

SLAM monocular com reconstrução de planos para ambientes internos

Nicolau Leal Werneck¹ and Anna Helena Reali Costa

Laboratório de Técnicas Inteligentes, USP, São Paulo, SP, Brazil,
nwerneck@usp.br, anna.reali@usp.br,
WWW home page: <http://www.lti.pcs.usp.br>

Resumo Este artigo apresenta um sistema de reconstrução 3D em desenvolvimento para modelar paredes de ambientes antrópicos a partir de uma única câmera em movimento que é rastreada em tempo real. O funcionamento em tempo real diferencia a técnica dos trabalhos existentes em reconstrução 3D estritamente visual, e o modelo utilizado também a diferencia das técnicas existentes de SLAM visual, que se baseiam em nuvens de pontos e pequenos segmentos de reta. O cenário de aplicação pretendido é o de salas onde grandes bordas podem ser visualizadas sem muita dificuldade. As bordas estruturais do ambiente são os elementos fundamentais do modelo utilizado, e ainda assume-se que as direções destas bordas são mutuamente ortogonais. Assim a técnica proposta alcança uma melhoria na detecção de bordas, e a possibilidade de desacoplar os processos de estimação da orientação e localização da câmera. As bordas também são rastreadas de uma maneira relativamente simples e peculiar, sobre linhas e colunas fixas da matriz de imagem. A organização do sistema segue a tendência de rastrear a câmera de forma independente, e apenas atualizar o mapa quando novos quadros-chave são selecionados.

Keywords: Computer vision, Bundle Adjustment, monocular vision, visual SLAM, real-time SFM, vanishing point, Hough transform

Nível: Doutorado. Dada de conclusão esperada: Agosto de 2011.

1 Introdução

O trabalho aqui apresentado contribui para o estudo de técnicas de reconstrução de sólidos em tempo real a partir de câmeras em movimento. Trata-se de um problema de localização e mapeamento simultâneos, ou SLAM (de *simultaneous localization and mapping*), que consiste na estimação da localização de marcos em um ambiente a partir de leituras de um sensor em movimento, que também precisa ser localizado [1,2]. No *SLAM visual* são utilizadas uma ou mais câmeras como sensor, permitindo o uso de técnicas de SFM (*shape from motion*) [3], mais especificamente de BA (*bundle adjustment*) [4], além de reconstrução 3D a partir de uma única imagem [5]. O presente trabalho segue uma tendência atual de pesquisa em adaptar estas técnicas de Visão Computacional para realizar SLAM

em tempo real utilizando uma única câmera em movimento (*SLAM monocular*). O objetivo principal da pesquisa apresentada aqui é realizar SLAM monocular em ambientes internos e produzir um modelo tridimensional das paredes do ambiente. Os sistemas atuais de SLAM monocular criam mapas exíguos, que apenas permitem o rastreamento da câmera. Mas modelos 3D de um ambiente seriam úteis para realizar segmentação de fundo em tempo real, navegação de robôs, ou ainda viabilizar aplicações como as de *realidade aumentada* (AR) [6]. A técnica proposta aqui visa produzir tais modelos, mas aproveitando o próprio modelo como a fonte principal de dados para estimar a localização da câmera

Ao invés de partir de marcos visuais simples para inferir modelos mais complexos *a posteriori* [32,39], nesta proposta é assumido desde o início o desejo de se obter um modelo 3D onde há vértices com longas arestas em direções ortogonais. Isto afeta a forma como se pode realizar a detecção das bordas nas imagens, e a estimação de orientação e localização espacial da câmera e dos marcos, resultando em um processo diferente dos existentes para SLAM monocular. As bordas podem ser detectadas sobre linhas e colunas fixas das matrizes de imagem, e a condição de ortogonalidade permite estimar a orientação da câmera de forma independente, além de melhorar a detecção das bordas.

Deve ser ressaltado que seria possível construir o modelo 3D utilizando uma técnica estabelecida de SLAM monocular para localizar a câmera e alimentar um segundo processo de reconstrução 3D de um ambiente. A técnica apresentada aqui se diferencia disto, porém, por efetivamente utilizar o modelo dentro do SLAM como um mapa de fato, criando um único processo fechado.

Durante esta pesquisa estão sendo testadas também técnicas novas para alguns dos componentes do sistema que podem vir a ser reutilizados em aplicações similares. Uma delas se refere a permitir o uso do algoritmo do Gradiente Conjugado para realizar BA, que é um tema sendo recentemente abordado por outros pesquisadores [8,9]. Outra técnica sendo desenvolvida é o uso do algoritmo de transformada rápida de Hough [10] para realizar um mapeamento inicial.

Este artigo prossegue com uma avaliação de trabalhos anteriores da área, uma descrição mais detalhada do problema, e apresentação de resultados preliminares.

1.1 SFM e SLAM visual

Em SLAM visual tradicional utilizam-se técnicas como o Filtro de Kalman Estendido e o algoritmo FastSLAM 2.0 [11,12]. Nestes sistemas são mantidas estimativas probabilísticas da posição e orientação da câmera e das posições dos marcos, e a cada nova imagem é feita uma atualização a partir de um modelo de movimento e de observação [11,12,13,14,15,16]. Estas técnicas são muito empregadas em robótica móvel com outros tipos de sensores. Apesar de resultados satisfatórios serem obtidos, elas apresentam um aumento elevado do custo computacional conforme o número de marcos aumenta, podendo chegar a $O(n^3)$ operações por iteração, e além disso os resultados não possuem a melhor precisão possível, mesmo com o uso de parametrizações mais complexas [17], descritores de marcos mais genéricos [18] ou modelos probabilísticos mais flexíveis [19]. O

desejo de se obter melhor precisão, e os impedimentos práticos encontrados na realização de filtragem acabam estimulando os pesquisadores a recorrer às técnicas de BA. Isto permite rastrear a câmera com mais precisão, por exemplo, e inicializar os marcos com melhores estimativas. A trajetória específica realizada pela câmera também pode passar a ter menos importância no resultado obtido uma vez que se pode gerenciar com mais liberdade quais medições utilizar.

Em um problema de BA geralmente não existe nenhuma informação a respeito do movimento da câmera, e os dados são a princípio considerados todos ao mesmo tempo, e não de forma incremental como em filtros. Por este motivo BA é geralmente realizado *off-line*, e também porque as implementações mais simples costumam ser muito custosas para operação em tempo real. Enquanto em filtros as estimativas são atualizadas apenas para absorver novas leituras, em BA realiza-se uma regressão não-linear para minimizar a soma do erro quadrático de reprojeção dos marcos em um conjunto de imagens simultaneamente. Esta otimização é muito frequentemente conduzida através do algoritmo de Levenberg-Marquardt (LM) [4,20].

Esta otimização realizada no BA pode ser interpretada como uma maximização de verossimilhança, que é uma forma muito comum para estimar os parâmetros de um modelo. O problema entretanto não se resume a isto, porque antes da otimização é necessário realizar o passo de obtenção das correspondências (*matching*) entre as detecções nas imagens e os marcos. Ou seja, é preciso determinar a que marco cada observação corresponde, ou quais conjuntos de pontos detectados em diferentes imagens correspondem a um mesmo marco. Encontrar estas correspondências pode ser bastante difícil. Alguns exemplos de técnicas destinadas a isto são o algoritmo *Expectation-Maximization* (EM) [21,22,23,24] e os chamados estimadores robustos, epitomizado pelo RANSAC [25,26]. Há também alternativas como o uso de programação linear [27], que é especialmente útil para trabalhar com métricas diferentes da quadrática.

1.2 Estado da arte

Como dito anteriormente, o uso de filtragem atingiu uma barreira. É necessário adaptar técnicas de BA ao problema para se obter resultados mais precisos, dar maior flexibilidade ao uso dos dados e permitir operar em ambientes extensos [3]. Uma forma de realizar esta adaptação é demonstrado em [28,29], onde BA é utilizado para realizar uma forma de hometria visual. Nestes sistemas a câmera é rastreada conforme se movimenta apenas a partir do mapa já obtido. Quando se atende uma determinada condição, a última imagem capturada é selecionada para o conjunto dos *keyframes*, e então realiza-se a regressão não-linear do BA tanto para rastrear a câmera quanto atualizar o mapa. É importante ressaltar a diferença ao procedimento usual de BA, em que se estuda apenas o conjunto completo de dados disponíveis e não há cálculo incremental dos parâmetros.

Estes princípios de rastreamento com BA estão sendo utilizados nos mais recentes sistemas de reconstrução 3D em tempo real, como [6,30,31]. Mas existem vários pontos em que estes sistemas ainda precisam ser desenvolvidos, como:

- Realização de ajustes em qualquer parte do mapa e a qualquer instante, e não apenas no início da inserção de novos parâmetros.
- Uso de bordas estruturais do ambiente como marcos.
- Detecção de estruturas de mais alto nível, como planos.
- Utilização de algumas técnicas clássicas de SLAM para ambientes extensos, como fechamento de laços e utilização de sub-mapas.
- Incorporação de leituras de sensores de movimento.

Alguns exemplos de sistemas que possuem algumas destas características, mas não utilizam técnicas de BA em tempo real são [12,32,33,34].

1.3 Objetivos

O objetivo do presente trabalho é desenvolver uma técnica para criar um modelo computacional sólido de um ambiente interno, como utilizado em computação gráfica, a partir de imagens obtidas por uma câmera em movimento dentro deste ambiente, e com resultados produzidos em tempo real. O modelo deve conter pisos e paredes de salas, assim como objetos grandes tais como batentes de portas, janelas, estantes e mesas. A própria forma e dimensões destes objetos torna necessário utilizar bordas como marcos visuais, e ainda utilizar um número relativamente pequeno de marcos por imagem, comparado com o grande número que se costuma encontrar em ambientes externos ou repletos de pequenos objetos.

Está sendo investigada no projeto a possibilidade de se detectar bordas apenas sobre algumas linhas e colunas fixas da imagem, o que é uma forma muito simples de análise, mas pode ser suficiente para muitas aplicações neste cenário onde bordas bastante extensas se deslocam ao redor da câmera [35]. A busca por contornos de objetos sobre uma linha já é bem estabelecida em Visão Computacional, mas não sobre uma grade fixa sobre a imagem como proposto aqui.

Boa parte dos mapas sendo produzidos hoje por sistemas de SLAM visual são constituídos apenas por pontos, por vezes associados a algum tipo de descritor. Apesar de reconstrução 3D de objetos com planos texturizados ser uma atividade comum (e.g. [31]), isto ainda não foi devidamente utilizado em mapeamento de ambientes. Isto se deve a dificuldades como o acúmulo de pequenas incertezas na localização da câmera em longos trajetos. Um modelo com planos texturizados das paredes de um ambiente fechado seria valioso por permitir realizar a segmentação do fundo para detectar objetos, uma técnica muito utilizada principalmente em aplicações com câmeras paradas e objetos em movimento. Um dos objetivos do trabalho descrito aqui é realizar esta segmentação com uma câmera em movimento e em tempo real.

Além desta pesquisa relativa à forma de detecção de marcos, o trabalho também investiga formas de realizar o procedimento de regressão não-linear global e em tempo-real para o levantamento de parâmetros. É proposto o uso de matrizes de condicionamento baseadas na transformada discreta de senos [7] para a aplicação do algoritmo do Gradiente Conjugado para a solução dos sistemas lineares necessários durante a otimização. Este algoritmo pode apresentar um excelente desempenho, além de permitir controlar o comprometimento entre

precisão e tempo de processamento. Este pode ser também o caminho para um eventual uso de algoritmos de otimização correlatos.

Além da detecção e identificação de marcos, e do procedimento de otimização, um sistema de reconstrução 3D em tempo real requer uma forma de inicialização dos marcos a partir apenas de detecções. Propomos o uso de técnicas oriundas da pesquisa em Transformada de Hough [36] para realizar isto.

2 Metodologia

Nesta seção o problema estudado é descrito em mais detalhes, e são apresentados a seguir alguns desenvolvimentos já realizados.

2.1 Detecção de marcos visuais

A maioria dos sistemas de SLAM visual se baseiam em algoritmos que extraem *pontos de interesse* de imagens, que ainda podem ser complementados com informações obtidas por descritores. Entre estes extratores/descritores podemos citar o de Harris, SIFT, SURF e FAST [37]. Todos localizam pontos de interesse sobre a imagem, se assumem ser as projeções de pontos no espaço tridimensional sendo mapeado. A partir das projeções de um conjunto de pontos em diferentes imagens estima-se a posição dos pontos no espaço, além do rastro da câmera.

Apesar da popularidade dos detectores de pontos, muita informação pode ser obtida também de curvas, especialmente linhas retas, que são muito comuns em ambientes internos. O uso deste tipo de marco visual já foi demonstrado com sucesso em outros sistemas [12,38]. A Figura 1 demonstra o funcionamento destes dois tipos de extratores. Na imagem da esquerda as quinas do retângulo central foram encontradas por *correlação cruzada normalizada* (NCC), como utilizado por [11]. Na imagem da direita foram detectados *edgels*, como proposto por [12]. Os mapas obtidos desta forma podem ser utilizados para rastrear uma câmera, ou podem ainda passar por análises subsequentes para inferir a existência de estruturas nos marcos detectados [32,39].

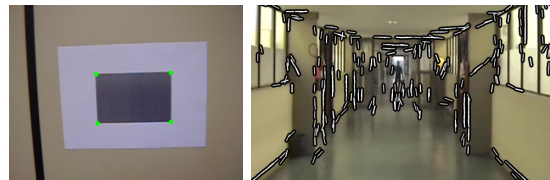


Figura 1. Exemplo de detecção de marcos pontuais (esquerda), e *edgels* (direita).

No presente trabalho está sendo investigado o uso de uma forma mais simples de detecção que pode vir a ser útil em aplicações de robótica móvel, ou em outras ocasiões em que há fortes restrições de movimento. Suponha que o ambiente

sendo capturado possui grandes bordas que cruzam a imagem de um lado ao outro. Isto ocorre quando uma câmara (com distância focal razoável) cruza uma porta, por exemplo, ou se aproxima de retas desenhadas no chão, ou de obstáculos como escadarias e tampos de mesas. Nestes casos as bordas podem ser facilmente detectadas apenas através de uma busca por picos na derivada dos valores de intensidade dos pixels ao longo da linha ou coluna centrais da imagem.

A Figura 2 demonstra o resultado de uma simulação feita com computação gráfica em que uma câmara se movimenta em direção à beirada de uma mesa dentro de uma sala de estar, carregada por um hipotético robô móvel. Pretende-se utilizar o SLAM visual neste caso para não só localizar a câmara, mas também detectar o obstáculo negativo constituído por esta beirada, que não deve ser ultrapassada. Para isto foi analisada a coluna central de pixels das imagens obtidas conforme a câmara movimentou-se em linha reta [40].

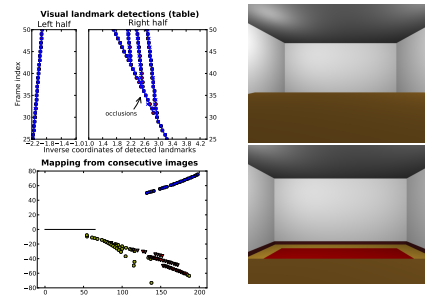


Figura 2. Rastreamento de bordas horizontais em simulação com CGI, e mapa produzido a cada novo quadro.

No primeiro gráfico da Figura 2 os pontos são o inverso das coordenadas de bordas detectadas. Para cada borda de cada imagem foi realizada uma busca na imagem seguinte por uma borda correspondente, utilizando comparações com os valores de intensidade dos pixels ao redor do ponto na primeira imagem. A partir de cada par de coordenadas associadas foi calculada uma posição no espaço do suposto marco visual, assumindo um deslocamento unitário da câmara em linha reta. O gráfico inferior mostra estas coordenadas estimadas para os marcos para cada par de imagens conforme a câmara se movimentou.

Esta técnica de detecção de bordas e mapeamento foi descrita em [35]. Um algoritmo mais sofisticado seria preciso para se obter melhores resultados e rastrear os marcos ao longo de um percurso. Porém esta forma rudimentar de mapeamento já permite uma capacidade mínima de detecção de obstáculos.

2.2 Estimação de Parâmetros

A seção anterior demonstrou formas de se detectar marcos visuais em imagens que podem ser utilizadas em sistemas de SLAM visual. Após estas detecções é

necessário empregar algum algoritmo de estimação de parâmetros para integrar todas as leituras em uma estimativa reiteradamente atualizada do rastro da câmera ou posições dos marcos.

Como foi explicado anteriormente, o uso de filtros possui desvantagens que se deseja evitar, e para isto estão se pesquisando formas de utilizar técnicas de BA em tempo real. Duas atividades sendo realizadas neste trabalho com este intuito são determinar boas formas de se inicializar marcos a partir de conjuntos de observações, e o uso de algoritmo do gradiente conjugado para realizar a otimização não-linear necessária ao BA em tempo real. O resto da seção detalha estas pesquisas.

2.3 Mapeamento por Transformada de Hough

Antes de realizar qualquer tipo de otimização para ajustar os parâmetros do mapa e do rastro da câmera, é preciso utilizar algum procedimento que recebe em sua entrada apenas os sinais detectados e uma estimativa do rastro, e propõe localizações iniciais de marcos a serem rastreados. Este procedimento se inicia com a associação das medições, e uma vez que as correspondências estejam disponíveis, as coordenadas do marco referente às observações são calculadas. Uma forma de realizar este cálculo é uma simples triangulação a partir de duas medições, como é feito no método de rastreamento em [28].

A localização de um marco é o ponto onde convergem as linhas determinadas pela posição do ponto focal da câmera e as coordenadas da projeção na imagem. As imprecisões existentes no processo de medição, além das variáveis desconhecidas, impedem que estas retas se encontrem com perfeição a partir das estimativas iniciais. O problema de estimar este ponto de convergência de retas com parâmetros ruidosos é muito similar ao problema da determinação de parâmetros de uma única reta em uma imagem utilizando a chamada *Transformada de Hough* (HT). Nesta técnica desenha-se em um espaço de parâmetros retas associadas a cada um dos pontos da imagem que pertencem à reta que se quer medir, e o lugar onde estas retas se cruzam no espaço de parâmetros são os parâmetros da reta no espaço de imagem [10].

A proposta introduzida aqui é utilizar técnicas desenvolvidas para realizar a HT de forma mais rápida e eficiente para realizar mapeamento. Foi escolhida especificamente a Transformada Rápida de Hough [36] para isto, e foram conduzidos testes com dados coletados com uma câmera real deslocando-se por um corredor. Os resultados foram relatados em [41].

2.4 Bundle Adjustment Utilizando Gradiente Conjugado

O algoritmo do Gradiente Conjugado é uma técnica iterativa para resolver sistemas lineares, baseada em subespaços de Krylov [42]. Este método não só é um dos mais indicados para obter soluções de alta precisão em sistemas muito grandes, como possui a vantagem de poder fornecer resultados intermediários. Ele também dá origem a algumas técnicas de otimização não-linear que poderiam ser utilizadas como alternativa à de LM, tal como a otimização de Fletcher-Reeves [20].

Apesar de tantas vantagens, o uso deste algoritmo em BA não é muito popular, apesar da possibilidade ser reconhecida [4]. Um motivo para isto pode ser a falta de matrizes de condicionamento adequadas para serem utilizadas no problema. Um recente artigo propôs o uso de uma matriz de condicionamento, entretanto, buscando reverter este quadro [8]. Aquela pesquisa se baseou em criar bases multi-escala para os parâmetros, com auto-vetores similares ao que se utiliza na transformada de Haar.

No presente trabalho está sendo investigada a possibilidade de se utilizar outra matriz de condicionamento para o problema, baseada na *Transformada Discreta de Senos* (DST), e proposta já há muitos anos como uma boa escolha para matrizes simétricas, especialmente as possuidoras da condição de Toeplitz e de banda limitada [7]. Uma vantagem desta matriz é a existência de algoritmos para o cálculo rápido da DST, e os experimentos realizados até agora estão demonstrando que esta pode ser mesmo uma escolha que explora características fundamentais do problema. Os primeiros resultados devem ser publicados em breve.

2.5 Modelagem de planos texturizados

O objetivo final do trabalho é criar mapas com planos texturizados que permitam realizar segmentação frente-fundo em tempo real e com a câmera em movimento. A etapa final de produção deste modelo ainda não começou a ser investigada, mas o caminho escolhido nos passos anteriores busca facilitar isto.

Já existem sistemas de SLAM visual que produzem mapas com estruturas geométricas de mais alto nível acima dos marcos pontuais observados [32]. O mapa é produzido por filtragem, mas avaliado por um processo complementar para detectar pontos livres que possam ser substituídos por pontos sobre retas ou planos. Porém a forma de detecção dos marcos não é influenciada por isto, estas estruturas inferidas não participam do processo completo de mapeamento.

O uso de BA ao invés de filtragem trará benefícios para a modelagem de planos porque se torna possível analisar dados obtidos no passado para levantar parâmetros das superfícies, incluindo cor, após a determinação de sua existência, enquanto com filtragem este tipo de análise só pode ser feita se a região do espaço em questão for novamente observada pela câmera após a necessidade surgir.

3 Conclusões

Foi apresentada a proposta e primeiros resultados de uma pesquisa em SLAM visual monocular em tempo real para ambiente internos. O trabalho segue a tendência atual de tentar aproveitar técnicas de BA para operação em tempo real, ao invés de utilizar métodos de filtragem. São utilizadas ainda bordas como principal tipo de marco visual, e o mapa produzido ao final é um modelo tridimensional com planos texturizados das paredes da sala ao redor da câmera, além de possíveis objetos de tamanho considerável.

Alguns dos componentes já estão em desenvolvimento, e testes com vídeos reais também já estão sendo realizados. Os próximos passos são a integração de todos componentes num único procedimento de rastreamento e estimação de parâmetros, e a seguir a criação do procedimento que inicializa superfícies texturizadas a partir da análise do mapa construído, e de dados armazenados.

Referências

1. Thrun, S.: Robotic mapping: a survey. In: Exploring artificial intelligence in the new millennium. Morgan Kaufmann Publishers Inc. (2003) 1–35
2. Durrant-Whyte, H., Bailey, T.: Simultaneous localization and mapping: part I. Robotics Automation Magazine, IEEE **13**(2) (june 2006) 99–110
3. Strasdat, H., Montiel, J.M.M., Davison, A.: Real-time monocular SLAM: Why filter? In: Robotics and Automation, 2010. IEEE Intern. Conf. on. (2010)
4. Triggs, B., McLauchlan, P.F., Hartley, R.L., Fitzgibbon, A.W.: Bundle adjustment - a modern synthesis. In: ICCV '99: Proceedings of the International Workshop on Vision Algorithms, London, UK, Springer-Verlag (2000) 298–372
5. Saxena, A., Chung, S.H., Ng, A.Y.: Learning depth from single monocular images. In: NIPS. (2005)
6. Klein, G., Murray, D.: Parallel tracking and mapping for small ar workspaces. In: Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on. (June 2008) 225–234
7. Chan, R.H., Ng, M.K., Wong, C.K.: Sine transform based preconditioners for symmetric Toeplitz systems. Lin. Alg. and its Applications **232** (1996) 237–259
8. Byrod, M., Astrom, K.: Bundle adjustment using conjugate gradients with multiscale preconditioning. In: British Machine Vision Conference. (2009)
9. Jeong, Y., Nister, D., Steedly, D., Szeliski, R., Kweon, I.S.: Pushing the envelope of modern methods for bundle adjustment. (jun. 2010) 1474–1481
10. Illingworth, J., Kittler, J.: A survey of the hough transform. Comput. Vision Graph. Image Process. **44**(1) (1988) 87–116
11. Davison, A., Reid, I., Molton, N., Stasse, O.: MonoSLAM: Real-time single camera SLAM. PTAM, IEEE Transactions on **29**(6) (2007) 1052–1067
12. Eade, E., Drummond, T.: Edge landmarks in monocular SLAM. Image Vision Comput. **27**(5) (2009) 588–596
13. Neira, J., Davison, A.J., Leonard, J.J.: Guest editorial special issue on visual SLAM. Robotics, IEEE Transactions on **24**(5) (oct. 2008) 929–931
14. Barra, R., Ribeiro, C., Costa, A.H.R.: Fast vertical line correspondence between images for mobile robot localization. In: 9th International IFAC Symposium on Robot Control (SYROCO2009). (2009) 153–158
15. Kim, S., Oh, S.Y.: SLAM in indoor environments using omni-directional vertical and horizontal line features. J. Intell. Robotics Syst. **51**(1) (2008) 31–43
16. Wongphati, M., Niparnan, N., Sudsang, A.: Bearing only FastSLAM using vertical line information from an omnidirectional camera. In: ROBIO '09: Proceedings of the 2008 IEEE International Conference on Robotics and Biomimetics, Washington, DC, USA, IEEE Computer Society (2009) 1188–1193
17. Solà, J.: Consistency of the EKF-SLAM algorithm for three different landmark parametrizations. In: 2010 IEEE Intern. Conf. on Robotics and Automation. (2010)
18. Chekhlov, D., Pupilli, M., Mayol-cuevas, W., Calway, A.: Real-time and robust monocular slam using predictive multi-resolution descriptors. In: In 2nd International Symposium on Visual Computing. (2006)

19. Pupilli, M., Calway, A.: Real-time camera tracking using a particle filter. In: In Proc. British Machine Vision Conference. (2005) 519–528
20. Seber, G.A.F., Wild, C.J.: Nonlinear Regression. Wiley series in probability and mathematical statistics. John Wiley & Sons, Inc. (1989)
21. Dellaert, F.: Monte Carlo EM for Data-Association and its Applications in Computer Vision. PhD thesis, Robotics Institute, Carnegie Mellon University (2001)
22. Dellaert, F., Seitz, S., Thorpe, C., Thrun, S.: Feature correspondence: A Markov chain Monte Carlo approach. In: Advances in Neural Information Processing Systems 13 (NIPS 2000). (2001)
23. Kaess, M., Dellaert, F.: Probabilistic structure matching for visual SLAM with a multi-camera rig. *Comp. Vision and Image Understanding* **114** (2010) 286–296
24. Liu, Y., Emery, R., Chakrabarti, D., Burgard, W., Thrun, S.: Using em to learn 3d models of indoor environments with mobile robots. In Brodley, C.E., Danyluk, A.P., eds.: ICML, Morgan Kaufmann (2001) 329–336
25. Choi, S., Kim, T., Yu, W.: Performance evaluation of RANSAC family. In: 20th British Machine Vision Conference. (2009)
26. Torr, P.H.S., Zisserman, A.: MLESAC: A new robust estimator with application to estimating image geometry. *Comp. Vision and Image Understanding* **78** (2000)
27. Ke, Q., Kanade, T.: Quasiconvex optimization for robust geometric reconstruction. In: In International Conference on Computer Vision. (2005) 986–993
28. Mouragnon, E., Lhuillier, M., Dhome, M., Dekeyser, F., Sayd, P.: Real time localization and 3d reconstruction. In: CVPR '06: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Washington, DC, USA, IEEE Computer Society (2006) 363–370
29. Nistér, D., Naroditsky, O., Bergen, J.: Visual odometry. In: CVPR '04: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. (2004) 652–659
30. Newcombe, R., Davison, A.: Live dense reconstruction with a single moving camera. In: CVPR '10: Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE Computer Society (2010)
31. Pan, Q., Reitmayr, G., Drummond, T.: ProFORMA: Probabilistic Feature-based On-line Rapid Model Acquisition. In: Proc. 20th BMVC. (2009)
32. Gee, A.P., Chekhlov, D., Calway, A., Mayol-Cuevas, W.: Discovering higher level structure in visual SLAM. *IEEE Transactions on Robotics* **24**(5) (2008) 980–990
33. Schleicher, D., Bergasa, L.M., Ocaña, M., Barea, R., López, E.: Real-time hierarchical stereo visual SLAM in large-scale environments. *Robotics and Autonomous Systems* **In Press, Corrected Proof** (2010) –
34. Kaess, M., Ranganathan, A., Dellaert, F.: iSAM: Incremental smoothing and mapping. *IEEE Trans. on Robotics, TRO* **24**(6) (Dec 2008) 1365–1378
35. Werneck, N.L., Costa, A.H.R.: Mapping with monocular vision in two dimensions. *International Journal of Natural Computing Research* **1** (2010) In press.
36. Li, H., Lavin, M.A., Le Master, R.J.: Fast hough transform: A hierarchical approach. *Computer Vision, Graphics and Image Processing* **36**(2-3) (1986) 139–161
37. Tuytelaars, T., Mikolajczyk, K.: Local invariant feature detectors: A survey. *Foundations and Trends in Computer Graphics and Vision* **3**(3) (2007) 177–280
38. Smith, P., Reid, I., Davison, A.: Real-time monocular slam with straight lines. In: British Machine Vision Conference. Volume 1. (September 2006) 17–26
39. Flint, A., Mei, C., Reid, I., Murray, D.: Growing semantically meaningful models for visual slam. (jun. 2010) 467–474

40. Werneck, N.L., Costa, A.H.R., Truzzi, F.S.: Medição de distância e altura de bordas horizontais com visão monocular linear para robôs móveis. In: Anais do V Workshop de Visão Computacional. (2009)
41. Werneck, N., Costa, A.H.R.: Monocular visual mapping with the fast hough transform. In: VI Workshop de Visão COmputacional. (2010)
42. Golub, G.H., Van Loan, C.F.: Matrix Computations (Johns Hopkins Studies in Mathematical Sciences). 3rd edn. The Johns Hopkins University Press (1996)