# Reconhecimento de sinais estáticos de LIBRAS com *Support Vector Machines* usando Kinect

**Leonardo Perdomo**[1], **Mozart Lemos de Siqueira**[1]

[1]Curso de Ciência da Computação – Centro Universitário La Salle (UNILASALLE)
Caixa Postal 2.288 – 92.010-000 – Canoas – RS – Brazil

leonardo.perdomo.edu@outlook.com, mozarts@unilasalle.edu.br

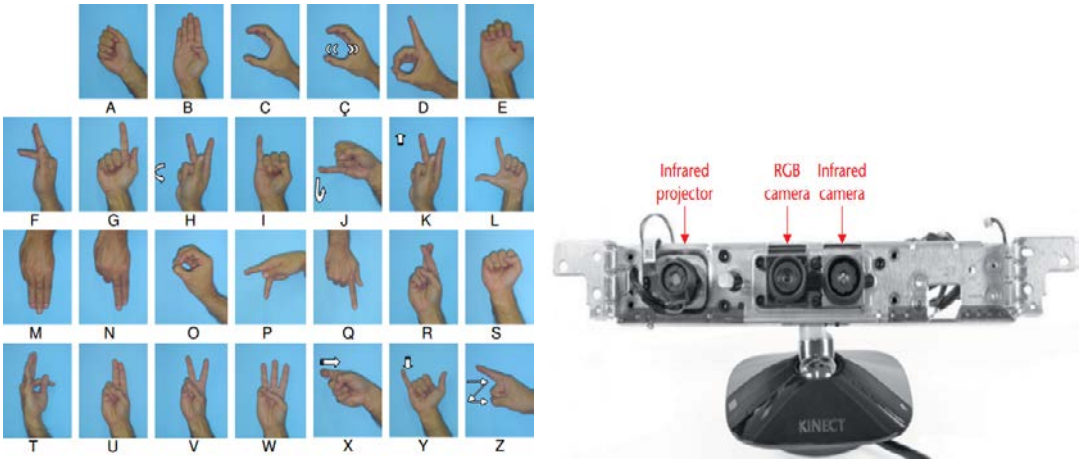
***Abstract.*** *This paper presents the author experiments with his own developed prototype aiming computer recognition of the manual alphabet static signs from the Brazilian Sign Language (LIBRAS), captured with Microsoft Kinect depth sensor, using image pattern recognition techniques along with Multiclass Support Vector Machines (SVM) classifiers. The prototype results and an efficiency analysis with execution time measures are presented. The device practical distance limits interval (0.8m to 2.5m) was considered.*

***Resumo.*** *Este artigo apresenta a experimentação realizada pelo autor em um protótipo desenvolvido pelo próprio para o reconhecimento computacional dos sinais estáticos do alfabeto manual da Língua Brasileira de Sinais (LIBRAS), capturados através do sensor de profundidade do Microsoft Kinect, utilizando técnicas de reconhecimento de padrões em imagens com classificação por Support Vector Machines (SVM) em uma abordagem multiclasse. São apresentados os resultados do protótipo e uma análise de eficiência em medições de tempo de execução e acerto no reconhecimento de sinais. Foi considerado o intervalo de distância dentro dos limites práticos (0,8m à 2,5m) do near-mode do dispositivo.*

## 1. Introdução

O desafio do reconhecimento das línguas de sinais por sistemas computacionais envolve a compreensão de sua organização estrutural através de estudos linguísticos, como os realizados em [Brito 1995] e [de Quadros 1997], que descrevem a composição da Língua Brasileira de Sinais (LIBRAS) em relação à *American Sign Language* (ASL) e às línguas faladas, também observando sua independência e originalidade sociocultural, a exemplo das diferenças gestuais existentes por regiões do Brasil.

Baseando-se nas diferentes abordagens adotadas em alguns dos trabalhos publicados sobre o tema [de Souza et al. 2012b], [de Souza et al. 2012a], [Zhu e Wong 2012] e [de Souza e Pizzolato 2013], este estudo apresenta o desenvolvimento de um protótipo para reconhecimento de 20 letras do alfabeto manual de LIBRAS (Figura 1a), escolhidas em razão da ausência de movimento na realização de seus respectivos gestos. A captura do gesto ocorre pelo sensor de profundidade do Microsoft Kinect (Figura 1b), seguida da segmentação e extração de descritores da imagem, posteriormente classificados por *Support Vector Machines* (SVM). Também são apresentadas medições de eficiência do protótipo, analisando os resultados obtidos em relação ao observado em outros estudos publicados. A concepção, implementação, experimentação e análise de resultados do trabalho foram realizadas pelo aluno Leonardo Perdomo, sob orientação do professor Mozart Lemos de Siqueira.




(a) Alfabeto manual de LIBRAS, setas indicam gestos com movimento [FADERS 2010]

(b) Hardware do Kinect com projetor IR e câmeras IR e RGB em destaque [Zhang 2012].

**Figura 1. Alfabeto manual de LIBRAS e o dispositivo Kinect utilizado na captura de movimento do *console* de *videogame* Xbox 360.**

No decorrer da seção 2 são apresentados outros trabalhos publicados sobre o tema estudado e seus respectivos resultados. Em seguida, na seção 3, é descrita a metodologia adotada para o desenvolvimento e análise do protótipo, disponível em código-aberto[1], para reconhecimento em tempo real dos sinais estáticos do alfabeto manual de LIBRAS, utilizando classificação por SVM de descritores SURF extraídos da captura de profundidade do Kinect. Os resultados de medições de eficiência do protótipo em percentuais médios de acerto e tempo de execução, considerando três implementações de *kernel* SVM (linear, polinomia e radial) no intervalo de distância dentro dos limites práticos em *near-mode* ($0.8m$ à $2.5m$) do dispositivo são apresentados na seção 4, com a exposição na seção 5 das conclusões obtidas pelo estudo realizado.

## 2. Trabalhos Relacionados

Os meios utilizados para o reconhecimento computacional de gestos das línguas de sinais podem envolver, conforme análise realizada por [Mitra e Acharya 2007], a coleta de dados por meio de sensores equipados em luvas, ou a utilização de câmeras em conjunto com técnicas para análise de imagens. Este levantamento também identifica a adoção, por trabalhos que abrangem este tema, da classificação dos gestos em sinais estáticos ou dinâmicos, definidos pela ocorrência ou não de movimento após sua postura inicial. Para um aprofundamento mais específico do tema, este estudo limitou-se a investigar referências que utilizaram técnicas de visão computacional para o reconhecimento do alfabeto e de palavras das línguas de sinais. Os trabalhos analisados apresentam resultados significativos, com limitações quanto à iluminação do ambiente quando no uso de câmeras RGB convencionais [Carneiro et al. 2009], [Pizzolato et al. 2010], relativamente superadas com a adoção de dispositivos com sensores de profundidade (RGB-D), como o Microsoft Kinect [Zafrulla et al. 2011], [Zhu e Wong 2012], [de Souza e Pizzolato 2013], [Almeida et al. 2014], [Pedersoli et al. 2014], [Rioux-Maldague e Giguere 2014], [Dong et al. 2015].

[1]Disponível em https://github.com/lperdomo/reconhecedor_sinal_estatico

Entre os trabalhos analisados, [de Souza et al. 2012b], [de Souza et al. 2012a] destacam-se pela compreensiva análise de técnicas para o reconhecimento de gestos, verificando, por exemplo, uma superioridade de *Support Vector Machines* (SVM) com *kernels* não-lineares (polinomial quadrático e radial) no melhor resultado encontrado em relação às *Feedfoward Neural Networks* (FNN). Ambos os trabalhos citados também apontam a necessidade de treinamentos significativamente menores com uso de classificadores por SVM em relação às FNNs. Aprofundando os estudos realizados apenas com câmeras RGB, [de Souza e Pizzolato 2013] investigam a adição do Kinect, e de um modelo de etapas para classificação de postura e sequência de movimento de sinais dinâmicos, apresentado em [Pizzolato et al. 2010]. Os autores verificam uma superioridade substancial na aprendizagem discriminativa com o classificador de sequência *Hidden Conditional Random Fields* (HCRF) em relação à generativa de *Hidden Markov Models* (HMM).

No uso do Kinect para captura de gestos, [Zhu e Wong 2012] destacam-se ao investigar uma implementação, com extração de descritores SIFT da *American Sign Language* (ASL) e classificação por SVM, considerando 5 distâncias do usuário em relação ao Kinect ($0,6m$, $0,9m$, $1,2m$, $1,5m$ e $1,8m$). Os autores verificam uma redução substancial da qualidade dos descritores de profundidade após $1,5m$ que inviabilizou resultados satisfatórios, optando então pelo uso em conjunto com a câmera RGB do dispositivo.

Entre as publicações mais recentes analisadas, vale destacar a implementação em código aberto de [Pedersoli et al. 2014], para reconhecimento de sinais dinâmicos da ASL usando SVM e HMM. Em [Almeida et al. 2014] é implementada a extração de múltiplas características (velocidade, distância entre pixels, descritores SURF, entre outras), classificando gestos dinâmicos apenas com o uso de SVM. Distinguindo-se na abordagem em relação aos demais estudos observados, [Rioux-Maldague e Giguere 2014] utilizaram *Deep Belief Network* (DBN) com *Restricted Boltzmann Machines* (RBM). Já [Dong et al. 2015] optaram pela utilização de uma luva colorida para segmentação da estrutura das mãos, aplicando então um classificador em *Random Forest* (RF).

Este estudo se assemelha ao realizado por [Zhu e Wong 2012] quanto à utilização de técnicas para extração de descritores e classificação com SVM, avaliando a eficiência do reconhecimento em relação à distância do Kinect, porém distinguindo-se ao considerar somente descritores SURF da captura de profundidade. Também há semelhança na verificação de resultados entre os *kernels* SVM linear, polinomial e radial realizada em [de Souza et al. 2012a], [de Souza et al. 2012b] e [de Souza e Pizzolato 2013].

## 3. Metodologia

Para desenvolvimento e experimentação do protótipo reconhecedor de sinais estáticos do alfabeto manual da Língua Brasileira de Sinais (LIBRAS), foi utilizado um computador Intel I5 4690 3.5GHz com ASUS Z-97 PRO, 8Gb de RAM e placa gráfica GeForce GTX760, em conjunto com um dispositivo Microsoft Kinect *for* Xbox 360. A linguagem de programação C++ foi utilizada em ambiente Windows 8.1 64bits, com IDE Visual C++ 2010 em conjunto com a biblioteca OpenCV 2.4.10, integrada às bibliotecas OpenNI 1.5.4.0 32bits e NITE 1.5.2.21 32bits, acessando os recursos do Kinect pelo driver PrimeSense 5.1.2.1 32bits. A descrição sobre o desenvolvimento do protótipo é apresentada na subseção 3.1, enquanto a experimentação para investigação da eficiência do mesmo, em uma abordagem quantitativa, é apresentada na subseção 3.2.

### 3.1. Desenvolvimento do Protótipo

Para contornar as restrições impostas do modelo Kinect *for* Xbox 360, escolhido pelo custo significativamente menor em relação ao modelo de desenvolvimento, foram utilizadas as bibliotecas OpenCV, NITE e OpenNI com o *driver* PrimeSense, devido à integração entre as mesmas [Cruz et al. 2012], além desta última possuir a biblioteca SURF integrada e implementação SVM *1-vs-1* baseada em LIBSVM [OpenCV 2011]. Desta forma o mapeamento de disparidade capturado pelo sensor de profundidade do Kinect em *near-mode* foi realizado sem uso da sua biblioteca SDK. Já o acesso às estruturas do esqueleto do usuário em nodos foi obtido por meio da biblioteca SkeletonSensor, desenvolvida por [Johnson et al. 2012].

Para segmentação, a área quadrada de 180x180 pixels centralizada no nodo do mão direita foi arbitrariamente escolhida como Região de Interesse (ROI), baseando-se na observação do espaço ocupado na distância de $0,8m$ em relação ao Kinect. O fundo então é removido com base na informação de profundidade do nodo da mão, enquanto, para remoção do antebraço, inicialmente é aplicado *thresholding* com método de Otsu [Otsu 1979], utilizando-o na identificação do polígono que abrange mão e antebraço. Neste polígono é realizada a busca pelo maior círculo interno, pressupondo a palma da mão ao considerar a situação ideal onde o usuário não utilize vestimentas espessas no antebraço que comportariam círculos internos maiores. A informação de profundidade no centro do círculo identificado é utilizada para remoção de regiões mais distantes abaixo do mesmo, na tentativa de remover o antebraço e preservar estruturas da mão para sinais com postura dos dedos apontada para baixo. Em seguida o algoritmo de Canny [Canny 1986] é usado para detecção das as estruturas internas da mão entre as diferentes profundidades detectadas, como pode ser observado na Figura 2.

Após a segmentação das características da mão, a biblioteca SURF é utilizada para extração dos descritores de cada *frame* capturado. Estes são utilizados para composição de histogramas, medindo as ocorrências de descritores extraídos em relação aos previamente definidos em um vocabulário visual. Este vocabulário é composto por descritores agrupados e otimizados por gestos, através do algoritmo *k-means*, também usado para este fim em outros estudos [Csurka et al. 2004], [Jiang et al. 2007]. Por fim, os histogramas são encaminhados às três SVMs distintas, com *kernels* linear, polinomial e radial, para treinamento ou classificação entre as 20 letras definidas do alfabeto manual de LIBRAS.

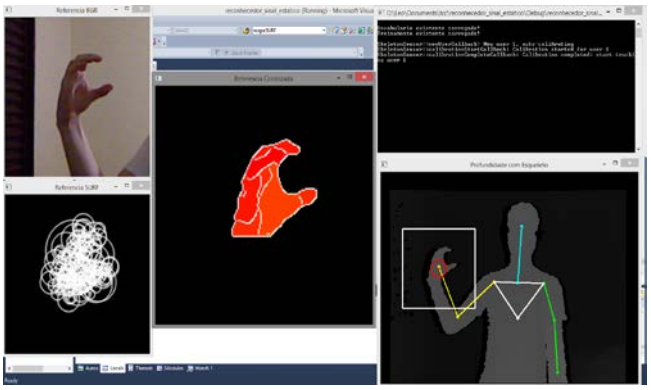


**Figura 2. Protótipo desenvolvido em funcionamento, demonstrando o esqueleto detectado na cena, com a mão segmentada e os descritores SURF encontrados.**

### 3.2. Experimentação do protótipo

Considerando os limites práticos do *near-mode* do Kinect ($0,8m$ à $2,5m$) [Microsoft 2015], foram definidos intervalos de $0,2m$, inicialmente compreendendo as distâncias entre $0,8m$ e $2,4m$ para avaliação do protótipo, com o dispositivo a $1,28m$ de altura. Após as primeiras observações, optou-se pela remoção das distâncias de $2,2m$ e $2,4m$, devido à extração insuficiente de descritores em razão da resolução pequena de captura, somada à alta incidência de ruído (dispersão de IR). Também optou-se pela composição do vocabulário e da primeira avaliação do protótipo com amostras capturadas a $1m$, devido à incidência de ruído ocorrida em $0,8m$ (limite prático do *near-mode*).

Baseando-se na quantidade adotada por [Zhu e Wong 2012], os treinamentos foram realizados capturando 40 amostras por letra do conjunto definido (20 letras) à cada distância treinada, usando a mão direita do autor. O conjunto de amostras, em formato PNG, foi fornecido às três SVMs utilizando o processo de treinamento e otimização existente na biblioteca OpenCV. Os parâmetros para cada *kernel* foram estimados por buscas em grade com validação cruzada *k-fold*, com $k = 10$ (padrão da biblioteca), obtendo os parâmetros $C = 3,125 * 10^2$ (coeficiente de distinção das classes) para os três *kernels*, $\gamma = 0,50625 + 9 * 10^{-11}$ para o polinomial e o radial, além de grau $d = 0,07 + 7 * 10^{-16}$ e $coef0 = 0,1 + 1 * 10^{-15}$ para o polinomial. As validações utilizaram o mesmo critério adotado nos treinamentos, totalizando 5600 amostras, também gravadas em formato PNG.

## 4. Resultados

Em sua primeira avaliação com treinamentos a $1m$, o protótipo obteve percentual médio de acerto entre os três *kernels* de $81,75\% \pm 0,5885\%$, conforme Figura 3. Treinamentos adicionais em $0,8m$, $1,2m$, $1,4m$ e $1,6m$, resultaram em um aumento significativo de acertos em distâncias até $1,4m$. A interrupção de treinamentos em $1,6m$ ocorreu pela insuficiente diminuição do índice de dispersão ($CV = 36,8089\%$), com pequeno aumento no acerto desta distância (aprox. $10,58\%$), de forma semelhante ao verificado por [Zhu e Wong 2012] em descritores de profundidade devido à baixa resolução após $1,5m$.

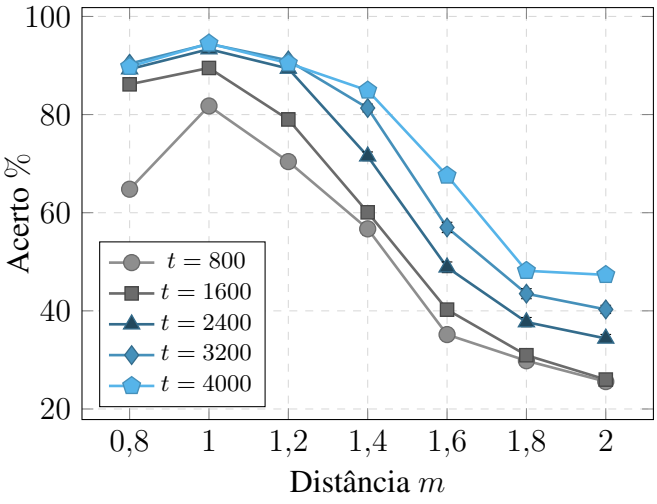
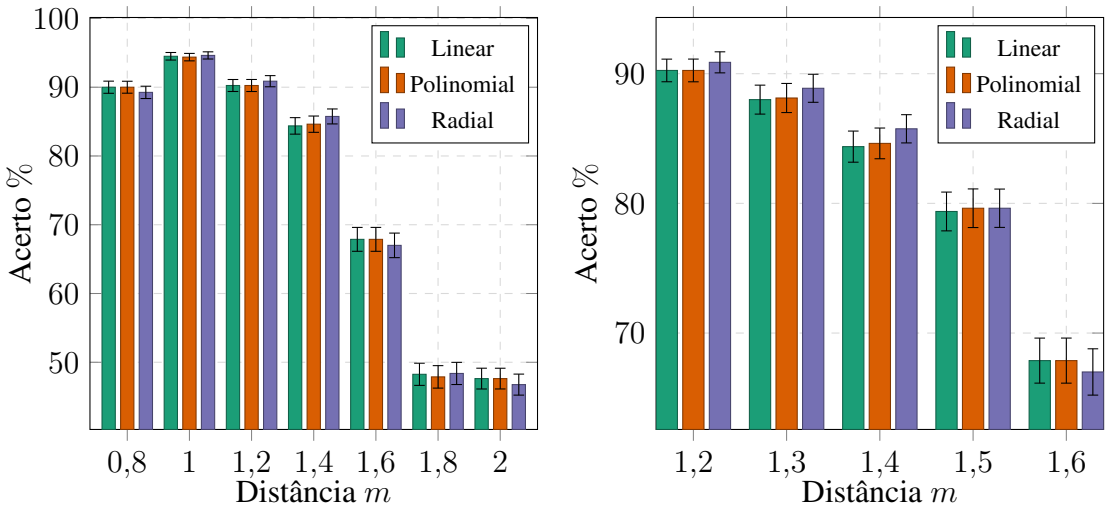



**Figura 3. Percentuais médios de acerto dos *kernels* SVM em relação à quantidade de treinamento *t* realizada.**

Os resultados individuais de cada *kernel* são observados na Figura 4a, que demonstra a ocorrência na distância de $1m$ dos maiores percentuais médios de acerto em relação às demais distâncias, com o *kernel* linear apresentando o valor de $94,5\% \pm 0,5496\%$, o polinomial, $94,375\% \pm 0,5417\%$ e o radial, $94,625\% \pm 0,5221\%$, considerando um intervalo de confiança (IC) de 95%. Comparativamente, os *kernels* demonstram grande proximidade entre seus resultados percentuais de acerto, nesta e em outras distâncias, não sendo possível determinar a superioridade de um em relação aos demais por este critério. É importante destacar que estes resultados, utilizando apenas sensores de profundidade na captura, são semelhantes aos valores obtidos com câmeras RGB por [de Souza et al. 2012a] e com ambos (RGB-D) em [Zhu e Wong 2012].

A ocorrência dos melhores resultados na distância de $1m$ tem como principal motivo a composição do vocabulário visual por amostras capturadas nesta distância. Como resultado, descritores capturados na mesma posição possuem maior semelhança aos utilizados nos histogramas representativos de cada letra, em relação às capturas realizadas em outras distâncias. O critério adotado para este procedimento, conforme citado na seção 3, envolve a preferência por uma posição mais próxima ao dispositivo, proporcionando a extração de descritores mais precisos, em uma imagem com maior resolução. A distância de $0,8m$ foi considerada inadequada devido à observação de uma maior ocorrência de ruído, ocasionado por tratar-se da limitação prática do *near-mode* do dispositivo.

A investigação mais aprofundada da diminuição do percentual médio de acertos entre as distâncias $1,2m$ e $1,6m$, apresentada na Figura 4b, verificou a relação ao comportamento observado após $1,5m$ por [Zhu e Wong 2012] em descritores de profundidade obtidos em captura com o Kinect. Por meio de validações adicionais nas distâncias de $1,3m$ e $1,5m$, seguindo os mesmos critérios adotados anteriormente nas demais validações, demonstrou a diminuição gradual da precisão do protótipo até $1,5m$, seguida de uma queda brusca de aproximadamente $11,5\%$, para o *kernel* linear, $11,75\%$, para o polinomial, e $12,625\%$ para o radial.



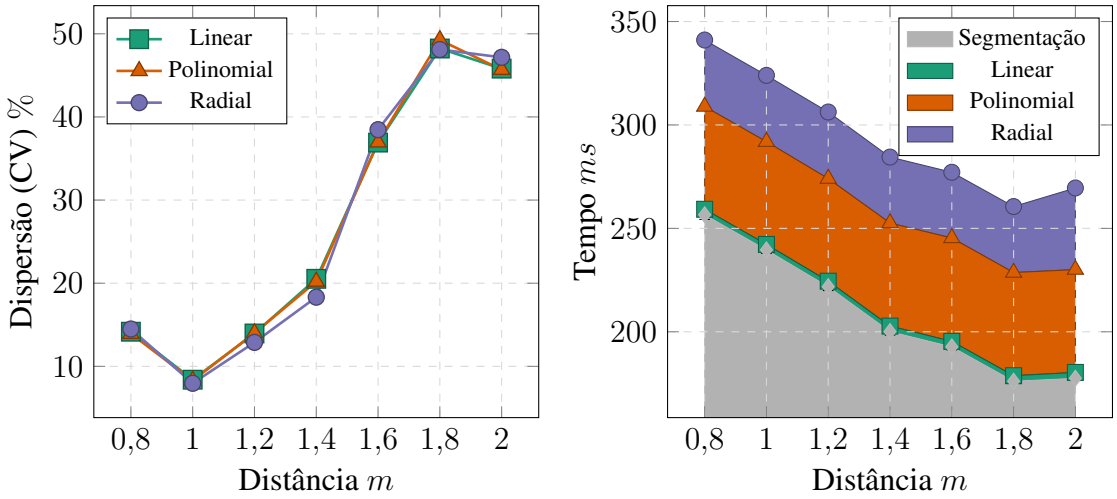

(a) Validação de 0,8*m* à 2*m*.

(b) Validação de 1,2*m* à 1,6*m*.

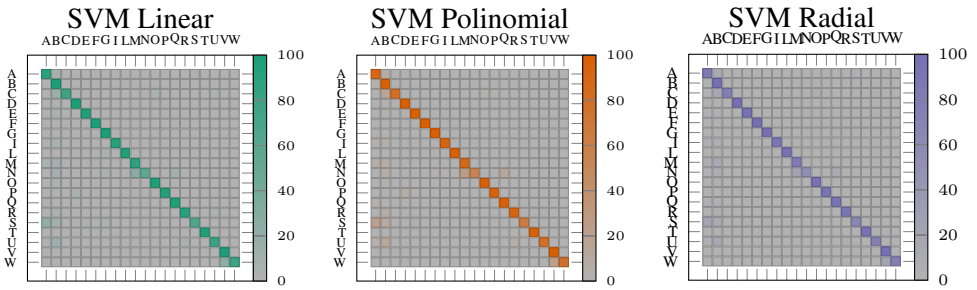
**Figura 4. Percentuais médios de acerto por *kernel* SVM em validação, considerando IC de 95%.**

Ao investigar a dispersão dos percentuais de acerto através do coeficiente de variação (CV), é observada uma maior homogeneidade na distância $1m$, com o *kernel* linear apresentando $8,3928\%$, o polinomial $8,2827\%$ e o radial $7,9619\%$. A Figura 5a permite afirmar que, entre as distâncias até $1,4m$ e independentemente do *kernel*, os resultados apresentam-se de forma homogênea, com dispersão de até 20%, em relação à heterogeneidade devido à queda brusca de qualidade na captura a partir de $1,6m$. As ocorrências de erros no reconhecimento de cada letra na distância de $1m$, apresentadas na Figura 5c, demonstram confusão especialmente entre as letras N e M, e, em menor intensidade, entre as letras A e S. Este problema ocorre pela semelhança entre estes gestos, com diferença apenas na posição de um dedo que exige uma captura de maior precisão.

Observando os resultados das medidas de desempenho por tempo de execução, apresentado na Figura 5b, é verificada a distinção entre o tempo decorrido para classificação por cada *kernel*, com a implementação linear tomando apenas $2,6681 \pm 0,0026ms$, em relação aos $52,4260 \pm 0,0088ms$ e $85,4959 \pm 0,0702ms$ (IC de 95%) das implementações polinomial e radial, respectivamente. As variações no tempo total do processo de reconhecimento ocorrem no tempo de segmentação, devido relação da proximidade com a quantidade de características segmentadas e extraídas no *frame*, diminuindo gradualmente com o aumento da distância.




(a) Dispersão (CV) por *kernel* SVM entre 0,8*m* e 2*m* (b) Tempo de execução por etapas entre 0,8*m* e 2*m*




(c) Matrizes de confusão por *kernel* SVM dos resultados à 1*m* do dispositivo.

**Figura 5. Análise de dispersão (CV), tempo de execução e matrizes de confusão por *kernel* com ocorrências entre classes na distância de 1*m*.**

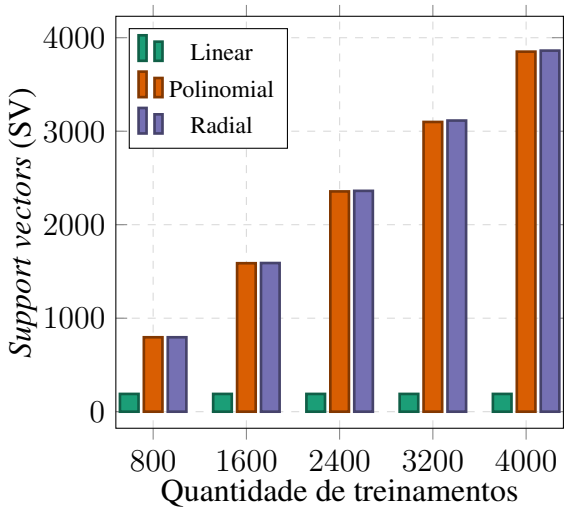



**Figura 6. *Support vectors* (SV) por *kernel* SVM após treinamento e otimização.**

Considerando as diferenças de desempenho observadas entre os três *kernels* quanto ao tempo de execução do processo de classificação, optou-se pela verificação da quantidade de *Support Vectores* (SVs) encontrados após os processos de treinamento e otimização realizados, existentes para a implementação SVM da biblioteca OpenCV. Como apontado por [de Souza et al. 2012a], a quantidade de SVs impacta no desempenho do modelo, podendo não produzir uma boa generalização em quantidade insuficiente, ou *overfitting* se em excesso. Baseando-se nisto, é possível observar na Figura 6 que o *kernel* linear manteve sua quantidade de SVs (190) constante, após sucessivas adições na quantidade de treinamentos. Já para os *kernels* não-lineares há um aumento progressivo de seus SVs, com uma pequena e crescente diferença entre ambos, a ponto de, na última adição de treinamentos, o *kernel* radial possuir 11 SVs adicionais em relação ao polinomial. Estas diferenças indicam a otimização obtida em cada SVM, justificando o melhor desempenho em tempo de execução do *kernel* linear em relação aos não-lineares.

## 5. Conclusão

O reconhecimento de sinais estáticos de LIBRAS proposto foi implementado utilizando *Support Vector Machines* (SVM) para classificação de descritores extraídos após segmentação da captura de profundidade do Kinect. As avaliações realizadas com o protótipo apresentaram maiores percentuais médios de acerto na distância de $1m$, com $94,5\% \pm 0,5496\%$ para o *kernel* linear, $94,375\% \pm 0,5417\%$ para o polinomial e $94,625\% \pm 0,5221\%$ para o radial, considerando um intervalo de confiança (IC) de 95%. Estes valores percentuais não são suficientemente distintos para determinar a abordagem SVM com maior precisão, sendo então investigado o desempenho em tempo de execução de classificação. Nesta perspectiva, a implementação linear obteve a maior eficiência considerando seus resultados alcançados com tempo de execução consumindo em média $2,6681 \pm 0,0026ms$ em relação aos $52,4260 \pm 0,0088ms$ e $85,4959 \pm 0,0702ms$ das implementações polinomial e radial, respectivamente. Este melhor desempenho pode ser justificado através de uma generalização otimizada obtida em treinamento, com apenas 190 *Support Vectors* (SVs) em relação aos respectivos 3851 SVs e 3862 SVs de ambas implementações não-lineares.

Através da avaliação do protótipo considerando as distâncias do usuário em relação ao dispositivo, foi possível reproduzir a queda drástica em percentuais de acerto após $1,5m$ observada por [Zhu e Wong 2012], ocorrida devido à baixa resolução da câmera de profundidade do Kinect. Já a superioridade dos resultados apresentados na distância de $1m$, em comparação às demais posições, justifica-se pela sua utilização na captura de amostras para composição dos histogramas de descritores do vocabulário visual. É importante destacar que os valores obtidos nesta distância são semelhantes aos resultados encontrados em [de Souza et al. 2012a] e [Zhu e Wong 2012], considerando que o primeiro utiliza câmera RGB, e o segundo RGB-D do Kinect, enquanto este estudo apenas utilizou a captura do sensor de profundidade.

Por fim, sugere-se uma investigação deste protótipo com participação de voluntários, a fim de observar seu desempenho utilizando amostras com mãos de diferentes formatos. Outra sugestão consiste na experimentação com outros dispositivos, como o Kinect 2, em busca de uma maior precisão na captura, possivelmente superando as limitações verificadas no Kinect neste estudo. Também recomenda-se experimentar novas abordagens, utilizando outras técnicas de segmentação, extração e classificação a fim de observar possíveis vantagens para precisão e desempenho em sua utilização.

## Referências

Almeida, S. G. M., Guimarães, F. G., e Ramírez, J. A. (2014). Feature extraction in Brazilian Sign Language Recognition based on phonological structure and using RGB-D sensors. *Expert Systems with Applications*, 41(16):7259–7271.

Brito, L. F. (1995). *Por uma Gramática de Línguas de Sinais*. Tempo Brasileiro, 1 edition.

Canny, J. (1986). A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–698.

Carneiro, A. T. S., Cortez, P. C., e Costa, R. C. S. (2009). Reconhecimento de Gestos da LIBRAS com Classificadores Neurais a partir dos Momentos Invariantes de Hu. In *Anais do 1º Congresso Regional de Design de Interação - Interaction South America 09*, pages 193–198.

Cruz, L., Lucio, D., e L.Velho (2012). Kinect and RGBD Images: Challenges and Applications. In *2012 25th SIBGRAPI Conference on Graphics, Patterns and Images Tutorials (SIBGRAPI-T)*, pages 36–49.

Csurka, G., Dance, C. R., Fan, L., Willamowski, J., e Bray, C. (2004). Visual categorization with bags of keypoints. In *In Workshop on Statistical Learning in Computer Vision, ECCV*, pages 1–22.

de Quadros, R. M. (1997). *Educação de surdos - A aquisição da linguagem*. Artmed.

de Souza, C. R. e Pizzolato, E. B. (2013). Sign Language Recognition with Support Vector Machines and Hidden Conditional Random Fields: Going from Fingerspelling to Natural Articulated Words. In *Proceedings of the 9th International Conference on Machine Learning and Data Mining in Pattern Recognition (MLDM 2013)*, pages 84–98. Springer.

de Souza, C. R., Pizzolato, E. B., e dos Santos Anjo, M. (2012a). Fingerspelling Recognition with Support Vector Machines and Hidden Conditional Random Fields: A

Comparison with Neural Networks and Hidden Markov Models. In *Proceedings of the 13th Ibero-American Conference on Artificial Intelligence (IBERAMIA'12)*, pages 561–570. Springer.

de Souza, C. R., Pizzolato, E. B., e dos Santos Anjo, M. (2012b). Recognizing Static Signs from the Brazilian Sign Language: Comparing Large-Margin Decision Directed Acyclic Graphs, Voting Support Vector Machines and Artificial Neural Networks.

Dong, C., Leu, M. C., e Yin, Z. (2015). American Sign Language Alphabet Recognition Using Microsoft Kinect. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 44–52.

FADERS (2010). *Mini dicionário*. Fundação de Articulação and Desenvolvimento de Políticas Públicas para Pessoas com Deficiências and Altas Habilidades no Rio Grande do Sul (FADERS).

Jiang, Y.-G., Ngo, C.-W., e Yang, J. (2007). Towards optimal bag-of-features for object categorization and semantic video retrieval. In *Proceedings of the 6th ACM International Conference on Image and Video Retrieval*, CIVR '07, pages 494–501. ACM.

Johnson, G. P., Abram, G. D., Westing, B., Navrátil, P., e Gaither, K. (2012). DisplayCluster: An Interactive Visualization Environment for Tiled Displays. In *2012 IEEE International Conference on Cluster Computing (CLUSTER)*, pages 239–247.

Microsoft (2015). Kinect for Windows Programming Guide.

Mitra, S. e Acharya, T. (2007). Gesture Recognition: A Survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 37(3):311–324.

OpenCV (2011). *OpenCV 2.4 documentation*. OpenCV Foundation.

Otsu, N. (1979). A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man and Cybernetics*, 9(1):62–66.

Pedersoli, F., Benini, S., Adami, N., e Leonardi, R. (2014). XKin: an open source framework for hand pose and gesture recognition using kinect. In *The Visual Computer*, volume 30, pages 1107–1122. Springer.

Pizzolato, E. B., dos Santos Anjo, M., e Pedroso, G. (2010). Automatic Recognition of Finger Spelling for LIBRAS Based on a Two-Layer Architecture. In *Proceedings of the 2010 ACM Symposium on Applied Computing (SAC '10)*, pages 969–973.

Rioux-Maldague, L. e Giguere, P. (2014). Sign Language Fingerspelling Classification from Depth and Color Images Using a Deep Belief Network. In *2014 Canadian Conference on Computer and Robot Vision (CRV)*, pages 92–97.

Zafrulla, Z., Brashear, H., Starner, T., Hamilton, H., e Presti, P. (2011). American Sign Language Recognition with the Kinect. In *Proceedings of the 13th International Conference on Multimodal Interaction (ICMI'11)*, pages 279–286.

Zhang, Z. (2012). Microsoft Kinect Sensor and Its Effect. *IEEE MultiMedia*, 19(2):4–12.

Zhu, X. e Wong, K. K. (2012). Single-frame hand gesture recognition using color and depth kernel descriptors. In *Proceedings of the 21st International Conference on Pattern Recognition (ICPR)*, pages 2989–2992. IEEE.