Reconhecimento de Texto para Sistemas Air Writing: Um Estudo Experimental

Carlos. E. S. Barbosa[1], Thiago B. Pereira[1], Israel M. do Carmo[1], Richard J. M. G. Tello[1], Francisco A. Boldt[1], Thiago M. Paixão[1]

[1]Instituto Federal de Educação, Ciência e Tecnologia do Espírito Santo (IFES) - Serra, ES, Brasil

{carlos.barbosa,thiago.pereira,israel.carmo}@estudante.ifes.edu.br

{richard,franciscoa,thiago.paixao}@ifes.edu.br

Abstract. *This study explores Air Writing (AW) as a contactless human-machine interface for text input, assessing its feasibility with Optical Character Recognition (OCR). AW enables users to write in the air without physical surfaces, presenting challenges for recognition accuracy and user intent detection. Quantitative experiments using open-source OCR algorithms on simulated AW data demonstrate promising results, particularly with the implementation of stroke smoothing techniques. This research contributes valuable insights into improving AW's practicality and OCR performance, aiming to enhance its usability across various interactive applications.*

Resumo. *Este estudo explora o Air Writing (AW) como uma interface humano-máquina sem contato para entrada de texto, avaliando sua viabilidade com Reconhecimento Óptico de Caracteres (OCR). O AW permite que os usuários escrevam no ar sem superfícies físicas, apresentando desafios para a precisão de reconhecimento e detecção de intenção do usuário. Experimentos quantitativos utilizando algoritmos de OCR de código aberto em dados simulados de AW demonstram resultados promissores, especialmente com a implementação de técnicas de suavização de traços. Esta pesquisa oferece insights valiosos para melhorar a praticidade do AW e o desempenho do OCR, com o objetivo de aprimorar sua usabilidade em diversas aplicações interativas.*

1. Introdução

Na última década, a interação com o mundo digital evoluiu significativamente. Telas sensíveis ao toque e outros dispositivos eletrônicos se tornaram comuns para realizar processamento de dados e conexão à internet. O Metaverso, a realidade virtual e a realidade aumentada lideram essa transformação, projetando informações diretamente nos olhos dos usuários através de óculos especializados [Abir et al. 2021].

As tecnologias da próxima geração buscam eliminar a dependência de dispositivos intermediários, como smartphones, mesas digitalizadoras ou smartwatches. De particular interesse, tem-se o Air Writing (AW), tecnologia que possibilita a interface humano-máquina por meio da escrita "no ar" sem a necessidade de qualquer contato físico com superfícies, como quadros ou papel [Chen et al. 2016]. Em termos simples, o AW pode ser utilizado para criar livremente formas e palavras em formato digital. Um exemplo de interface AW pode ser visto na Figura 1.

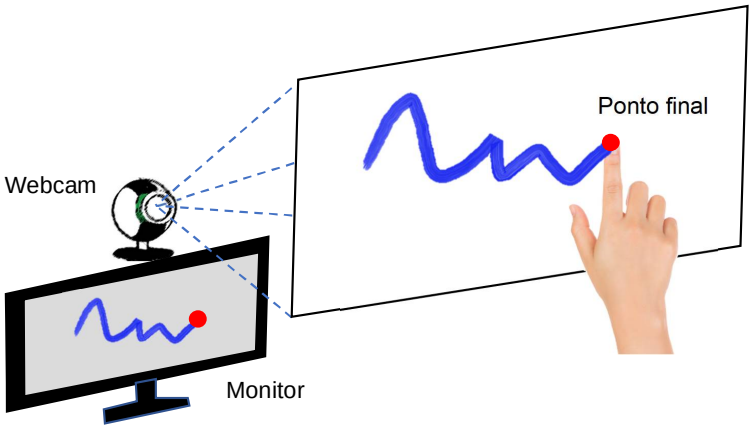


Figura 1. Exemplo de interface AW

Nos últimos anos, o interesse em publicações relacionadas com AW tem aumentado significativamente. Esse interesse tem relação com a demanda de sistemas com interfaces sem contato físico, demanda esta que se tornou mais acentuada com a pandemia de COVID-19 [Elshenaway and Guirguis 2021]. Nesse contexto, AW poderia ser útil permitindo o usuário interagir com um caixa eletrônico e/ou realizar assinaturas sem contato com telas ou canetas [Bashir et al. 2011]. AW também pode ser interessante para aplicações educacionais de alfabetização e desenvolvimento de coordenação motora [Itaguchi et al. 2015], para jogos digitais e, inclusive, como forma alternativa de comunicação e apoio (ex. pessoas com dislexia [Vaidya et al. 2022]). Para cada contexto, diferentes formas de sensoreamento podem ser utilizadas para promover interação AW. Segundo Elshenaway e Guirguis (2021), as principais formas são: ondas de rádio, dispositivos, sensores vestíveis e visão computacional. A Figura 2 apresenta um detalhamento dessas abordagens.

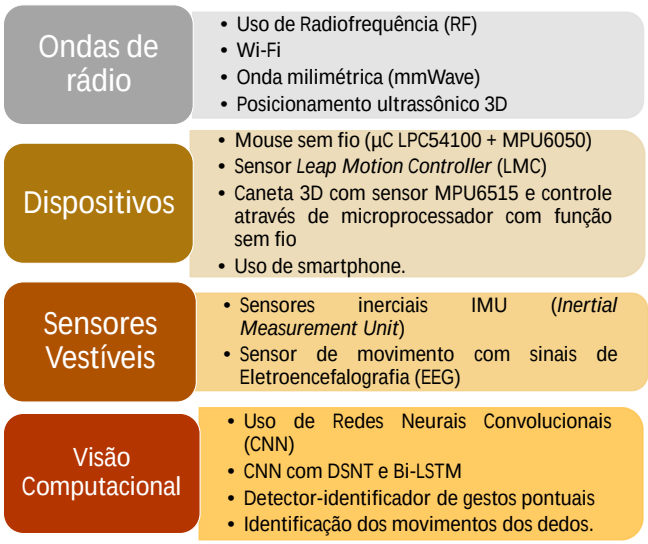


Figura 2. Categorias e exemplos de abordagem da tecnologia Air Writing segundo Elshenaway e Guirguis (2021).

O tipo de sensoreamento abordado neste trabalho é totalmente realizado por meio de visão computacional. Como estudo de caso, implementamos um sistema AW que permite controlar a passagem de slides e realizar desenhos sobre eles. O sensor utilizado para comunicação com o sistema consiste de uma câmera RGB que captura, frame a frame, a configuração espacial das mãos de um usuário. Baseado em tal configuração, o sistema realiza diferente ações, como desenhar ou escrever, passar slides e encerrar a execução do software.

O foco deste trabalho é investigar a viabilidade do reconhecimento textual via OCR (do inglês *Optical Character Recognition*) a partir da escrita livre no ar com o sistema AW. No contexto da escrita, o AW mostra-se notavelmente mais flexível e intuitivo em comparação com o reconhecimento gestual tradicional, no qual gestos estáticos (poses) ou dinâmicos pré-definidos são associados a tokens como caracteres e palavras [Lee and Kim 2021]. Contudo, a escrita livre no ar impõe dois desafios principais. Primeiro, identificar a intenção de escrever em detrimento de apenas deslocar a mão: na escrita tradicional a intenção é inerente ao contato com uma superfície (quadro ou papel). Segundo, a ausência de uma superfície de apoio e de outros delimitadores padrões prejudicam a caligrafia [Mukherjee et al. 2019], que pode se apresentar ruidosa e significativamente diferente da caligrafia comumente utilizada pelo usuário. Além disso, o reconhecimento de caracteres manuscritos é, por si só, um problema computacional mais complexo do que o reconhecimento de caracteres tipográficos.

No nosso estudo, investigamos a viabilidade de dois OCRs populares de código aberto sob os pontos de vista quantitativo e qualitativo. Um experimento quantitativo foi realizado com uma base de dados de texto manuscrito (219.510 amostras), adicionando ruído para simular a escrita livre no ar. Este experimento também serviu para verificar a eficácia do algoritmo de suavização de traços por meio de interpolação para o reconhecimento textual. Posteriormente, amostras obtidas com nosso sistema AW foram utilizadas para análise qualitativa, identificando situações de êxito ou limitação no reconhecimento. Em síntese, o estudo revela que o algoritmo de suavização tem impacto positivo no reconhecimento e que é possível reconhecer texto com um CER (do inglês Character Error Rate) de 4,94%.

As seções a seguir abordam, respectivamente: o sistema Air Writing, a metodologia experimental, os resultados e discussão, e, por fim, a conclusão do trabalho.

2. Sistema Air Writing

O sistema AW baseado em visão computacional é composto por três módulos principais (Figura 3): captura de dados, interação AW e reconhecimento textual. A captura de dados é realizada por meio de uma webcam a uma taxa de 30 frames por segundo (fps). As imagens capturadas possuem dimensões 640×480. O algoritmo de equalização de histograma CLAHE (*Contrast Limited Adaptive Histogram Equalization*) foi aplicado para ajuste do contraste apresentado na imagem, o que é particularmente eficaz em ambientes com pouco controle de iluminação, ou seja, com iluminação insuficiente ou excessiva. Essa implementação ajuda a oferecer uma iluminação uniforme, favorecendo a visualização da pose do usuário, principalmente se tratando de webcams de baixo custo.

A interação humano-sistema AW é realizada totalmente por meio de gestos estáticos e movimentos com as mãos. Um gesto estático corresponde a uma pose (ou

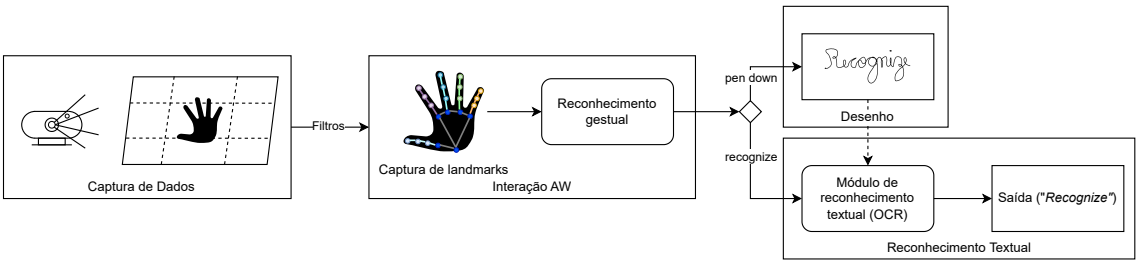
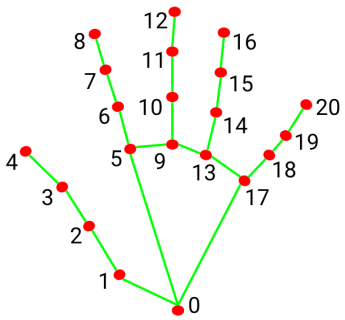


Figura 3. Módulos do sistema AW.



0. WRIST
1. THUMB_CMC
2. THUMB_MCP
3. THUMB_IP
4. THUMB_TIP
5. INDEX_FINGER_MCP
6. INDEX_FINGER_PIP
7. INDEX_FINGER_DIP
8. INDEX_FINGER_TIP
9. MIDDLE_FINGER_MCP
10. MIDDLE_FINGER_PIP
11. MIDDLE_FINGER_DIP
12. MIDDLE_FINGER_TIP
13. RING_FINGER_MCP
14. RING_FINGER_PIP
15. RING_FINGER_DIP
16. RING_FINGER_TIP
17. PINKY_MCP
18. PINKY_PIP
19. PINKY_DIP
20. PINKY_TIP

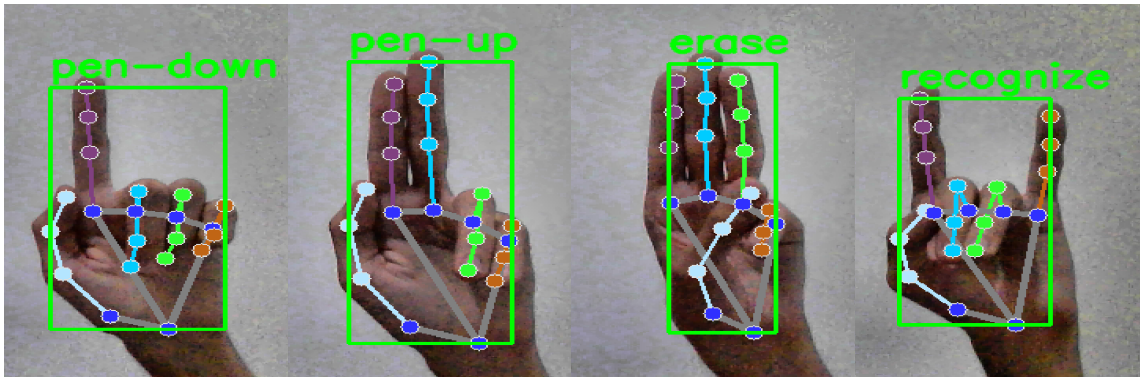
Figura 4. Modelo de mão adotado pelo MediaPipe Hands.

configuração) instantânea das mãos. No presente sistema, o reconhecimento gestual serve para controlar ações específicas do sistema. A combinação da ação de ativar escrita com o movimento da mão produz os desenhos na tela. Ao executar a ação de reconhecer texto, o conteúdo desenhado na tela é processado pelo módulo de reconhecimento textual, que cujo resultado é o reconhecimento realizado pelo OCR ativo no momento. As seções a seguir apresentam em mais detalhes os módulos de interação AW e reconhecimento textual.

2.1. Interação AW

O reconhecimento gestual é realizado por heurísticas que consideram a posição relativa de pontos de referência (landmarks) da mão e a distância entre um subconjunto desses pontos (os detalhes das heurísticas foram omitidos por fugirem ao escopo deste trabalho). O sistema AW assume que a interação é realizada com apenas uma das mãos. No caso de ambas as mãos estarem presentes na cena, o sistema baseia suas ações naquela que foi reconhecida em primeiro lugar. A detecção dos landmarks é realizada em tempo real, frame a frame, com o auxílio do framework MediaPipe Hands [Zhang et al. 2020]. O modelo geométrico das mãos é ilustrado na Figura 4.

Apesar da simplicidade e eficiência, cada heurística é restrita ao gesto em questão, ou seja, se o gesto para realizar uma ação for trocado, uma nova heurística deve ser implementada. Como alternativa, métodos mais gerais (ex.: baseados em redes neurais) foram testados; contudo, a precisão do reconhecimento não foi satisfatória em comparação às heurísticas. Para os propósitos deste trabalho, as principais funcionalidades relacionadas ao reconhecimento gestual são (Figura 5): **1. pen down:** ativar escrita; **2. pen up:** mostrar cursor (escrita inativa); **3. erase:** apagar tela; **4. recognize:** reconhecer texto desenhado na tela. Para desenhar na tela, o usuário deve movimentar a mão mantendo o gesto *pen down* ativo. A posição da ponta do indicador (posição 8 na Figura 4) é usada



(a) pen down (b) pen up (c) erase (d) recognize

Figura 5. Gestos utilizados para controlar o sistema AW.

como referência para posicionar o cursor de desenho. No caso de sentenças ou palavras compostas, o cursor de escrita deve se mover com a escrita inativa. Isso é realizado movimentando a mão mantendo o gesto *pen up* ativo.

Uma funcionalidade importante do sistema é a suavização dos traços após o desenho. Além de melhorar o aspecto visual, essa funcionalidade favorece a etapa posterior de reconhecimento de texto. Para esta finalidade, foi aplicado um algoritmo de interpolação por *spline* nos pontos capturados durante o processo de escrita, isto é, enquanto o gesto *pen down* estava ativo. O resultado é um traçado mais natural e legível, que compensa a instabilidade natural do processo de escrita no ar.

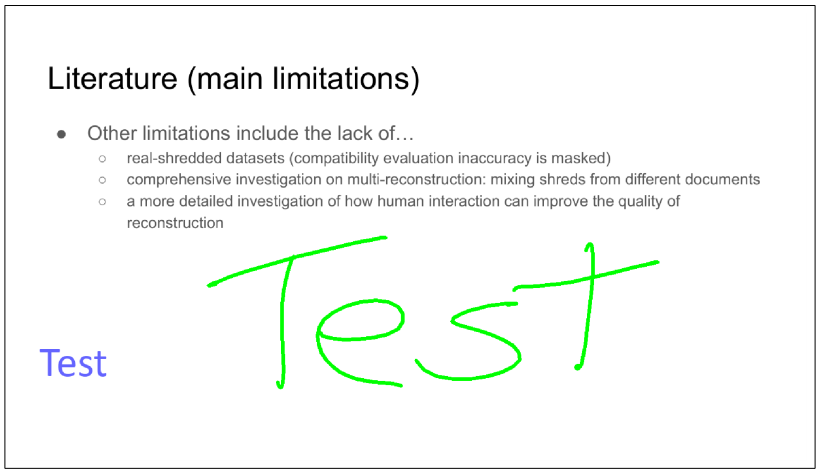


Figura 6. Sistema AW para controle de slides com reconhecimento textual. A palavra "Test" foi desenhada (em verde) pelo usuário e o reconhecimento (em azul) é exibido no canto inferior esquerdo da tela.

2.2. Reconhecimento Textual

Em sistemas AW, o reconhecimento textual permite formas de interação mais sofisticadas, além de possibilitar busca por conteúdo textual. No sistema proposto, o reconhecimento é realizado a partir do conteúdo atual desenhado sobre a tela. A Figura 6 ilustra uma tela cujo *background* é o conteúdo de um slide de uma apresentação, e o *foreground* consiste dos traços desenhados pelo usuário. Além de renderizar os traços na tela, o

sistema mantém uma imagem binária (*bitmap*) de dimensões iguais à área de desenho (slide) em que apenas as posições desenhadas estão ativas. Uma vez que o usuário realiza o gesto *recognize*, uma área de interesse delimitando o texto no bitmap é passada para o OCR para posterior reconhecimento. No exemplo em questão, a palavra "Test" foi devidamente reconhecida.

A modalidade de reconhecimento empregada no sistema AW é de reconhecimento de caracteres manuscritos, tendo em vista que o texto é proveniente de uma escrita manual. Neste estudo, foram investigados dois modelos de OCR treinados em bases de dados manuscritas: *Handwritten Text Recognition* (HTR) [Vloison and Xiwei 2021][1] e TrOCR [Li et al. 2021]. O HTR é um OCR baseado em CRNN (Convolutional Recurrent Neural Network) - que é uma rede que consiste em camadas convolucionais para extração de características, seguidas de uma rede recorrente (LSTM) -, enquanto o TrOCR é baseado na arquitetura Transformer [Vaswani et al. 2017]. Além do resultado compatível com o estado da arte, esses modelos foram escolhidos por estarem publicamente disponíveis e por possuírem código aberto, o que favorece a transparência, a reprodutibilidade dos experimentos e a possibilidade de personalização e adaptação para necessidades específicas.

3. Metodologia Experimental

A investigação experimental deste trabalho abrange uma análise quantitativa e qualitativa do reconhecimento textual. Para viabilizar a análise quantitativa, é desejável uma base de dados com diversas amostras e um certo grau de variabilidade. Nesse sentido, o caminho adotado foi utilizar uma base pública de sentenças manuscritas e aplicar ruído em diferentes graus de distorção para simular as distorções inerentes à escrita no ar. As amostras (com e sem ruído) foram utilizadas para o reconhecimento pelos modelos HTR e TrOCR, e a taxa de erro (CER) foi calculada de acordo com a equação

$$CER(\%) = \frac{\text{Número de caracteres incorretos}}{\text{Número total de caracteres no texto de referência}} \times 100. \qquad (1)$$

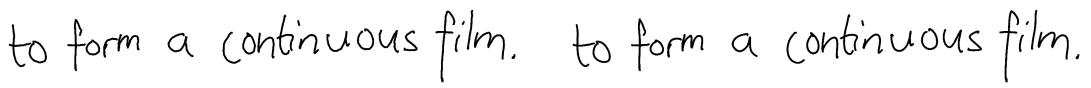
Os modelos foram avaliados com suavização de traços ativada e desativada, de modo a permitir verificar a eficácia desse procedimento. Para análise qualitativa, o OCR de melhor desempenho no experimento quantitativo foi utilizado para reconhecimento em um conjunto de amostras (palavras) reduzido obtidos com o sistema AW. A seguir, são fornecidos mais detalhes sobre o processo de formação da base de dados e da plataforma computacional utilizada nos experimentos.

3.1. Base de Dados

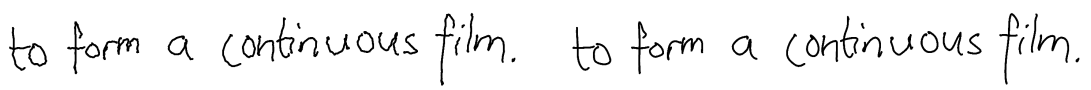
Para avaliação quantitativa dos OCRs, foi montada uma base de dados a partir de sentenças manuscritas da coleção IAM Online Dataset[2], que inclui um total de 12.195 amostras com sentenças na língua inglesa - por isso a escolha desta base de dados, haja visto que os modelos foram pré-treinados na língua inglesa. A coleção IAM é dita online

[1]Este modelo está disponível em repositório público (`https://github.com/vloison/Handwritten_Text_Recognition`), porém, até onde sabemos, não está associado a um preprint ou publicação revisada por pares.

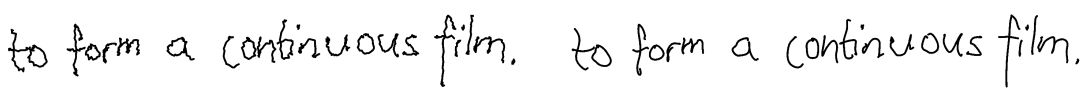
[2]Base de dados disponível em: `https://fki.tic.heia-fr.ch/databases/iam-on-line-handwriting-database`



(a) $\sigma = 0$ **(sem ruído)**



(b) $\sigma = 4$



(c) $\sigma = 8$

Figura 7. Amostras com diferentes graus de distorção (σ). A coluna da esquerda representa amostras imediatamente após aplicação do ruído, enquanto a coluna da direita representam as amostras após suavização de traços.

pois as amostras são coletadas em tempo real enquanto os autores escrevem, capturando não apenas a imagem final do texto, mas também o traçado da caneta no formato de sequência temporal de pontos. Desta forma, cada amostra i da coleção pode ser representada como um conjunto de coordenadas $\{(x_t, y_t)\}_{t=1}^{N_i}$, onde t indica o tempo discretizado e N_i é o número total de pontos na amostra i.

Para simular as distorções inerentes à escrita no ar, cada ponto (x_t, y_t) é perturbado por um vetor $\Delta\mathbf{r}_t = (\Delta x_t, \Delta y_t)$, resultando em novas coordenadas $\mathbf{r}'_t = (x_t + \Delta x_t, y_t + \Delta y_t)$. Aqui, Δx_t e Δy_t são amostras de uma distribuição normal com média zero e desvio padrão σ, que representa o grau de distorção, isto é, $\Delta x_t \sim \mathcal{N}(0, \sigma^2)$ e $\Delta y_t \sim \mathcal{N}(0, \sigma^2)$. Para gerar a base de dados, foi considerado $\sigma = 0, 1, 2, \ldots, 8$, onde 0 indica ausência de ruído, e também, as amostras com e sem suavização de traços. Logo, foram geradas um total de 219.510 amostras ($12.195 \times 9 \times 2$). A partir das novas coordenadas, a amostra (sentença) pode ser renderizada em formato de imagem lingando os pontos de uma mesma palavra na sequência temporal. A Figura 7 apresenta algumas amostras em diferentes graus de distorção.

3.2. Plataforma Computacional

Hardware: AMD Ryzen 5 CPU @ 3.30GHz com 16GB de RAM, rodando Linux Ubuntu 22.04.4 LTS, e equipado com uma GPU NVIDIA Geforce GTX 1650 com 4GB de memória. Software: O código fonte do HTR foi escrito em Python 3.6.8, usando PyTorch 1.0.1 para treinamento e inferência. O código fonte do TrOCR foi escrito em Python 3.11.7, usando PyTorch 2.3 para treinamento e inferência. Os modelos TrOCR e HTR estão disponíveis, respectivamente, no HuggingFace[3] e no repositório GitHub[4].

4. Resultados e Discussão

O gráfico na Figura 8 exibe o CER médio (%) para os OCRs investigados em função do grau de distorção das amostras. Os resultados consideram dois cenários: com e sem

[3]`https://huggingface.co/microsoft/trocr-large-handwritten`
[4]`https://github.com/vloison/Handwritten_Text_Recognition`

aplicação da suavização de traços. Numa análise geral, é notável a diferença de desempenho entre os OCRs. Para amostras com ausência de ruído ($\sigma = 0$), o CER obtido com o TrOCR é quase 15 p.p. inferior em relação ao HTR. Como esperado, o aumento do grau de distorção implica em aumento do CER. No entanto, a curva de aumento é significativamente mais acentuada para o HTR, o que significa que seu desempenho relativo ao TrOCR piora em cenários mais realísticos (maior grau de distorção).

As Tabelas 1 e 2 oferecem uma visão alternativa dos resultados ilustrados na Figura 8. A Tabela 1 mostra que, para o HTR, a eficácia da suavização de traços é observada para $\sigma \geq 2$, podendo chegar a quase 16 p.p. no caso extremo ($\sigma = 8$). Em relação ao TrOCR (Tabela 2), a suavização implica em leve aumento de CER para $\sigma < 4$, com uma diferença máxima de 0.18 p.p. para $\sigma = 1$. Contudo, para cenários mais realísticos caracterizados por dados mais ruidosos, a suavização de traços torna-se útil, atingindo no caso extremo ($\sigma = 8$) uma redução no CER de 1.48 p.p.. Considerando o melhor desempenho de ambos os OCRs no caso extremo, o TrOCR apresentou um CER aprox. 19 p.p. inferior ao obtido com o HTR. Deste modo, o TrOCR com suavização de traços foi escolhido para ser integrado ao sistema AW para análise qualitativa.

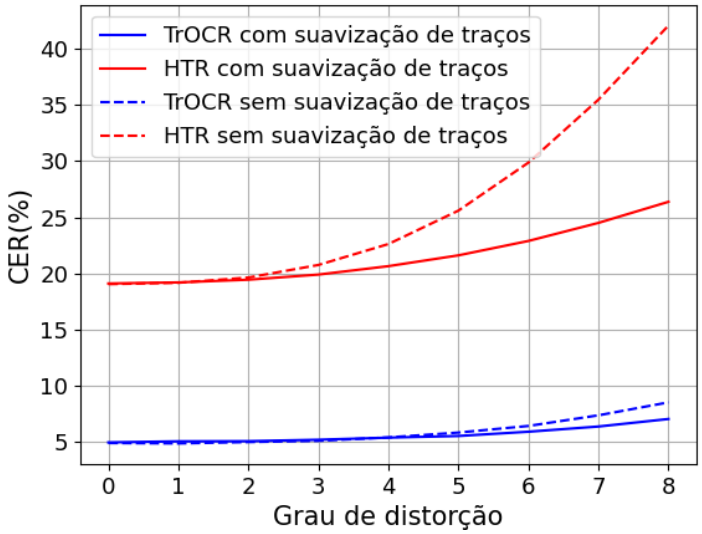


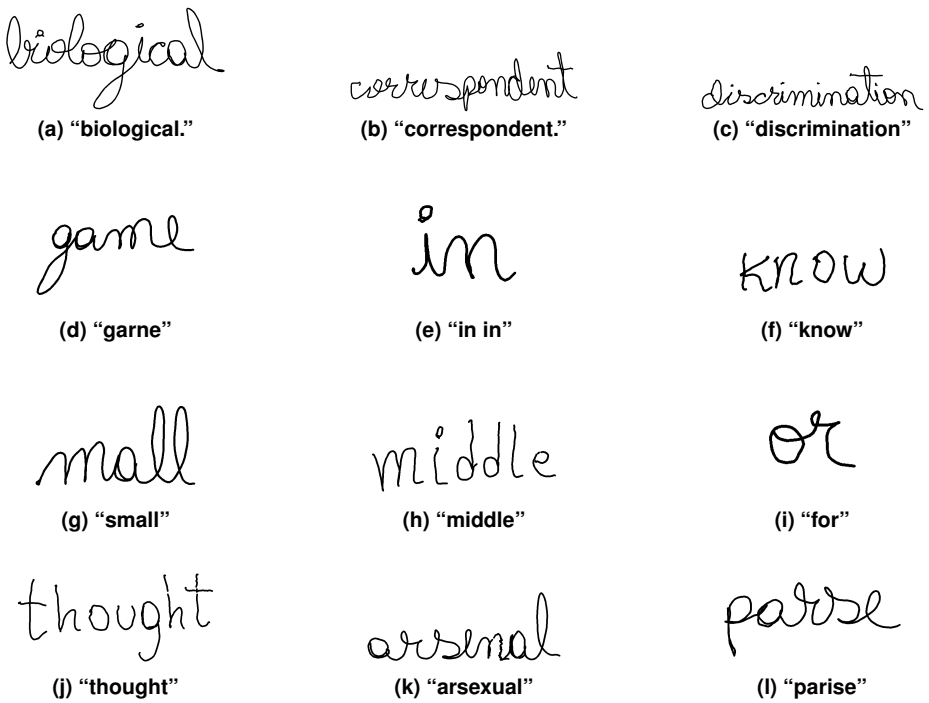
Figura 8. Comparação entre os modelos HTR (em vermelho) e TrOCR (em azul).

Tabela 1. CER (%) - Modelo HTR

Grau de distorção (σ)	0	1	2	3	4	5	6	7	8
Sem suavização	19,04	19,16	19,63	20,75	22,62	25,59	29,87	35,48	42,08
Com suavização	19,1	19,2	19,44	19,89	20,64	21,61	22,89	24,5	26,38

Tabela 2. CER (%) - Modelo TrOCR

Grau de distorção (σ)	0	1	2	3	4	5	6	7	8
Sem suavização	4,88	4,85	4,97	5,08	5,38	5,82	6,41	4,35	8,51
Com suavização	4,94	5,03	5,05	5,18	5,36	5,51	5,89	6,36	7,03



(a) "biological." (b) "correspondent." (c) "discrimination"

(d) "garne" (e) "in in" (f) "know"

(g) "small" (h) "middle" (i) "for"

(j) "thought" (k) "arsexual" (l) "parise"

Figura 9. Reconhecimento textual com TrOCR integrado ao sistema AW.

Análise Qualitativa A Figura 9 exibe 12 amostras geradas com o uso real do sistema AW. O resultado do reconhecimento textual é exibido na legenda de cada amostra. Em geral, o TrOCR apresenta boa capacidade de reconhecimento, tanto para caligrafia cursiva (ex.: (a), (b) e (c)), como para caligrafia não-cursiva (ex.: (f), (h) e (j)). Entretanto, são observados alguns erros: troca de "m" por "rn" (d), "n" por "xu" (k), "rs" por "ris" (l).

5. Conclusão

Este trabalho abordou o problema do reconhecimento textual via OCR para um sistema AW. O desafio do reconhecimento de caracteres manuscritos é intensificado com a escrita livre ao ar, que é um elemento fundamental no nosso caso de estudo. Do ponto de vista do sistema, foram apresentadas as principais funcionalidades, inclusive aquelas relacionadas à renderização do texto na tela. Em relação ao reconhecimento, foram conduzidas análises de caráter quantitativo e qualitativo com o TrOCR e o HTR, OCRs populares de código aberto.

Para a análise quantitativa, realizamos um experimento com a base de dados de texto manuscrito chamada IAM Online Dataset. Para simular o cenário de escrita no ar, foi adicionado ruído em diversos níveis de intensidade. Os resultados obtidos demonstraram a relevância do algoritmo de suavização de traços por interpolação para ambos os OCRs testados. Além disso, o experimento mostrou que o TrOCR levou às menores taxas de erro: CER de 7,03%, considerando o maior nível de ruído. Dado seu melhor desempenho no experimento quantitativo, o TrOCR foi utilizado na análise qualitativa a partir de algumas amostras obtidas diretamente do sistema AW. De um modo geral, o reconhecimento foi satisfatório (CER médio de 15,68%), apresentando alguns erros pontuais.

Para trabalhos futuros, está prevista a adaptação do TrOCR para o contexto AW

por meio de fine-tuning do modelo. Além de possibilitar maior robustez num contexto da escrita no ar, esse procedimento visa estender a aplicabilidade do TrOCR para a língua portuguesa. O sistema pode ser treinado com dados simulados pela adição de ruído e validado tanto em dados simulados, como em uma base real construída a partir de amostras de diversos usuários do sistema AW. Em outra linha de pesquisa, pretende-se integrar o sistema AW com modelos generativos (ex. ChatGPT) para construção de aplicações mais complexas.

Referências

Abir, F. A., Siam, M. A., Sayeed, A., Hasan, M. A. M., and Shin, J. (2021). Deep learning based air-writing recognition with the choice of proper interpolation technique. *Sensors*, 21(24).

Bashir, M., Scharfenberg, G., and Kempf, J. (2011). Person authentication by handwriting in air using a biometric smart pen device. In *BIOSIG 2011 – Proceedings of the Biometrics Special Interest Group*, pages 219–226. Gesellschaft für Informatik e.V., Bonn.

Chen, M., AlRegib, G., and Juang, B.-H. (2016). Air-writing recognition—part i: Modeling and recognition of characters, words, and connecting motions. *IEEE Transactions on Human-Machine Systems*, 46(3):403–413.

Elshenaway, A. R. and Guirguis, S. K. (2021). On-air hand-drawn doodles for iot devices authentication during covid-19. *IEEE Access*, 9:161723–161744.

Itaguchi, Y., Yamada, C., and Fukuzawa, K. (2015). Writing in the air: Contributions of finger movement to cognitive processing. *PLOS ONE*, 10(6):1–17.

Lee, S.-K. and Kim, J.-H. (2021). Air-text: Air-writing and recognition system. In *Proceedings of the 29th ACM International Conference on Multimedia*, MM '21, page 1267–1274, New York, NY, USA. Association for Computing Machinery.

Li, M., Lv, T., Cui, L., Lu, Y., Florencio, D., Zhang, C., Li, Z., and Wei, F. (2021). Trocr: Transformer-based optical character recognition with pre-trained models. `https://github.com/microsoft/unilm/tree/master/trocr`.

Mukherjee, S., Ahmed, S. A., Dogra, D. P., Kar, S., and Roy, P. P. (2019). Fingertip detection and tracking for recognition of air-writing in videos. *Expert Systems with Applications*, 136:217–229.

Vaidya, V., Pravanth, T., and Viji, D. (2022). Air writing recognition application for dyslexic people. In *2022 International Mobile and Embedded Technology Conference (MECON)*, pages 553–558.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

Vloison, V. and Xiwei, H. (2021). Deep learning framework for line-level handwritten text recognition. `https://github.com/vloison/Handwritten_Text_Recognition`.

Zhang, F., Bazarevsky, V., Vakunov, A., Tkachenka, A., Sung, G., Chang, C.-L., and Grundmann, M. (2020). Mediapipe hands: On-device real-time hand tracking.