

Detecção e Rastreamento de um Alvo no Campo Visual de um Robô Autônomo Usando um Sensor RGB-D

Leonardo C. Ribeiro¹, Eduardo Max A. Amaral²

¹Instituto de Informática – Instituto Federal do Espírito Santo (IFES)
Caixa Postal 15.064 – 91.501-970 – Espírito Santo – ES – Brasil

leo.camilo.ribeiro@gmail.com, emaxamaral@gmail.com

Abstract. *Studies in robotics's field have become commonplace in the scientific environment, due this, innumerable activities in society, carried out by people, today can be realized by robots, such as the security and surveillance area due to its high unhealthiness and propensity for human failure. In this work, was developed a system to detect and trace a target object in the visual field of a stand-alone robot using an RGB-D sensor for applications in surveillance systems. The proposed system operates in four stages. Pre-processing, segmentation, association, and classification of the 3D point cloud. Experimental results showed that the proposed system was able to perform the proposed task with good performance.*

Resumo. *Estudos na área da robótica se tornaram comuns no meio científico, por conta disso, inúmeras atividades na sociedade, exercidas por pessoas, hoje podem ser realizadas por robôs, como por exemplo a área de segurança e vigilância devido sua alta insalubridade e propensão à falha humana. Neste trabalho, foi desenvolvido um sistema de detecção e rastreamento de um objeto alvo no campo visual de um robô autônomo usando um sensor RGB-D para aplicações em sistemas de patrulhamento. O sistema proposto opera em quatro etapas. O pré-processamento, a segmentação, a associação e a classificação da nuvem de pontos 3D. Resultados experimentais mostraram que o sistema proposto foi capaz de realizar com bom desempenho a tarefa proposta.*

1. Introdução

Com a ampla divulgação da tecnologia no cotidiano, sensores e peças eletromecânicas vem se tornando algo cada vez mais de baixo custo e de fácil acesso. Por conta disso, houve uma crescente dedicação à pesquisa de soluções tecnológicas para melhorar a qualidade de vida das pessoas por meio da robótica. Segundo [Sebastian Thrun 2005], robótica é a ciência de perceber e manipular o mundo físico através de dispositivos controlados por computadores. O recente progresso da pesquisa em robótica avançada permite a aplicação de soluções em áreas nunca antes imaginadas, a qual a ideia da realização de uma tarefa por um sistema eletrônico era a pouco tempo impensável, como por exemplo o ato de direção de um veículo.

O uso de robôs autônomos, sejam eles por software, hardware, ou ambos, se faz cada vez mais necessário em áreas críticas da sociedade, visto que mesmo após anos de evolução, ainda existem tarefas extremamente insalubres, e uma dessas áreas é a da segurança, por se tratar de uma área demasiadamente insalubre. Por conta disso, se faz

cada vez mais necessário a utilização de robôs e sistemas automatizados nessa área, salvando assim a integridade de quem está disposto a defender a segurança dos outros.

Diversas atividades já são exercidas por robôs nessa área, como por exemplo desarmadores de bombas, drones (Veículo aéreo não tripulado) de reconhecimento, etc. e uma possibilidade que se torna cada vez mais relevante é a de perseguição autônoma, a qual robôs automatizados possam realizar perseguições de alvos sem que seja necessário alguém estar em risco para isso.

Neste contexto, talvez seja possível a criação de mecanismos automatizados de perseguição de alvo a serem implementados em robôs de patrulha, de modo que não seja necessária a interferência humana. Desta forma, o objetivo desse trabalho é a implementação de um sistema de reconhecimento e perseguição de um alvo em um sistema robótico autônomo a ser aplicado em sistemas de patrulhamento.

2. Referencial Teórico

Para realizar a tarefa de perseguição nesse trabalho, utilizamos o conceito de DATMO (Detection And Tracking of Moving Objects), que por sua vez se trata de métodos de detecção e rastreamento de objetos móveis utilizando dados de sensores. A necessidade de se automatizar tarefas deste tipo não é uma ideia nova. Segundo [Petrovskaya et al. 2012], a necessidade de mecanismos de DATMO nasceu por volta de 1980 juntamente com o interesse em veículos inteligentes e autônomos. Um grande número de projetos exploratórios foram desenvolvidos na Europa em 1986 sobre o nome de PROMETHEUS (*PROgramMe for a European Traffic of Highest Efficiency and Unprecedented Safety*), seguido por iniciativas no Japão e nos Estados Unidos.

Existem inúmeras aplicações diferentes para o problema de DATMO, desde a automatização de um robô de chão para ambientes internos e movimentados [Gritti et al. 2014], como a automatização de um carro [Thrun et al. 2006], onde uma equipe da universidade de Stanford desenvolveu um carro autônomo.

Porém, independente do contexto em que é aplicado, no que se refere a trabalhos da área de DATMO, grande parte da literatura [AMARAL et al. 2015] utiliza o mesmo fluxo de processamento para realização da tarefa, visto na Figura 1, métodos de segmentação, detecção ou classificação, e por fim, associação e rastreamento. Neste trabalho iremos utilizar alguns desse métodos para realizar a perseguição de um alvo, no alcance visual do Kinect, por um robô autônomo.

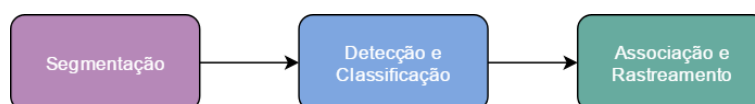


Figura 1. Fluxo de processamento.

A etapa de segmentação visa agrupar sub-conjuntos de dados pertencentes a um conjunto maior em agrupamentos distintos, de modo que todos os pontos referentes ao um mesmo objeto sejam segmentados juntos. É comumente utilizado o método de distância limiar para realizar a segmentação em um conjunto de dados de pontos tridimensionais. É uma parte essencial para o processo, possibilitando assim, que seja

possível realizar as etapas seguintes, possui alguns métodos conhecidos como por exemplo o DBSCAN (*Density-Based Spatial Clustering of Applications with Noise*) definido em [Ester et al. 1996] e o JDC (*Jump Distance Clustering*) [Spinello et al. 2010].

A etapa de associação e rastreamento tem como função informar onde o objeto está na cena, tanto espacialmente, como temporalmente. Ou seja, possui a tarefa de associar um objeto o_1 no tempo t_n com outro objeto o_2 no tempo t_{n+1} e confirmar com determinado grau de certeza que o_1 e o_2 são a representação do mesmo objeto captado pelo sensor em instantes de tempo t_i diferentes. Possui alguns métodos conhecidos como JPDA (*Joint Probabilistic Data Association*) e MHT (*Multiple Hypothesis Tracking*) ambos encontrados em [Cox 1993] e [Blackman 2004].

A etapa de descrição e classificação visa identificar um grupo de características pertencentes a um conjunto de dados, que sejam capazes de diferenciar esse conjunto de outro, ou dizer que dois conjuntos são semelhantes. Em uma nuvem de pontos, esse processo se dá através de uma análise de um conjunto de pontos no espaço, primeiro é realizada a etapa de identificação das características, realizada por meio de métodos descritores. Métodos descritores são técnicas que reconhecem um conjunto de características de um modelo observável, de modo que seja possível diferenciá-los de outros modelos através de suas características, comparamos as características de um objeto com o conjunto de características que temos de outros modelos, utilizados para treinar o classificador. Assim, é possível classificar o objeto observado, de acordo com o nível de semelhança de características entre o objeto e os modelos de treino.

Existem inúmeros métodos de descrição, tanto para imagens bidimensionais, como por exemplo o SIFT (*Scale Invariant Feature Transform*) [Lowe 1999], como também para imagens tridimensionais, como CSH (*Complex Shape Histogram*) [Będkowski et al. 2016], PFH (*Point Feature histogram*) [Rusu et al. 2008], PPF (*Point Pair Feature*) [Drost et al. 2010]. Porém, a escolha de qual utilizar varia de uma aplicação para outra, visto que cada um possui suas próprias peculiaridades.

3. Desenvolvimento

O sistema proposto desenvolvido neste trabalho opera em quatro etapas. A cada leitura do sensor é realizado o pré-processamento, a segmentação, a associação e a classificação da nuvem de pontos 3D. Na etapa de pré-processamento ocorre a remoção dos pontos 3D do chão e a redução do conjunto de amostragem. Na etapa de segmentação, a nuvem de pontos 3D é segmentada em agrupamentos de pontos usando a distância Euclidiana, sendo que cada agrupamento representa um objeto no ambiente. Na etapa de associação, os objetos observados na varredura atual do sensor são associados aos mesmos objetos observados em varreduras anteriores usando o algoritmo do vizinho mais próximo (*nearest neighbor*). Finalmente, na etapa de classificação, os segmentos são comparados com os modelos predefinidos e após a identificação, o alvo é rastreado. O fluxo de funcionamento do sistema proposto pode ser visualizado na Figura 2.

3.1. Pré-processamento

Na etapa de pré-processamento é realizado a modelagem e filtragem do dado, tornando possível as etapas posteriores. Neste trabalho a etapa de pré-processamento é dividida em duas etapas, remoção do Plano do chão e redução de amostragem do conjunto de dados.

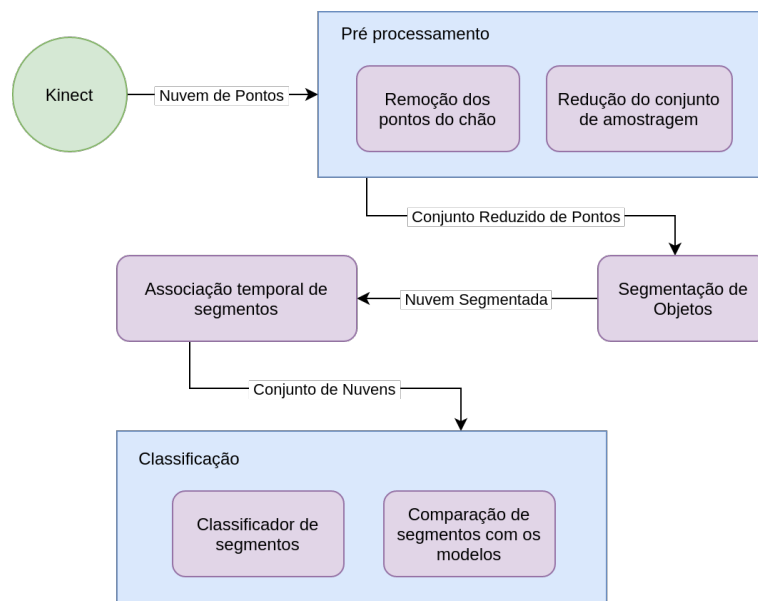


Figura 2. Fluxo de funcionamento do sistema proposto.

O chão é geralmente o plano dominante na maioria das cenas observáveis por um robô terrestre. Como demonstrado por [Douillard et al. 2011], a extração do solo melhora significativamente o desempenho da segmentação. A não remoção do chão impossibilita a segmentação de pontos pois é considerado como parte do segmento. A etapa de redução de amostragem do conjunto é opcional, porém se fez necessária devido ao fato de ser essencial a execução do sistema em tempo real, como demonstrado na Figura 3.

Neste trabalho o *RANSAC* foi utilizado como pré-processamento para a remoção do chão. ***Random Sample Consensus (RANSAC)*** é um método iterativo para estimar parâmetros de um modelo matemático em um conjunto de dados observáveis. O algoritmo foi primeiramente publicado em [Fischler and Bolles 1981].

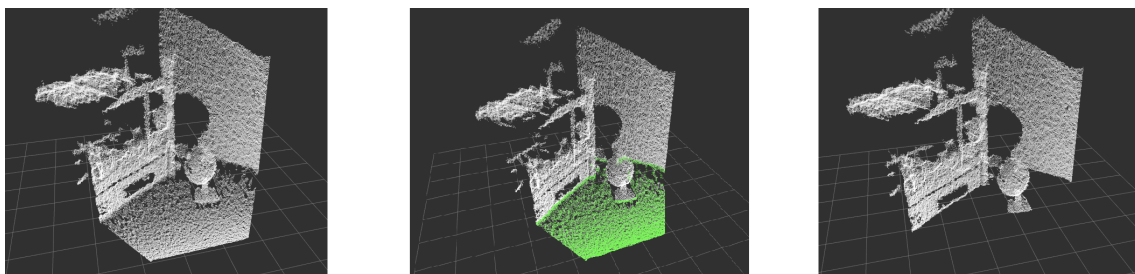


Figura 3. Aplicação do RANSAC para remoção do chão.

Para realizar a redução de amostragem foi utilizado o *VoxelGrid*, se trata de um método de filtro para nuvens de pontos, onde se é possível diminuir o conjunto de amostragem de forma significativa, mantendo consideravelmente a semelhança do conjunto filtrado com o original. O método de *VoxelGrid* possui implementação na PCL [Rusu and Cousins 2011]. O Filtro *VoxelGrid*, ou grade de voxel é um método que discretiza os pontos de uma nuvem dentro de um intervalo contínuo, desta forma, é gerado um conjunto muito menor de dados para processar, conforme visto na Figura 4.

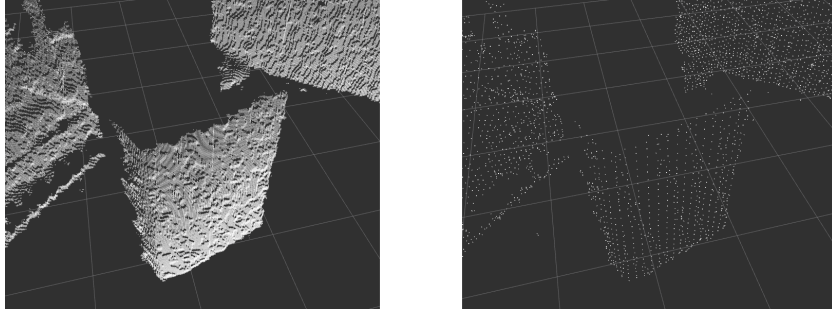


Figura 4. Aplicação do *VoxelGrid* na nuvem de pontos.

3.2. Segmentação

A segmentação é uma etapa de extrema importância, pois, a partir dessa etapa, é possível extrair subconjuntos de dados do conjunto principal, onde, os dados desses subconjuntos possuem características semelhantes. Tais características a serem levadas em conta são determinadas pela segmentação, tornando um grande e desconhecido conjunto de dados em inúmeros conjuntos menores, os quais podem ser mais facilmente analisados. Esta etapa é essencial, como é demonstrado em [Boucher 2012]. Outra afirmação verdadeira é a de que uma segmentação a priori melhora a classificação em aplicações de visão computacional, e isso é demonstrado em [Malisiewicz and Efros 2007].

Para a segmentação de objetos neste trabalho foi utilizada uma abordagem simples, porém eficiente de agrupamento por distância Euclidiana. Essa abordagem é utilizada em inúmeros trabalhos como em [Boucher 2012] e [AMARAL et al. 2015], e demonstra ótimos resultados quanto ao objetivo de agrupar pontos pertencentes ao mesmo objeto no campo visual do sensor. Na Figura 5 é visto objetos representados por cores diferentes, demonstrando que foi possível evidenciar cada um dos objetos na cena.

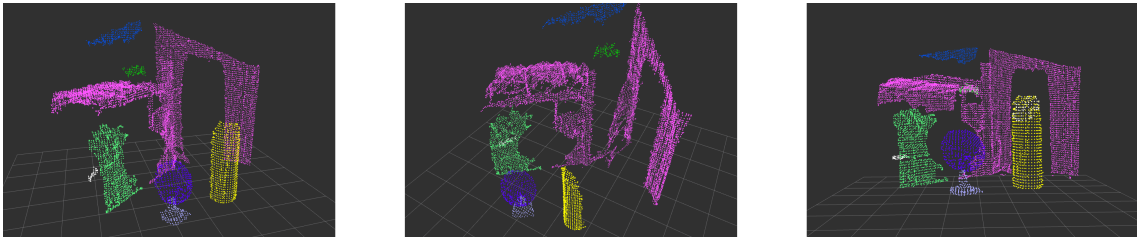


Figura 5. Aplicação da segmentação na nuvem de pontos.

3.3. Associação

A associação por vizinhos próximos possui aplicação em inúmeros trabalhos da área, como [Becker et al. 2009], [Vu et al. 2007], [Azim and Aycard 2012], e demonstra resultados satisfatórios na execução da tarefa.

Neste trabalho foi utilizada uma abordagem pelo centro de massa para realizar a associação por vizinhos próximos. Para cada segmento de nuvem da cena $sn_k \subset N$, é obtido o centro de massa $cm_k = \{x_k, y_k, z_k\}$ do mesmo. A partir disso, é então definido um conjunto de nuvens associadas $na = \emptyset$, a cada leitura do sensor, uma verificação é realizada, e o centro de massa cm_k de cada segmento $sn_k \subset N$ é comparado com

os centros de massa dos objetos $o_j \in na$, procurando-se o vizinho mais próximo. A proximidade de vizinhança é definida como a distância Euclidiana dos centros de massa cm_k do segmento sn_k e do objeto o_j . Caso a distância entre ambos seja menor que um círculo com raio $R < limiar$, onde o *limiar* é um valor mínimo de distância predefinido por nós, o elemento sn_k substitui o objeto $o_j \in na$, atualizando o conjunto de dados pertencentes ao mesmo, como por exemplo, nuvem de pontos e centro de massa. Caso nenhum elemento do conjunto na seja associado ao segmento sn_k , o mesmo é adicionado ao conjunto na como novo elemento.

3.4. Classificação

Para essa aplicação foi utilizado o modelo de classificação VFH (*Viewpoint Feature Histogram*) [Rusu et al. 2010] ou Histograma de característica de ponto de vista, foi criado em 2010, e se trata de um descritor para nuvens de pontos tridimensional, capaz de classificar informações de geometria e de ponto de vista de uma nuvem de pontos. Foi desenvolvido para uma aplicação robótica de manipulação de objetos por um braço robótico, e portanto, possui um excelente funcionamento em tempo real.

Para um funcionamento relativamente bom, assume-se que certas etapas de pré-processamento sejam feitas, como a identificação e remoção de planos, a segmentação dos pontos restantes em um espaço Euclidiano, e o cálculo de normais de cada ponto. Para o cálculo da normal de um ponto, primeiro é inferido uma superfície para cada ponto p_i e seus p_i^k vizinhos, dada então a superfície, é calculado o vetor normal \vec{n} da mesma. É então verificado se a orientação do vetor normal \vec{n} do ponto p_i consiste com a orientação do ponto de visão Pv do observador, caso não esteja o sentido do vetor é invertido. Por se tratar de uma estimativa, não pode-se assumir o vetor estimado como verdade absoluta, porém o método atinge resultados satisfatórios, segundo os resultados demonstrados em [Holzer et al. 2012].

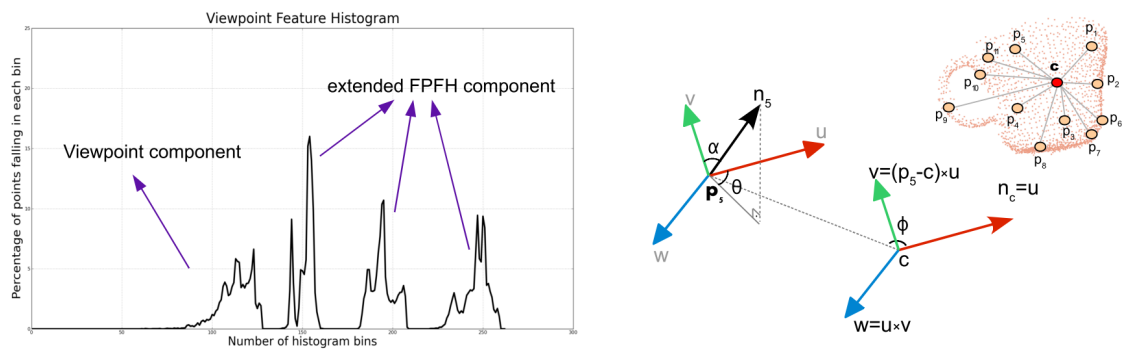


Figura 6. Modelo VFH de descrição da nuvem de pontos.

A ideia do VFH é semelhante com a dos trabalhos anteriores [Rusu et al. 2009] e [Rusu 2010], porém adicionando um componente de ponto de vista ao histograma padrão. O VFH utiliza a metodologia do FPFH [Rusu et al. 2009] com um histograma que coleta um conjunto de informações angulares entre pares de normais na nuvem (Figura 6). Especificamente, para cada par de pontos $\langle p_i, p_j \rangle$ e suas respectivas normais $\langle \vec{n}_i, \vec{n}_j \rangle$, o seu conjunto de divergências angulares pode ser definido como:

$$\alpha = v \cdot \vec{n}_j \quad \phi = \frac{(p_j - p_i)}{d} \quad \theta = \arctan(w \cdot \vec{n}_j, u \cdot \vec{n}_j)$$

Onde u, v, w representam um sistema de coordenadas *Darboux frame* no ponto p_i , esse sistema de coordenadas possui três vetores ortonormais centrados no ponto. Uma melhor definição de como funciona o *Darboux frame* e como é realizado a extração dessas informações de uma nuvem de pontos podem ser encontradas em [Hameiri and Shimshoni 2003].

Após isso adicionamos inúmeras amostras e seus respectivos histogramas em diferentes ângulos de visão, conforme visto na Figura 7. Essa será a base de conhecimento do classificador, e após isso, cada elemento no campo de visão do sensor será comparado ao conjunto de histogramas na base de conhecimento através de uma classificação por KNN. Para o cálculo de distância de histogramas foi utilizado a distância quadrada de Chi [Pele and Werman 2010], definida pela seguinte fórmula:

$$\frac{1}{2} \sum_{i=1}^n \frac{(x_i - y_i)^2}{(x_i + y_i)}$$

Onde n é o numero de *bins* (intervalos do eixo x) do histograma e x_i e y_i os valores dos histogramas no índice i . Essa fórmula é utilizada para medir a distância quadrática entre dois histogramas. Ou seja, para realizar uma análise de correspondência entre ambos, obtendo-se então um índice de similaridade, onde quanto menor o valor, mais semelhantes são os histogramas comparados.

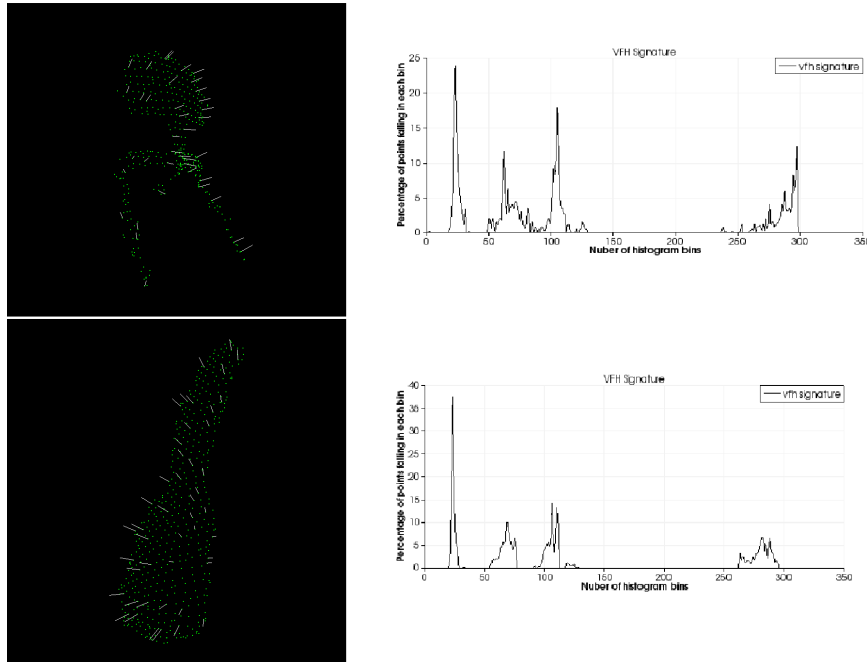


Figura 7. Aplicação do VFH em alguns segmentos de nuvens.

4. Resultados

Para realizar a avaliação quantitativa do método de classificação proposto neste trabalho, são utilizados dois índices avaliativos: o índice de precisão e o índice de revocação. O índice de precisão é expresso por uma relação entre a quantidade de acertos (verdadeiros positivos) e a quantidade total de classificações realizadas. O índice de revocação, por outro lado, expressa a relação entre a quantidade de classificações corretamente realizadas de um objeto na cena e a quantidade real de amostragem desse mesmo objeto. Nesse trabalho uma classificação correta é definida como um verdadeiro positivo (vp), uma classificação errada como falso positivo (fp), e uma não classificação como um falso negativo (fn). Ambos os índices possuem resultado entre zero e um, indicando a porcentagem de precisão e são definidos como:

$$precisão = \frac{vp}{vp + fp} \qquad revocação = \frac{vp}{vp + fn}$$

Para realizar a contagem de verdadeiros positivos vp , de falsos positivos fp e de falsos negativos fn , foi observado o histograma gerado a partir da variação das coordenadas (x, y, z) do objeto classificado durante um período de teste. Quando um objeto (diferente do objeto alvo) é classificado erroneamente, ocorre uma mudança repentina no centro de massa do objeto classificado. Essas ocorrências (mudanças abruptas da coordenada do centro de massa do objeto) são classificadas como fp . Todos os outros valores de coordenadas de objetos classificados são quantificados como vp . E, em todos os *frames* que não houve classificação são contabilizados como fn . Durante a avaliação foram quantificados 681 verdadeiros positivos, 388 falsos negativos e 13 falsos positivos.

Tabela 1. Dados obtidos dos testes.

Resultados dos testes		
Leituras = 1082	Classificadas = 694	681 vp
		13 fp
	Não classificadas = 388	388 fn

Desta forma, obteve-se uma taxa de precisão de 98,12% na classificação. Para a taxa de revocação obteve-se um índice de 63,70%. Os resultados da precisão e revocação podem ser visualizados na Tabela 2.

Tabela 2. Índices avaliativos da classificação.

Resultados dos testes	
precisão	98,12%
revocação	63,70%

Após análise dos experimentos, pode-se verificar que o sistema apresentou boa eficiência em detectar e rastrear um objeto alvo através de dados de uma nuvem de pontos, sendo capaz de processar as informações obtidas com um bom desempenho de execução, podendo ser aplicado em sistemas de tempo real.

Referências

- AMARAL, E. M. A., BADUE, C., OLIVEIRA-SANTOS, T., and DE SOUZA, A. F. (2015). Detecção e rastreamento de veículos em movimento para automóveis robóticos autônomos. *XII Simpósio Brasileiro de Automação Inteligente (SBAI)*, (2015):801–806.
- Azim, A. and Aycard, O. (2012). Detection, classification and tracking of moving objects in a 3d environment. *Intelligent Vehicles Symposium (IV)*, 2012 IEEE, pages 802–807.
- Becker, M., Hall, R., Kolski, S., Maček, K., Siegwart, R., and Jensen, B. (2009). 2D laser-based probabilistic motion tracking in urban-like environments. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 31(2):83–96.
- Będkowski, J., Majek, K., Majek, P., Musialik, P., Pełka, M., and Nüchter, A. (2016). Intelligent Mobile System for Improving Spatial Design Support and Security Inside Buildings. *Mobile Networks and Applications*, 21(2):313–326.
- Blackman, S. (2004). Multiple hypothesis tracking for multiple target tracking. *IEEE Aerospace and Electronic Systems Magazine*, 19(1):5–18.
- Boucher, S. (2012). Obstacle Detection and Avoidance Using TurtleBot Platform and Xbox Kinect.
- Cox, I. J. (1993). A review of statistical data association techniques for motion correspondence. *International Journal of Computer Vision*, 10(1):53–66.
- Douillard, B., Underwood, J., Kuntz, N., Vlaskine, V., Quadros, A., Morton, P., and Frenkel, A. (2011). On the segmentation of 3D LIDAR point clouds. *2011 IEEE International Conference on Robotics and Automation*, pages 2798–2805.
- Drost, B., Ulrich, M., Navab, N., and Ilic, S. (2010). Model globally, match locally: Efficient and robust 3D object recognition. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, number July 2010, pages 998–1005. IEEE.
- Ester, M., Kriegel, H.-P., Sander, J., and Xu, X. (1996). A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, KDD’96*, pages 226–231. AAAI Press.
- Fischler, M. a. and Bolles, R. C. (1981). Paradigm for Model. *Communications of the ACM*, 24(6):381–395.
- Gritti, A. P., Tarabini, O., Guzzi, J., Di Caro, G. A., Caglioti, V., Gambardella, L. M., and Giusti, A. (2014). Kinect-based people detection and tracking from small-footprint ground robots. *IEEE International Conference on Intelligent Robots and Systems, (Iros)*:4096–4103.
- Hameiri, E. and Shimshoni, I. (2003). Estimating the principal curvatures and the darboux frame from real 3-d range data. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 33(4):626–637.
- Holzer, S., Rusu, R. B., Dixon, M., Gedikli, S., and Navab, N. (2012). Real-Time Surface Normal Estimation from Organized Point Cloud Data Using Integral Images. *Computer*, pages 2684–2689.

- Lowe, D. G. (1999). Object recognition from local scale-invariant features. *Proceedings of the Seventh IEEE International Conference on Computer Vision*, 2(18):1150–1157.
- Malisiewicz, T. and Efros, a. a. (2007). Improving Spatial Support for Objects via Multiple Segmentations. *Proceedings of the British Machine Vision Conference 2007*, pages 55.1–55.10.
- Pele, O. and Werman, M. (2010). The Quadratic-Chi Histogram Distance Family. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 6312 LNCS, pages 749–762.
- Petrovskaya, A., Perrollaz, M., Oliveira, L., Spinello, L., Triebel, R., Makris, A., Yoder, J.-D., Laugier, C., Nunes, U., and Bessiere, P. (2012). Awareness of Road Scene Participants for Autonomous Driving. In *Handbook of Intelligent Vehicles*, pages 1383–1432. Springer London, London.
- Rusu, R. B. (2010). Semantic 3D Object Maps for Everyday Manipulation in Human Living Environments. *KI - Kunstliche Intelligenz*, 24:345–348.
- Rusu, R. B., Blodow, N., and Beetz, M. (2009). Fast Point Feature Histograms (FPFH) for 3D registration. In *2009 IEEE International Conference on Robotics and Automation*, pages 3212–3217. IEEE.
- Rusu, R. B., Bradski, G., Thibaux, R., and Hsu, J. (2010). Fast 3D recognition and pose using the Viewpoint Feature Histogram. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2155–2162. IEEE.
- Rusu, R. B. and Cousins, S. (2011). 3D is here: point cloud library. *IEEE International Conference on Robotics and Automation*, pages 1 – 4.
- Rusu, R. B., Marton, Z. C., Blodow, N., and Beetz, M. (2008). Learning informative point classes for the acquisition of object model maps. In *2008 10th International Conference on Control, Automation, Robotics and Vision*, pages 643–650. IEEE.
- Sebastian Thrun, Wolfram Burgard, D. F. (2005). *Probabilistic Robotics: Intelligent robotics and autonomous agents*. MIT Press.
- Spinello, L., Arras, K. O., Triebel, R., and Siegwart, R. (2010). A layered approach to people detection in 3d range data. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence*, AAAI’10, pages 1625–1630. AAAI Press.
- Thrun, S., Montemerlo, M., Dahlkamp, H., Stavens, D., Aron, A., Diebel, J., Fong, P., Gale, J., Halpenny, M., Hoffmann, G., Lau, K., Oakley, C., Palatucci, M., Pratt, V., Stang, P., Strohband, S., Dupont, C., Jendrossek, L.-E., Koelen, C., Markey, C., Rummel, C., van Niekirk, J., Jensen, E., Alessandrini, P., Bradski, G., Davies, B., Ettinger, S., Kaehler, A., Nefian, A., and Mahoney, P. (2006). Stanley: The robot that won the DARPA Grand Challenge. *Journal of Field Robotics*, 23(9):661–692.
- Vu, T.-D., Aycard, O., and Appenrodt, N. (2007). Online Localization and Mapping with Moving Object Tracking in Dynamic Outdoor Environments. In *2007 IEEE Intelligent Vehicles Symposium*, pages 190–195. IEEE.