Efficient OCR for Building a Diverse Digital History

Jacob Carlson[1], Tom Bryan[1], Melissa Dell[1,2],
[1]Harvard University, Cambridge, MA, USA
[2]National Bureau of Economic Research, Cambridge, MA, USA
{jacob_carlson,tom_bryan,melissadell}@fas.harvard.edu

Abstract

Many users consult digital archives daily, but the information they can access is unrepresentative of the diversity of documentary history. The sequence-to-sequence architecture typically used for optical character recognition (OCR) – which jointly learns a vision and language model – is poorly extensible to low-resource document collections, as learning a language-vision model requires extensive labeled sequences and compute. This study models OCR as a character level image retrieval problem, using a contrastively trained vision encoder. Because the model only learns characters' visual features, it is more sample efficient and extensible than existing architectures, enabling accurate OCR in settings where existing solutions fail. Crucially, it opens new avenues for community engagement in making digital history more representative of documentary history.

1 Introduction

Digital texts are central to the study, dissemination, and preservation of human knowledge. Tens of thousands of users consult digital archives daily in Europe alone (Chiron et al., 2017), yet billions of documents remain trapped in hard copy in libraries and archives around the world. These documents contain extremely diverse character sets, languages, fonts or handwriting, printing technologies, and artifacts from scanning and aging. Converting them into machine-readable data that can power indexing and search, computational textual analyses, and statistical analyses - and be more easily consumed by the public - requires highly extensible, accurate, efficient tools for optical character recognition (OCR).

Current predominant OCR technology – developed largely for small-scale commercial applications in high resource languages – falls short of these requirements. OCR is typically modeled as a sequence-to-sequence (seq2seq) problem, with learned embeddings from a neural vision model taken as inputs to a learned neural language model. The seq2seq architecture is challenging to extend and customize to novel, lower resource settings (Hedderich et al., 2021), because training a vision-language model requires a vast collection of labeled image-text pairs and significant compute. This study shows that on printed Japanese documents from the 1950s, the best performing existing OCR mis-predicts over half of characters. Poor performance is widespread, spurring a large post-OCR error correction literature (Lyu et al., 2021; Nguyen et al., 2021; van Strien. et al., 2020) and skewing digital history towards limited settings that are not representative of the diversity of documentary history.

This study develops a novel, open source OCR architecture, EffOCR (**Eff**icient**OCR**), designed for researchers and archives seeking a sample-efficient, customizable, scalable OCR solution for diverse documents. EffOCR combines the simplicity of early OCR systems, such as Tauschek's 1920s reading machine, with deep learning, bringing OCR back to its roots: the *optical* recognition of *characters*. Deep learning-based object detection methods are used to localize individual characters or words in the document image. Character (word) recognition is modeled as an image retrieval problem, using a vision encoder contrastively trained on character (word) crops.

EffOCR performs accurately, even when using lightweight models designed for mobile phones that are cheap to train and deploy. Using documents that are fundamental to studying Japan's remarkable 20th century economic growth, the study shows EffOCR can provide a sample efficient, highly accurate OCR architecture for contexts where all current solutions fail. EffOCR's blend of accuracy and efficient runtime also makes it attractive for digitizing massive-scale collections

Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 8105–8115

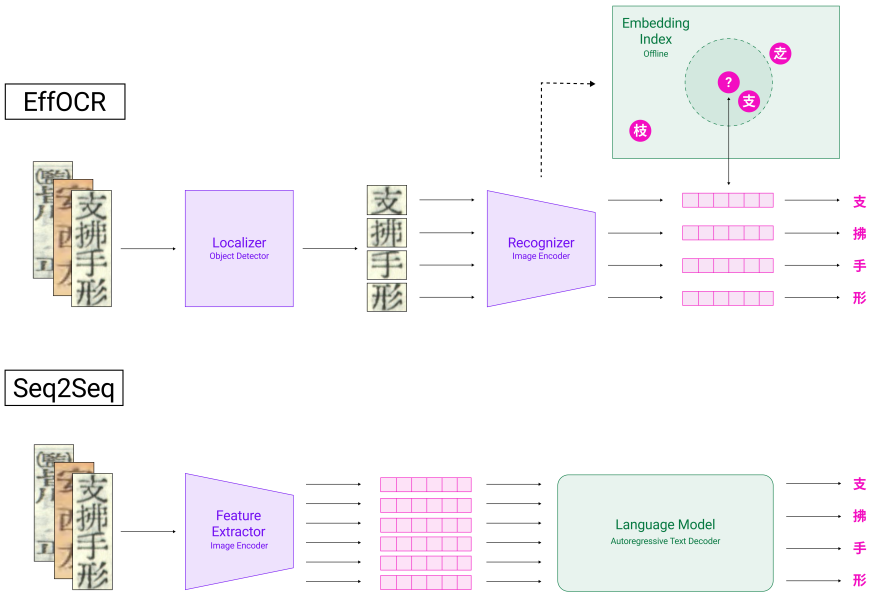

Figure 1: **EffOCR and Seq2Seq Model Architectures.** This figure represents the EffOCR architecture, as compared to a typical sequence-to-sequence OCR architecture.

in high resource languages, which the study illustrates with Library of Congress's collection of historical U.S. newspapers (Library of Congress, 2022). EffOCR has been used to cheaply and accurately digitize the over 20 million page scans in this collection (Dell et al., 2023).

In principle, contextual understanding could be extremely valuable to OCR, but in practice state-of-the-art transformer seq2seq models are extremely costly to train, expensive to deploy, and do not exist for lower resource languages, with advances concentrated in a handful of languages. This study shows that taking a step back from seq2seq models unlocks massive gains in sample efficiency. Researchers, with a modest number of annotations and modest compute, can train their own OCR for settings where all existing solutions fail, using our user-friendly EffOCR open-source package. New characters specific to a setting can also be added at inference time – since they don't need to be seen in sequence during training – important for contexts such as archaeology and certain historical applications where new characters are regularly encountered. These features facilitate making digital history more representative of documentary history.

2 Methods

Modern OCR overwhelmingly uses deep neural networks – either a convolutional neural network (CNN) or vision transformer (ViT) – to encode images. The representations created by passing an input image through a neural encoder are then decoded to the associated text.

Figure 1 underscores two fundamental differences between EffOCR and seq2seq. First, sequence-to-sequence architectures typically require line level inputs, and individual characters or words are not localized; rather, images or their representations are divided into fixed size patches. In contrast, EffOCR localizes characters and words using modern object detection methods (Cai and Vasconcelos, 2018; Jocher, 2020) via the "localizer" module. Second, seq2seq sequentially decodes the learned image representations into text using a learned language model that takes the image representations as inputs. In contrast, EffOCR recognizes text by using contrastive training (Khosla et al., 2020) to learn a meaningful metric space for character or word-level OCR. A vision encoder, the "recognizer" module, projects crops of the same character (word) – regardless of style – nearby, whereas crops of different characters (words) are projected further apart.

EffOCR thus generates full lines of text in the following way: (1) the localizer produces bounding boxes for characters (words) in the input image; (2) these localized character (word) images are embedded with the recognizer; (3) the character (word) embeddings are decoded to machine-readable text in parallel by retrieving the label of their nearest neighbor in an offline index of exemplar character

(word) embeddings, created by rendering labeled character (word) images with a digital font; and (4) the bounding boxes from the localizer are re-used to robustly infer the order of the machine-readable characters (words) and the presence of white spaces. Embedding distances are computed using cosine similarity with a Facebook Artificial Intelligence Similarly Search (FAISS) backend (Johnson et al., 2019). The vision embeddings alone are sufficient to infer text since they represent characters – not text lines like in seq2seq – and hence decoding them does not require a language model with learned parameters.

This study develops both character and word level OCR models, with the former being more suitable for character-based languages and the latter more suitable for alphabet-based languages. When modeling OCR as a word level problem, EffOCR defaults to character level recognition if the distance between a word crop embedding and the nearest embedding in the offline dictionary of word embeddings is below a threshold cosine similarity. This is important, as hyphenated words at the end of lines, acronyms, proper nouns, and antiquated terms often make it infeasible to construct a comprehensive word dictionary.

EffOCR is trained on digital font renders, along with a modest number of labeled crops from target datasets. The recognizer is trained using the Supervised Contrastive ("SupCon") loss function (Khosla et al., 2020), a generalization of the InfoNCE loss (Oord et al., 2018) that allows for multiple positive and negative pairs for a given anchor. We use the "outside" SupCon loss formulation,

$$\mathcal{L}_{\text{out}}^{\text{sup}} = \sum_{i \in I} \frac{-1}{|P(i)|} \sum_{p \in P(i)} \log \frac{\exp\left(\boldsymbol{z}_i \cdot \boldsymbol{z}_p / \tau\right)}{\sum_{a \in A(i)} \exp\left(\boldsymbol{z}_i \cdot \boldsymbol{z}_a / \tau\right)}$$

as implemented in PyTorch Metric Learning (Musgrave et al., 2020), where τ is the temperature, i indexes a sample in a "multiviewed" batch (in this case multiple fonts/augmentations of characters with the same identity), $P(i)$ is the set of indices of all positives in the multiviewed batch that are distinct from i, $A(i)$ is the set of all indices excluding i, and z is an embedding of a sample in the batch.

To create training batches for the recognizer, EffOCR uses a custom m per class sampling algorithm without replacement. This metric learning batch sampling algorithm also implements batching and training with hard negatives, where the negative samples in a batch are selected to be semantically close to one another, and thus contrasts made between anchors and hard negatives may be especially informative.

Different vision encoders can be used interchangeably for the EffOCR character localizer - which locates the character/word crops – and recognizer – which learns a metric space for these crops. Three models are considered for character level EffOCR: a vision transformer model (EffOCR-T Base) with XCiT (Small) (Ali et al., 2021) for both the localizer and recognizer, a convolutional base model (EffOCR-C Base) with ConvNeXt (Tiny) (Liu et al., 2022) for both the localizer and recognizer, and a convolutional small model (EffOCR-C Small), which uses lightweight architectures designed for mobile phones – YOLOv5 (Small) (Jocher, 2020) for the localizer and MobileNetV3 (Small) for the recognizer. For word level OCR, we develop EffOCR-Word (Small), which uses the same lightweight architectures as EffOCR-C (Small). EffOCR-Word (Small) defaults to EffOCR-C (Small) when the cosine similarity between a word crop embedding and the nearest embedding in the offline word embedding dictionary is below 0.82, a hyperparameter that is (like all model hyperparameters) tuned on the validation set. The base models use a two-stage object detector for character localization, specifically a Cascade R-CNN (Cai and Vasconcelos, 2019), whereas the small models use one-stage object detection for faster speed (Jocher, 2020). The supplementary materials describe the EffOCR architecture and training recipes with no detail spared and evaluate models using alternative vision transformer encoders.

3 Related Literature

EffOCR's architecture draws inspiration from metric learning methods for efficient image retrieval (El-Nouby et al., 2021), joining a recent literature on self-supervision through simple data augmentation for image encoders (Grill et al., 2020; Chen et al., 2021; Chen and He, 2021). The closest frameworks to EffOCR in their overall design are the original OCR conceptualizations, such as Tauschek's 1920s reading machine, which used human engineered features to recognize localized characters. More recently, CharNet (Xing et al., 2019), developed for scene text (not documents), uses separate convolutional networks for dense classification and regression at a single scale, outputting a character

class and bounding box at every spatial location, and then aggregates this information with confidence scores to make final predictions. EffOCR in contrast deploys widely used, highly optimized object detection methods to localize characters and then feeds character crops to a contrastively trained recognizer.[1] Other OCR frameworks - that are widely used, have state-of-the-art performance, or provide an instructive architectural contrast with EffOCR - are described in Section 5, which introduces the comparisons that we will make.

4 Training and Evaluation datasets

Evaluating EffOCR requires benchmark datasets that are representative of the diversity of documentary history. Traditional OCR benchmarks focus on commercial applications like receipts (Huang et al., 2019) - and SOTA OCR systems evaluate on these data - which are not relevant to digital history.

Instead, the study draws on the literature on historical image datasets (Nikolaidou et al., 2022). First, it uses documents from historical Japan that can elucidate fundamental questions that have been understudied due to a lack of digital data, such as the drivers of Japan's rapid transformation from a poor agrarian economy to a wealthy industrialized nation. Horizontally and vertically written tabular data – providing rich information on Japanese firms and their personnel – are drawn from two 1950s publications (Jinji Koshinjo, 1954; Teikoku Koshinjo, 1957). A 1930s prose publication providing detailed biographies of tens of thousands of individuals (Jinji Koshinjo, 1939) is also examined. These texts could use over 13,000 *kanji* characters.

The second context is Library of Congress's Chronicling America (LoCCA) collection, which contains over 19 million historical public domain newspaper page scans. This collection is highly diverse, as shown in Figure 2.

Library of Congress provides an OCR, but the quality is low (Smith et al., 2015). There is a large literature studying historical newspapers at scale, which overwhelmingly uses keyword search and does not unlock the power of large language models due to poor quality digitization (Hanlon and Beach, 2022). LoCCA elucidates how EffOCR: 1) performs in the highest resource setting, English; 2) extensibility across Latin and *kanji* characters, which differ significantly in their aspect ratios and complexity; 3) extensibility to the many Unicode renderable languages that use the Latin script.

Layout datasets exist for Chronicling America and some of the Japanese publications (Shen et al., 2020; Lee et al., 2020). Adding word/character bounding boxes and transcription annotations builds upon the existing work of the historical image dataset literature (Nikolaidou et al., 2022).

Because seq2seq requires lines as inputs, to build the Japanese and Chronicling America datasets we draw lines at random from the Japanese volumes and from 10 randomly selected newspapers in LoCCA. Lines correspond to cells in tables and single lines within columns/rows in prose. The baseline training sets range from 291 lines for Chronicling America to 1309 cells for horizontal Japanese, highly feasible for researchers to label in an afternoon, and also includes validation and test splits. The annotations were double-entered by the study authors, with all discrepancies hand-resolved. While the randomly selected lines/table cells in the labeled data can contain names, the underlying images are already public.

For the newspapers, we also provide an additional evaluation-only dataset that consists of a sample of 225 textlines, randomly drawn from all scans in the Chronicling America collection published on March 1st of years ending in "6," from 1856-1926. This sample is balanced across these decades, with 25 textlines sampled randomly from each of the days. A selection of textlines from this set is shown in Figure 2. The day-per-decade set is designed to be challenging, by weighting older, much harder to read scans from the mid-19th century equally despite their relative scarcity in the Chronicling America collection.

In addition to this gold quality data, we create silver quality training data for training EffOCR-Word (Small) by applying the EffOCR-C (Small) model to a random sample of newspapers. We limited the number of crops with model-generated labels to 20 – so each word can have 0-20 silver-quality crops depending upon its frequency of occurrence in our random sample. This limit is binding for common words, *e.g.,* "the." We also use the gold word crops from the 291 line training set, which cover only a small share of words. Using silver quality data leads to high performance, achieved essentially for free. The study's datasets are publicly released.

Finally, we examine EffOCR on an existing Polytonic Greek benchmark (Gatos et al., 2015), se-

[1] Others have also used contrastive learning for OCR, in particular (Aberdam et al., 2021) use a self-supervised, sequence-to-sequence contrastive learning approach.

Figure 2: **Diversity in the Chronicling America Dataset.** This figure shows examples sampled from the Chronicling America (LoCCA) dataset, along with EffOCR predicted transcriptions.

lected because it contains both line-level and word transcriptions. Polytonic Greek uses five diacritics to notate older Greek texts. It is challenging because the diacritics have a similar appearance. The supplemental materials show example documents from all benchmarks.

5 Measurement and comparisons

OCR accuracy is measured using the character error rate (CER), the Levenshtein distance between the OCR'ed string and the ground truth, normalized by the length of the ground truth. A CER of 0.5, for instance, translates to mispredicting approximately half of characters.

The most widely used OCR engines are commercial products that do not currently support fine-tuning and have proprietary architectures. The study compares EffOCR to Google Cloud Vision (GCV) and Baidu OCR (popular for Asian languages). We include these comparisons because they are relevant to practitioners.

We also consider four open source architectures: EasyOCR's convolutional recurrent neural network (CRNN) framework (Shi et al., 2016), TrOCR's sequence-to-sequence encoder-decoder transformer (base and small) (Li et al., 2021), Tesseract's bi-directional LSTM, and PaddleOCR's Single Vision Text Recognition (SVTR), which also abandons seq2seq, dividing text images into small (non-character) patches, using mixing blocks to perceive inter- and intra-character patterns, and recognizing text by linear prediction (Du et al., 2022). A large literature has examined a variety of custom-designed OCR systems. We focus on those that either (1) make similar architectural choices (SVTR), (2) are considered SOTA, regardless of architectural choices (TrOCR), or (3) are very popular (Tesseract and EasyOCR).

The pre-trained EasyOCR, PaddleOCR, and TrOCR models are fine-tuned on the same target data as EffOCR. Considerable resources have been devoted to pre-training these models. For example, TrOCR was pre-trained on 684 million English synthetic text lines. Hence, these comparisons elucidate performance when these pre-trained models are further tuned on the target datasets. For a more apples-to-apples comparison, the study examines the accuracy of these architectures when trained from scratch (using a pre-trained checkpoint not trained for OCR, when supported by the architecture) on 8,000 synthetic text lines (like EffOCR) and the same target crops. EasyOCR and PaddleOCR do not support vertical Japanese, and TrOCR does not support any Japanese. Tesseract offered little support for fine-tuning until recently and hence most of its applications have been off-the-shelf, which is this study's focus. All results come from a single model run, with training details provided in the supplemental materials.

6 Results

EffOCR provides a highly accurate OCR with minimal training data, in contexts where current solutions fail. For vertical Japanese tables, the best EffOCR CER is 0.7% (Table 1). The next best alternative, Baidu OCR, has a CER of 55.6%, making nearly 80 times more errors. The best EffOCR CER is modestly higher for the Japanese prose (2.7%); these scans are low resolution and some characters are illegible, to provide a context where OCR with language modeling could offer a clear advantage. Yet EffOCR makes 5 times fewer er-

Model/Engine	Seq2Seq?	Transformer?	Pretraining	Parameters	Character Error Rate						Lines/second	
					Horiz. Jap.	Vertical Jap. (tables)	Vertical Jap. (prose)	Chron. Amer. Eval	Chron. Amer. Day/Decade	Anci. Greek	Horiz. Jap.	Chron. Amer.
EffOCR-C (Base)	×	×	from scratch	112.5 M	**0.006**	**0.007**	0.030	0.023	0.062	0.049	0.79	0.49
EffOCR-C (Small)	×	×	from scratch	9.3 M	0.010	0.009	0.036	0.028	0.080	0.052	19.46	13.40
EffOCR-T (Base)	×		from scratch	101.8 M	0.009	0.007	**0.027**	0.022	0.059	**0.047**	0.19	0.31
EffOCR-Word (Small)	×	×	from scratch	10.6 M	-	-	-	0.015	0.043	-	-	21.36
Google Cloud Vision OCR	?	?	off-the-shelf	?	0.173	0.695	0.135	**0.005**	**0.019**	0.065	?	?
Baidu OCR	?	?	off-the-shelf	?	0.060	0.556	0.177	-	-	-	?	?
Tesseract OCR (Best)		×	off-the-shelf	1.4 M	1.021	0.996	0.744	0.106	0.170	0.251	4.90	4.47
EasyOCR CRNN		×	off-the-shelf	3.8 M	0.191	-	-	0.170	0.274	-	33.55	19.80
			fine-tuned		0.082	-	-	0.036	0.157			
			from scratch		0.132	-	-	0.131	0.204			
PaddleOCR SVTR	×	×	off-the-shelf	11 M	0.085	-	-	0.304	0.314	-	13.34	13.56
			fine-tuned		0.032	-	-	0.103	0.129			
			from scratch		0.097	-	-	0.104	0.138			
TrOCR (Base)			off-the-shelf	334 M	-	-	-	0.015	0.038	-	-	0.43
			fine-tuned		-	-	-	0.013	0.027			
			from scratch		-	-	-	0.809	0.831			
TrOCR (Small)			off-the-shelf	62 M	-	-	-	0.039	0.121	-	-	0.97
			fine-tuned		-	-	-	0.075	0.091			
			from scratch		-	-	-	0.773	0.820			

Table 1: **Baseline Results and Comparisons.** This table reports the performance of different OCR architectures, *off-the-shelf* (without fine-tuning on target data), *fine-tuned* on the target publication training set from a pre-trained OCR checkpoint, and trained *from scratch* on synthetic text lines and the target publication training set. "?" indicates that the field is unknown due to the proprietary nature of the architecture.

rors than the next best alternative (GCV), whose CER of 13.5% will not support applications that require high accuracy. For horizontal Japanese – a higher resource setting – the EffOCR CER is 0.6%, whereas the next-best-alternative (Paddle OCR fine-tuned on target crops) makes more than five times more errors. The different EffOCR models produce strikingly similar results, despite the significant differences in architecture (convolutional versus transformer) and model size (9.3M to 112.5M parameters). By making an accurate digitization of such collections feasible - with minimal training data requirements - EffOCR can contribute to the diversity of digital texts available to researchers.

The CER (uncased) for the LoCCA newspapers is 1.5%. GCV has the best performance (0.5%), followed by fine-tuned TrOCR (Base) (1.3% CER). The advantage of EffOCR on English - the quintessential high resource setting - is its open-source codebase and fast runtime. GCV makes significant layout errors when fed full newspaper page scans, which have complex layouts (Shen et al., 2021), and hence the performance in Table 1 cannot be replicated when it is fed scans. GCV charges per image, and the supplementary materials estimate a cost at current prices of $23 million USD to digitize LoCCA at the line image level, versus $60K for EffOCR-Word (Small), which researchers have used to cheaply and accurately digitize this collection (Dell et al., 2023).

Table 1 examines CPU runtime for open source architectures, measured by lines processed per second on identical dedicated hardware (four 2200 MHz CPU cores, selected to represent a plausible and relatively affordable research compute setup). GPUs are prohibitively costly for mass digitization. EffOCR-Word (Small) is 50 times faster than TrOCR (Base), which is likely to be cost prohibitive for larger scale applications. EffOCR supports inference parallelization across characters – promoting faster inference – whereas seq2seq requires autoregressive decoding. On English, the most plausible scalable alternative is fine-tuned EasyOCR. With a third of the parameters of EffOCR-Word (Small), it is slightly slower and the CER is around 29% higher. For horizontal Japanese, EffOCR-C (Small) is three times more accurate and faster than PaddleOCR SVTR (fine-tuned), the next best alternative.

Figure 3 provides representative examples of errors, showing the target crop, the localized crop, and its five nearest neighbors, with the correct pre-

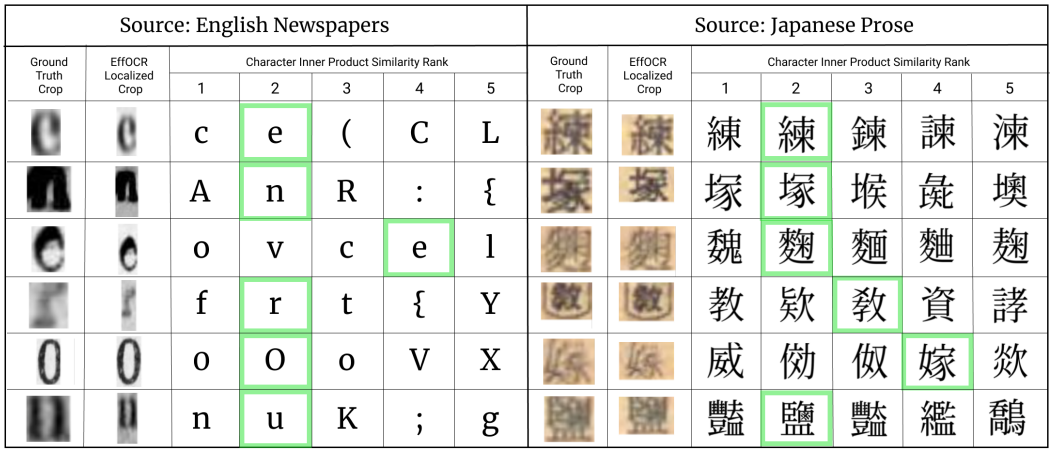

Figure 3: **Error Analysis.** Representative examples of EffOCR errors, showing the target crop, the EffOCR localized crop, and the five nearest characters in the embedding index, with the correct character highlighted in green.

diction highlighted in green. Errors tend to occur when the character is illegible or homoglyphic to another character (*e.g.,* O and 0). For example, a 0 in one font can occasionally be indistinguishable from an O in another, an error that would be straightforward to correct in post-processing.

The supplementary materials report results from additional encoders, and examine how different architecture and design choices for EffOCR contribute to its performance. In particular, we notice little difference between the best performing CNN encoders and vision transformer encoders in terms of CER, regardless of language, when holding approximately constant the number of model parameters. This is consistent with an existing literature on the convergent performances of (appropriately modernized) CNNs and vision transformers (Liu et al., 2022).

EffOCR outperforms all other architectures that support Polytonic Greek, including Google Cloud Vision. This illustrates the versatility of the architecture.

EffOCR's parsimonious architecture allows it to learn efficiently. To quantify this, we train different OCR models from scratch using varying amounts of annotated data. All architectures are pre-trained from scratch on 8,000 synthetic text lines, starting from pre-trained checkpoints not customized for OCR when supported by the framework. They are then fine-tuned on the study's benchmark datasets, with varying train splits: 70%, 50%, 20%, 5%, and 0% (using only synthetic data). These exercises are performed for Chronicling America and horizontal Japanese, as vertical Japanese is not supported by the comparison architectures.

Figure 4 plots the percentage of the benchmark dataset used in training on the x-axis and the CER on the y-axis. On just 99 labeled table cells for Japanese and 21 labeled rows for LoCCA (the 5% train split), EffOCR's CER is around 4%, showing viable few shot performance. The other architectures remain unusable. EffOCR performs nearly as well using 20% of the training data as using 70%, where it continues to outperform all other alternatives.

Here, our focus is on the design of bespoke, efficient models for low-resource contexts. One might wish to assess how EffOCR performs on completely out-of-domain texts. Elsewhere, researchers have used the EffOCR package and EffOCR-Word (Small) model trained only on newspapers to process randomly selected, highly diverse documents from the U.S. National Archives (Bryan et al., 2023). EffOCR performs similarly to other open-source OCR engines, achieving a CER of 11.2% as compared with a 11.8% CER from Tesseract (Best), a 12.1% CER from EasyOCR, and a 51% CER from TrOCR (Small). The sample efficiency of EffOCR suggests it could be trained to perform well off-the-shelf on diverse archival documents by labeling a small number of samples across a wide range of common historical document types, an effort that could be crowd-sourced.

7 Discussion

Indexing, analyzing, disseminating, and preserving diverse documentary history requires community engagement of stakeholders with the requi-

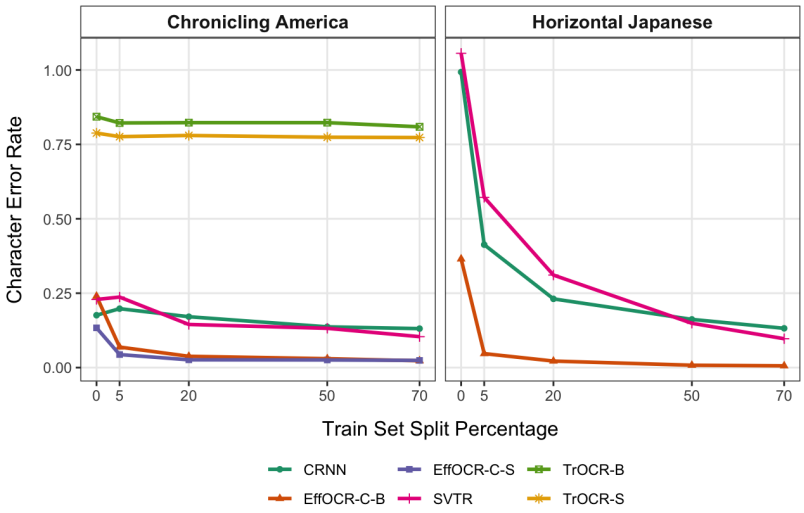

Figure 4: **Sample Efficiency.** This figure plots the percentage of the benchmark dataset used in training against the character error rate, for different OCR model architectures.

site fine-grained knowledge of the relevant settings. EffOCR facilitates this engagement because it is highly extensible to low-resource settings, sample-efficient to customize, and simple and cheap to train and deploy. In contrast, seq2seq is more aligned with the commercial objective of designing a product that is difficult for competitors to imitate. For example, EffOCR can be trained in the cloud with free student compute credits, whereas TrOCR required training on a multi-million dollar cluster with 32 32GB V100 cards. Lower resource languages may lack the pre-trained language models required to initialize a transformer seq2seq model, and sufficient compute resources are also unlikely to be available. EffOCR encourages community engagement by integrating the follow features:

Character/word level: EffOCR creates semantically rich visual embeddings of individual characters (words), a parsimonious problem. Annotators can select which of the most probable predictions from the pre-trained recognizer are correct, potentially using a simple mobile interface, or line level labels can be mapped to the character (word) level once a localizer has been developed.

Language Extensibility: Language modeling advances have concentrated around less than two dozen modern languages, out of many thousands (Joshi et al., 2020). Omitting the language model makes EffOCR extensible and easy-to-train. To extend EffOCR to a new language, all one needs are renders for the appropriate character set. Additionally, characters do not need to be seen in sequence during training, so new characters can be added at inference time, valuable for archaeological contexts where new characters are regularly discovered. Omitting the language model makes it easy to mix scripts, necessary for some languages. The recognizer can also be exposed to characters in training using any desired sequencing. This is not true of multilingual seq2seq training, which leads to many OCR errors with endangered languages (Rijhwani et al., 2020).

Decoupling localization and recognition: Theoretically, localization and recognition (akin to classification) may rely on different features of the image, suggesting modularity (Song et al., 2020). Practically, decoupling allows localization and recognition to use different training sets, economizing on annotation costs since these tasks can require very different numbers of labels depending on the script. It also encourages community innovation and future-proofness, because it simplifies training recipes and makes it straightforward to swap in new localizers or recognizers - including zero-shot models such as Kirillov et al. (2023) - as the literature advances.

Scalable: The small EffOCR models achieve fast CPU inference that can scale cheaply to hundreds of millions of documents.

Open-Source: The open-source EffOCR python package (Bryan et al., 2023) makes it straightforward to use existing EffOCR models off-the-shelf with just a few lines of code, including for those who lack familiarity with deep learning frame-

works. It also includes functionality to train custom models and guides users with tutorials.

8 Reproducibility

We release all code and training data used to create EffOCR. Scripts in the public repository exactly reproduce the figures cited above. All other material needed to reproduce these results is detailed in the supplemental materials, including training hyperparmeters. The models in this paper can also be deployed through the open-source EffOCR python package (CC-BY 4.0 license).

9 Limitations

This study does not focus on handwriting due to space constraints, but the approach would be analogous. Synthetic handwriting generators, *e.g.,* Bhunia et al. (2021), could provide extensive data for pre-training, analogous to this study's use of digital fonts.

There are some settings where EffOCR's framework is not suitable. If large portions of a document are illegible, context is necessary. Moreover, the heavy use of ligatures and/or slanting in some character sets and handwriting could lead to more challenging character localization. This challenge is mitigated with the word-level EffOCR model.

10 Ethical Considerations

EffOCR presents no major ethical concerns. Its methods are entirely open source, and its training data are entirely in the public domain. Its core functionality, accurately transcribing texts in low-resource settings, is ethically sound. By making it easier to digitize scanned document texts in low-resource settings, it can promote the inclusion of more diverse groups in NLP, social science, and humanities research. Its sample and computational efficiency minimizes environmental harm by reducing compute requirements at training and inference time.

Some applications of EffOCR could raise ethical flags. We discourage users from applying EffOCR to copyrighted documents unless the application is protected by fair use. While EffOCR is a potentially useful tool for studying bias, *e.g.,* through analyses of historical documents, potentially harmful or offensive content transcribed by EffOCR should not be shared without proper context.

References

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

Alaaeldin Ali, Hugo Touvron, Mathilde Caron, Piotr Bojanowski, Matthijs Douze, Armand Joulin, Ivan Laptev, Natalia Neverova, Gabriel Synnaeve, Jakob Verbeek, et al. 2021. Xcit: Cross-covariance image transformers. *Advances in neural information processing systems*, 34.

Ankan Kumar Bhunia, Salman Khan, Hisham Cholakkal, Rao Muhammad Anwer, Fahad Shahbaz Khan, and Mubarak Shah. 2021. Handwriting transformers. *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1086–1094.

Tom Bryan, Jacob Carlson, Abhishek Arora, and Melissa Dell. 2023. EfficientOCR: An extensible, open-source package for efficiently digitizing world knowledge. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 579–596, Singapore. Association for Computational Linguistics.

Zhaowei Cai and Nuno Vasconcelos. 2018. Cascade r-cnn: Delving into high quality object detection. *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6154–6162.

Zhaowei Cai and Nuno Vasconcelos. 2019. Cascade r-cnn: High quality object detection and instance segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(5):1483–1498.

Xinlei Chen and Kaiming He. 2021. Exploring simple siamese representation learning. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15750–15758.

Xinlei Chen, Saining Xie, and Kaiming He. 2021. An empirical study of training self-supervised vision transformers. *arXiv preprint arXiv:2104.02057*.

Guillaume Chiron, Antoine Doucet, Mickaël Coustaty, Muriel Visani, and Jean-Philippe Moreux. 2017. Impact of ocr errors on the use of digital libraries: Towards a better access to information. In *Proceedings of the 17th ACM/IEEE Joint Conference on Digital Libraries*, JCDL '17, page 249–252. IEEE Press.

Melissa Dell, Jacob Carlson, Tom Bryan, Emily Silcock, Abhishek Arora, Zejiang Shen, Luca D'Amico-Wong, Quan Le, Pablo Querubin, and Leander Heldring. 2023. American stories: A large-scale structured text dataset of historical us newspapers. *NeurIPS, Datasets and Benchmark Track, PMLR*.

Yongkun Du, Zhineng Chen, Caiyan Jia, Xiaoting Yin, Tianlun Zheng, Chenxia Li, Yuning Du, and Yu-Gang Jiang. 2022. Svtr: Scene text recognition with a single visual model. *arXiv preprint arXiv:2205.00159*.

Alaaeldin El-Nouby, Natalia Neverova, Ivan Laptev, and Hervé Jégou. 2021. Training vision transformers for image retrieval. *arXiv preprint arXiv:2102.05644*.

Basilis Gatos, Nikolaos Stamatopoulos, Georgios Louloudis, Giorgos Sfikas, George Retsinas, Vassilis Papavassiliou, Fotini Sunistira, and Vassilis Katsouros. 2015. Grpoly-db: An old greek polytonic document image database. In *2015 13th international conference on document analysis and recognition (ICDAR)*, pages 646–650. IEEE.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. 2020. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*.

W Walker Hanlon and Brian Beach. 2022. Historical newspaper data: A researcher's guide and toolkit.

Michael A. Hedderich, Lukas Lange, Heike Adel, Jannik Strötgen, and Dietrich Klakow. 2021. A survey on recent approaches for natural language processing in low-resource scenarios. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2545–2568, Online. Association for Computational Linguistics.

Zheng Huang, Kai Chen, Jianhua He, Xiang Bai, Dimosthenis Karatzas, Shijian Lu, and CV Jawahar. 2019. Icdar2019 competition on scanned receipt ocr and information extraction. *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1516–1520.

Jinji Koshinjo. 1939. *Jinji koshinroku*. Jinji Koshinjo.

Jinji Koshinjo. 1954. *Nihon shokuinrokj*. Jinji Koshinjo.

Glenn Jocher. 2020. YOLOv5 by Ultralytics.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 7(3):535–547.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the nlp world. *arXiv preprint arXiv:2004.09095*.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. *arXiv preprint arXiv:2004.11362*.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. 2023. Segment anything. *arXiv preprint arXiv:2304.02643*.

Benjamin Charles Germain Lee, Jaime Mears, Eileen Jakeway, Meghan Ferriter, Chris Adams, Nathan Yarasavage, Deborah Thomas, Kate Zwaard, and Daniel S Weld. 2020. The newspaper navigator dataset: extracting headlines and visual content from 16 million historic newspaper pages in chronicling america. In *Proceedings of the 29th ACM international conference on information & knowledge management*, pages 3055–3062.

Minghao Li, Tengchao Lv, Lei Cui, Yijuan Lu, Dinei Florencio, Cha Zhang, Zhoujun Li, and Furu Wei. 2021. Trocr: Transformer-based optical character recognition with pre-trained models. *arXiv preprint arXiv:2109.10282*.

Library of Congress. 2022. Chronicling America: Historic American Newspapers.

Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. 2022. A convnet for the 2020s. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11976–11986.

Lijun Lyu, Maria Koutraki, Martin Krickl, and Besnik Fetahu. 2021. Neural ocr post-hoc correction of historical corpora. *Transactions of the Association for Computational Linguistics*, 9:479–483.

Kevin Musgrave, Serge Belongie, and Ser-Nam Lim. 2020. Pytorch metric learning.

Thi Tuyet Hai Nguyen, Adam Jatowt, Mickael Coustaty, and Antoine Doucet. 2021. Survey of post-ocr processing approaches. *ACM Comput. Surv.*, 54(6).

Konstantina Nikolaidou, Mathias Seuret, Hamam Mokayed, and Marcus Liwicki. 2022. A survey of historical document image datasets. *International Journal on Document Analysis and Recognition (IJDAR)*, 25(4):305–338.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Shruti Rijhwani, Antonios Anastasopoulos, and Graham Neubig. 2020. Ocr post correction for endangered language texts. *arXiv preprint arXiv:2011.05402*.

Zejiang Shen, Kaixuan Zhang, and Melissa Dell. 2020. A large dataset of historical japanese documents with complex layouts. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 548–549.

Zejiang Shen, Ruochen Zhang, Melissa Dell, Benjamin Charles Germain Lee, Jacob Carlson, and Weining Li. 2021. Layoutparser: A unified toolkit for deep learning based document image analysis. *International Conference on Document Analysis and Recognition*, 12821.

Baoguang Shi, Xiang Bai, and Cong Yao. 2016. 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*, 39(11):2298–2304.

David A Smith, Ryan Cordell, and Abby Mullen. 2015. Computational methods for uncovering reprinted texts in antebellum newspapers. *American Literary History*, 27(3):E1–E15.

Guanglu Song, Yu Liu, and Xiaogang Wang. 2020. Revisiting the sibling head in object detector. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11563–11572.

Teikoku Koshinjo. 1957. *Teikoku Ginko Kaisha Yoroku*. Teikoku Koshinjo.

Daniel van Strien., Kaspar Beelen., Mariona Coll Ardanuy., Kasra Hosseini., Barbara McGillivray., and Giovanni Colavizza. 2020. Assessing the impact of ocr quality on downstream nlp tasks. In *Proceedings of the 12th International Conference on Agents and Artificial Intelligence - Volume 1: ARTIDIGH,*, pages 484–496. INSTICC, SciTePress.

Linjie Xing, Zhi Tian, Weilin Huang, and Matthew R Scott. 2019. Convolutional character networks. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9126–9136.