A Picture is worth a Thousand Words:

Detecting words using OCR

Jay Patel, Danel Shifrin, and Outhai Xayavongsa

Master of Science in Applied Artificial Intelligence, University of San Diego

AAI-521: Applied Computer Vision for AI

Professor Saeed Sardari, Ph.D.

December 9, 2024

Author Note

This project, "*A Picture is Worth a Thousand Words: Detecting Words Using OCR*," was developed for AAI-521: Applied Computer Vision for AI at the University of San Diego. Team members include Outhai Xayavongsa (Team Leader), Jay Patel (Team Member), and Daniel Shifrin (Team Member). The project showcases a robust OCR pipeline using EasyOCR and TrOCR to extract and interpret text from diverse image datasets. The project code can be accessed at https://github.com/oxayavongsa/aai-521-computer-vision-final.

Abstract

This study explores the development and evaluation of Optical Character Recognition (OCR) pipelines using EasyOCR and TrOCR for text extraction from diverse and challenging image datasets. The TextOCR dataset, comprising over 900,000 word-level annotations, was used to train and evaluate these models (Singh et al., 2021). Preprocessing techniques such as grayscale conversion, noise reduction, binarization, and deskewing were applied to ensure consistent input quality. EasyOCR, a lightweight tool, excelled in handling simpler layouts with its rule-based methods, while TrOCR, leveraging a transformer-based VisionEncoderDecoder architecture, outperformed EasyOCR in recognizing distorted layouts, multilingual content, and text in noisy environments. Performance evaluation metrics, including Word Error Rate (WER) and Character Error Rate (CER), highlighted TrOCR's superior accuracy (WER: 1.00%, CER: 0.99%) compared to EasyOCR (WER: 3.10%, CER: 3.87%). The findings underscore the importance of advanced preprocessing techniques, robust model architectures, and ground truth validation in building scalable and efficient OCR systems, suitable for applications like document digitization, accessibility, and automated data processing.

Keywords: Optical Character Recognition (OCR), EasyOCR, TrOCR, TextOCR Dataset, Preprocessing Techniques, Word Error Rate (WER), Character Error Rate (CER), Transformer Models, VisionEncoderDecoder, Text Extraction, Ground Truth, Image Preprocessing.

Extracting meaningful information from images is a critical task in today's digital landscape, with applications like document digitization and automated data entry driving the demand for robust Optical Character Recognition (OCR) systems. OCR technology enables machines to interpret textual content from diverse image formats, making it a cornerstone for processing unstructured data (Chaudhuri et al., 2016). Our project, *"A Picture is Worth a Thousand Words: Detecting Words Using OCR,"* focuses on leveraging advanced OCR tools—EasyOCR and TrOCR—to create a robust text-recognition pipeline. Utilizing the TextOCR dataset, the project aims to address diverse image qualities, layouts, and complexities to achieve reliable and efficient text extraction.

Addressing the limitations of existing OCR systems is critical, as they often struggle with poor resolution, distorted or curved text, multilingual content, and environmental noise such as shadows or cluttered backgrounds. These challenges result in inconsistent performance across diverse datasets. The goal is to develop a robust and scalable OCR pipeline by integrating EasyOCR's efficiency with TrOCR's transformer-based accuracy. By leveraging advanced preprocessing techniques, iterative validation, and metrics like Word Error Rate (WER) and Character Error Rate (CER), the system is designed to extract text reliably from complex and noisy image datasets, enabling applications like large-scale digitization and automated data processing.

Data Cleaning and Preprocessing

To prepare the dataset we applied both filtering and preprocessing techniques to structure the data most optimally. Specifically, our original data set started with 25,144 images and filtered to 19,851. We applied image quality checks based on brightness, contrast, and sharpness. For brightness, we set our value to 50 using a scale of 0-255 scale to filter out images that are too

dark. Secondly, we checked the contrast threshold to measure differences between the lightest and darkest areas of the image with a threshold set to 20 indicating text and background that would be similar making the model difficult to pick up text. Last, we removed motion blur, nonfood, or low-quality images for the sharpness threshold by setting the threshold to 100.

Following this initial filtering, preprocessing techniques were applied to prepare the images for input into the OCR models. These techniques included grayscale conversion to reduce color-related noise and focus on text features, noise reduction through image smoothing, deskewing to correct any text misalignment, and binarization to enhance the distinction between text and background. These preprocessing steps were implemented to have input data that met the quality standards for the models to perform optimally for text identification and extraction. The data cleaning and preprocessing pipeline were needed to address the challenges posed by the diverse TextOCR dataset with the variability of the image quality that introduced text orientation, and image complexity. This is an important role, to make sure the input that feeds the model has a strong base image structure to enable the models to be reliable and accurate as the input directly impacts the performance of the models.

Exploratory Data Analysis (EDA) and Feature Analysis

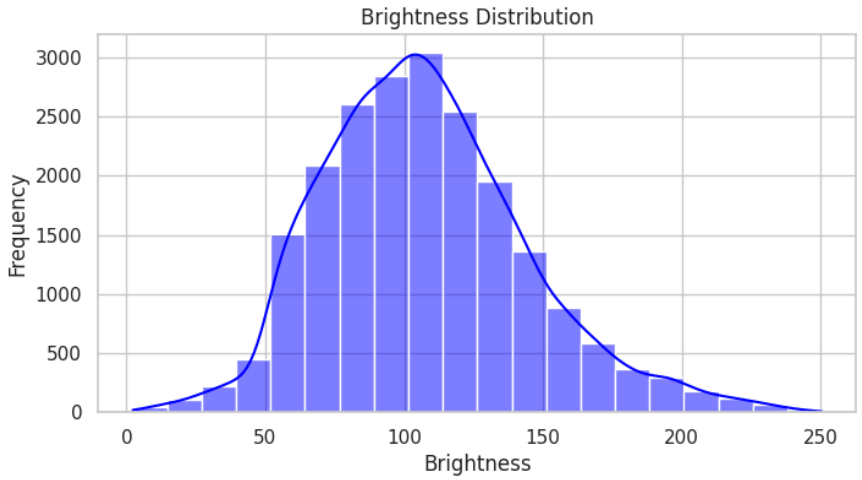
The TextOCR dataset provides a rich and diverse collection of images annotated with textual data, serving as a critical resource for training and evaluating OCR models. To optimize modeling performance, an Exploratory Data Analysis (EDA) was conducted to understand dataset characteristics and identify potential challenges. The dataset includes approximately 1 million word annotations stored in JSON format, detailing text location, shape, and content. These annotations are crucial for generating ground truth data, which is essential for model training and validation. The dataset's real-world diversity presents significant challenges, such as

varied text orientations (horizontal, vertical, and curved), multilingual content, and inconsistent image quality, all of which test OCR systems' ability to generalize effectively.

The EDA highlighted several key patterns. Most words are fewer than ten characters, reflected in a positively skewed text length distribution. Text orientation varies widely, underscoring the need for models that can handle arbitrary alignments. While the dataset predominantly contains English text, the inclusion of multilingual content offers an opportunity to assess OCR systems' cross-lingual performance. Image quality variability is evident in the brightness distribution (Figure 1). The brightness distribution reveals a concentration within a specific range, but outliers with excessively dark or overly bright areas indicate the need for normalization to enhance text clarity.

Figure 1

Brightness Distribution

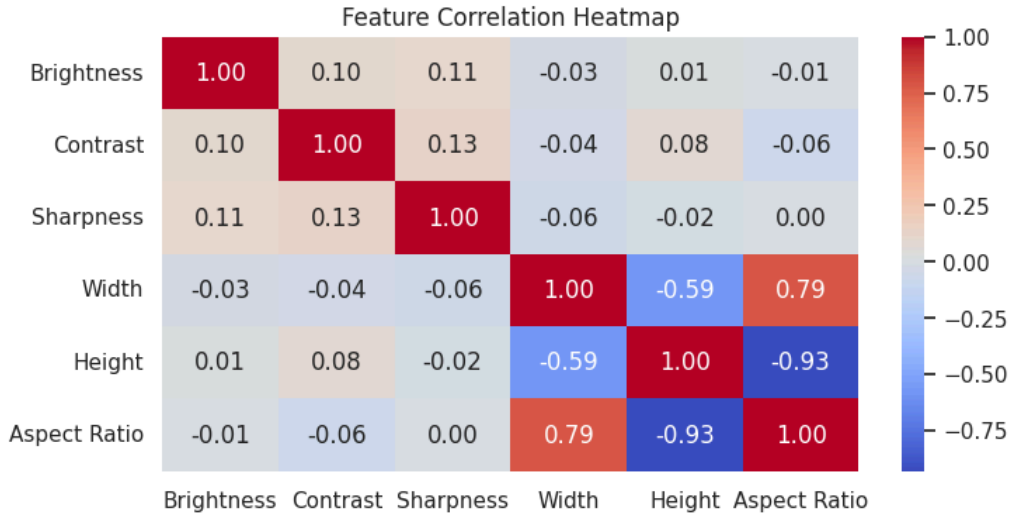


For feature analysis, TrOCR utilizes a transformer-based VisionEncoderDecoder model that leverages a self-attention mechanism for advanced feature extraction, enabling it to

effectively interpret diverse and complex text patterns. This method, illustrated by the Feature Correlation Heatmap (Figure 2), highlights the interactions between brightness, sharpness, and image dimensions, emphasizing their influence on OCR performance. In contrast, EasyOCR relies on heuristic-based pre-trained pipelines, offering simplicity and efficiency but needing to catch up in handling intricate scenarios. While EasyOCR is well-suited for straightforward text layouts, TrOCR's deep learning architecture demonstrates superior adaptability to varied and challenging datasets.

Figure 2

Feature Correlation Heatmap



Figures like the brightness and sharpness distributions and the annotated examples of bounding boxes (Figure 3) further illustrate the dataset's complexity. These insights highlight the importance of robust preprocessing and model design to address the challenges posed by this real-world dataset effectively.

Figure 3

Annotated Examples of Bounding Boxes after Ground Truth Merging

Modeling Methods and Algorithms

The modeling methods combined EasyOCR's simplicity with TrOCR's robustness to handle diverse image datasets. EasyOCR, a lightweight tool, processed text efficiently using its *readtext* function to detect regions, extract text, and compile results. Configured with a language setting, *easyocr.Reader(["en"])*, it provided quick and effective text recognition for straightforward layouts. However, its reliance on rule-based methods limited its accuracy with noisy data, distorted text, and complex layouts.

TrOCR provided an advanced solution with its VisionEncoderDecoder transformer architecture, effectively handling complex text orientations, fonts, and backgrounds.

Preprocessed images were converted to RGB and processed by the TrOCR processor to extract pixel values, which were input into the VisionEncoderDecoder model. The model generated predictions that were decoded into human-readable text using TrOCR's tokenizer, achieving significantly lower WER and CER compared to EasyOCR.

Alternative methods like CRNN, YOLO-based text detection, Keras OCR, PaddleOCR, and Tesseract were explored but excluded due to issues like task misalignment, dependency conflicts, or integration challenges. EasyOCR and TrOCR emerged as the best solutions, with EasyOCR offering simplicity and speed, while TrOCR excelled in performance and handling complexity.

Model Training, Validation, and Metrics

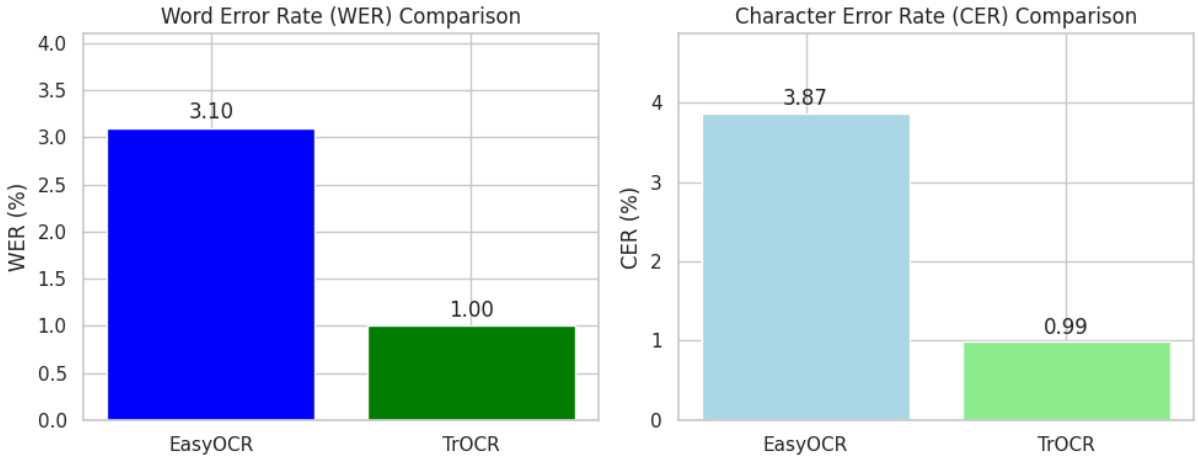
EasyOCR, pre-trained and rule-based, required no additional training and directly extracted text using its *readtext* function. In contrast, TrOCR utilized a deep-learning approach, processing RGB images through a VisionEncoderDecoder transformer where the encoder captured visual features, and the decoder predicted text sequences based on ground truth. Ground truth also played a pivotal role in validating both models, allowing reliable performance evaluation and debugging of bounding box alignment and text errors. Training with a batch size of 16 over three epochs, optimized by AdamW and a learning rate scheduler, ensured stability and efficiency using Hugging Face's *Seq2SeqTrainer*.

Validation assessed both models on separate datasets to ensure generalization and prevent overfitting. EasyOCR processed validation images with its readtext function, comparing outputs to ground truth. TrOCR generated and decoded text predictions, also compared to ground truth. Metrics like WER and CER measured accuracy after each epoch, while visual validation highlighted issues with bounding boxes, text orientation, and noise for refinement.

WER and CER, calculated using the Jiwer library, provided a quantitative assessment of the models' word and character recognition accuracy by comparing their outputs with ground truth text. TrOCR outperformed EasyOCR with a WER of 1.00% and CER of 0.99%, compared to EasyOCR's WER of 3.10% and CER of 3.87% (Figure 4). TrOCR demonstrated superior capabilities in managing complex layouts and text distortions, leveraging its transformer-based architecture for enhanced accuracy. EasyOCR, while less effective in challenging scenarios, offered faster inference, making it suitable for simpler tasks. These results highlight the complementary strengths of the two models in addressing varied OCR needs.

Figure 4

Word Error Rate (WER) versus Character Error Rate (CER) for EasyOCR and TrOCR Models.



Results and Analysis, Findings, Discussion, and Conclusion

The results demonstrated that TrOCR significantly outperformed EasyOCR in accuracy, as shown in Figure 4. TrOCR achieved a Word Error Rate (WER) of 1.00% and a Character Error Rate (CER) of 0.99%, compared to EasyOCR's WER of 3.10% and CER of 3.87%. These

metrics were computed using the Jiwer library, which compared the models' outputs with high-quality ground truth annotations. TrOCR's transformer-based VisionEncoderDecoder architecture enabled it to handle complex layouts, curved text, and multilingual content effectively. In contrast, EasyOCR, a pre-trained and rule-based model, demonstrated faster inference but was less robust in scenarios with distorted or noisy text. This distinction highlights the architectural advantages of TrOCR for datasets with intricate text arrangements. EasyOCR struggled with text containing detailed fonts and curved layouts, leading to inaccuracies such as misinterpreting "TRADITION LIBERATED" as "TTaDITIOI} LIBeraten." This visual example (Figure 5) highlights EasyOCR's limitations when faced with more challenging text scenarios.

Figure 5

EasyOCR Prediction on a Curved Text Label Demonstrating Recognition Challenges

The preprocessing pipeline also played a critical role in improving OCR accuracy. Techniques such as grayscale conversion, Gaussian blur for noise reduction, adaptive binarization, and deskewing ensured high-quality inputs for both models, reducing potential error

rates early in the process. These steps addressed text misalignment and noise issues that could otherwise hinder OCR performance. Despite these improvements, both models exhibited limitations when processing images with extremely low light or resolution, which contributed to residual error rates, even in TrOCR's outputs.

In conclusion, EasyOCR and TrOCR offer complementary strengths. EasyOCR's lightweight, resource-efficient design makes it suitable for tasks involving simple, well-structured text and scenarios requiring quick inference. TrOCR, leveraging its deep-learning framework, excels in handling complex, noisy, or multilingual datasets. TrOCR was the optimal choice, as its architecture and preprocessing integration provided superior accuracy for the dataset's challenging layouts and text conditions. Future work could focus on addressing low-light and low-resolution challenges through enhanced preprocessing or additional model fine-tuning to further improve OCR performance.

References

Chaudhuri, A., Mandaviya, K., Badelia, P., & Ghosh, S. K. (2016). Optical character recognition systems for different languages with soft computing. In *Optical Character Recognition Systems* (pp. 9–41). Springer. https://doi.org/10.1007/978-3-319-50252-6_2

Singh, A., Pang, G., Toh, M., Huang, J., Galuba, W., & Hassner, T. (2021). *TextOCR - Text Extraction from Images Dataset*. Retrieved from https://www.kaggle.com/datasets/robikscube/textocr-text-extraction-from-images-dataset

UBIAI. (2022, November 23). *Top open-source OCR programs*. Retrieved from https://ubiai.tools/top-open-source-ocr-programs/