Comparação de APIs de OCR para Reconhecimento de Dígitos em Imagens de Mostrador de Sete Segmentos

Jonathan R. da Silva[1], Leandro C. Resendo[2], Jefferson O. Andrade[2], Karin S. Komati[2]

[1]Coordenação de Informática

[2]Programa de Pós-graduação em Computação Aplicada (PPComp)
Campus Serra do Instituto Federal do Espírito Santo (IFES)

jota.ribeirosilva@gmail.com, {leandro,jefferson.andrade,kkomati}@ifes.edu.br

Abstract. *Cloud computing platforms make text recognition technology accessible. However, the most suitable solution for a given application is not always evident. This paper evaluated five different text recognition solutions: AWS Rekognition, Microsoft Azure, Cloudmersive, Google OCR, and OCRSpace. A database of images of seven-segment displays in electricity meters, the "YUVA EB Dataset", was used. There was no pre-processing to improve image quality, to improve lighting or to eliminate noise. Google Cloud showed better results, hitting 100 results of the 169 input images, with an accuracy of 86.5 % considering the 965 digits. The results obtained suggest that the use of the solutions offered commercially are not suitable for use in production without a previous stage of pre-processing of the images.*

Resumo. *As plataformas de computação em nuvem tornam acessível a tecnologia de reconhecimento de texto. Entretanto, a escolha da plataforma mais adequada para uma determinada aplicação nem sempre é evidente. Este trabalho avaliou cinco soluções diferentes para reconhecimento de texto: AWS Rekognition, Microsoft Azure, Cloudmersive, Google OCR e OCRSpace. Foi utilizada uma base de dados de imagens de mostradores de sete segmentos em medidores de energia elétrica, a "YUVA EB Dataset". Não houve pré-processamento para a melhoria da qualidade da imagem, para melhoria de iluminação ou para eliminação de ruídos. O Google Cloud apresentou melhores resultados acertando 100 resultados das 169 imagens de entrada, com acurácia de 86,5% considerando os 965 dígitos. Os resultados obtidos sugerem que as soluções oferecidas comercialmente não são adequadas para uso em produção sem uma etapa anterior de pré-processamento das imagens.*

1. Introdução

O mostrador (ou *display*) de sete segmentos, como o seu próprio nome diz, é composto de sete elementos, os quais podem ser ligados ou desligados individualmente de forma que a combinação desses elementos produz representações alfanuméricas. O uso mais comum do mostrador de 7 segmentos é na representação de algarismos arábicos, embora seja possível representar algumas das letras do alfabeto romano. A Figura 1 apresenta uma das representações dos algarismos arábicos (de 0 a 9) usando o mostrador de 7 segmentos.

Variações destas representações são possíveis, tal como a retirada de um dos segmentos para os dígitos 6, 7 e 9.

Muitos são os equipamentos que usam este mostrador, tais como em calculadoras, relógios, medidores de água, medidores de energia, medidores de pressão, medidores de temperatura, etc. Vários desses equipamentos de medição que estão em atividade em campo são antigos, dado que o mostrador de 7 segmentos teve sua primeira patente em 1903[1]. Apesar de antigos, muitos medidores funcionam perfeitamente, mas não possuem forma de comunicação para transferir os dados medidos para um sistema de informação. Como em muitos casos, o custo para a troca de um grande parque instalado é alto, ou exige parada de processos de produção e/ou fornecimento de serviço, prefere-se usar soluções computacionais para o reconhecimento automático destes dígitos [Finnegan et al. 2019] [Bonačić et al. 2015].

Figura 1. Algarismos arábicos representados em 7 segmentos.

Destaca-se o trabalho de [Kanagarathinam and Sekar 2019] que propõe uma solução aplicada em medidores de consumo de energia elétrica. Em muitos lugares, a leitura é feito por um ser humano que depois envia/cadastra o dado para um sistema de cálculo de fatura. Inicialmente, as imagens passam por operações de processamento de imagens, pelos algoritmos MSER (*Maximally Stable Extremal Regions*) e posteriormente pela operação de dilatação de imagem. Estas imagens processadas é que são avaliadas pelo OCR (do inglês *Optical Character Recognition*) do MATLAB, que reconhece caracteres em uma imagem, convertendo as imagens dos caracteres em texto. Os resultados obtidos são bem satisfatórios, alcançando uma taxa de acurácia superior a 90% para maioria dos dígitos. Além disso, o trabalho disponibiliza a base de dados “YUVA EB Dataset”, que contém uma coleção de imagens de medidores digitais de energia. As imagens foram capturadas em condições de luz diurna e noturna, inclinadas e com baixa resolução.

Nos últimos tempos a tecnologia de reconhecimento ótico de caracteres (OCR) evoluiu bastante e se tornou mais conhecida devido o crescimento das plataformas de serviço em nuvem. Existem várias ferramentas OCR disponíveis no mercado capazes de realizar o reconhecimento de caracteres, criando opções para solucionar problemas de transferência de dados de medidores digitais. A questão de pesquisa é: “será que as ferramentas de OCR de mercado teriam um resultado melhor que o trabalho [Kanagarathinam and Sekar 2019], usando a mesma base de dados?”

A proposta deste trabalho é comparar ferramentas de OCR usando a base de dados “YUVA EB Dataset” sem a etapa de pré-processamento das imagens. A escolha das ferramentas foi guiada por APIs[2] mais populares e que fossem de uso gratuito. Assim, foram selecionadas 5 APIs: (i) AWS Rekognition[3], (ii) Microsoft Azure[4], (iii) Cloudmersive[5],

---

[1] https://patents.google.com/patent/US1126641

[2] API é um conjunto de rotinas e padrões de programação para acesso a um aplicativo de software ou plataforma baseado na Web. A sigla API refere-se ao termo em inglês “Application Programming Interface” que significa em tradução para o português “Interface de Programação de Aplicativos”.

[3] https://docs.aws.amazon.com/rekognition/latest/dg/text-detection.html

[4] https://docs.microsoft.com/en-us/azure/cognitive-services/computer-vision/concept-recognizing-text

[5] https://cloudmersive.com/ocr-api

(iv) Google OCR[6] e (v) OCRSpace[7].

Para cada imagem da base de dados, serão coletados os resultados de cada API selecionada. As análises e comparações serão feitas levando em consideração as características da base de dados, como condição de captura das imagens e por dígito. Além disso, será feita a comparação com o trabalho base [Kanagarathinam and Sekar 2019].

A Seção 2 descreve alguns trabalho correlatos, a Seção 3 detalha a base de dados e o ambiente de experimentos, a Seção 4 apresenta os resultados e discussão e a Seção 5 fecha com as considerações finais.

2. Trabalhos Correlatos

Nesta seção serão descritos dois artigos que comparam o uso de ferramentas em nuvem. O trabalho [Anda et al. 2018] comparou quatro API: Kairos, AWS, DEX e Azure, para predição de idade aparente. A base de dados de 10.140 imagens de faces de pessoas de idade da faixa de 0–77 anos. A análise foi feita via métrica de erro médio absoluto (MAE), e a melhor ferramenta foi o Azure. O trabalho também mostrou que as ferramentas erram mais no gênero feminino do que no masculino. E que as ferramentas são melhores entre 10–60 anos.

No estudo de [Torres et al. 2020], foi investigada a adequação do reconhecimento óptico de caracteres (OCR) para extrair texto de modelos gráficos, tais como diagramas de classe UML, casos de uso em UML, diagramas de sequência em UML e diagramas feitos em Matlab Simulink. A comparação foi entre o Google Cloud e o Microsoft Cognitive Services. O Google Cloud teve um desempenho melhor do que o Microsoft Cognitive Services, sendo capaz de detectar texto de 70% dos elementos do modelo. Os erros cometidos pelo Google Cloud Vision são devidos à ausência de suporte para texto comum em fórmulas de engenharia, por exemplo, letras gregas, equações e subscritos, bem como texto composto em várias linhas.

3. Materiais e Métodos

Nesta seção serão detalhadas as características da base de imagens "YUVA EB Dataset", bem como o desenvolvimento do ambiente de testes usando as ferramentas de OCR.

3.1. Base de imagens "YUVA EB Dataset"

O conjunto de dados utilizado neste trabalho, de nome "YUVA EB Dataset"[8], é o mesmo utilizado em [Kanagarathinam and Sekar 2019]. Esta base de dados consiste em uma coleção de imagens de telas de medidores digitais de energia de sete segmentos, coletadas na região de Tamil Nadu na Índia. As imagens presentes nessa base de dados estão todas no formato JPEG.

As imagens estão divididas em subconjuntos de acordo com as condições de captura, diurna, noturna, inclinada e baixa resolução. A base de dados tem um total de 169 imagens, divididas da seguinte forma: base *Day Time* com 50 imagens capturadas à luz

[6]https://cloud.google.com/functions/docs/tutorials/ocr

[7]https://ocr.space/

[8]Disponível em https://drive.google.com/drive/folders/1J9TYUiLKdJKfSeotL-_EIyvXQ-3pE282

do dia, base *Night time* 49 noturnas, base *Tilted* com 50 imagens inclinadas, e base *Blurred* com 20 imagens desfocadas, alguns exemplares são mostradas na Figura 2, Figura 3, Figura 4 e Figura 5, respectivamente. As imagens noturnas, diurnas e inclinadas foram capturadas usando uma câmera digital no modo de alta resolução e as imagens de baixa resolução foram capturadas com a mesma câmera, mas no modo de baixa resolução.

Figura 2. Exemplos de imagens capturadas à luz do dia.

Figura 3. Exemplos de imagens capturadas à noite.

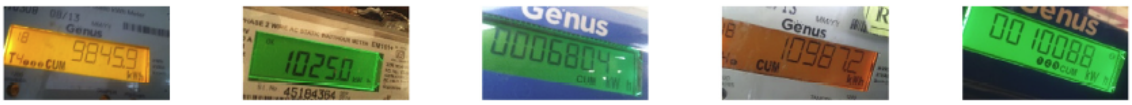

Figura 4. Exemplos de imagens inclinadas.

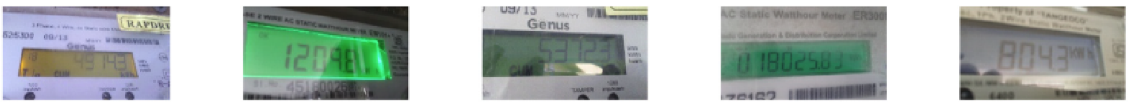

Figura 5. Exemplos de imagens de baixa resolução.

Pelas imagens, é possível verificar que a cor do fundo varia entre várias cores: verde, laranja, lilás, azul, em diferentes tons. Todas as imagens dos medidores, além dos dígitos da medida, contém outras informações/caracteres em torno do *display* (tais como nome do fabricante e código do produto), e mesmo dentro do *display* (tais como "kWh" e a palavra "CUM"). Todos esses caracteres que não são os dígitos da leitura do consumo de energia são desconsiderados para a avaliação de reconhecimento de caracteres. A quantidade de dígitos varia por imagem. Há uma predominância de telas que apresentam 5 dígitos sendo 74 amostras de imagens, e apenas 8 amostras de imagens com 8 dígitos. Não há imagens com menos de 4 dígitos, nem com mais de 8 dígitos.

3.2. Ambiente de experimentos

Para o processamento das imagens foi desenvolvido um projeto em .NET Core do tipo "Console Application", o projeto se encontra no GitHub[9]. Na Figura 6 é possível ver o diagrama de classes da aplicação, mostrando as relações entre as classes. Basicamente a aplicação consiste em uma classe **Program** que chama cada serviço OCR.

[9]https://github.com/JonathanRibeiro92/OCRComparer.git

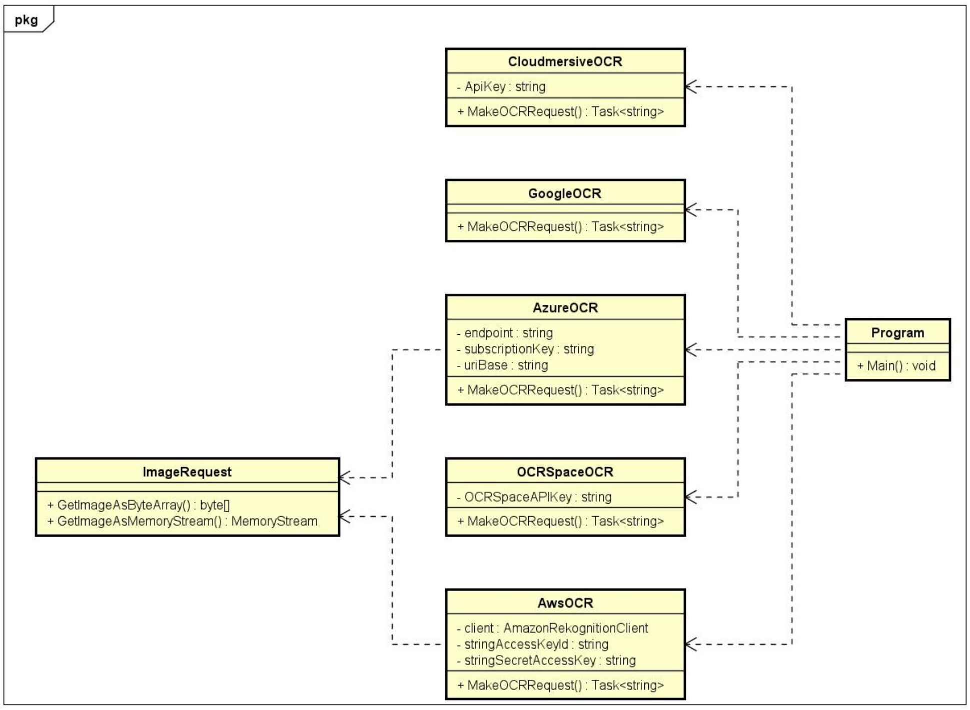

Figura 6. Diagrama de Classes da Aplicação de Testes.

Para cada arquivo de imagem processado é criado um caminho de arquivo de resultado por serviço consultado, isto é, uma pasta específica para cada API, seguindo o mesmo padrão de nomenclatura do caminho da base de dados, uma pasta para cada condição de captura de imagem. A aplicação realiza as chamadas às APIs que fornecem o serviço OCR e armazena os resultados em formato JSON, e depois armazena os arquivos de resultado utilizando os caminhos de pasta de resultados montados anteriormente.

Tabela 1. Relação dos Serviços OCR escolhidos

Serviço OCR	Formatos aceitos	Suporte às tecnologias	Custo
AWS	Arquivos PDF, JPEG, PNG. Vídeos armazenados no próprio serviço	REST, JavaScript, Python, PHP, .Net, Ruby, Java, Go, Node.js, C++	Nível Gratuito 12 meses 5.000 imagens por mês
Azure	PDF, TIFF, JPEG, PNG, BMP	REST, .NET, Python, Java, Node.js, Go	Gratuito até 5.000 transações por mês
Cloudmersive	PNG, JPEG	REST, .Net, Java, Node.js, Python, PHP, Objective-C, Ruby, Zapier	Gratuito até 800 transações por mês
Google Cloud	Documentos PDF e TIFF. Imagens codificadas em base64.	REST, C#, GO, Java, Node.js, PHP, Python, Ruby	Gratuito para primeiras 1.000 imagens por mês
OCRSpace	PDF, JPG, GIF, PNG	REST, C#, C++, Java, Javascript, Node.js, PHP, Python, Ruby, Swift, Objective-C	Gratuito para até 25.000 requisições no mês, com restrição de 500 por dia

Para facilitar a comparação das cinco ferramentas selecionadas para este trabalho, um resumo das características é apresentada na Tabela 1. A primeira coluna da tabela tem o nome do serviço/ferramenta, seguido pelos formatos aceitos, quais as tecnologias suportadas e o custo. Todas as ferramentas são gratuitas, mas cada uma apresenta uma limitação diferenciada, determinando restrições de uso.

4. Resultados e Discussão

A análise dos experimentos será feita tal como no trabalho de [Kanagarathinam and Sekar 2019], em que o erro é qualquer dígito não identificado, ou identificado erroneamente. Por exemplo, para um *display* com o número "03525.9", foram considerados corretos os resultados:"03525.9" ou "035259" ou "035259*" ou "035259kW" ou "035:259" ou "035 259" ou "03525 9kW". Isto é, o '.' (ponto decimal) pode ou não ser reconhecido, mas ainda conta-se como correto; a inclusão de um ou mais caracteres (podendo ser um espaço em branco ou caracteres especiais) no meio ou após a sequencia de dígitos, também foi considerado correto. Foi considerado incorreto "35259", pois não reconheceu o '0' (zero) à esquerda. O desempenho geral das ferramentas testadas é apresentado na Figura 7. A ordem de apresentação das ferramentas foi a mesma ordem dos experimentos realizados.

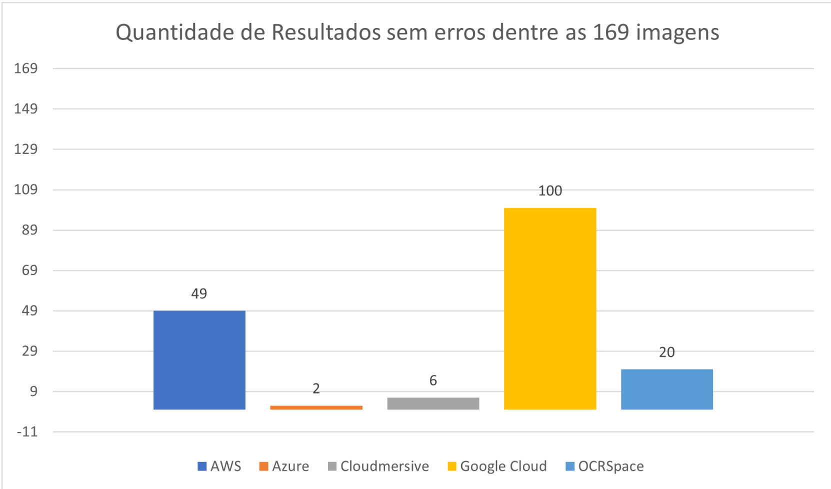

Figura 7. Gráfico de resultados sem erros por ferramenta.

Fazendo uma análise geral das ferramentas, comparando os resultados sem erros, o melhor resultado foi do Google Cloud com 100 acertos dentre as 169 imagens da base de dados (59,17% de acerto), o segundo foi a AWS com 49 imagens que tiveram todos os dígitos identificados corretamente (28,99%), OCRSpace com 11,83% de acertos (20 das 169 imagens), o Cloudmersive com 6 reconhecimentos corretos (3,55% de acerto) e a AZURE com apenas 2 imagens (1,18% de acerto).

4.1. Comparação por Dígito

Nesta seção a análise será contada de forma diferente da comparação apresentada na Figura 7. Nesta análise, serão contabilizados os acertos de cada dígito reconhecido, assim, se na análise anterior a população total era de 169 imagens, nesta seção, a população total é de 985 dígitos. Assim, para o número "035259", se a resposta for "085259", a contagem é de 1 (um) erro no dígito '3' (houve a troca do '3' pelo '8'), mas conta-se um acerto para os dígitos '0', '2' e '9' e dois acertos para o dígito '5'. O resultado, tanto por dígito e por ferramenta de OCR, é apresentado na Tabela 2, valores maiores que 75% foram marcados em negrito.

De novo, Google Cloud teve um melhor desempenho apresentando uma quantidade maior de acertos que as demais ferramentas, tanto no geral (acurácia de 86,5%, acertando 852 dígitos dos 985), quanto por dígito (o percentual de acerto foi maior que 75% em todos os dígitos). O Google Cloud erra mais o dígito '1' que os demais, pois muitas das vezes responde como o dígito ’7’, e acerta mais o dígito ’5’. Em segundo lugar, segue o AWS (68,2% de taxa de acerto no geral) e pelo OCRSpace (42,84%). O pior resultado foi da Azure que apresentou um número de acertos consideravelmente baixo (apenas 50 dos 985, taxa de acerto de 5,08%), seguido do Cloudmersive (234 dos 965, taxa de acerto de 23,76%).

Tabela 2. Performance das ferramentas em relação aos dígitos por serviço

Dígitos	Total Dígitos	Serviço									
		AWS		AZURE		Cloudmersive		Google Cloud		OCRSpace	
		Acertos	%acerto	Acertos	%acerto	Acertos	%acerto	Acertos	%acerto	Acertos	%acerto
0	243	134	55,14%	8	3,29%	52	21,40%	209	**86,01%**	81	33,33%
1	108	57	52,78%	8	7,41%	26	24,07%	83	**76,85%**	26	24,07%
2	101	85	**84,16%**	2	1,98%	30	29,70%	91	**90,10%**	49	48,51%
3	113	81	71,68%	5	4,42%	19	16,81%	103	**91,15%**	58	51,33%
4	85	70	**82,35%**	4	4,71%	18	21,18%	75	**88,24%**	39	45,88%
5	69	57	**82,61%**	10	14,49%	30	43,48%	63	**91,30%**	41	59,42%
6	53	29	54,72%	2	3,77%	18	33,96%	46	**86,79%**	30	56,60%
7	68	42	61,76%	1	1,47%	19	27,94%	54	**79,41%**	27	39,71%
8	62	47	**75,81%**	4	6,45%	15	24,19%	56	**90,32%**	23	37,10%
9	83	70	**84,34%**	6	7,23%	7	8,43%	72	**86,75%**	48	57,83%
Total	985	672	68,22%	50	5,08%	234	23,76%	852	**86,50%**	422	42,84%

4.2. Comparação por Condição de Captura

Esta análise é feita levando em consideração as condições em que a imagem foi capturada: Luz do Dia, Noturna, Inclinada ou Baixa Resolução. A Tabela 3 apresenta um comparativo do percentual de acertos de cada uma das ferramentas, valores acima dos 75% foram marcados em negrito.

Tabela 3. Percentual de acertos por tipo de condição na captura da imagem, considerando os 985 dígitos

Serviço	Luz do Dia (%)	Noturna (%)	Inclinada (%)	Baixa Resolução (%)
AWS	**80,65**	**83,00**	39,86	71,30
AZURE	5,38	5,67	4,12	5,22
Cloudmersive	29,39	27,33	18,90	13,04
Google Cloud	**87,81**	**84,00**	**91,41**	**77,39**
OCRSpace	43,01	51,33	42,96	20,00

Os dados confirmam que o Google Cloud é melhor em todas as condições, e apresentam os melhores resultados em imagens inclinadas. Na base Noturna, a diferença com relação ao AWS é bem pequena equivalente à uma única imagem. O AWS tem bons resultados nas bases de Luz do Dia e Noturna, mas tem baixa taxa de acerto para a base Inclinada e na de Baixa Resolução. De uma forma diferente, o AWS reconhece mais corretamente as imagens noturnas do que as diurnas, supõe-se que essa ferramenta lide melhor com imagens com menos incidência de luz. Apesar da primeira impressão ser de que as imagens de Baixa Resolução teriam um resultado pior em todas as ferramentas, percebe-se que isso não é verdadeiro para o AWS.

5. Considerações Finais

Neste trabalho foram investigadas as ferramentas da AWS Rekognition, Microsoft Azure, Cloudmersive, Google OCR e OCRSpace. O reconhecimento de caracteres em imagens é muito útil em diversos cenários. Um exemplo, abordado neste trabalho, se dá quando o parque instalado possui muitos equipamentos antigos e o custo para substituir por equipamentos mais modernos, com interface de comunicação, torna a troca de equipamentos praticamente inviável. A ferramenta que possui um desempenho melhor dentre as cinco avaliadas é a da Google Cloud, o Google OCR, com taxa de acerto geral de 86,5% considerando cada dígito ou 59,17% considerando a imagem como um todo. O pior resultado foi da ferramenta da AZURE com taxa de acerto de apenas 5,08% considerando cada dígito ou 1,18% considerando a imagem como um todo (acertou o display completo de apenas duas imagens da base de dados).

Considerando o percentual de acurácia (percentual de acerto), nenhuma das ferramentas testadas aqui apresentou um resultado melhor que os apresentados pelo trabalho de [Kanagarathinam and Sekar 2019] (93,17%) por dígito, o mais próximo é o Google Cloud com 86,50% de acurácia. É importante considerar que neste trabalho não foi feito nenhum tipo de pré-processamento, nem foi realizado treinamento das ferramentas usando as imagens da base de dados. Com isso, considera-se que utilizar as ferramentas de reconhecimento de imagens sem um pré-processamento pode não ser uma boa opção quando é necessário realizar análises mais assertivas.

Embora os objetivos propostos no trabalho tenham sido alcançados, algumas melhorias são possíveis visando trabalhos futuros: realizar a análise utilizando uma base com o número maior de imagens, realizar pré-processamento das imagens (melhorando ruído e normalizando a iluminação), desenvolver um APP para dispositivos que possuam interface que facilite o usuário a delimitar a área do *display*, estudar as variações dos parâmetros das configurações dos OCR testados, aprofundar as discussões de taxa de acerto levando em consideração outras métricas, e realizar experimentos com outros OCR de mercado.

6. Agradecimentos

Agradecemos à FAPES (Fundação de Amparo à Pesquisa e Inovação do Espírito Santo) e a CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) pelo apoio financeiro dado por meio do PDPG (Parcerias Estratégicas nos Estados da CAPES).

Referências

Anda, F., Lillis, D., Le-Khac, N., and Scanlon, M. (2018). Evaluating automated facial age estimation techniques for digital forensics. In *Proceedings of 2018 IEEE Security and Privacy Workshops (SPW)*, pages 129–139. IEEE.

Bonačić, I., Herman, T., Krznar, T., Mangić, E., Molnar, G., and Čupić, M. (2015). Optical character recognition of seven-segment display digits using neural networks. In *32st International Convention on Information and Communication Technology, Electronics and Microelectronics*, volume 3.

Finnegan, E., Villarroel, M., Velardo, C., and Tarassenko, L. (2019). Automated method for detecting and reading seven-segment digits from images of blood glucose metres and blood pressure monitors. *Journal of Medical Engineering & Technology*, 43(6):341–355.

Kanagarathinam, K. and Sekar, K. (2019). Text detection and recognition in raw image dataset of seven segment digital energy meter display. *Energy Reports*, 5:842–852.

Torres, W., van den Brand, M. G., and Serebrenik, A. (2020). Suitability of optical character recognition (ocr) for multi-domain model management. In *International Conference on Systems Modelling and Management*, pages 149–162. Springer.