FEDERAL UNIVERSITY OF JUIZ DE FORA
INSTITUTE OF EXACT SCIENCES
POSTGRADUATE PROGRAM IN COMPUTER SCIENCE

Ana Beatriz Kapps dos Reis

A Visual Similarity-Based Method for Children's Handwriting Recognition

Juiz de Fora
2026

Ana Beatriz Kapps dos Reis

A Visual Similarity-Based Method for Children's Handwriting Recognition

Dissertation presented to the Postgraduate Program in Computer Science at the Federal University of Juiz de Fora as a partial requirement to obtain the Master's degree in Computer Science.

Advisor: Marcelo Bernardes Vieira, Ph.D.

Co-advisors: Luiz Maurílio da Silva Maciel, Ph.D., and Saulo Moraes Villela, Ph.D.

Juiz de Fora

2026

Ficha catalográfica elaborada através do Modelo Latex do CDC da UFJF
com os dados fornecidos pelo(a) autor(a)

Kapps, Ana Beatriz.
A Visual Similarity-Based Method for Children's Handwriting Recognition / Ana Beatriz Kapps dos Reis. – 2026.
74 f. : il.

Advisor: Marcelo Bernardes Vieira, Ph.D.
Co-advisors: Luiz Maurílio da Silva Maciel, Ph.D., and Saulo Moraes Villela, Ph.D.
Dissertação (Mestrado) – Federal University of Juiz de Fora, Institute of Exact Sciences. Postgraduate Program in Computer Science, 2026.

1. Handwritten Text Recognition. 2. Children's Handwriting. 3. Character Level Recognition. 4. Visual Similarity. 5. Histogram of Oriented Gradients. I. Vieira, Marcelo Bernardes, advisor. II. Maciel, Luiz Maurílio, co-advisor. III. Villela, Saulo Moraes, co-advisor.

Ana Beatriz Kapps dos Reis

A Visual Similarity-Based Method for Children's Handwriting Recognition

Dissertação apresentada ao Programa de Pós-graduação em Ciência da Computação da Universidade Federal de Juiz de Fora como requisito parcial à obtenção do título de Mestre em Ciência da Computação. Área de concentração: Ciência da Computação.

Aprovada em 09 de setembro de 2026.

BANCA EXAMINADORA

Prof. Dr. Marcelo Bernardes Vieira - Orientador
Universidade Federal de Juiz de Fora

Prof. Dr. Luiz Maurílio da Silva Maciel - Coorientador
Universidade Federal de Juiz de Fora

Prof. Dr. Saulo Moraes Villela - Coorientador
Universidade Federal de Juiz de Fora

Prof. Dr. Jairo Francisco de Souza
Universidade Federal de Juiz de Fora

Prof. Dr. Hélio Pedrini
Universidade Estadual de Campinas

Juiz de Fora, 03/08/2026.

Documento assinado eletronicamente por **Marcelo Bernardes Vieira**, **Professor(a)**, em 09/09/2026, às 11:48, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Helio Pedrini**, **Usuário Externo**, em 10/09/2026, às 12:09, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Saulo Moraes Villela**, **Professor(a)**, em 10/09/2026, às 14:35, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Jairo Francisco de Souza**, **Professor(a)**, em 10/09/2026, às 18:07, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Luiz Maurílio da Silva Maciel**, **Professor(a)**, em 10/09/2026, às 20:43, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de

novembro de 2020.

A autenticidade deste documento pode ser conferida no Portal do SEI-Ufjf (www2.ufjf.br/SEI) através do ícone Conferência de Documentos, informando o código verificador **3118491** e o código CRC **7D0EF30F**.

ACKNOWLEDGEMENTS

I thank my mother, Elisangela Cristina Kapps, and my siblings for their encouragement, care, and every effort that allowed me to dedicate myself to my studies. I thank my boyfriend, Jorão Gomes Jr., for his companionship and emotional support throughout this journey. I am also grateful to my friends for their friendship, encouragement, and for making this journey easier.

I thank my advisors, Marcelo, Luiz, and Saulo, for their trust, patience, and dedication over the past two years. I also thank the professors of the PGCC program for sharing their knowledge and the staff of UFJF for their contributions to my education.

I thank Daniel, a fellow graduate student, for his collaboration throughout the past year, for sharing his knowledge, and for contributing to the studies and analyses conducted as part of this research.

I thank the Center for Public Policy and Education Evaluation (CAEd) for the opportunity and support, as well as for providing the data used in this research. Without this contribution, this work would not have been possible.

Finally, I thank CAPES for its financial support.

"We can only see a short distance ahead, but we can see plenty there that needs to be done." - Alan Turing, *Computing Machinery and Intelligence* (1950)

RESUMO

A transcrição automática de palavras manuscritas por crianças pode auxiliar no processamento de avaliações de alfabetização em larga escala. No entanto, a transcrição deve preservar a sequência exata de caracteres escrita pela criança, incluindo erros ortográficos que fornecem informações sobre o desenvolvimento da alfabetização. Esta dissertação investiga diferentes arquiteturas de Reconhecimento de Texto Manuscrito (*HTR*) para transcrever palavras escritas por crianças. Para realizar os experimentos, um conjunto de dados de palavras produzidas por crianças brasileiras em avaliações de alfabetização foi construído e anotado. A comparação entre as arquiteturas mostrou que algumas não são adequadas às necessidades específicas dessa tarefa. Os resultados indicaram que a arquitetura E-TrOCR, com vocabulário definido no nível do caractere é mais adequado para preservar a resposta original. Outro desafio identificado foi a ambiguidade visual entre letras na escrita infantil. Para resolver esse problema, foi proposto um método que modifica a função de perda do E-TrOCR. A nova função de perda ensina o modelo a considerar quais caracteres apresentam formas visuais semelhantes. Para representar essas relações, foi construída uma matriz de similaridade visual. Os valores dessa matriz foram calculados com descritores de Histograma de Gradientes Orientados (*HOG*) extraídos de imagens sintéticas renderizadas em fontes de estilo manuscrito. O método foi avaliado utilizando um protocolo exclusivo de palavras e aumentou a acurácia de 72,57% para 78,06%, além de reduzir a Taxa de Erro de Caracteres (*CER*) de 7,61% para 5,78%, contribuindo para transcrições mais fiéis. Esses resultados fornecem evidências de que considerar a similaridade visual entre caracteres durante o treinamento melhora a transcrição de palavras escritas por crianças sem depender de dicionários, regras linguísticas ou corpora textuais.

Palavras-chave: Reconhecimento de texto manuscrito; Caligrafia infantil; Reconhecimento em nível de caractere; Similaridade visual; Histograma de Gradientes Orientados.

ABSTRACT

Automatic transcription of words handwritten by children can support the processing of large scale literacy assessments. However, the transcription must preserve the exact character sequence written by the child, including spelling errors that provide information about literacy development. This dissertation investigates different Handwritten Text Recognition (HTR) architectures for recognizing handwriting produced by children. To conduct the experiments, a dataset of words written by Brazilian children during literacy assessments was constructed and annotated. The comparison of different architectures showed that some architectures are not well suited to the specific requirements of this task. The results indicated that the E-TrOCR architecture with a vocabulary defined at the character level is more suitable for preserving the original response. Another challenge is the visual ambiguity between letters in children's handwriting. To address this problem, a method that modifies the E-TrOCR loss function was proposed. The new loss function teaches the model to consider visual similarities between characters. To represent these relationships, a visual similarity matrix was constructed. The values in this matrix were calculated using Histogram of Oriented Gradients (HOG) descriptors extracted from synthetic images rendered in handwriting-style fonts. The method was evaluated using an exclusive words protocol and increased accuracy from 72.57% to 78.06% and reduced the Character Error Rate (CER) from 7.61% to 5.78%, contributing to more faithful transcriptions. These results provide evidence that considering visual similarity between characters during training improves the transcription of words written by children without relying on dictionaries, linguistic rules, or text corpora.

Keywords: Handwritten Text Recognition; Children's Handwriting; Character Level Recognition; Visual Similarity; Histogram of Oriented Gradients.

LIST OF FIGURES

Figure 1 – Examples of variability in children's handwriting. 14
Figure 2 – TrOCR architecture for text recognition. The input image is divided into patches and processed by a Vision Transformer encoder. The decoder uses the visual representations produced by the encoder to generate the output sequence. 19
Figure 3 – Decoder adaptation in E-TrOCR. The original TrOCR wordpiece projection is replaced by a character-level projection. 22
Figure 4 – Main steps of the HOG descriptor. The image is represented by local gradient orientations, accumulated into cell histograms, normalized in blocks, and concatenated into a feature vector. 24
Figure 5 – Examples of errors caused by character segmentation. 29
Figure 6 – Schematic representation of an assessment sheet containing two word-dictation items. 33
Figure 7 – Schematic representation of the dictation-based assessment workflow used to construct the dataset. 34
Figure 8 – Examples of handwritten responses in literacy assessments. 34
Figure 9 – Examples of handwritten sentences used for vocabulary expansion. . . 39
Figure 10 – Workflow for extracting word images from sentence-writing assessment questions. 39
Figure 11 – Number of words in each image frequency range. Notice that 163 words have only one image. Only seven words have more than 50 images. 40
Figure 12 – Examples of visual ambiguity between handwritten characters in word images from the dataset. 44
Figure 13 – Fold configuration across the three experimental runs. 45
Figure 14 – Examples of sample distribution under both evaluation protocols. . . . 46
Figure 15 – Recognition examples for unseen words. Incorrect characters are shown in red. 50
Figure 16 – Rendering of the lowercase letter "a" using the five handwritten-style fonts. 52
Figure 17 – Heatmap of pairwise cosine similarities between the HOG descriptors of characters in the vocabulary. Characters are grouped by category, with colors ranging from purple to yellow indicating increasing similarity. 54
Figure 18 – Visual similarity matrix S after thresholding ($\tau = 0.3$) and row normalization. The maximum value in the matrix is approximately 0.27. Most values are below 0.05. 55
Figure 19 – Character substitution matrix for E-TrOCR trained with cross-entropy. 63
Figure 20 – Character substitution matrix for E-TrOCR trained with the proposed soft targets. 64

Figure 21 – Selected qualitative results for the E-TrOCR baseline and the proposed method. The arrows indicate character substitutions. 67

LIST OF TABLES

Table 1 – Description of the annotation fields in the JSON file. 35
Table 2 – Writing development categories used in the annotation. 36
Table 3 – Distribution of valid responses by writing category. 36
Table 4 – Distribution of target words in the dataset. 37
Table 5 – Frequency of character occurrences in the transcriptions of the 5,052 word images selected after filtering. 38
Table 6 – Character bigram distribution in the experimental subset (5,052 samples, 22,905 occurrences, and 143 distinct bigrams). 38
Table 7 – Distribution of words in the dataset after vocabulary expansion. 41
Table 8 – Character distribution in the 6,924 word images after vocabulary expansion. 42
Table 9 – Character bigram distribution after vocabulary expansion (6,924 samples, 30,286 occurrences, and 255 distinct bigrams). 43
Table 10 – Recognition performance using three-fold cross-validation with random distribution. 48
Table 11 – Recognition performance on unseen words using three-fold cross-validation with disjoint words. 48
Table 12 – Amount of predictions of incorrect words that matched one word belonging to at least one of the training folds. 49
Table 13 – Examples of substitutions between structurally similar sequences. The numbers indicate how many time the substitution was applied for each model. . 49
Table 14 – Composition of the character vocabulary used in this work. 51
Table 15 – Validation CER (%) for different combinations of ϵ and τ. 59
Table 16 – Results for the baseline, uniform label smoothing, and the proposed method. 60
Table 17 – Results of Cochran's Q test and the pairwise McNemar tests. The McNemar tests use continuity correction, and their p-values are compared with the Bonferroni-corrected significance level of $\alpha = 0.0167$. 61
Table 18 – Per-word accuracy (%) for the baseline and the proposed method. . . . 62
Table 19 – Character-level edit operations produced by each method. 62

ACRONYMS

AdamW	Adam optimizer with decoupled weight decay
BEiT	Bidirectional Encoder Representation from Image Transformers
BERT	Bidirectional Encoder Representations from Transformers
BRESSAY	Brazilian Portuguese Dataset for Offline Handwritten Text Recognition
CAEd	Center for Public Policies and Education Assessment
CER	Character Error Rate
CLS	Classification Token
CNN	Convolutional Neural Network
CTC	Connectionist Temporal Classification
DeiT	Data-efficient Image Transformers
DTrOCR	Decoder-only Transformer for Optical Character Recognition
E-TrOCR	Extended Transformer-based Optical Character Recognition
GELU	Gaussian Error Linear Unit
HOG	Histogram of Oriented Gradients
HTR	Handwritten Text Recognition
IAM	Institute of Computer Science and Applied Mathematics Handwriting Database
ICDAR	International Conference on Document Analysis and Recognition
LSTM	Long Short-Term Memory
MiniLM	Mini Language Model
OCR	Optical Character Recognition
ResNet-50	50-layer Residual Network
RIMES	Recognition and Indexing of Handwritten Documents and Faxes
RoBERTa	Robustly Optimized BERT Pretraining Approach
TrOCR	Transformer-based Optical Character Recognition
VAN	Vertical Attention Network
VGG-16	16-layer Visual Geometry Group Network
ViT	Vision Transformer

CONTENTS

1 INTRODUCTION 13
1.1 PROBLEM DEFINITION 15
1.2 MOTIVATION 15
1.3 OBJECTIVES 15
1.4 CONTRIBUTIONS 16
1.5 OUTLINE 16
2 BACKGROUND 17
2.1 VERTICAL ATTENTION NETWORK 17
2.2 TrOCR MODEL 18
2.3 E-TrOCR MODEL 22
2.4 HISTOGRAM OF ORIENTED GRADIENTS 23
2.5 CLASS-SIMILARITY BASED LABEL SMOOTHING 26
3 RELATED WORK 28
3.1 EDUCATIONAL APPLICATIONS 28
3.2 HANDWRITING DATASETS 29
3.3 HTR METHODS AND TRAINING STRATEGIES 30
4 CHILDREN'S HANDWRITING DATASET 33
4.1 SOURCE 33
4.2 ANNOTATION PROTOCOL 35
4.3 STATISTICS 36
4.4 VOCABULARY EXPANSION 38
4.5 VISUAL AMBIGUITY IN THE DATASET 42
5 MODEL SELECTION FOR CHILDREN'S HANDWRITING 45
5.1 THREE-FOLD EVALUATION PROTOCOLS 45
5.2 EVALUATION METRICS 46
5.3 RESULTS AND DISCUSSION 47
6 PROPOSED METHOD 51
6.1 E-TrOCR ADAPTATION 51
6.2 VISUAL SIMILARITY MATRIX 52
6.3 VISUAL SIMILARITY-AWARE LOSS 55
7 EXPERIMENTAL RESULTS 58
7.1 EXPERIMENTAL PROTOCOL 58
7.2 EVALUATION METRICS 58
7.3 PARAMETER ANALYSIS 59
7.4 QUANTITATIVE RESULTS 60
7.5 DISCUSSION 65
8 CONCLUSION 68

REFERENCES **71**

1 INTRODUCTION

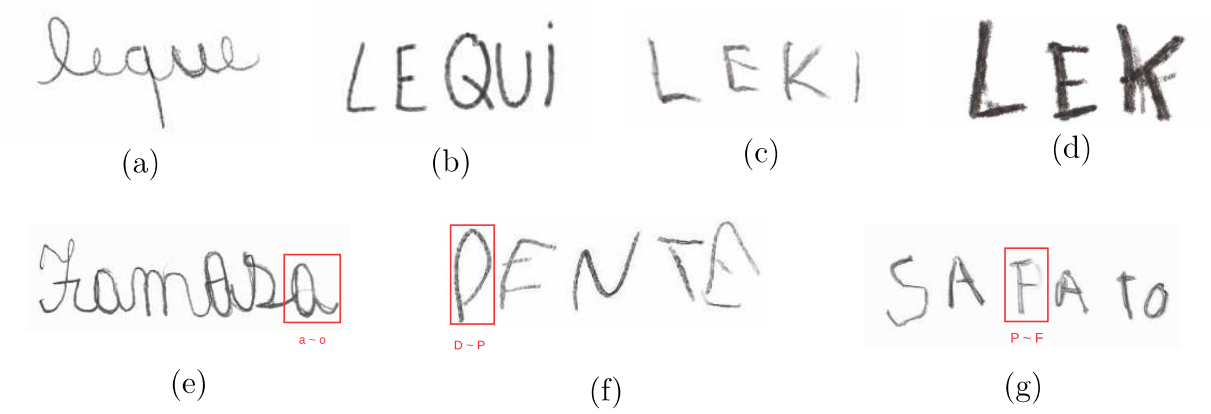
Literacy assessment plays an important role in monitoring children's learning (MAJORANO *et al.*, 2021). Word dictation tasks are used to assess writing (ENDLICH *et al.*, 2025). In these tasks, the examiner dictates a word, and the child writes what they hear. The response shows how the child represents the sounds in the dictated word using letters. Spelling errors, such as substitutions and omissions, provide information about the learning process. This information helps teachers to identify difficulties and plan pedagogical interventions (NIOLAKI *et al.*, 2023).

In large-scale educational assessments, reviewers must analyze thousands of handwritten responses. This task requires considerable time and effort. Computational methods can support this process by reducing the manual workload (GHADEKAR *et al.*, 2025). Handwritten Text Recognition (HTR) models convert handwritten text in images into character sequences (GARRIDO-MUñOZ; RíOS-VILA; CALVO-ZARAGOZA, 2025). Advances in deep learning have improved their performance on adult handwriting and historical documents (CASCIANELLI *et al.*, 2022). However, literacy assessments have a specific requirement. The transcription must preserve the exact sequence written by the child, including spelling errors that provide information about writing development. Some HTR models may instead replace the written sequence with a familiar and correctly spelled word (RANGASRINIVASAN *et al.*, 2025). When this occurs, information relevant to the assessment is lost.

Children's handwriting also shows visual variability because children are still developing control of letter formation. Strokes may be irregular, letter sizes may vary, and the same letter may take different forms (ASSELBORN *et al.*, 2021). Figure 1 illustrates both this variability and the visual similarity between different characters. Examples (a)–(d) show different responses to the dictated word *leque*: *leque*, *lequi*, *leki*, and *lek*. These responses differ in handwriting style. Examples (e)–(g) show characters with similar shapes. The letter *a* in *famosa* resembles the letter *o*. Similarly, the letter *d* in *dente* resembles *p*, and the letter *p* in *sapato* resembles *f*. These visual similarities can cause HTR models to substitute one character for another, resulting in incorrect transcriptions. Most HTR models were not specifically designed to transcribe the spelling variations found in children's writing (RANGASRINIVASAN *et al.*, 2025).

To address these limitations, this dissertation first investigates which HTR architecture is most suitable for transcribing words written by children. Three models with different architectures are compared using words seen and unseen during training. The models also differ in their output vocabularies. Some generate individual characters, whereas others can generate subwords or complete words. The results indicate that a character-level vocabulary is more appropriate for literacy assessment. Generating one

Figure 1 – Examples of variability in children's handwriting.



(a) (b) (c) (d)

(e) (f) (g)

Source: Created by the author (2026).

character at a time reduces the model's tendency to automatically correct words with spelling errors. This investigation constitutes the first contribution of this work.

The second contribution is a method designed to reduce confusion between visually similar characters during children's handwriting recognition. The method modifies the loss function by incorporating a visual similarity matrix, allowing the model to consider relationships between character shapes during training. This matrix is constructed by computing cosine similarities between Histogram of Oriented Gradients (HOG) descriptors extracted from synthetic character images rendered in handwritten-style fonts. This matrix is incorporated into the training targets by assigning a small probability to characters that resemble the reference character. The method aims to reduce substitutions between visually similar characters while preserving the sequence written by the child.

Adapting and evaluating HTR models for this context requires data that reflect the characteristics of children's handwriting. However, most available HTR datasets contain adult handwriting. Datasets containing handwriting produced by children remain limited, particularly for Brazilian Portuguese (KANG *et al.*, 2020; SILVA *et al.*, 2024). As its third contribution, this dissertation constructs and annotates a dataset of words handwritten by Brazilian children in the second year of elementary school. The responses were collected during literacy assessments administered between 2022 and 2024 across eight Brazilian states.

The remainder of this chapter is organized as follows. Section 1.1 defines the research problem and presents the research question. Section 1.2 describes the motivation for this work. Section 1.3 presents the main objective of the dissertation. Section 1.4 summarizes its main contributions. Finally, Section 1.5 describes the organization of the remaining chapters.

1.1 PROBLEM DEFINITION

Given an image of a handwritten response, the task is to generate the character sequence that reproduces exactly what the child wrote. This requirement distinguishes the task from conventional HTR settings. Recognition is further complicated by variations in letter formation, visual similarities between characters, and the limited availability of Brazilian Portuguese datasets containing children's handwriting.

Thus, this work investigates the following research question: How can words written by children be automatically transcribed while preserving the exact character sequence in each response?

1.2 MOTIVATION

The purpose of this dissertation is to investigate how HTR can produce reliable transcriptions of words written by children. These transcriptions can later support the correction and analysis of large collections of responses. This is important for the Center for Public Policies and Education Assessment at the Federal University of Juiz de Fora (CAEd/UFJF) that develops large-scale educational assessments in partnership with public education systems in Brazil. The results of these assessments provide information about student learning and support decisions made by education systems and schools (BAUER; ALAVARSE; OLIVEIRA, 2015).

In this dissertation, the data used to construct the dataset was provided by CAEd. It consists of images of handwritten responses produced by Brazilian children in literacy assessments. Automatic transcription can support the large-scale assessment task by converting each response image into a sequence of characters that can be processed computationally.

1.3 OBJECTIVES

The main objective of this dissertation is to reduce the Character Error Rate (CER) and also minimize the auto-correction of words, by adapting an HTR model to recognize words written by children in literacy assessments. The resulting transcriptions can support the automated analysis of handwritten responses and reduce evaluator workload.

Transcription represents the first stage of an automated correction process. Before classifying a response as correct or incorrect, the exact sequence of characters written by the child must be identified. The resulting transcription can then be used as input for another model that classifies the response.

1.4 CONTRIBUTIONS

The main contributions of this dissertation are:

- A construction and annotation process for a dataset of Brazilian children's handwriting collected from literacy assessments.
- A comparative evaluation of different HTR models on Brazilian children's handwritten responses.
- A visual-similarity-aware method for character-level HTR. Visual similarity is calculated from HOG features extracted from synthetic images of handwritten characters.

1.5 OUTLINE

This work is structured as follows: Chapter 2 describes the background that supports the proposed method; Chapter 3 presents relevant works in the literature to the focus of this dissertation; Chapter 4 describes the construction and annotation of the children's handwriting dataset. Chapter 5 compares different HTR models and presents the selection of the model used in this work. Chapter 6 details the proposed methods; Chapter 7 organizes the experiments and results; Chapter 8 discusses the conclusions obtained and possible future work.

2 BACKGROUND

This chapter presents the background that underpins the proposal of this work. Section 2.1 introduces the Vertical Attention Network (VAN), an end-to-end encoder-decoder architecture for handwritten text recognition based on a fully convolutional encoder and a hybrid vertical attention mechanism. Section 2.2 presents the TrOCR architecture, which combines a vision Transformer encoder with a text Transformer decoder. Section 2.3 describes E-TrOCR, an adaptation of TrOCR that uses a character level vocabulary to recognize handwriting produced by children. Section 2.4 explains the Histogram of Oriented Gradients descriptor, which is used to represent the visual structure of handwritten characters. Lastly, Section 2.5 presents label smoothing based on class similarity.

2.1 VERTICAL ATTENTION NETWORK

The Vertical Attention Network (VAN) is an end-to-end architecture for handwritten text recognition (COQUENET; CHATELAIN; PAQUET, 2023). It was originally developed to recognize complete paragraphs without requiring each line of text to be segmented before recognition. The complete architecture contains a fully convolutional encoder, a vertical attention module, and a character decoder. The authors also present a line-level configuration based on the same encoder. Since the input already contains a single line, this configuration does not require the vertical attention module.

The encoder receives an image $X \in \mathbb{R}^{H \times W \times C}$. It is composed of convolutional blocks and depthwise separable convolutional blocks. These layers extract visual information and produce a feature map $f \in \mathbb{R}^{H_f \times W_f \times 256}$, where $H_f = H/32$ and $W_f = W/8$. The spatial dimensions are reduced while the horizontal sequence of the text is retained.

In the paragraph-level architecture, the vertical attention module selects one text line at a time from the feature map. The selected features are processed by a LSTM layer with 256 hidden units. A one-dimensional convolution then produces the character probabilities. This process is repeated until the model detects the end of the paragraph.

For line-level recognition, the attention module and the LSTM decoder are removed. The feature map produced by the encoder is passed to an AdaptiveMaxPooling layer. This layer collapses the vertical dimension and produces a sequence of size $1 \times W_f \times 256$. The horizontal dimension is preserved because it represents the order of the characters in the line. A convolutional layer then estimates the probability of each character at every horizontal position.

The line-level model is trained with the Connectionist Temporal Classification (CTC) loss. In addition to the N characters in the vocabulary, CTC uses a blank symbol to represent positions that are not associated with a character. This allows the model

to align its predictions with the target transcription without requiring the position of each character in the image. During decoding, blank symbols and consecutive repeated predictions are removed to obtain the final character sequence.

In this dissertation, this line-level configuration is applied to isolated word images. Each word is treated as a single text line. Therefore, the model recognizes the character sequence without using the vertical attention module required for paragraph recognition.

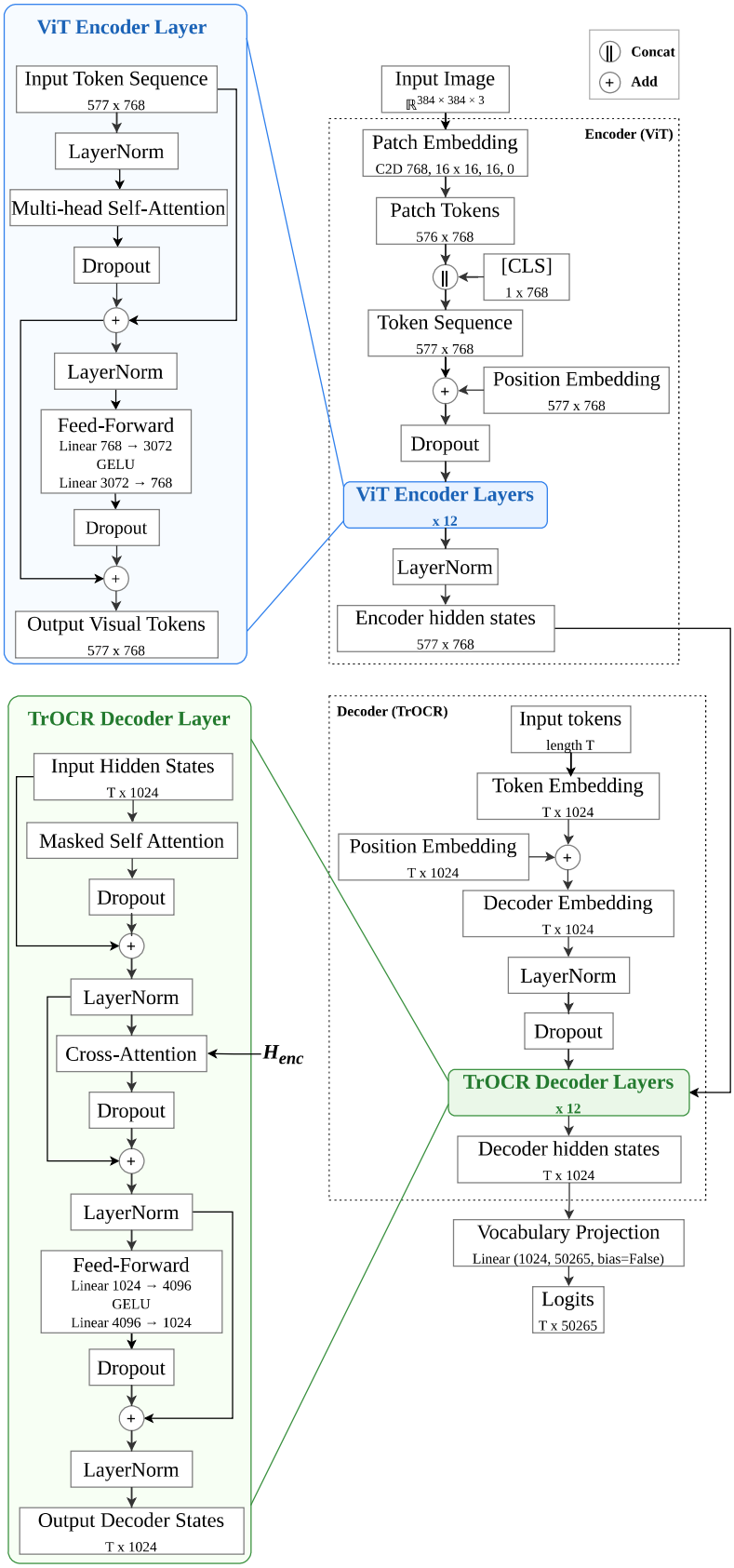
2.2 TrOCR MODEL

TrOCR is a Transformer-based model for text recognition (LI *et al.*, 2023). It receives an image containing text and generates a sequence of textual tokens. The model uses an encoder-decoder architecture composed of a pre-trained image Transformer and a pre-trained text Transformer. In the original TrOCR models, the encoder is initialized from pre-trained image Transformer models, such as DeiT (TOUVRON *et al.*, 2021) or BEiT (BAO *et al.*, 2022), and the decoder is initialized from pre-trained text Transformer models, such as RoBERTa (LIU *et al.*, 2019) or MiniLM (WANG *et al.*, 2020). The encoder processes the input image and produces visual representations. The decoder uses these representations to generate the output sequence. Figure 2 organizes this architecture into two main components: the Vision Transformer encoder and the TrOCR decoder. The figure also details one encoder layer and one decoder layer.

The encoder starts from an RGB image represented with size $384 \times 384 \times 3$. In the configuration shown in Figure 2, the patch embedding module is implemented as a convolutional projection with 768 output channels, a kernel size of 16×16, stride 16, and no padding. Therefore, the image is divided into non-overlapping patches of 16×16 pixels. Since $384/16 = 24$, this produces a grid of $24 \times 24 = 576$ patches. Each patch is projected into a vector of dimension 768, resulting in a sequence of patch tokens with size 576×768. A learnable classification token ([CLS]) with size 1×768 is then concatenated to this sequence, increasing it to 577×768. This element is used by the ViT encoder to represent global information about the image. Positional embeddings with the same size are added to the token sequence because the Transformer does not encode the spatial position of each patch. After this addition, dropout is applied, and the resulting sequence is passed to the Vision Transformer encoder layers.

The sequence is then processed by the Vision Transformer (ViT) encoder. The encoder is composed of 12 stacked Transformer layers. Each layer receives a token sequence with size 577×768 and preserves this size at its output. The layer first applies layer normalization and then multi-head self-attention. In the Transformer, multi-head self-attention is computed using queries, keys, and values (VASWANI *et al.*, 2017). In the encoder, learned linear projections are applied to the patch representations of the previous layer. From these projections, each patch produces a query, a key, and a value. The

Figure 2 – TrOCR architecture for text recognition. The input image is divided into patches and processed by a Vision Transformer encoder. The decoder uses the visual representations produced by the encoder to generate the output sequence.



Source: Created by the author (2026).

query represents what the patch is looking for, the key is used to compare this patch with the other patches, and the value contains the information that can be passed to the output. The model compares queries with keys to compute attention weights. These weights indicate how much each patch should attend to the others. The output is then obtained by combining the values according to these weights. The scaled dot-product attention used by the encoder is defined as:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \tag{2.1}$$

where Q represents queries, K represents keys, V represents values, and d_k is the dimension of the key vectors. The product QK^T measures the similarity between queries and keys. The division by $\sqrt{d_k}$ scales these values before the softmax operation. The softmax produces attention weights that are used to combine the values.

Multi-head self-attention applies this operation several times in parallel. Each head can focus on different connections between patches. In a handwritten word image, one head may capture local stroke characteristics, while another may capture broader visual structure. In this arrangement, multi-head self-attention allows each patch representation to interact with the other patch representations. This is important in handwritten text recognition because strokes and letters are not always isolated. A local region of a character may depend on nearby strokes or surrounding visual information.

After multi-head self-attention, dropout is applied and the result is added back to the input through a residual connection. A second layer normalization is then applied before the feed-forward network. The feed-forward network processes each representation independently. It applies learned transformations to the features produced by attention. This network expands each token representation from 768 to 3072 dimensions, applies the GELU activation, and projects it back to 768 dimensions. Another dropout and residual addition complete the layer.

This process is repeated across the 12 ViT encoder layers. After these layers, a final layer normalization is applied. The output is a sequence of encoder hidden states with size 577×768. These hidden states summarize the visual information extracted from the image. At this point, the model has not processed text yet. The encoder hidden states, denoted as H_{enc}, are passed to the TrOCR decoder and used during cross-attention.

The decoder receives two sources of information. The first is the input token sequence, which provides textual context. The second is the sequence of encoder hidden states, denoted as H_{enc}. These hidden states provide the visual information extracted from the image. During training, the input token sequence corresponds to the target transcription shifted to the right, starting with a beginning-of-sequence token. In this way, the decoder learns to predict each next token from the previous tokens while using H_{enc} as visual information from the word image.

The input token sequence has length T, which varies according to the number of tokens in the transcription. Each token is converted into an embedding with dimension 1024, producing a representation with size $T \times 1024$. Positional embeddings with the same size are added to represent the order of the sequence. After this addition, layer normalization and dropout are applied. The resulting decoder embeddings are passed to the TrOCR decoder layers.

The TrOCR decoder is composed of 12 stacked decoder layers. Each layer receives hidden states with size $T \times 1024$. The first operation inside the layer is masked self-attention. This module allows each token position to attend only to previous positions in the output sequence. Future positions are masked because the decoder generates the transcription from left to right. After masked self-attention, dropout is applied, and the result is added back through a residual connection. Layer normalization is then applied.

The next operation is cross-attention. This is the module that connects the textual decoder to the visual encoder. In cross-attention, the queries come from the decoder hidden states, while the keys and values come from the encoder hidden states H_{enc}. This allows each output token representation to attend to the visual information extracted from the image. After cross-attention, the layer applies dropout, residual addition, and layer normalization.

The decoder layer then applies a feed-forward network. This network expands each decoder representation from 1024 to 4096 dimensions, applies the GELU activation, and projects it back to 1024 dimensions. After another dropout, residual addition, and layer normalization, the layer outputs decoder states with size $T \times 1024$. This process is repeated across the 12 decoder layers. The final output is a sequence of decoder hidden states.

After the decoder layers, the vocabulary projection maps the decoder hidden states to the vocabulary space. This projection is a linear layer from 1024 dimensions to 50265 vocabulary units. The result is a sequence of logits with size $T \times 50265$. These logits are converted into probabilities over the vocabulary, and the model selects the next token according to these probabilities. This process continues until the end of the sequence is generated.

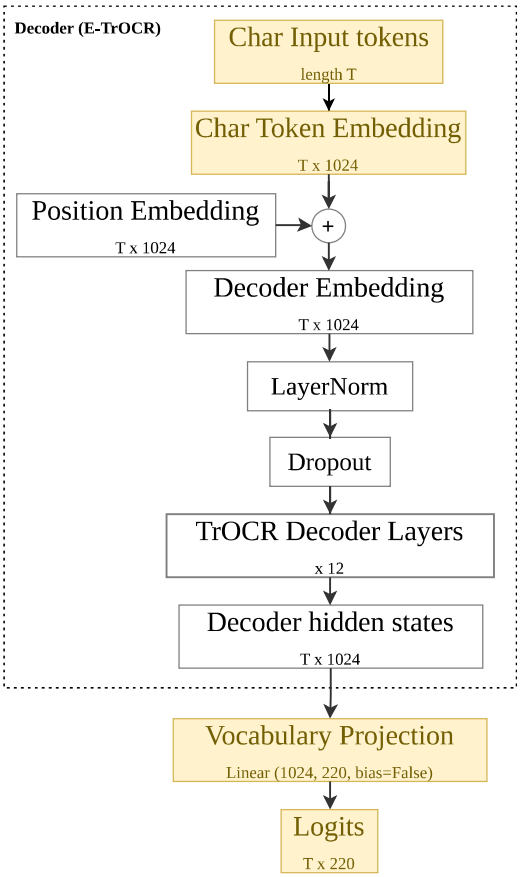
The output vocabulary is an important part of TrOCR. In the original model, the decoder performs wordpiece-level text generation (LI *et al.*, 2023). Wordpiece tokens are useful for general text recognition because they represent frequent textual units. However, this can be less suitable when the goal is to preserve spelling errors. This point is important for this dissertation. TrOCR provides a strong image-to-text recognition architecture. It connects visual understanding and text generation in a single model. However, its original formulation was not designed for children's spelling productions in literacy assessment. This motivates the use of an adaptation focused on child handwriting and character-level

output.

2.3 E-TrOCR MODEL

E-TrOCR adapts TrOCR for child handwriting recognition (RANGASRINIVASAN *et al.*, 2025). Figure 3 illustrates this adaptation. The model keeps the encoder-decoder structure of TrOCR, but modifies the decoder output representation. In this setting, recognition is performed at the character level. The encoder still extracts visual representations from the input image. The decoder predicts a sequence of individual characters instead of wordpiece tokens.

Figure 3 – Decoder adaptation in E-TrOCR. The original TrOCR wordpiece projection is replaced by a character-level projection.



Source: Created by the author (2026).

The decoder receives character input tokens with sequence length T. The value of T is not fixed, because it depends on the number of tokens in the transcription. Each character token is converted into an embedding vector with dimension 1024. This produces

a sequence of character token embeddings with size $T \times 1024$. Positional embeddings with the same size are added to these embeddings to represent the order of the character sequence. After this addition, the decoder embedding still has size $T \times 1024$.

The decoder embedding is then passed through layer normalization and dropout. The resulting sequence is processed by the 12 TrOCR decoder layers. These layers keep the same Transformer structure used in TrOCR, including masked self-attention, cross-attention with the encoder hidden states, and feed-forward networks. Therefore, the main change in E-TrOCR is not the internal structure of the decoder layers. The change is in the units that the decoder receives and predicts.

After the TrOCR decoder layers, the model produces decoder hidden states with size $T \times 1024$. These hidden states are passed to the vocabulary projection layer. In the E-TrOCR architecture this projection is a linear layer of 220 vocabulary units instead of the original size of 1024. These units include alphabet characters and additional symbols for reversed letters, inverted letters, and letters that are both reversed and inverted. As a result, the output logits have size $T \times 220$. Each position in the output sequence contains scores over the character vocabulary.

This change is important because spelling errors in children's writing often occur at the character level. A substitution, deletion, or insertion of one character can change the written response, and the model must be able to represent this sequence. A wordpiece vocabulary may favor frequent textual units learned during training. A character-level vocabulary gives the decoder more flexibility to generate sequences that do not necessarily correspond to common words.

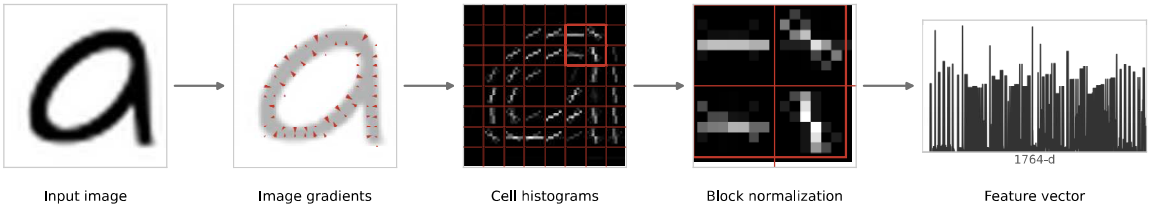
For this reason, E-TrOCR is more suitable for the task addressed in this work than the original TrOCR model. It keeps the Transformer-based recognition structure, but adapts the output to child handwriting. In the original E-TrOCR work, the decoder uses 220 output units. In the implementation used in this work, this idea is adapted to a Portuguese character vocabulary. Also, based on this character-level perspective, we extend the training process with visual similarity information to help the model deal with ambiguous handwritten forms.

2.4 HISTOGRAM OF ORIENTED GRADIENTS

The Histogram of Oriented Gradients (HOG) is a visual descriptor introduced by Dalal and Triggs (2005) for object detection. The descriptor represents local shape by using the distribution of gradient orientations in small regions of an image. The image is divided into cells, and each cell stores a histogram of edge or gradient directions. These histograms describe the local structure of the image, such as contours and strokes. Figure 4 illustrates the main steps of the HOG descriptor. First, the image gradients are computed. Then, the image is divided into small regions called cells. In each cell, a histogram of

gradient orientations is accumulated. After that, neighboring cells are grouped into blocks and normalized. The final HOG descriptor is obtained by concatenating the normalized histograms from all blocks.

Figure 4 – Main steps of the HOG descriptor. The image is represented by local gradient orientations, accumulated into cell histograms, normalized in blocks, and concatenated into a feature vector.



Source: Created by the author (2026).

Let $I(x, y)$ be a grayscale image, where x and y indicate the pixel position. The first step is to compute the horizontal and vertical gradients of the image. This can be done with a simple derivative filter, such as $[-1, 0, 1]$, which was shown to be effective without smoothing (DALAL; TRIGGS, 2005). The gradients can be written as:

$$g_x(x, y) = I(x + 1, y) - I(x - 1, y), \tag{2.2}$$

$$g_y(x, y) = I(x, y + 1) - I(x, y - 1), \tag{2.3}$$

where $g_x(x, y)$ represents the horizontal intensity variation and $g_y(x, y)$ represents the vertical intensity variation. From these gradients, the magnitude and orientation are computed. The gradient magnitude measures the strength of the local variation, while the orientation indicates the direction of this variation. They are defined as:

$$m(x, y) = \sqrt{g_x(x, y)^2 + g_y(x, y)^2}, \tag{2.4}$$

$$\theta(x, y) = \operatorname{atan2}\left(g_y(x, y), g_x(x, y)\right). \tag{2.5}$$

The magnitude $m(x, y)$ is high in regions where there is a strong change in intensity, such as edges and strokes. The orientation $\theta(x, y)$ indicates the direction associated with this change. In the case of handwritten characters, these directions are related to the local structure of the strokes. After computing the gradients, the image is divided into cells. Each cell accumulates a histogram of gradient orientations. The orientation range is divided into a fixed number of bins, and each pixel contributes to the histogram according to its gradient orientation. This contribution is weighted by the gradient magnitude. Thus,

the histogram of a cell summarizes how the gradient directions are distributed in that local region of the image. Pixels with stronger gradients contribute more to the histogram because they are more likely to belong to relevant strokes or edges.

Gradient magnitudes may change because of contrast, illumination, or stroke intensity. To reduce this effect, HOG groups neighboring cells into larger regions called blocks. Each block contains the histograms of its cells. Let v be the vector formed by concatenating the histograms inside a block. A common normalization is based on the L^2 norm:

$$\hat{v} = \frac{v}{\sqrt{\|v\|_2^2 + \epsilon^2}}, \tag{2.6}$$

where $\|v\|_2$ is the Euclidean norm of the block vector and ϵ is a small constant used to avoid division by zero. This normalization reduces the influence of global intensity differences and makes the descriptor more dependent on the relative distribution of orientations.

Another normalization used in HOG is L^2-Hys. In this scheme, the vector is first normalized using L^2 normalization. Then, large values are clipped to a maximum threshold and the vector is normalized again. In the HOG implementation used in `scikit-image` library[1], this threshold is 0.2. The clipping operation can be written component-wise as:

$$\hat{v}_{\text{clip},i} = \min(\hat{v}_i, 0.2), \tag{2.7}$$

where $\hat{v}_i$ is the i-th component of the normalized block vector. Thus, each component of $\hat{v}$ is limited to a maximum value of 0.2. After clipping, the vector is normalized again:

$$\tilde{v} = \frac{\hat{v}_{\text{clip}}}{\sqrt{\|\hat{v}_{\text{clip}}\|_2^2 + \epsilon^2}}. \tag{2.8}$$

The final HOG descriptor is obtained by concatenating the normalized vectors from all blocks in the image. In the `scikit-image` implementation, when `feature_vector=True`, this descriptor is returned as a flattened one-dimensional feature vector. This vector summarizes the local orientation structure of the image.

In this work, the HOG descriptor is used to represent individual characters. Each character image is converted into a feature vector that describes the local edge and contour structure of the character. Characters with similar shapes tend to produce similar HOG vectors. For example, handwritten forms of “a” and “o” may share curved stroke patterns, while “d” and “p” may share vertical strokes and rounded parts. This makes HOG suitable for estimating visual similarity between characters.

[1] SCIKIT-IMAGE. **Histogram of Oriented Gradients**. 2025. Available at: <https://scikit-image.org/docs/0.25.x/api/skimage.feature.html#skimage.feature.hog>.

2.5 CLASS-SIMILARITY BASED LABEL SMOOTHING

In character-level recognition, the decoder predicts a probability distribution over the character vocabulary at each position of the output sequence. Although the model generates a sequence, the prediction at each position can be described as a classification over K possible characters. The output probability for a class k is obtained with the softmax function:

$$h(k|x) = \frac{\exp(z_k)}{\sum_{i=1}^{K} \exp(z_i)}, \tag{2.9}$$

where x is the input sample, z_k is the logit associated with class k, and $h(k|x)$ is the predicted probability for class k.

In the standard formulation, the target label is represented as a one-hot vector. This means that the correct class receives probability 1 and all other classes receive probability 0. For a sample with true class y, the loss is defined as:

$$\ell(x) = -\log h(y|x). \tag{2.10}$$

This formulation trains the model to assign high probability to the correct class. However, it treats all incorrect classes in the same way. In character recognition, this can be limiting because some characters are visually closer than others. For example, confusing two characters with similar handwritten shapes is different from confusing two characters with very distinct shapes.

Label smoothing changes the target representation. Instead of assigning all probability to the correct class, it reserves a small part of the probability mass for other classes (SZEGEDY *et al.*, 2016). In uniform label smoothing, this extra probability is distributed equally among the classes. The smoothed target can be written as:

$$\pi_y(k) = (1 - \alpha)e_y(k) + \alpha u(k), \tag{2.11}$$

where α is the smoothing factor, $e_y(k)$ is the one-hot target for class y, and $u(k)$ is a uniform distribution over the classes.

The limitation of uniform label smoothing is that it assumes that all non-target classes are equally related to the correct class. Class-similarity based label smoothing changes this assumption by assigning more probability to classes that are more similar to the correct one (LIU; JAJA, 2021). Let $s_y(k)$ be a similarity-based distribution for the reference class y. The class-similarity based smooth label is defined as:

$$\pi_y(k) = (1 - \alpha)e_y(k) + \alpha s_y(k), \tag{2.12}$$

where α controls how much probability is assigned to the similarity-based distribution. The correct class remains the main target, but the remaining probability is assigned according to the similarity between the reference class and the other classes.

Using this target, the cross-entropy loss becomes:

$$\ell(x) = -\sum_{k=1}^{K} \pi_y(k) \log h(k|x). \tag{2.13}$$

This loss still trains the model to predict the correct class. The difference is that the target distribution also represents the affinities between classes. Thus, the model is not trained only with a rigid one-hot target. It receives a soft target that indicates which classes are closer to the correct one.

This idea is relevant to this work because children's handwriting contains visual ambiguity between characters. Some recognition errors may occur because two characters have similar handwritten forms. Using class similarity during training allows the target distribution to reflect visual ambiguity between characters. For this reason, class-similarity based label smoothing provides the theoretical basis for the loss adaptation investigated in this work.

3 RELATED WORK

This chapter reviews studies related to the recognition of children's handwriting in educational contexts. The chapter is organized into three sections. Section 3.1 discusses educational applications that process handwritten responses. Section 3.2 presents handwriting datasets. Section 3.3 reviews HTR methods, discusses evaluation protocols and label-smoothing approaches used in recognition tasks.

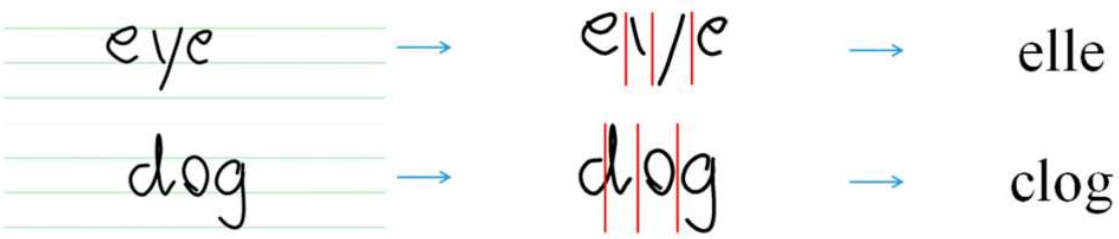
3.1 EDUCATIONAL APPLICATIONS

Handwriting recognition can support the review of handwritten answers in educational activities. This is useful in schools, where many assessments still require manual review. In this context, HTR can help convert handwritten answers into digital text. Automatic transcription can also reduce the time needed to process large sets of responses. An example of a large-scale educational application involving handwritten responses is the study by Portela *et al.* (2024). Teachers used a mobile application to photograph handwritten essays. The system first located the handwritten text in each image. The handwritten content was then transcribed into digital text. Finally, Natural Language Processing (NLP) was used to assess the essays automatically. The NLP component evaluated four aspects of each essay: formal writing, thematic coherence, text type, and cohesion. The system then generated a feedback report for students and teachers. The paper does not describe the HTR architecture used to transcribe the texts. The paper also does not report the metrics needed to compare recognition models. The solution was used in 8,238 schools and reached 164,545 students. This study shows that handwriting recognition can support educational assessment on a large scale in Brazil.

Research has also examined the recognition of words written by children in educational applications. Zin *et al.* (2021) developed an Android application for recognizing English words written by primary school students. The system first divides each word into individual characters. A CNN then recognizes each character separately. On 1,000 words, the system achieved 98.50% segmentation accuracy, 98.76% character accuracy, and 95.60% word accuracy. These results depend on the segmentation stage because the CNN analyzes only the separated characters. The authors report errors when a letter has disconnected strokes. Some examples are shown in Figure 5. In the word *eye*, the system interprets *y* as *l* followed by *l*, producing *elle*. In another example, the system interprets *d* as *c* and *l*, producing *clog*. The authors state that the method may not be suitable for complex handwriting styles. This limitation is particularly relevant to literacy assessments, where incomplete strokes and irregular spacing are common. This study is related to the present dissertation because it addresses the recognition of words written by children. However, the method depends on separating the characters before recognition.

This is difficult when letters are incomplete, irregular, or connected. In contrast, the models evaluated in this dissertation receive the complete word image and generate its transcription. They do not segment the word into individual characters.

Figure 5 – Examples of errors caused by character segmentation.



Source: Adapted from Zin *et al.* (2021).

For Brazilian Portuguese, Aires *et al.* (2026) followed a different approach and evaluated VGG-16 and ResNet-50 as word classifiers. The dataset used in the study contained 3,476 images from approximately 50 children aged 6 to 10. These images represent 84 predefined word classes. With transfer learning and data augmentation, VGG-16 achieved 96.55% accuracy, while ResNet-50 achieved 95.39%. However, the task was formulated as closed-vocabulary classification. The models can only select one of the words defined during training. The authors identify this setting as a limitation and suggest character-level models for open-vocabulary recognition. This study is related to the present dissertation because both focus on words handwritten by Brazilian children. Unlike their approach, this dissertation focuses on transcribing character sequences without restricting the output to predefined word classes.

These applications show different uses of handwriting recognition in education. They include large-scale essay processing, character recognition after segmentation, and classification of predefined words. However, these formulations do not fully represent the task investigated in this dissertation. This dissertation focuses on transcribing children's handwritten words, including spelling deviations, without relying on a predefined vocabulary.

3.2 HANDWRITING DATASETS

The IAM Handwriting Database (MARTI; BUNKE, 2002) is one of the most widely used datasets for offline HTR (HUSSAIN *et al.*, 2015). It contains English handwriting annotated at page, line, and word levels. The dataset includes 1,539 pages, 13,353 text lines, and 115,320 word images produced by 657 writers. Its size and different annotation levels have supported the development of word and line-recognition methods. RIMES is another HTR dataset (GROSICKI; EL-ABED, 2011) that contains French handwritten

correspondence and provides data for word and line-level recognition. IAM and RIMES are useful for comparing general HTR models. IAM and RIMES can also be used to pretrain HTR models before adaptation to smaller datasets.

For Brazilian Portuguese, Neto *et al.* (2024a) introduced BRESSAY. The dataset contains 1,000 handwritten essay pages and includes annotations for different text levels. It represents a broad range of writing styles and document conditions. BRESSAY was later used in the ICDAR 2024 competition (NETO *et al.*, 2024b). The competition evaluated line, paragraph, and page-level recognition. BRESSAY is an important resource for Brazilian Portuguese HTR. The public release includes images of individual words. However, these images have small dimensions. This limits the visual detail of the characters.

The dataset used by Aires *et al.* (2026) is closer to the context of this dissertation because it contains words written by Brazilian children. It includes 3,476 images produced by approximately 50 children aged 6 to 10. The images are organized into 84 predefined word classes and were used for closed vocabulary classification. However, access to the dataset is restricted, and the image collection is not included in the public code repository. For this reason, it could not be incorporated into the experiments conducted in this dissertation.

In general, public HTR datasets primarily contain handwriting produced by adults and represent languages other than Brazilian Portuguese. Publicly available datasets focused on handwriting produced by children remain limited. Their release may be restricted by privacy, data protection, and ethical requirements. This limited availability motivates the construction of the dataset presented in Chapter 4.

3.3 HTR METHODS AND TRAINING STRATEGIES

HTR research has explored different approaches for converting images of handwritten text into character sequences. Some architectures combine convolutional layers for visual feature extraction with recurrent layers for sequence modeling. These architectures can be trained using Connectionist Temporal Classification (CTC), which enables the recognition of handwritten words or lines without explicit character segmentation (GRAVES *et al.*, 2009). The methods proposed by Puigcerver (2017) and Bluche (2016) are examples of these approaches. Other methods address larger text regions. OrigamiNet, for example, performs complete page recognition without requiring explicit line segmentation (YOUSEF; BISHOP, 2020). These studies illustrate the use of deep learning methods at different text levels. However, they were mainly evaluated on datasets containing adult handwriting rather than handwriting produced by children in literacy assessments.

More recent works have explored architectures that process larger regions with handwritten text. The Vertical Attention Network (VAN), proposed by Coquenet, Chatelain and Paquet (2023), is an end-to-end model for handwritten line and paragraph recognition.

It combines convolutional feature extraction with attention mechanisms and can process handwritten text without requiring explicit line segmentation. VAN achieved competitive results on datasets such as IAM (MARTI; BUNKE, 2002) and RIMES (GROSICKI; EL-ABED, 2011).

Transformer-based models have also become important in Optical Character Recognition (OCR) and HTR. TrOCR, proposed by Li *et al.* (2023), combines a Vision Transformer encoder with a Transformer decoder for text generation. The model achieved promising results in printed, handwritten, and scene text recognition tasks. DTrOCR, proposed by Fujitake (2024), follows another direction and uses a decoder-only Transformer for OCR. These works show the potential of Transformer architectures for text recognition. However, their goal is general text recognition.

More recently, E-TrOCR was proposed as a Transformer model adapted to recognize handwriting produced by children (RANGASRINIVASAN *et al.*, 2025). E-TrOCR modifies TrOCR by using tokenization at the character level and adapting the model to children's handwriting. Generating one character at a time can reduce the tendency to correct the sequence written by the child. The model was also developed to preserve spelling errors, reversed letters, and irregular character shapes. Among the studies reviewed, E-TrOCR is the most closely related to the present work. However, it does not address Brazilian Portuguese, accented characters, or literacy assessments based on dictation. It also does not consider visual similarity between characters.

Evaluation protocols also affect the interpretation of HTR results. In many works, training and test sets are created with random splits. This can be useful, but it may overestimate performance when the same words appear in both sets. Recent studies on HTR generalization show that textual differences between training and test data can strongly affect performance (GARRIDO-MUÑOZ; CALVO-ZARAGOZA, 2025). This discussion is important for this dissertation. The recognition model should not be evaluated only on familiar samples. The evaluation should also test the model's ability to deal with new words and visual variation.

In children's handwriting, recognition errors can occur when different letters have similar shapes. Label smoothing has been used to make training targets less rigid by distributing a small part of the target probability to other classes (SZEGEDY *et al.*, 2016). In its standard form, this distribution is usually uniform. Class-similarity based label smoothing changes this idea by assigning more weight to classes that are more similar to the correct class (LIU; JAJA, 2021). In the method investigated in this dissertation, this probability is distributed according to the visual similarity between handwritten characters.

Similarity-aware strategies have also been explored in sequence recognition. Huang *et al.* (2021) proposed Context-Aware Selective Label Smoothing to calibrate sequence

recognition models. The method uses contextual prediction statistics and confusion information to adjust the smoothing process. This work is relevant because it applies smoothing ideas to sequence recognition.

The reviewed studies address complementary aspects of the problem investigated in this dissertation. Educational applications show that handwriting recognition can support the large-scale processing of student responses. Existing HTR methods provide different approaches to recognize handwritten text. Some studies remark the importance of evaluating models with words that do not appear in the training. Research on similarity-based label smoothing provides strategies for representing relationships between classes. In this context, we propose the use of visual similarity between characters in the recognition of isolated words written by Brazilian children in literacy assessments.

4 CHILDREN'S HANDWRITING DATASET

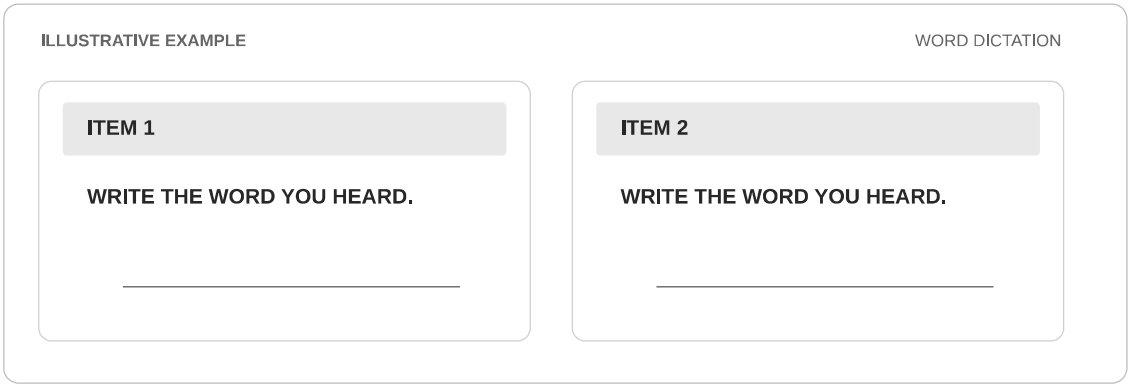
This chapter presents the dataset constructed in this dissertation. Section 4.1 describes the source of the data. Section 4.2 presents the annotation protocol. Section 4.3 presents the main statistics of the dataset. Section 4.4 describes the vocabulary expansion process. Finally, Section 4.5 discusses examples of visual ambiguity observed in the dataset.

4.1 SOURCE

The dataset was constructed from handwritten responses collected during literacy assessments administered between 2022 and 2024 in eight Brazilian states: Bahia, Espírito Santo, Pernambuco, Mato Grosso do Sul, Paraná, Alagoas, Maranhão, and Ceará. These states represent the Northeast, Southeast, Central-West, and South regions of Brazil. All responses were collected from word dictation tasks administered to children in the second year of elementary school. The assessment sheets were scanned, and the resulting digital images were provided by CAEd/UFJF.

Each assessment sheet contained two word dictation items. For each item, the teacher dictated a word, and the student wrote a response on the sheet. Thus, each sheet contained two handwritten responses from the same student. Figure 6 presents a schematic representation of this format. The assessment sheets were randomly selected from the material available for each year and state. Different dictated words were included to increase the diversity of the samples. However, the data did not contain student identifiers. Therefore, it was not possible to determine whether the same student contributed responses on more than one sheet.

Figure 6 – Schematic representation of an assessment sheet containing two word-dictation items.

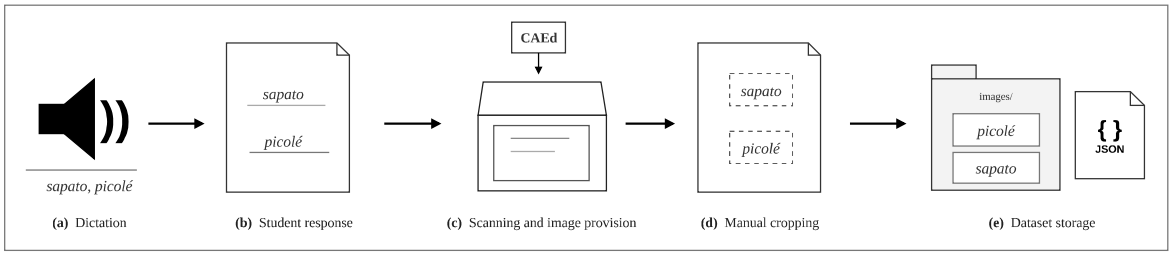


Source: Created by the author (2026).

Figure 7 illustrates the dictation task and the main stages of dataset construction.

After the assessments were administered, the sheets containing the students' responses were scanned, and the resulting digital images were provided by CAEd. During annotation, the region containing each response was manually identified and cropped. Each resulting word image was then stored with its corresponding annotation file.

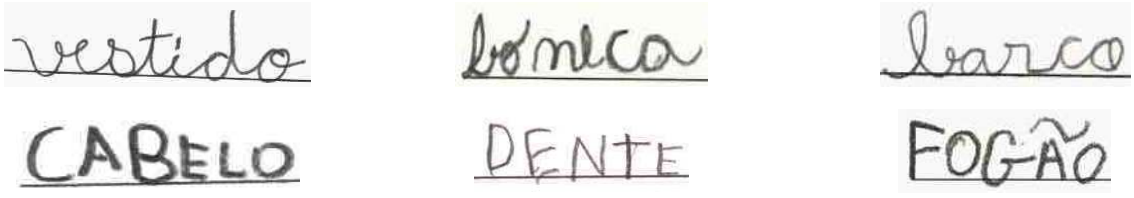
Figure 7 – Schematic representation of the dictation-based assessment workflow used to construct the dataset.



Source: Created by the author (2026).

This setting differs from that of standard HTR datasets. The images were not produced in a controlled data collection environment. Consequently, the dataset contains natural variation in handwriting style, letter shape, spacing, stroke quality, and spelling. Figure 8(a) shows words spelled correctly by different students. In contrast, Figure 8(b) presents words containing phonetic substitutions and letter omissions. These examples highlight the diversity and complexity of the handwriting samples in the dataset in terms of spelling, letter formation, and spacing.

Figure 8 – Examples of handwritten responses in literacy assessments



(a) Expected spelling



(b) Incorrect spellings

Source: Created by the author (2026).

4.2 ANNOTATION PROTOCOL

Each image in the dataset is associated with a JSON annotation file and a corresponding plain text file (`.txt`). The annotation file includes the transcription of the produced text and metadata fields that characterize the response. Spelling errors are preserved because they provide relevant information about the stage of literacy development. The plain text file provides the ground truth used to train the HTR models.

The metadata fields in the JSON annotation provide additional information about each response. For each item in the dictation task, the annotation stores the fields listed in Table 1. The fields `image`, `number`, and `id_item` identify the sample and link it to the corresponding dictation item. The fields `target_word` and `transcription` store the expected word and the text written by the student, respectively. Spelling errors are preserved in the transcription as part of the annotation.

Table 1 – Description of the annotation fields in the JSON file.

Field	**Type**	**Description**
`image`	string	Filename of the answer sheet image.
`number`	integer	Number of the question.
`id_item`	string	Unique identifier of the dictation item.
`target_word`	string	The word dictated to the student.
`transcription`	string	Transcription of what the student wrote.
`status`	string	Response status code.
`orthography`	string	Label indicating the writing development category.
`trainable`	boolean	Indicates if the response can be used for training.
`mirrored`	boolean	Flags responses with mirrored letters.

Source: Created by the author (2026).

The `status` field indicates the response condition. Its possible values are `NORMAL` for valid responses, `DEFECT` for unclear handwriting, `BLANK` when no response is provided, and `ITEM-SWAPPED` when the response does not correspond to the expected item. The `orthography` field stores the writing development category assigned to the response. This label is based on the psychogenesis of written language, which describes how children develop their understanding of how writing represents speech (FERREIRO; TEBEROSKY, 1979; FRITH, 1985). Table 2 summarizes the categories used in the annotation.

The annotation also includes the boolean flags `trainable` and `mirrored`. The `trainable` field indicates whether the response can be used for training. A response is considered trainable when the handwriting is legible. The `mirrored` field indicates whether the response contains mirrored letters. Together, these fields enable a more complete characterization of each response. To the best of our knowledge, it is the first

Table 2 – Writing development categories used in the annotation.

Category	Description
Pre-syllabic	The child does not establish a relation between letters and sounds.
Syllabic	The child begins to associate one written mark or letter with each syllable.
Syllabic-alphabetic	The child combines syllabic representations with partial phoneme representations.
Alphabetic	The child establishes a more consistent relation between letters and sounds.
Orthographic	The child writes according to conventional spelling rules.

Source: Created by the author (2026).

dataset containing Brazilian Portuguese handwritten responses produced by children in real educational assessments.

4.3 STATISTICS

The complete annotation set contains 3,882 JSON files, corresponding to 7,764 annotated responses. Each file contains annotations for two dictation items. Among these responses, 7,148 have the `NORMAL` status. Most valid responses belong to the orthographic category, which contains 5,530 instances. Smaller groups are distributed across the alphabetic, syllabic-alphabetic, syllabic, and pre-syllabic categories. Table 3 presents the distribution of valid responses by writing category.

Table 3 – Distribution of valid responses by writing category.

Writing category	Number of responses
Orthographic	5,530
Alphabetic	444
Syllabic-alphabetic	325
Syllabic	334
Pre-syllabic	515

Source: Created by the author (2026).

Table 4 presents the distribution of words in the dataset. The distribution is imbalanced because the data were collected from real educational assessments. The dataset contains many images but only 39 unique words. To increase vocabulary diversity, additional word images were extracted from responses to sentence writing activities in other assessments. The resulting expanded dataset is described in Section 4.4.

The experiments conducted in this work used a filtered subset of the complete annotation set. A response was included in the experimental subset only if it was marked

Table 4 – Distribution of target words in the dataset.

Word	Count	Word	Count	Word	Count	Word	Count
cabelo	841	campo	132	barco	92	gato	60
leque	840	martelo	130	doce	75	foca	58
picolé	821	vaca	126	famoso	73	trave	56
dente	797	cantina	125	banco	71	frango	54
sapato	778	vestido	120	girafa	71	pijama	54
sabão	741	bode	118	futuro	68	blusa	45
porta	169	mamão	113	vareta	68	jacaré	45
volume	169	sanfona	110	domingo	65	equipe	39
boneca	137	tela	109	comida	64	visita	25
pente	136	fogão	108	peteca	61		

Source: Created by the author (2026).

as `trainable`, `NORMAL` status, and belonged to the orthographic category. The experiments were limited to this category to reduce the scope of the problem and provide a more controlled evaluation of the HTR models. Responses from the other writing development categories contain fewer samples, which would result in an unbalanced comparison among categories. Therefore, this work focuses on the orthographic category, which provides the largest subset for the initial evaluation. After these requirements were applied, the experimental subset contained 5,052 word images. These images represent legible responses suitable for HTR training and evaluation. The complete annotation set remains relevant for future educational analyses.

Table 5 presents the character distribution in the transcriptions of the experimental subset. These transcriptions contained 27,957 character occurrences distributed across 29 distinct characters. The characters *a*, *o*, and *e* were the most frequent, whereas accented characters such as *â*, *ô*, and *õ* occurred infrequently. These differences show that the experimental subset was also imbalanced at the character level.

The distribution of character bigrams was also analyzed. A character bigram consists of two consecutive characters within a word. For example, the word *cabelo* contains the bigrams *ca*, *ab*, *be*, *el*, and *lo*. The transcriptions of the 5,052 word images contained 22,905 bigram occurrences, representing 143 distinct bigrams. Table 6 presents the frequency of each bigram. The sequences *ca*, *sa*, and *te* were the most frequent, occurring 1,032, 921, and 819 times, respectively. In contrast, 51 of the 143 distinct bigrams occurred fewer than 50 times, including 21 that occurred only once. This imbalance reflects the repeated occurrence of a limited set of target words in the original dictation assessments. This analysis complements the distribution of individual characters by showing the character sequences present in the experimental subset.

Table 5 – Frequency of character occurrences in the transcriptions of the 5,052 word images selected after filtering.

Character	Count	Character	Count	Character	Count
a	4,095	i	995	q	296
o	3,703	m	841	j	94
e	3,255	d	824	ó	12
t	1,930	r	694	ê	10
c	1,780	u	619	á	7
l	1,551	v	517	í	5
p	1,492	ã	503	ô	3
n	1,293	f	501	â	2
b	1,130	é	349	õ	1
s	1,123	g	332		

Source: Created by the author (2026).

Table 6 – Character bigram distribution in the experimental subset (5,052 samples, 22,905 occurrences, and 143 distinct bigrams).

Bigram	Count	Bigram	Count	Bigram	Count	Bigram	Count	Bigram	Count
ca	1,032	fo	259	mp	119	da	56	cé	2
sa	921	ue	259	go	115	fr	51	dé	2
te	819	ta	246	ng	115	av	50	ué	2
ab	696	bo	238	od	110	ij	50	vô	2
co	669	do	236	nf	105	tr	50	ár	2
nt	663	on	233	es	103	bl	44	ód	2
ol	619	eq	224	st	103	us	44	aõ	1
el	610	na	222	mã	101	ip	34	câ	1
en	545	ti	220	og	100	ui	34	cã	1
at	529	pe	217	la	99	is	22	cô	1
de	529	lu	201	gã	97	it	22	dê	1
to	529	ec	185	re	94	si	22	fá	1
pi	519	va	185	ja	93	vi	22	já	1
lo	515	in	181	rc	85	ré	13	pé	1
ão	501	ac	165	mo	72	ól	9	pó	1
ap	470	or	160	os	72	ao	6	sâ	1
pa	470	id	159	so	72	có	5	sã	1
ic	469	ra	159	éq	72	pí	5	uê	1
le	409	me	157	ce	63	té	5	ác	1
be	394	um	157	ga	62	íc	5	âb	1
am	341	ve	153	fu	61	cê	4	ân	1
an	332	vo	151	ro	61	lê	4	ãb	1
lé	323	ba	149	tu	61	vó	4	ãm	1
ar	311	fa	129	ur	61	él	4	éc	1
bã	303	ne	128	ut	61	sá	3	én	1
qu	296	oc	123	nc	60	áb	3	ét	1
rt	282	et	120	af	58	ôl	3	ór	1
po	279	mi	120	gi	58	bó	2		
ma	272	om	120	ir	58	cá	2		

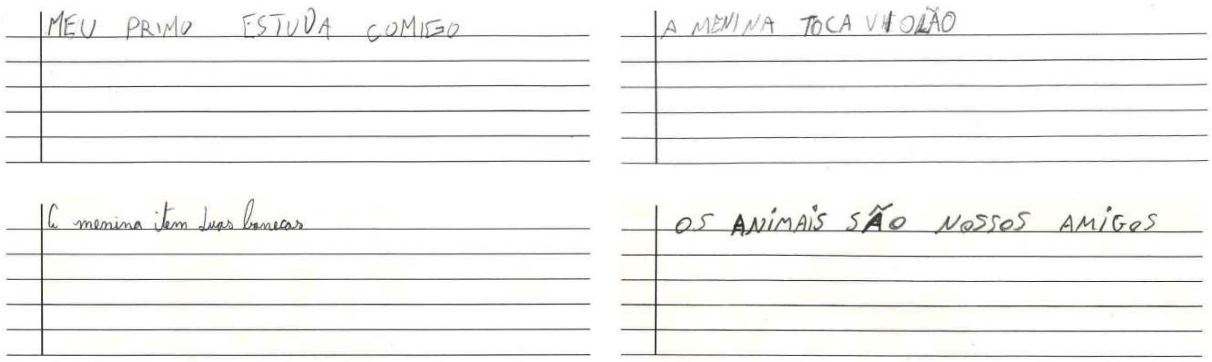
Source: Created by the author (2026).

4.4 VOCABULARY EXPANSION

The word distribution shown in Table 4 corresponds to responses collected from word dictation tasks. These tasks use a limited set of target words, some of which occur much more frequently than others. To increase lexical diversity, additional word images

were obtained from a different type of assessment item. In these items, students observe an image and write a sentence describing the scene. These responses contain longer sequences and a broader vocabulary than the dictation responses. Individual word images were then extracted from the sentences and added to the original dataset. Figure 9 shows examples of the handwritten responses used in this process.

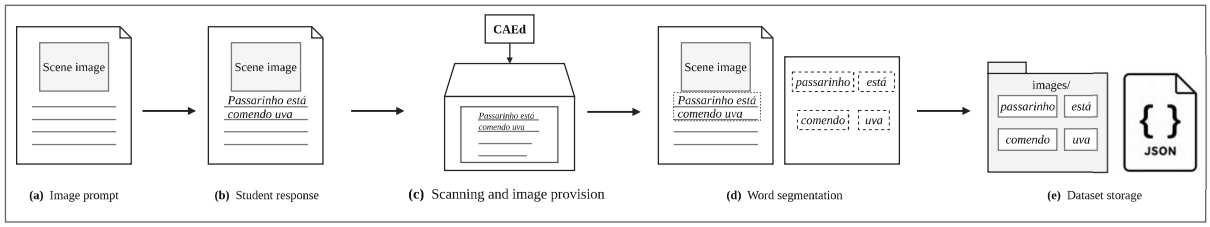
Figure 9 – Examples of handwritten sentences used for vocabulary expansion.



Source: Created by the author from the assessment materials (2026).

Each scanned image contains a long written text from which multiple words can be extracted. We extract new samples by carefully cropping subimages of correctly written words sufficiently spaced from other visual objects. They are included in the dataset marked as `NORMAL` and orthographic. This process added 1,872 word images to the original 5,052 images. As a result, the expanded dataset contains 6,924 word images.

Figure 10 – Workflow for extracting word images from sentence-writing assessment questions.

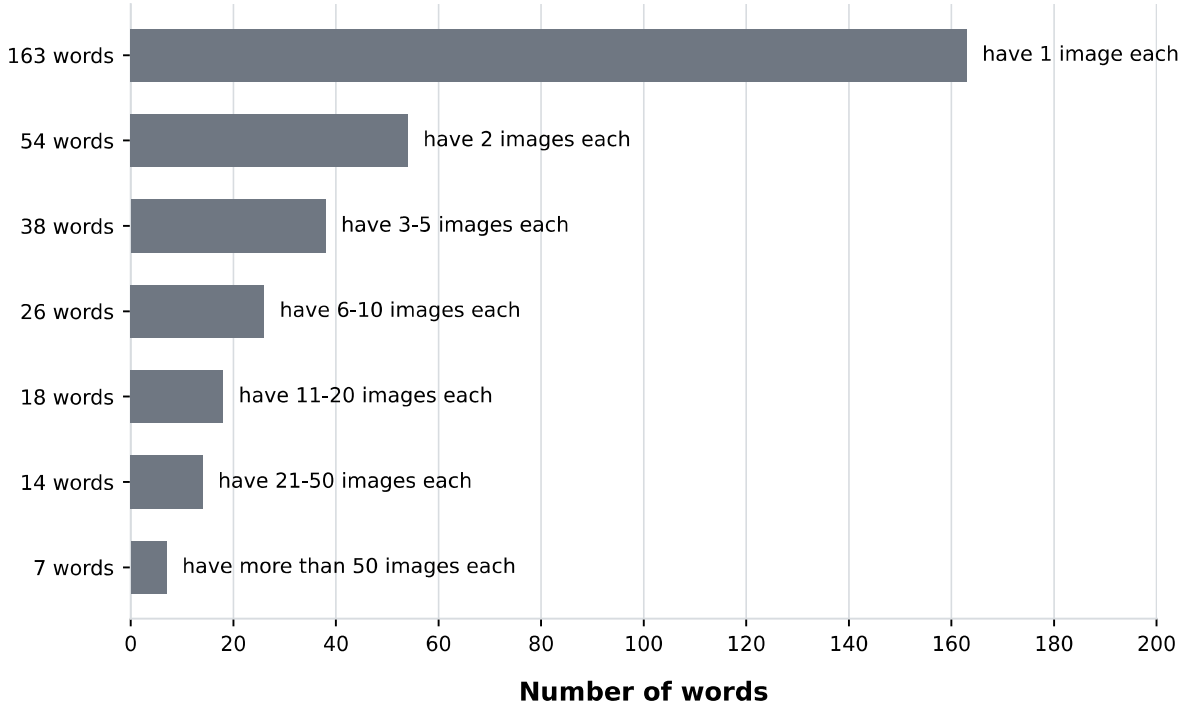


Source: Created by the author (2026).

Figure 10 illustrates the process used to expand the vocabulary from sentences written in response to scene images. The student observes a scene and writes a sentence describing it. The completed sheets were scanned, and the resulting digital images were provided by CAEd. During annotation, the sentence region was manually identified, and each word was segmented and cropped into a separate image. The resulting word images were added to the dataset and associated with a JSON annotation file containing their transcriptions and metadata.

The original dataset contained 39 target words. The additional samples are images of 313 unique words. After merging both sources, the expanded dataset contains 352 unique words. This number is not the sum of the two vocabularies because some words appear in both sources. This expansion adds more target words, different word lengths, and more combinations of characters to the dataset. Table 7 presents the frequency distribution of the 313 words obtained during vocabulary expansion. Table 7(a) lists the 157 words with more than one image, while Table 7(b) lists the 163 words with only one image in the dataset. Figure 11 summarizes this distribution by grouping the words according to the number of available images. Notice that more than half of the new words are represented by a single sample. This imbalance reflects the data collection process since the additional words were extracted randomly from naturally occurring sentences.

Figure 11 – Number of words in each image frequency range. Notice that 163 words have only one image. Only seven words have more than 50 images.



Source: Created by the author (2026).

Vocabulary expansion increased the total number of character occurrences from 27,957 to 37,210 and the number of distinct characters from 29 to 33. The additional samples introduced four characters that were not present in the original subset: “h”, “z”, “x”, and “ç”. Table 8 presents the character distribution in the expanded dataset. The characters *a*, *o*, and *e* remained the most frequent. In contrast, the four newly introduced characters occurred less frequently, ranging from 111 occurrences of *h* to a single occurrence of *ç*.

Table 7 – Distribution of words in the dataset after vocabulary expansion.

(a) Words represented by more than one image

Word	Count	Word	Count	Word	Count	Word	Count	Word	Count
menina	124	pipoca	13	tava	6	fila	3	filme	2
esta	108	toca	13	ajudando	5	homem	3	fone	2
menino	100	comeu	12	beijo	5	levando	3	frito	2
caju	76	duas	12	casa	5	sabiá	3	história	2
peteca	66	primo	12	dela	5	sua	3	juntos	2
fubá	60	tocando	12	fim	5	uso	3	lata	2
cavalo	54	leão	11	triste	5	vai	3	mamãe	2
ovo	47	flor	10	urso	5	viu	3	meia	2
vacina	43	fruta	10	beijando	4	animais	2	meninos	2
com	35	gosta	10	cobra	4	auto	2	musica	2
palito	35	que	10	fada	4	barulho	2	mãe	2
abelha	34	mão	9	frutas	4	carinho	2	nada	2
rádio	31	passarinho	9	milho	4	cena	2	olho	2
violão	28	pegando	9	minha	4	cereja	2	passaro	2
cenoura	26	boneca	8	pinta	4	cesta	2	pega	2
uma	25	comigo	8	pintando	4	coleta	2	pegar	2
bigode	24	estuda	8	seus	4	colhendo	2	pegou	2
ele	22	feijão	8	vovó	4	comer	2	perdida	2
para	22	não	8	amigo	3	costa	2	rosa	2
meu	21	pais	8	amo	3	cão	2	rua	2
pai	21	papai	8	banana	3	dele	2	segurando	2
comendo	20	comi	7	barco	3	eles	2	sozinha	2
tem	19	dia	7	bebe	3	era	2	subiu	2
ela	18	gosto	7	bem	3	escreva	2	são	2
foi	16	medo	7	bolo	3	escuro	2	televisão	2
mel	16	muito	7	bonita	3	estrada	2	tinha	2
uva	16	seu	7	bonito	3	familia	2	tirando	2
come	15	beija	6	comida	3	feliz	2	tocou	2
estava	15	cachorro	6	estar	3	ficou	2	tv	2
bonecas	14	dente	6	falou	3	filho	2	um	2
filha	13	estão	6	fez	3	filhote	2	xicara	2
mala	13	gostou	6						

(b) Words represented by one image

Word	Word	Word	Word	Word	Word	Word
acontece	cachorrinho	depois	garotinha	mima	peixe	suga
agente	cada	desenha	garoto	minhas	pela	tacando
agora	caminha	desenhado	gato	mochila	pelucia	tchau
ajuda	carrinho	desenhando	gostoso	mora	pintar	tenho
alegria	cascão	direto	gota	moro	porque	tesoura
alguem	casão	dono	ilha	mostrando	pote	tio
algumas	cem	embora	isso	nariz	quando	tira
amiga	cera	esperando	jabuticaba	navegando	quem	tocar
amigos	ceu	esperto	jacare	necta	quer	tocava
amora	ceus	esse	lapis	noite	queria	toco
andando	chapando	estavam	legal	nossos	quão	toda
ate	chorando	estou	leva	nova	raio	tomando
ato	cinema	falando	liga	nunca	rato	tão
avós	coletou	fase	lobo	oi	rei	usa
azuis	colocou	fazendo	longe	ola	rindo	vara
bala	colorindo	felizes	lua	olhando	rio	vazia
bebendo	comido	fico	lugar	olhou	saboroso	vendo
beijou	como	flauta	magico	orando	saiu	ver
bicho	contando	floresta	mais	outro	sala	vez
bico	criança	frase	mas	ouvidos	sempre	vir
boca	dado	frente	matou	paga	será	vivi
bola	dando	frutinha	meio	papais	soa	você
bom	dedo	galho	meta	pego	sou	vou
bora	dentro					

Source: Created by the author (2026).

Table 8 – Character distribution in the 6,924 word images after vocabulary expansion.

Character	Count	Character	Count	Character	Count
a	5,508	d	1,048	h	111
o	4,736	u	1,005	á	69
e	4,266	r	935	ó	16
t	2,367	v	766	z	13
c	2,301	f	666	ê	11
n	2,011	ã	585	í	5
l	1,882	g	429	ô	4
p	1,750	é	349	x	3
i	1,679	q	312	õ	1
s	1,465	j	206		
m	1,373	â	2		
b	1,335	ç	1		

Source: Created by the author (2026).

A similar increase was observed in the character bigrams. The total number of occurrences increased from 22,905 to 30,286, while the number of distinct bigrams increased from 143 to 255. Table 9 presents their frequency distribution. The sequences *ca*, *sa*, and *te* remained the most frequent, with 1,318, 946, and 927 occurrences, respectively. However, 147 bigrams occurred fewer than 50 times, and 38 were observed only once. Thus, the expansion introduced a broader variety of character sequences.

All experiments conducted in this dissertation use the expanded dataset. The goal is to reduce the dependence on a small group of repeated words and provide a broader evaluation of HTR models with a larger vocabulary.

4.5 VISUAL AMBIGUITY IN THE DATASET

Handwriting produced by children often contains visually similar characters because letter formation is still developing during early literacy. Incomplete strokes, variable proportions, and unclear internal structures can make different letters difficult to distinguish. The dataset includes several examples of visual ambiguity. The letter “a” may resemble an “o” when its internal stroke is faint or incomplete. Similarly, “r” may resemble “n” when their strokes are incomplete. The letters “d” and “p” may be difficult to distinguish depending on the formation of their vertical and rounded components. Figure 12 illustrates these cases. The red boxes mark regions in which one handwritten character may resemble another, and the labels identify the corresponding visual ambiguity.

These examples show that recognizing handwritten responses requires more than preserving the spelling errors produced by the child. The model must also distinguish characters with similar visual forms. This challenge motivates the method proposed in

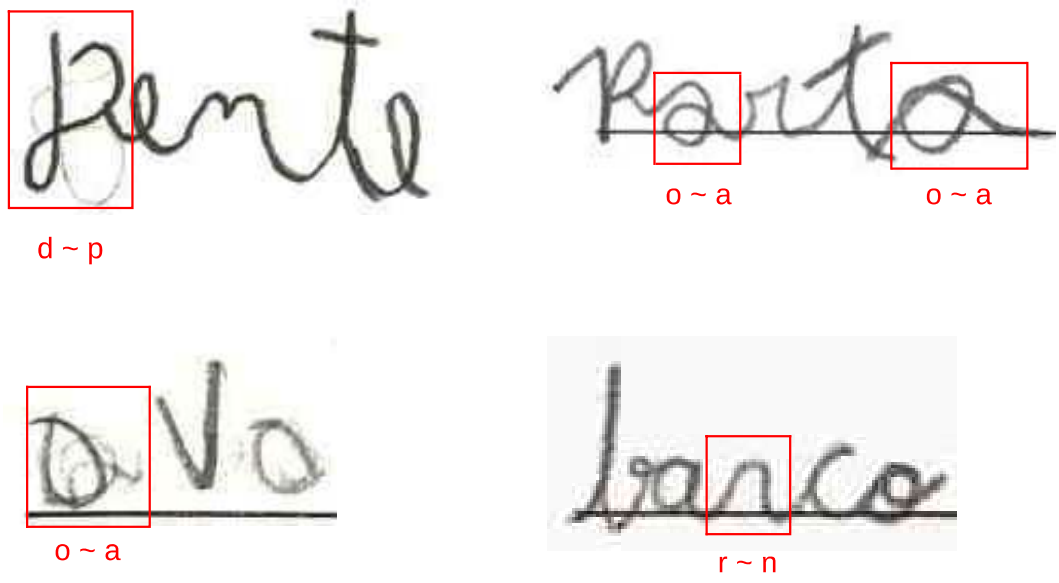
Table 9 – Character bigram distribution after vocabulary expansion (6,924 samples, 30,286 occurrences, and 255 distinct bigrams).

Bigram	Count	Bigram	Count	Bigram	Count	Bigram	Count	Bigram	Count	Bigram	Count
ca	1,318	es	264	aj	82	si	24	íc	5	ze	2
sa	946	ni	234	ut	81	tá	24	ag	4	ár	2
te	927	om	229	ro	79	eg	21	au	4	ód	2
en	837	ti	229	ij	74	se	21	br	4	ór	2
co	800	eq	224	it	73	im	20	cã	4	ae	1
ab	736	ac	218	éq	72	ia	19	eb	4	aõ	1
el	716	vo	205	fr	70	oi	19	ez	4	bu	1
nt	685	lu	204	tu	69	ru	19	gu	4	ct	1
ol	663	et	191	ha	66	ua	19	iz	4	câ	1
to	636	or	190	lh	65	uv	17	lê	4	cô	1
lo	590	ba	187	io	63	er	15	rá	4	dê	1
pa	587	um	185	ir	63	fe	14	él	4	ep	1
ão	580	go	180	ub	62	ss	14	az	3	fá	1
de	572	oc	170	nc	61	ud	14	cr	3	gr	1
pi	542	id	166	tr	60	pr	13	hi	3	iv	1
at	536	mi	157	gi	59	ré	13	sá	3	ix	1
in	487	ve	157	vi	59	ch	12	un	3	já	1
me	485	ne	154	af	58	du	12	vô	3	mb	1
ap	481	la	149	us	55	fl	12	zi	3	nu	1
ic	480	fa	141	as	53	eã	11	áb	3	nç	1
le	457	od	135	ou	52	mu	9	ád	3	oa	1
be	455	no	132	ov	52	ól	9	ãe	3	pé	1
ta	406	av	129	ip	47	ed	8	ôl	3	pó	1
na	401	fu	121	is	47	ev	8	bó	2	rq	1
an	398	mp	120	ai	46	nã	8	cu	2	sâ	1
ar	367	ng	116	eu	46	rr	8	cá	2	tc	1
am	356	al	113	ci	45	ao	7	cé	2	tó	1
va	333	mã	113	bl	44	jã	7	dé	2	uc	1
do	332	os	111	ui	42	tã	7	ej	2	uã	1
ma	323	ja	108	di	41	vó	7	ge	2	uê	1
lé	323	nf	105	li	41	iu	6	he	2	xe	1
qu	312	re	104	ri	41	jo	6	iá	2	zu	1
pe	306	ce	100	ho	40	ot	6	lg	2	ác	1
bã	303	og	100	ad	39	su	6	lm	2	âb	1
po	295	gã	97	ig	38	sã	6	oz	2	ân	1
st	284	ur	97	bá	32	cê	5	rd	2	ãb	1
rt	283	mo	93	bi	31	có	5	sp	2	ãm	1
fo	277	so	89	em	31	ob	5	tv	2	ça	1
bo	276	da	88	fi	30	pí	5	ug	2	éc	1
ec	275	rc	88	il	30	rs	5	ul	2	én	1
ue	274	ga	86	ei	29	sc	5	ué	2	ét	1
on	267	ju	84	nh	29	té	5	xi	2	ós	1
ra	265	nd	84	lã	28						

Source: Created by the author (2026).

Chapter 6.

Figure 12 – Examples of visual ambiguity between handwritten characters in word images from the dataset.



Source: Created by the author (2026).

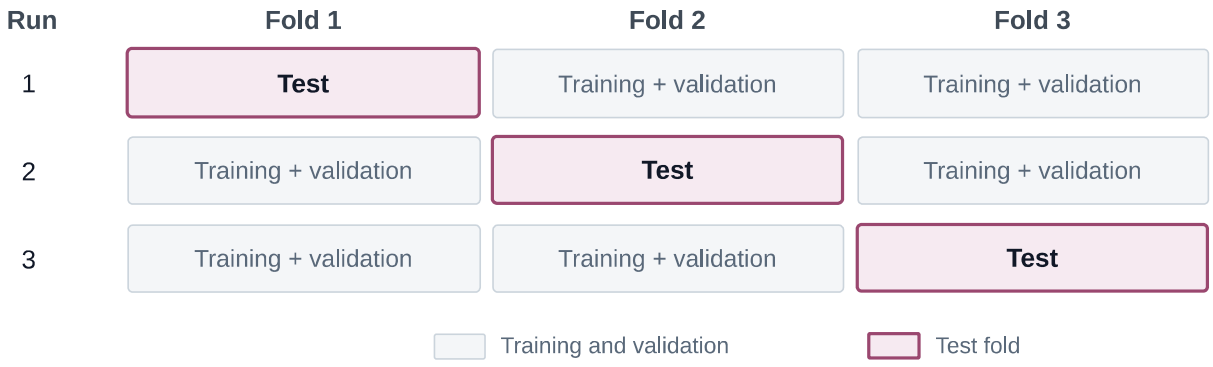
5 MODEL SELECTION FOR CHILDREN'S HANDWRITING

This chapter compares VAN, TrOCR, and E-TrOCR to identify the model most suitable for children's handwriting recognition. The models are evaluated under two three-fold protocols. In the first, the images were randomly distributed in the three folds. In the second, each fold has exclusive words, so the words used for testing are not observed during training. Both protocols are described in Section 5.1. Using three folds in both scenarios keeps the evaluation conditions comparable with bias reduction. Section 5.2 defines the metrics used to compare the models at the word and character levels. Finally, Section 5.3 presents the results and supports the selection of the recognition model used in the proposed method.

5.1 THREE-FOLD EVALUATION PROTOCOLS

The models were evaluated under two three-fold protocols. Both protocols used the expanded dataset described in Section 4.4, which contains 6,924 handwritten word images representing 352 unique words. The same training procedure and evaluation metrics were used in both protocols. The protocols differed only in how the images were assigned to the folds. Three experimental runs were then conducted. In each run, one fold was used as the test set, while the samples from the other two folds were divided into training and validation sets, as illustrated in Figure 13.

Figure 13 – Fold configuration across the three experimental runs.

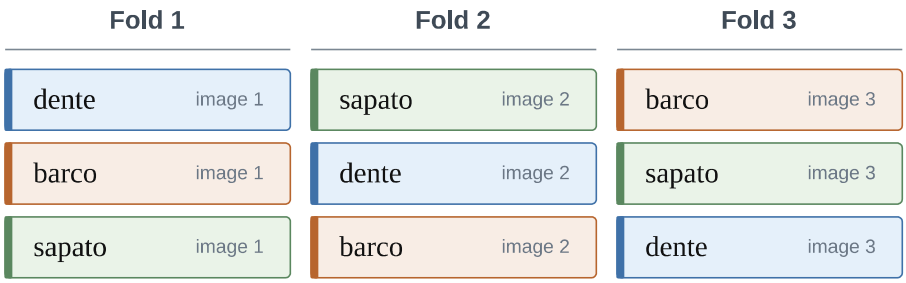


Source: Created by the author (2026).

In the random protocol, the images are randomly distributed among the three folds. As illustrated in Figure 14(a) images of the same word could consequently occur in different folds. Thus, a word evaluated in the test set could also be represented by other images in the training or validation sets. In the exclusive word protocol, shown in Figure 14(b), all images of the same word were assigned to one fold. Therefore, when a fold was used for testing, none of its words occurred in the training or validation sets.

This protocol provided a more restrictive evaluation because it measures the ability of the models to generalize beyond the training set. The data did not contain student identifiers. Therefore, it was not possible to separate the responses produced by the same student.

Figure 14 – Examples of sample distribution under both evaluation protocols.



(a) Random protocol



(b) Word-disjoint protocol

Source: Created by the author (2026).

The same three-fold structure was used in both protocols, allowing the results to be compared under the same experimental configuration. Each model is trained and tested three times in both protocols.

5.2 EVALUATION METRICS

The evaluation of HTR models requires metrics that measure how close the predicted transcription is to the reference transcription. In this work, errors are evaluated at the word and character levels. These two perspectives are complementary. Word-level accuracy measures the proportion of samples for which the complete predicted word matches the reference transcription. It is defined as:

$$Accuracy = \frac{n_c}{n}, \tag{5.1}$$

where n_c is the number of correctly recognized word images and n is the total number of evaluated samples. Notice that the prediction is considered correct only when the full word matches the reference. Therefore, a word with only one incorrect character is counted as incorrect.

Character Error Rate (CER) evaluates the difference between the predicted transcription and the reference transcription at the character level. It is based on the edit distance between the two sequences and considers substitutions, deletions, and insertions. CER is defined as:

$$CER = \frac{s + d + i}{n_{\text{char}}}, \tag{5.2}$$

where s is the number of substitutions, d is the number of deletions, i is the number of insertions, and n_{char} is the number of characters in the reference transcription. CER is useful because it captures partial recognition errors. For example, if only one character is incorrect, the metric reflects the size of this error rather than treating the entire word as an incorrect prediction.

Word-level accuracy and CER provide complementary views of model behavior. Accuracy indicates how often the complete word is recognized correctly. CER measures character-level deviations. The overall results are calculated by pooling the predictions from the three test folds. Overall Accuracy is defined as:

$$Accuracy_{\text{overall}} = \frac{\sum_{i=1}^{3} n_c^{(i)}}{\sum_{i=1}^{3} n^{(i)}}, \tag{5.3}$$

where $n_c^{(i)}$ is the number of correctly recognized word images in test fold i, and $n^{(i)}$ is the total number of images in that fold.

Overall CER is defined as:

$$CER_{\text{overall}} = \frac{\sum_{j=1}^{3} \left(s^{(j)} + d^{(j)} + i^{(j)} \right)}{\sum_{j=1}^{3} N_{\text{char}}^{(j)}}, \tag{5.4}$$

where $s^{(j)}$, $d^{(j)}$, and $i^{(j)}$ are the edit operations obtained in test fold j. The term $n_{\text{char}}^{(j)}$ is the number of characters in the reference transcriptions of that fold.

These metrics are used in this chapter to compare VAN, TrOCR, and E-TrOCR under the two three-fold evaluation protocols. They are also used in Chapter 7 to evaluate the proposed method.

5.3 RESULTS AND DISCUSSION

Table 10 presents the results obtained using three-fold cross-validation with random distribution, the test fold may contain images of words that are also present in the training folds.

TrOCR achieved the best overall performance in terms of accuracy under three-fold cross-validation with random distribution. It obtained an overall Accuracy of 94.44% and an overall CER of 1.79%. E-TrOCR obtained a lower Accuracy of 92.91%, but achieved the lowest CER of 1.63%. Thus, TrOCR produced more exact word transcriptions, whereas

Table 10 – Recognition performance using three-fold cross-validation with random distribution.

Fold	VAN		TrOCR		E-TrOCR	
	Accuracy ↑	**CER ↓**	**Accuracy ↑**	**CER ↓**	**Accuracy ↑**	**CER ↓**
1	79.86	7.26	94.28	1.84	92.29	1.74
2	79.25	7.48	94.02	1.92	92.94	1.67
3	80.42	7.06	95.02	1.60	93.50	1.49
Overall	79.84	7.27	**94.44**	1.79	92.91	**1.63**

Source: Created by the author (2026).

E-TrOCR produced fewer character errors. VAN showed lower recognition performance than the other two models, achieving 79.84% Accuracy and 7.27% CER.

In this protocol, images of the same word may appear in the training, validation, and test sets. Therefore, the models can benefit from word patterns learned during training. The results show their ability to recognize new handwriting samples, but they do not reflect their ability to recognize unseen words. To evaluate this ability, the models were also tested using the three-fold cross-validation protocol with exclusive words described in Section 5.1. In this protocol, the words in each test fold are not included in the corresponding training and validation sets. Table 11 presents the results.

Table 11 – Recognition performance on unseen words using three-fold cross-validation with disjoint words.

Fold	VAN		TrOCR		E-TrOCR	
	Accuracy ↑	**CER ↓**	**Accuracy ↑**	**CER ↓**	**Accuracy ↑**	**CER ↓**
1	24.89	30.02	39.90	18.75	72.89	7.78
2	31.07	25.95	43.27	18.52	76.36	7.02
3	37.51	26.29	37.90	20.45	68.60	8.57
Overall	31.87	27.13	40.40	19.33	**72.57**	**7.61**

Source: Created by the author (2026).

The performance of all models decreased when the testing was restricted to unseen words. The largest decrease occurred for VAN and TrOCR. VAN decreased by 47.97 percentage points, from 79.84% to 31.87%. TrOCR showed the largest decrease in accuracy, from 94.44% to 40.40%, corresponding to 54.04 percentage points. E-TrOCR showed a smaller decrease of 20.35 percentage points, from 92.91% to 72.57%. These differences indicate that the random distribution masked the limitations in the recognition of unseen character sequences, particularly for VAN and TrOCR.

To investigate these errors, each incorrect prediction was compared with the vocabulary used to train the model in the corresponding fold. A prediction was counted when its complete character sequence matched a word observed during training. For each

test fold, the percentage was calculated by dividing the number of these predictions by the total number of incorrect predictions. The overall percentage was calculated from the combined counts of the three folds. Table 12 presents the percentage of incorrect predictions that matched the training vocabulary for each model.

Table 12 – Amount of predictions of incorrect words that matched one word belonging to at least one of the training folds.

Model	**Fold 1**	**Fold 2**	**Fold 3**	**Overall**
VAN	98.07%	91.01%	88.72%	92.24%
TrOCR	54.78%	54.56%	54.61%	54.64%
E-TrOCR	3.51%	4.16%	0.00%	**2.22%**

Source: Created by the author (2026).

For VAN, 92.24% of the incorrect predictions exactly matched a sequence observed during training. This percentage remained high across the three folds, ranging from 88.72% to 98.07%. For TrOCR, training sequences represented 54.64% of the errors, ranging from 54.56% to 54.78%. In contrast, only 2.22% of the E-TrOCR errors matched a training sequence. No such problem was observed for E-TrOCR in the third fold.

Most errors produced by VAN and TrOCR consisted of replacing a word from a test fold with a word observed in the corresponding training folds. E-TrOCR also produced recognition errors, but these errors generally resulted in character sequences that were not present in the training folds. This analysis shows quantitative evidence of a bias towards familiar training sequences.

Table 13 – Examples of substitutions between structurally similar sequences. The numbers indicate how many time the substitution was applied for each model.

GT → Prediction	**Shared structure**	**VAN**	**TrOCR**	**E-TrOCR**
dente → pente	`_ente`	320	137	17
barco → banco	`ba_co`	60	30	12
foca → toca	`_oca`	29	15	2
meu → mel	`me_`	11	10	1
mamão → mamãe	`mamã_`	56	39	0
cenoura → tesoura	`_e_oura`	13	8	0
vaca → vara	`va_a`	84	0	0
boneca → bonita	`bon_a`	68	15	0

Source: Created by the author (2026).

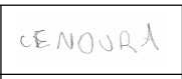
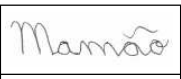
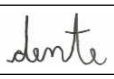
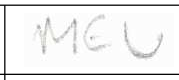
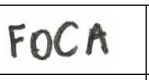
Table 13 presents examples of recurrent substitutions between structurally similar sequences. In each example, the ground truth (left word) was not present in the corresponding training data, but the predicted sequence was (right word). The shared structure

indicates the characters that remained unchanged. Each underscore represents a character position that differed between the ground truth and the prediction. The model columns report the number of test images in which the indicated substitution occurred.

These substitutions occurred more frequently for VAN and TrOCR. VAN replaced *dente* with *pente* in 320 test images, while TrOCR produced the same substitution in 137 images. E-TrOCR produced this substitution in only 17 images. Similarly, *mamão* was replaced with *mamãe* 56 times by VAN and 39 times by TrOCR, but never by E-TrOCR. VAN and TrOCR were more likely to replace unseen sequences with structurally similar sequences.

To complement this analysis, Figure 15 presents selected test images and their corresponding predictions. These examples illustrate some of the substitutions identified in the quantitative analysis.

Figure 15 – Recognition examples for unseen words. Incorrect characters are shown in red.

						
VAN	tesoura	mamão	pente	mel	foca	banco
TrOCR	cenoura	mamãe	tente	meu	toca	banco
E-TrOCR	cenoura	mamõo	dente	mel	foca	bafco

Source: Created by the author (2026).

In the selected examples, VAN transcribed *cenoura* as *tesoura*, *dente* as *pente*, *meu* as *mel*, and *barco* as *banco*. TrOCR showed the same tendency. It transcribed *mamão* as *mamãe*, *foca* as *toca*, and *barco* as *banco*. E-TrOCR correctly recognized *cenoura*, *dente*, and *foca*. In the remaining examples, its predictions differed from the ground truth by only one character.

In conclusion, E-TrOCR is the most suitable model for recognizing unseen words. It achieved the highest Accuracy and the lowest CER. In addition, only 2.22% of its errors matched words observed during training. This result indicates less dependence on the training vocabulary than VAN and TrOCR. Therefore, it was selected as the recognition model used in this dissertation. The next chapter presents the proposed method for reducing substitutions between visually similar characters.

6 PROPOSED METHOD

This chapter presents the second contribution of this work. It consists of a visual similarity-aware adaptation of E-TrOCR for children's handwriting recognition. The proposed method maintains the original encoder-decoder architecture. It changes the target distribution used during training by including information about visual similarities between characters. They are represented by a similarity matrix constructed with HOG descriptors. Section 6.1 describes the adaptation of E-TrOCR. Section 6.2 presents the construction of the visual similarity matrix. Finally, Section 6.3 defines the proposed loss function.

6.1 E-TrOCR ADAPTATION

The proposed method adopts E-TrOCR as the baseline recognition model. The first architectural adaptation consists of replacing the original vocabulary of 220 elements with a vocabulary of 123 elements designed for Brazilian Portuguese. The decoder is configured to predict 123 character and symbol classes. The original E-TrOCR vocabulary includes additional symbols for mirrored and inverted characters (RANGASRINIVASAN *et al.*, 2025). These symbols were not included in the vocabulary used in this work. The dataset contains a limited number of mirrored samples and no inverted samples, which results in insufficient training data for the model to learn these special cases. Instead, the vocabulary was adapted to include the accented characters required for Brazilian Portuguese. The composition of the resulting vocabulary is presented in Table 14. It includes special symbols, uppercase and lowercase letters, digits, space, punctuation, and accented characters.

Table 14 – Composition of the character vocabulary used in this work.

Vocabulary group	**Elements**	**Count**
Special symbols	`<pad>`, `<s>`, `</s>`, `<unk>`	4
Uppercase letters	A–Z	26
Lowercase letters	a–z	26
Digits	0–9	10
Space	Whitespace character	1
Punctuation	ASCII punctuation symbols	32
Accented lowercase	á, à, â, ã, é, ê, í, ó, ô, õ, ú, ç	12
Accented uppercase	Á, À, Â, Ã, É, Ê, Í, Ó, Ô, Õ, Ú, Ç	12
Total		**123**

Source: Created by the author (2026).

The model is trained on the children's handwriting dataset. The training uses the

proposed vocabulary containing 123 elements. The proposed visual similarity-aware loss replaces the standard cross-entropy loss. This loss was designed to account for similarities between character shapes during training.

6.2 VISUAL SIMILARITY MATRIX

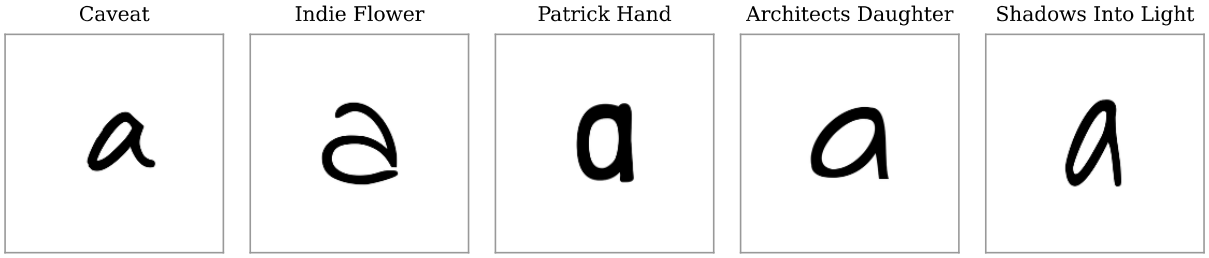
The visual similarity matrix represents the affinities between the elements of the E-TrOCR vocabulary. Let V denote the vocabulary size. The matrix is defined as:

$$S \in \mathbb{R}^{V \times V}, \tag{6.1}$$

where each row and column correspond to one vocabulary element. Thus, the resulting matrix has dimensions 123×123. Each element S_{ij} represents the visual relation between characters i and j. Higher values indicate that their shapes are more similar. The matrix is constructed before training and reused in all training runs.

The dataset contains word-level transcriptions. It does not provide isolated and reliably labeled images for every character. For this reason, synthetic character images are used to construct the matrix. Each character is rendered as a grayscale image with a resolution of 64×64 pixels. Five handwritten-style fonts are used: *Caveat*, *Indie Flower*, *Patrick Hand*, *Architects Daughter*, and *Shadows Into Light*. Figure 16 illustrates the rendering of the lowercase letter "a" using these five fonts. Using different fonts reduces dependence on a single character style. Special symbols, such as `<pad>`, `<s>`, `</s>`, `<unk>`, and space character, are not rendered because they do not have a visual form. Their corresponding rows in the similarity matrix are therefore set to zero.

Figure 16 – Rendering of the lowercase letter "a" using the five handwritten-style fonts.



Source: Created by the author (2026).

After rendering, a HOG descriptor is computed for each character image. The image color is first inverted so that the character stroke has higher values than the background. The parameters for the HOG calculation are 9 gradient orientations, 8×8 pixels per cell, and 2×2 cells per block. The blocks are normalized using L^2-Hys norm. With these parameters, each rendered image produces a feature vector with 1,764 values.

The descriptors obtained from the five fonts are concatenated. Therefore, each rendered character i is represented by:

$$\mathbf{v}_i = \left[\mathbf{h}_i^{(1)} \parallel \mathbf{h}_i^{(2)} \parallel \mathbf{h}_i^{(3)} \parallel \mathbf{h}_i^{(4)} \parallel \mathbf{h}_i^{(5)}\right], \tag{6.2}$$

where $\mathbf{h}_i^{(f)}$ is the HOG descriptor of character i rendered with font f, and $\parallel$ denotes vector concatenation. The resulting vector contains 8,820 values.

Concatenating the descriptors enables the comparison to consider different character styles. This reduces the influence of a shape that occurs in only one font. The cosine similarity between the feature vectors of characters i and j is calculated as:

$$c_{ij} = \frac{\mathbf{v}_i \cdot \mathbf{v}_j}{\|\mathbf{v}_i\|_2 \|\mathbf{v}_j\|_2}, \tag{6.3}$$

where c_{ij} are the components of initial similarity matrix C. Characters with similar HOG descriptors produce higher cosine similarity values. The calculation is performed for every pair of characters in the vocabulary.

Figure 17 presents the initial matrix C before thresholding and row normalization. Characters are grouped into uppercase letters, lowercase letters, digits, symbols, lowercase accent letters, and uppercase accent letters. The heatmap shows that the highest similarities are concentrated among accented variants of the same letter, such as “á”, “à”, “â”, and “ã”. Unaccented letters and digits generally appear in intermediate colors, indicating moderate similarity. A more detailed analysis reveals high similarities among the rounded characters, including “c”, “e”, “o”, “O”, “0”, and “Q”. The matrix also captures ambiguities observed in handwriting produced by children, such as “a” and “o”, “m” and “n”, and “n” and “r”, and “d” and “p”. In contrast, the region containing symbols predominantly has smaller similarities.

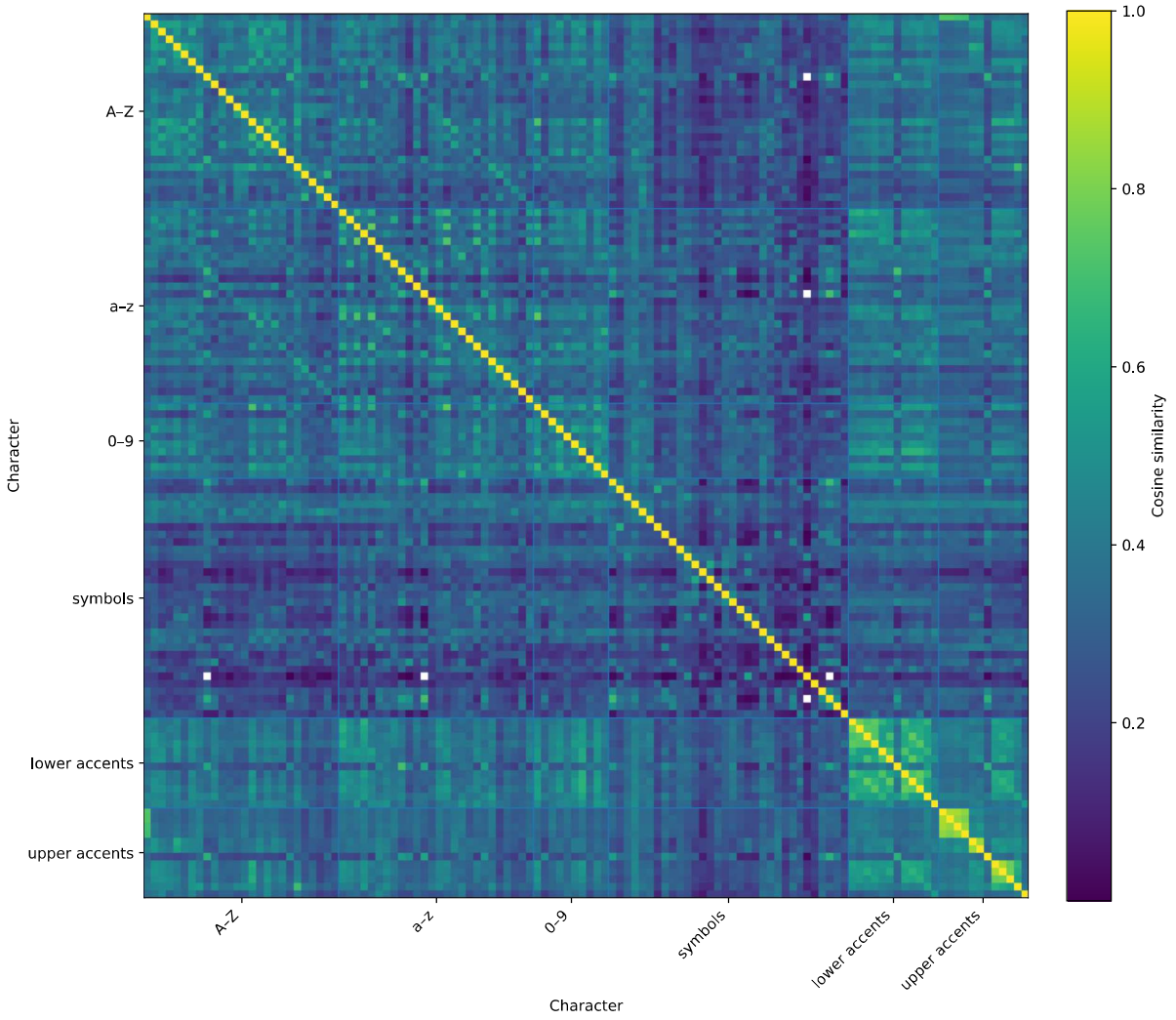
The matrix C is then processed before usage. First, the diagonal is set to zero. The similarity of a character with itself is not included because the correct class is already represented by the main term of the target distribution. Second, a threshold τ is applied. Similarity values below the threshold are set to zero. The value of τ is treated as a hyperparameter and is selected through the experiments presented in Section 7.3.

After removing the diagonal, weak similarities, and special symbols, the intermediate matrix $\widehat{S}$ is defined as:

$$\widehat{s}_{i,j} = \begin{cases} c_{i,j}, & \text{if } i \neq j \text{ and } c_{i,j} \geq \tau, \\ 0, & \text{otherwise,} \end{cases} \tag{6.4}$$

where $i, j \in \{1, \ldots, V\}$ and V is the vocabulary size. The filtered matrix $\widehat{S}$ is then

Figure 17 – Heatmap of pairwise cosine similarities between the HOG descriptors of characters in the vocabulary. Characters are grouped by category, with colors ranging from purple to yellow indicating increasing similarity.



Source: Created by the author (2026).

normalized row by row to obtain the final similarity matrix S:

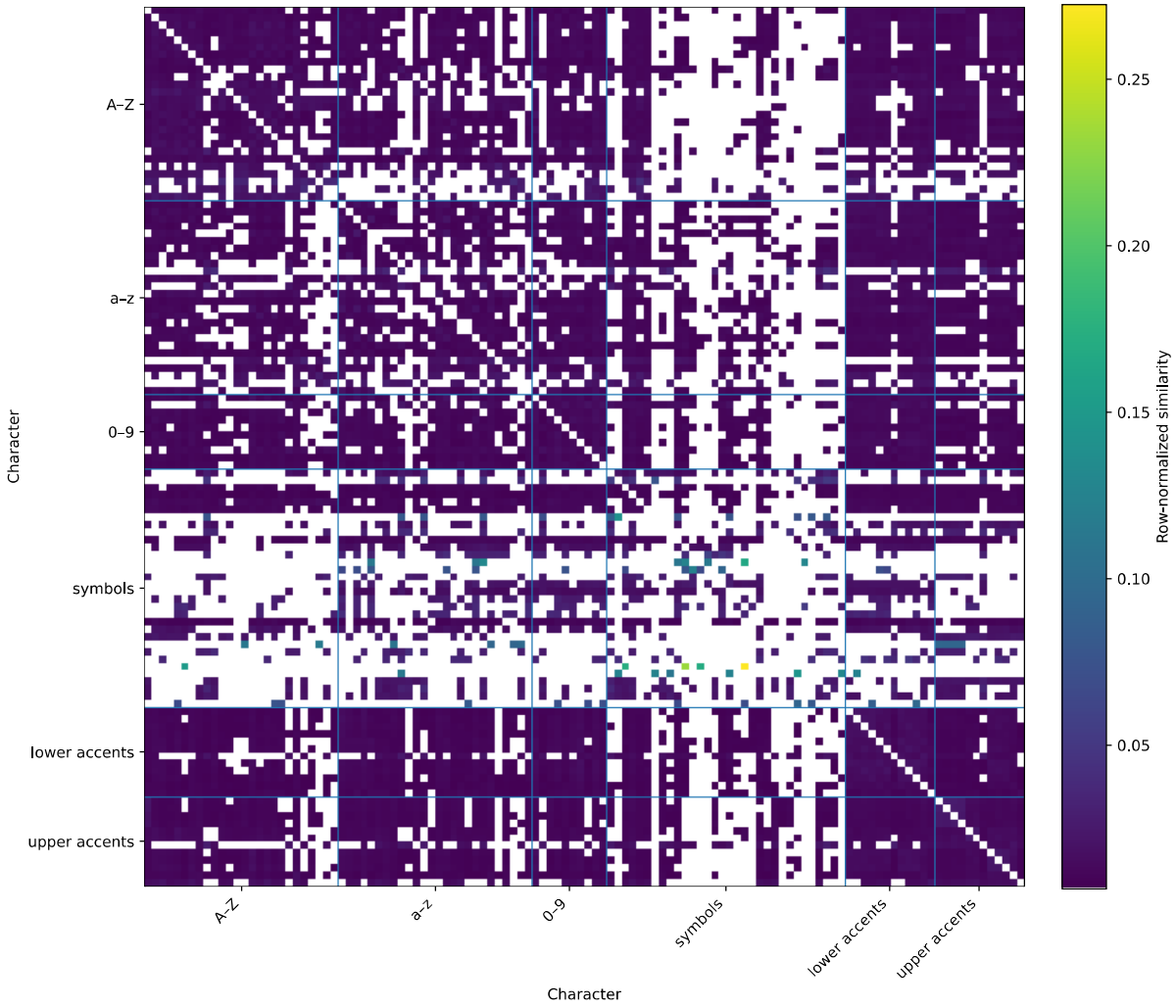
$$s_{i,j} = \begin{cases} \dfrac{\widehat{s}_{i,j}}{\sum_{k=1}^{V} \widehat{s}_{ik}}, & \text{if } \sum_{k=1}^{V} \widehat{s}_{ik} > 0, \\ 0, & \text{otherwise,} \end{cases} \tag{6.5}$$

where $s_{i,j}$ are the components of S. If a row contains at least one valid visual neighbor, its values are divided by the sum of the row. As a result, every nonzero row of S sums one. Rows without visually similar neighbors remain zero.

Figure 18 presents the final matrix S after thresholding ($\tau = 0.3$) and row normalization. White cells have zero value or were removed by the thresholding. The remaining values represent the proportion of similarity assigned to each visual neighbor. Unlike the

initial matrix C, the normalized matrix S is not necessarily symmetric because each row is normalized one by one. Therefore, the complete matrix is displayed instead of only its lower triangular portion.

Figure 18 – Visual similarity matrix S after thresholding ($\tau = 0.3$) and row normalization. The maximum value in the matrix is approximately 0.27. Most values are below 0.05.



Source: Created by the author (2026).

6.3 VISUAL SIMILARITY-AWARE LOSS

The proposed loss uses the visual similarity matrix to modify the target distribution. Let y_t denote the reference character at sequence position t, and let $\mathbf{e}_{y_t} \in \mathbb{R}^V$ denote its one-hot representation. In standard cross-entropy, the complete probability is assigned to y_t, while all other classes receive zero.

In the proposed method, part of the target probability is distributed among the visual neighbors of y_t according to the corresponding row $\mathbf{s}_{y_t,:}$. Each nonzero row of S

defines a probability distribution over these visual neighbors.

Some characters may have no visually similar neighbors after thresholding. Their corresponding rows in S remain zero. In these cases, the original one-hot target is retained. The similarity-aware target distribution is therefore defined as:

$$\widetilde{\mathbf{y}}_t = \begin{cases} (1-\epsilon)\mathbf{e}_{y_t} + \epsilon \mathbf{s}_{y_t,:}, & \text{if } \sum_{k=1}^{V} s_{y_t,k} > 0, \\ \mathbf{e}_{y_t}, & \text{otherwise} \end{cases} \tag{6.6}$$

where $\mathbf{s}_{y_t,:} = [s_{y_t,1}, s_{y_t,2}, s_{y_t,3}, \cdots, s_{y_t,V}]$ is the y_t-th row, and $\epsilon \in [0,1]$ controls the total probability assigned to visual neighbors. When the similarity row is nonzero, the reference character receives probability $1-\epsilon$. The remaining probability ϵ is distributed according to the values in $\mathbf{s}_{y_t,:}$. Because the diagonal of S is zero, the similarity term does not assign additional probability to the reference character.

Let $\mathbf{z}_t \in \mathbb{R}^V$ denote the logits produced by the decoder at position t. The softmax function converts these logits into a probability distribution over the vocabulary:

$$p_t(k) = \frac{\exp(z_{t,k})}{\sum_{j=1}^{V} \exp(z_{t,j})}, \tag{6.7}$$

where $p_t(k)$ is the probability predicted for character k at position t.

The character loss is computed as the cross-entropy between the proposed target distribution and the model prediction:

$$\ell_t = -\sum_{k=1}^{V} \widetilde{y}_{t,k} \log p_t(k). \tag{6.8}$$

For a nonzero similarity row, this expression includes a contribution from the reference character and contributions from its visual neighbors. The reference character remains the main target, while visually related characters receive smaller weights according to the corresponding similarity values. When the similarity row is zero, the target remains one-hot, and Equation 6.8 reduces to the standard cross-entropy loss.

The final loss is obtained by averaging the character losses over the set $\mathcal{T}$ of decoder positions:

$$\mathcal{L}_{\text{visual}} = \frac{1}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} \ell_t. \tag{6.9}$$

Padding positions are excluded from $\mathcal{T}$. In the implementation, the loss is computed from the decoder logits using the target distribution defined in Equation 6.6. When $\epsilon = 0$, the target becomes one-hot, and the formulation reduces to standard cross-entropy.

This formulation follows class-similarity-based label smoothing (LIU; JAJA, 2021). In this work, however, the similarity distribution is derived exclusively from visual character

representations constructed with HOG descriptors. The method does not use word frequencies, dictionaries, language rules, or an external language model. Consequently, the reference transcription remains the primary training target, including any spelling variations present in the child's response.

7 EXPERIMENTAL RESULTS

This chapter presents the experiments used to evaluate the proposed E-TrOCR adaptation based on visual similarity. Section 7.1 describes the experimental setup adopted in this dissertation. Section 7.2 defines the evaluation metrics. Section 7.3 evaluates different combinations of the smoothing coefficient ϵ and the similarity threshold τ. Section 7.4 compares the selected configuration with the E-TrOCR baseline and uniform label smoothing, which does not use the visual similarity matrix. This comparison evaluates the matrix's contribution to the recognition results. Finally, Section 7.5 discusses the main findings and limitations.

7.1 EXPERIMENTAL PROTOCOL

The experiments used the expanded dataset described in Section 4.4, which contains 6,924 word images representing 352 unique words. In addition, we used the exclusive word-disjoint protocol described in Section 5.1. The experiments used the E-TrOCR adaptation described in Section 6.1. In this adaptation, the decoder uses the character-level vocabulary containing 123 elements.

The experiments were implemented using PyTorch and Hugging Face Transformers. Training was performed on a single NVIDIA A30 GPU with 24 GB of memory. The model was trained for up to 50 epochs using AdamW with a learning rate of 5×10^{-5}, a batch size of 128, and a weight decay of 1×10^{-3}. These settings were kept fixed across all folds and experimental configurations.

7.2 EVALUATION METRICS

The proposed method was evaluated using Accuracy and CER, defined in Section 5.2. The overall values were calculated from the pooled predictions of the three test folds, as defined in Equations 5.3 and 5.4. Mean accuracy per-word is also reported as a complementary measure. In this case, accuracy is first calculated separately for each word. The resulting values are then averaged across the vocabulary:

$$Accuracy_{\text{per-word}} = \frac{1}{|\mathcal{V}|} \sum_{w \in \mathcal{V}} \frac{n_c(w)}{n(w)}, \tag{7.1}$$

where $\mathcal{V}$ is the set of evaluated words, $n_c(w)$ is the number of correctly recognized images of word w, and $n(w)$ is the total number of images of that word. This measure assigns the same weight to every word in the accuracy computation. It differs from overall accuracy (Eq. 5.3), which assigns the same weight to every image. This is interesting in the context of unbalanced datasets.

7.3 PARAMETER ANALYSIS

The proposed adaptation depends on the smoothing coefficient ϵ and the similarity threshold τ. The coefficient ϵ controls how much probability is redistributed from the reference class to its visual neighbors. The threshold τ controls which character pairs remain in the matrix. The following values were considered:

$$\epsilon \in \{0.02, 0.03, 0.05, 0.08, 0.10\}, \qquad \tau \in \{0.20, 0.30, 0.40\}. \tag{7.2}$$

In total, fifteen parameter combinations were evaluated. Each combination was evaluated on the validation subset of the three cross-validation configurations. Table 15 presents the CER obtained on these subsets.

Table 15 – Validation CER (%) for different combinations of ϵ and τ.

ϵ	τ	**Fold 1** ↓	**Fold 2** ↓	**Fold 3** ↓	**Overall** ↓
0.02	0.20	6.87	6.82	5.24	6.24
	0.30	6.44	6.31	5.31	5.97
	0.40	6.90	6.75	5.76	6.42
0.03	0.20	6.61	6.49	5.47	6.14
	0.30	**6.19**	6.20	5.31	5.87
	0.40	6.78	6.70	5.75	6.37
0.05	0.20	6.55	6.42	5.39	6.07
	0.30	6.26	**6.05**	**5.18**	**5.78**
	0.40	6.84	6.73	5.75	6.39
0.08	0.20	6.68	6.55	5.55	6.21
	0.30	6.41	6.20	5.21	5.88
	0.40	7.11	6.95	6.04	6.65
0.10	0.20	6.81	6.69	5.67	6.34
	0.30	6.48	6.36	5.34	6.01
	0.40	7.33	7.17	6.26	6.87

Source: Created by the author (2026).

The validation results show that $\tau = 0.30$ achieved the lowest overall CER for every tested value of ϵ. At this threshold, CER decreased from 5.97% with $\epsilon = 0.02$ to 5.78% with $\epsilon = 0.05$. It then increased to 5.88% with $\epsilon = 0.08$ and 6.01% with $\epsilon = 0.10$. The threshold $\tau = 0.20$ preserved weaker similarities that may provide little benefit for recognition. In contrast, $\tau = 0.40$ produced the highest CER for every value of ϵ, suggesting that it excluded useful similarities between characters. Therefore, $\tau = 0.30$ provided the best balance between removing weak similarities and preserving relevant ones. The combination $\epsilon = 0.05$ and $\tau = 0.30$ achieved the lowest overall CER on the validation data and was selected for the subsequent experiments.

7.4 QUANTITATIVE RESULTS

This section presents the quantitative results of the comparison between the E-TrOCR baseline and the proposed method. The baseline uses one-hot targets and cross-entropy. An additional experiment with uniform label smoothing is included to determine whether the improvement comes from visual similarity or only from label smoothing.

Uniform label smoothing combines the one-hot target with a uniform distribution over all vocabulary classes. For a reference character y_t at sequence position t, the uniform target distribution is defined as:

$$\widetilde{y}_{t,k}^{\text{uniform}} = \begin{cases} 1 - \epsilon + \dfrac{\epsilon}{V}, & \text{if } k = y_t, \\ \dfrac{\epsilon}{V}, & \text{otherwise.} \end{cases} \tag{7.3}$$

where V is the vocabulary size and, ϵ is the smoothing coefficient. Therefore, the reference character receives probability $1 - \epsilon + \epsilon/V$, while every other character receives probability ϵ/V. This distribution does not consider visual similarity between characters.

Three training strategies were compared. The E-TrOCR baseline uses one-hot targets and standard cross-entropy. Uniform label smoothing uses the distribution defined in Equation 7.3. The proposed method uses the visual similarity-based distribution defined in Equation 6.6. Both smoothing methods use $\epsilon = 0.05$, and the proposed method uses $\tau = 0.30$. After these parameters were fixed, each method was evaluated three times, with each fold used once as the test split. Accuracy and CER were calculated using only the predictions obtained on the corresponding test fold. The predictions from the three test folds were then averaged to calculate the overall Accuracy and CER. Table 16 presents the results.

Table 16 – Results for the baseline, uniform label smoothing, and the proposed method.

Method	**Fold 1 Acc. ↑**	**Fold 2 Acc. ↑**	**Fold 3 Acc. ↑**	**Overall Acc. ↑**	**Overall CER ↓**
E-TrOCR (cross-entropy)	72.89	76.36	68.60	72.57	7.61
Uniform label smoothing	75.39	77.99	71.89	75.04	6.78
Proposed method	**78.41**	**79.95**	**75.94**	**78.06**	**5.78**

Source: Created by the author (2026).

Uniform label smoothing increased accuracy from 72.57% to 75.04% and reduced overall CER from 7.61% to 6.78%. These results shows that label smoothing alone improved recognition in our dataset. The proposed method achieved an accuracy of 78.06% and CER of 5.78%. Compared with uniform label smoothing, it increased the accuracy by 3.01 percentage points and reduced overall CER by 1.0 percentage point. Compared to the baseline, the total improvement in accuracy was 5.49 percentage points, while overall CER decreased by 1.83 percentage points. The proposed method also achieved the highest

Accuracy in all three folds. These results show that visual similarity provided an additional improvement beyond uniform label smoothing.

Cochran's Q test was used to determine whether the three methods differed significantly in their ability to correctly recognize words. For each image in the test set, the predictions produced by the baseline, uniform label smoothing, and the proposed method were classified as correct or incorrect. The test evaluates the null hypothesis that all three methods have the same probability of correctly recognizing a word. The result showed a statistically significant difference among the methods ($Q(2) = 71.34$, $p = 3.22 \times 10^{-16}$). Therefore, the null hypothesis was rejected.

After identifying this difference, pairwise comparisons were performed using McNemar's test with continuity correction. Since three methods were compared, the Bonferroni correction was applied to the significance level, resulting in an adjusted threshold of $\alpha = 0.05/3 = 0.0167$. Table 17 presents the discordant predictions and the results obtained in each comparison.

Table 17 – Results of Cochran's Q test and the pairwise McNemar tests. The McNemar tests use continuity correction, and their p-values are compared with the Bonferroni-corrected significance level of $\alpha = 0.0167$.

Test	**Comparison**	**Only first method correct**	**Only second method correct**	**Statistic**	p
Cochran's Q	All three methods			$Q(2) = 71.34$	3.22×10^{-16}
McNemar	Baseline vs. uniform	1,248	1,416	$\chi^2(1) = 10.47$	1.21×10^{-3}
McNemar	Baseline vs. proposed	248	625	$\chi^2(1) = 161.94$	4.26×10^{-37}
McNemar	Uniform vs. proposed	1,127	1,336	$\chi^2(1) = 17.57$	2.78×10^{-5}

Source: Created by the author (2026).

The p-values obtained in the three comparisons fell below the adjusted significance level of 0.0167. Therefore, statistically significant differences were identified among the baseline, the uniform smoothing method, and the proposed method. These results confirm the performance ranking observed in terms of accuracy: the proposed method achieved the best result (78.06%), followed by the uniform smoothing method (75.05%) and the baseline (72.62%). Thus, the gain achieved by the proposed method was statistically significant relative to both the baseline and the uniform smoothing method.

In the comparison between the baseline and the proposed method, only the baseline correctly recognized the word in 248 images, whereas only the proposed method correctly recognized it in 625 images. Thus, the proposed method corrected more baseline errors than it introduced. McNemar's test confirmed that this difference was statistically significant ($\chi^2(1) = 161.94$, $p = 4.26 \times 10^{-37}$).

Cochran's Q and McNemar's tests treat each image in the test set as an individual

observation. However, the number of images per word differs across the 352 words in the dataset (Table 7). Notice that several words have a small amount of images. The per word accuracy is presented in Table 18. In this case, words with less images have a greater influence on the result.

Table 18 – Per-word accuracy (%) for the baseline and the proposed method.

Method	**Fold 1 ↑**	**Fold 2 ↑**	**Fold 3 ↑**	**Overall accuracy ↑**
E-TrOCR baseline	82.54	**87.43**	78.99	82.98
Proposed method	**85.24**	86.50	**83.87**	**85.20**

Source: Created by the author (2026).

The proposed method increased the mean per-word accuracy from 82.98% to 85.20%, an improvement of 2.22 percentage points. The mean per-word accuracy increased by 2.70 percentage points in Fold 1 and by 4.88 percentage points in Fold 3. However, in Fold 2, it decreased from 87.43% to 86.50%, a reduction of 0.93 percentage points. This reduction can be explained by the sample distribution in Fold 2. Of the 117 words in this fold, 47 are represented by only one image. For three of these words, the proposed method transcribed *galho*, *saboroso*, and *tocava* as *galha*, *saboraso*, and *tocana*, respectively. These three errors alone reduced the mean accuracy per word by approximately 2.6 percentage points.

The mean per-word accuracy gives the same weight to every word. In the dataset, 163 words are represented by only one image. For these words, accuracy can only be 0% or 100%. Therefore, the mean per-word result is more sensitive to individual predictions for words with few images. Although improvement was not observed in all folds, the overall gain is positive. This shows that the proposed method also improved performance when frequent and infrequent words receive the same weight.

To further examine the reduction in CER, the character-level errors were separated into substitutions, deletions, and insertions. These operations were obtained from the edit-distance alignments used to calculate CER. Table 19 presents the totals in the three folds.

Table 19 – Character-level edit operations produced by each method.

Method	**Substitutions ↓**	**Deletions ↓**	**Insertions ↓**
E-TrOCR baseline	2431	320	165
Proposed method	**1799**	**234**	**140**

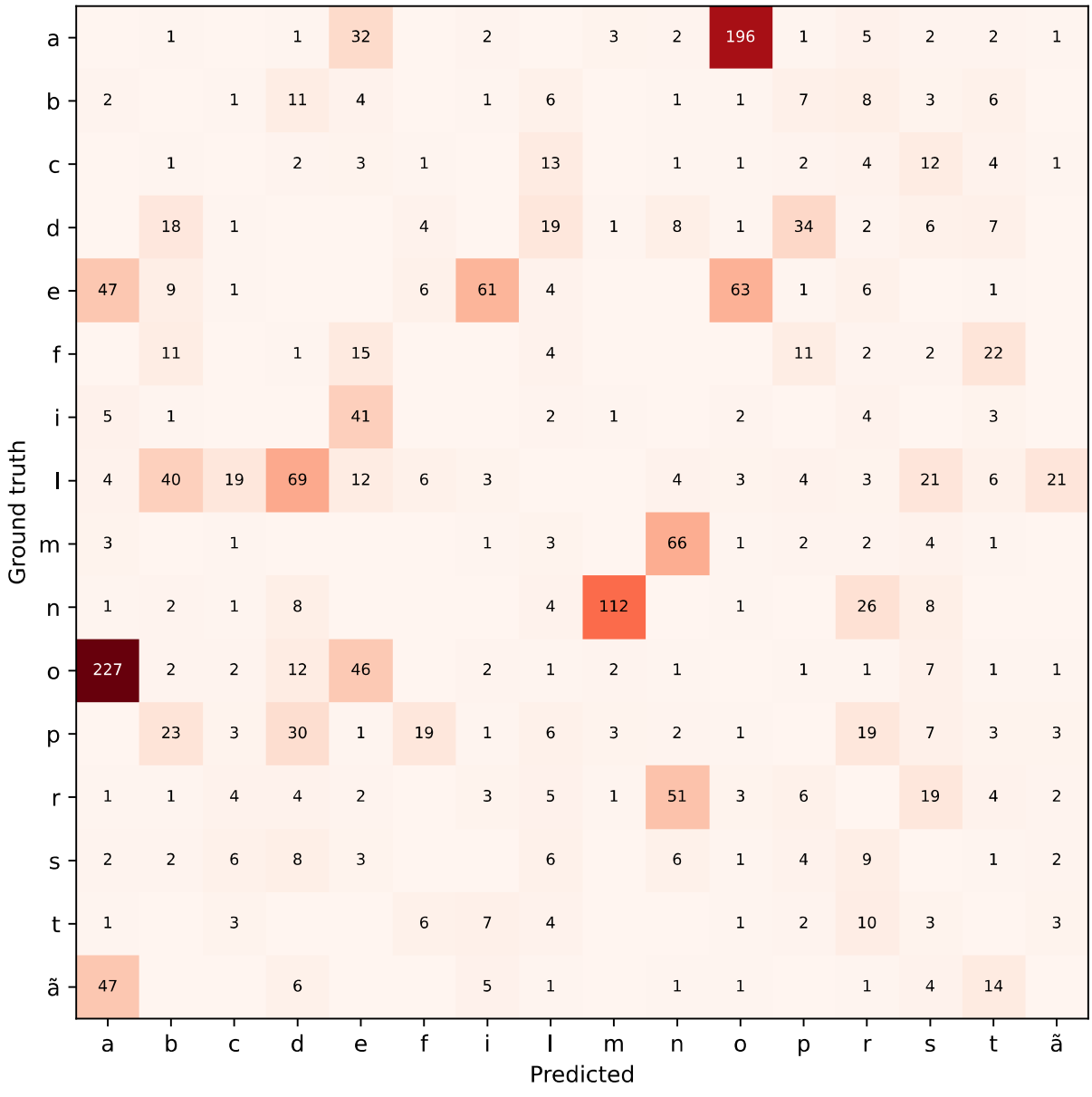
Source: Created by the author (2026).

The proposed method reduced all three types of edit operations. Substitutions decreased from 2431 to 1799, corresponding to 632 fewer errors and a relative reduction

of 26.0%. Deletions decreased from 320 to 234, corresponding to 86 fewer errors and a reduction of 26.9%. Insertions decreased from 165 to 140, corresponding to 25 fewer errors and a reduction of 15.2%. Overall, the number of character-level edits decreased from 2916 to 2173. This represents 743 fewer edits and a relative reduction of 25.5%.

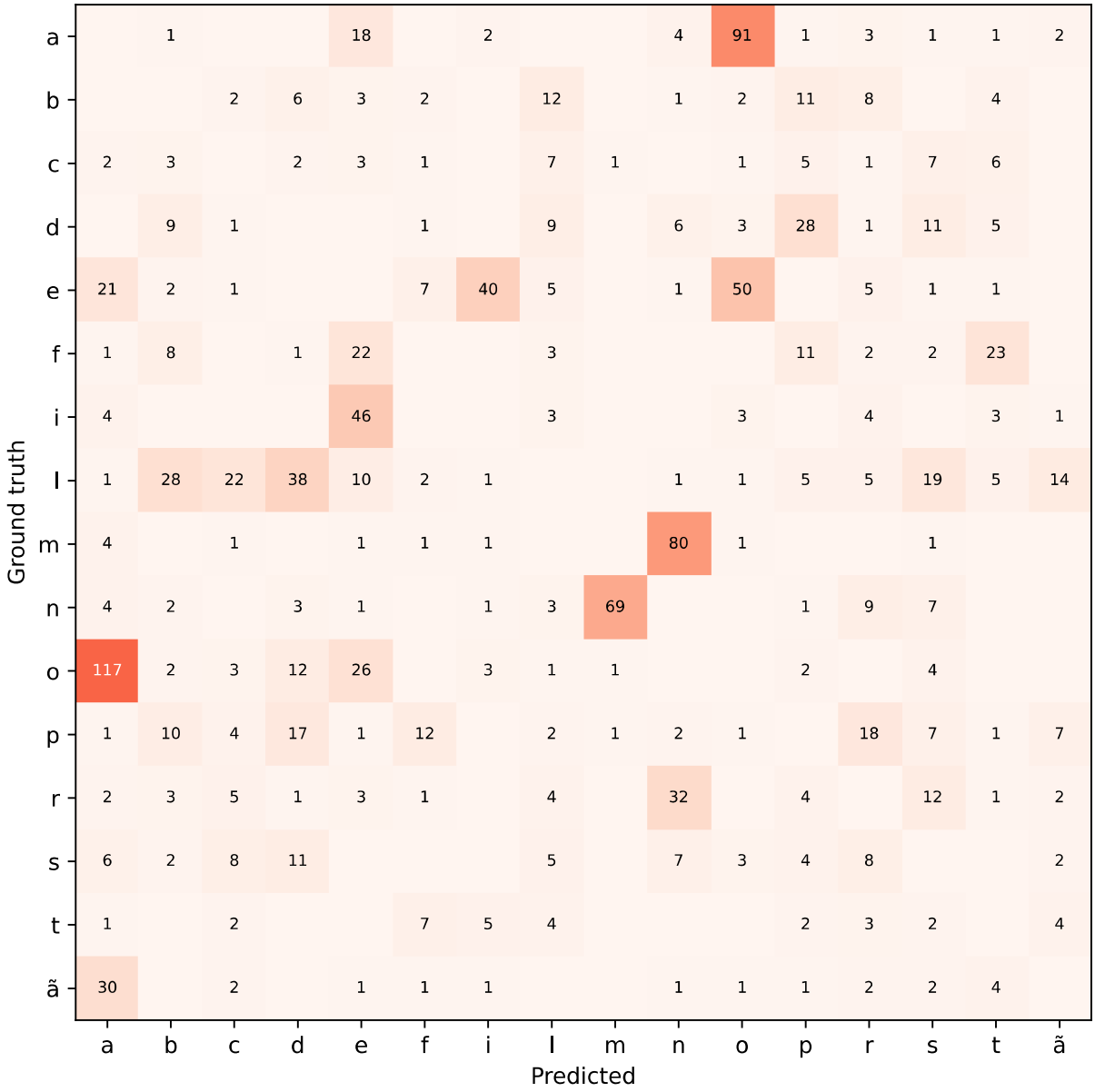
The proposed method produced 632 fewer substitution errors. This result is relevant to the proposed method because character substitutions often occur when handwritten characters have similar visual forms. Figures 19 and 20 present the direction and frequency of the substitutions produced by each method. The rows represent reference characters, and the columns represent predicted characters. The matrices show only substitution errors. Correctly recognized characters are not included.

Figure 19 – Character substitution matrix for E-TrOCR trained with cross-entropy.



Source: Created by the author (2026).

Figure 20 – Character substitution matrix for E-TrOCR trained with the proposed soft targets.



Source: Created by the author (2026).

The baseline matrix contains recurrent errors between characters with similar handwritten structures. The most frequent confusion is between “a” and “o”. Substitutions from “a” to “o” decreased from 196 to 91, while substitutions from “o” to “a” decreased from 227 to 117. The same pattern appears for other rounded characters. Substitutions from “e” to “a” decreased from 47 to 21, from “e” to “o” from 63 to 50, and from “o” to “e” from 46 to 26. These reductions indicate that the proposed method improved the distinction between characters whose shapes share curved or closed regions.

Reductions were also observed between characters formed by similar vertical strokes and arches. Substitutions from “n” to “m” decreased from 112 to 69, from “n” to “r” from

26 to 9, and from “r” to “n” from 51 to 32. For “d” and “p,” the number of substitutions decreased from 34 to 28 in one direction and from 30 to 17 in the other. These results show that the reduction involved different groups of visually similar characters represented in the matrix.

Although this pattern was observed across several character groups, it did not occur in every substitution direction. Substitutions from “m” to “n” increased from 66 to 80, while the opposite direction decreased from 112 to 69. Similarly, substitutions from “i” to “e” increased from 41 to 46, while substitutions from “e” to “i” decreased from 61 to 40. Therefore, the total number of substitutions decreased for both character pairs, even though one direction increased. The pair “f” and “e” presented a different behavior, increasing from 21 to 29 substitutions. In general, the method reduced substitutions across the character set.

7.5 DISCUSSION

The results show that adding visual similarity information to E-TrOCR improves recognition performance. In Chapter 5, E-TrOCR was selected because it generates the transcription one character at a time. This characteristic is important in literacy assessment because the sequence written by the child must be preserved. The proposed adaptation improved the recognition of characters with similar shapes. These improvements were obtained on words that were not included in training, following the exclusive word protocol.

Uniform label smoothing also improved the E-TrOCR baseline. Therefore, part of the gain can be associated with the use of softer target distributions. However, the proposed method achieved better accuracy and CER than uniform label smoothing. The difference between these results shows that the way the probability is distributed is important. Assigning a small probability to visual neighbors was more effective than distributing it equally across the complete vocabulary.

The parameter analysis also shows that visual information must be used in a controlled way. The threshold $\tau = 0.30$ produced the lowest overall CER for every value of ϵ in validation sets. The results became worse with $\tau = 0.40$, indicating that this threshold set useful character relations to zero. The best result was obtained with $\epsilon = 0.05$. As defined in Equation 6.6, this value assigns a weight of 5% to the visual similarity matrix, which was sufficient to improve the transcriptions. Larger values of ϵ increased the overall validation CER.

The reduction in character-level errors agrees with the purpose of the method. Substitutions decreased by 26.0%, including recurrent confusions involving “a” and “o”, “n” and “m”, “n” and “r”, and “d” and “p”. These pairs contain characters with similar handwritten forms and are represented in the visual similarity matrix. This result indicates that the information provided during training helped the model to distinguish several

ambiguous character pairs. Deletions and insertions also decreased, although the method was mainly designed to reduce substitutions.

The accuracy per word showed a small decrease in Fold 2. This result must be considered together with a limitation of the dataset. The number of images varies greatly among words, and 163 words have only one image. For these words, one incorrect prediction results in an accuracy of 0% or 100%, affecting the corresponding averages.

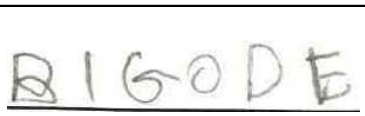
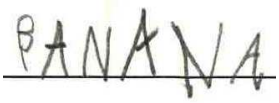
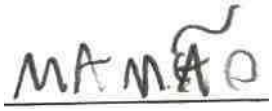
Figure 21 presents selected predictions from the baseline and the proposed method. The examples include cases where both methods were correct, cases corrected by the proposed method, and cases where the proposed method introduced a new error.

The examples show that the proposed method corrected substitutions such as “a” to “o” in “banco,” “b” to “p” in “bigode,” “e” to “o” in “cenoura,” “r” to “n” in “barco,” and “m” to “n” in “mamão”. They also show that the method preserved predictions that were already correct. However, it introduced new substitutions in examples such as “barco” and “banana”. The total number of errors decreased, but not every character pair improved. These examples are illustrative and do not show how often corrections and new errors occurred across the complete test set.

Although the proposed method achieved an overall accuracy of 78.06%, this result remains insufficient for fully automated use. The remaining 21.94% of responses contained at least one transcription error, since a prediction is considered correct only when the entire word matches the reference. In an assessment involving thousands of responses, this percentage could lead to a large number of incorrect transcriptions. Even a single character error may change the interpretation of a child’s response. Furthermore, errors introduced by the model could be mistaken for spelling errors made by the child. Therefore, the method can support the analysis of handwritten responses, but human review remains necessary.

One limitation of the proposed method concerns the construction of the similarity matrix. The HOG descriptors were obtained from characters rendered with five handwritten-style fonts. The use of several fonts reduces the dependence on a single style, but these fonts do not represent all forms found in children’s handwriting. Despite these limitations, the proposed method reduces ambiguities between visually similar characters without using a dictionary to correct the recognized words. Overall, the results show that visual similarity is a useful source of information for training E-TrOCR. The proposed method reduces recognition errors and improves the character-level output necessary for analyzing children’s handwriting.

Figure 21 – Selected qualitative results for the E-TrOCR baseline and the proposed method. The arrows indicate character substitutions.

Image	Ground truth (child transcription)	Baseline	Proposed
	avós	avós (correct)	avós (correct)
	amigo	amigo (correct)	amigo (correct)
	banco	bonco (a→o)	banco (correct)
	bigode	pigode (b→p)	bigode (correct)
	cascão	cascão (correct)	cascão (correct)
	barco	barco (correct)	bafco (r→f)
	cenoura	conoura (e→o)	cenoura (correct)
	banana	banana (correct)	panana (b→p)
	barco	banco (r→n)	barco (correct)
	mamão	manão (m→n)	mamão (correct)

Source: Created by the author (2026).

8 CONCLUSION

This dissertation sought to answer the following research question: How can words written by children be automatically transcribed while preserving the exact sequence of characters in each response? This task differs from conventional handwritten text recognition (HTR) applications, in which a model may replace the written sequence with an orthographically correct word. In educational assessments, however, substitutions and omissions in a child's response provide information about literacy development. Thus, the system must transcribe the response as written rather than correct it.

The first stage of this work involved constructing and annotating a dataset. The responses were obtained from literacy assessments applied to children in the second year of elementary school in several Brazilian states. The complete annotation set contained 7,764 handwritten responses. Each response was associated with a transcription and metadata describing its status, writing development category, legibility, and presence of mirrored characters. Of these responses, 5,052 met the selection criteria for the recognition experiments: they were legible, had the `NORMAL` status, and belonged to the orthographic category. Although this subset contained many images, it included only 39 unique words. This limited vocabulary restricted the diversity of words and character sequences available for evaluating model generalization. To expand the vocabulary, additional word images were extracted from responses to sentence writing tasks. This process added 1,872 images to the dataset. The expanded experimental set contained 6,924 images and 352 unique words. The expansion also introduced additional characters and increased the variety of character combinations represented in the dataset. The dataset used in this dissertation is owned by CAEd/UFJF. The samples provided for this research did not contain participant identifiers. Any access, sharing, or future use of the data must be authorized by CAEd.

The evaluation of VAN, TrOCR, and E-TrOCR showed that a random split of images can overestimate the ability of the models to generalize. When evaluated exclusively on words not observed during training, all three architectures showed lower performance, with a more decline for VAN and TrOCR. Error analysis further revealed that VAN and TrOCR more frequently generated complete character sequences present in the corresponding training folds, i.e. they are more susceptible to word-level auto-correction. E-TrOCR achieved the best performance under the disjoint word protocol. The model also showed less dependence on the training vocabulary. Among its incorrect predictions, only 2.22% exactly matched complete sequences observed in the corresponding training folds. These results indicate that character-level generation is better suited to the objective of this research. Therefore, E-TrOCR was selected as the basis for the proposed method. Its decoder vocabulary was adapted to contain 123 elements, including accented characters required for Brazilian Portuguese.

Despite the advantages of E-TrOCR, error analysis revealed recurring substitutions between visually similar characters. To address this issue, an adaptation of the loss function based on visual similarity was proposed. Each character in the vocabulary was rendered using five handwriting fonts. HOG descriptors were extracted from the resulting images and concatenated to represent the character across the different fonts. Cosine similarity was then calculated between each pair of character representations to construct a similarity matrix. This matrix was incorporated into the loss function by assigning a small weight to characters visually similar to the reference character. The method uses only visual information and does not rely on dictionaries, word frequencies, linguistic rules, or text corpora that could favor orthographically correct words.

The proposed method increased overall accuracy from 72.57% of the baseline to 78.06% for words not seen during training. It also reduced the Character Error Rate (CER) from 7.61% to 5.78%. The uniform label smoothing technique also improved recognition performance but yielded results inferior to those of the proposed method. This comparison provides evidence that the visual relationships represented in the matrix contributed to the observed improvements.

The proposed method correctly recognized 625 images that were misrecognized by the baseline, while introducing errors in 248 images that the baseline recognized correctly. Mean accuracy per word also increased from 82.98% to 85.20%. This metric assigns equal weight to every word, regardless of its frequency. At the character level, the total number of edit operations decreased from 2,916 to 2,173, representing a relative reduction of 25.5%. Substitutions decreased by 26.0%, including reductions in recurring confusions involving “a” and “o”, “n” and “m”, “n” and “r”, and “d” and “p”. These results are consistent with the objective of the method.

Some limitations should be considered. The similarity matrix was constructed from synthetic character images rendered using five handwriting fonts. Although using different fonts reduces dependence on a single writing style, these fonts do not represent the full variability of children’s handwriting. The dataset also remains imbalanced: 163 of the 352 words are represented by only one image. Furthermore, the experiments used only responses from the orthographic category. The results therefore do not establish that the same performance would be achieved in other writing development categories, in which character and sequence formation may show greater variability. The dataset did not include student identifiers, making it impossible to determine whether the same child completed multiple assessment sheets or to ensure that all responses from the same child were assigned to the same fold. Finally, the achieved accuracy remains insufficient for using the model in real applications without human review. These limitations reinforce the need for further research.

Future work could increase the number of available images for each word and

improve the balance of character and sequence distributions. The method should also be evaluated across other categories of writing development. Another research direction would be to investigate large language models (LLMs) for automatically classifying responses into the five writing development categories. The model could compare the expected word with the word written by the child and identify patterns such as the number of letters used to represent syllables, letter-sound correspondences, and character omissions. Based on these patterns, it could estimate the probability associated with each category. To refine the proposed method, the similarity matrix could be constructed or adjusted using characters extracted directly from handwriting produced by children. Future studies could also investigate confidence estimation methods to direct the most uncertain transcriptions to human review automatically.

In summary, this dissertation contributed to the construction and characterization of a dataset of Brazilian Portuguese children's handwriting. Using this dataset, a comparative evaluation of HTR architectures was conducted. The results revealed that some architectures have difficulty recognizing unseen character sequences. E-TrOCR achieved the best results because its character-level output reduced the tendency to replace the sequence written by the child with familiar words. Despite this advantage, E-TrOCR still had difficulty distinguishing visually similar characters. To address this limitation, a method that incorporates visual similarity into E-TrOCR training was proposed. The results provide evidence that visual relationships between characters improve recognition and reduce errors while helping to preserve the sequence written by the child. Although the model is not yet sufficiently accurate for use without human review, this work provides a foundation for tools that support the analysis of handwritten responses in literacy assessments.

REFERENCES

AIRES, S. B. K.; GIANISELLA, M. E. G. P.; MATOS, S. N.; PASSOS, M. V. S.; SANTOS, T. J. Computer vision-assisted literacy: recognizing children's handwritten words. In: **Proceedings of the 21st International Conference on Computer Vision Theory and Applications (VISAPP)**. Setúbal: SCITEPRESS, 2026. v. 2, p. 341–350.

ASSELBORN, T.; JOHAL, W.; TLEUBAYEV, B.; ZHEXENOVA, Z.; DILLENBOURG, P.; MCBRIDE, C.; SANDYGULOVA, A. The transferability of handwriting skills: from the cyrillic to the latin alphabet. **npj Science of Learning**, v. 6, p. 6, 2021.

BAO, H.; DONG, L.; PIAO, S.; WEI, F. BEiT: BERT pre-training of image transformers. In: **International Conference on Learning Representations (ICLR)**. [S.l.: s.n.], 2022.

BAUER, A.; ALAVARSE, O. M.; OLIVEIRA, R. P. de. Avaliações em larga escala: uma sistematização do debate. **Educação e Pesquisa**, v. 41, n. esp., p. 1367–1382, 2015.

BLUCHE, T. Joint line segmentation and transcription for end-to-end handwritten paragraph recognition. **arXiv preprint arXiv:1604.08352**, 2016.

CASCIANELLI, S.; CORNIA, M.; BARALDI, L.; CUCCHIARA, R. Boosting modern and historical handwritten text recognition with deformable convolutions. **International Journal on Document Analysis and Recognition**, v. 25, p. 207–217, 2022.

COQUENET, D.; CHATELAIN, C.; PAQUET, T. End-to-end handwritten paragraph text recognition using a vertical attention network. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 45, n. 1, p. 508–524, 2023.

DALAL, N.; TRIGGS, B. Histograms of oriented gradients for human detection. In: **IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)**. [S.l.]: IEEE, 2005. v. 1, p. 886–893.

ENDLICH, D.; LENHARD, W.; MARX, P.; RICHTER, T. Measuring spelling skills: a meta-analysis of the convergence of dictation and error detection tasks. **Educational Research Review**, v. 48, p. 100706, 2025.

FERREIRO, E.; TEBEROSKY, A. **Los sistemas de escritura en el desarrollo del niño**. México: Siglo XXI Editores, 1979.

FRITH, U. Beneath the surface of developmental dyslexia. In: PATTERSON, K. E.; MARSHALL, J. C.; COLTHEART, M. (Ed.). **Surface dyslexia: neuropsychological and cognitive studies of phonological reading**. Hillsdale: Lawrence Erlbaum Associates, 1985. p. 301–327.

FUJITAKE, M. DTrOCR: decoder-only transformer for optical character recognition. In: **Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)**. [S.l.]: IEEE, 2024. p. 8010–8020.

GARRIDO-MUÑOZ, C.; CALVO-ZARAGOZA, J. On the generalization of handwritten text recognition models. In: **Proceedings of the IEEE/CVF Conference on**

Computer Vision and Pattern Recognition (CVPR). [S.l.]: IEEE, 2025. p. 15275–15286.

GARRIDO-MUñOZ, C.; RíOS-VILA, A.; CALVO-ZARAGOZA, J. Handwritten text recognition: a survey. **arXiv preprint arXiv:2502.08417**, 2025.

GHADEKAR, P.; KHANVILKAR, O.; KHARAT, J.; JOSHI, K.; KULKARNI, T.; KULKARNI, Y. Automating the grading of handwritten examinations through the integration of optical character recognition and machine learning algorithms. **Journal of Integrated Science and Technology**, v. 13, n. 6, p. 1128, 2025.

GRAVES, A.; LIWICKI, M.; FERNÁNDEZ, S.; BERTOLAMI, R.; BUNKE, H.; SCHMIDHUBER, J. A novel connectionist system for unconstrained handwriting recognition. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 31, n. 5, p. 855–868, 2009.

GROSICKI, E.; EL-ABED, H. ICDAR 2011 french handwriting recognition competition. In: **2011 International Conference on Document Analysis and Recognition (ICDAR)**. [S.l.]: IEEE, 2011. p. 1459–1463.

HUANG, S.; LUO, Y.; ZHUANG, Z.; YU, J.-G.; HE, M.; WANG, Y. Context-aware selective label smoothing for calibrating sequence recognition model. In: **Proceedings of the 29th ACM International Conference on Multimedia (MM '21)**. New York: Association for Computing Machinery, 2021. p. 4591–4599.

HUSSAIN, R.; RAZA, A.; SIDDIQI, I.; KHURSHID, K.; DJEDDI, C. A comprehensive survey of handwritten document benchmarks: Structure, usage and evaluation. **EURASIP Journal on Image and Video Processing**, v. 2015, n. 1, p. 46, 2015.

KANG, L.; RUSIÑOL, M.; FORNéS, A.; RIBA, P.; VILLEGAS, M. Unsupervised adaptation for synthetic-to-real handwritten word recognition. In: **Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)**. [S.l.]: IEEE, 2020. p. 3502–3511.

LI, M.; LV, T.; CHEN, J.; CUI, L.; LU, Y.; FLORENCIO, D.; ZHANG, C.; LI, Z.; WEI, F. TrOCR: transformer-based optical character recognition with pre-trained models. **Proceedings of the AAAI Conference on Artificial Intelligence**, v. 37, n. 11, p. 13094–13102, 2023.

LIU, C.; JAJA, J. Class-similarity based label smoothing for confidence calibration. In: **Artificial Neural Networks and Machine Learning – ICANN 2021**. Cham: Springer, 2021. (Lecture Notes in Computer Science, v. 12894), p. 190–201.

LIU, Y.; OTT, M.; GOYAL, N.; DU, J.; JOSHI, M.; CHEN, D.; LEVY, O.; LEWIS, M.; ZETTLEMOYER, L.; STOYANOV, V. RoBERTa: a robustly optimized BERT pretraining approach. **arXiv preprint arXiv:1907.11692**, 2019.

MAJORANO, M.; BASTIANELLO, T.; BODEA-HATEGAN, C.; FANTUZZI, P.; FONTANA, G.; HOSTE, E.; LOMBARDI, M.; STANDAERT, A.; TALAS, D.; TRIFU, R.; VESCOGNI, L.; PERSICI, V. Early literacy skills and later reading and writing performance across countries: the effects of orthographic consistency and preschool curriculum. **Child & Youth Care Forum**, v. 50, p. 1063–1085, 2021.

MARTI, U.; BUNKE, H. The IAM-database: an english sentence database for offline handwriting recognition. **International Journal on Document Analysis and Recognition**, v. 5, n. 1, p. 39–46, 2002.

NETO, A. F. S.; BEZERRA, B. L. D.; ARAÚJO, S. S.; SOUZA, W. M. A. S.; ALVES, K. F.; OLIVEIRA, M. F.; LINS, S. V. S.; HAZIN, H. J. F.; ROCHA, P. H. V.; TOSELLI, A. H. BRESSAY: a brazilian portuguese dataset for offline handwritten text recognition. In: **Document Analysis and Recognition – ICDAR 2024**. Cham: Springer Nature Switzerland, 2024. (Lecture Notes in Computer Science, v. 14805), p. 315–333.

NETO, A. F. S.; BEZERRA, B. L. D.; ARAúJO, S. S.; SOUZA, W. M. A. S.; ALVES, K. F.; OLIVEIRA, M. F.; LINS, S. V. S.; HAZIN, H. J. F.; ROCHA, P. H. V.; TOSELLI, A. H. ICDAR 2024 competition on handwritten text recognition in brazilian essays: BRESSAY. In: **Document Analysis and Recognition – ICDAR 2024**. Cham: Springer Nature Switzerland, 2024. (Lecture Notes in Computer Science, v. 14809), p. 345–362.

NIOLAKI, G. Z.; NEGOITA, A.; VOUSDEN, J. I.; TERZOPOULOS, A. R.; TAYLOR, L.; MASTERSON, J. What spelling errors can tell us about the development of processes involved in children's spelling. **Frontiers in Psychology**, v. 14, p. 1178427, 2023.

PORTELA, C.; PALOMINO, P.; CHALLCO, G.; SOBRINHO, A.; CORDEIRO, T.; MELLO, R.; DERMEVAL, D.; BITTENCOURT, I.; ISOTANI, S. AI in education unplugged support equity between rural and urban areas in brazil. In: **Proceedings of the 13th International Conference on Information & Communication Technologies and Development (ICTD)**. New York: ACM, 2024. p. 143–154.

PUIGCERVER, J. Are multidimensional recurrent layers really necessary for handwritten text recognition? In: **2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)**. [S.l.]: IEEE, 2017. v. 1, p. 67–72.

RANGASRINIVASAN, S.; S., S. S. M.; SETLUR, S.; JAYARAMAN, B.; GOVINDARAJU, V. From scribbles to text: a novel transformer-based recognition model for child handwriting. In: **Document Analysis and Recognition – ICDAR 2025**. Cham: Springer, 2025. (Lecture Notes in Computer Science, v. 16023).

SILVA, L. C. E.; SOBRINHO, A.; CORDEIRO, T.; SILVA, A. P. D.; DERMEVAL, D.; MARQUES, L. B.; BITTENCOURT, I. I.; SANTOS, J.; MELO, R. F.; PORTELA, C. S.; SOUZA, M. R. A.; PEREIRA, R. L.; YASOJIMA, E. K. K.; ISOTANI, S. Assessing students' handwritten text productions: a two-decades literature review. **Expert Systems with Applications**, v. 250, p. 123827, 2024.

SZEGEDY, C.; VANHOUCKE, V.; IOFFE, S.; SHLENS, J.; WOJNA, Z. Rethinking the inception architecture for computer vision. In: **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**. [S.l.]: IEEE, 2016. p. 2818–2826.

TOUVRON, H.; CORD, M.; DOUZE, M.; MASSA, F.; SABLAYROLLES, A.; JÉGOU, H. Training data-efficient image transformers and distillation through attention. In: **Proceedings of the 38th International Conference on Machine Learning (ICML)**. [S.l.]: PMLR, 2021. v. 139, p. 10347–10357.

VASWANI, A.; SHAZEER, N.; PARMAR, N.; USZKOREIT, J.; JONES, L.; GOMEZ, A. N.; KAISER, L.; POLOSUKHIN, I. Attention is all you need. **Advances in Neural Information Processing Systems**, v. 30, 2017.

WANG, W.; WEI, F.; DONG, L.; BAO, H.; YANG, N.; ZHOU, M. MiniLM: deep self-attention distillation for task-agnostic compression of pre-trained transformers. **Advances in Neural Information Processing Systems**, v. 33, p. 5776–5788, 2020.

YOUSEF, M.; BISHOP, T. E. OrigamiNet: weakly-supervised, segmentation-free, one-step, full page text recognition by learning to unfold. In: **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)**. [S.l.]: IEEE, 2020. p. 14710–14719.

ZIN, T. T.; THANT, S.; PWINT, M. Z.; OGINO, T. Handwritten character recognition on android for basic education using convolutional neural network. **Electronics**, v. 10, n. 8, p. 904, 2021.