**UNIVERSIDADE DE SANTA CRUZ DO SUL**

**DEPARTAMENTO DE INFORMÁTICA**

**CURSO DE CIÊNCIA DA COMPUTAÇÃO**

**Jeferson Maiquier Pagel**

**Um método para detecção de vagas de estacionamento baseado em processamento de imagens**

**Santa Cruz do Sul**

**2015**

**Jeferson Maiquier Pagel**

**Um método para detecção de vagas de estacionamento baseado em processamento de imagens**

Trabalho de Conclusão de Curso I apresentado ao Curso de Ciência da Computação da Universidade de Santa Cruz do Sul para obtenção parcial do título de Bacharel em Ciência da Computação.

Orientador Dr. Leonel Pablo Tedesco

**RESUMO**

Esse trabalho de conclusão tem por objetivo desenvolver um método de reconhecimento de vagas de estacionamento, aliado aos conceitos teóricos necessários para elaborar um sistema de visão computacional para reconhecer padrões e identificar objetos do mundo real. Para o desenvolvimento da ferramenta foi utilizado funções da biblioteca OpenCV (Intel *Open Source Computer Vision Library*) para realizar o processamento das imagens que foram coletas por um DRONE (*Dynamic Remotely Operated Navigation Equipment*). Para a validação do sistema serão feitos testes em ambientes reais com diferentes condições de luminosidade.

**Palavras Chave:** OpenCV, Estacionamento, Visão Computacional, DRONE, Processamento de imagens.

**LISTA DE FIGURAS**

**LISTA DE TABELAS**

**LISTA DE ABREVIATURAS**

**OpenCV** Intel *Open Source Computer Vision Library*

**DRONE** *Dynamic Remotely Operated Navigation Equipment*

**HD** *High Definition*

**GPS** *Global Positioning System*

**SUMARIO**

1. **INTRODUÇÃO**

Com o desenvolvimento tecnológico no século 20, os centros urbanos sofreram um grande crescimento para suprir a demanda de trabalhadores das industrias, com isso o problema de locomoção nas cidades começou a ganhar destaque. A ideia de desenvolvimento se concentrava em torno do automóvel, que de certa forma favorecia o transporte particular como o principal meio de locomoção. Por outro lado, os altos preços de transportes coletivos e a falta de investimento na infraestrutura do trânsito acabaram por retardar ações de transporte de massa capazes de viabilizar a dinâmica de mobilidade.

Conforme descrito pela Assembleia Legislativa do Rio Grande do Sul (2013), o elevado número de automóveis em utilização fazem como que os principais centros comerciais das cidades fiquem aglomerados, o que leva a saturar os estacionamentos. Na tentativa de solucionar o problema, tentou-se implementar estacionamentos rotativos, porem com a alta demanda que muitas vezes existe o sistema não se mostra eficiente e capaz de atender a demanda, uma vez que os motoristas precisam dirigir por várias quadras para encontrar um lugar vago.

Outro aspecto que chama a atenção é o desenvolvimento de veículos autônomos, que possuem a habilidade de se autodirigir para qualquer local. Nos EUA esta tecnologia está prestes a virar realidade, pois vários estados já permitiram a realização de testes com veículos autônomos desde que estejam acompanhados de um motorista humano que possua habilitação. Nestes veículos, um sistema de visão computacional que interprete corretamente as imagens, não apenas identificando sinais de trânsito, mas estradas, pessoas e objetos em geral, em tempo real, é fundamental (THRUN, 2006; KEVICZKY, 2007; GALLO 2008; ESS, 2009).

O presente trabalho de conclusão busca sugerir uma solução para que a movimentação de veículos em estacionamentos possa ocorrer de forma mais eficaz, através de informações geradas pelo processamento de imagens. A solução a ser aqui empregada irá utilizar um DRONE para a coleta e envio das imagens a serem processadas por um servidor, no qual será analisada a existência de vagas de estacionamento disponíveis, suas respectivas localizações e hora da captura.

Para a análise e processamento das imagens será utilizada a biblioteca OpenCV que foi desenvolvida pela Intel (BRADSKI e KAEHLER, 2008). De acordo com o OpenCV manual online, essa biblioteca possui mais de 500 funções de processamento de imagens, muitas delas complexas que podem ser acessadas com facilidade em linguagens como C, C++, Phyton e Java. Para a solução do problema, podem ser utilizadas funções de conversão de cores, cálculo de distância e funções de medidas. Os parâmetros obtidos através da análise das imagens com a biblioteca OpenCV serão armazenados em um banco de dados que será acessado através de um webservice que alimentará um aplicativo Android com as informações que serão disponibilizadas para os usuários.

**1.1 OBJETIVOS**

O objetivo deste trabalho de conclusão é desenvolver uma ferramenta que auxilie motoristas a encontrar vagas de estacionamento de forma rápida e eficaz. A ferramenta será dividida em três partes as quais são: a coleta dos dados, processamento dos dados coletados e a disponibilização da informação para os usuários.

**Específicos**

* Utilizar um DRONE para coleta de imagens e da localização no estacionamento.
* Dominar técnicas de processamento de imagens para detecção de vagas de estacionamento.
* Determinar as melhores posições para captura de imagens sobre a ocupação em estacionamentos.
* Definir modelos de processamento de imagens de estacionamentos utilizando a biblioteca OpenCV.
* Estimar as taxas de ocupação de estacionamentos, através da captura de imagens em diferentes momentos.

1. **FUNDAMENTAÇÃO TEÓRICA**

Nesse capítulo é abortado os temas envolvidos na proposta deste trabalho, no qual será descrito os passos necessários para realizar a coleta e o processamento dos dados, a disponibilização das informações para os usuários e os trabalhos relacionados.

* 1. **COLETA DOS DADOS**

Para a coleta de dados pensou-se na utilização de um DRONE, pois permite a mobilidade da câmera para coleta das imagens, assim tornando o valor total da aplicação mais baixo e facilitando também a manutenção do sistema.

O modelo do DRONE utilizado nesse trabalho é um AR.Drone 2.0 que é produzido pela Parrot Inc, que é uma plataforma robótica que pode ser controlada por uma rede WIFI, onde pode ser controlado o voo, a captura de fotos e vídeos, assim como acessar a posição do GPS do equipamento. Para o voo pode ser ajustado a velocidade de deslocamento, o angulo máximo de voo, a altura em que o DRONE pode voar. Esse modelo de DRONE em particular se destaca pelo baixo valor, sua robustez contra batidas e vasta variedade de sensores. Para a captura de imagens e vídeos, o equipamento possui uma câmera frontal que trabalha na qualidade HD, porem o equipamento possui também uma câmera posicionada embaixo do aparelho, a qual possui qualidade inferior e é utilizada para estimar a velocidade de deslocamento. (Skoda e Bartak)

Na imagem 1 é possível observar os movimentos e os ajustes possíveis na configuração do voo, os quais são realizados através do firmware do dispositivo que tem como interface de usuário o software AR.FreeFlight produzido pela própria Parrot Inc.



Imagem 1 - AR.Drone com seu sistema de coordenação. (Krajnik et al. 2011)

A ferramenta AR.FreeFlight é gratuita e de código aberto, onde existe um vasto número de usuários que desenvolvem e pesquisam para que seja possível aumentar o número e qualidade das funcionalidades oferecidas pela ferramenta.

Na tela inicial da ferramenta, pode-se acessar as fotos e fotos já capturas em outros voos, acessar o AR.Drone Acedemy onde pode-se encontrar o histórico de voos salvos com a posição de GPS, horários, fotos e vídeos, ou pode-se iniciar um novo voo.

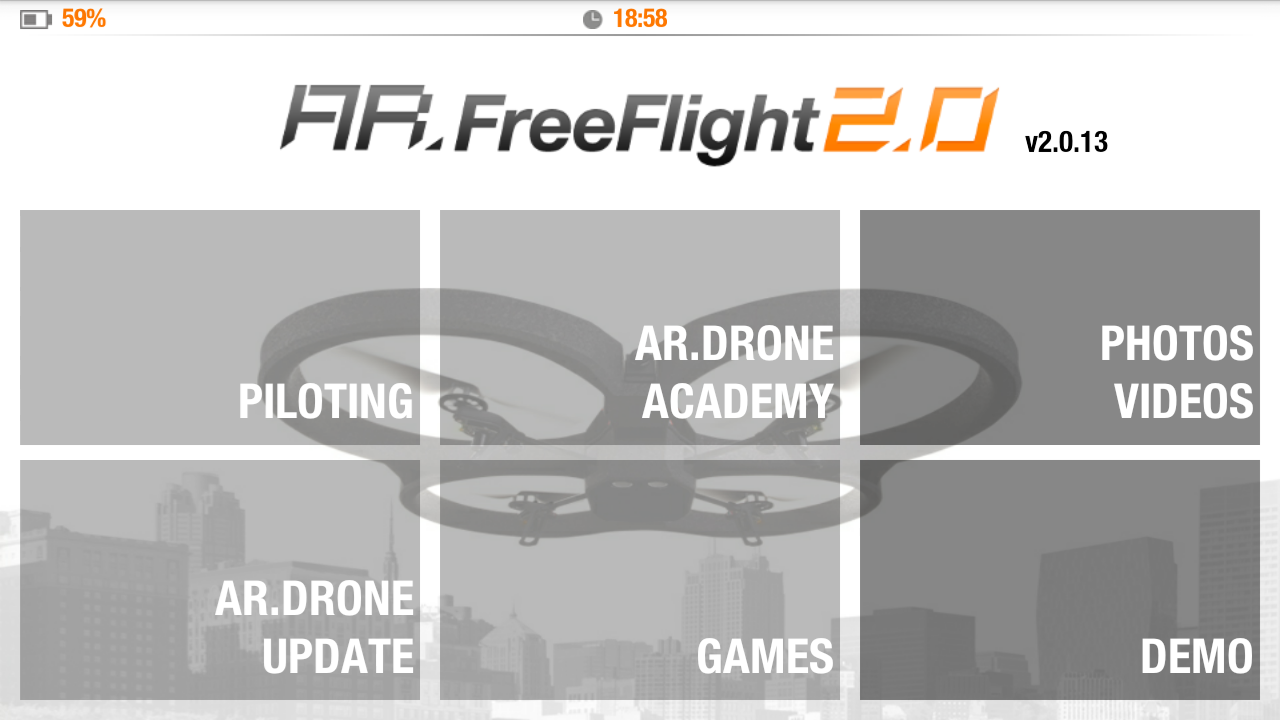


Imagem 2 – Tela Inicial da ferramenta

Na imagem 3, pode-se ver as opções de ajustes de voo disponibilizados para os voos, nessa seção pode se alterar o máximo de altitude, velocidade de deslocamento, velocidade de rotação e o ângulo de inclinação do voo, ou ainda, alterar a forma que o DRONE será controlado.

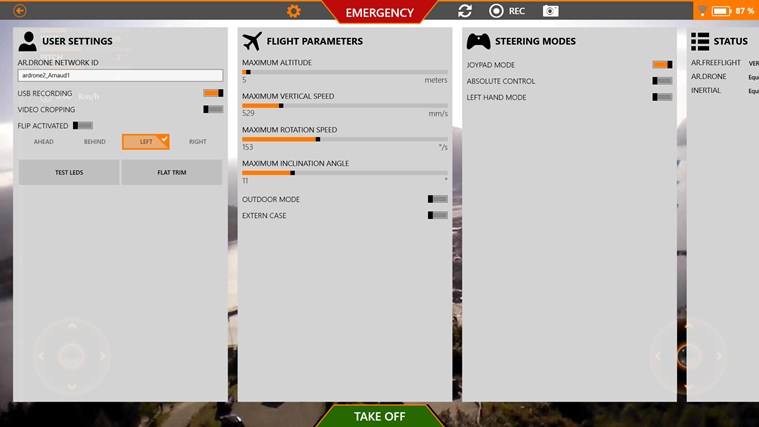


Imagem 3 – Ajustes de voo

Na imagem 4, é possível observar a interface de voo, onde o usuário pode ver o ambiente através do DRONE e controla-lo para que vá para lugares que deseja. Nessa tela o usuário pode também tirar fotos ou gravar o voo. As fotos e vídeos podem ser feitos através da câmera HD que fica na parte frontal do equipamento ou pela câmera que fica embaixo do DRONE, porem essa possui qualidade inferior se comparada com a câmera frontal onde pode-se ter problemas para o processamento nas imagens, quando não se tenha uma boa visibilidade do terreno analisado.



Imagem 4 – Tela de voo

Imagens, configurações.

Coleta da Imagens

Dificuldades encontradas, bom, ruim, qualidade das informações coletadas.

Analise em ambientes mutáveis

Diferentes ângulos

Diferentes iluminações

**2.2 Processamento de imagens**

O porquê do processamento das opções existentes.

OPEN CV

Para desenvolver o sistema proposto neste trabalho, foram utilizadas algumas funções da biblioteca OpenCV, que disponibiliza funções complexas que podem ser utilizadas de forma simplificada em sistemas desenvolvidos nas linguagens C, C++, Python e Java. Esta foi desenvolvida pela empresa Intel (INTEL, 2012) e implementada nas linguagens C e C++, proporcionando suporte ao multiprocessamento e eficiência otimizada nos processadores desenvolvidos pela empresa (linha de processadores Intel; OPENCV, 2013).

A biblioteca OpenCV foi projetada com o intuito de tornar o desenvolvimento de sistemas de visão computacional mais rápido, eficiente e otimizado em diversas áreas, como a interação homem-máquina, em sistemas de tempo real e robótica. É uma biblioteca de código aberto e inclui recursos avançados para desenvolver sistemas que utilizam o processamento de imagens, efetuar transformações, analisar movimentos, calibrar a câmera, reconstrução 3D, detectar características e aprendizado de máquina. Atualmente, a biblioteca disponibiliza mais de 500 funções de PDI, reconhecimento de padrões e aprendizagem de máquina (OPENCV, 2013).

O pacote de arquivos que compõem a biblioteca OpenCV, bem como manuais de referência, são disponibilizados gratuitamente na internet (OPENCV, 2013). A biblioteca pode ser dividida em cinco grupos de funções básicas, conforme descrito abaixo:

A biblioteca OpenCV possui uma estrutura modular, em que seus pacotes incluem várias bibliotecas estáticas ou compartilhadas. Os principais módulos dessa biblioteca são:

 CORE: possui estruturas básicas de dados, e álgebra linear.

 IMGPROC: possui funções de filtragem linear e não-linear, transformações geométricas, conversões entre modelos de cor, histogramas, entre outros.

39

 VIDEO: este modulo traz funções para análise de vídeos, incluindo estimativas de movimentos, subtração de fundo e algoritmos de rastreamento de objetos.

 CALIB3D: possui algoritmos básicos de geometria, calibração de câmera e sistema de som, estimativa de posição de objetos e elementos de reconstrução 3D.

 OBJDETECT: possui funções de detecção de objetos e instâncias de classes pré-definidas, como por exemplo, rosto, olhos, pessoas e carros).

 HIGHGUI (*High-level Graphical User Interface*): disponibiliza funções relacionadas a interfaces gráficas com o usuário, além de entrada e saída de vídeos (OPENCV, 2013).

A Figura 15, a seguir, mostra a estrutura básica da biblioteca OpenCV. Esta disponibiliza mais de 500 funções que podem ser utilizadas em processamento de imagens.

Como fazer

Melhorar o processamento(C++)

**2.3 Mostrar os resultados**

Colocar de acordo com os objetivos, como pode ser naquele sentido.

Disponibilizar os resultados

1. **Trabalhos Relacionados**

Na literatura, trabalhos relacionados foram pesquisados, utilizando como técnicas computacionais para processamento de imagens em ambientes mutáveis. A seguir, são apresentados alguns trabalhos relacionados pesquisados e estudados.

* 1. **Trabalho 1**
  2. **Trabalho 2**
  3. **Trabalho 3**
  4. **Trabalho 4**
  5. **Trabalho 5**
  6. **Comparativo dos trabalhos relacionados**

Durante as pesquisas realizadas para a realização deste trabalho, vários projetos semelhantes foram encontrados. A abordagem utilizada no presente trabalho não é exatamente a mesma da encontrada em outros trabalhos, mas o estudo destes foi relevante para a compreensão de alguns conceitos e diferentes técnicas possíveis. Na sequência, são descritos alguns destes trabalhos.

No trabalho de (SOUSA, 2007) é utilizada a biblioteca OpenCV para desenvolver um sistema de detecção da FTP em imagens. Esse sistema é direcionado para dispositivos móveis com sistema operacional Android na versão 2.3. Inicialmente é detectada a FTP e em seguida é verificada a presença de veículos e/ou pessoas sobre esta faixa para definir se a travessia é segura ou não. Nesta proposta, a interação com o usuário é feita através de comandos de voz e do recurso de vibração do aparelho móvel utilizado. Além disso, as imagens são convertidas em escalas de cinza, diferentemente da proposta apresentada nesse trabalho, onde se pretende realizar os processamentos em imagens coloridas.

Outra proposta de sistema autônomo de reconhecimento de sinais de trânsito em imagens pode ser encontrada em (BARGHAVA, 2010). Nesta, é identificado o semáforo em imagens e verificado o seu estado. A arquitetura proposta consiste na instalação de micro câmeras na armação de um óculos de sol. Estas micro câmeras se comunicam com um dispositivo móvel (aparelho celular, por exemplo) com acesso à internet. Este, por sua vez, envia a imagem para um servidor remoto, o qual realiza o processamento e retorna o resultado para o dispositivo móvel. Logo, o processamento é feito em nuvem e não é local, conforme proposto neste projeto. Para identificar a localização atual do usuário, o sistema utiliza-se de coordenadas de sinal de GPS e se comunica com o servidor Google Maps. O sistema captura a imagem e envia para o servidor, o qual realiza o processamento. Na sequência, o sistema informa ao usuário, em forma de comando de voz no dispositivo móvel, quando o momento é seguro para atravessar a rua (BARGHAVA, 2010).

Em seu trabalho, (KIM, 2011) apresenta um método de identificação da sinaleira de trânsito e o seu estado para sistemas veiculares, em ambiente real, considerando apenas o turno diurno. É utilizado o modelo de cores *YCbYr*, uma extensão do modelo *YUV*. São propostos pré-processamentos para eliminar ruídos na imagem e a área e objetos que não são de interesse, tais como pessoas, carros, prédios, entre outros, trabalhando apenas com a parte da imagem que representa a sinaleira. Após isso, são encontrados os BLOBs, para os quais

44

existem predefinições para avaliar se os mesmos representam as luzes da sinaleira, de acordo com seu formato, tamanho e cor (KIM, 2011).

O trabalho de (KIM, 2011) se diferencia do proposto neste projeto em vários aspectos. Um deles é o fato do processamento ser feito em um microcomputador normal ao invés de ser utilizado um hardware mais específico, com limitações. Outro aspecto é o fato do sistema ser voltado para veículos autônomos e não para robôs autônomos, como o modelo proposto nesse trabalho. Apesar disso, o estudo do mesmo é de suma importância para a compreensão e entendimento de várias técnicas de processamento de imagens utilizadas, principalmente nos pré-processamentos e na identificação da sinaleira.

Outra proposta semelhante, encontrado na literatura, é apresentada por (GUO-SHENG, 2009). O objetivo dos autores é apresentar um sistema para veículos autônomos capaz de evitar acidentes de trânsito, em especial, atropelamento de pessoas sobre a FTP. Inicialmente, é detectada a FTP, como área de interesse na imagem. Em seguida, é verificada a presença de um pedestre sobre essa faixa. Na sequência é calculado o momento (tempo) em que o veículo alcança a FTP, de acordo com a distância entre a posição atual do veículo, a FTP e a velocidade atual do veículo. É utilizado o método probabilístico kalman (THRUN, 2005) para estimar a posição do pedestre no momento em que o veículo alcança a FTP. A partir dessas informações, o sistema de controle do veículo gerencia a velocidade do mesmo, para que este prossiga na atual velocidade, reduza ou até mesmo pare, para evitar uma colisão com o pedestre, de acordo com o cenário (GUO-SHENG, 2009).

Em (SICHELSCHMIDT, 2010) é apresentada outra proposta de sistema com propósito semelhante ao deste trabalho. Os pesquisadores alemães apresentam um sistema de detecção de FTP do tipo zebrada utilizando a transformada de Fourier, bipolaridade aumentada e a relação de orientação de borda para a classificação. O processamento ocorre em imagens em tons de cinza. O fluxo básico do sistema consiste em: a) capturar a imagem em tons de cinza, através de uma câmera. b) segmentar a imagem. c) extração de características. d) classificação. e) pós-processamento (SICHELSCHMIDT, 2010).

O processo de segmentação é estruturado de forma bastante interessante, sendo feito um processamento específico para compensar o ângulo de inclinação da imagem e determinada a região de interesse. Na sequência, é feita a equalização do histograma para aprimorar o contraste da imagem para que possa ser aplicada a detecção de bordas com maior eficiência. É adotado o algoritmo de *CANNY* para a detecção das bordas, para auxiliar na posterior

45

identificação das faixas. É adotada também a técnica de dilatação para contornar problemas de FTP irregulares e/ou apagadas parcialmente. O sexto passo é a extração dos contornos. A seguir, é aplicada a transformada de Fourier em cada linha da imagem, visto que o padrão de FTP indica uma determinada largura das linhas brancas da faixa, que é igual ao espaço entre elas. Finalmente é utilizada a abordagem da transformada de Hough, para excluir linhas que não pertencem a FTP (SICHELSCHMIDT, 2010).

Para evitar a taxa de erros, ou seja, reconhecimento incorreto da FTP, é realizada uma etapa de pós-processamento. Nesta, é utilizado um algoritmo de rastreamento baseado no filtro de Kalman, o que exige mais processamento e consequentemente torna o reconhecimento mais lento. Entretanto, com o uso dessa técnica, são diminuídos os reconhecimentos incorretos, tornando o sistema mais confiável (SICHELSCHMIDT, 2010).

Os trabalhos explicados nesta seção apresentaram bons resultados, porém, em sua maioria as imagens são binarizadas ou convertidas para escala de cinza. Além disso, em nenhum deles os testes são realizados com um hardware similar ao utilizado neste trabalho, onde as imagens são ainda tratadas de forma diferente, de acordo com as condições climáticas.

1. **Considerações Finais**

**Referencias**

**Skoda, J.; Bartak, R. Camera-Based Localization and Stabilization of a Flying Drone. Proceedings of the Twenty-Eighth International Florida Artificial Intelligence Research Society Conference**