

VISUALIZAÇÃO TRIDIMENSIONAL BASEADA EM PONTOS APLICADA À ROBÓTICA MÓVEL

Nils Alexandre Lima Bergsten¹; Anfranserai Morais Dias²

1. Bolsista PROBIC, Graduando em Engenharia de Computação, Universidade Estadual de Feira de Santana, e-mail: nilsalexberg@gmail.com

2. Orientador, DTEC, Universidade Estadual de Feira de Santana, e-mail: anfranserai@ecomp.uefs.br

PALAVRAS-CHAVE: Visão computacional, robótica móvel, visualização 3D baseada em pontos

INTRODUÇÃO

Um dos problemas fundamentais da computação gráfica é a representação e visualização de modelos tridimensionais[1]. Um dos esquemas de representação para modelos geométricos 3D mais utilizado é a representação por malhas poligonais [2]. Em particular, malhas de triângulos são muito usadas visto que as Unidades de Processamento Gráfico (GPU) oferecem suporte otimizado para esse tipo de representação [3].

Com o avanço dos equipamentos e métodos de aquisição de dados tridimensionais [4], os algoritmos de visualização 3D mais difundidos, baseados em malhas poligonais, se tornaram inadequados. Isto se deve ao fato de que os scanners 3D coletam enormes quantidades de pontos, porém sem uma conectividade associada, ou seja, sem definir de forma explícita que pontos formam polígonos. Esse conjunto de pontos, também chamado de nuvem de pontos, pode ser convertido para uma malha poligonal, porém o processo é complicado e trabalhoso, exigindo a execução de algoritmos complexos, que dependem de análises baseadas em algoritmos geométricos, sujeitos a erros numéricos.

Para resolver esse problema foram desenvolvidos diversos métodos de visualização 3D baseados em pontos [5]. Por ser a mais simples primitiva geométrica, o ponto oferece vantagens sobre os outros métodos de representação. Malhas poligonais armazenam não só informações referentes aos pontos (coordenadas, cor, textura, etc.) como as relações topológicas que representam as conexões entre os eles, de modo a representar arestas e faces. Portanto, do ponto de vista de armazenamento, modelos baseados em pontos são menos custosos computacionalmente.

Dispositivos de escaneamento tridimensional vêm sendo utilizados em diversas áreas, inclusive em robótica. A partir do escaneamento do ambiente onde o robô está imerso, é possível reconstruir e reconhecer esse ambiente, permitindo ao robô se locomover de forma autônoma [6]. O desafio nesse caso é integração entre os processos de reconstrução do ambiente e locomoção do robô, visto que são processos que acontecem em paralelo e são, sob certo ponto de vista, complementares.

Este trabalho integra-se ao projeto “Construção de uma Colônia de Robôs Autônomos para reconhecimento, busca e inspeção”. Dentro do contexto do projeto, a proposta de trabalho é desenvolver uma ferramenta que permita a reconstrução e visualização do ambiente 3D, a partir dos dados coletados pelos sensores dos robôs (sensores laser, sonares e duas câmeras). A partir da geração do modelo do ambiente o robô deverá ser capaz de se locomover e se auto-localizar no ambiente. Desta forma os robôs poderão inspecionar ambientes de difícil acesso ao ser humano.

METODOLOGIA

Na primeira etapa do projeto, foi realizado um estudo sobre as técnicas e algoritmos de visualização tridimensional baseada em pontos aplicáveis ao contexto do projeto. Tal estudo permitiu obter a fundamentação teórica necessária para desenvolver da fase inicial do projeto.

Na segunda etapa do projeto, o desenvolvimento da visualização de modelos. Foi feita a utilização das tecnologias da linguagem de programação C++, OpenGL e a biblioteca disponível para manipulação de arquivos de extensão PLY, padrão para modelos tridimensionais baseados em pontos, criada por pesquisadores da Universidade de Stanford [7]. Essas tecnologias foram utilizadas para desenvolver uma ferramenta capaz de realizar a visualização de imagens tridimensionais baseadas em pontos no formato PLY.

Na terceira etapa do projeto, foi realizado o projeto e a codificação do componente da solução relativo à aquisição da nuvem de pontos a partir de um par estéreo de imagens. Para esta etapa, foi feita a utilização da linguagem C++ e a biblioteca OpenCV, para manipulação de imagens. O uso destas tecnologias possibilitou o desenvolvimento de uma ferramenta capaz de, a partir de duas fotos tiradas de pontos de vista ligeiramente diferentes para originar a paralaxe, retirar informações de profundidade para gerar um modelo tridimensional baseado em pontos.

Para esta etapa, foi aplicado o algoritmo de pareamento de imagens estéreo por intensidade luminosa [8]. O programa deveria abrir as imagens como matrizes, sendo que cada pixel seria representado por um elemento da matriz. A seguir o programa varre a primeira imagem, e para cada ponto da mesma, é feita uma janela (uma matriz 3x3), para buscar na mesma linha da segunda imagem a janela que mais se assemelha com a janela da primeira imagem. Depois de realizada a busca, é calculada a distância dos pontos correspondentes para retirar a informação de profundidade.

Devido à complexidade do projeto de pesquisa e ao tempo de trabalho, não foi possível integrar as duas ferramentas ao robô e fazer os testes associados.

RESULTADOS

Para visualização de modelos tridimensionais baseados em pontos foi desenvolvido um programa de computador capaz de receber como entrada um arquivo no formato PLY contendo dados de coordenadas dos pontos do modelo e gerar como saída a visualização 3D do modelo.

Para testar o programa desenvolvido, foi utilizado o repositório de reconstrução tridimensional da Universidade de Stanford. O modelo escolhido foi o bunny, uma escultura de um coelho escaneada pelos pesquisadores da universidade. A imagem a seguir ilustra a saída do programa, a nuvem de pontos do modelo.

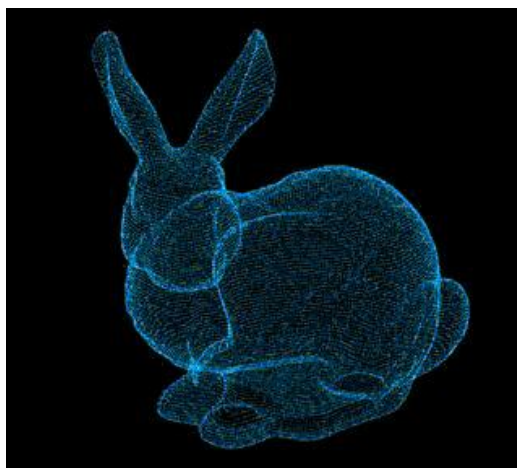


Figura 1: Ilustração do modelo bunny visualizado por pontos

Como já foi anteriormente citado, para a implementação da geração de nuvem de pontos foi utilizado o algoritmo por intensidade luminosa. Neste algoritmo, para cada pixel da primeira imagem, serão acessados todos os pixels da segunda imagem. Considerando duas

imagens de n pixels de comprimento e m pixels de altura, o número de iterações será n^2m . Assim, se pode concluir que o algoritmo de pareamento estereográfico por intensidade luminosa tem complexidade de ordem n^3 .

Foram testados alguns pares de imagens como entrada da solução. As imagens a seguir ilustram as entradas e a saída para cada teste realizado, sendo que a cor preta representa a ausência de profundidade no ponto ($z = 0$), a cor verde representa a profundidade negativa ($z \leq 0$) e a cor azul representa positiva ($z \geq 0$).

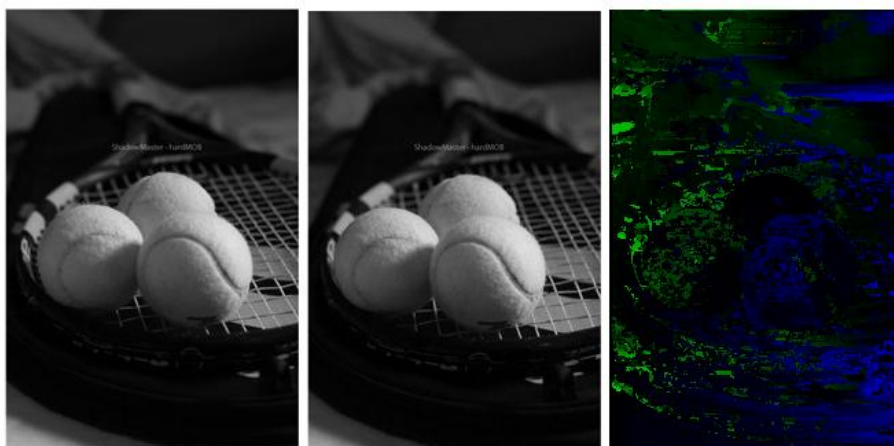


Figura 2: Resultado do pareamento por intensidade luminosa

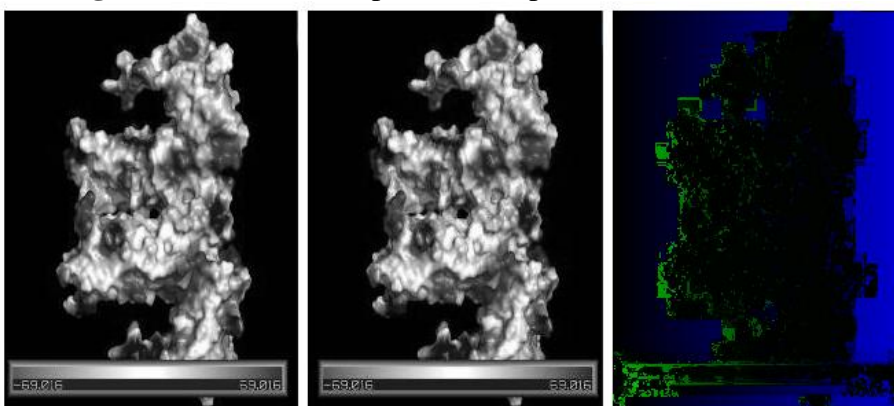


Figura 3: Mais um resultado do pareamento por intensidade luminosa

A primeira solução se mostrou bastante eficiente para a solução da situação-problema, visto que é capaz de exibir modelos tridimensionais baseados em pontos a partir de arquivos de extensão largamente utilizada na área.

Já na segunda solução, como pode ser observado nas imagens acima exibidas, o ruído da saída foi consideravelmente alto. Além disso, como o algoritmo é de ordem n^3 , o tempo de execução é alto comparado à necessidade da robótica móvel.

A utilização das funções que o OpenCV oferece específicas para o pareamento de pares de imagens estereo pode otimizar consideravelmente a parte da solução relativa a geração da nuvem de pontos.

CONCLUSÃO

Apesar de não alcançar os todos os objetivos inicialmente propostos, o trabalho tem sua importância ao embasar trabalhos futuros nesta área. O levantamento bibliográfico realizado, os códigos desenvolvidos e a produção textual servirão como uma excelente base para uma possível continuidade na pesquisa.

O trabalho expôs a importância da visualização tridimensional baseada em pontos para a robótica móvel, descreveu os principais algoritmos de reconstrução tridimensional e exibiu os resultados da implementação de um desses algoritmos.

REFERÊNCIAS

- [1] FOLEY, J. et al. 1997. Computer Graphics: Principles and Practice. Addison-Wesley: Reading.
- [2] AZEVEDO, Eduardo; CONCI, Aura. 2003. Computação Gráfica: Teoria e Prática. Campus.
- [3] AKENINE-MOLLER, T., MOLLER, T., and HAINES, E. 2008 Real-Time Rendering. 3a. edição. A. K. Peters, Ltd.
- [4] RUSINKIEWICZ, S., HALL-HOLT, O., and LEVOY, M. 2002. Real-time 3D model acquisition. ACM Trans. Graph. 21, 3, p.438-446.
- [5] GROSS, M. and PFISTER, H. 2007 Point-Based Graphics (The Morgan Kaufmann Series in Computer Graphics). Morgan Kaufmann Publishers Inc.
- [6] WANG, Jianhua; LI, Bing; CHEN, Weihai; RONG, Lixia. 3D Reconstruction Embedded System Based on Laser Scanner for Mobile Robot. School of Automation and Electrical Engineering - Beijing University of Aeronautics and Astronautics.
- [7] The Digital Michelangelo Project. Disponível em: <<http://www.graphics.stanford.edu/projects/mich/>>
- [8] OTUYAMA, Júlio M. Visão Estéreo, 1998. Pós-graduação em Ciência da Computação, Universidade Federal de Santa Catarina.