Extending TrOCR for Text Localization-Free OCR of Full-Page Scanned Receipt Images

Hongkuan Zhang[1], Edward Whittaker[2], Ikuo Kitagishi[3]
[1] Nagoya University [2] K.K. Best Path Research [3] Money Forward, Inc.
zhang.hongkuan.k5@s.mail.nagoya-u.ac.jp[1]
ed@bestpathresearch.com[2] kitagishi.ikuo@moneyforward.co.jp[3]

Abstract

Digitization of scanned receipts aims to extract text from receipt images and save it into structured documents. This is usually split into two sub-tasks: text localization and optical character recognition (OCR). Most existing OCR models only focus on the cropped text instance images, which require the bounding box information provided by a text region detection model. Introducing an additional detector to identify the text instance images in advance adds complexity, however instance-level OCR models have very low accuracy when processing the whole image for the document-level OCR, such as receipt images containing multiple text lines arranged in various layouts. To this end, we propose a localization-free document-level OCR model for transcribing all the characters in a receipt image into an ordered sequence end-to-end. Specifically, we finetune the pretrained instance-level model TrOCR with randomly cropped image chunks, and gradually increase the image chunk size to generalize the recognition ability from instance images to full-page images. In our experiments on the SROIE receipt OCR dataset, the model finetuned with our strategy achieved 64.4 F1-score and a 22.8% character error rate (CER), respectively, which outperforms the baseline results with 48.5 F1-score and 50.6% CER. The best model, which splits the full image into 15 equally sized chunks, gives 87.8 F1-score and 4.98% CER with minimal additional pre or post-processing of the output. Moreover, the characters in the generated document-level sequences are arranged in the reading order, which is practical for real-world applications.

1. Introduction

Scanned receipt digitization aims at documenting text in receipts. This process was formally defined as Scanned Receipts OCR and Information Extraction (SROIE) task in the ICDAR 2019 competition [7], which provides a benchmark dataset, called SROIE, and splits the task into localization and recognition sub-tasks. Existing works focus on either the text detection [21, 18, 11] or character recognition [15, 3, 24], and most OCR models can only transcribe text from cropped text instance-level images as opposed to document-level receipt images. Introducing an additional detector to identify the text instance images in advance increases system complexity, and it requires post-processing to combine the instance-level sequences to obtain a document-level transcription. To this end, finetuning an instance-level OCR model to generalize its recognition ability to full page images could be more efficient for document-level OCR, while maintaining the same accuracy.

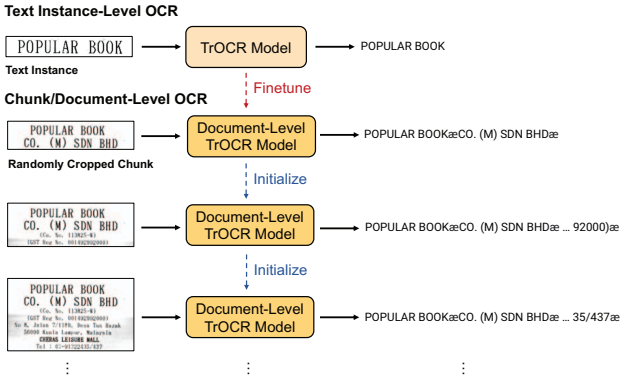

Figure 1. Our proposed step-by-step finetuning strategy for adapting TrOCR to the document-level OCR. "æ" indicates the separator token for dividing characters in each text-line.

Recently, a pretrained Transformer-based OCR model TrOCR [10] was proposed which achieved state-of-the-art performance on the SROIE dataset. However, TrOCR was trained using only text instance images, which makes its adaptation to document-level OCR challenging because of the great variation in input images, which include many more characters and more lines than the single lines it was trained on. To explore the potential of TrOCR for end-to-end document-level OCR without text localization, we propose an efficient step-by-step finetuning strategy as shown in Fig.1. Specifically, we first randomly split the whole receipt image into image chunks whose size is closer to

the original text instance images. These are then used to finetune the TrOCR model for what we call "chunk-level" OCR, and we gradually increase the chunk size to introduce more difficult chunks containing more lines and characters. Every time we use the model finetuned in the previous step to initialize the current model. Finally, we train using the entire, un-chunked, receipt images to achieve document-level OCR. The intuition behind this strategy is to progressively get the model to generalize its recognition ability to larger images. We define the order of characters in the chunk-level label as top-left to bottom-right, and propose a method to construct the reference label for each chunk automatically. We also include a text-line separator token in the constructed reference label, which aims to encode the line segmentation for the layout learning and also facilitate post-processing the model output into lines during inference.

We conduct experiments on the SROIE dataset for the document-level OCR. We cannot directly compare our results with those in the literature, since they only focus on instance-level OCR. Thus we construct two baseline finetuning methods for comparison. The experimental results show that our method achieves better performance than the two baselines on both word-level and character-level metrics. We finetuned TrOCR as well as the document understanding model Donut [9] for comparison. Both models have different input sizes, and using our method we expect we could eventually find an optimal input size in terms of accuracy and computational efficiency. The main contributions of our work are summarized as follows:

- We propose a method to construct chunk-level reference labels automatically using only annotated instance-level labels. This method can be easily applied to other OCR datasets.
- The generated characters are arranged in the reading order with a unique text-line separator token for post-processing, which is practical for real-world applications.
- We propose a step-by-step finetuning strategy to adapt TrOCR for document-level OCR, which can process the entire image and achieve competitive performance.

2. Related Works

CNN-Based OCR models Based on our target, we mainly focus on end-to-end OCR models which include two modules for detection and recognition, respectively. Previous research treats these two problems independently [1, 20, 19] by combining a text detector with a recognition model. Since the interaction between the two modules can complement each other to avoid the error propagation, recent research [5, 6, 12, 8] jointly optimizes the two modules by sharing the intermediate results. However, all these models include a text detection module explicitly or implicitly whereas we aim to perform document-level OCR without extracting intermediate text regions.

Transformer-Based OCR models A Transformer-based model TrOCR was proposed recently for the receipt OCR task. TrOCR incorporates a vision transformer and a language model in its encoder-decoder architecture, which was trained on large-scale printed and handwritten OCR data for robust text recognition. Several Transformer-based models were also proposed focusing on hand written text OCR [22, 16] or scene text OCR [2, 17, 14]. However, all these models are restricted to the transcription of the cropped text-line or instance images instead of the full images. Recently, an OCR-free document understanding model Donut [9] has been proposed lately, which includes a pseudo-OCR pre-training task to transcribe texts from document images.

3. Methodology

3.1. TrOCR Model Architecture

We will first introduce TrOCR as the backbone model for our finetuning work. TrOCR is a Transformer-based OCR model which consists of a pretrained vision Transformer encoder BEiT [4] and a pretrained language model decoder RoBERTa [13] as shown in Fig.2. To recognize characters in the cropped text instance images, the images are first resized into square boxes of size 384×384 pixels and then flattened into a sequence of 576 patches, which are then encoded by BEiT into high-level representations and decoded by RoBERTa into corresponding characters step-by-step. TrOCR was pretrained with 684M textlines in English, which ensures the robust recognition ability for characters in various formats. However, TrOCR can only handle cropped single-line text instance images, which leads to underperformance when finetuning the model directly using whole receipt images. Therefore, a better finetuning strategy to adapt the model recognition from cropped images to full images is required.

3.2. Chunk-Level OCR Finetuning

To leverage TrOCR for whole image text recognition, we propose to finetune the model with image chunks extracted from the whole images for chunk-level OCR. As Fig.2 shows, our finetuning pipeline contains three modules: (i) randomly sample image chunks from the full receipt image; (ii) construct the label for the sampled chunks; and (iii) finetune the model with chunks and corresponding chunk labels. We will introduce the random sampling and the finetuning process in this section.

Randomly sampling image chunks from whole images aims to obtain larger images for training the model. The reason for introducing randomness is that we hope to extract different chunks from each image across different epochs, which can increase data diversity and improve model gener-

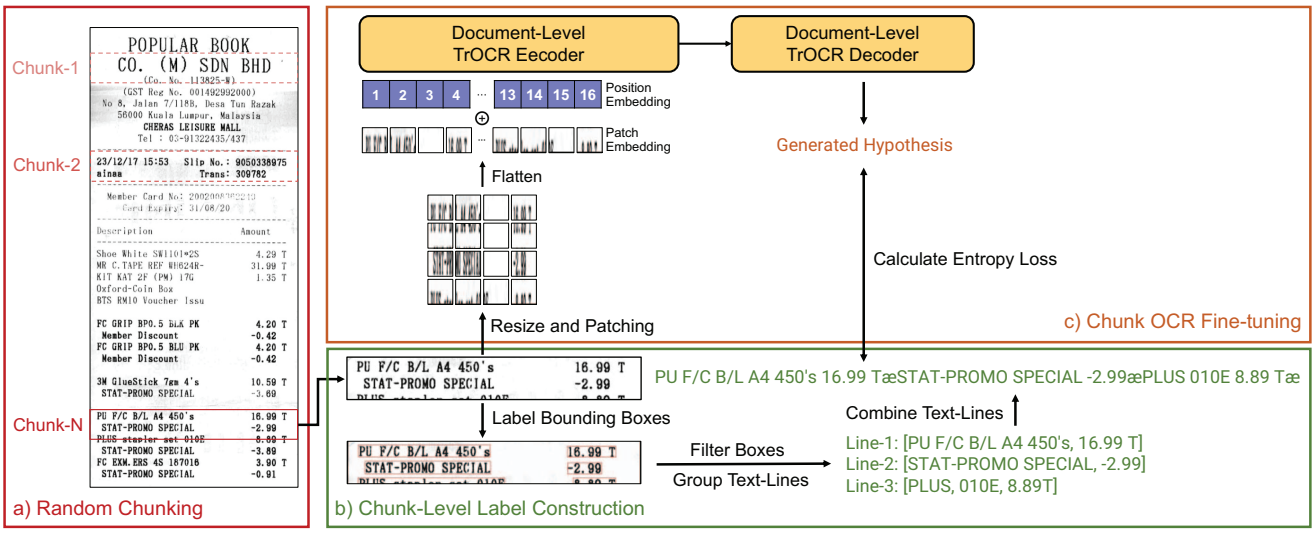

Figure 2. Finetuning the TrOCR model for chunk-level OCR.

alization. Formally, to sample a chunk from a receipt image of width W and height H, we first set a hyper-parameter L for the chunk numbers that we will split a receipt image into, then define the image chunk size whose width w is always the same as the corresponding image width W and height h equal to H/L. With the determined chunk size, we randomly select an image chunk starting point s on the y-axis whose value ranges from 0 to $H - H/L$, and crop the chunk between s and $s + h$ on the y-axis. Last, we repeat the sampling N times to extract multiple chunks from each receipt image.

With the randomly cropped chunk images, we can obtain corresponding chunk-level labels with the method that will be described in the next section, and use the chunks and labels for chunk-level OCR finetuning. The chunk size determines the contents in the chunk which in turn affect the learning difficulty. Therefore we feed the model with smaller chunks whose resolution is similar to the text instance images at the early stage, and increase the chunk size gradually to finetune the model with progressively more difficult image chunks. Concretely, we start by setting L with a large value which produces smaller chunks, then after the training is finished using the current L, we start the next stage training with smaller L, and every time we use the best model checkpoint from the previous training step to initialize the model in the current step. Finally, we increase the chunk size to the full image size (L=1) for document-level OCR. We use the notation Growing-Finetune for our proposed method.

3.3. Chunk-Level Label Construction

Chunks usually contain multiple text-lines and some characters are split vertically in half as shown in Fig.2. Therefore, to construct labels for randomly cropped chunks, a definition for the character order in the label and which split characters should be included in the label is required. In this paper, we define the top-left to bottom-right reading order for characters in the chunk as the correct order, and set a overlapping threshold θ to include characters with an overlapping rate larger than θ to make sure no unrecognizable split characters are mistakenly included in the label. Based on these definitions, we construct the chunk-level label with the use of annotated texts and corresponding bounding boxes information as shown in Algorithm 1. We first gather the I bounding boxes in the randomly cropped image chunk and filter out boxes with overlapping rate less than θ, and sort boxes as well as labels based on the y-axis values of the left-upper anchors of boxes. To align boxes horizontally in the same line, we define a merging threshold δ to merge boxes that overlap vertically over the threshold into text-line level labels, and sort the boxes in each group based on the x-axis values of the left-upper anchors from left to right. Lastly, we concatenate labels of boxes in each group with white space as text-line labels, which are concatenated with text-line label separator token "æ". This character does not appear in any of the receipts and is used to encode the line segmentation for the receipt layout learning. To clarify, we only use the boxes for the label construction and no box is used during inference.

4. Experiments

4.1. Settings

Dataset We use the SROIE dataset from the ICDAR 2019 competition for our experiments. The SROIE OCR task focuses on text recognition of cropped receipt images. There are 626 and 361 images in the training and testing set, respectively, which are annotated with ground truth bounding

Algorithm 1 Chunk-Level Label Construction

Input: overlapping threshold θ, merging threshold δ, chunk numbers to split L, chunk numbers to sample N
Output: chunk set X and chunk label set Y
Data: input image $V = W \times H$

Init $X \leftarrow \{\}, Y \leftarrow \{\}, n \leftarrow 1, h \leftarrow H/L$
for n = 1 to N **do**
  Randomly select a starting point $s \in (0, H - H/L)$
  Chunk x between s and $s + h$
  Gather boxes $B = \{b_i\}_{i=1}^{I}$ and labels $R = \{r_i\}_{i=1}^{I}$
  Sort boxes and labels by y-axis values of anchors
  for b_i in B **do** ▷ filtering boxes
    if $overlap(b_i, x) \leq \theta$ **then**
      Remove b_i from B
    end if
  end for
  Init merged boxes $B^{'} \leftarrow \{\}$ and labels $T^{'} \leftarrow \{\}$
  for b_i in B **do** ▷ merging text-line labels
    if $b_i \notin B'$ **then**
      Init text-line label set $t_i \leftarrow \{r_i\}$
      for b_j in $B - \{b_i\}$ **do**
        if $v\_overlap(b_i, b_j) \geq \delta$ **then**
          Add r_j to t_i
        end if
      end for
      Sort t_i based on x-axis values of anchors
      $t_i \leftarrow$ Concat labels in t_i with whitespace
      Add t_i to $T^{'}$
      Add b_i to $B^{'}$
    end if
  end for
  $y \leftarrow$ Concat labels in $T^{'}$ with the separator æ
  Add x to X
  Add y to Y
end for
return X, Y

boxes and corresponding texts, and we keep the train and test data split the same as the TrOCR setting. We randomly sample 60 images from the training set to construct the validation set. For the chunk-level OCR setting, the training chunks are randomly sampled from each image in the training set, while the validation and testing chunks are sequentially cropped from each image to ensure the full image area is covered for the evaluation. All the labels are obtained by the chunk-level label construction method in Section 3.3.

Hyper-parameters We use `trocr-base-printed`[1] model checkpoint finetuned with the original SROIE training data from Hugging Face [23] for our own finetuning experiments. The boxes filtering threshold θ, text-line merging threshold δ and sampled chunk number N for training

[1]https://huggingface.co/microsoft/trocr-base-printed

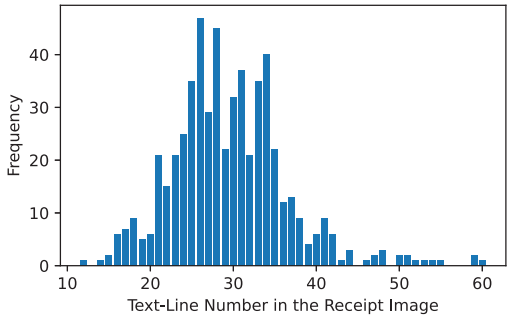

Figure 3. Distribution of the number of text lines in receipt images.

data are 0.3, 0.5, and 20, respectively, which are determined by results on the validation set. For the split chunk number L, we first compute the distribution of text-line numbers of training images as shown in Fig.3, then use the median number 30 as the initial value. By decreasing the value to 15, 7, 4, 2 and 1, we train the model with increasingly larger chunks at subsequent stages. We set the beam search size as 5 for the text generation. We also finetune the Donut model (`donut-base`[2]) which has been pretrained to read texts from a large number of document images for comparison.

Evaluation Metrics We use two evaluation metrics adopted in OCR tasks to evaluate the performance: Word-Level precision, recall, F1 (Word-Level PRF) and Character Error Rate (CER). The Word-Level PRF focuses on correctly matched words in the hypothesis without considering the word order, while the CER focuses on the character-level substitutions, deletions, and insertions as:

$$CER = (S + D + I)/(S + D + C)$$

where S, D, I, and C are the number of substitutions, deletions, insertions, and correct characters, respectively.

4.2. Baselines

Direct-Finetune This is a straight-forward method that uses the whole receipt image for the document-level OCR directly. Concretely, we resize the whole image as input and split it into patches, and finetune the model with patches and document-level labels end-to-end. Resizing the whole image to 384×384 pixels will cause a large variation in the average resolution of each character for the recognition.

Concatenate-Finetune This is a compromise strategy to keep the input resolution closer to the original setting, which splits the image equally into several chunks, and embeds each chunk into sequences that are concatenated in the temporal dimension to construct document-level inputs. Since the linearly increased input length brings higher computational cost, we restrict the split number to 4 and interpolate the position embeddings to adjust for longer inputs.

[2]https://huggingface.co/naver-clova-ix/donut-base

CER = 0.13%

Generated:
SANYU#STATIONERY#SHOPæNO.#31G&33G,#JALAN#SETIA#INDAH#X,U13/Xæ40170#SETIA#ALAMæMOBIL E#/WHATSAPPS#:#+6012-918#7937æTEL:#+603-3362#4137æGST#ID#NO:#001531760640#TAX#INVOICEæOWNED#BY#:æSANYU#SUPPLY#SDN#BHD#(11357 72-K)æCASH#SALES#COUNTERæ1.#2012-0029#RESTAURANT#ORDER#CHIT#NCRæ3.5"X6"æ3#X#2.9000#8.70#SRæTOTAL#SALES#INCLUSIVE#GST#@6%#8.70æDISCOUNT#0.00æTOTAL#8.70æROUND#ADJ#0.00æFINAL#TOTAL#8.70æCASH#10.00æCHANGE#1.30æGST#SUMMARY#AMOUNT(RM)#TAX(RM)æSR#@#6%#8.21#0.49æINV#NO:#CS-SA-0116011#DATE#:#06/10/2017æGOODS#SOLD#ARE#NOT#RETURNABLE#&#REFUNDABLEæTHANK#YOU#FOR#YOUR#PATRONAGEæPLEASE#COME#AGAIN.æTERIMA#KASIH#SILA#DATANG#LAGIæ**#PLEASE#KEEP#THIS#RECEIPT#FOR#PROVE#OFæPURCHASE#DATE#FOR#I.T#PRODUCT#WARRANTYæPURPOSE#**æFOLLOW#US#IN#FACEBOOK#:#SANYU.STATIONERYæ

Label:
SANYU#STATIONERY#SHOPæNO.#31G&33G,#JALAN#SETIA#INDAH#X,U13/Xæ40170#SETIA#ALAMæMOBIL E#/WHATSAPPS#:#+6012-918#7937æTEL:#+603-3362#4137æGST#ID#NO:#001531760640#TAX#INVOICEæOWNED#BY#:æSANYU#SUPPLY#SDN#BHD#(11357 72-K)æCASH#SALES#COUNTERæ1.#2012-0029#RESTAURANT#ORDER#CHIT#NCRæ3.5"X6"æ3#X#2.9000#8.70#SRæTOTAL#SALES#INCLUSIVE#GST#@6%#8.70æDISCOUNT#0.00æTOTAL#8.70æROUND#ADJ#0.00æFINAL#TOTAL#8.70æCASH#10.00æCHANGE#1.30æGST#SUMMARY#AMOUNT(RM)#TAX(RM)æSR#@#6%#8.21#0.49æINV#NO:#CS-SA-0116031#DATE#:#06/10/2017æGOODS#SOLD#ARE#NOT#RETURNABLE#&#REFUNDABLEæTHANK#YOU#FOR#YOUR#PATRONAGEæPLEASE#COME#AGAIN.æTERIMA#KASIH#SILA#DATANG#LAGIæ**#PLEASE#KEEP#THIS#RECEIPT#FOR#PROVE#OFæPURCHASE#DATE#FOR#I.T#PRODUCT#WARRANTYæPURPOSE#**æFOLLOW#US#IN#FACEBOOK#:#SANYU.STATIONERYæ

CER = 14.82%

Generated:
RESTORAN#WAN#SHENGæ002043319-WæNO.2,#JALAN#TEMENGGUNG#19/9,æSEKSYEN#9,#BANDAR#MAHKOTA#CHERAS,æ43200#CHERAS,#SELANGORæGST#REG#NO:#001335787520æTAX#INVOICEæINV#NO.#:#1159924#CASHIER:#THANDARæDATE#:#01-06-2018#14:20:06æDESCRIPTION#QTY#U.PRICE#TOTAL#TAXæTEH#(B)æ2#X#2.20#4.40#ZRLæTAKE#AWAYæ2#X#0.20#0.40#ZRL#U.40#ZELLæTOTAL#QTY:#4æTOTAL#(EXCLUDING#GST):#4.60æTOTAL#(INCLUSIVE#OF#GST):#4.80æTOTAL#:#4.80æCASH#:#4.80

Label:
RESTORAN#WAN#SHENGæ002043319-WæNO.2,#JALAN#TEMENGGUNG#19/9,æSEKSYEN#9,#BANDAR#MAHKOTA#CHERAS,æ43200#CHERAS,#SELANGORæGST#REG#NO:#001335787520æTAX#INVOICEæINV#NO.:#1169924#CASHIER:#THANDARæDATE#:#01-06-2018#14:20:08æDESCRIPTION#QTY#U.PRICE#TOTAL#TAXæTEH#(B)æ2#X#2.20#4.40#ZRLæTAKE#AWAYæ2#X#0.20#0.40#ZRLæTOTAL#QTY:#4æTOTAL#(EXCLUDING#GST):#4.80æTOTAL#(INCLUSIVE#OF#GST):#4.80æTOTAL#:#4.80æCASH#:#4.80æGST#SUMMARY#AMOUNT(RM)#TAX(RM)æZRL#(@#0%)#4.80#0.00æ

Figure 4. Error analysis on the generated full-page receipt OCR texts. Characters in the red, blue, and green colors represent the substitution, insertion, and deletion errors, respectively. Black characters are the corresponding correct content.

4.3. Quantitative Analysis

Performance Comparison We first compare the performance of our method and two baselines. As the results in Table 1 show, our method outperforms the two baselines on all metrics, which demonstrates the improved generalization ability for full-page images with Growing-Finetune. Moreover, the worst performance with Concatenate-Finetune also highlights the increased computational complexity on the longer inputs. Compared with the original TrOCR model, our results are worse since the evaluation for the longer and ordered document-level sequence is more strict, which also reveals the efficiency of our method to achieve good results without using the bounding box information for the localization. Furthermore, when applying our method to Donut for comparision, we see that the model achieves similar performance to the Direct-Finetune, which indicates that Growing-Finetune is more effective when the discrepancy of input image resolution between pretraining and finetuning is large.

Model	Precision	Recall	F1	CER (↓)
Instance-Level OCR				
TrOCR	96.1	96.2	96.2	0.95
Document-Level OCR w/ TrOCR				
TrOCR+Direct-Finetune	51.9	45.4	48.5	50.6
TrOCR+Concatenate-Finetune	23.5	47.2	31.3	122.5
TrOCR+Growing-Finetune (Ours)	**66.4**	**62.5**	**64.4**	**22.8**
Document-Level OCR w/ Donut				
Donut+Direct-Finetune	**91.8**	**91.4**	**91.6**	**3.1**
Donut+Growing-Finetune (Ours)	91.3	91.0	91.1	3.5

Table 1. Model performance comparison with different finetuning strategies for whole document OCR.

L	Precision	Recall	F1	CER	S	I	D	Avg. length
30	87.4	87.2	87.3	6.88	5315	7379	8761	5.9
15	**87.9**	**87.7**	**87.8**	**4.98**	**3835**	4593	**5256**	9.6
7	84.6	83.8	84.2	5.64	5468	3076	5822	18.2
4	82.6	81.9	82.2	6.04	6542	**2874**	5416	30.1
2	77.5	76.6	77.0	9.51	13057	3992	8747	57.5
1	66.4	62.5	64.4	22.8	27450	6489	20400	**107.5**

Table 2. TrOCR model performance with different chunk numbers per image. Larger number L indicates smaller chunks on average.

Chunk Size Analysis We then analyzed how the TrOCR model performance changes according to the input chunk size. Since the chunk size is determined by the chunk number L, we test the performance under different L and the variation is shown in Table 2. We observed that performance improved with increased chunk size at first, since the larger chunk size will introduce fewer split characters which reduces errors caused by the insertion and deletion, but the performance decreases when the size becomes larger, since images with more content and a longer target sequence sig-

Label:
RESTORAN#WAN#SHENGæ002043319-WæNO.2,#JALAN#TEMENGGUNG#19/9,æSEKSYEN#9,#BANDAR#MAHKOTA#CHERAS,æ43200#CHERAS,#SELANGORæGST#REG#NO:#001335787520æTAX#INVOICEæINV#NO.:#1169924#CASHIER:#THANDARæDATE#:#01-06-2018#14:20:08æDESCRIPTION#QTY#U.PRICE#TOTAL#TAXæTEH#(B)æ2#X#2.20#4.40#ZRLæTAKE#AWAYæ2#X#0.20#0.40#ZRLæTOTAL#QTY:#4æTOTAL#(EXCLUDING#GST):#4.80æTOTAL#(INCLUSIVE#OF#GST):#4.80æTOTAL#:#4.80æCASH#:#4.80æGST#SUMMARY#AMOUNT(RM)#TAX(RM)æZRL#(@#0%)#4.80#0.00æ

Generated (bs=5):
RESTORAN#WAN#SHENGæ002043319-WæNO.2,#JALAN#TEMENGGUNG#19/9,æSEKSYEN#9,#BANDAR#MAHKOTA#CHERAS,æ43200#CHERAS,#SELANGORæGST#REG#NO:#001335787520æTAX#INVOICEæINV#NO.#:#1159924#CASHIER:#THANDARæDATE#:#01-06-2018#14:20:06æDESCRIPTION#QTY#U.PRICE#TOTAL#TAXæTEH#(B)æ2#X#2.20#4.40#ZRLæTAKE#AWAYæ2#X#0.20#0.40#ZRL#U.40#ZELLæTOTAL#QTY:#4æTOTAL#(EXCLUDING#GST):#4.60æTOTAL#(INCLUSIVE#OF#GST):#4.80æTOTAL#:#4.80æCASH#:#4.80

Generated (bs=1):
RESTORAN#WAN#SHENGæ002043319-WæNO.2,#JALAN#TEMENGGUNG#19/9,æSEKSYEN#9,#BANDAR#MAHKOTA#CHERAS,æ43200#CHERAS,#SELANGORæGST#REG#NO:#001335787520æTAX#INVOICEæINV#NO.#:#1159924#CASHIER:#THANDARæDATE#:#01-06-2018#14:29:06æDESCRIPTION#QTY#U.PRICE#TOTAL#TAXæTEH#(B)æ2#X#2.20#4.40#ZRLæTAKE#AWAYæ2#X#0.20#0.40#ZRL#U.40#ZRL#0.40#ZELLæTOTAL#QTY:#4æTOTAL#(EXCLUDING#GST):#4.80æTOTAL#(INCLUSIVE#OF#GST):#4.60æTOTAL#:#4.80æCASH#:#4.80

Generated (bs=7):
RESTORAN#WAN#SHENGæ002043319-WæNO.2,#JALAN#TEMENGGUNG#19/9,æSEKSYEN#9,#BANDAR#MAHKOTA#CHERAS,æ43200#CHERAS,#SELANGORæGST#REG#NO:#001335787520æTAX#INVOICEæINV#NO.#:#1159924#CASHIER:#THANDARæDATE#:#01-06-2018#14:20:06æDESCRIPTION#QTY#U.PRICE#TOTAL#TAXæTEH#(B)æ2#X#2.20#4.40#ZRLæTAKE#AWAYæ2#X#0.20#0.40#ZRL#GOOD#ZRLæTOTAL#QTY:#4æTOTAL#(EXCLUDING#GST):#4.60æTOTAL#(INCLUSIVE#OF#GST):#4.80æTOTAL#:#4.60æCASH#:#4.80æGST#SUMMARY#AMOUNT(RM)#TAX(RM)æZRL#(@#0%)#4.80#0.00æ

Figure 5. Generated Texts with Different Beam Search Size (bs). The underlined texts are deleted contents with the default beam size.

Model	Precision	Recall	F1	CER
Growing-Finetune (L=15)	87.9	87.7	87.8	4.98
– Separator Token	89.8	88.8	89.3	5.46
– Random Sampling	77.8	76.5	77.2	11.3

Table 3. Ablation study for the factor influence analysis.

nificantly increases the learning difficulty, especially for the half-page and full-page settings which doubles the error rates compared with their previous settings. The optimal trade-off is achieved at $L = 15$, where each chunk contains roughly 2 text-lines, on average. It is worth noting that if the text detection in the baseline TrOCR were to miss 4% of ground truth regions, our best method results would be better than the original TrOCR performance.

Ablation Study Lastly, we analyze the influence of two main factors in our strategy with the ablation study as shown in Table 3. By removing the separator token, we noticed the CER performance drop is minor. While by sampling chunks sequentially without randomness, we found the performance dropped significantly on all metrics, which indicates the importance of randomness in bringing more diverse data and larger data size among all epochs for the convergence. This may also be thought of as a form of data augmentation.

4.4. Qualitative Analysis

To understand the quality of generated document-level texts, we conduct the error analysis on generated full-page OCR texts with $L = 1$ as shown in Fig.4. For the first example, our model generates a much longer sequence than instance-level TrOCR and achieved 0.13% error rate where only one character was mistakenly substituted. This indicates the model can generate high-quality texts for full-image receipts. On the other hand, for the second example, our model achieved a 14.82% error rate and made substitution errors on numbers and insertion or deletion errors on sequences. We hypothesize that the insertions are caused by memorization during model training, and the deletions are caused by the small beam size. We generate texts for the second example with the beam size 1 and 7, respectively, and the results in Fig.5 show that decoding with a larger beam size can successfully generate the missing contents which supports our hypothesis.

5. Conclusions and Future Work

In this paper, we propose a step-by-step finetuning strategy and an automatic label construction method for adapting TrOCR to perform document-level receipt image OCR. The finetuned model can handle the full image input and transcribe all characters into long ordered sequences. Moreover, our method outperforms other straight-forward finetuning baselines, which indicates the efficiency of finetuning with image chunks of increasing size. We observed the trade-off between performance and chunk size, and learned the importance of random sampling from the ablation study. Besides, our method is more effective in improving the generalization ability when the discrepancy of input image resolution between pretraining and finetuning is large, as demonstrated with experiments on the Donut model. We expect this study can serve as a baseline for future studies that concentrate on efficient finetuning methods for the document-level receipt OCR model construction.

References

[1] Ouais Alsharif and Joelle Pineau. End-to-end text recognition with hybrid hmm maxout models, 2013.

[2] Rowel Atienza. Vision Transformer for fast and efficient scene text recognition. In *International Conference on Document Analysis and Recognition (ICDAR 2021)*, pages 319–334. Springer, 2021.

[3] Jeonghun Baek, Geewook Kim, Junyeop Lee, Sungrae Park, Dongyoon Han, Sangdoo Yun, Seong Joon Oh, and Hwalsuk Lee. What is wrong with scene text recognition model comparisons? dataset and model analysis. In *Proceedings Of The IEEE/CVF International Conference on Computer Vision (ICCV 2019)*, pages 4715–4723, 2019.

[4] Hangbo Bao, Li Dong, and Furu Wei. BEiT: BERT pre-training of image transformers, 2021.

[5] Michal Busta, Lukas Neumann, and Jiri Matas. Deep textspotter: An end-to-end trainable scene text localization and recognition framework. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV 2017)*, pages 2204–2212, 2017.

[6] Tong He, Zhi Tian, Weilin Huang, Chunhua Shen, Yu Qiao, and Changming Sun. An end-to-end textspotter with explicit alignment and attention. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2018)*, pages 5020–5029, 2018.

[7] Zheng Huang, Kai Chen, Jianhua He, Xiang Bai, Dimosthenis Karatzas, Shijian Lu, and CV Jawahar. ICDAR 2019 competition on scanned receipt ocr and information extraction. In *2019 International Conference on Document Analysis and Recognition (ICDAR 2019)*, pages 1516–1520. IEEE, 2019.

[8] Max Jaderberg, Andrea Vedaldi, and Andrew Zisserman. Deep features for text spotting. In *European Conference on Computer Vision (ECCV 2014)*, pages 512–528. Springer, 2014.

[9] 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)*, pages 498–517. Springer, 2022.

[10] Minghao Li, Tengchao Lv, Jingye Chen, Lei Cui, Yijuan Lu, Dinei Florencio, Cha Zhang, Zhoujun Li, and Furu Wei. TrOCR: Transformer-based optical character recognition with pre-trained models, 2021.

[11] Minghui Liao, Zhaoyi Wan, Cong Yao, Kai Chen, and Xiang Bai. Real-time scene text detection with differentiable binarization. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI 2020)*, volume 34, pages 11474–11481, 2020.

[12] Yuliang Liu, Hao Chen, Chunhua Shen, Tong He, Lianwen Jin, and Liangwei Wang. ABCNet: Real-time scene text spotting with adaptive bezier-curve network. In *proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2020)*, pages 9809–9818, 2020.

[13] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A robustly optimized bert pretraining approach, 2019.

[14] Shota Orihashi, Yoshihiro Yamazaki, Naoki Makishima, Mana Ihori, Akihiko Takashima, Tomohiro Tanaka, and Ryo Masumura. Utilizing resource-rich language datasets for end-to-end scene text recognition in resource-poor languages. In *ACM Multimedia Asia (MM Asia 2021)*, pages 1–5. ACM, 2021.

[15] Baoguang Shi, Xiang Bai, and Cong Yao. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI 2016)*, 39:2298–2304, 2016.

[16] Phillip Benjamin Ströbel, Simon Clematide, Martin Volk, and Tobias Hodel. Transformer-based htr for historical documents, 2022.

[17] Xin Tang, Yongquan Lai, Ying Liu, Yuanyuan Fu, and Rui Fang. Visual-semantic transformer for scene text recognition, 2021.

[18] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI 2020)*, 43:3349–3364, 2020.

[19] Kai Wang, Boris Babenko, and Serge Belongie. End-to-end scene text recognition. In *2011 International Conference on Computer Vision (ICCV 2011)*, pages 1457–1464. IEEE, 2011.

[20] Tao Wang, David J Wu, Adam Coates, and Andrew Y Ng. End-to-end text recognition with convolutional neural networks. In *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*, pages 3304–3308. IEEE, 2012.

[21] Wenhai Wang, Enze Xie, Xiaoge Song, Yuhang Zang, Wenjia Wang, Tong Lu, Gang Yu, and Chunhua Shen. Efficient and accurate arbitrary-shaped text detection with pixel aggregation network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV 2019)*, pages 8440–8449, 2019.

[22] Christoph Wick, Jochen Zöllner, and Tobias Grüning. Transformer for handwritten text recognition using bidirectional post-decoding. In *International Conference on Document Analysis and Recognition (ICDAR 2021)*, pages 112–126. Springer, 2021.

[23] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations (EMNLP 2020)*, pages 38–45, 2020.

[24] Deli Yu, Xuan Li, Chengquan Zhang, Tao Liu, Junyu Han, Jingtuo Liu, and Errui Ding. Towards accurate scene text recognition with semantic reasoning networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2020)*, pages 12113–12122, 2020.