

Controle de Veículo a partir da Visão Computacional

Karla A. Justen^{1,2}, Thiago Rateke^{1,2}, Vito F. Chiarella³, Antonio C. Sobieranski¹,
Sylvio L. M. Neto¹, Eros Comunello³, Aldo von Wangenheim^{1,2}

¹Lapix (Laboratório de Processamento de Imagens e Computação Gráfica)
INCoD (Instituto Nacional para Convergência Digital)
Florianópolis – SC – Brasil

²UFSC (Universidade Federal de Santa Catarina)
Florianópolis – SC – Brasil

³4Vision Lab – Universidade do Vale do Itajaí
Florianópolis – SC – Brasil

{justen, thiago, asobieranski, sylvio}@incod.ufsc.br

eros@univali.br, vitochiarella@gmail.com, aldo.vw@ufsc.br

Abstract. *Detect a navigable path and stay on this road is one of the main objectives of an autonomous vehicle. This paper is intended to show the possibility and efficiency of using computer vision and digital image processing techniques for controlling a vehicle using only passive vision, in other words, considering only the image captured by a camera attached to the vehicle. A small testing circuit was built to evaluate the developed work. The test results indicated that it is possible to autonomously maintain the vehicle in simulated road with different types of terrain and variability of conditions in natural daylight but still failures occur and improvements (hardware and software) should be made.*

Resumo. *Detectar um caminho navegável e se manter nesse caminho é um dos principais objetivos de um veículo autônomo. Este trabalho tem a intenção de mostrar a possibilidade e eficácia de se utilizar técnicas de visão computacional e processamento digital de imagem para controlar um veículo utilizando somente visão passiva, ou seja, considerando apenas a imagem capturada por uma câmera fixada ao veículo. Um pequeno circuito de testes foi construído, para avaliar o trabalho. Os resultados dos testes indicaram que é possível manter autonomamente o veículo no caminho simulado, com diferentes tipos de terreno e em condições de variabilidade na iluminação natural, porém ainda ocorrem falhas e melhorias (hardware e software) devem ser feitas.*

1. Introdução

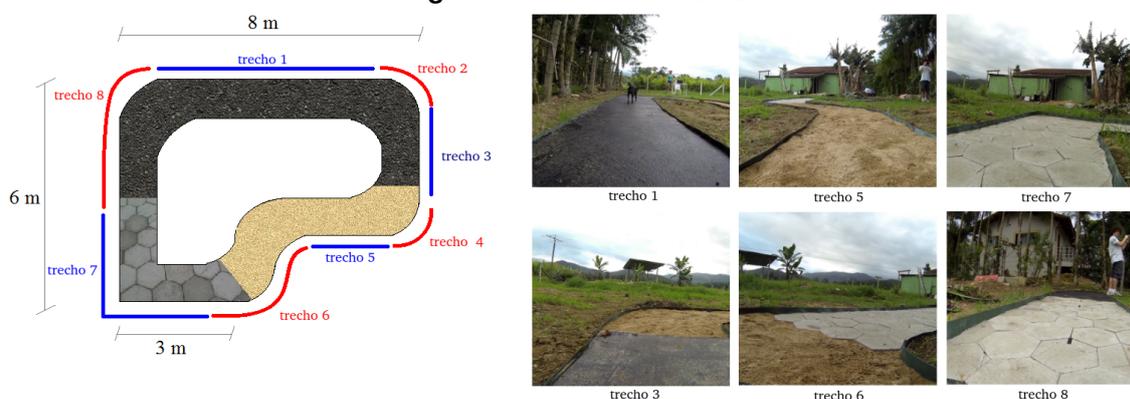
Um sistema de detecção de estrada baseado em Visão Computacional (VC), tem como objetivo possibilitar a detecção da estrada (caminho) com uma fonte de captura que esteja na frente do veículo. É uma tarefa bastante desafiadora, pois lida com cenários abertos e com o veículo em constante movimento, proporcionando assim problemas de iluminação, mudança constante do fundo da cena (*background*), presença de diferentes objetos (veículos, pessoas, etc.), diferentes cenários e tipos de terrenos (urbanos, rodovias, estradas de terra) [Álvarez et al. 2009].

Atualmente os casos de grande destaque na navegação autônoma utilizam diversos sensores, não somente câmera [Urmson et al. 2008], [Urmson 2014b], [Urmson 2014a] e [Fernandes et al. 2014]. Sendo, que um desses sensores, e talvez o principal, é o LIDAR (*Light Detection and Ranging*), que é um laser óptico utilizado para detecção remota de propriedades da luz refletida, de modo a medir distâncias entre o sensor e o objeto alvo [LIDAR 2015]. Nos projetos citados o LIDAR é considerado Class1, que é a categoria de menor impacto e com comprimentos de onda de 905nm. Em [Standard 2001] e [STANDARD 2005] é dito que lasers Class1 quando não tem contato direto com o olho humano durante um período mais extenso não representam perigo à retina. Entretanto, em [Standard 2001] é apresentado uma tabela com a categorização dos lasers e os possíveis riscos com o excesso de exposição à cada um dos níveis. Os lasers com comprimento de onda entre 780nm e 1400nm, que é a faixa de abrangência do LIDAR, podem causar catarata e queimar a retina. Além disso, caso veículos autônomos venham a se tornar uma realidade no dia-a-dia, uma grande quantidade de veículos teria de estar realizando varreduras LIDAR no mesmo local e ao mesmo tempo, causando cruzamentos, reflexos e, conseqüentemente, momentos de cegueira em condutores de veículos não-autônomos.

A visão passiva pode ser uma alternativa nesse cenário futuro e a grande maioria das pesquisas realizadas tem como enfoque um padrão de estrutura no caminho, ou seja, não avaliando situações onde ocorreram alterações bruscas de cor no caminho. Boa parte dos demais, são normalmente testados somente em vídeos gravados, onde um algoritmo de detecção de caminho é forçado a se adaptar, pois o veículo continua seu trajeto independente da detecção falhar em momentos de alteração do terreno ou problemas de iluminação.

Este trabalho tem como objetivo mostrar que é possível controlar um veículo (protótipo) de forma autônoma com base nos resultados da detecção de caminho, com métodos de visão computacional e processamento digital de imagens. A detecção do caminho ocorre em tempo real e de forma adaptativa, ou seja, continua detectando o caminho independente das mudanças nas características do terreno (asfalto, terra e lajotas). O cenário de captura são imagens frontais e horizontais aos veículos. O controle do veículo foi testado em um circuito preparado com alterações nas características do caminho (Figura 1).

Figura 1. Circuito de testes.



Para a realização dos experimentos, foram necessários, além do veículo (Figura 2): uma câmera fixada ao veículo capturando a cena; um notebook que recebeu as imagens para análise e processamento, sendo que as imagens foram transmitidas de forma analógica através de antenas de 900Mhz (uma conectada ao transmissor no veículo e outra acoplada a um receptor conectado ao notebook). Após a análise e processamento das imagens, os dados de controle (direção e aceleração) são enviados ao microcontrolador (Arduino Uno) que, conectado a um rádio controle, passa os comandos ao veículo.

Figura 2. Veículo para experimentos e avaliação do trabalho.



O trabalho está organizado da seguinte forma, na seção 2 é apresentado uma breve fundamentação sobre a Detecção de Caminho e quais métodos utilizados neste trabalho para esse objetivo (Importante salientar que a detecção do caminho em si não é o principal objetivo deste trabalho, pois além de já ter sido publicado essa abordagem, o principal foco aqui é em relação ao controle do veículo. Entretanto, é importante contextualizar e justificar as imagens utilizadas para a análise do controle). A seção 3 mostra como foi realizado o controle do veículo com base nas imagens do caminho detectado, onde os resultados podem ser vistos na seção 4. Por fim, na seção 5, com as conclusões e considerações sobre o trabalho.

2. Detecção de Caminho

Pode-se definir caminho como: rota, trajeto, rua, estrada, ou seja, uma superfície onde os veículos podem transitar. Uma estrada pode ser definida como uma estrutura atravessável, linear e com uma cor que é distinta dos arredores que a envolvem, também deve-se estender por uma certa distância em frente ao veículo e ser suficientemente larga. Ou seja, pode-se utilizar as seguintes propriedades para a detecção de caminho: superfície, borda e direção. Entretanto, a dificuldade na detecção de caminho reside justamente nas variações que essas propriedades podem ter, como por exemplo iluminação (sol, sombra) ou alterações na superfície do caminho [Ososinski and Labrosse 2013].

A detecção de caminho é uma das questões-chave no desenvolvimento de um veículo inteligente, por isso muita pesquisa vem sendo feita neste enfoque, inclusive verificando a viabilidade da utilização de métodos baseados em visão computacional [GUO et al. 2013]. Na extração de características do caminho com base nas imagens, pode-se utilizar técnicas de visão computacional e processamento digital de imagens, mais especificamente, métodos de segmentação, permitindo assim separar toda a região considerada como caminho do restante da imagem.

O trabalho de [Zhou et al. 2010] categoriza os algoritmos de detecção de estrada em três tipos, cujas diferenças entre eles são no que se baseiam para obter o resultado: técnicas baseadas em características; técnicas baseadas em modelo e técnicas baseadas em

regiões. Onde, as baseadas em características são mais eficazes, mas requerem que as estradas tenham marcações bem definidas e de fácil identificação, e ruídos podem interromper todo o processo de detecção (Ex.: [Bai and Wang 2011] e [Marzotto et al. 2010]). Os baseados em modelo são mais robustos, entretanto, tem uma grande restrição justamente pela geometria dos modelos (Ex.: [Álvarez et al. 2009]). Os mais eficazes das técnicas baseadas em regiões podem ser vistos como problemas de aprendizado de máquina. Utilizando o aprendizado de máquina, pode-se lidar com problemas de ruídos e com as constantes alterações no ambiente. Nesse caso a maior dificuldade é realizar o treinamento em tempo real da máquina para que ela consiga realizar o aprendizado do novo ambiente (Ex.: [Chen and Tai 2010], [Dornaika et al. 2011], [Kong et al. 2010], [Gopalan et al. 2012] e [Oniga and Nedeveschi 2010]).

As técnicas baseadas em regiões tratam, mais especificamente, da extração de informações de cor e textura dos pixels na imagem capturada, onde deve-se com base na cor dos pixels capturados, segmentar a imagem capturada em regiões consideradas ou não do caminho.

Normalmente o primeiro passo na análise de uma imagem é a sua segmentação. Na área de processamento digital de imagem (PDI), segmentar significa dividir a imagem de entrada I em partes, sabendo-se que a quantidade de divisões efetuadas depende do contexto do problema. Geralmente a segmentação é uma das tarefas mais difíceis em PDI, mas recompensa por tornar mais simples a análise computacional da imagem [Neves and Pelaes 2001].

A segmentação pode seguir duas propriedades: descontinuidade e similaridade. No caso da descontinuidade a partição da imagem é feita com base nas alterações bruscas de intensidade, como por exemplo, a detecção de contornos (bordas), detecção de pontos isolados e detecção de linhas. Para a similaridade a partição da imagem é feita com base na similaridade entre os *pixels*, seguindo um critério definido como, binarização, crescimento de regiões (*Region Growing*), divisão e junção de regiões (*Split & Merge*), aglomeração (*Clustering*) e limiares (*Thresholding*) [Neves and Pelaes 2001].

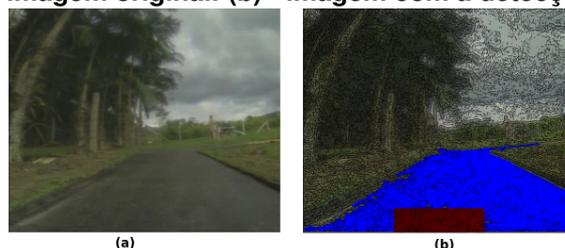
A segmentação de imagem para a detecção de caminho utilizada neste trabalho seguiu tanto a propriedade de descontinuidade como a de similaridade. Para realizar a detecção do caminho é aplicado a segmentação de imagens a cada frame enviado pela câmera. A ideia consiste, basicamente, em dividir a imagem em várias regiões, onde cada uma dessas regiões contenha um índice, separando dessa forma a imagem em diversas áreas indexadas. Para que a segmentação realmente defina uma parte da imagem como o caminho, foi aplicado primeiramente uma detecção de bordas (canny edge), permitindo então a utilização de um algoritmo de preenchimento de regiões (floodfill). Pode-se separar o processo em três etapas pré-processamento, segmentação da imagem e localização do caminho.

O pré-processamento consiste em realizar uma detecção de bordas, que deve pegar todas as fronteiras possíveis na imagem, ou seja, quando ocorrem mudanças bruscas no valor de intensidade entre um pixel e outro, isso evitando ao máximo que ruídos sejam considerados bordas, neste trabalho foi utilizado o algoritmo de Canny [Canny 1986], que trabalha de forma que os ruídos atrapalhem o mínimo possível o resultado final.

Após o pré-processamento da imagem, inicia-se a segmentação, que visa dividir a imagem em áreas homogêneas distintas. Para isso, é criado um mapa de indexação que mantém um determinado valor correspondente para o qual a área de algum dado pixel pertence. O método de segmentação é baseado em uma medida de similaridade para cada pixel adjacente, levando em consideração uma rede de gradientes responsável por determinar restrições durante o processo de fusão dos pixels, garantindo desta forma que nenhuma região será unida fora dos seus limites, após isso é aplicado o método de crescimento de regiões. Um algoritmo de segmentação baseado em crescimento de regiões consiste em agrupar pixels ou sub-regiões em regiões maiores. Para isso leva-se em consideração alguns pixels de entrada, conhecidos como *seed points* (pontos sementes), e a partir destes pontos é que se agrupam regiões próximas e similares.

Por fim, após a imagem segmentada, ocorre a localização do caminho navegável. O objetivo é encontrar qual das áreas anteriormente obtidas é o caminho na imagem. Esta etapa é realizada considerando que o caminho navegável existirá dentro de uma região de interesse na imagem, apresentada por um retângulo vermelho (Figura 3 (b)).

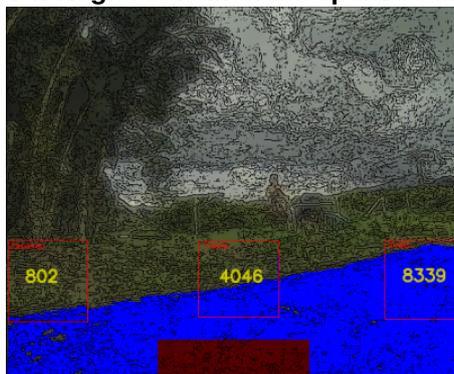
Figura 3. (a) - imagem original. (b) - imagem com a detecção de caminho.



3. Controle

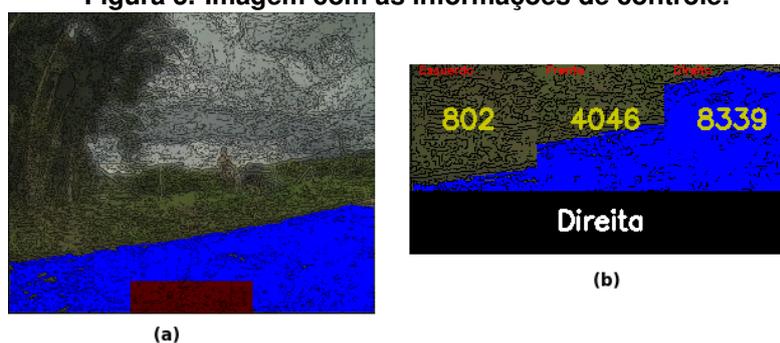
Com a imagem segmentada e o caminho navegável detectado, é iniciada a análise dessa imagem para o controle do veículo. Para isso, são definidas três regiões de interesse (region of interest - ROI) na imagem (Figura 4), uma para verificar quando o caminho inicia uma curva para a esquerda, outra para verificar quando o caminho inicia uma curva para a direita e outra para verificar quando deve-se continuar reto. Essas regiões podem ser alteradas dependendo do cenário e da posição da fonte de captura.

Figura 4. Regiões de interesse para o controle.



Com as três regiões de interesse definidas, é feita uma varredura a cada frame recebido com o caminho detectado em cada uma das ROI's, a região que estiver com a maior quantidade de pixels pertencentes a área do caminho é considerada como a região que o veículo autônomo deve seguir. Por exemplo, se das três regiões de interesse, a região que representa a verificação de curva para direita for a que tiver mais pixels pertencentes ao caminho, então é enviado comando ao carrinho para que ele vire à direita (Figura 5), e isso funciona da mesma forma para as outras duas regiões de interesse.

Figura 5. Imagem com as informações de controle.



Após os primeiros experimentos, verificou-se a necessidade de adaptar o controle. Pois em situações em que o veículo estava finalizando uma curva, a região de interesse de reta ter mais pixels de caminho do que a região pertencente a curva (direita ou esquerda), fazendo com que o veículo não completasse totalmente a curva. Então, após verificação dos resultados e análise do cenário, foi colocada mais uma condição para as curvas, considerando essa situação que se repetia. Assim, quando a região de interesse considerada como frente é a de maior quantidade de pixels de caminho, ainda é verificado a diferença entre sua quantidade de pixels com as regiões de curva, e estando dentro de um limiar definido, deve-se continuar a curva. Por fim, os valores de controle são passados ao Arduino.

4. Resultados

Este trabalho foi avaliado com dados reais e um veículo seguindo um caminho com base no resultado da captura de imagem e análise e interpretação da mesma (detecção do caminho). O veículo foi testado em um mini circuito que contém diferentes terrenos, como: asfalto, terra e lajota. O requisito de sucesso define-se da seguinte maneira: O veículo deve se manter dentro do caminho, ou seja, cumprir o trajeto desenhado realizando as curvas e fazendo correções de direção sempre que necessário. Lembrando que o veículo deve percorrer o trajeto definido sem nenhuma intervenção humana. Vale lembrar que a proposta é avaliada ao ar livre, estando sujeita há diferentes condições de iluminação a cada novo experimento. Os resultados obtidos são apresentados considerando cada trecho do circuito com base na imagem transmitida em tempo real (Figura 1).

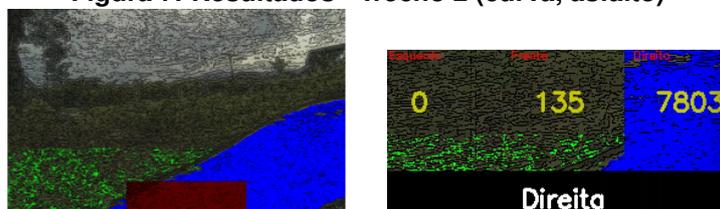
O primeiro trecho, todo de asfalto, pode ser considerado o mais simples pelo fato do asfalto ter um padrão de cor homogênea (preto) e por ser uma reta longa o que facilita a detecção do caminho. A detecção do caminho obteve sucesso e a navegação autônoma também (Figura 6).

Figura 6. Resultados - Trecho 1 (reta, asfalto)



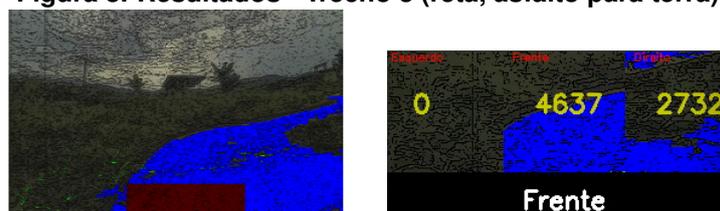
No segundo trecho, ocorre o primeiro desafio, neste trecho o veículo deve detectar o caminho em uma curva e realizar sozinho a curva. Neste trecho também houve êxito na detecção e navegação (Figura 7).

Figura 7. Resultados - Trecho 2 (curva, asfalto)



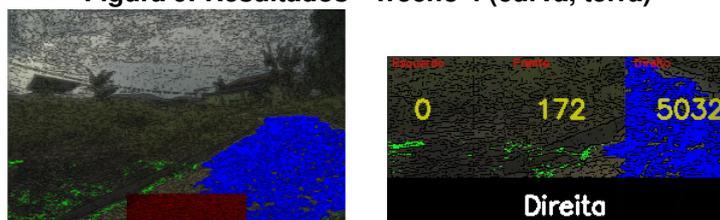
A alteração entre os tipos de terreno (padrões de cor) é provavelmente a maior dificuldade na detecção de caminho na VC e PDI, o veículo deve continuar detectando o caminho e seguindo o trajeto. O veículo só continua caso a detecção de caminho funcione, e neste caso o veículo detectou o caminho e continuou seu trajeto com sucesso (Figura 8).

Figura 8. Resultados - Trecho 3 (reta, asfalto para terra)



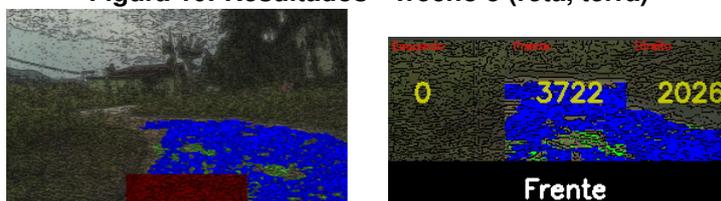
A detecção de caminho deve continuar mesmo em diferentes tipos de terreno, e dentre os padrões de terreno utilizados nos experimentos para avaliação da proposta, a parte de terra é provavelmente a mais difícil de se reconhecer e segmentar, justamente pela falta de homogeneidade e variação na textura do terreno (Figura 9).

Figura 9. Resultados - Trecho 4 (curva, terra)



No trecho 5, uma reta no terreno de terra, o veículo se alinhou corretamente e continuou reto no caminho (Figura 10).

Figura 10. Resultados - Trecho 5 (reta, terra)



No trecho 6 em diante, foram os mais complicados, devido, principalmente, a câmera do veículo pegar o sol de frente. Com isso a segmentação falhou (Figura 11) o que ocasionou o veículo perder o caminho e andar reto durante um curto tempo até parar. Quando o veículo fica um certo tempo, uma certa quantidade de frames, sem detectar o caminho, ele recebe o comando de parar, e foi o que ocorreu com o veículo. Isso acontece pela variação de luminosidade e também pelo fato da transmissão do vídeo pelas antenas diminuir consideravelmente a qualidade da imagem. Entretanto, em outros testes é possível ver que o controle do veículo estaria sendo feito de forma correta (até mesmo em alguns momentos com o sol, Figura 12). Também nota-se que a lajota é detectada na intersecção com a areia (Figura 13).

Figura 11. Resultados - Trecho 5 erro (reta, terra)

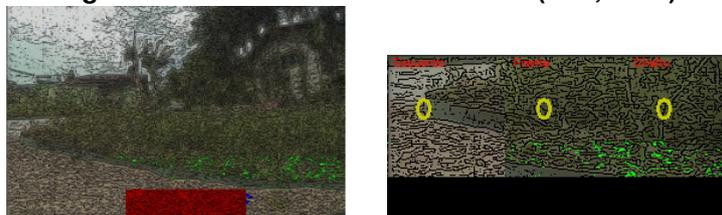


Figura 12. Detecção do caminho com luz solar forte.

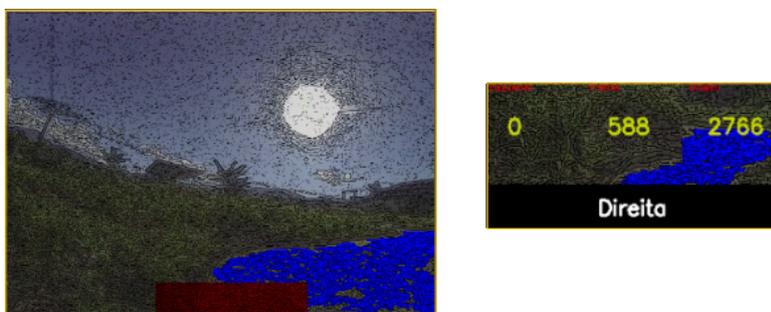
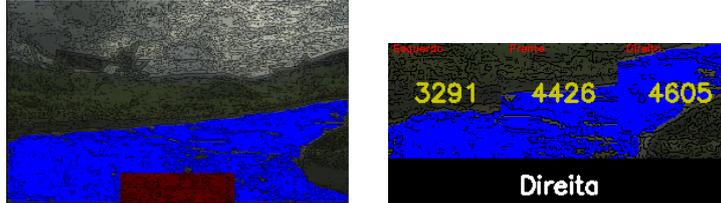


Figura 13. Resultados - Trecho 6 (curva, terra para lajota)



5. Conclusões

Este trabalho teve como grande motivação, verificar a possibilidade de utilizar a informação de imagem, para o controle de um veículo autônomo. Verificando assim a possibilidade do uso de técnicas de VC na navegação autônoma, em situações reais e não somente com imagens gravadas. Os resultados obtidos demonstram que é possível utilizar técnicas de VC e PDI para realizar o controle do veículo. Entretanto em diversas situações pode ocorrer, como realmente ocorreram, falhas na detecção do caminho, mas há possibilidade de ser atenuado, como por exemplo, pode-se estudar formas de se lidar com problema de iluminação, e aplicar outros métodos para atuarem em conjunto. Um agravante ainda maior nesse trabalho, foi o fato da transmissão ter sido realizada via rádio, o que diminuía muito a qualidade da imagem, conseqüentemente limitando a eficácia da detecção. Sendo este o principal foco na próxima etapa de desenvolvimento deste trabalho: Colocar o processamento em um microprocessador (Raspberry pi) que recebe as imagens diretamente da câmera.

Referências

- Álvarez, J. M., Gevers, T., and López, A. M. (2009). Vision-based road detection using road models. In *Proceedings of the 16th IEEE international conference on Image processing, ICIIP'09*, pages 2049–2052, Piscataway, NJ, USA. IEEE Press.
- Bai, L. and Wang, Y. (2011). Road tracking using particle filters with partition sampling and auxiliary variables. *Computer Vision and Image Understanding*, 115(10):1463 – 1471.
- Canny, J. (1986). A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(6):679–698.
- Chen, C.-L. and Tai, C.-L. (2010). Adaptive fuzzy color segmentation with neural network for road detections. *Eng. Appl. Artif. Intell.*, 23(3):400–410.
- Dornaika, F., Alvarez, J., Sappa, A., and Lopez, A. (2011). A new framework for stereo sensor pose through road segmentation and registration. *Intelligent Transportation Systems, IEEE Transactions on*, 12(4):954–966.
- Fernandes, L. C., Souza, J. R., Pessin, G., Shinzato, P. Y., Sales, D., Mendes, C., Prado, M., Klaser, R., Magalhães, A. C., Hata, A., Pigatto, D., Branco, K. C., Jr., V. G., Osorio, F. S., and Wolf, D. F. (2014). Carina intelligent robotic car: Architectural design and applications. *Journal of Systems Architecture*, 60(4):372 – 392.
- Gopalan, R., Hong, T., Shneier, M., and Chellappa, R. (2012). A learning approach towards detection and tracking of lane markings. *Intelligent Transportation Systems, IEEE Transactions on*, 13(3):1088–1098.

- GUO, C.-Z., YAMABE, T., and MITA, S. (2013). Drivable road boundary detection for intelligent vehicles based on stereovision with plane-induced homography. *Acta Automatica Sinica*, 39(4):371 – 380.
- Kong, H., Audibert, J.-Y., and Ponce, J. (2010). General road detection from a single image. *Trans. Img. Proc.*, 19(8):2211–2220.
- LIDAR (2015). *LIDAR*. UK - <http://www.lidar-uk.com/>, acesso em: 18 fev. 2015.
- Marzotto, R., Zoratti, P., Bagni, D., Colombari, A., and Murino, V. (2010). A real-time versatile roadway path extraction and tracking on an {FPGA} platform. *Computer Vision and Image Understanding*, 114(11):1164 – 1179.
- Neves, S. and Pelaes, E. (2001). *Estudo e implementação de técnicas de segmentação de imagens*. Campus Universitário do Guamá, Belém, Pará.
- Oniga, F. and Nedeveschi, S. (2010). Processing dense stereo data using elevation maps: Road surface, traffic isle, and obstacle detection. *Vehicular Technology, IEEE Transactions on*, 59(3):1172–1182.
- Ososinski, M. and Labrosse, F. (2013). Automatic driving on ill-defined roads: An adaptive, shape-constrained, color-based method. *Journal of Field Robotics*, pages n/a–n/a.
- STANDARD, A. N. (2005). *American National Standard for Safe use of Lasers Outdoors*. Orlando, FL - https://www.lia.org/PDF/Z136.6_s.pdf, acesso em: 18 fev. 2015.
- Standard, G. S. P. I. (2001). Safety of laser products. part 1: Equipment classification, requirements and user's guide - http://www.microasu.com/download/jqlaser/certificate/iec_60825_1.pdf, acesso em: 18 fev. 2015.
- Urmson, C. (2014a). Just press go: designing a self-driving vehicle.
- Urmson, C. (2014b). The latest chapter for the self-driving car: mastering city street driving.
- Urmson, C., Anhalt, J., Bae, H., Bagnell, J. A. D., Baker, C. R., Bittner, R. E., Brown, T., Clark, M. N., Darms, M., Demitrish, D., Dolan, J. M., Duggins, D., Ferguson, D., Galatali, T., Geyer, C. M., Gittleman, M., Harbaugh, S., Hebert, M., Howard, T., Kolski, S., Likhachev, M., Litkouhi, B., Kelly, A., McNaughton, M., Miller, N., Nickolaou, J., Peterson, K., Pilnick, B., Rajkumar, R., Rybski, P., Sadekar, V., Salesky, B., Seo, Y.-W., Singh, S., Snider, J. M., Struble, J. C., Stentz, A. T., Taylor, M., Whittaker, W. R. L., Wolkowicki, Z., Zhang, W., and Ziglar, J. (2008). Autonomous driving in urban environments: Boss and the urban challenge. *Journal of Field Robotics Special Issue on the 2007 DARPA Urban Challenge, Part I*, 25(8):425–466.
- Zhou, S., Gong, J., Xiong, G., Chen, H., and Iagnemma, K. (2010). Road detection using support vector machine based on online learning and evaluation. In *Intelligent Vehicles Symposium (IV), 2010 IEEE*, pages 256–261.