in Table 5, B and C improve the performance of OCR but degrades that of the other downstream tasks. We think the reasons are as follows: i) For the case of SupCon, a naive augmentation used in previous works [16, 20, 6, 8] would not be beneficial to other downstream tasks (compare A vs. B). ii) For the case of online text rendering, it is trained only with half of the real data because half of the batch is charged with rendered images (compare A vs. C). iii) For the case of SCOB, the renderer plays a critical role as a fitted augmentation of SupCon by providing strong variance to positive samples. Also, character-wise SupCon bridges all synthetic, document, and scene domains by enforcing the same characters close together, which presents the synergistic effect. Comparing D and E, SCOB using full supervision could not significantly improve the performance, which shows SCOB is successfully pre-trained with weak supervision. We conduct an ablation study with a down-scaled setting for efficiency. More details will be provided in the supplementary material.

Qualitative Analysis. In Figure 5, we visualize representations extracted from the final layer of the decoder using t-SNE [53] by mapping high-dimensional features into low-dimensional space through KL-divergence. Note that SCOB clusters embeddings more discriminatively than other methods. More figures are in the supplemental material.

6. Discussion

We reported a wide range of benchmark results, some of which may not be desirable for validating SCOB. This is because we hope to contribute to the text understanding field by presenting a transparent investigation rather than hiding adverse findings. The sequence generation model inherently suffers from the limitation of maximum sequence length [32, 55]. For future works, it would be important to solve this problem, which may help the model further learn document understanding.

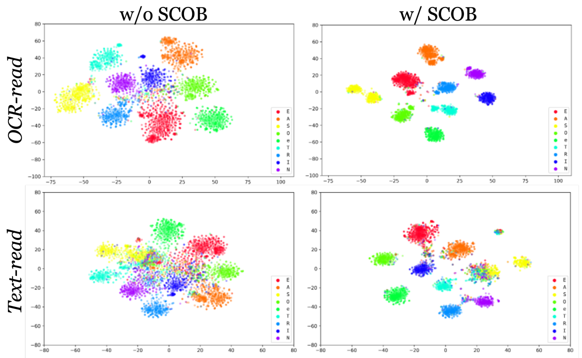

Figure 5. The t-SNE [53] visualization for *OCR-read*, *text-read*, and their respective SCOB applications. Note that different colors denote each class (character) and the nine most predicted characters are displayed. *Data*: ICDAR2015 [26] test set

Character vs. Subword Tokenizer. Despite a subword tokenizer's efficiency, we opted for a character tokenizer to improve performance on both VDU and STU tasks. Notably, using a subword tokenizer led to a drop in OCR performance, while other tasks maintained similar performance. This choice is consistent with OCR models like SPTS and UNITS [31]. While a decoding length of 512 may appear concerning, it facilitated larger batch sizes with the same GPU memory. For certain tasks, such as table reconstruction, we increased the decoding length during fine-tuning, thereby enhancing our final performance.

Random Placement of Words in Synthetic Images. In our study, we discerned a consistent stability in SCOB training across both the VDU and STU domains. Interestingly, even though the absence of real data coordinates in SCOB impinges upon the OCR's detection performance, it enhances recognition capabilities and promotes stability in training by leveraging more accessible rendering data. While the absence of word coordinates in real data could potentially disrupt the prediction of subsequent words, we posit that the employment of a teacher-forcing scheme, which feeds in ground-truth words during training, effectively mitigates this issue. Additionally, Sinha *et al.* [52] found that word co-occurrence statistics are more crucial than word order in MLM pre-training, which may explain why SCOB can pre-train effectively.

7. Conclusion

This paper investigates an effective pre-training on a total of eleven text-related tasks in the document and scene text domains. Our proposed SCOB is a new pre-training method for universal text understanding that leverages a character-wise supervised contrastive loss with online text rendering, enhancing the stability of training and reducing annotation costs. Experimental results on various visual text-related tasks validate that our SCOB is broadly applicable to read-based pre-training methods and improves performance.

Acknowledgements We greatly appreciate Teakgyu Hong and Han-Cheol Cho for their help with the initial project setup.

References

[1] Aviad Aberdam, Ron Litman, Shahar Tsiper, Oron Anschel, Ron Slossberg, Shai Mazor, R Manmatha, and Pietro Perona. Sequence-to-sequence contrastive learning for text recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15302–15312, 2021. 3, 5

[2] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *arXiv preprint arXiv:2204.14198*, 2022. 8

[3] Jeonghun Baek, Yusuke Matsui, and Kiyoharu Aizawa. 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 (CVPR)*, pages 3113–3122, June 2021. 3, 5

[4] Ali Furkan Biten, Ron Litman, Yusheng Xie, Srikar Appalaraju, and R Manmatha. Latr: Layout-aware transformer for scene-text vqa. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16548–16558, 2022. 8

[5] Ali Furkan Biten, Ruben Tito, Andres Mafla, Lluis Gomez, Marçal Rusinol, Ernest Valveny, CV Jawahar, and Dimosthenis Karatzas. Scene text visual question answering. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4291–4301, 2019. 6, 7, 8, 9

[6] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020. 2, 3, 9

[7] Ting Chen, Saurabh Saxena, Lala Li, David J Fleet, and Geoffrey Hinton. Pix2seq: A language modeling framework for object detection. *arXiv preprint arXiv:2109.10852*, 2021. 1, 4

[8] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 3, 9

[9] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9640–9649, 2021. 3

[10] Chee Kheng Ch'ng and Chee Seng Chan. Total-text: A comprehensive dataset for scene text detection and recognition. In *2017 14th IAPR international conference on document analysis and recognition (ICDAR)*, volume 1, pages 935–942. IEEE, 2017. 6, 7, 9

[11] Alex Clark. Pillow (pil fork) documentation, 2015. 5

[12] Brian Davis, Bryan Morse, Bryan Price, Chris Tensmeyer, Curtis Wigington, and Vlad Morariu. End-to-end document recognition and understanding with dessurt. *arXiv e-prints*, pages arXiv–2203, 2022. 2, 3, 6, 8

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

[14] Yuntian Deng, Anssi Kanervisto, Jeffrey Ling, and Alexander M Rush. Image-to-markup generation with coarse-to-fine attention. In *International Conference on Machine Learning*, pages 980–989. PMLR, 2017. 8

[15] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6904–6913, 2017. 6

[16] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020. 2, 3, 9

[17] Ankush Gupta, Andrea Vedaldi, and Andrew Zisserman. Synthetic data for text localisation in natural images. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 5

[18] Danna Gurari, Qing Li, Abigale J Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P Bigham. Vizwiz grand challenge: Answering visual questions from blind people. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3608–3617, 2018. 6

[19] Adam W Harley, Alex Ufkes, and Konstantinos G Derpanis. Evaluation of deep convolutional nets for document image classification and retrieval. In *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*, pages 991–995. IEEE, 2015. 6, 7, 8

[20] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020. 2, 3, 5, 9

[21] Teakgyu Hong, Donghyun Kim, Mingi Ji, Wonseok Hwang, Daehyun Nam, and Sungrae Park. Bros: A pre-trained language model focusing on text and layout for better key information extraction from documents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 10767–10775, 2022. 2

[22] Yupan Huang, Tengchao Lv, Lei Cui, Yutong Lu, and Furu Wei. Layoutlmv3: Pre-training for document ai with unified text and image masking. *arXiv preprint arXiv:2204.08387*, 2022. 2, 7, 8

[23] Ashraful Islam, Chun-Fu Richard Chen, Rameswar Panda, Leonid Karlinsky, Richard Radke, and Rogerio Feris. A broad study on the transferability of visual representations with contrastive learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8845–8855, 2021. 3

[24] Li Jing, Pascal Vincent, Yann LeCun, and Yuandong Tian. Understanding dimensional collapse in contrastive self-supervised learning. In *International Conference on Learning Representations*, 2022. 3

[25] Yash Kant, Dhruv Batra, Peter Anderson, Alexander Schwing, Devi Parikh, Jiasen Lu, and Harsh Agrawal. Spatially aware multimodal transformers for textvqa. In *European Conference on Computer Vision*, pages 715–732. Springer, 2020. 8

[26] Dimosthenis Karatzas, Lluis Gomez-Bigorda, Anguelos Nicolaou, Suman Ghosh, Andrew Bagdanov, Masakazu Iwamura, Jiri Matas, Lukas Neumann, Vijay Ramaseshan Chandrasekhar, Shijian Lu, et al. Icdar 2015 competition on robust reading. In *2015 13th international conference on document analysis and recognition (ICDAR)*, pages 1156–1160. IEEE, 2015. 6, 7, 9

[27] 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. Icdar 2013 robust reading competition. In *2013 12th international conference on document analysis and recognition*, pages 1484–1493. IEEE, 2013. 6, 7, 9

[28] Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186, 2019. 2, 6, 8

[29] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in Neural Information Processing Systems*, 33:18661–18673, 2020. 3, 5

[30] Jihyung Kil, Soravit Changpinyo, Xi Chen, Hexiang Hu, Sebastian Goodman, Wei-Lun Chao, and Radu Soricut. Prestu: Pre-training for scene-text understanding. *arXiv preprint arXiv:2209.05534*, 2022. 6, 7, 8

[31] Taeho Kil, Seonghyeon Kim, Sukmin Seo, Yoonsik Kim, and Daehee Kim. Towards unified scene text spotting based on sequence generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15223–15232, June 2023. 9

[32] Geewook Kim, Teakgyu Hong, Moonbin Yim, JeongYeon Nam, Jinyoung Park, Jinyeong Yim, Wonseok Hwang, Sangdoo Yun, Dongyoon Han, and Seunghyun Park. Ocr-free document understanding transformer. In *European Conference on Computer Vision (ECCV)*, 2022. 1, 2, 3, 5, 6, 7, 8, 9

[33] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[34] Yair Kittenplon, Inbal Lavi, Sharon Fogel, Yarin Bar, R Manmatha, and Pietro Perona. Towards weakly-supervised text spotting using a multi-task transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4604–4613, 2022. 2

[35] Ivan Krasin, Tom Duerig, Neil Alldrin, Vittorio Ferrari, Sami Abu-El-Haija, Alina Kuznetsova, Hassan Rom, Jasper Uijlings, Stefan Popov, Shahab Kamali, Matteo Malloci, Jordi Pont-Tuset, Andreas Veit, Serge Belongie, Victor Gomes, Abhinav Gupta, Chen Sun, Gal Chechik, David Cai, Zheyun Feng, Dhyanesh Narayanan, and Kevin Murphy. Openimages: A public dataset for large-scale multi-label and multi-class image classification. *Dataset available from https://storage.googleapis.com/openimages/web/index.html*, 2017. 6

[36] Kenton Lee, Mandar Joshi, Iulia Turc, Hexiang Hu, Fangyu Liu, Julian Eisenschlos, Urvashi Khandelwal, Peter Shaw, Ming-Wei Chang, and Kristina Toutanova. Pix2struct: Screenshot parsing as pretraining for visual language understanding. *arXiv preprint arXiv:2210.03347*, 2022. 2

[37] David Lewis, Gady Agam, Shlomo Argamon, Ophir Frieder, David Grossman, and Jefferson Heard. Building a test collection for complex document information processing. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 665–666, 2006. 4, 6

[38] Peizhao Li, Jiuxiang Gu, Jason Kuen, Vlad I Morariu, Handong Zhao, Rajiv Jain, Varun Manjunatha, and Hongfu Liu. Selfdoc: Self-supervised document representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5652–5660, 2021. 2

[39] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10012–10022, October 2021. 4

[40] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021. 6

[41] Shangbang Long, Siyang Qin, Dmitry Panteleev, Alessandro Bissacco, Yasuhisa Fujii, and Michalis Raptis. Towards end-to-end unified scene text detection and layout analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 6

[42] Jiasen Lu, Christopher Clark, Rowan Zellers, Roozbeh Mottaghi, and Aniruddha Kembhavi. Unified-io: A unified model for vision, language, and multi-modal tasks. *arXiv preprint arXiv:2206.08916*, 2022. 1

[43] Minesh Mathew, Viraj Bagal, Rubèn Tito, Dimosthenis Karatzas, Ernest Valveny, and CV Jawahar. Infographicvqa. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1697–1706, 2022. 6, 7

[44] Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. Docvqa: A dataset for vqa on document images. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 2200–2209, 2021. 6, 7, 8, 9

[45] Anand Mishra, Shashank Shekhar, Ajeet Kumar Singh, and Anirban Chakraborty. Ocr-vqa: Visual question answering by reading text in images. In *2019 international conference on document analysis and recognition (ICDAR)*, pages 947–952. IEEE, 2019. 6

[46] Ahmed Nassar, Nikolaos Livathinos, Maksym Lysak, and Peter Staar. Tableformer: Table structure understanding with transformers. In *Proceedings of the IEEE/CVF Conference*

on Computer Vision and Pattern Recognition, pages 4614–4623, 2022. 7, 8

[47] Seunghyun Park, Seung Shin, Bado Lee, Junyeop Lee, Jaeheung Surh, Minjoon Seo, and Hwalsuk Lee. Cord: a consolidated receipt dataset for post-ocr parsing. In *Workshop on Document Intelligence at NeurIPS 2019*, 2019. 7, 8, 9

[48] Dezhi Peng, Xinyu Wang, Yuliang Liu, Jiaxin Zhang, Mingxin Huang, Songxuan Lai, Jing Li, Shenggao Zhu, Dahua Lin, Chunhua Shen, et al. Spts: Single-point text spotting. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 4272–4281, 2022. 2, 7

[49] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 3, 5

[50] Amanpreet Singh, Vivek Natarjan, Meet Shah, Yu Jiang, Xinlei Chen, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8317–8326, 2019. 6, 7, 8, 9

[51] Amanpreet Singh, Guan Pang, Mandy Toh, Jing Huang, Wojciech Galuba, and Tal Hassner. Textocr: Towards large-scale end-to-end reasoning for arbitrary-shaped scene text. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8802–8812, 2021. 6

[52] Koustuv Sinha, Robin Jia, Dieuwke Hupkes, Joelle Pineau, Adina Williams, and Douwe Kiela. Masked language modeling and the distributional hypothesis: Order word matters pre-training for little. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2888–2913, Online and Punta Cana, Dominican Republic, Nov. 2021. Association for Computational Linguistics. 9

[53] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008. 9

[54] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. 4, 6

[55] Jianfeng Wang, Zhengyuan Yang, Xiaowei Hu, Linjie Li, Kevin Lin, Zhe Gan, Zicheng Liu, Ce Liu, and Lijuan Wang. Git: A generative image-to-text transformer for vision and language. *arXiv preprint arXiv:2205.14100*, 2022. 6, 8, 9

[56] Ronald J Williams and David Zipser. A learning algorithm for continually running fully recurrent neural networks. *Neural computation*, 1(2):270–280, 1989. 4, 5

[57] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3733–3742, 2018. 3

[58] Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. Layoutlm: Pre-training of text and layout for document image understanding. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1192–1200, 2020. 2, 8

[59] Yang Xu, Yiheng Xu, Tengchao Lv, Lei Cui, Furu Wei, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, Wanxiang Che, et al. Layoutlmv2: Multi-modal pre-training for visually-rich document understanding. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2579–2591, 2021. 2, 8

[60] Chuhui Xue, Wenqing Zhang, Yu Hao, Shijian Lu, Philip HS Torr, and Song Bai. Language matters: A weakly supervised vision-language pre-training approach for scene text detection and spotting. In *European Conference on Computer Vision*, pages 284–302. Springer, 2022. 3

[61] Zhengyuan Yang, Zhe Gan, Jianfeng Wang, Xiaowei Hu, Faisal Ahmed, Zicheng Liu, Yumao Lu, and Lijuan Wang. Unitab: Unifying text and box outputs for grounded vision-language modeling. In *Proceedings of the European Conference on Computer Vision (ECCV)*. ECCV, 2022. 3

[62] Zhengyuan Yang, Yijuan Lu, Jianfeng Wang, Xi Yin, Dinei Florencio, Lijuan Wang, Cha Zhang, Lei Zhang, and Jiebo Luo. Tap: Text-aware pre-training for text-vqa and text-caption. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8751–8761, 2021. 8

[63] Xu Zhong, Elaheh ShafieiBavani, and Antonio Jimeno Yepes. Image-based table recognition: data, model, and evaluation. In *European Conference on Computer Vision*, pages 564–580. Springer, 2020. 6, 7, 8

[64] Xu Zhong, Jianbin Tang, and Antonio Jimeno Yepes. Publaynet: largest dataset ever for document layout analysis. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1015–1022. IEEE, 2019. 7