

Segmentação de Regiões de Interesse em Mapas de Disparidade

Thiago Rateke^{1,3}, Marcelo Dornbusch Lopes¹, Antonio Carlos Sobieranski^{1,2}, Eros Comunello^{1,2}, Aldo von Wangenheim¹

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

² 4Vision Lab – Universidade do Vale do Itajaí – Florianópolis – SC – Brasil

³ Programa de Pós-Graduação em Ciência da Computação, Universidade Federal de Santa Catarina (PPGCC / UFSC) – Florianópolis – SC – Brasil

{thiago, marcelo}@incod.ufsc.br, {asobieranski, eros.com}@univali.br, awangenh@inf.ufsc.br

Abstract. *This paper presents the concepts and steps involved in stereoscopic vision, since the camera calibration step to the stereo matching methods, for this we used available functions in the OpenCV library. This work purpose use a threshold on the disparity map, excluding failures and poor correlations between pixels, leaving only the region of greatest interest, which is the person or object closer to the capture sources. The results obtained demonstrate that post-processing can eliminate the distant objects to the capture sources and failures in correlation. In addition, make clear the possibilities of future work based on the post-processing disparity map.*

Resumo. *Este trabalho apresenta os conceitos e etapas que envolvem a visão estereoscópica, desde a etapa de calibração das câmeras até os métodos de correspondência estéreo; para isso foram utilizadas as funções disponíveis na biblioteca OpenCV. A proposta deste trabalho é utilizar um Threshold no mapa de disparidade, buscando assim eliminar ruídos e más correlações entre pixels, deixando somente a região de maior interesse, que é a pessoa ou objeto mais próximo às fontes de captura. Os resultados obtidos demonstram que o pós-processamento pode eliminar as correlações mais distantes, ruídos e falhas da correlação. Além disso, deixa claro as possibilidades de trabalhos futuros baseados no pós-processamento em mapa de disparidade.*

1. Introdução

Sistemas de Visão Estéreo (VE) são alimentados por duas fontes de captura com posições distintas. Duas imagens bidimensionais são então adquiridas, com conteúdos ligeiramente diferentes devido ao seu posicionamento, bem como a diferença na incidência de luz. Tais imagens são submetidas então a algoritmos de correspondência e correlação, com a finalidade de obter-se um Mapa de Disparidade (MP) [Bleyer 2010].

Algumas das possíveis aplicações que se tem com a visão estéreo são: reconstruções 3-D a partir de imagens aéreas, com câmeras colocadas em aviões para obter um mapa do terreno [Hirschmuller e Scharstein 2009]; reconstruções em 3-D das cidades [Agarwal 2009]; carros autônomos, assistente de motorista [Klette et al 2008] podendo medir a distância para outros veículos.

Tem-se também a navegação de robôs móveis; como por exemplo, os exploradores de Marte (Spirit e Opportunity) que reconstroem a superfície de Marte usando visão estéreo. Outra aplicação é a captura de movimento humano para a interação homem-máquina [Plankers e Fua 2003]. E por fim, bicamada de segmentação (Z-Keying) dividindo a imagem em *foreground* e *background* [Criminisi et al. 2006].

No entanto, a correspondência estéreo é um dos principais problemas encontrados em visão computacional. Para se obter o MD é preciso levar em consideração alguns fatores tais como: mudanças espaciais nas posições dos pixels correspondentes, ambiguidade, ruído na imagem, repetições e oclusões [Aydin e Akgul 2010].

Neste trabalho é apresentada uma proposta de segmentação de objetos de interesse a partir do MD obtido utilizando a biblioteca OpenCV. Buscando com isso melhorar os resultados obtidos com a utilização de outras técnicas de processamento digital de imagens, como por exemplo, utilizar um *threshold* para separar somente as regiões mais próximas e de maior interesse no mapa de disparidade da cena capturada.

A seguir são listados alguns dos principais motivos que justificam a complexidade envolvida na visão estereoscópica:

- Inconsistência de cores: Quando se resolve o problema da correspondência estéreo, assume-se tipicamente que os pixels correspondentes têm a mesma intensidade (cor). Porém isso pode não ser verdade, pois pode haver ruídos na imagem, diferentes condições de iluminação entre a imagem esquerda e a imagem direita, diferentes características dos sensores nas duas câmeras, espelhamento e entrelaçamento de objetos [Bleyer 2010].
- Regiões sem textura: É preciso haver certa quantidade de variação de intensidade (cor), textura. De modo que um pixel possa ser unicamente correspondido em outro ponto de vista. Por exemplo, como descrever a profundidade se estiver parado em frente a uma parede branca? [Bleyer 2010].
- Problema de Oclusão: É difícil encontrar disparidade em pontos correspondentes que não existem. Há pixels que somente são visíveis numa exata visão (esquerda ou direita), podem-se chamar esses pixels de “pixels oclusos” ou “parcialmente oclusos”. É difícil estimar a profundidade para esses pixels. Os problemas de oclusão fazem da visão estéreo uma das áreas mais desafiantes da visão computacional [Bleyer 2010].

Pressupostos são necessários para resolver o problema da correspondência estéreo:

- Pressuposto de Consistência de Cores: Pixels correspondentes têm a mesma intensidade (cor) nas duas imagens.

- Pressuposto Epipolar: O ponto de correspondência de um pixel tem que ficar na mesma linha de varredura (horizontal) da outra imagem.
- Pressuposto da Ordenação: A ordem em que os pixels ocorrem é preservada nas duas imagens.

Antes de montar o mapa de disparidade é feito uma inicialização de dados para encontrar as correspondências estéreo, para isso é utilizado uma função *cvCreateStereoBMState()*, que cria um bloco com a estrutura para a correspondência estéreo e inicializa esse bloco. Os parâmetros podem ser substituídos antes das chamadas da função de correspondência estéreo *cvFindStereoCorrespondenceBM()*. A variável que recebe a função *cvCreateStereoBMState()* é chamada de *BMState* do tipo *CvStereoBMState*, que é a estrutura para o bloco de inicialização de correspondência estéreo. A função procura apenas os blocos correspondentes na direção *x*, isso significa que o par de imagens deve ser retificado.

Então o método *cvFindStereoCorrespondenceBM()* pode enfim ser chamado. Esse método é o responsável pelo cálculo do mapa de disparidade e utiliza para isso as imagens capturadas, esquerda e direita, a estrutura de correspondência estéreo (*BMState*) que foi inicializada logo após a retificação estéreo e retorna o mapa de disparidade para essa cena. Um exemplo com resultado do mapa de disparidade obtido pela biblioteca da OpenCV pode ser visto na Figura 1, que mostra a cena capturada (pela câmera esquerda e pela câmera direita) e o mapa de disparidade desta cena.

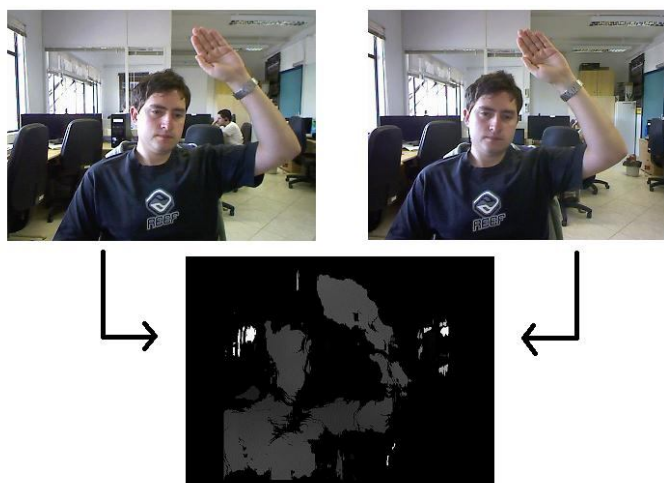


Figura 1 Mapa de disparidade

O trabalho está organizado da seguinte forma: Capítulo 2 apresenta uma fundamentação sobre a visão estéreo e os métodos da biblioteca OpenCV utilizados, falando sobre calibração das câmeras, retificação das imagens, correspondência estéreo, mapas de disparidade e sobre alguns trabalhos correlatos para a correspondência estéreo. O Capítulo 3 apresenta os resultados obtidos com este trabalho. E por fim, Capítulo 4 apresenta as conclusões.

2. Visão Estereoscópica

Segundo [Zureiki, Devy e Chatila 2007] pode-se chamar de visão estéreo binocular o uso de duas imagens captadas por duas câmeras, onde a fase preliminar da calibração é

necessária para estimar os diferentes parâmetros de um equipamento estéreo e as relações espaciais entre as duas câmeras. Esse conhecimento permite o cálculo das coordenadas 3D de um ponto de suas projeções nas duas imagens através de uma triangulação simples.

As etapas contidas em um sistema de visão estereoscópica com uso de câmeras comuns (*webcams*) iniciam-se na calibração das câmeras, obtendo dados, como a projeção dos pontos em duas dimensões da cena capturada, valores internos (ponto focal e centro da imagem) e externos da câmera (para rotação e translação), dados esses que serão utilizados posteriormente.

Após a calibração das câmeras pode-se realizar a captura das imagens da cena (esquerda e direita) pelas fontes de captura utilizadas, e então, é realizada a etapa de retificação das imagens colocando as duas imagens no mesmo plano e podendo ser utilizadas na correspondência estéreo e correlação entre os pixels das imagens através da disparidade entre eles, construindo o mapa de disparidade. Ao final, consegue-se a profundidade da cena, podendo gerar a cena em três dimensões virtualmente.

2.1. Calibração de Câmeras

Para um sistema de visão estereoscópica, ambas as fontes de captura devem passar por uma etapa de calibração inicial, onde sejam adquiridos os parâmetros internos (ponto focal e centro da imagem) e parâmetros externos (rotação e translação) para que as imagens capturadas de uma mesma cena possam ser devidamente alinhadas. Este é um dos problemas mais desafiadores para a visão estereoscópica [Koller 1997].

Para a etapa de calibração pode-se, por exemplo, focar em uma calibração automática, não necessitando de um objeto padrão (como o tabuleiro de xadrez), buscando por bordas de objetos na cena, ou pontos em comum de forma automática e usando estes pontos como objetos padrões sem necessitar de uma etapa só de captura para a calibração. Alguns trabalhos que tratam da calibração automática podem ser vistos em, [Koller 1997], [Deutscher, Isard e McCormick 2002] e [Liu e Hubbold 2006].

Para este trabalho a calibração é necessária para que se possam obter os parâmetros das câmeras e utilizar esses dados para a etapa de retificação posteriormente. Os parâmetros obtidos são os parâmetros intrínsecos e os parâmetros extrínsecos, onde parâmetros intrínsecos são os parâmetros internos da câmera com suas características ópticas e geométricas como ponto focal e ponto principal (normalmente centro da imagem).

Já os parâmetros extrínsecos são os parâmetros externos e passam as informações de posição e orientação da câmera em relação a um sistema de coordenadas, podendo com isso, descrever o movimento de uma cena estática em torno da câmera ou o oposto. Os parâmetros extrínsecos descrevem uma mudança da representação no sistema de coordenadas através da rotação e translação de um ponto 3D (x, y, z) para algum sistema de coordenadas fixo em relação à câmera.

As coordenadas 3D dos pontos e suas projeções 2D correspondentes em cada cena (captura de cada câmera) devem ser especificadas. Isso pode ser alcançado através de um objeto com geometria conhecida e pontos com características mais acessíveis de serem detectadas.

Esse objeto é chamado de padrão de calibração, e a biblioteca OpenCV tem suporte para um método que usa como padrão o tabuleiro de xadrez, conforme Figura 2, e pode ser utilizado fazendo uma chamada a função *cvFindChessboardCorners()*. Essa função busca por um número determinado de bordas no tabuleiro, que para este trabalho ficou definido em 54 pontos do tabuleiro de xadrez, sendo nove colunas e seis linhas (com tabuleiro na horizontal).



Figura 2 Padrão de calibração

A função *cvFindChessboardCorners()* busca determinar se a imagem de entrada tem o padrão de tabuleiro de xadrez e localizar os cantos internos. A função retorna um valor diferente de zero caso todos os cantos tenham sido encontrados e eles são colocados em uma determinada ordem, linha por linha, da esquerda para a direita em cada linha, uma vez que a função não encontre todos os cantos é retornado zero.

E finalmente, para realizar a calibração estéreo é utilizado o método *cvStereoCalibrate()*, que tem como parâmetros de entrada, a quantidade de pontos do padrão de tabuleiro de xadrez, as projeções dos pontos no sistema de coordenadas de cada câmera, tamanho das imagens utilizadas (altura e largura) e tem como saída os dados de distorção das câmeras, matrizes de parâmetros intrínsecos, matriz de rotação entre os sistemas de coordenadas das duas câmeras, vetor de translação, matriz essencial e matriz fundamental.

Esses dados são armazenados, e contanto que não haja alteração no foco ou posicionamento das câmeras, os mesmos dados de calibração podem ser reutilizados mais vezes.

Deve-se utilizar uma quantidade mínima de imagens com o padrão conhecido para a calibração, obtendo-se assim pontos suficientes para o sistema de coordenadas e melhora os dados a serem utilizados para as etapas seguintes. Neste trabalho foram utilizados sete pares de imagens para cada calibração realizada.

2.2. Retificação de Imagens

A geometria epipolar define a geometria de imagens capturadas por duas câmeras, ou seja, um sistema estereoscópico [Taraglio e Chiesa 2011]. Geometria epipolar entre duas visões de duas fontes é a geometria da intersecção dos planos das imagens com as linhas de varredura dos planos tendo como eixo a linha de base (é a linha de base que une os

centros das câmeras). Essa geometria é necessária para a busca de pontos correspondentes na etapa de Correspondência Estéreo [Hartley e Zisserman 2004].

Com a finalização do processo de retificação as imagens no plano esquerdo e do plano direito são colocadas em um plano comum entre as duas, o eixo x é paralelo a Linha de Base de plano comum, que se pode chamar de plano epipolar. As linhas epipolares coincidem com as linhas de varredura (*scanlines*) horizontais. Os pixels correspondentes têm a mesma coordenada y ; então para encontrar o pixel correspondente, tem-se apenas que procurar ao longo da linha de varredura (*scanline*) horizontal, mais conveniente do que o rastreamento arbitrário das linhas epipolares [Bleyer 2010] e [Kumar et al. 2010].

As entidades geométricas envolvidas na geometria epipolar são: (1) Linha de Base, ou também chamado de epipole, que é o ponto de intersecção da linha que une o centro das câmeras com o plano da imagem. (2) Plano Epipolar, que é um plano que contém a linha de base. Existe um parâmetro familiar (uma linha) de planos epipolares. (3) Linha Epipolar, que é a intersecção de um plano epipolar com o plano da imagem.

Todas as linhas epipolares cruzam-se na Linha de Base (epipole). Um plano epipolar intersecta os planos das imagens esquerda e direita em linhas epipolares, definindo assim, a correspondência entre essas linhas [Hartley e Zisserman 2004].

Para a realização da retificação das imagens neste trabalho, primeiramente utiliza-se o método *cvComputeCorrespondEpilines()*, que para cada ponto em uma das duas imagens do par de imagens capturadas, essa função localiza as linhas epipolares correspondentes aquele ponto na outra imagem do par.

Então se utiliza o método *cvStereoRectify()* que calcula as matrizes de rotação de cada câmera fazendo com que os planos das duas imagens (esquerda e direita) fiquem no mesmo plano, e conseqüentemente deixando as linhas epipolares em paralelo, facilitando assim o trabalho de correspondência estéreo posteriormente. Como entrada, essa função pega as matrizes calculadas pelo *cvStereoCalibrate()* e como saída devolve duas matrizes de rotação e duas matrizes de projeção dos pontos, com novas coordenadas.

Ao final desta função os dados da câmera serão orientados de forma diferente no espaço de coordenadas, de acordo com as matrizes de rotação com as transformações da função *cvStereoRectify()*, isso faz com que as linhas epipolares entre as duas imagens tornem-se paralelas e tenham o mesmo valor na coordenada y (no caso deste trabalho, onde foram utilizadas câmeras na horizontal), na Figura 3 um exemplo é apresentado.



Figura 3 Retificação de imagem

2.3. Correspondência Estéreo

Segundo [Zureiki, Devy e Chatila 2007] a definição de correspondência estereo é o problema de encontrar nas imagens esquerda e direita os pontos homólogos, ou seja, os pontos que são as projeções da mesma entidade na cena. A correspondência estereo é um problema de minimização, seja para muitos problemas de energia, representando os custos locais de cada correspondência, onde são independentes dos outros custos para outras correspondências, ou para uma função de energia global única, que representa o custo global de correspondência entre os pontos das imagens.

Já [Gu *et al.* 2008] descreve a correspondência estereo como o processo de encontrar pontos correspondentes entre duas ou mais imagens. É um dos assuntos mais importantes e um desafio em visão por computador.

Métodos de análise de correspondência podem ser classificados como métodos globais ou como métodos locais. Métodos globais, resolvem um problema de otimização para uma linha de varredura ou uma imagem inteira, considerando uma função de custo global sujeita a uma restrição de suavidade. A análise de correspondência é realizada linha por linha. Cada pixel de interesse na imagem da esquerda é comparado com cada pixel de interesse na linha correspondente epipolar na imagem à direita [Barrois *et al.* 2010].

Métodos globais expressam o pressuposto de suavidade de uma forma explícita. O objetivo é encontrar um MD de baixo custo. Então, define-se uma função de custo, onde altos custos significam um MD ruim e baixos custos significam que o MD é bom [Scharstein e Szeliski 2002].

Já os métodos locais calculam a disparidade de um determinado ponto com base nos valores de intensidade em uma janela finita [Barrois *et al.* 2010]. Os métodos locais, comparados com os métodos globais, têm abordagens mais eficientes e com estrutura mais simples. Entretanto, para pontos em regiões homogêneas pode ser difícil de encontrar perfeitamente a correspondência desses pixels em outras imagens usando métodos baseados em janela. Então, os métodos locais geralmente exigem dois fatores importantes para determinar as medidas de correlação e a correspondência de janelas locais [Gu *et al.* 2008].

2.4. Mapa de Disparidade

A disparidade é a diferença que um mesmo pixel tem entre as duas imagens, ou seja, é o quanto esse pixel é “deslocado” de uma imagem em relação à outra. A disparidade de cada pixel é codificada por um valor de cinza, onde altos valores de cinza representam altas disparidades e baixos valores representam baixas disparidades.

A disparidade de cada pixel é inversamente proporcional à sua profundidade na cena, portanto é comumente utilizada como sinônimo de profundidade. Então, ao final da correspondência estereo o resultado é o Mapa de Disparidade, que contém informações suficientes para gerar um modelo 3D [Bleyer 2010].

3. Pós-processamento

Com os resultados obtidos até este momento, observa-se que o mapa de disparidade, apesar de ser possível identificar a pessoa ou o objeto mais próximo na cena, fica com

ruídos e falhas na correlação. Então, foi pensado num pós-processamento desse mapa de disparidade resultante da correlação de pixels através da função *cvFindStereoCorrespondenceBM()* da biblioteca OpenCV.

O objetivo de se fazer um pós-processamento é refinar e buscar um mapa de disparidade com menos ruídos, deixando somente as regiões de maior interesse, que é o que está mais próximo da fonte de captura (objeto de interesse).

Para realizar esse pós-processamento é utilizado um *threshold* na imagem do mapa de disparidade, que basicamente divide a imagem por um ponto de corte definido e comumente chamado de nível de *threshold*.

Após a definição do nível de *threshold* a imagem torna-se uma cena em preto e branco, onde os pixels pretos são os valores que ficaram abaixo do nível de *threshold* e os pixels brancos são os valores que ficaram acima do nível de *threshold*.

Após o pós-processamento tem-se como resultado as imagens vistas na Figura 4 onde se observa que o *threshold* deixou somente as regiões mais próximas da fonte de captura (objeto de interesse).

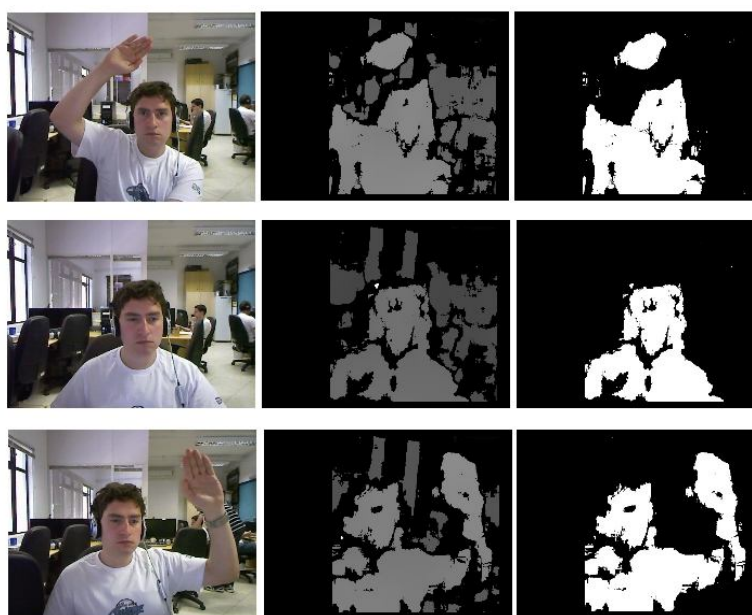


Figura 1 Pós-processamento

4. Conclusões

Através dos resultados obtidos, foi possível observar que com os mapas de disparidade é possível identificar pessoas ou objetos próximos à fonte de captura. Importante ressaltar que não foi encontrada nenhuma métrica para avaliação dos resultados do mapa de disparidade. Mesmo assim, fica constatado a necessidade de melhorias que podem ser feitas, buscando aperfeiçoar o resultado.

Na busca por extrair somente o objeto de interesse do mapa de disparidade resultante da correlação entre imagens de uma cena, este trabalho realiza um pós-processamento do mapa de disparidade. Para isso foi utilizado um nível de *threshold*

para separar as regiões mais próximas da fonte de captura, que no mapa de disparidade são representadas por um cinza mais claro. Com isso eliminou-se as regiões mais distantes com o nível de cinza abaixo do nível de *threshold*.

Este trabalho além de apresentar os conceitos e etapas envolvidas na visão estereoscópica, mostrou ser possível extrair um objeto de interesse (ex.: obstáculo mais próximo da fonte de captura) do mapa de disparidade obtido. Com isso, abre-se uma série de possibilidades e trabalhos futuros, onde diferentes técnicas de processamento digital de imagens podem melhorar os resultados da correlação de pixels e mapa de disparidade.

Referências

- Agarwal, S., Snavely, N., Simao, I., Szeliski, R. (2009) “Building Roma in a Day”, ICCV.
- Aydin, T., Akgul, Y.S. (2010) “Stereo depth estimation using synchronous optimization with segment based regularization”, Pattern Recognition Letter. Ed.31.
- Barrois, B., Konrad, M., Wohler, C., Gro, H.M. (2010) “Resolving stereo matching errors due to repetitive structures using model information”, Pattern Recognition Letters, Ed.31.
- Bleyer, M. (2010) “VU Stereo Vision”, Technical Report, Interactive Media Systems Group, Institute for Software Technology and Interactive Systems, Vienna University of Technology, http://www.ims.tuwien.ac.at/teaching_detail.php?ims_id=188.HQK, September.
- Criminisi, A., Cross, G., Blake, A., Kolmogorov, V. (2006) “Bilayer Segmentation of Live Video”, Microsoft Research Ltd, Cambridge, United Kingdom.
- Deutscher, J., Isard, M., McCormick, J. (2002) “Automatic camera calibration from a single Manhattan image”, ECCV '02 Proceedings of the 7th European Conference on Computer Vision-Part IV.
- Gu, Z., Su, X., Liu, Y., Zhang, Q. (2008) “Local stereo matching with adaptive support-weight, rank transform and disparity calibration”, Pattern Recognition Letters.
- Hartley, R., Zisserman, A. (2004) “Multiple View Geometry in computer vision”, 2.ed. Cambridge University Press, Cambridge, USA.
- Hirschmuller, H., Scharstein, D. (2009) “Evaluation of Stereo Matching Costs on Images with Radiometric Differences”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume 31.
- Klette, R., Liu, Z., Guan, S., Vaudrey, T., Morales, S., Sanchez, J., Milburn, J. (2008) “Stereo-Vision-Based Driver Assistance – Predict Situations and Automatically Adapt to Traffic”, The University of Auckland, Tamaki Campus, Auckland.
- Koller, D., Klinder, G., Rose, E., Breen, D., Whitaker, R., Tuceryan, M. (1997) “Automated Camera Calibration and 3D Egomotion Estimation for Augmented Reality Applications”, Proceedings of the 7th International Conference on Computer Analysis of Images and Pattern.

- Kumar, S., Micheloni, C., Piciarelli, C., Foresti, G.L. (2010) "Stereo rectification of uncalibrated and heterogeneous images", *Pattern Recognition Letters*, Ed. 31.
- Liu, J., Hubbold, R. (2006) "Automatic Camera Calibration and Scene Reconstruction with Scale-Invariant Features", *Advances in Visual Computing*, Vol. 4291.
- Plankers, R., Fua, P. (2003) "Articulated Soft Objects for Multiview Shape and Motion Capture", *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
- Scharstein, D., Szeliski, R. (2002) "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms", *International Journal of Computer Vision*, Ed.47.
- Taraglio, S., Chiesa, S. (2011) "Evolutionary Approach to Epipolar Geometry Estimation", ENEA, Robotics Lab, Roma, Itália.
- Zureiki, A., Devy, M., Chatila, R. (2007) "Stereo matching reduced-graph cuts", *IEEE, International Conference Image Processing*, T. San Antonio, Ed., Vol. 1.