

**UNIVERSIDADE DO ESTADO DO RIO GRANDE DO NORTE
NÚCLEO AVANÇADO DE ENSINO SUPERIOR DE NOVA CRUZ
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

CARLOS EDUARDO SILVA MORAIS

**SISTEMA DE ODOMETRIA VISUAL 3D UTILIZANDO O SENSOR RGB-D
KINECT**

**NOVA CRUZ - RN
AGOSTO/2014**

CARLOS EDUARDO SILVA MORAIS

**SISTEMA DE ODOMETRIA VISUAL 3D UTILIZANDO O SENSOR RGB-D
KINECT**

Monografia apresentada à Universidade do Estado do Rio Grande do Norte como um dos pré-requisitos para obtenção do grau de bacharel em Ciência da Computação.

ORIENTADORES: Prof. Dr. Anderson Abner de Santana Souza
Profa. Dra. Rosiery da Silva Maia

**NOVA CRUZ - RN
AGOSTO/2014**

CARLOS EDUARDO SILVA MORAIS

**SISTEMA DE ODOMETRIA VISUAL 3D UTILIZANDO O SENSOR RGB-D
KINECT**

Monografia apresentada à Universidade do Estado do Rio Grande do Norte como um dos pré-requisitos para obtenção do grau de bacharel em Ciência da Computação.

Aprovado em:

Conceito Final:

Banca Examinadora

Prof. Dr. Anderson Abner de Santana Souza (UERN)

Profa. Dra. Rosiery da Silva Maia (UERN)

Prof. Dr. Wilfredo Blanco Figuerola (UERN)

Dedico esse trabalho aos meus pais, professores e amigos, que ao longo desta caminhada, tive o enorme prazer de tê-los ao meu lado.

AGRADECIMENTOS

Durante esta minha jornada acadêmica, até agora, foram cinco anos de erros e acertos, entre Universidades, laboratórios, trabalhos de pesquisa, e intercâmbio, tenho muito que agradecer ao meu bom Deus. A verdade é que jamais imaginei que teria tamanha oportunidade, e hoje me encho de orgulho ao lembrar-me de cada passo dado, de cada conquista obtida.

Além disso, não há nada melhor que ter o conforto e a segurança, em um ambiente cheio de amor e carinho, como o meu lar, ao lado dos meus pais e irmãs, onde descanso, reflito, e discuto sobre planos futuros, e que, como sempre, tenho o pronto apoio de todos. Agradeço a eles, à minha família, que apesar de dispor de poucos recursos financeiros, puderam e, foram a base por trás da realização deste sonho, me incentivando e apoiando integralmente.

Agradeço, em especial, ao meu Professor e Orientador, o Dr. Anderson Abner, pela atenção e paciência para comigo, e pela oportunidade a mim concedida de participar de dois importantes projetos de pesquisa, tendo, inclusive, um desses trabalhos, como base para confecção dessa monografia.

Agradeço ao amigo Dr. Alexander Fink pela confiança e acompanhamento dos meus estudos, e pelo financiando de parte deles através do projeto de bolsistas da ONG “Associação de Formação e Incentivo ao Nordeste Karente” (AFINK).

Agradeço, imensamente, ao amigo Me. Franz Bauer, da Universidade Técnica de Munique (TUM), que me acolheu de braços abertos na casa de sua família e me acompanhou durante meu intercâmbio na cidade de Munique, Alemanha.

Agradeço aos amigos Me. Rasso Reng, Me. Qu Zheng, e Me. Paulo Henrique pela atenção e acompanhamento, dentro e fora da Universidade; todos eles da TUM.

Agradeço também ao amigo Jurasildo Oliveira Reinaldo, que sempre me ajudou, na Universidade do Estado do Rio Grande do Norte (UERN), dispondo de tempo, paciência e companheirismo.

Enfim, obrigado a todos que fizeram parte, de forma direta ou indiretamente, desta conquista! Que Deus abençoe à todos!

Imagine uma nova história para sua vida, e acredite nela.

(Paulo Coelho)

RESUMO

A localização de um robô em seu ambiente é uma tarefa fundamental para dar autonomia ao mesmo. Diante deste problema, a Odometria Visual surge como uma alternativa à Odometria Mecânica realizada através do uso de *encoders* (sensores de rotação), que por serem sensores ruidosos, são suscetíveis ao acúmulo de erros de cálculo da distância percorrida pelo robô ao longo do tempo. Neste trabalho, propomos o uso da Odometria Visual para estimar o movimento de translação e rotação do robô em períodos de tempo, apenas fazendo uso de imagens capturadas do ambiente em que o robô está inserido e, assim, saber a sua pose atual, tendo por base um referencial fixo. Para captura das imagens, é utilizado um sensor RGB-D da Microsoft, o *Kinect*. A cada par de imagens geradas em instantes de tempos e localizações diferentes, aplica-se um detector de pontos sobre ambas, com intuito de identificar pontos notáveis, características comuns entre as imagens. Posteriormente é utilizado um algoritmo de correlação, que faz a correspondência entre os pontos comuns reconhecidos nas imagens, e por fim, obtém-se a profundidade por meio das informações de um mapa de profundidade. Com essas informações, algumas operações são realizadas para se estimar o deslocamento relativo entre as posições nas quais as imagens foram geradas. Dessa forma, este trabalho preocupa-se em apresentar um sistema de Odometria Visual 3D, que possibilite um resultado satisfatório e rápido para auxiliar um sistema de navegação para robôs móveis. Experimentos preliminares, com dados reais, mostram que este sistema, com as devidas adaptações, pode ser utilizado em um robô real.

PALAVRAS-CHAVES: Odometria Visual, Odometria Mecânica, *Kinect*, SURF.

ABSTRACT

The robot localization in an environment is a fundamental task to ensure the robot autonomy. Based on this scenario, the visual odometry is the better option than mechanical odometry that is carried out by the use of encoders which, for the fact that they are noisy sensors, they are susceptible to accumulating calculation errors obtained from the distance travelled by the robot over time. In this work, it is proposed the use of visual odometry to estimate the translation and rotation motion of a robot in periods of time by only using images captured from the environment where it is in, and thus knowing its current pose based on a fixed reference pose. We used the Kinect, a Microsoft's RGB-D sensor, to capture the images. For each image pair created in different periods of time and locations, it is applied a point detector on both of them to identify similar characteristics between the images. Subsequently, it is used a matching algorithm to match the recognized points in the images and then, it is obtained the depth by the information of a depth map. By the results, some operations were carried out to estimate the relative displacement between the poses in which the images were generated. Therefore, this work presents a 3-D Visual Odometry system that allows obtaining a satisfactory and fast result for help a navigation system of a mobile robot. Preliminary experiments, with real data, show that it is possible uses this system by a real robot with some adaptations.

KEYWORDS: Visual Odometry, Mecanic Odometry, Kinect, SURF.

LISTA DE ABREVIATURAS E SIGLAS

2D: Duas dimensões

3D: Três dimensões

API: *Application Programming Interface*

BSD: Termo de licença que designa um software de domínio público

cm: Centímetros

DoF: *degree-of-freedom*

FLANN: *Fast Library for Approximate Nearest Neighbors*

GPS: *Global Positioning System*

LAR: Laboratório de Aprendizagem Robótica

m: Metros

mm: Milímetros

ms: Milissegundos

OpenCV: *Open Source Computer Vision*

OpenNI: *Open Natural Interaction*

OV: Odometria Visual

RANSAC: *RANdom SAmple Consensus*

RGB-D: *Red, Green, Blue, + Depth*

SDK: *Software Development Kit*

SFM: *Structure from Motion*

SIFT: *Scale-invariant feature transform*

SVD: *Singular Value Decomposition*

SURF: *Speeded Up Robust Features*

VGA: *Vídeo Graphics Array*

LISTA DE TABELAS

Tabela 1: Equipamentos utilizados para desenvolvimento do sistema	38
Tabela 2: Comparação dos detectores e descritores de pontos.....	40
Tabela 3: Tabela com erros de cálculos da distância obtidas pelos cinco experimentos	50

LISTA DE FIGURAS

Figura 3.1: Princípio do funcionamento do <i>encoder</i>	22
Figura 3.2: Deslocamento dado pela odometria.....	23
Figura 3.3: Fontes de erros no cálculo do deslocamento por odometria mecânica.....	24
Figura 3.4: Representação do processo de estimação de pose e deslocamento, a partir da captura de pares de imagens com um sistema de câmeras estéreo.....	27
Figura 3.5: Passo a passo para se realizar Odometria Visual.....	29
Figura 3.6: Modelo de rotação.	32
Figura 4.1: Arquitetura do Sistema de Odometria Visual 3D.....	34
Figura 4.2: Imagens 2D I_{k-1} (a) e I_k (b) e os respectivos mapas de profundidade da imagem I_{k-1} (c) e da imagem I_k (d).....	35
Figura 4.3: Aplicação do detector de pontos SURF sobre a imagem I_{k-1} (à esquerda) e I_k (à direita).....	36
Figura 4.4: Aplicação do algoritmo de FLANN para correspondência de pontos notáveis entre as imagens I_{k-1} e I_k . Os pontos com correspondentes em ambas as imagens são utilizados para a criação dos conjuntos X_{k-1} e X_k	36
Figura 4.5: Sensor RGB-D <i>Kinect</i> (a) e seu sistema de coordenadas (b).....	37
Figura 5.1: Experimento sem movimento do <i>Kinect</i>	43
Figura 5.2: Experimento com movimento do <i>Kinect</i> de 0,05m sobre o eixo x	45
Figura 5.3: Experimento com movimento do <i>Kinect</i> de 0,045m em z	46
Figura 5.4: Experimento com movimento de -0,2m em y	48
Figura 5.5: Experimento com movimento de 0,1m em x e -0,1m em z	49

SUMÁRIO

1	INTRODUÇÃO.....	14
1.1	Objetivo Geral.....	16
1.2	Objetivos Específicos	16
1.3	Estrutura do Trabalho	17
2	TRABALHOS RELACIONADOS.....	18
3	FUNDAMENTAÇÃO TEÓRICA	21
3.1	Localização de Robôs Móveis	21
3.2	Odometria Mecânica.....	22
3.2.1	Encoders	22
3.2.2	Cálculo do deslocamento	22
3.2.3	Erros Sistemáticos e Não-sistemáticos.....	23
3.3	Odometria Visual.....	25
3.3.1	Sistemas Estéreo.....	25
3.3.2	Sistemas Monocular	26
3.3.3	Formulação da Odometria Visual	27
3.3.4	Estimação do Movimento.....	29
4	SISTEMA IMPLEMENTADO	34
4.1	Metodologia Adotada no Desenvolvimento do Sistema	34
4.2	O Sensor RGB-D Kinect e outros Equipamentos Utilizados	36
4.3	Bibliotecas OpenCV e OpenNI	38
4.4	Algoritmo de SURF.....	39
4.5	Algoritmo de FLANN	40
5	EXPERIMENTOS E ANÁLISES	42
5.1	Contextualização dos Experimentos.....	42
6	CONCLUSÃO E PERSPECTIVAS DE TRABALHOS FUTUROS	51

REFERÊNCIAS BIBLIOGRÁFICAS	52
---	-----------

1 INTRODUÇÃO

Ao longo do tempo, a história nos mostra que o ser humano sempre esteve em busca de novos mecanismos que facilitassem a realização de determinadas tarefas, que por ventura, trouxessem algum risco à integridade física do homem ou até mesmo o desgaste por conta da necessidade de realizar trabalhos repetitivos. Contudo, antigamente, substituir o ser humano por máquinas era algo impossível, já que faltavam recursos tecnológicos para tal. Atualmente, nossa realidade se difere deste passado não tão distante, graças aos avanços tecnológicos que crescem a um ritmo exponencial. Os robôs são exemplos desta busca constante aliada à evolução da tecnologia.

Segundo Russell & Norvig (2003), os robôs são agentes físicos que executam tarefas manipulando o mundo físico. Para isso, eles são equipados com atuadores como pernas, rodas, articulações e garras, e sensores que lhes permitem perceber o seu ambiente. A Robótica surge então, como sendo a ciência de perceber e manipular o mundo físico através de dispositivos controlados computacionalmente, ou seja, a ciência responsável pelo desenvolvimento dos robôs (THRUN; BURGARD; FOX, 2005).

Inicialmente, concebido para substituir o homem em tarefas repetitivas, com certo grau de periculosidade e/ou precisão, os robôs passaram a ser amplamente difundidos, principalmente nas linhas de montagem da indústria de automobilística (SIEGWART & NOURBAKHS, 2004).

Existem diversos tipos de robôs, os quais podem ser agrupados em três principais categorias: robôs manipuladores, robôs móveis, e robôs híbridos. Os robôs manipuladores (ou braços robóticos), geralmente, são utilizados nas linhas de produção automobilísticas, com base fixa, e espaço de atuação restrito. A categoria dos robôs móveis é constituída por robôs que têm a capacidade de se locomover no espaço tridimensional (robôs aeroespaciais e subaquáticos) ou no espaço planar (robôs terrestres) (SOUZA, 2008). Os robôs terrestres podem ser subdivididos em robôs com pernas (*legged robots*) e robôs com rodas (*wheeled robots*) (SANTANA, 2007). A terceira categoria de robôs, os robôs híbridos, são os robôs móveis equipados com manipuladores, como por exemplo, os humanoides (RUSSEL & NORVIG, 2003).

Um dos problemas fundamentais para se ter um robô móvel com capacidades de autonomia, ou seja, habilidade de realizar tarefas sem a interferência humana, é conseguir localizar o robô em seu ambiente (COX, 1991). Contudo, há mais de 30 anos, diversos

pesquisadores veem trabalhando na solução deste problema, desenvolvendo e aperfeiçoando métodos para obtenção da localização do robô no ambiente (SCARAMUZZA & FRAUNDORFER, 2011), ou seja, as coordenadas e a orientação do robô, em relação a um referencial fixo no ambiente. Esta estimativa realizada a cada intervalo de tempo chama-se pose (SOUZA, 2008).

A pose do robô pode ser estimada através do uso do método de odometria. A odometria é comumente obtida pela integração de informações dos *encoders* nas rodas. Os *encoders* são sensores ópticos utilizados em sistemas de odometria que geralmente são acoplados aos motores das rodas, tanto do lado direito como do lado esquerdo, para medir a rotação das rodas, velocidade ou aceleração. A maioria dos robôs comerciais fazem estimativas de locomoção a partir da integração de informações disponíveis a cada intervalo de tempo pelo método de odometria (THRUN; BURGARD; FOX, 2005).

O método de Odometria utilizado para estimar a pose do robô através do uso de *encoders* é tratado por este trabalho como Odometria Mecânica. Este tipo de Odometria é suscetível à propagação e ao acúmulo de erros que aumenta proporcionalmente à distância percorrida (SOUZA, 2008). Mesmo assim, ainda é bastante usada devido ao seu baixo custo, precisão em pequenos trechos percorridos e alta taxa de amostragem (BORENSTEIN & FENG, 1996).

Como alternativa à Odometria Mecânica surge então a Odometria Visual (OV), que é o processo de estimar o movimento de um agente (robô, veículo, humano, por exemplo) usando apenas a entrada de uma ou múltiplas câmeras ligada a ele. A vantagem de utilizar a OV ao invés da Odometria Mecânica é que aquela não é afetada por derrapagens das rodas dos robôs em terrenos irregulares ou outras condições adversas (SCARAMUZZA & FRAUNDORFER, 2011).

Diante dos inúmeros trabalhos realizados sobre o tema Odometria Visual, percebeu-se o recente aumento do número de pesquisas dessa área utilizando sensores RGB-D's (sensores que dispõem de uma câmera com sistema de cores primárias: vermelho, verde e azul; mais um sensor de infravermelho, responsável pela captura de informação de profundidade) como o *Kinect*, para estimar a pose do robô. O fator custo x benefício tem sido um atrativo a mais para o emprego desta tecnologia. Sensores como *lasers*, por exemplo, que também são utilizados para calcular o deslocamento do robô no espaço, tem um custo maior que o *Kinect*, e *encoders* têm menor desempenho. Assim, resolve-se nesse trabalho utilizar um *Kinect* para captura de imagens 3D, que posteriormente são processadas, retomando as coordenadas e a orientação do robô.

1.1 Objetivo Geral

Diante do contexto apresentado, o objetivo desse trabalho é apresentar o desenvolvimento de um sistema de Odometria Visual que auxilia a navegação robótica em ambientes internos e externos, capaz de estimar a pose de um robô (suas coordenadas e orientação) a partir do reconhecimento de pontos em pares de imagens geradas pelo sensor *Kinect* em tempo real, utilizando, para isso, informações de imagens 2D para reconhecimento de características específicas detectadas no ambiente, e imagens 3D para coletar dados tridimensionais de tais características.

1.2 Objetivos Específicos

Para alcançar o objetivo geral descrito, alguns objetivos específicos precisaram ser atingidos:

- Inicialmente, investigar toda a teoria envolvida na Odometria Visual, analisando desde a captura de imagens a partir do sensor *Kinect* aos cálculos geométricos necessários para obter coordenadas 3D dos pontos detectados;
- Configurar o ambiente de programação com as bibliotecas *OpenNI (Open Natural Interaction)*, para manipulação do *Kinect* e *OpenCV (Open Source Computer Vision)*, para manipulação de imagens;
- Capturar pares de imagens com o uso do *Kinect*, com ajuda da biblioteca *OpenNI*;
- Conhecer os métodos de manipulação de imagens 2D e 3D geradas pelo *Kinect*, a partir da biblioteca de Visão Computacional *OpenCV*;
- Aplicar o *SURF (Speeded Up Robust Features)*, a cada par de imagens 2D, para o reconhecimento de pontos salientes das imagens;

- Realizar a correspondência de pontos identificados no par de imagens 2D através do FLANN (*Fast Approximate Nearest Neighbor Search*);
- Obter a profundidade dos pontos com correspondências conhecidas por meio de imagens 3D;
- Unir as informações extraídas das imagens geradas pela câmera *Vídeo Graphics Array* (VGA) a estimativa de profundidade obtida, para calcular o deslocamento relativo entre imagens. E, conseqüentemente, entre as posições do *Kinect* em que as imagens foram coletadas;
- Com o deslocamento relativo entre as posições do *Kinect*, calcular a posição do *Kinect* em relação ao referencial fixo no mundo;
- Testar o resultado obtido com o sistema, com base em comparações entre poses absolutas no mundo (obtidas por algum instrumento de medição métrica) e poses estimadas com o sistema.

1.3 Estrutura do Trabalho

Esse trabalho está dividido em seis capítulos: O primeiro capítulo faz uma rápida análise do problema a ser resolvido, mostrando o objetivo geral e os objetivos específicos do trabalho proposto. O segundo capítulo apresenta diversos outros trabalhos relacionados a mesma área de pesquisa. O terceiro capítulo mostra a teoria necessária por trás do desenvolvimento do sistema: localização de robôs móveis, conceitos relacionados a Odometria Mecânica e a Odometria Visual. O quarto capítulo é formado pelo escopo do sistema, onde se apresenta a arquitetura e explica o funcionamento dos equipamentos e métodos utilizados para cada etapa do desenvolvimento desse sistema (Bibliotecas OpenCV e OpenNI, e os algoritmos SURF e FLANN). O quinto capítulo está relacionado à parte de testes e análises dos resultados, verificando e comparando os dados coletados com os dados reais. O sexto capítulo contém as considerações finais sobre o sistema e as perspectivas de trabalhos futuros. Todas as partes desse trabalho estão interligadas e necessitam seguir a sequência estabelecida para um entendimento completo.

2 TRABALHOS RELACIONADOS

A inspiração para esse trabalho veio de dois relevantes artigos-tutoriais da área de Odometria Visual: (SCARAMUZZA & FRAUNDORFER, 2011) e (FRAUNDORFER & SCARAMUZZA, 2012). No trabalho de Scaramuzza & Fraundorfer (2011), eles fornecem uma pesquisa introdutória à Odometria Visual, onde é feita uma revisão histórica dos 30 primeiros anos de pesquisa nesse campo e seus fundamentos, desde as implementações *off-line* até o aparecimento de sistemas de tempo real. Fraundorfer & Scaramuzza (2012) complementa esse importante trabalho abordando o recurso da correspondência de pontos, robustez e aplicações. Fraundorfer & Scaramuzza (2012) também analisam os principais detectores e descritores de pontos e as possíveis alternativas para minimizar erros de estimação de movimento.

A Odometria Visual surgiu em 2004, quando Nister; Naroditsky; Bergen (2004) citaram tal termo em seu trabalho. Porém, o problema de estimação do deslocamento de um robô a partir de imagens, iniciou em 1980, sendo Moravec (1980) o primeiro a descrevê-lo.

A Odometria Visual é uma subárea da Visão Computacional que vem se popularizando na comunidade científica ao longo das últimas três décadas como um dos melhores métodos para se estimar o movimento de um robô no ambiente em que ele está inserido.

O reconhecimento da pose de um robô usando um conjunto de imagens (geradas por câmeras calibradas ou não-calibradas), é conhecido entre a comunidade de Visão Computacional como “estrutura a partir de movimento” (*structure from motion – SFM*). A origem disso é atribuída aos trabalhos de Longuet-Higgins (1981) e Harris & Pike (1987).

Outra importante contribuição para a área foi dada por Frahm et. al (2010), quando propôs uma otimização *off-line* sobre a estrutura final e poses da câmera (o uso do método *bundle adjustment*) a partir de um tempo de computação crescente de acordo com o número de imagens. Esse método pode ser utilizado para refinar a estimação de um local no trajeto, e assim, diminuir o erro de cálculo.

Um fato interessante é que, no início, a maioria das pesquisas realizadas na área de Odometria Visual eram incentivadas pelo programa de exploração espacial NASA Mars, no esforço de conseguir criar um robô planetário com 6 graus-de-liberdade de movimento (DoF), o que diminuiria o risco de patinagem das rodas em terrenos irregulares.

O trabalho de Moravec (1977) se destacou não apenas por apresentar o primeiro *pipeline* de “estimação de movimento” – cujo conceito é utilizado até hoje - como também

por ter descrito um dos primeiros detectores de canto proposto em 1974 por Hanna (1974), que é conhecido atualmente como “detector de canto de Moravec” (1977), um predecessor do que foi proposto por Forstner (1986) e Harris & Pike (1987), Harris & Stephens (1988). O objetivo desses métodos é realizar a detecção de cantos em imagens capturadas por câmeras e caracterizá-los por descritores (informações relacionadas às propriedades das imagens, por exemplo, cor, textura e forma).

Vários trabalhos como os de Matthies & Shafer (1987) e Matthies (1989) foram construídos sobre o método de procedimento para detecção e rastreamento de cantos de Moravec. A diferença é que ao invés de usarem um escalar para representação das incertezas como foi feito por Moravec, eles aproveitaram as vantagens da matriz de covariância de erro, fizeram a triangulação de características e incorporaram isso ao processo de estimação de movimento. Depois disso, Oslon et. al (2000) introduziram um sensor de orientação absoluta e mostraram que o acúmulo de erro gerado nos trabalhos de Matthies & Shafer (1987) e Matthies (1989) pode ser reduzido por uma função linear.

Um abordagem similar a de Oslon et. al (2000) foi dada por Lacroix et. al (1999), onde eles implementaram um sistema estéreo de Odometria Visual para um robô planetário. Ao invés de usar o detector de Forstner, Lacroix et. al (1999) usaram uma mapa de disparidade denso (estrutura contendo vasta quantidade de informações de profundidade) para selecionar os candidatos a pontos-notáveis, analisando a função de correlação em torno de seu pico. Essa escolha de Lacroix et. al (1999) veio da observação que existia uma forte correlação entre a forma da curva de correlação e o desvio padrão das características de profundidade. Cheng; Maimores; Matthies (2006) e Maimone; Cheng; Matthies (2007), usaram essa observação para a implementação final do sistema de Odometria Visual *onboard* do robô marte, onde usou o detector de cantos de Harris e a curvatura da função de correlação em volta das características, e depois, como proposto por Nister; Naroditsky; Bergen (2004), eles usaram o *RANdom SAmple Consensus* (RANSAC) do trabalho de Fischler & Bolles (1981), na detecção de erros de estimação do movimento (FRAUNDORFER & SCARAMUZZA, 2012).

Outros trabalhos de Odometria Visual baseados em sistema de uma única câmera foram descritos por Corke; Strelow; Singh (2005), Lhuillier (2005) e Tardif; Pavlidis; Danilidis (2008). Corke; Strelow; Singh (2005) oferecem uma abordagem para Odometria Visual baseada em imagens de uma câmera omnidirecional catadrióptica e óptica. Tardif; Pavlidis; Danilidis (2008) apresentam uma abordagem para implementar em um sistema de

Odometria Visual de um carro. Eles dissociaram a rotação e a estimativa de translação, e conseguiram estimar a rotação usando pontos no infinito de um mapa 3D recuperado.

Milford & Wyeth (2008) apresentaram um método para extrair as informações sobre a velocidade aproximada de rotação e translação de uma única câmera montada sobre um carro.

Recentemente têm surgido várias pesquisas na área de Odometria Visual que utilizam sensores RGB-D's ou *smartphones*. Kerl; Sturm; Cremers (2012), por exemplo, oferecem um método rápido, e com acurácia, para estimar o movimento de uma câmera a partir de imagens RGB-D's. Tal método é baseado em uma função de correção de erros fotométricos empregada em sistemas de tempo real. Assim como Kerl; Sturm; Cremers (2012), Huang et. al (2011) também utiliza utilizam este método para estabilizar Quadricópteros durante o voo.

O trabalho de Schöps; Engel; Cremers (2014) apresenta um sistema de Odometria Visual monocular de tempo real para *smartphones*, que consegue mapear uma superfície e simular o comportamento de um carro sobre ela, com o uso da realidade aumentada.

3 FUNDAMENTAÇÃO TEÓRICA

3.1 Localização de Robôs Móveis

O problema da localização de um robô, no ambiente em que ele está inserido, consiste basicamente em estimar o ponto onde o mesmo está, ou seja, computar as suas coordenadas e orientação (ou a pose do robô), tendo por base um referencial fixo, podendo ser, na prática, o ponto de partida do robô no ambiente. Os métodos de localização existentes podem ser classificados em três categorias: Localização Relativa, Localização Absoluta e Fusão de Multi-sensores que se trata da junção das localizações Relativa e Absoluta (SOUZA, 2008).

A Localização Relativa, também chamada de *dead reckoning*, baseia-se nas localizações em instantes anteriores, onde as medidas se deduzem por integração, o que favorece a propagação e acúmulo de erros. A estimativa da pose de um robô por meio das abordagens de Localização Relativa pode ser feita por dois métodos: Odometria Mecânica ou Navegação Inercial. A Odometria Mecânica determina a localização pela integração incremental do movimento das rodas, calculando o deslocamento linear e a orientação do robô através do uso de *encoders* (sensores de rotação). Enquanto a Navegação Inercial se baseia nas informações de giroscópios e acelerômetros para medir a orientação e a aceleração (SOUZA, 2008).

Já a Localização Absoluta fornece informação sobre a localização do robô independente de localizações previamente calculadas. A localização atual não é derivada de integrações, mas derivada diretamente de uma medição, onde a dedução da pose do robô pode ser feita por três métodos: a utilização de Faróis Ativos, Marcas no ambiente (naturais ou artificiais) e Mapas (NEGENBORN, 2003).

Os Faróis Ativos são dispositivos que ativamente podem enviar informações, que são utilizadas em um processo de triangulação para inferir uma localização, como o *Global System Position* (GPS), por exemplo. As Marcas do ambiente podem ser naturais, como: portas, janelas, quinas, luz no teto, etc.; e artificiais, como: código de barras, figuras geométricas coloridas, entre outras. Além disso, a utilização de Mapas também é um método de localização comumente usado, onde medições sensoriais são adquiridas e comparadas com as informações de um mapa previamente adquirido (SOUZA, 2008).

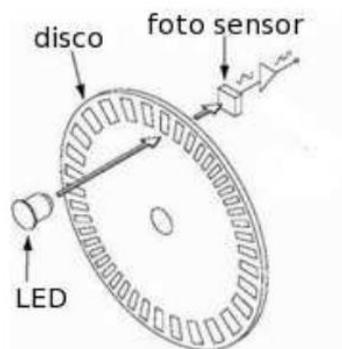
3.2 Odometria Mecânica

A Odometria Mecânica, baseada na integração incremental do movimento das rodas medido através da contagem dos pulsos gerados por *encoders*, sofre ruídos que afetam a estimativa de deslocamento. Esses ruídos produzem erros que são acumulativos, e conseqüentemente levam a uma interpretação equivocada do ponto onde o robô está inserido no ambiente e, por conseguinte, afeta a percepção que o robô tem do seu ambiente. Para melhor entendimento, faz-se necessário um estudo sobre o funcionamento dos *encoders*, de como se calcula o deslocamento a partir de suas informações, e quais os possíveis erros que podem afetar a acurácia de um sistema de Odometria Mecânica.

3.2.1 Encoders

Os *encoders* são sensores ópticos acoplados ao eixo das rodas do robô, cujo princípio de funcionamento é a transmissão de luz, de um LED para um foto sensor, através de um disco perfurado. Assim, é possível calcular a velocidade de rotação da roda a partir da frequência de pulsos gerados e medir o deslocamento do robô no espaço (ver Figura 3.1).

Figura 3.1: Princípio de funcionamento do *encoder*.



Fonte: (SOUZA, 2008).

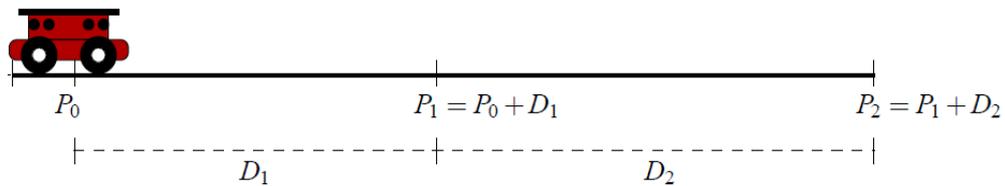
3.2.2 Cálculo do deslocamento

Para se estimar a localização de um robô baseada na odometria, é necessário a integração dos deslocamentos incrementais das rodas, tendo um referencial fixo como ponto

de partida. Considerando um robô que se desloca de forma linear em seu ambiente conforme certa trajetória (ver Figura 3.2), temos:

- P_0 – ponto inicial em que o robô está localizado;
- P_1 – um ponto qualquer seguinte ao P_0 ;
- P_2 – a posição final do robô em sua trajetória;
- D_1 – a distância percorrida do ponto inicial (P_0) ao ponto P_1 ;
- D_2 – a distância percorrida do ponto P_1 ao ponto final, P_2 .

Figura 3.2: Deslocamento dado pela odometria.



Fonte: (SOUZA, 2008).

Inicialmente, o robô se encontra no ponto inicial P_0 , depois ele chega ao ponto P_1 em seu trajeto. O ponto P_1 pode ser calculado somando-se ao ponto anterior (P_0) o deslocamento realizado D_1 . Continuando sua trajetória, o robô chega ao ponto P_2 , a qual pode ser calculada somando-se ao P_2 o novo deslocamento D_2 . Assim, a localização final é calculada com base nos deslocamentos realizados em relação à localização inicial (SOUZA, 2008).

3.2.3 Erros Sistemáticos e Não Sistemáticos

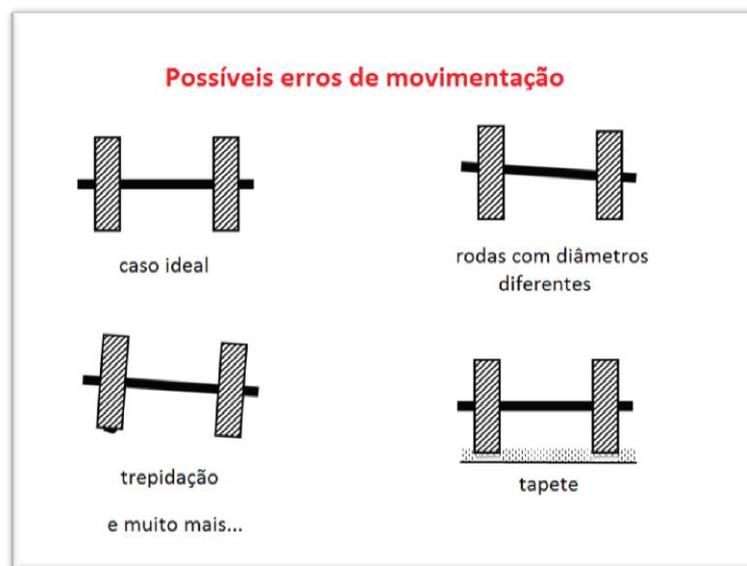
Pela presença de fatores sistemáticos e não-sistemáticos, os valores calculados pela Odometria Mecânica possuem erros que se acumulam durante a movimentação do robô, de forma proporcional à distância percorrida. Um erro causado em um determinado instante se propaga para os instantes seguintes, afetando estimativas futuras.

Os erros sistemáticos são devidos às incertezas nos parâmetros que fazem parte do modelo cinemático do robô (diâmetro das rodas diferentes, comprimento do eixo diferente do nominal, taxa de amostragem finita com *encoders*, etc.). Já os erros não-sistemáticos são

devidos a situações inesperadas (objetos inesperados, escorregamento das rodas, etc.) (SANTANA, 2007).

Como dito anteriormente, os *encoders* são acoplados ao eixo das rodas do robô. Isso significa que, caso as rodas estejam suspensas, ou derrapem, ou por algum outro motivo, são impedidas de rotacionar como deveriam, isso implicará em um erro de estimativa de deslocamento, visto que as rodas não estarão percorrendo a distância “percebida” pelos *encoders*. A Figura 3.3 mostra alguns dos possíveis casos de origens de erros.

Figura 3.3: Fontes de erros no cálculo do deslocamento por odometria mecânica.



Fonte: Adaptada de <http://www.probabilistic-robotics.org>

A maioria dos erros sistemáticos podem ser reduzidos com investimentos em estudos mais elaborados sobre o desenho mecânico do robô e com a calibração de seus sensores. Porém, os erros não-sistemáticos não podem ser estimados com exatidão, por terem a característica de serem imprevisíveis (BORENSTEIN & FENG, 1995).

Contudo, nesse trabalho, a estimativa da localização do robô é feita pelo método de Odometria Visual. Embora haja semelhança de objetivos da Odometria Visual com o método convencional de Odometria Mecânica, a primeira se baseia em marcas naturais do ambiente para estimar a localização de um robô (ou agente). Essas marcas são chamadas de características ou pontos notáveis (ou ainda, do inglês, *features*), e através de uma correlação entre elas, em uma sequência de imagens capturadas em instantes diferentes, é possível estimar o deslocamento realizado pelo robô. Essa estimativa de deslocamento pode ser

concatenada à pose anterior, para calcular a pose atual. A Odometria Visual não sofre com os erros apresentados na Odometria Mecânica, o que pode ser visto como uma vantagem.

3.3 Odometria Visual

A Odometria Visual é o processo de estimar o deslocamento de um agente (veículo, humano, ou robô, por exemplo) através de imagens sequenciais geradas por uma ou múltiplas câmeras anexadas ao agente (neste trabalho foi utilizado um *Kinect*). A estimativa da pose é feita de forma incremental, através da análise dos efeitos que os movimentos do agente provocam nas imagens capturadas por suas câmeras. Para isso, é necessário que o ambiente seja suficientemente iluminado e que possua texturas que possam ser facilmente diferenciadas, permitindo com que movimentos aparentes entre imagens sejam extraídos (SCARAMUZZA & FRAUNDORFER, 2011).

O termo Odometria Visual foi concebido por Nister; Naroditsky; Bergen (2004), por se tratar de um método com princípios similares aos da Odometria Convencional (a Odometria Mecânica). Uma importante diferença entre os dois métodos é que a Odometria Visual não é afetada com derrapagens de rodas ou com efeitos causados pela irregularidade de qualquer terreno.

Há vários domínios de aplicação da Odometria Visual, entre eles estão: a robótica, a realidade aumentada, a indústria automobilística, os correios aéreo (por *drones*), a vigilância ambiental, defesa, entre outros.

Durante o desenvolvimento das técnicas de Odometria Visual, percebeu-se o progresso mais evidente de duas abordagens de linhas de pesquisas: sistemas baseados em visão estéreo e sistemas baseados em de visão monocular (SCARAMUZZA & FRAUNDORFER, 2011).

3.3.1 Sistemas Estéreo

O ser humano, assim como alguns animais, possui a chamada visão binocular ou visão estéreo. A visão estéreo é caracterizada pela obtenção das características tridimensionais dos objetos presente no ambiente, através da captação das imagens por cada olho projetadas sobre as duas retinas. Como os olhos possuem certa distância um do outro, no caso do ser humano

aproximadamente 65 milímetros, possibilita obter duas imagens de um mesmo ponto no espaço de perspectivas diferentes (KIRNER & TORI, 2004).

Assim como o sistema de visão humana, sistemas de câmeras estéreo partem do mesmo princípio. Com eles, é possível se obter a profundidade (ou coordenadas 3D) de uma cena de um ambiente, de forma direta, a partir de duas imagens 2D capturadas de câmeras posicionadas em pontos diferentes (REINALDO, 2012).

A maioria das pesquisas feitas em Odometria Visual têm sido produzidas usando sistemas de câmeras estéreo. Existem vários trabalhos realizados com sistema binocular na literatura, como os de Matthies & Shafer (1987), Matthies (1989), Oslon et. al (2000), Milella e Siegwart (2006) e Howard (2008). A abordagem mais comum de Odometria Visual com visão estéreo faz o uso de triangulação para cada par de imagens estéreo, com o fim de se descobrir a profundidade. O movimento relativo entre pares tomados em posições diferentes (ou deslocamento), passa a ser estimado pela solução de um problema de alinhamento 3D-3D.

3.3.2 Sistemas Monocular

Segundo Kirner & Tori (2004), a visão monocular é uma forma primitiva e rudimentar de se ver o mundo, tendo como fundamento as leis da perspectiva. Na Odometria Visual baseada em sistemas monoculares, o deslocamento no mundo 3D é calculado a partir de dados o 2D. Uma vez que a escala absoluta é desconhecida (pela falta de informações métricas 3D do mundo), a distância absoluta (em unidades métricas) entre as primeiras poses da câmera é, geralmente, definida (ou adotada) como uma medida padrão de escala. Assim, os novos cálculos de deslocamento são feitos com base nessa escala definida.

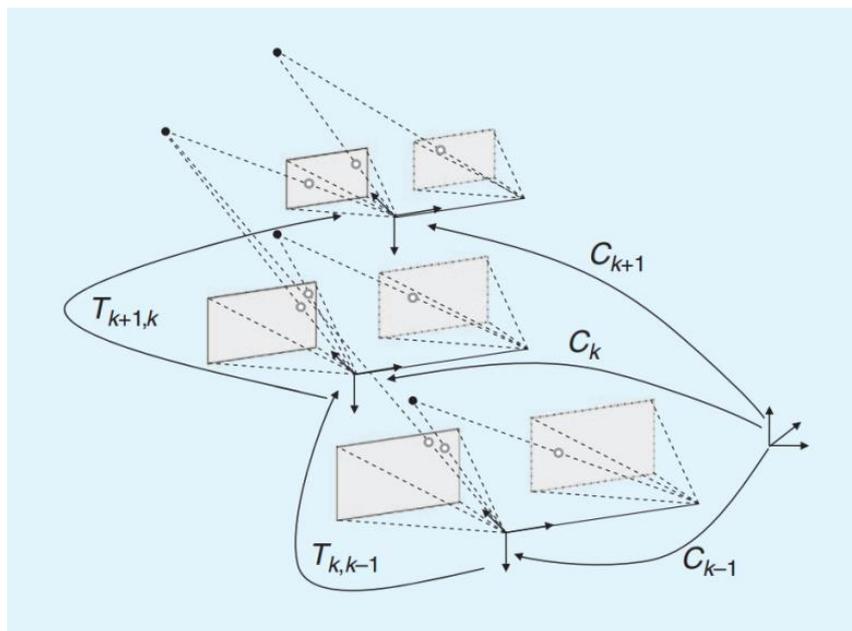
Trabalhos relacionados a esta abordagem podem ser divididos em três categorias: métodos baseados em características, métodos baseados em aparência e métodos híbridos. Métodos baseados em características são métodos que se baseiam na saliência e na repetição das características (pontos notáveis nas imagens), que são marcadas em cada nova imagem; métodos baseados em aparência usam a informação de intensidade de todos os *pixels* numa imagem ou sub-região dela; e métodos híbridos usa a combinação das duas primeiras (FRAUNDORFER & SCARAMUZZA, 2012).

3.3.3 Formulação da Odometria Visual

Neste trabalho será adotada a explicação exposta no trabalho de Scaramuzza & Fraundorfer (2011). Para estimar a posição de um agente (robô, veículo, entre outros) no espaço, através da Odometria Visual, considera-se que sua pose inicial tenha as coordenadas x , y , e z , inicializadas de acordo com as necessidades do sistema. Para simplificar, assume-se que o sistema de coordenadas (ou *frame*) da câmera e do agente os coincidente, assim a pose da câmera é a mesma do agente.

Ao movimentar-se pelo ambiente, o agente utiliza um sistema de câmera(s) (anexado a ele) para capturar imagens em instantes discretos de tempo k . No caso de um sistema monocular, o conjunto de imagens I feitas no tempo k é definido por $I_{0:n} = \{I_0, \dots, I_n\}$. Enquanto que em sistemas estéreos, tem-se dois conjuntos de imagens que representam as imagens da esquerda e as da direita, a cada instante de tempo, como definido por $I_{l,0:n} = \{I_{l,0}, \dots, I_{l,n}\}$ e por $I_{r,0:n} = \{I_{r,0}, \dots, I_{r,n}\}$ respectivamente. A Figura 3.4 mostra uma ilustração da obtenção desses conjuntos por um sistema de visão estéreo, com o cálculo dos movimentos (transformações homogêneas $T_{k,k-1}$ e $T_{k+1,k}$) e das poses (C_{k-1} , C_k e C_{k+1}).

Figura 3.4: Representação do processo de estimação de pose e deslocamento, a partir da captura de pares de imagens com um sistema de câmeras estéreo.



Fonte: Scaramuzza & Fraundorfer (2011).

Para simplificar, as coordenadas dos *frames* da câmera são assumidas como as coordenadas do agente. No caso de um sistema estéreo, as coordenadas da câmera esquerda podem ser usadas como as de origem.

Duas posições de câmera em instantes de tempo adjacentes $k - 1$ e k são relacionadas por uma transformação homogênea $T_{k,k-1} \in R^{4 \times 4}$ seguindo o modelo (Equação (1)):

$$T_{k,k-1} = \begin{bmatrix} R_{k,k-1} & t_{k,k-1} \\ 0 & 1 \end{bmatrix}, \quad (1)$$

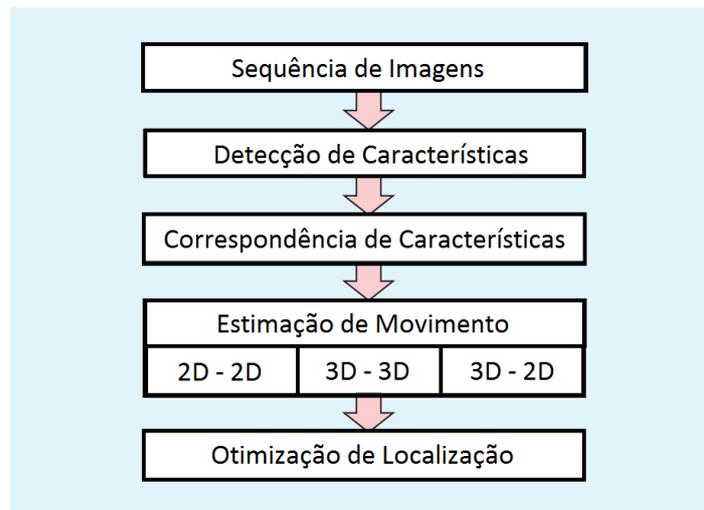
Onde $R_{k,k-1}$ é uma matriz de rotação, e $t_{k,k-1} \in R^{3 \times 1}$ um vetor de translação. O conjunto $T_{1:n} = \{T_{1,0}, \dots, T_{n,n-1}\}$ contém todos os movimentos subsequentes realizados pelo agente. Para simplificar ainda mais a notação, a partir de agora, T_k será usada em vez de $T_{k,k-1}$ (Equação (2)).

$$T_k = \begin{bmatrix} R_k & t_k \\ 0 & 1 \end{bmatrix} \quad (2)$$

Com isso, o conjunto de poses de câmera $C_{0:n} = \{C_0, \dots, C_n\}$ contém as transformações em relação ao sistema de coordenadas iniciais no instante $k=0$. A pose atual C_n pode ser calculada pela concatenação de todas as transformações $T_k = (k = 1 \dots n)$ e, portanto, $C_n = C_{n-1}T_n$, com C_0 sendo a pose da câmera no instante $k=0$, o que também pode ser definido pelo próprio usuário do sistema.

O principal desafio da Odometria Visual é saber como calcular as transformações T_k de uma imagem I_k e I_{k-1} e, depois, concatená-las para recuperar a trajetória completa $C_{0:n}$ da câmera. Isso significa que a Odometria Visual incrementa a estimativa do caminho, pose a pose. O funcionamento da Odometria Visual pode ser resumido na Figura 3.5.

Figura 3.5: Passo a passo para se realizar odometria visual.



Fonte: Adaptado de Scaramuzza & Fraundorfer (2011).

Para cada nova imagem I_k (ou par de imagens, no caso de sistemas estéreo) capturada, o primeiro passo é detectar pontos notáveis na imagem. Depois, faz-se a correspondência dessas características com as da imagem anterior. Ou seja, verifica-se quais pontos notáveis são comuns às imagens I_{k-1} e I_k , constituindo a criação dos conjuntos de características com correspondências, X_{k-1} e X_k .

O quarto passo consiste em calcular a movimentação T_k entre os instantes de tempo $k-1$ e k . Dependendo se as correspondências são especificadas em tri ou bidimensional, existem diversos métodos para superar esse problema (a subseção Estimação do Movimento se aprofunda mais nessa questão). Posteriormente, é comum se executar um passo de otimização, que tem como função melhorar as estimativas da movimentação calculada no passo anterior, para, por fim, ter o cálculo da pose.

3.3.4 Estimação do Movimento

A estimação do movimento é a base do cálculo para um sistema de Odometria Visual. Nessa etapa calcula-se o movimento da câmera, considerando a imagem atual e a imagem anterior, para depois concatená-lo à trajetória do agente, e assim, poder estimar sua pose. Essa sessão explica como a transformação T_k entre duas imagens I_{k-1} e I_k pode ser calculada a partir de dois conjuntos de características (pontos-notáveis das imagens) correspondentes

X_{k-1} , X_k , obtidos de I_{k-1} e I_k , nos instantes de tempo $k-1$ e k , respectivamente. Existem três diferentes métodos para isso, dependendo se as correspondências de características são especificadas em duas ou três dimensões (SCARAMUZZA & FRAUNDORFER, 2011):

- 2D – 2D: nesse caso, ambas X_{k-1} e X_k são especificadas em coordenadas de imagem 2D.
- 3D – 3D: nesse caso, ambas X_{k-1} e X_k são especificadas em 3D. Para fazer isso, é necessária uma triangulação dos pontos 3D a cada instante de tempo;
- 3D – 2D: nesse caso, X_{k-1} são especificadas em 3D e X_k corresponde a reprojeção delas em 2D sobre a imagem I_k .

No trabalho de Scaramuzza & Fraundorfer (2011) existe uma explicação geral desses três métodos para o cálculo de estimação do movimento. Neste trabalho será apresentada a estimação de movimentos a partir de características com coordenadas 3D, ou seja, foi utilizado o método 3D-3D, devido às propriedades do sensor utilizado, o *Kinect*, o qual permite a derivação das coordenadas tridimensionais de cada característica.

Movimentação a partir de características correspondentes 3D

Para correspondência de características 3D, o movimento da câmera T_k pode ser calculado através da determinação do alinhamento dos dois conjuntos de características 3D, X_{k-1} e X_k . É necessário que esse conjunto de características possuam no mínimo três pontos cada um. Esse tipo de correspondência é possível com o uso de câmeras estéreo ou de um dispositivo que estime a profundidade das cenas capturadas, é o caso dos sensores RGB-D.

Nesse trabalho, utilizou-se um sensor RGB-D *Kinect* para estimar o movimento do robô (câmera). Para melhor entendimento sobre este assunto, toma-se como exemplo duas imagens, I_{k-1} e I_k , capturas em instantes diferentes pelo *Kinect*, com seus respectivos conjuntos de características, X_{k-1} e X_k , detectadas a partir da análise e processamento das duas imagens. Considerando que o agente se movimentou no espaço entre o instante $k-1$ (instante em que capturou a imagem I_{k-1}) e k (em que capturou a imagem I_k), consegue-se obter o deslocamento relativo entre os dois instantes, a partir da computação da matriz de transformação homogênea T_k , que alinha os dois conjuntos de dados (ou características) correspondentes. T_k pode ser obtida pela Equação (2).

Para isso, é necessário seguir três passos:

1. encontrar os centroides dos conjuntos de características 3D encontradas;
2. trazer os dois conjuntos de dados X_k e X_{k-1} para a origem e em seguida, encontrar a rotação (matriz R_k);
3. encontrar o vetor de translação t_k .

- **Passo 1: Cálculo dos centroides**

Os centroides são os pontos centrais de um conjunto de pontos e podem ser calculados pelas Equações (3) e (4):

$$\text{centroide}_{k-1} = \frac{1}{N} \sum_{i=1}^N X_{k-1}^i \quad (3)$$

$$\text{centroide}_k = \frac{1}{N} \sum_{i=1}^N X_k^i \quad (4)$$

Onde X_{k-1}^i e X_k^i são pontos nos conjuntos X_{k-1} e X_k respectivamente. Esses valores dos centroides serão úteis no próximo passo. Os centroides são vetores colunas 3×1 .

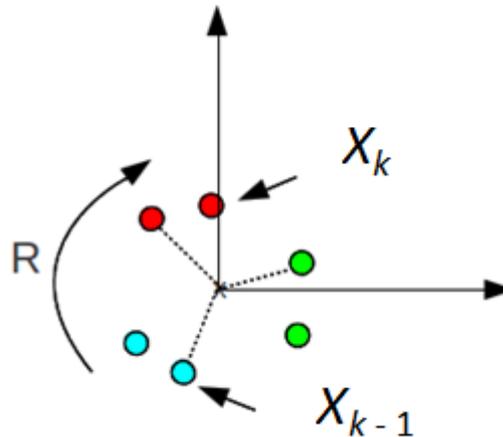
- **Passo 2: Cálculo da matriz de rotação**

A matriz de rotação R_k pode ser computada através do método *Singular Value Decomposition* (SVD), onde uma matriz $M_{n \times m}$ é decomposta em três matrizes, sendo duas ortogonais, $U_{n \times n}$ e $V_{n \times n}$ (matrizes de rotação), e uma matriz diagonal $S_{n \times n}$ (matriz de escala).

A SVD é uma função matemática muito utilizada, que está disponível na biblioteca OpenCV (OPENCV, 2014b) e em outras ferramentas de computação numérica (como Scilab e MATLAB), para implementações que necessitem dessa operação matricial na resolução de problemas numéricos.

Para achar a rotação é preciso primeiro centralizar os conjuntos de dados X_{k-1} e X_k (características com correspondências conhecidas) de modo que ambos os centroides estejam na origem de um sistema de coordenadas comum. A Figura 3.6 mostra os dois conjuntos de dados, sendo X_{k-1} derivado da imagem I_{k-1} e X_k derivado da imagem I_k . Deve-se encontrar uma rotação R , que ao ser aplicada ao conjunto X_{k-1} , resulte no conjunto X_k .

Figura 3.6: Modelo de Rotação.



Fonte: (NGHIA HO, 2014).

O próximo passo é a computação da matriz de covariância H , através da Equação (5).

$$H = \sum_{i=0}^N (X_{k-1}^i - \text{centroide}_{k-1}) (X_k^i - \text{centroide}_k)^T \quad (5)$$

Depois do cálculo da matriz de covariância, é aplicado o método SVD sobre a mesma para encontrar a matriz de rotação R_k , como mostram as Equações (6) e (7) à seguir:

$$[U, S, V] = \text{SVD}(H) \quad (6)$$

$$R_k = VU^T \quad (7)$$

O resultado da Equação (7) deve ser uma matriz 3x3.

- **Passo 3: Cálculo do vetor de translação t_k**

Para calcular o vetor de translação basta utilizar a matriz de rotação calculada e os centroides dos conjuntos de características X_{k-1} e X_k . Veja a Equação (8), à seguir:

$$t_k = -R * \text{centroide}_{k-1} + \text{centroide}_k \quad (8)$$

Assim, é possível se obter a matriz de rotação e o vetor translação relativos ao movimento de um agente. Porém, é aconselhável o uso de cálculos de otimizações para

diminuir possíveis erros do sistema (NGHIA HO, 2014). Um aprofundamento na matemática por trás desta solução é dado por Besl & Mckay (1992).

A computação da pose atual C_k do agente pode ser dada pela concatenação da transformação T_k , que traduz o deslocamento entre os instantes $k - 1$ e k , com a pose C_{k-1} calculada no instante $k - 1$, como mostra a Equação (9).

$$C_k = C_{k-1}T_k \quad (9)$$

O Algoritmo 1 resume os passos do processo de Odometria Visual com correspondências 3D – 3D.

Algoritmo 1: Odometria Visual a partir de Correspondências 3D – 3D, considerando um sistema de visão estéreo.

- 1) Capture dois pares de imagens $I_{l,k-1}, I_{r,k-1}$ e $I_{l,k}, I_{r,k}$
- 2) Extrair e corresponder características entre $I_{l,k-1}$ e $I_{r,k-1}$
- 3) Fazer a triangulação dos pontos correspondidos para cada par de imagens estéreo
- 4) Calcular T_k de características 3D X_{k-1} e X_k
- 5) Concatenar a transformação através do cálculo $C_k = C_{k-1}T_k$
- 6) Repetir do passo 1.

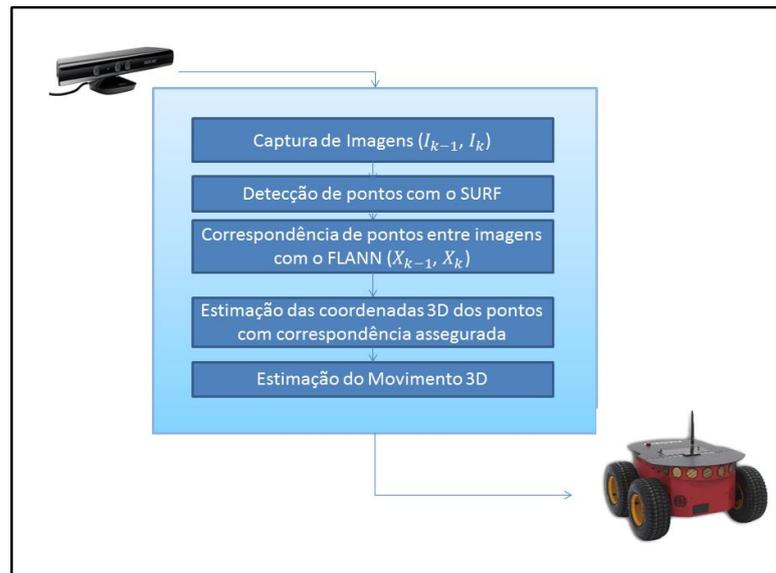
A matemática explicada acima se refere ao uso de sistemas de câmeras estéreo, porém, as características do *Kinect* possibilitam que essa formulação seja adotada para esse dispositivo. Dessa forma, a implementação do sistema de Odometria Visual neste trabalho está baseada nos cálculos matemáticos descrito anteriormente.

4 SISTEMA IMPLEMENTADO

4.1 Metodologia Adotada no Desenvolvimento do Sistema

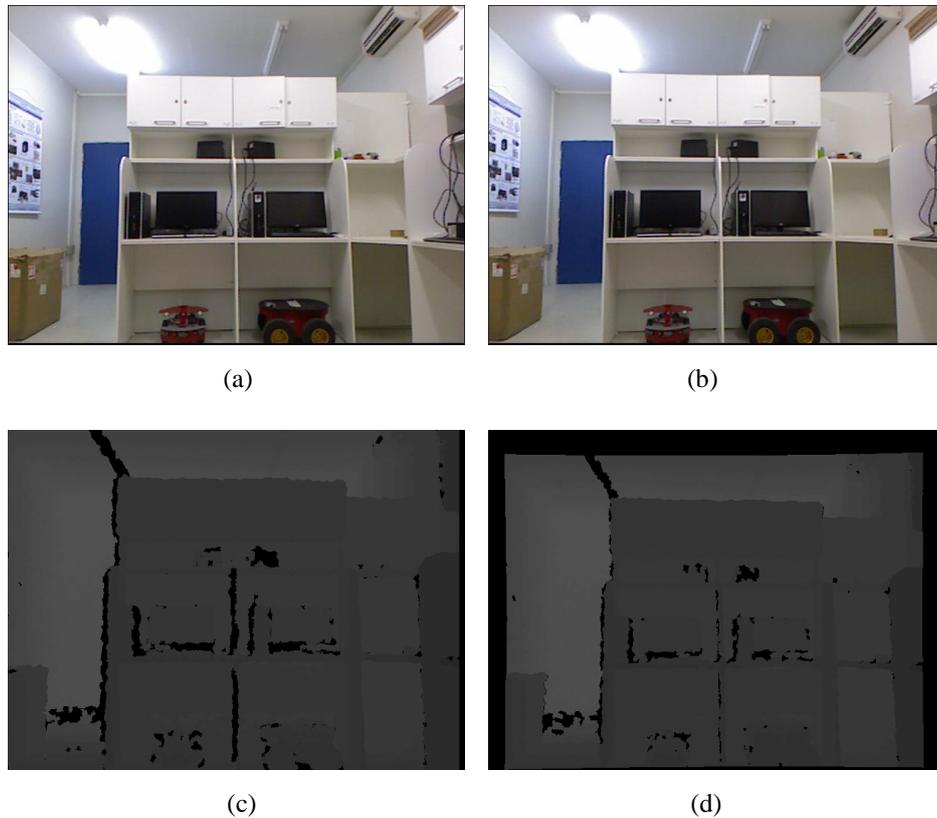
A Figura 4.1 apresenta a arquitetura do sistema de Odometria Visual 3D implementado, e mostra o funcionamento principal do sistema.

Figura 4.1: Arquitetura do Sistema de Odometria Visual 3D Implementado.



Ao iniciar o sistema, são capturadas duas imagens em instantes diferentes de tempo. Cada imagem possui seu próprio mapa de profundidade. Os mapas de profundidade são estruturas de dados que guardam valores da distância entre o sensor e os objetos que fazem parte do ambiente (ver Figura 4.2). Nesse trabalho as imagens foram geradas pelo sensor RGB-D *Kinect*.

Figura 4.2: Imagens 2D I_{k-1} (a) e I_k (b) e os respectivos mapas de profundidade da imagem I_{k-1} (c) e da imagem I_k (d).



O mapa de profundidade é detectado pelo sensor de infravermelho simultaneamente à captura de uma imagem 2D pela câmera VGA, ambos do Kinect. A profundidade é representada pelos tons de cinza, quanto mais próximo o objeto do sensor, mais escura é sua imagem.

Posteriormente, é feita a detecção de características aplicando o algoritmo *Speeded Up Robust Features* (SURF) sobre as imagens 2D (ver Figura 4.3). Em seguida, é realizada a correspondência de pontos detectados em ambas as imagens, com o algoritmo FLANN (ver Figura 4.4), configurando-se na criação dos conjuntos X_{k-1} e X_k . Em geral, nem todas as correspondências realizadas são coerentes, por isso, é feita uma triagem para identificar apenas as boas correspondências.

Figura 4.3: Aplicação do detector de pontos SURF sobre a imagem I_{k-1} (à esquerda) e I_k (à direita).



Figura 4.4: Aplicação do algoritmo de FLANN para correspondência de pontos notáveis entre as imagens I_{k-1} e I_k . Os pontos com correspondentes em ambas as imagens são utilizados para a criação dos conjuntos X_{k-1} e X_k .



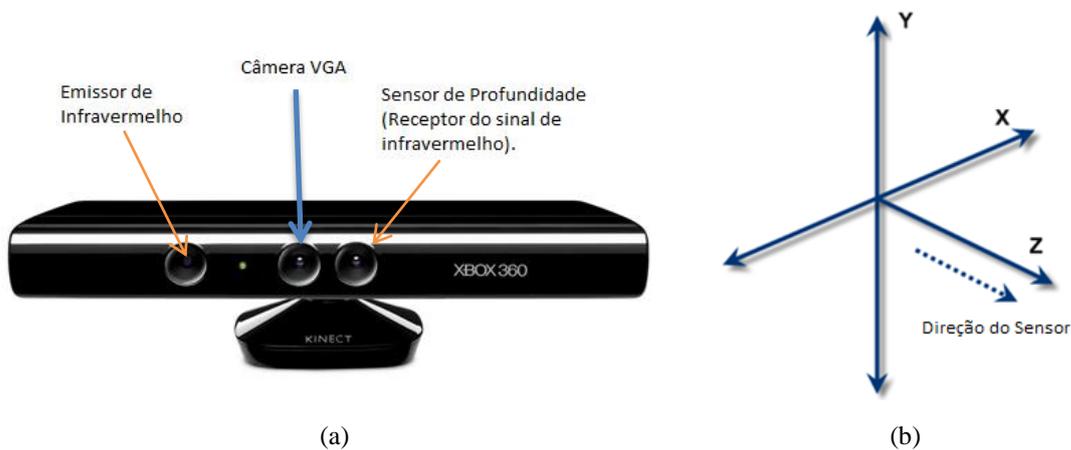
Após, segue-se com o cálculo das coordenadas 3D das características com correspondência asseguradas, através do uso das informações dos mapas de profundidades. Com essas coordenadas, faz-se a estimação do movimento realizado com base nos cálculos mostrados na seção 3.3.4, do capítulo anterior.

4.2 O Sensor RGB-D *Kinect* e Demais Equipamentos Utilizados

Para realizar este trabalho, optou-se pelo *Kinect* por ser um sensor RGB-D robusto, que tem melhor desempenho do que câmeras em ambientes sem boa iluminação (HOLZ et. al, 2012), e permite obter ricas informações de cenas, por meio de uma câmera VGA e um sensor infravermelho. Enfim, o fator *custo x benefício* é um atrativo para a adoção desse sensor, que hoje em dia é amplamente utilizado em pesquisas na área de percepção de veículos autônomos. A Figura 4.5 representa o sensor RGB-D *Kinect*, que possui especificações:

- Ângulo do campo de visão é de 43° na vertical e 57° na horizontal;
- Intervalo máximo de inclinação de $\pm 27^\circ$;
- 30 quadros por segundo;
- Uso de 4 microfones embutidos para captura de áudio no formato de 16 kHz, trabalhando com 24-bit para modulação por código de pulso (PCM);
- Um acelerômetro 8G configurado para a faixa de 2G, com limite máximo de 1° de precisão;
- Intervalo padrão de distância em que o infravermelho pode identificar objetos é de 0,8 a 4 metros.

Figura 4.5: Sensor RGB-D Kinect (a) e seu sistema de coordenadas (b).



Fonte: msdn.microsoft.com

A Tabela 1 mostra os demais equipamentos de *hardware* e ferramentas de *software* utilizados para desenvolvimento da aplicação, juntamente com suas especificações.

Tabela 1: Equipamentos utilizados para o desenvolvimento do sistema.

Hardware		Software	
<i>Kinect</i>	<i>30 frames por segundo</i> Resolução máxima: 1280 x 960 Sensores: Câmera VGA e Infravermelho	Sistema Operacional: <i>Linux</i>	Distribuição: <i>Ubuntu 12.04</i>
Computador: <i>HP Compaq Presario CQ40-314BR Notebook PC</i>	Processador: Core 2 Duo Intel ~1.6GHz Memória RAM: 4 Giga bytes, 800MH-DDR2 Placa de Video: Intel Graphics Media Accelerator 4500MHD.	Biblioteca: OpenCV (<i>Open Source Computer Vision</i>)	<i>Versão 2.4.9</i>
		Biblioteca: OpenNI (<i>Open Natural Interaction</i>)	<i>Versão 1.5.7</i>

Fonte: msdn.microsoft.com/en-us/library/hh973078.aspx#Depth_Ranges

4.3 Bibliotecas OpenCV e OpenNI

A OpenCV foi desenvolvida inicialmente pela equipe da Intel, que posteriormente veio a liberá-la sobre a licença *BSD*, portanto, é gratuito o seu uso tanto acadêmico como comercial. A OpenCV é uma biblioteca de visão computacional *Open Source* escrita em C e C++ que roda em diversos sistemas operacionais como o Windows, Mac OS X, distribuições Linux, e que possibilita o desenvolvimento de interfaces para Python, Ruby, Matlab, entre outras linguagens (BRADSKI & KAEHLER, 2008).

A OpenCV é composta por diversos módulos funcionais de manipulação de imagens, sendo reconhecida como uma das melhores bibliotecas para se trabalhar com visão computacional e processamento de imagens (REINALDO, 2012).

Nesse trabalho, a OpenCV teve um papel fundamental em seu desenvolvimento, auxiliando na captura de imagens, no reconhecimento de características, na correspondência

de pontos e no cálculo das coordenadas. Além disso, a biblioteca oferece um grande número de documentos e exemplos de implementações, o que facilitou o entendimento das funcionalidades e diminuiu o tempo de desenvolvimento.

Para trabalhar com o sensor RGB-D Kinect, precisou-se usar a biblioteca OpenNI que é um *Software Development Kit (SDK) Open Source* utilizado para o desenvolvimento de aplicações e de sensoriamento 3D. Em conjunto com a OpenCV, ela permite a entrada de dados por sensores externos RGB-D's, como o *PrimerSensor*¹ ou o *Kinect*, para posterior processamento (OPENNI, 2014).

4.4 Algoritmo de SURF

Na Odometria Visual utilizam-se detectores para o reconhecimento de pontos notáveis nas imagens do ambiente, também chamados de características, que podem ser cantos, quinas, bolhas, retas, entre outras. As características mais comumente utilizadas nessas aplicações são cantos e bolhas. Um canto é definido como um ponto de intersecção de duas ou mais esquinas, enquanto que bolhas são amostras de uma imagem que diferem de regiões vizinhas em termos de intensidade, cores e textura. Entretanto, o reconhecimento de cantos por detectores acontece em menor escala do que bolhas (FRAUNDORFER & SCARAMUZZA, 2012).

Segundo Fraundorfer & Scaramuzza (2012), um bom detector de pontos deve ser preciso quanto à localização (posição e escala) e capaz de redetectar os mesmos pontos nas imagens seguintes. O *Speeded-Up Robust Features (SURF)* é um detector de pontos (bolhas) construído sobre um popular descritor de pontos chamado *Scale Invariant Feature Transform (SIFT)*. Descritores de pontos são detectores, que além de reconhecerem regiões de uma imagem a partir da intensidade dos pixels, consegue descrever a orientação e escala de cada ponto. Embora a resposta dos descritores seja mais rica, este método requer um alto custo computacional, ao contrário dos detectores, que são mais rápidos (FRAUNDORFER & SCARAMUZZA, 2012).

É possível encontrar muitos trabalhos sobre Odometria Visual na literatura que abordam os mais diversos detectores de pontos, tanto detectores de cantos (MORAVEC (1980), FORSTNER (1986), HARRIS & PIKE (1987), TOMASI, SHI (1994), e ROSTEN &

¹Sensores externos, que podem ser acoplados a robôs e utilizados em automação residencial.

DRUMMOND (2006)) como detectores de bolhas (LOWE (2003), BAY; TUYTELAARS; GOOL (2006), e AGRAWAL; KONOLIGE; BLAS (2008)). Um resumo sobre detectores pode ser encontrado em (SIEGWART; NOURBAKHS; SCARAMUZZA, 2011).

Fraundorfer & Scaramuzza (2012) apresentam uma comparação das propriedades e desempenho de diferentes detectores de cantos e bolhas como mostra a Tabela 2.

Tabela 2: Comparação dos detectores e descritores de pontos.

	Detector de Cantos	Detector de Bolhas	Rotação Invariante	Escala Invariante	Afins Invariante	Repetitividade	Acurácia de Localização	Robustez	Eficiência
Harris	X		X			+++	+++	++	++
Shi-Tomasi	X		X			+++	+++	++	++
FAST	X		X	X		++	++	++	++++
SIFT		X	X	X	X	+++	++	+++	+
SURF		X	X	X	X	+++	++	++	++
CENSURE		X	X	X	X	+++	++	+++	+++

Fonte: Adaptado de Fraundorfer & Scaramuzza (2012).

Nesse trabalho foi utilizado o SURF como detector de características por ser invariante à escala e à rotações (isso significa que qualquer tipo de movimento pode ser detectado), e por oferecer um bom desempenho computacional. Além disso, este algoritmo está disponível em forma de uma função na biblioteca OpenCV, o que reduziu o tempo de desenvolvimento desse trabalho.

4.5 Algoritmo de FLANN

O FLANN (*Fast Approximate Nearest Neighbor Search*), desenvolvido por Muja & Lowe (2003) é uma biblioteca que contém uma coleção de algoritmos otimizados para encontrar, com rapidez, o vizinho mais próximo de um ponto em um grande conjunto de dados e características de grandes dimensões. A OpenCV oferece a interface *FlannBasedMatcher* para realizar uma correspondência de pontos de forma eficiente e rápida utilizando o FLANN. Assim, após a detecção dos pontos de características da primeira e segunda imagem, com o uso do detector de pontos SURF, é feita uma etapa de comparação e posterior correspondência (se encontrada a correspondência entre pontos), através do

algoritmo FLANN. Os pontos reconhecidos em ambas as imagens, isto é, com correspondência assegurada, são utilizados para o cálculo do movimento do robô.

5 EXPERIMENTOS E ANÁLISES

Para verificar se o sistema de Odometria Visual implementado apresentava resultados coerentes, foram realizados alguns testes da estimação do movimento do *Kinect* (no robô) em instantes diferentes de tempo. Após coletar a primeira imagem, o tempo mínimo em média para coletar a próxima imagem (da sequência) é de cerca de 250 ms.

5.1 Contextualização dos Experimentos

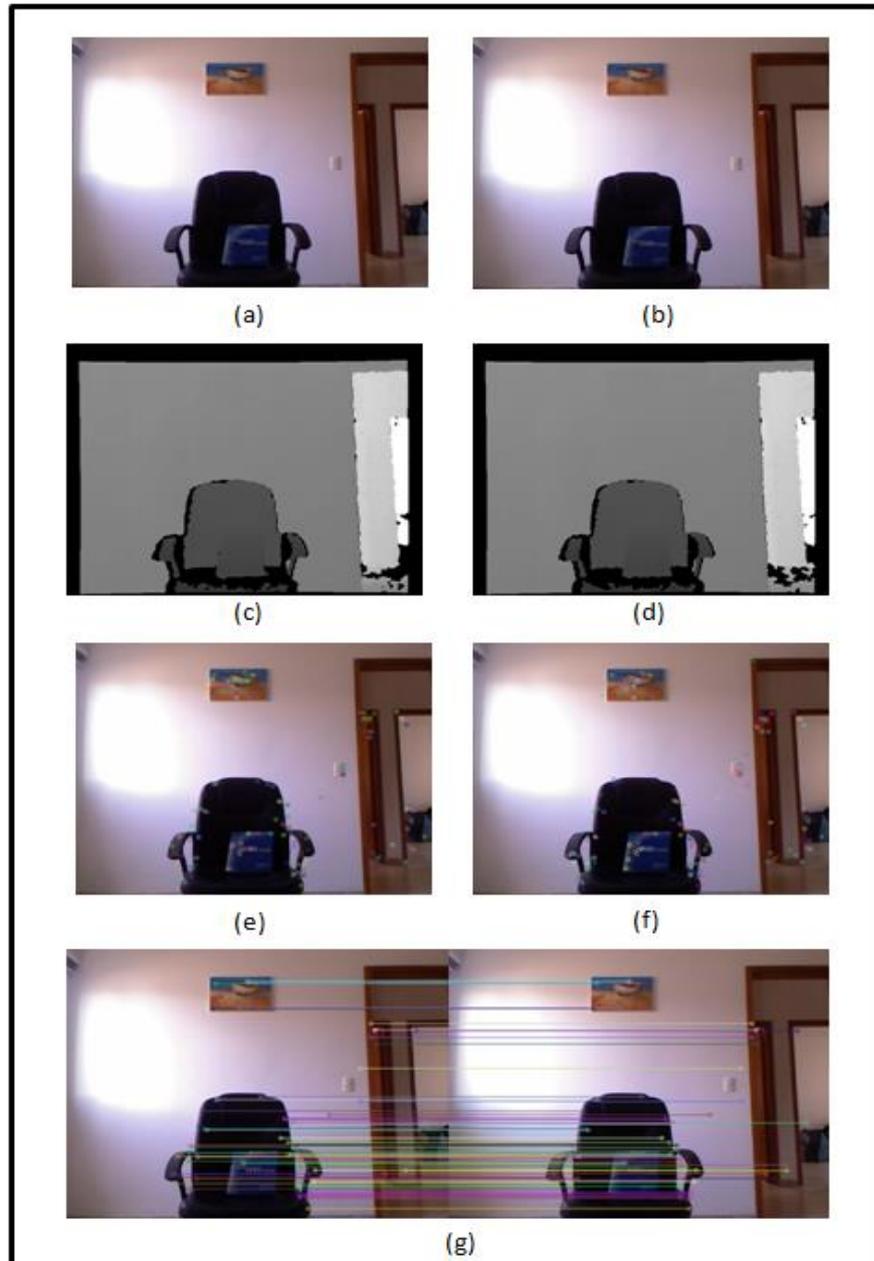
Os experimentos do sistema de Odometria Visual foram realizados em um ambiente *indoor* bem iluminado e com poucos objetos.

Foi utilizada uma trena métrica para calcular o deslocamento real do *Kinect* no espaço e assim, comparar os valores reais com os valores obtidos pelo sistema de Odometria Visual. Todos os experimentos são ilustrados nas figuras abaixo da seguinte forma: a imagem (a) foi capturada no instante $k-1$ e a imagem (b) foi capturada no instante k . As imagens (c) e (d) são os mapas de profundidade de (a) e (b) respectivamente, enquanto que, as imagens (e) e (f) mostram os pontos identificados pelo algoritmo SURF. A imagem (g) mostra a correlação feita pelo algoritmo de FLANN entre os pontos notáveis comuns às imagens (a) e (b).

5.2 Primeiro Experimento: sem movimentação

O primeiro experimento consiste em testar se os valores das coordenadas do sistema de Odometria Visual sofrem alterações sem haver qualquer deslocamento do *Kinect*, como mostra a Figura 5.1. Constatou-se portanto que:

Figura 5.1: Experimento sem movimento do *Kinect*.



- a pose esperada como resultado para esse experimento, em unidades métricas (m), era:

$$C_k = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

- a pose calculada pelo sistema de Odometria Visual, em unidades métricas (m), foi:

$$C_k = \begin{bmatrix} 0,0000238 \\ 0,00019 \\ 0 \end{bmatrix}$$

Isso nos mostra que praticamente não houve movimentação, já que os valores das coordenadas x , y e z podem ser truncados ou arredondados para zero. É possível notar, neste caso, um maior número de correspondências entre os pontos notáveis, já que não houve qualquer deslocamento do *Kinect*. Ou seja, grande parte das características detectadas na primeira imagem (ver Figura 5.1(e)), foi detectada na segunda imagem (ver Figura 5.1(f)).

5.2 Segundo Experimento: deslocamento sobre o eixo x do *Kinect*

O segundo experimento consiste no cálculo do deslocamento translacional do *Kinect* de 0,05m sobre o eixo x , considerando o seu sistema de coordenadas. A Figura 5.2 mostra as imagens capturadas neste experimento. Sendo assim, foi constatado que:

- a pose esperada como resultado para esse experimento, em unidades métricas (m), era:

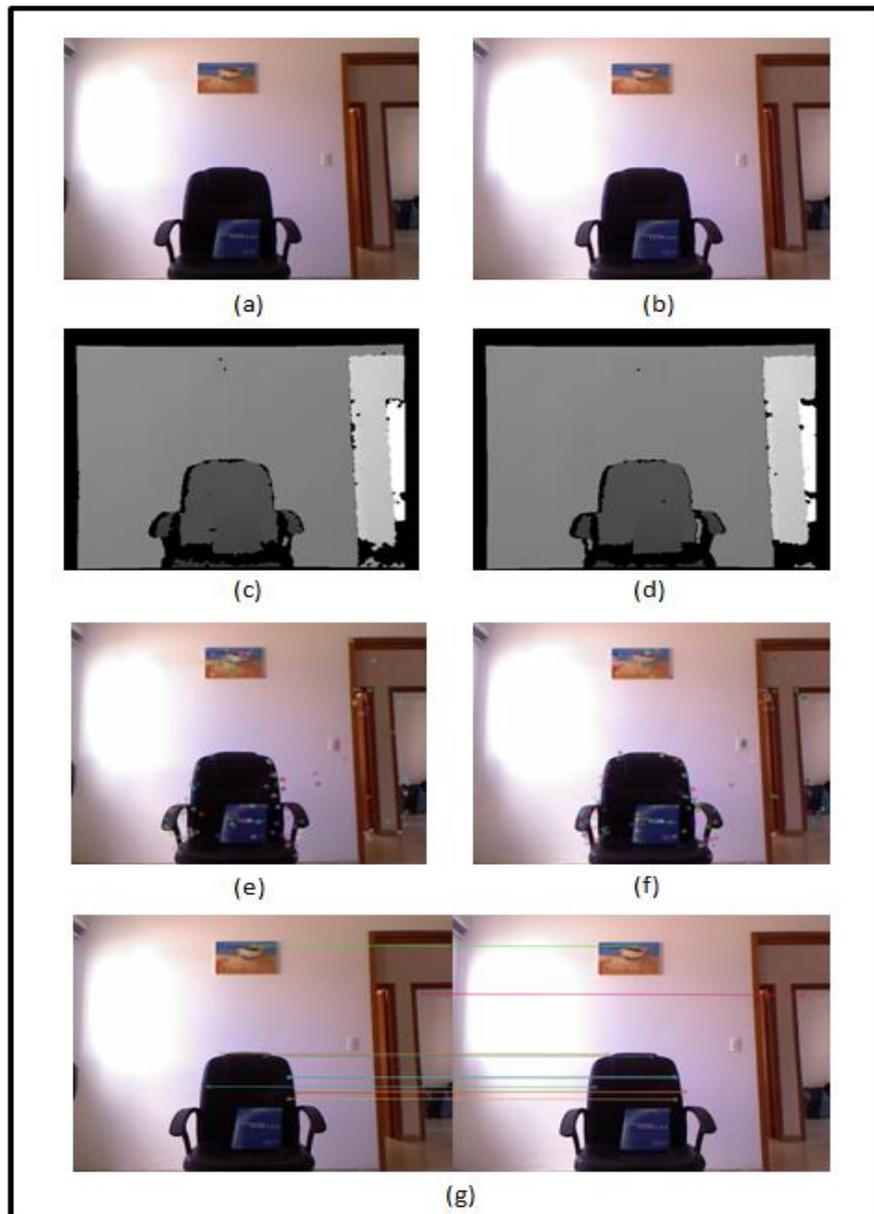
$$C_k = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 0,05 \\ 0 \\ 0 \end{bmatrix}$$

- a pose calculada pelo sistema de Odometria Visual, em unidades métricas (m), foi:

$$C_k = \begin{bmatrix} 0,0454 \\ 0,0015 \\ 0 \end{bmatrix}$$

Percebe-se que o sistema apresentou um deslocamento em x próximo ao valor esperado e uma milimétrica variação de deslocamento no eixo y . Esse resultado é considerado coerente. Observou-se também que, com o movimento, menos característica da primeira imagem tiveram suas correspondentes na segunda imagem verificadas, se comparado ao primeiro experimento.

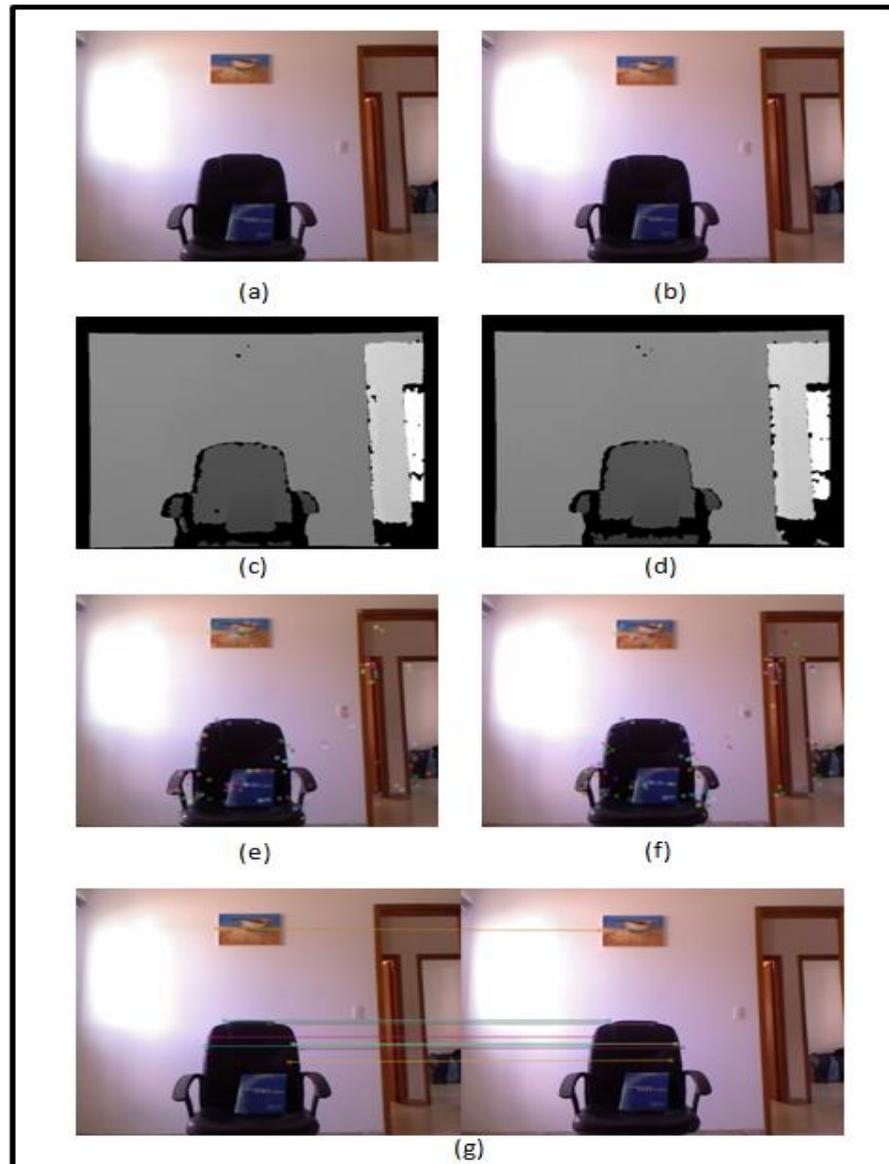
Figura 5.2: Experimento com movimento do *Kinect* de 0,05m sobre o eixo x .



5.3 Terceiro Experimento: deslocamento sobre o eixo z do *Kinect*

O terceiro experimento consistiu na translação do *Kinect* de 0,045m sobre o eixo z , considerando o seu sistema de coordenadas. A Figura 5.3 mostra as imagens capturadas neste experimento. Dessa forma:

Figura 5.3: Experimento com movimento do *Kinect* de 0,045m em z .



- a pose esperada como resultado para esse experimento, em unidades métricas (m), era:

$$C_k = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0,045 \end{bmatrix}$$

- a pose calculada pelo sistema de Odometria Visual, em unidades métricas (m), foi:

$$C_k = \begin{bmatrix} -0,0052 \\ -0,014 \\ 0,039 \end{bmatrix}$$

O valor resultante da coordenada z foi igual a 0,039m, uma diferença de 0,006m da distância real do deslocamento. Além disso, também houve uma variação de 0,014m no eixo y , e uma variação de 0,0052m no eixo x , o que afasta um pouco mais o resultado esperado do resultado estimado.

5.4 Quarto Experimento: deslocamento sobre o eixo y do *Kinect*

O quarto experimento consistiu na translação do *Kinect* de -0,2m sobre o eixo y , considerando o seu sistema de coordenadas. A Figura 5.4 mostra as imagens capturadas neste experimento, que apresentou:

- a pose esperada como resultado para esse experimento, em unidades métricas (m), era:

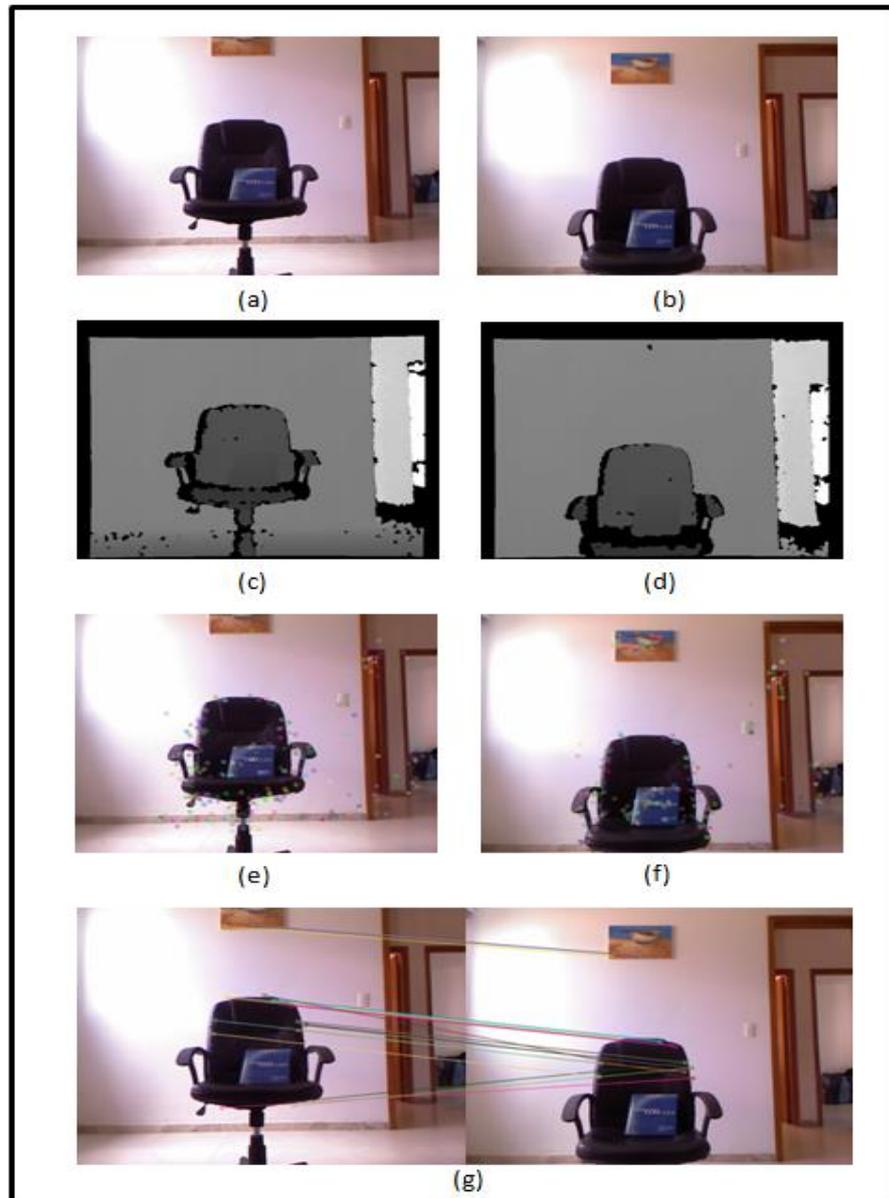
$$C_k = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 0 \\ -0,2 \\ 0 \end{bmatrix}$$

- a pose calculada pelo sistema de Odometria Visual, em unidades métricas (m), foi:

$$C_k = \begin{bmatrix} -0,0034 \\ -0,212 \\ 0,0077 \end{bmatrix}$$

O valor resultante da coordenada y foi igual a -0,212m, uma diferença de 0,012m da distância real do deslocamento. Além disso, também houveram milimétricas variações no deslocamento relativo aos eixos x e z . Percebe-se visualmente na Figuras 5.4, que algumas correspondências falharam (pontos detectados na cadeira com correspondências falsas), porém isso não afetou, de forma considerável, na estimativa de um deslocamento coerente, pois a média geral das correspondências consideradas adequadas minimizou esse efeito.

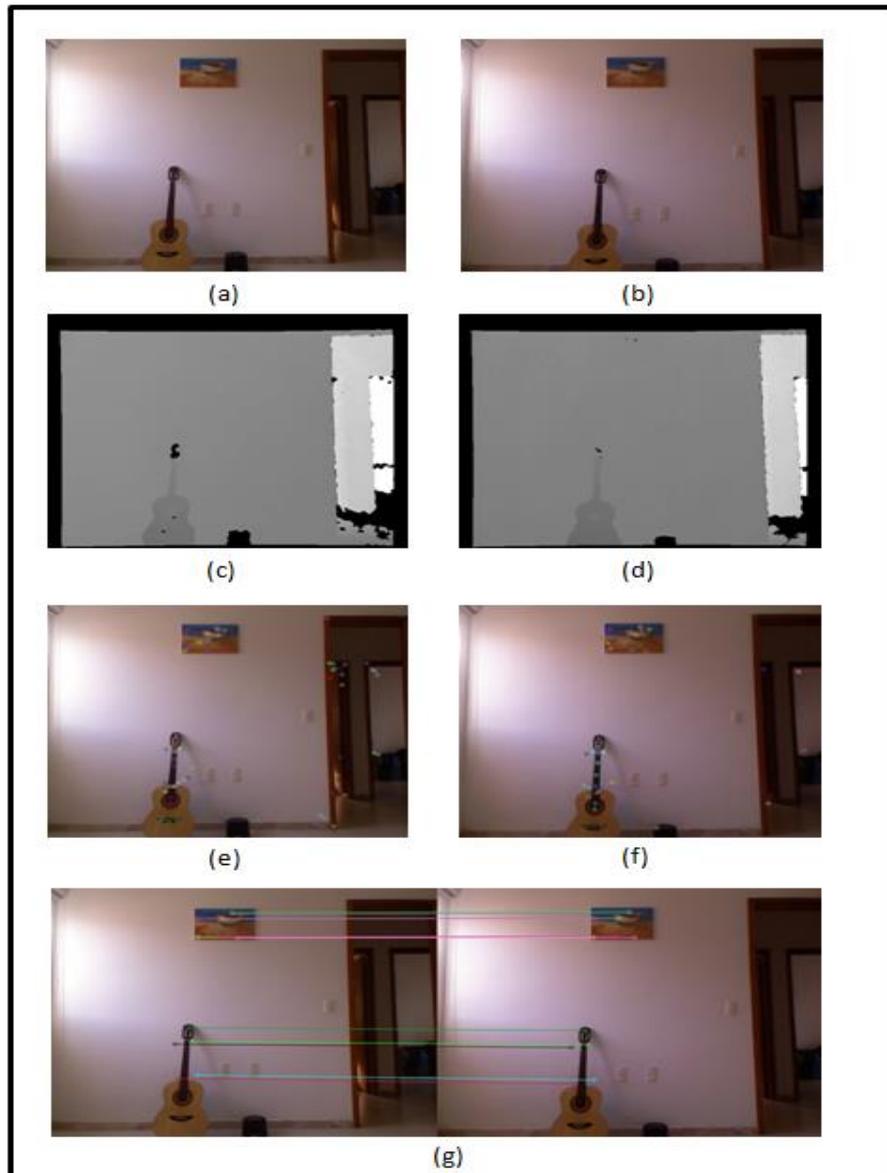
Figura 5.4: Experimento com movimento do *Kinect* de $-0,2\text{m}$ em y .



5.5 Quinto Experimento: deslocamento sobre os eixos x e z do *Kinect*

O quinto experimento consiste no cálculo do deslocamento em x e z combinados (em forma de diagonal), considerando o sistema de coordenadas do *Kinect*. O valor real do deslocamento para a coordenada x foi de $0,1\text{m}$ e para a coordenada z foi também de $-0,1\text{m}$. Já a Figura 5.5 mostra as imagens capturadas neste experimento.

Figura 5.5: Experimento com movimento de 0,1m em x e -0,1m em z .



- a pose esperada como resultado para esse experimento, em unidades métricas (m), era:

$$C_k = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 0,1 \\ 0 \\ -0,1 \end{bmatrix}$$

- a pose calculada pelo sistema de Odometria Visual, em unidades métricas (m), foi:

$$C_k = \begin{bmatrix} 0,0998 \\ -0,0047 \\ -0,0880 \end{bmatrix}$$

O valor resultante da coordenada x apresentou uma variação de aproximadamente 0,0002m em relação ao valor real, enquanto que na coordenada z foi verificada uma diferença de aproximadamente 0,012m do deslocamento real.

5.6. Análise dos Experimentos

A Tabela 3 mostra dados dos cinco experimentos realizados. São mostradas as coordenadas do movimento real, as coordenadas calculadas pelo sistema de Odometria Visual e o erro absoluto (obtido através do cálculo da distância euclidiana) considerando as três coordenadas para cada experimento.

Tabela 3: Tabela com erros de cálculos das distâncias obtidas pelos cinco experimentos.

Exp.	Coordenadas Reais (m)			Coordenadas Calculadas (m)			Erro absoluto (m)
	X	Y	Z	X	Y	Z	
1	0	0	0	0,0000238	0,00019	0,0	0,000191
2	0,05	0	0	0,0454	0,0015	0,0	0,004838
3	0	0	0,045	-0,0052	-0,014	0,039	0,016095
4	0	-0,2	0	-0,0034	-0,212	0,0077	0,008502
5	0,1	0	-0,1	0,0998	-0,0047	-0,088	0,012889

Apesar de haver diferenças em relação aos movimentos reais executados, os experimentos realizados com o sistema de Odometria Visual apresentaram resultados coerentes. Eles mostraram que esse sistema pode ser utilizado em um sistema robótico ou em agentes móveis para medir movimentos executados com relativa precisão.

É importante enfatizar que, em geral, os sistemas de Odometria Visual possuem uma etapa de otimização, a fim de melhorar os resultados conseguidos. Neste trabalho, a etapa de otimização não foi implementada. Assim, pode-se acreditar em resultados mais promissores com a implementação de tal refinamento. Além disso, ressalta-se que os deslocamentos foram realizados de forma manual e medidos com uma trena métrica (instrumento com pouca precisão), o que pode interferir na medição dos dados reais dos experimentos.

6 CONCLUSÃO E PERSPECTIVAS DE TRABALHOS FUTUROS

O sistema de Odometria Visual 3D utilizando um sensor RGB-D *Kinect* desenvolvido nesse trabalho mostra-se capaz de realizar cálculos de movimentação, obtendo resultados coerentes mesmo com a presença erros nesses cálculos.

É notável também que o método de Odometria Visual mostra-se como uma melhor alternativa para se estimar o deslocamento do robô que a Odometria Mecânica, justamente por não ser suscetível aos Erros Sistemáticos e Não-sistemáticos.

O *Kinect* responde bem a necessidade de estimar o deslocamento do robô, sem sofrer com pouca iluminação do ambiente, como acontece com câmeras. Essa é uma alternativa relativamente barata comparada ao preço de *lasers* (sensores que também são usados para calcular a movimentação de um agente) e mais eficiente que *encoders* (sensores ópticos utilizados em abordagens de Odometria Mecânica).

Esse sistema posteriormente poderá ser utilizado como um subsistema que auxilie na navegação de robôs móveis do Laboratório de Aprendizagem Robótica (LAR) da Universidade do Estado do Rio Grande do Norte (UERN), como: o robô Pioneer 3-AT, o X80, os *Cellbots* (robôs baseados em *smartphones*), entre outros. Esse sistema pode, ainda, ser facilmente utilizado para implementar sistemas de navegação para automóveis autônomos ou *drones*, com algumas adaptações. Também pode-se pensar em sistemas de navegação para deficientes visuais usando essa técnica com câmeras de *smartphones*, para avisá-los através de um aplicativo que há obstáculos no ambiente.

Durante a implementação desse sistema de Odometria Visual 3D com o *Kinect* foi observado que o uso de algoritmos de otimização é necessário para se conseguir estimar a pose do robô com maior precisão. Assim, em trabalhos futuros, pode-se dar continuidade a este sistema, tratar os erros inerentes ao cálculo do deslocamento e, conseqüentemente, aumentar a acurácia do sistema de odometria visual. Outra alternativa seria uma análise numérica dos resultados dos experimentos (incluindo mais dois: x-y e y-z), ou criar uma função interpoladora que descreva a “função erro” de deslocamento nos eixos, com a finalidade de tentar constatar se é possível desenvolver um sistema livre de erros .

Com esse trabalho, espera-se contribuir com o a ciência da Visão Computacional e que mais pesquisadores prossigam, tendo êxito com novos trabalhos relacionados.

REFERÊNCIAS

- AGRAWAL, M.; KONOLIGE, K.; BLAS, M. **Censure: Center surround extremas for realtime feature detection and matching**. European Conf. Computer Vision, pp. 102–115, 2008.
- BAY, H.; TUYTELAARS, T.; GOOL. L. V. **Surf: Speeded up robust features**, in *Proc ECCV*, pp. 404–417, 2006.
- BESL, Paul J.; MCKAY Neil D. **A Method for Registration of 3-D Shapes**, IEEE Transactions on pattern analysis and machine intelligence vol 14, nº 2, fev. 1992.
- BORENSTEIN, Johan; FENG, Liqiang. **Correction of Systematic Odometry Erros in Mobile Robots**. Pittsburgh, Pennsylvania: International Conference On Intelligent Robots And Systems (iros '95), pp. 569-574, 5-9 ago. 1995.
- BORENSTEIN, Johan; FENG, Liqiang. **Where am I? Sensors and Methods for Mobile Robot Positioning**, 1ª Ed, University of Michigan, Michigan, EUA, 1996.
- BRADSKI, Gary; KAEHLER, Adrian. **Learning OpenCV**, 1ª Ed, Gravenstein Highway North, Sebastopol, CA 95472: O'Reilly, 2008.
- CHENG, Y.; MAIMONE, M. W.; e MATTHIES, L. **Visual odometry on the mars exploration rovers**, *IEEE Robotm Automat. Mag.*, vol. 13, nº 2, pp. 54-62, 2006.
- CORKE, P. I.; STRELOW, D.; e SINGH, S. **Omnidirectional visual odometry for a planetary rover**, in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, pp. 4007-4012, 2005.
- COX, Ingemar J. **Blanche - an experiment in guidance and navigation of an autonomous robot vehicle**. IEEE Transactions on Robotics and Automation 7(2), 1991.
- FISCHLER, M. A.; BOLLES, R. C. **Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartograph**. *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.
- FRAHM, J.-M; GEORGEL, P.; GALLUP, D.; JOHNSON, T.; RAGURAM, R.; WU, C.; JEN, Y.-H.; DUNN, E., CLIPPS, B.; LAZEBNIK Lazebnik; e POLLEFEYS, M. **Building rome on a cloudless day**, in *Proc. European Conf. Computer Vision*, pp. 368-381, 2010.
- FRAUNDORFER, Friedrich; SCARAMUZZA, Davide. **Visual odometry – Part II: Matching, Robustness, Optimization, and Applications**. IEEE Robotics & Automotion Magazine, Junho, 2012.
- FORSTNER, W. **A feature based correspondence algorithm for image matching**, *Int. Arch. Photogrammetry*, vol. 26, nº 3, pp. 150-166, 1986.
- HANNAH, M. **Computer matching of areas in stereo images**. Ph.D. dissertation, Stanford Univ., Stanford, CA, 1974.

- HARRIS, C.; PIKE, J. **3d positional integration from image sequences**, in *Proc. Alvey Vision Conf.*, pp. 87-90, 1987.
- HARRIS, C.; STEPHENS, M. **A combined corner and edge detector**, in *Proc. Alvey Vision Conf.*, pp. 147-151, 1988.
- HOLZ, Dirk; HOLZER, Stefan; RUSU, R. Bogdan; BEHNKE, Sven. **Robocup 2011: Robot Soccer World Cup XV Real-Time Plane Segmentation Using RGB-D Cameras**. Lecture Notes in Computer Science Volume 7416, pp 306-317, 2012.
- HOWARD, A. **Real-time stereo visual odometry for autonomous ground vehicles**, in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, pp. 3946-3952, 2008.
- HUANG, Albert S; BACHRACH, Abraham; HENRY, Peter; KRAININ, Michael; MATURANA, Daniel; FOX, Dieter; e ROY, Nicholas. **Visual Odometry and Mapping for Autonomous Flight Using an RGB-D Camera**. Int. Symposium on Robotics Research (ISRR), Flagstaff, Arizona, USA, ago. 2011.
- KERL, Christian; STURM, Jürgen; CREMERS, Daniel. **Odometry from RGB-D Cameras for Autonomous Quadcopters**. Master's Thesis, 12 nov. 2012.
- KIRNER, Claudio; TORI, Romero. **Realidade Virtual: Conceitos e Tendências**. 1.ed. São Paulo: Mania de Livro, 2004.
- LACROIX, S.; MALLET, A.; CHALITA R.; e GALLOR L. **Rover self localization in planetary-like environments**, in *Proc. Int. Symp. Artificial Intelligence, Robotics, and Automation for Space (i-SAIRAS)*, pp. 433-440, 1999.
- LHULLIER, M. **Automatic structure and motion using a catadioptric camera**, in *Proc. IEEE Workshop Omnidirectional Vision*, pp. 1-8, 2005.
- LONGUET-HIGGINS, H. **A computer algorithm for reconstructing a scene from tow projections**, “*Nature*”, vol. 293, nº 10, pp. 133-135, 1981.
- LOWE, D. **Distinctive image features from scale-invariant key-points**, *Int. J. Comput. Vis.*, vol. 20, no. 2, pp. 91–110, 2003.
- MAIMONE, M.; CHENG, Y.; e MATTHIES, L. **Two years of visual odometry on the mars exploration rovers: Fields report**. *J. Field Robot.*, vol. 24, nº 3, pp. 169-186, 2007.
- MATTHIES, L. **Dinamic stereo vision**. Ph.D. dissertation, Carnegie Melon Univ, Pittsburgh, PA, 1989.
- MATTHIES, L.; SHAFER S. **Error modeling in stereo navigation**, “*IEEE J. Robot. Automat.*”, vol. 3, nº3, pp. 239-248, 1987.
- MILELLA, A. e SIEGWART R. **Stereo based ego-motion estimation using pixel tracking and iterative closet point**, in *Proc. IEEE Int. Conf. Vision Systems*, pp. 21-24, 2006.
- MILFORD, M. J.; WYETH, G. **Single camera vision-only SLAM on a suburban road network**, in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA ' 08)*, pp. 3684-3689, 2008.

MORAVEC, H. **Obstacle avoidance and navigation in the real world by a seeing robot robô planetário**. PhD. dissertation, Standford Univ., Standford, CA, 1980.

MORAVEC, H. **Towards automatic visual obstacle avoidance**, in *Proc. 5th Int. Join Conf. Artificial Intelligence*, p. 584, ago. 1977.

MUJA, Marius; LOWE, David G. **Fast Approximate Nearest Neighbors With Automatic Algorithm Configuration**, 2009.

NEGENBORN, Rudy. **Robot localization and kalman filters: On finding your positions in a noisy world**. Dissertação de mestrado, Utrecht University, 2003.

NISTER, David, NARODITSKY, O.; BERGEN, J. **Visual odometry**, in *Proc. Int. Conf. Computer Vision and Pattern Recognition*, pp. 652-659, 2004.

NGHIA HO. **Finding optimal rotation and translation between corresponding 3D poitns**. Disponível em: <nghiaho.com/?page_id=671>. Acesso em: 04 ago. 2014.

OSLON, C.; MATTHIES L.; SCHOPPERS M.; e MAIMONE M. W. **Robust stereo ego-motion for long distance navigation**, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 453-458, 2000.

OPENCV. **Operation on Arrays**. Disponível em: <docs.opencv.org/modules/core/doc/operations_on_arrays.html?highlight=svd#SVD>. Acesso em: 04 ago. 2014.

OPENNI. **OPENNI**. Disponível em: <www.openni.ru>. Acesso em: 21 mar. 2014.

REINALDO, J. Oliveira. **Reconstrução de Superfícies em Tempo Real Usando Visão Estéreo**, 2012.

ROSTEN, E.; DRUMMOND, T. **Machine learning for high-speed cor-ner detection**, in *Proc. European Conf. Computer Vision*, vol. 1, pp. 430–443, 2006.

RUSSEL, Stuart; NORVIG, Peter. **Artificial Intelligence – A Modern Approach**. Second Edition, Editor Pearson Education, New Jersey, USA, 2003.

SANTANA, A. Macêdo. **Localização e planejamento de caminhos para um robô humanoide e um robô escravo com rodas**. Dissertação de Mestrado, Universidade Federal do Rio Grande do Norte, UFRN, Natal-RN, 2007.

SCARAMUZZA, Davide; FRAUNDORFER, Friedrich. **Visual odometry – Part I: 30 Years and Fundamentals**. IEEE Robotics & Automotion Magazine, dezembro 2011.

SCARAMUZZA, D.; FRAUNDORFER, F.; SIEGWART, R. **Real-time monocular visual odometry for on-road vehicles with 1-point RANSAC**, in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, pp. 4293-4299, 2009.

SCHMIDT, A.; KRAFT. M.; KASINSKI, A. **An evaluation of image feature detectors and descriptors for robot navigation**, in *Proc. Int. Conf. Computer Vision and Graphics*, pp. 251-259, 2010.

SIEGWART, Rolland; NOURBAKHS, Illah R. **Introduction to Autonomous Mobile Robot**, 1a edição, MIT Press, Cambridge, Massachusetts, EUA, 2004.

SIEGWART, Rolland; NOURBAKHS, Illah R.; SCARAMUZZA, David. **Introduction to Autonomous Mobile Robots**, 2nd ed. Cambridge, MA, MIT Press, 2011.

SCHÖPS, Thomas; ENGEL, Jakob; CREMERS, Daniel. **Semi-Dense Visual Odometry for AR on a Smartphone**, 2014.

SOUZA, Anderson A. de S; GONÇALVES, Luiz M. Garcia. **Mapeamento com Sonar Usando Grade de Ocupação Baseado em Modelagem Probabilística**. Dissertação de Mestrado, Rio Grande do Norte, 2008.

TARDIF, J.; PAVLIDIS, Y.; e DANILIDIS, K. **Monocular visual odometry in urban environments using an omnidirectional camera**, in *Proc IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, pp. 2531-2538, 2008.

TOMASI, C.; e SHI, J. **Good features to track**, in *Proc. CVPR*, 1994, pp. 593-600.

THRUN, Sebastian; BURGARD Wolfram; FOX Dieter. **Probabilistic Robotics**. Intelligent Robotics and Autonomous Agents Series. Pp 672, Editora The MIT Press, 2005.