

Reconhecimento Automático de Tipos para Ensino de Tipografia a Deficientes Visuais

Fernanda G. Vilela, João S.A.R. Lima e Mylène C.Q. Farias

Resumo—O deficiente visual vivencia diariamente diversos tipos de exclusão, entre elas as exclusões social, educacional e cultural. Adicionalmente, no campo das artes, percebe-se deficientes visuais têm dificuldades de apreciar a mais variadas expressões artísticas, como teatros, museus, filmes, pinturas, etc. De forma semelhante, a linguagem estética transmitida pela tipografia é inacessível a deficientes visuais. Neste trabalho, apresentamos um algoritmo para reconhecimento automático de tipos (fontes) que é parte de um sistema para ensino de tipografia a deficientes visuais. Este sistema é composto por uma material tátil e um software que reconhece uma foto contendo um caractere, identifica e fornece informações sobre o seu tipo. O algoritmo de reconhecimento de tipos proposto utiliza o operador Padrão Binário Local (LBP – *Local Binary Pattern*) para extração de atributos. O algoritmo foi testado com dois classificadores: a Máquina de Vetor de Suporte (SVM, em inglês *Support Vector Machine*) e a Floresta Aleatória (RF, do inglês *Random Forest*). Como parte do projeto, foi construído um banco de imagens de caracteres, contendo exemplos de 9 famílias tipográficas. Os resultados mostraram que, apesar do reconhecimento de tipos ser um problema difícil, o algoritmo proposto é capaz de identificar os tipos com uma acurácia de 84.91%.

Palavras-Chave—Aprendizado de Máquina, SVM, Floresta Aleatória, Sistemas de Inclusão, Tipografia, LBP, reconhecimento de padrões.

I. INTRODUÇÃO

Promover a inclusão de pessoas com deficiências na sociedade continua sendo uma desafio nos dias atuais. Bersch [1] define tecnologia assistiva como o conjunto de recursos que amplia as habilidades daqueles que possuem alguma deficiência. Segundo ele, a tecnologia assistiva em um contexto educacional (ou informacional) permite que a pessoa com deficiência rompa barreiras que limitam ou impedem o seu acesso à informação. A tecnologia assistiva favorece a participação e a autonomia do cidadão, provendo-lhe o acesso a objetos e conteúdos. Segundo Radabaugh [2], “Para as pessoas sem deficiência a tecnologia torna as coisas mais fáceis, enquanto que para as pessoas com deficiência a tecnologia torna as coisas possíveis.”

A tipografia, assim como a escrita, é utilizada para comunicação. No entanto, a tipografia comunica também por meio de formas, criando uma linguagem visual e estética. Ao associar o significado das palavras à essa linguagem visual e estética, interpretações específicas do conteúdo podem ser geradas. Frequentemente, este tipo de linguagem visual é mais importante do que o conteúdo das palavras em si, o que



Fig. 1. Exemplo de tipografia aplicada como comunicação majoritariamente estética. Cartaz de Bill Graham feito para banda de Blues de Chicago.

aproxima a tipografia das artes, conforme exemplificado na Figura 1. O deficiente visual é privado desse viés visual da informação, transmitido por meio da tipografia [3].

Nos últimos anos, alguns trabalhos que combinam recursos hápticos com instruções ou acompanhamento sonoro vem sendo desenvolvidos como o objetivo de auxiliar o ensino de deficientes visuais (totais ou parciais). Por exemplo, no sistema proposto por Plimmer [4], os deficientes visuais são ensinados a assinarem o próprio nome por meio de instruções de áudio. Ao mesmo tempo, por meio do tato, os deficientes percebem a sua própria assinatura em alto relevo. No trabalho de Ballestero-Alvarez, além do tato e da audição, outros componentes sensoriais são usados, o que permite que diversos aspectos do ambiente sejam percebidos [5].

Um exemplo de uma plataforma desenvolvida para deficientes visuais é o DOSVOX, desenvolvido pela Universidade Federal do Rio de Janeiro [6]. O DOSVOX é um sistema operacional que possibilita ao deficiente visual utilizar um computador para realizar variadas tarefas. Esta ferramenta gratuita é bastante popular no Brasil, onde se tornou uma ferramenta muito importante na inclusão dos deficientes visuais. Atualmente, o DOSVOX inclui jogos didáticos, tradução do Braille para a escrita latina e vice-versa, leitura de textos por meio de síntese de voz, composição e impressão de partituras, entre outras funcionalidades [7].

Outras tecnologias assistivas que se destacam são o *Jaws* e o *Virtual Vision*, que são ambos *softwares* leitores de tela para



Fig. 2. Conjunto de placas apresentando as nove tipografias contidas no Sistema Tipo Tátil [13], [12].

uso em computadores pessoais¹. Outra ferramenta interessante é o *TactileView* da *Tactile Graphics*², que é um software que permite a impressão de figuras e gráficos com pontos em alto relevo [7].

Apesar da grande variedade de tecnologias assistivas já desenvolvidas, até o nosso conhecimento, existem poucas ferramentas dedicadas ao ensino da tipografia a deficientes visuais. Mais especificamente, o reconhecimento de tipos (fontes ou famílias tipográficas) é um campo não muito explorado na área de reconhecimento de padrões [8]. Geralmente, utiliza-se algoritmos simples de reconhecimento de tipos para melhorar o desempenho de sistemas de reconhecimento de caracteres, especialmente em casos em que os documentos possuem vários tipos [9] [10]. Entre os trabalhos de reconhecimento de tipos, destaca-se o trabalho de Zramdini [8], que utiliza as características básicas das famílias tipográficas, tais como a anatomia de cada letra e as peculiaridades de determinados tipos, para identificá-las. Uma das restrições deste trabalho é que a identificação dos tipos requer o conhecimento do caractere que está sendo avaliado.

O trabalho apresentado neste artigo é parte do projeto “Tipografia Tátil para Deficientes Visuais”, desenvolvido na Universidade de Brasília, que tem como objetivo auxiliar no ensino de tipografia a deficientes visuais [12], [13]. O sistema, denominado **Tipo Tátil**, inclui um material tátil, com o qual o deficiente visual pode interagir, que consiste em placas tridimensionais com as letras inseridas em alto relevo. A Figura 2 mostra um subconjunto destas placas, onde podem ser visualizadas as representações do caractere ‘a’ em vários tipos ou fontes. O material tátil inclui 9 famílias tipográficas, que foram escolhidas com base na literatura de classificação tipográfica. Exemplos dos 9 tipos disponíveis no sistema, para o caractere “a”, são apresentados na Figura 3. Além do material tátil, o sistema inclui o algoritmo de reconhecimento de tipos (apresentado neste trabalho) e um material descritivo (áudio e escrito) sobre as famílias tipográficas.

O sistema de ensino de tipografia a deficientes visuais funciona da seguinte forma. O usuário (deficiente visual)

a adobe Jenson
 a adobe Garamond
 a Baskerville
 a Didot
a Clarendon bold
a FranklinGothic URW Demi
 a Helvetica
 a Futura book
 a Gill sans

Fig. 3. Conjunto de 9 tipos incluídos no sistema Tipo Tátil, exemplificados aqui pela letra ‘a’ [13].

examina fisicamente uma das peças. Após examiná-la, ele a insere em uma área pré-determinada para que uma imagem da peça seja capturada. O algoritmo de reconhecimento identifica o tipo do caractere representado na peça, sem a necessidade de identificação do caractere ou mesmo de qualquer interação com o usuário. Finalmente, o sistema provê ao usuário (por meio de áudio) informações detalhadas sobre a sua tipografia, incluindo especificações estilísticas e contexto histórico.

Neste artigo, apresentamos o algoritmo para o reconhecimento do tipo da peça utilizado para complementar o reconhecimento de caracteres e contribuir para a transmissão dos significados implícitos na tipografia.

O algoritmo proposto utiliza técnicas de aprendizado de máquinas (*Machine Learning*) para reconhecer o tipo do caractere contido em uma imagem (fotografia capturada pelo sistema) [14]. Mais especificamente, utiliza-se o operador Padrão Binário Local (LBP -*Local Binary Pattern*) [15] para extração de atributos. Neste artigo, apresentamos resultados do reconhecimento utilizando dois tipos de classificadores: a Máquina de Vetor de Suporte (SVM, do inglês *Support Vector Machine*) [16], [17] e a Floresta Aleatória (RF, do inglês *Random Forest*) [18].

O artigo está dividido da seguinte forma. Na Seção II, descrevemos o funcionamento do LBP. Na Seção III, descrevemos o banco de imagens criado neste trabalho. Na Seção IV detalhamos o processo de treinamento e validação do algoritmo de reconhecimento de tipos. Na Seção V discutimos os resultados obtidos. Finalmente, na Seção VI descrevemos as nossas conclusões e trabalhos futuros.

II. O OPERADOR PADRÃO BINÁRIO LOCAL (LBP)

O operador Padrão Binário Local (LBP, do inglês *Local Binary Pattern*) é um dos descritores de texturas mais poderosos. Foi proposto em 2002 por Ojala *et al.* [15] e, desde então, tem sido utilizado com sucesso em aplicações na área de visão computacional.

¹<http://www.freedomscientific.com/Products/Blindness/JAWS>

²<http://www.indexbraille.com/en-us/support/braille-editors/tactileview-tactile-graphics>

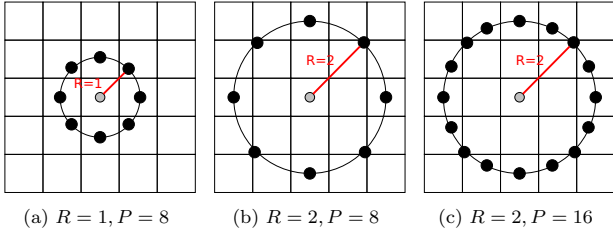
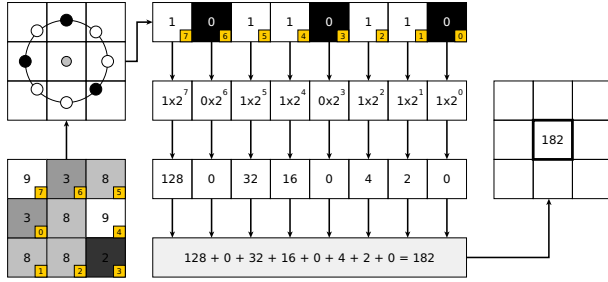
Fig. 4. P vizinhos extraídos a uma distância circular R do pixel central.

Fig. 5. Cálculo dos rótulos LBP.

O operador LBP tradicional tem a seguinte forma:

$$LBP_{R,P}(I_c) = \sum_{p=0}^{P-1} S(I_p - I_c) 2^p, \quad (1)$$

na qual

$$S(t) = \begin{cases} 1 & t \geq 0, \\ 0 & \text{caso contrário.} \end{cases} \quad (2)$$

Na Equação 1, $I_c = I(x, y)$ é o pixel central na posição (x, y) e $I_p = I(x_p, y_p)$ é um pixel na vizinhança de I_c (posição $x_p = x + R \cos(2\pi p/P)$ e $y_p = y - R \sin(2\pi p/P)$), P é o número total de pixels nesta vizinhança e R é a distância destes pixels ao pixel central. A Figura 4 ilustra exemplos de amostragens simétricas para diferentes valores de P e R .

A Figura 5 ilustra os passos para aplicação do operador LBP em um único pixel ($I_c = 8$) localizado no centro de um bloco 3×3 de uma imagem (ver parte inferior esquerda da figura). Os números nos quadrados amarelos do bloco representam a ordem em que o operador é calculado (sentido anti-horário). Nesta figura, usamos um raio de vizinhança unitário ($R = 1$) e oito pixels na vizinhança ($P = 8$). Após calcular a função $S(t)$ (Equação 2) para cada pixel I_p de vizinhança, obtemos uma saída binária para cada I_p ($0 \leq p \leq 7$), como ilustrado na parte superior esquerda da Figura 5. Os círculos pretos correspondem a '0' e os círculos brancos a '1'. Estas saídas binárias são armazenadas em formato binário, de acordo com a sua posição (índices nos quadrados amarelos). Em seguida, o número binário resultante é convertido para o formato decimal. Este número decimal é a saída gerada pelo operador LBP para o pixel I_c .

Quando uma imagem é rotacionada, os valores amostrados I_p mudam a sua posição no perímetro de amostragem em torno do pixel central I_c , o que gera um deslocamento circular no número binário gerado. Para eliminar este efeito, usamos a

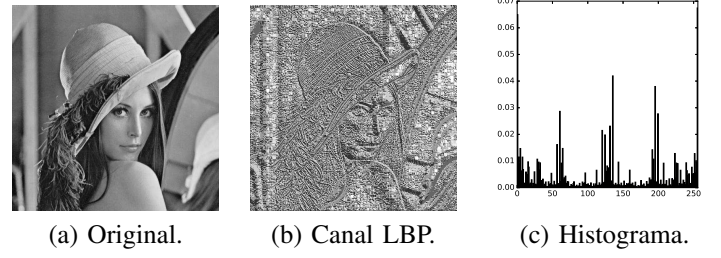


Fig. 6. Exemplo de uma imagem, o seu canal LBP, e o histograma LBP correspondente (atributos extraídos).



Fig. 7. Ilustração do processo de pré-processamento das imagens do banco: imagem original, imagem em escalas-de-cinza, imagem binarizada, imagem após dilatação, imagem original sobreposta com retângulos delimitadores dos caracteres.

seguinte variante do operador LBP, que é invariante à rotação:

$$LBP_{R,P}^{ri}(I_c) = \min\{ROT_R(LBP_{R,P}(I_c), k)\}, \quad (3)$$

na qual $k = \{0, 1, 2, \dots, P-1\}$ e $ROT_R(x, k)$ é o operador de deslocamento à direita circular, que desloca a t -tuple x de k posições.

Após a aplicação do operador LBP em todos os pixels da imagem, obtemos um conjunto de rótulos que forma o Canal LBP. Em seguida, calculamos o histograma normalizado do Canal LBP. Este histograma forma o vetor de atributos que é a entrada de um algoritmo de aprendizado de máquina. No nosso caso, o vetor de atributos é a entrada dos classificadores SVM e RF. Para efeitos de ilustração, a Figura 6 mostra uma imagem, o seu Canal LBP e o histograma do Canal LBP.

III. CONSTRUÇÃO DO BANCO DE IMAGENS

A maioria das imagens usadas neste trabalho foi obtida no *Fonts in Use*, um website que disponibiliza fotografias de projetos de *design* gráfico classificadas por tipografia³. Também utilizamos imagens obtidas em sítios de busca convencionais, o que exigiu uma análise manual da tipografia dos caracteres das imagens.

Após esta análise, as imagens foram pré-processadas para identificação dos caracteres. Primeiramente, elas foram convertidas para níveis de cinza e, depois, inversamente binarizadas. Em seguida foi realizada uma dilatação morfológica da imagem binária, utilizando um *kernel* em cruz com dimensão 3×3 . Esta operação permitiu que os contornos dos caracteres ficassem melhor definidos. Em seguida, foram identificados os caracteres individuais de cada imagem. Este processo é ilustrado na Figura 7 para uma imagem do tipo Futura.

A Figura 8 apresenta amostras das imagens do banco de imagens construído neste trabalho. Nesta figura, acima de

³Disponível em: <http://www.fontsinuse.com>

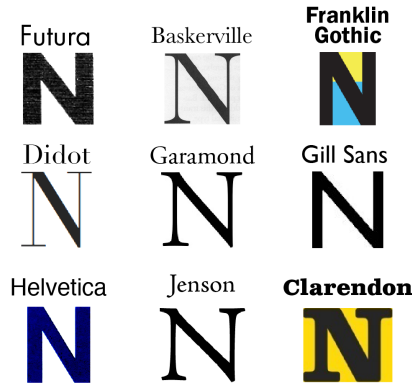


Fig. 8. Amostras do banco de imagens. Para efeito de ilustração, em cada uma das imagens nesta figura, foi adicionada o tipo correspondente.

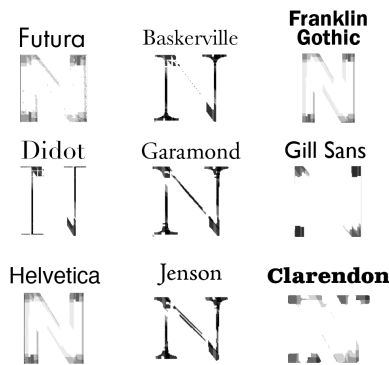


Fig. 9. Amostras de imagens do banco de imagens após processo de extração de atributos, aplicação do LBP.

cada caractere, adicionamos o nome do tipo correspondente. A versão final do banco de imagens contém 6.750 imagens no total, com 750 imagens de cada família tipográfica. A resolução espacial mínima das imagens é 45×45 pixels.

IV. PARÂMETROS DO TREINAMENTO E VALIDAÇÃO

A Figura 9 apresenta os Canais LBP das imagens da Figura 8. Os parâmetros do operador LBP utilizados neste trabalho foram $R = 9$ e $P = 21$, que foram escolhidos empiricamente. Como já mencionado na seção II, após o cálculo dos Canais LBP, calculamos os seus histogramas. Estes histogramas normalizados formam os vetores de atributos (*feature vectors*). Cada imagem do banco possui um vetor de atributo e um rótulo de tipo. Os algoritmos de classificação (SVM e RF) utilizam estas informações (como entrada) para realizar a predição do tipo do caractere contido na imagem capturada.

Para treinar e testar os algoritmos de classificação, utilizamos o método de validação cruzada *Stratified K-Fold* [19], que é uma variação do método *K-Fold*. No método *K-Fold*, o conjunto de amostras (imagens) é particionado aleatoriamente em K subconjuntos de tamanhos iguais. Destes subconjuntos, $K - 1$ suconjuntos são utilizados para treinamento e 1 subconjunto é utilizado para validação. O processo de validação cruzada é repetido K vezes e a média dos resultados destas simulações são reportados. No método *Stratified K-Fold*, os subconjuntos possuem aproximadamente o mesmo número

Kernels	C	γ	d
Linear	100		
Polinomial	100	19	3
Sigmoidal	100		
RBF	100	10950	

TABELA I

PARÂMETROS DOS *kernels* DO SVM QUE OBTIVERAM MELHOR ACURÁCIA NA VALIDAÇÃO. FOI UTILIZADO RANDOM STATE = 1 E VALORES PADRÕES DA BIBLIOTECA SCIKIT-LEARN.

Classificador	Acurácia[%]
SVM-Linear	45,17
SVM-RBF	74,70
SVM-Polinomial	54,19
SVM-Sigmoidal	35,61
RF	95,78

TABELA II

ACURÁCIA DOS CLASSIFICADORES RF E SVM COM DIVERSOS TEXTITKernels.

de amostras para cada classe. No caso de uma classificação binária, isto significa que cada subconjunto tem aproximadamente as mesmas proporções em cada uma das duas classes.

Neste trabalho, utilizamos a implementação dos classificadores SVM e RF da biblioteca *SciKit-Learn*, com $K = 7$. A Tabela I apresenta os parâmetros utilizados para cada *kernel* do classificador SVM. Para o treinamento do SVM, foi utilizada uma abordagem um-contra-todos. Para o classificador RF, o parâmetro ‘número de árvores de decisão’ utilizado foi igual a 82. Com relação ao restante dos parâmetros, utilizamos os valores padrões da biblioteca *SciKit-Learn*. Nos resultados reportamos a acurácia da classificação que é definida como a porcentagem de predições corretas.

V. RESULTADOS

A Tabela II apresenta o sumário dos valores médios de acurácia obtidos com o classificador SVM e com o classificador RF, considerando 9 tipos e 7 simulações (*Stratified 7-Fold*). Como já discutido, utilizamos os *kernels* linear, RBF, polinomial e sigmoidal para o classificador SVM. Observa-se dos resultados que o *kernel* que produziu o melhor resultado foi o RBF. Entretanto, o classificador RF apresentou resultados bem mais expressivos do que qualquer uma das versões do SVM.

Para analisar a robustez dos algoritmos, no que diz respeito ao número de classes utilizadas, variamos o número de tipos de 2 a 9. Para os diversos números de classes, mantemos a mesma divisão do banco de imagens em subgrupos de treinamento e validação (conforme descrito na seção IV). A Tabela III apresenta os valores médios de acurácia para os classificadores RF, SVM-linear e SVM-RBF quando variamos o número de classes de 2 a 9. A Figura 10 apresenta o gráfico correspondente a estes resultados.

Percebe-se que o aumento na quantidade de classes (tipos) diminui a acurácia média dos classificadores. Entretanto, o

Quantidade de Tipos	2	3	4	5	6	7	8	9
SVM-Linear: Acurácia [%]	82,54	71,05	67,77	57,74	56,33	50,84	48,59	45,17
SVM-RBF: Acurácia [%]	90,11	85,85	83,21	80,90	78,40	77,17	76,08	74,70
RF: Acurácia [%]	95,78	93,40	92,74	88,86	88,79	86,47	86,24	84,91

TABELA III

ACURÁCIA DA CLASSIFICAÇÃO UTILIZANDO OS CLASSIFICADORES RF E SVM (KERNEL LINEAR E RBF), COM NÚMERO DE TIPOS VARIANDO DE 2 A 9.

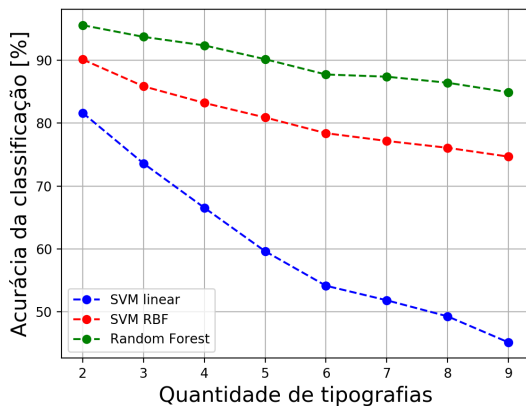


Fig. 10. Gráfico da acurácia média dos classificadores FA, SVM-Linear e SVM-RBF, com base na variação da quantidade de classes (tipos).

classificador SVM-linear tem o seu desempenho mais severamente afetado do que os outros dois classificadores. Isto se deve à alta semelhança entre os atributos dos diferentes tipos [16], o que dificulta uma separação linear no espaço de atributos. Embora o SVM-RBF apresente um resultado mais robusto, possivelmente devido à transformação dos atributos para um espaço no qual os exemplos podem ser linearmente separáveis, o classificador RF apresentou os melhores resultados em todos os testes. O alto desempenho deste classificador é atribuído à sua robustez a pequenas diferenças entre os vetores de atributos [20]. Além disso, devido à utilização de diversas árvores de decisão, o modelo é mais robusto a ruídos.

VI. CONCLUSÕES E FUTUROS TRABALHOS

Neste trabalho, apresentamos um algoritmo para reconhecimento automático de tipos que é parte de um sistema para ensino de tipografia a deficientes visuais. O algoritmo de reconhecimento de tipos proposto utiliza o operador LBP para extração de atributos. Foram testados os classificadores SVM e RF, sendo que este último apresentou um melhor desempenho. Como parte do projeto, foi construído um banco de imagens de caracteres, contendo até 9 tipos diferentes. Os resultados mostraram que, apesar do reconhecimento de tipos ser um problema difícil devido à similaridade entre alguns tipos, o algoritmo proposto é capaz de identificar os tipos com uma acurácia de 84,91%. Um possível trabalho futuro consiste em testar o uso de uma Rede Neural Convolucional (*Convolutional Neural Network* ou CNN em inglês) para reconhecimento de tipos.

REFERÊNCIAS

- [1] Bersch, Rita. "Introdução à Tecnologia Assistiva" Porto Alegre: CEDI, 2013.
- [2] M.P. Radabaugh, "NIDRR's long range plan-technology for access and function research", Section two: NIDRR research agenda, Chapter 5: Technology for access and function. *United States*, 1993.
- [3] C. Velasco, A. T. Woods, S. Hyndman, C. Spence, "The taste of type-face", *i-Perception* 6, Vol. 6, No. 4, pp. 1–10, 2015 (ISSN 20416695).
- [4] Beryl Plimmer, Andrew Crossan, Stephen A. Brewster, and Rachel Blagojevic, "Multimodal collaborative handwriting training for visually-impaired people." In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 393-402. ACM, 2008.
- [5] Ballesterio-Álvarez, Jose Alfonso. "Multissensorialidade no ensino de desenho a cegos." Tese de Doutorado, Universidade de São Paulo, 2003.
- [6] J. A. Borges, "Dosvox: Uma nova realidade educacional para deficientes visuais", *Revista Benjamin Constant*, 1998.
- [7] E. S. M. d. S. Amorim, J. L. Carvalho, L. K. B. Menezes, "Educação de cegos mediada pela tecnologia", *Secretaria de Educação de Salvador: Salvador*, p. 1–12, 2009.
- [8] Abdelwahab Zramdini, "Study of optical font recognition based on global typographical features." University of Fribourg, Theses 1106 (1995).
- [9] Shi, Hongwei, and Theo Pavlidis. "Font recognition and contextual processing for more accurate text recognition." In *Proceedings of the Fourth International Conference on Document Analysis and Recognition*, vol. 1, pp. 39-44. IEEE, 1997.
- [10] S. L. Manna, A. Colia, A. Sperduti, "Optical font recognition for multi-font OCR and document processing". *Proceedings of the Tenth International Workshop on Database and Expert Systems Applications*, pp. 549–553, 1999.
- [11] Virtual Vision: Acessibilidade para pessoas com deficiência visual (2016), Disponível em: <http://www.virtualvision.com.br/>.
- [12] Cruz, L. E.; Mayanardes, A. C. "Tipografia Tátil". XX Congresso de la Sociedad Iberoamericana de Gráfica Digital, Vol. 3, no. 1, pp. 360–365, 2016.
- [13] Cruz, L.; "Tipografia Tátil para Deficientes Visuais". Monografia do Bacharelado em Design, Universidade de Brasília, 2017.
- [14] C. M. Bishop, *Pattern recognition and machine learning*, Vol. 128, pp. 1–3, 2006.
- [15] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 7, pp. 971–987, 2002.
- [16] Alpaydin, Ethem "Kernel Machines." *Introduction to Machine Learning*. 2nd ed. MIT, pp. 309-339, 2010.
- [17] Corinna Cortes, Vladimir N. Vapnik, "Support-vector networks", *Machine Learning*, Vol. 20, No. 3, pp. 273-297, 1995.
- [18] Tin Kam Ho, "Random Decision Forests", in: *Proceedings of the 3rd International Conference on Document Analysis and Recognition*, Montreal, QC, 14-16 August 1995, pp. 278-282.
- [19] Payam Refaeilzadeh, Lei Tang, and Huan Liu. "Cross-validation." In *Encyclopedia of database systems*, pp. 532-538. Springer US, 2009.
- [20] Leo Breiman, "Random forests", *Machine learning* 45, No. 1 (2001), pp. 5-32.