

Reconhecimento de Textos e sua Sintetização em Linguagem Natural Humana

Leonardo Yuji Fuiguti¹, Pedro Henrique Fuzario Custódio¹, Maxmillian J. de Melo¹,
Guilherme F. Terenciani¹

¹Instituto Federal de Ciência e Tecnologia de Mato Grosso do Sul (IFMS)
Rua Hilda 203 - Bairro Boa Vista - Naviraí - MS

leonardo.fuiguti, pedro.custodio {@estudante.ifms.edu.br}

guilherme.terenciani, maximilian.melo {@ifms.edu.br}

Abstract. *This research proposes the development of a computational tool in the form of a product dedicated to the support of visually impaired people or illiterate people. The tool aims at the identification of texts and their synthesis in audio format, thus facilitating and encouraging, the habit of reading for this profile of people. Given the problem dimension, a character recognizer is displayed in images using the computational tool textit Tesseract and the speech synthesizer textit Espeak. In the development of the software will be used the programming language textit Python, which will integrate several tools of computational vision, textit tesseract and speech synthesize.*

Resumo. *Este trabalho propõe o desenvolvimento de uma ferramenta computacional nos moldes de um produto dedicado ao apoio de pessoas com deficiência visual ou pessoas analfabetas. A ferramenta objetiva a identificação de textos e sua sintetização em formato de áudio, facilitando e incentivando assim, o hábito da leitura para esse perfil de pessoas. Visto a dimensão do problema, é apresentado um reconhecedor de caracteres em imagens utilizando a ferramenta computacional Tesseract e o sintetizador de voz Espeak. No desenvolvimento do software será utilizada a linguagem de programação Python, a qual irá integrar diversas ferramentas de visão computacional, tesseract e sintetização de voz.*

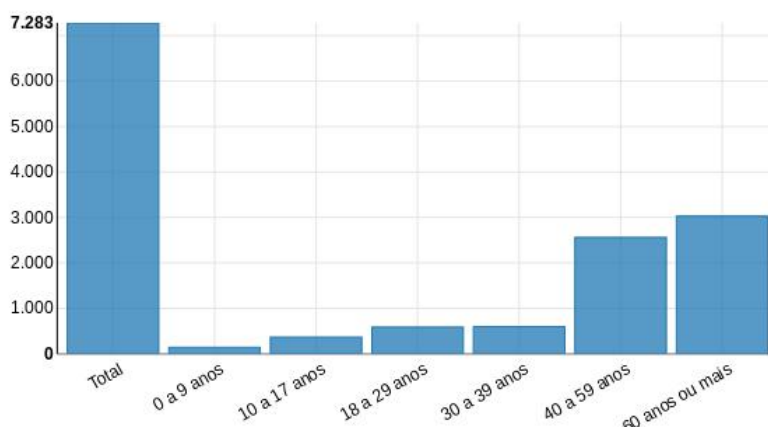
1. Introdução

No Brasil existem, segundo dados do IBGE 2010, aproximadamente 18,6% da população brasileira de pessoas com algum grau de deficiência visual sendo 528.624 pessoas que não tem 100% da visão¹. No gráfico da Figura 1 podemos observar que a cada um milhão de pessoas, 7.283 mil tem algum tipo de deficiência visual.

Em evidência desses fatos, grande parte dessas pessoas tem certa dificuldade em realizar muitas das atividades cotidianas. Tarefas como: navegar pela internet, interagir com o celular, procurar itens de vendas em supermercados, ler placas de anúncios pela cidade, leitura de livros, entre outras. Atividades que para a maioria das pessoas seriam consideradas simples, para deficientes visuais acabam se tornando uma tarefa complicada.

¹<http://www.pessoacomdeficiencia.gov.br/app/sites/default/files/publicacoes/cartilha-censo-2010-pessoas-com-deficiencia-reduzido.pdf>

Figura 1. Total de pessoas com deficiência visual por milhão de pessoas



Fonte: IBGE - Pesquisa Nacional de Saúde - 2013

Deficiência visual é definida com sendo a perda parcial ou total da visão, que não pode ser corrigida com uso de lentes ou cirurgias. Doenças como miopia, hipermetropia e astigmatismo não pode ser considerado como uma deficiência visual, pois elas podem ser ajustadas com instrumentos, como óculos, e cirurgias. Segundo a OMS existem diferentes graus de deficiência visual que são classificados em:

- Baixa visão: Que pode ser compensada com uso de lentes, lupas ou aparelhos auxiliares como bengala;
- Próximo à cegueira: Quando a visão tem uma alta debilitação, porém ainda existe uma certa percepção de luz, ainda que ínfimo;
- Cegueira: Quando percepção de luz torna-se nula, necessitando assim do uso de aparelhos como bengala, *braille* ou orientadores.

Ainda vale ressaltar que no Brasil, segundo dados da IBGE do censo de 2016, existem 11,8 milhões de analfabetos. Esse número representa 7,2% da população brasileira. Essas pessoas analfabetas, por sua vez, apresentam uma dificuldade para exercer grande parte das práticas cotidianas, uma vez que muitas atividades tem algum envolvimento com a leitura.

Estes são alguns dos problemas que vem trazendo um grande prejuízo dentro de instituições acadêmicas, mercado de trabalho, no âmbito doméstico, entre outros. Durante a realização de uma prova, por exemplo, uma pessoa com deficiência visual precisaria de um auxiliar para que ele possa responder as questões de sua prova. Além disso não é grande o número de livros disponíveis em *braille* (escritas com pontos em relevos).

Nos dois problemas citados acima, a falta da capacidade de leitura, tanto pelo fato de ter uma deficiência visual, quanto por algum grau de analfabetismo, tem levado as pessoas a serem excluídas da sociedade. Deixando assim de garantir a plena execução da constituição federal no seu Art. 3º² onde devemos diminuir a desigualdade social dessas pessoas marginalizadas pela falta da capacidade de leitura.

Com o avanço da computação, muito tem se falado de desenvolver técnicas capazes de facilitar a vida de pessoas por meio da tecnologia. Existem muitas pes-

²<https://www.senado.leg.br/atividade/const/con1988/con1988atual/art3.asp>

quisas no ramo da computação envolvendo trabalhos que possam melhorar a qualidade de vida da população [Bruckschen et al. 2007] [Ossada and Rodrigues 2015] [Rigolon and Almeida 2016]. Visando tal avanço e buscando maneiras de melhorar a qualidade de vida de pessoas com deficiência visual.

Atualmente, existem projetos que auxiliam pessoas com deficiências, como: o relógio inteligente em *braille*³ que utiliza ímãs e um conjunto de pinos para criar caracteres em *braille*, a bengala sensorial de [Rigolon and Almeida 2016] acessório que detecta o ambiente por meio de sensores que emitem sinais transformados em vibração, e o *bookreader*, dispositivo que identifica e converte o texto para a pronúncia, tem a mesma estética de uma impressora, possuindo objetivos semelhantes ao deste projeto.

Resolver os problemas de detecção de caracteres em textos é uma tarefa com alta complexidade. A tarefa de detectar caracteres em textos digitalizados ou textos contidos em fotografias é uma tarefa árdua. Muito se tem estudado sobre formas de conseguir identificar de forma correta caracteres sobre texto. Na literatura [Agarwal and Kaushik 2015] [Chen et al. 2004] [Coates et al. 2011] [Neumann and Matas 2010] [Neumann and Matas 2012], utilizam técnicas de inteligência artificial, mais especificamente, na área de visão computacional para resolver tal problema. Agarwal e Kaushik (2015), em seu trabalho, utilizam redes neurais artificiais (*Artificial Neural Network*) e algoritmos genéticos (*Genetic Algorithm*) para extrair de forma eficientes caracteres e textos contidos em arquivos digitalizados de hospitais. No trabalho, os resultados quanto à extração de textos em imagens têm se mostrado promissores.

Em [Antonello 2017], os autores realizam a automatização de uma cancela, regulando a entrada e saída de veículos em uma instituição. No trabalho, os autores utilizaram de amplo arsenal de visão computacional e inteligência artificial para o cumprimento dos objetivos propostos, dentre os quais destacam-se o reconhecimento e segmentação da placa e reconhecimento dos caracteres pertencentes à placa. Para tanto, ferramentas para o auxílio de *OCR* (Reconhecimento óptico de caracteres) como a *Tesseract* foram empregadas, garantindo uma taxa de acerto de 93,54% nos testes com um conjunto de testes contendo 10 placas cadastradas.

Em ([Hazra et al. 2017]; [Chang et al. 2016]; [Kang et al. 2014]), os autores utilizaram investigaram diversas abordagens para realizar *OCR* com as mais diversas finalidades, como por exemplo reconhecimento de *handwritten text* (texto escrito à mão) e equações. Foram usadas estratégias que envolveram o uso de redes neurais artificiais diversas, como KNN (*K-Nearest Neighbor*) e CNN (*Convolutional Neural Network*), além de SVM (*Support Vector Machine*), com taxas de acerto superiores a 90%.

O objetivo deste projeto é desenvolver um *software* de apoio para automatizar a leitura de textos, de modo a contribuir na melhoria de qualidade de vida de pessoa com deficiência visual e/ou com analfabetismo. Para tanto, partindo de uma imagem que contenha texto, por exemplo um livro, através de um código de reconhecimento e identificação de textos, o projeto sintetiza o texto reconhecido em formato de áudio em linguagem natural humana. Ademais, também promove a inserção dessas minorias representativas na sociedade, por conta das possibilidades entregues pelo projeto, possibilitando-os realizar

³<https://olhardigital.com.br/noticia/veja-como-funciona-um-relogio-inteligente-em-braille/67013>

a leitura de algum texto, na qual representa uma das atividades mais importantes no dia a dia.

O restante do documento está organizado como segue. A seção 2 aborda os conceitos necessários para a compreensão do trabalho. A seção 3 explica como segue cada procedimento para o desenvolvimento do projeto. A seção 4 apresenta os resultados obtidos durante o desenvolvimento do projeto. A seção 5 realiza um fechamento do projeto discutindo os aspectos mais relevantes do projeto.

2. Fundamentação teórica

2.1. Processamento digital de imagens

Para realizar o reconhecimento de caracteres é preciso que haja uma nitidez na imagem para que se possa alcançar tal objetivo. Portanto, para se ter uma imagem de qualidade quanto a sua clareza, sem elementos desnecessário como ruídos ou *RGB*, deve-se realizar o processamento digital da imagem, uma vez que este é um método eficiente para tal processo [Campos et al. 2005].

O interesse nos métodos de processamento digital de imagens provém de duas áreas principais de aplicação: melhora das informações visuais para a interpretação humana e processamento de dados de imagens para armazenamento, transmissão e representação, considerando a percepção automática por máquinas. Este capítulo tem vários objetivos: (1) definir o escopo da área que chamamos de processamento de imagens; (2) apresentar uma perspectiva histórica das origens dessa área; (3) dar uma ideia do que há de mais avançado em processamento de imagens analisando algumas das principais áreas nas quais ele é aplicado analisar brevemente as principais abordagens utilizadas no processamento digital de imagens;

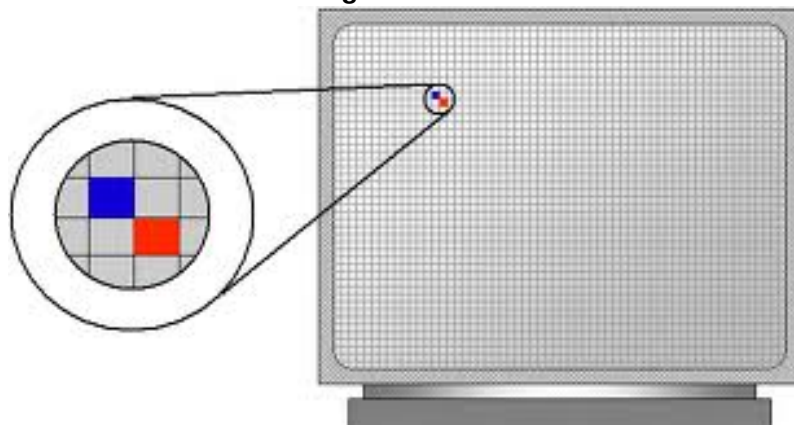
(5) mostrar uma visão geral dos componentes contidos em um sistema de processamento de imagens típico; e (6) oferecer um referencial bibliográfico e outras fontes nas quais o trabalho relativo ao processamento de imagens normalmente é relatado. [Gonzalez and Woods 2000].

Segundo o livro o processamento digital de imagens de Gonzales e Woods, 2000, refere-se ao processamento de imagens digitais em uma máquina digital. Uma imagem digital é formada por conjunto de *pixel*, na qual este é composto por coordenada espacial (x, y), valor específico de intensidade das matizes **RGB**, entre outros.

Mais especificamente, um *pixel* é o menor ponto no qual se forma uma imagem digital, ao qual se atribui uma cor, sendo assim, um conjunto de muitos *pixels* origina uma imagem completa. Cada um é composto por 3 pontos: vermelho, verde e azul (*RGB*), e cada ponto pode ser capaz de apresentar 256 tonalidades de cores.

Um pixel não apresenta as mesmas propriedades em todas as direções, as distâncias entre um ponto e seus vizinhos não é a mesma para todos os tipo de vizinho. Esta propriedade faz com que um pixel tenha quatro vizinhos nas bordas e quatro vizinhos na diagonal. Essa propriedade faz com que seja definido o tipo de conectividade que será utilizada, B4 (considerando os vizinhos de borda) ou B8 (considerando os vizinhos de borda e os de diagonal) [Esquef et al. 2003].

Figura 2. Pixel



Fonte: ELGScreen ⁴

2.2. Tesseract

O *tesseract OCR* (*Optical Characters Recognition*) é uma biblioteca, desenvolvida pela *Google*, utilizada para realizar o reconhecimento de caracteres. A sua precisão quanto aos acertos de caracteres irá ser fortemente influenciada pela etapa de pré-processamento da imagem e segmentação de textos.

O *tesseract* funciona da seguinte forma: Primeiro é feito o limiar adaptativo, na qual realiza a conversão da imagem para uma imagem binária. Em seguida é feita a análise dos componentes conectados com o intuito de extrair o contorno dos caracteres. A próxima etapa é a conversão dos contornos em *Blobs* (região isolada e pequena da imagem digitalizada delimitada por um contorno). As regiões conectadas, são separadas pela lógica *fuzzy*. Posteriormente, é feita o reconhecimento de texto. Ele reconhece cada palavra utilizando um classificador adaptativo. Como ele é um classificador adaptativo, quando encontra uma nova palavra ele se realimenta para tornar possível o seu reconhecimento.

2.3. Extração de características

Quando se fala em reconhecimento de padrões é fundamental desenvolver um método de extração de característica eficiente. Quando se obtém um extrator de características eficientes, temos a premissa de que iremos conseguir um classificador eficiente. Uma das principais características de uma imagem é a sua textura [Souza and Pistori 2005].

A característica da textura de uma imagem é observada na distribuição de *pixels* de determinada região com seu brilho cor e tamanho [Costa et al. 2013]. Estes são alguns fatores que designam certo objeto contido em uma imagem. Quando se encontra algum grau de padrão entre duas ou mais texturas pode-se classificar como sendo um objeto de uma mesma classe.

Uma textura, quando analisado visualmente, pode-se qualificar em fina, grossa, granulada, lisa, entre outras. Essas são uma das características que tornam possível a

⁴<https://elgscreen.zendesk.com/hc/pt-br/articles/201863184-PIXEL-O-que-%C3%A9-um-pixel-E-um-dead-pixel-pixel-morto->

padronização do objeto, lembrando que quanto mais características forem extraídas mais preciso será o resultado.

Outra forma que um objeto pode ser categorizado é a morfologia do objeto. Um objeto pode apresentar a forma cúbica, esférica ou até mesmo uma forma distorcida. Na identificação de uma maçã e banana, por exemplo, pode ser analisada pelas suas respectivas formas, onde a banana tem a forma de lua crescente e a maçã uma forma esférica. Já quando comparar uma maçã e uma laranja, não poderá ser classificadas pelas suas formas, já que apresentam uma morfologia semelhante. Nesse caso, poderão ser comparadas pela cor e pela textura.

3. Materiais e métodos

O livro [GONZALEZ; WOODS, 2000] é utilizado no projeto com a intenção de aprendizagem sobre como remover ruídos que podem se tornar obstáculos quando se trata de reconhecimento de caracteres.

Para a execução do projeto, é utilizada a linguagem de programação *Python*. Embora a linguagem ainda se mostre ser menos eficiente em tempo de execução em comparação às outras linguagens, como a linguagem C, *Python* conquistou um grande espaço no que se diz respeito à inteligência artificial e Processamento Digital da Imagem, visto o amplo ferramental nativo e diversas outras bibliotecas abertas, dedicadas à área. O grande suporte, com uma sintaxe clara e simples, além de bibliotecas de classificação de imagens recorrentemente usadas na literatura, são os pontos mais promissores e que mais contribuíram na escolha da linguagem.

Primeiramente é feita uma interface gráfica com o uso da biblioteca *QT*. A interface contém uma logo original, o título do Trabalho de Conclusão de Curso e um botão que inicia a execução do *software*.

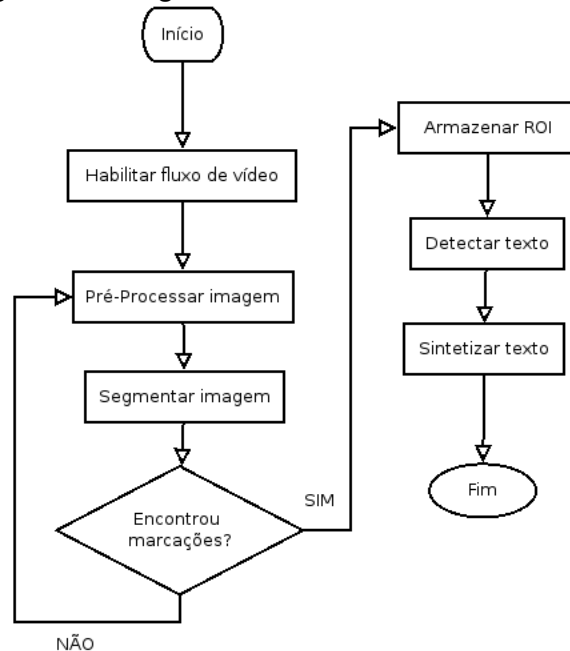
Além da escolha da linguagem *Python*, este trabalho também utiliza a biblioteca *OpenCV* (*Open Source Computer Vision Library*). A biblioteca *OpenCV* traz inúmeras ferramentas de visão computacional, mais especificamente na área de PDI (Processamento Digital de Imagens), importantes para a extração de características de imagens e é de suma importância para o desenvolvimento deste trabalho. Com mais de 47 mil usuários espalhados pelo mundo a biblioteca ainda conta com otimizações. A biblioteca permite ainda a captura e processamento a partir de fluxo de vídeo, obtido por qualquer tipo de sensor de captura de imagens. Tal capacidade torna a biblioteca extremamente adequada ao projeto.

A ferramenta *Tesseract* realiza detecção de caracteres via OCR, a partir de imagens capturadas pelo fluxo de vídeo de uma *webcam*, acoplada a um computador pessoal. Esse comparativo terá a finalidade de validação da eficiência do projeto. Na figura 3 está representada de maneira simplificada os passos do funcionamento do *software*.

[Pedregosa et al. 2011].

A primeira etapa é a iniciação do *software* pela interface desenvolvida. Ao começar a execução, o *software* habilita o fluxo de vídeo, ou seja, ativar a câmera que está sendo utilizada. Em seguida, começa a fase de pré-processamento da imagem. Nesta fase é feita a conversão do *RGB* da imagem para a escala de cinza, na qual atribui-se um mesmo valor nas três matizes (*red*, *green* e *blue*) de cada *pixel*, utilizando-se o método *cvt-*

Figura 3. Fluxograma do funcionamento do software



Fonte: Elaborada pelo autor

Color do *OpenCV*. Ainda na mesma etapa, é feito a média gaussiana, que capta um *pixel* e realiza uma média entre os *pixels* vizinhos deste e realiza uma média. Depois, recolhe esse novo valor e armazena em um *pixel* na posição equivalente em uma nova imagem, utilizando o método *bilateralFilter* do *OpenCV*. Posteriormente, passa-se para a segmentação da imagem. Aqui ocorre a binarização imagem, onde é feita uma comparação de cada *pixel* com o valor de *threshold*, e caso ultrapassar o valor o *pixel* ganha uma coloração branca, caso contrário, uma cor preta, que é feita pelo método *canny*. Em continuidade, é realizado a detecção da região de interesse(*ROI*), que é o retângulo, pela método *findContours*. Optou-se por essa forma geométrica, pois os caracteres estão, geralmente, em uma folha ou livro e esses materiais apresentam uma configuração retangular. Caso for encontrado a *ROI*, ela é armazenada em formato de imagem. Caso contrário será repetida o ciclo de pré-processamento da imagem até que se ache a *ROI*. A partir da *ROI* armazenada é efetuado a detecção do texto pela biblioteca *Tesseract*. Por fim, o contexto reconhecido pelo *Tesseract* é sintetizado em formato de áudio, por meio da biblioteca *Espeak*.

4. Resultados

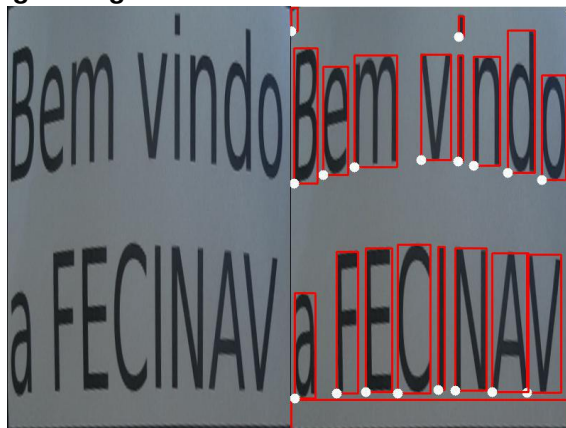
Até o presente momento, foi elaborado somente o software e as etapas concluídas foram:

- Criado a interface inicial, contendo a logo, nome do projeto e botão para iniciar o software.
- Reconhecimento do retângulo(folha): encontrar a *ROI* (região da imagem que apresenta texto) e isolá-la do restante, graficamente representado por um retângulo.
- Reconhecimento dos caracteres: fase na qual ocorre mais erros, por conta do *Tesseract* apresentar alguns falso-positivos.

- Sintetização do texto: o texto reconhecido é reproduzido pelo *Espeak*⁵, um software de sintetização de áudios.

Estas etapas são representadas pelas figuras 4, 5 e 6, respectivamente.

Figura 4. Imagem original vs ROI detectada e caracteres segmentados



Fonte: Elaborada pelo autor

Figura 5. Segmentação dos caracteres



Fonte: Elaborada pelo autor

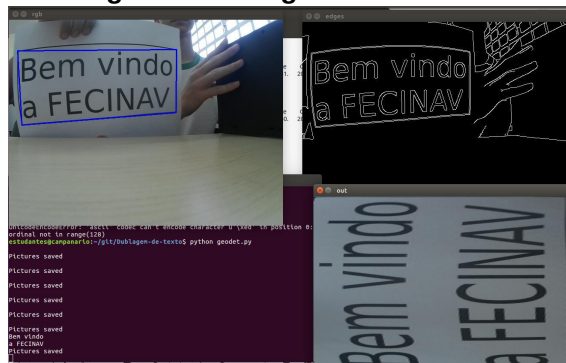
Além disso, há uma interface gráfica para que se possa iniciar o *software*. Na figura 7 está a representação:

O código desenvolvido para o reconhecimento de caracteres está disponível na plataforma online *github*⁶

⁵<http://espeak.sourceforge.net/>

⁶<https://github.com/leonardoYuji/Dublagem-de-texto>

Figura 6. Visão geral do software



Fonte: Elaborada pelo autor

Figura 7. Interface Gráfica



Fonte: Elaborada pelo autor

Durante os testes o software apresentou erros, como:

- Falhas no reconhecimento da ROI: detectando polígonos indesejados contendo quatro arestas, este é um erro pouco ocorrido;
- Falhas no reconhecimento dos caracteres: quando a imagem capturada não tem uma boa definição, devido a angulação, escala, incidência de luz e também por conta da câmera de baixa qualidade utilizada, acaba reconhecendo caracteres falso-positivos podendo encerrar o software, erro que comumente decorre;
- Falha durante a sintetização: por conta do *Tesseract* registrar letras e palavras erradas, o software sintetiza em voz o texto errôneo capturado, acontece conforme o erro anterior;

5. Conclusão

O resultado obtido cumpre totalmente o objetivo inicial que era desenvolver um software capaz de reconhecer textos diversos e sintetizá-lo em formato de áudio com o intuito de auxiliar a vida de pessoas que apresentam uma limitação na leitura e assim integrá-los na sociedade.

Percebe-se no software desenvolvido um grande potencial comercial e latente possibilidade de patente, sendo possível também integrá-lo em diversas áreas do conhecimento.

Como trabalhos futuros destacam-se estudos e aplicação de técnicas baseadas em inteligência artificial para a detecção dos caracteres contidos nas imagens, o desenvolvimento do hardware (prototipação de uma armação em forma de óculos, o circuito de comunicação com um dispositivo externo, que executa o nosso *software*), portar o *software* para uma versão *mobile*, para integrá-lo ao celular da pessoa assistida. Além disso são previstos implementar um software de reconhecimento, provavelmente baseado em rede neural próprio e realizar uma comparação com o que foi desenvolvido a partir do *Tesseract* para se ter uma noção da qualidade. Para isso, serão estudadas redes neurais como KNN ([Hazra et al. 2017]) e CNN ([Chang et al. 2016] [Kang et al. 2014]), além da biblioteca *TensorFlow*⁷ e levar esse projeto para dispositivos móveis. Além do mais, serão estudadas a aplicação da prototipação com ferramentas de modelagem em 3D, como *Blender*⁸ e posteriormente impressão com uma impressora 3D. Para a captura de imagens pretende-se usar um sensor CMOS e para comunicação um circuito de comunicação sem fio, como por exemplo o ESP8266-1⁹.

Referências

- Agarwal, M. and Kaushik, B. (2015). Text recognition from image using artificial neural network and genetic algorithm. In *Green Computing and Internet of Things (ICG-CIoT), 2015 International Conference on*, pages 1610–1617. IEEE.
- Antonello, L. L. R. (2017). Identificação automática de placa de veículos através de processamento de imagem e visão computacional.
- Bruckschen, M., Rigo, S. J., and Fagundes, É. (2007). Desenvolvimento de software educacional livre e inclusão de alunos com deficiência visual. *RENOTE*, 5(2).
- Campos, T., Bampi, S., and Susin, A. (2005). Sistema de identificação de placas por processamento automático de imagens. *Porto Alegre*, 10.
- Chang, J., Gupta, S., and Zhang, A. (2016). Painfree latex with optical character recognition and machine learning.
- Chen, D., Odobez, J.-M., and Boulard, H. (2004). Text detection and recognition in images and video frames. *Pattern recognition*, 37(3):595–608.
- Coates, A., Carpenter, B., Case, C., Satheesh, S., Suresh, B., Wang, T., Wu, D. J., and Ng, A. Y. (2011). Text detection and character recognition in scene images with unsupervised feature learning. In *Document Analysis and Recognition (ICDAR), 2011 International Conference on*, pages 440–445. IEEE.
- Costa, Y. M. et al. (2013). Reconhecimento de gêneros musicais utilizando espectrogramas com combinação de classificadores.
- Esquef, I. A., ALBUQUERQUE, M. P. d., and ALBUQUERQUE, M. P. d. (2003). Processamento digital de imagens. *CENTRO BRASILEIRO DE PESQUISAS FÍSICAS-CBPF*.
- Gonzalez, R. C. and Woods, R. E. (2000). *Processamento de imagens digitais*. Edgard Blucher.

⁷<https://www.tensorflow.org/?hl=pt-br>

⁸<https://www.blender.org/>

⁹<https://www.tensorflow.org/?hl=pt-br>

- Hazra, T. K., Singh, D. P., and Daga, N. (2017). Optical character recognition using knn on custom image dataset. In *Industrial Automation and Electromechanical Engineering Conference (IEMECON), 2017 8th Annual*, pages 110–114. IEEE.
- Kang, L., Ye, P., Li, Y., and Doermann, D. (2014). A deep learning approach to document image quality assessment. In *Image Processing (ICIP), 2014 IEEE International Conference on*, pages 2570–2574. IEEE.
- Neumann, L. and Matas, J. (2010). A method for text localization and recognition in real-world images. In *Asian Conference on Computer Vision*, pages 770–783. Springer.
- Neumann, L. and Matas, J. (2012). Real-time scene text localization and recognition. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3538–3545. IEEE.
- Ossada, S. A. R. and Rodrigues, S. C. M. (2015). Uma análise de softwares para inclusão de deficientes auditivos na educação à distância. *Reverte-Revista de Estudos e Reflexões Tecnológicas da Faculdade de Indaiatuba*, (13).
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830.
- Rigolon, D. and Almeida, H. V. S. d. (2016). Projeto de equipamento sensorial para orientação e mobilidade de deficientes visuais. B.S. thesis, Universidade Tecnológica Federal do Paraná.
- Souza, K. P. and Pistori, H. (2005). Implementação de um extrator de características baseado em momentos da imagem. In *XVIII Brazilian Symposium on Computer Graphics and Image Processing-SIBGRAPI, III Workshop de Trabalhos de Iniciação Científica em Computação Gráfica e Processamento de Imagens-WICCGPI, Natal*.