

Recuperação de Objectos VRML por esboços

Estudo Prático

Tiago Brito Manuel J. Fonseca Joaquim A. Jorge

INESC-ID/IST/UTL

R. Alves Redol, 9, 1000-029 Lisboa

tmtb@mega.ist.utl.pt, mjf@inesc-id.pt, jorgej@acm.org

Resumo

Este artigo compara a utilização de dois métodos de descrição de formas para a realização de um motor de busca de objectos tridimensionais, utilizando os seus contornos como meio de pesquisa a partir de um esboço criado pelo utilizador. A informação sobre cada objecto é obtida na forma de descritores gerados pela biblioteca de reconhecimento CALI ou utilizando Momentos de Zernike. Cada objecto tridimensional é processado apenas uma vez, sendo criadas dez vistas de cada objecto. Estas são depois utilizadas para calcular os contornos exteriores dos objectos, que servirão para criar os descritores atrás referidos.

Palavras chave

CALI, Momentos de Zernike, Recuperação de Objectos Tridimensionais, Recuperação Baseada em Esboços.

1. INTRODUÇÃO

A crescente existência e utilização de programas que criam e manipulam objectos tridimensionais tende a aumentar a disponibilidade e procura de tais objectos para utilização em aplicações virtuais. Para tais aplicações, é mais fácil procurar objectos já feitos do que perder tempo a criá-los de raiz. A Internet é uma boa fonte de informação onde a disponibilidade desses objectos tridimensionais é grande, mas mesmo assim é difícil encontrá-los, ou pelo menos difícil encontrar um que corresponda às nossas necessidades.

O *VRML Finder* surgiu dessa ideia, criar um motor de busca para objectos tridimensionais. Embora tenha como base de partida o trabalho de Ding-Yun Chen [Chen03], esta implementação não está preparada para funcionar numa página de Web, mas a aproximação pode ser utilizada para esse efeito.

Com este desafio lançado, propôs-se a comparação de dois métodos de descrição de formas (a biblioteca CALI e Momentos de Zernike).

A ideia base da busca de objectos VRML é a redução do problema 3D ao 2D, isto é, a pesquisa na base de dados baseia-se no princípio de que, se dois objectos são semelhantes, então têm vistas semelhantes. Sendo assim, o que se pretende é, para um dado esboço criado pelo utilizador, calcular os descritores respectivos e comparar estes com os descritores das várias vistas 2D de cada objecto. Se os descritores do esboço e os descritores de pelo menos uma dessas vistas forem semelhantes, considera-se que o objecto faz parte da lista de resultados.

2. TRABALHO RELACIONADO

Existem vários projectos sobre recuperação de objectos baseada em esboços, para diversas aplicações desde a recuperação de objectos multimédia, passando pelo reconhecimento de gestos, até às aplicações em sistemas de CAD que foram utilizados como referencia para esta aplicação. Um destes trabalhos consiste num motor de busca 3D desenvolvido por Ding-Yun Chen [Chen03], que se baseia em Momentos de Zernike e Descritores de Fourier para descrever as formas na base de dados. A metodologia de extracção de vistas do VRML Finder foi baseada neste trabalho.

Outro trabalho estudado consiste num reconhecedor de gestos que utiliza Momentos de Zernike, para descrever simbologia desenhada pelo utilizador. Este trabalho é o HHReco, desenvolvido por Heloise Hse [Hse03b] que contribuiu para um melhor conhecimento do algoritmo computacional que transforma contornos em descritores utilizando os momentos de Zernike.

[Elad02] apresenta uma aproximação iterativa e interactiva para pesquisa de objectos VRML. O utilizador vai refinando os resultados obtidos ao longo de várias pesquisas (ou iterações), indicando, em cada uma, quais os resultados que menos se parecem com o resultado esperado, de modo a que o algoritmo tenha uma componente de aprendizagem e apresente uma grande melhoria em cada uma das iterações. Este trabalho utiliza Momentos Geométricos para gerar os descritores, utilizando o algoritmo de *Support Vector Machine* para a decisão sobre os objectos mais próximos do esboço.

Por fim, foi estudado o SBR (*Sketch Based Retrieval*) de Manuel J. Fonseca e de Alfredo Ferreira [Fonseca03b], onde foram baseadas as ideias de utilização do CALI, e da estrutura de indexação. Esta aplicação procura desenhos técnicos a partir de esboços, utilizando informação vectorial presente nos ficheiros a pesquisar.

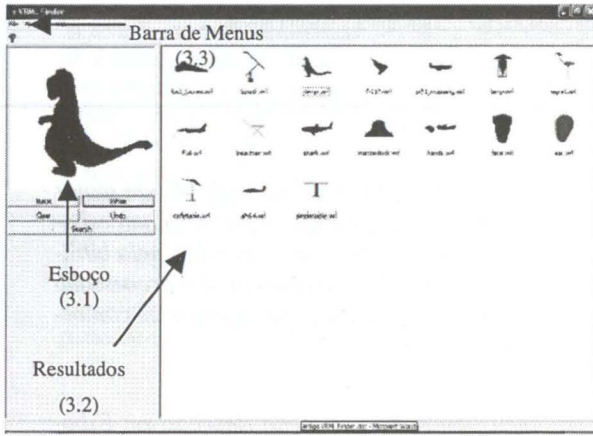


Figura 1: Interface da aplicação.

3. A APLICAÇÃO

A aplicação desenvolvida tem uma interface simples, com um campo onde os utilizadores podem desenhando interrogações a uma base de dados, sob a forma de esboços, e outra área onde serão exibidos os resultados da pesquisa. Estes resultados são apresentados sob a forma de pequenas imagens dos objectos, com o nome do ficheiro correspondente.

A interface inclui ainda uma barra de menus onde se pode mudar o método de descrição de formas geométricas, isto é, escolher entre o CALI e os Momentos de Zernike.

Na Figura 1 podemos ver o esboço de um dinossauro na esquerda, e os respectivos resultados da pesquisa à direita.

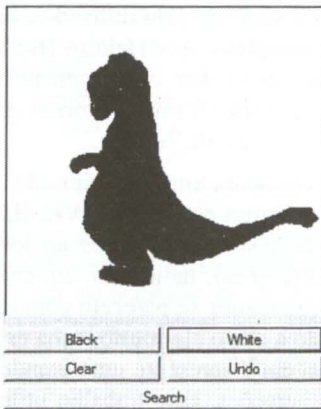


Figura 2: Região de esboço

3.1 Região de esboço

A região de esboço (figura 2) é uma área 256x256 pixels, onde o utilizador pode desenhando uma forma, que será usada para pesquisar a base de dados. Por baixo desta região do ecrã estão cinco botões com as seguintes funcionalidades: desenhando a preto, desenhando a branco (criar buracos no desenho), limpar o conteúdo da janela, desfazer a última acção de desenho e efectuar a procura (ver figura 2).

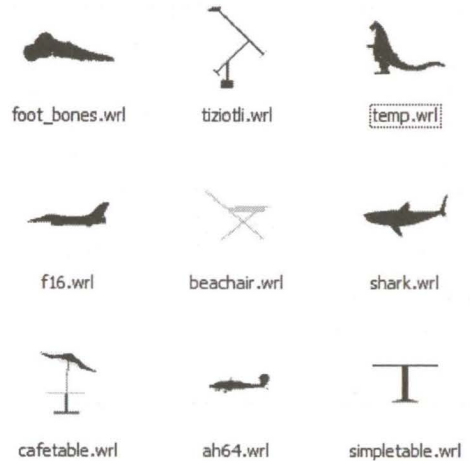


Figura 3: Região de resultados

3.2 Região de resultados

Na região de resultados são exibidos os resultados na forma de pequenas imagens (que representam o objecto exactamente como vem no ficheiro respectivo, sem qualquer rotação) com a respectiva legenda indicando o nome do ficheiro correspondente (figura 3).

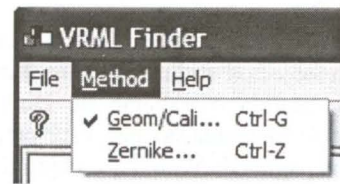


Figura 4: Barra de Menus

3.3 Barra de Menus

A barra de menus (figura 4) tem a função única de permitir a escolha de qual o método de descrição geométrica a utilizar (CALI ou Momentos de Zernike).

4. EXTRACÇÃO DE VISTAS

Durante a extracção de vistas, são criadas dez vistas 2D, diferentes, de cada objecto, de modo a que as imagens resultantes tenham tamanho 256x256, e haja um bom contraste entre objecto e o fundo. Este efeito é

facilmente criado, eliminando qualquer tipo de iluminação, forçando o objecto a ser negro na imagem criada a partir de uma vista, sendo apenas necessário configurar o fundo da imagem para branco.

4.1 As vistas de cada objecto

Para a extracção das vistas de cada objecto, é primeiro necessário deslocar e redimensionar o objecto, de modo a que este fique totalmente dentro do campo de visão da câmara.

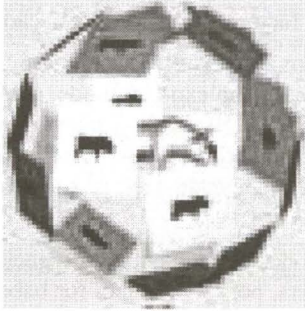


Figura 5: Imagens obtidas em várias posições da câmara [Chen03]

De seguida posiciona-se a câmara de modo a tirar várias *snapshots*. Este processo foi baseado no trabalho descrito em [Chen03] (ver figura 5). A câmara será posicionada nos vértices de um dodecaedro que envolve o objecto tridimensional (um porco). Assim obtêm-se vinte vistas diferentes do objecto (ver figura 5). Dessas vinte imagens, dez são descartadas, pois as imagens não contêm informação de cor do objecto, sendo iguais duas a duas, i.e., uma imagem é igual à imagem obtida da direcção oposta do objecto. Temos então dez vistas de cada objecto que são usadas para calcular os descritores do objecto correspondente.

5. DETECÇÃO DE CONTORNOS

Esta fase consiste na detecção dos contornos exteriores de cada uma das silhuetas presentes nas vistas de um objecto ou num esboço do utilizador. Os contornos interiores são ignorados, isto é, eventuais buracos na estrutura do objecto não são considerados.

Para que a detecção de contornos seja possível, primeiro é necessário saber quantas silhuetas contém uma vista de um objecto. Para isto é utilizada uma operação de etiquetagem sobre a imagem correspondente. No fim deste procedimento teremos o conhecimento de quantas silhuetas existem, e o conhecimento espacial de cada uma. Durante a etiquetagem da imagem, são guardados os pontos onde cada silhueta é visitada pela primeira vez pelo algoritmo. Estes pontos servirão como ponto de partida para o algoritmo de detecção de contornos, o algoritmo da *Vizinhança de Moore* [Ghuneim00].

Este procedimento é utilizado não só nas vistas dos objectos guardados na base de dados mas é também o primeiro passo do processamento do esboço desenhado pelo utilizador.

5.1 Algoritmo da Vizinhança de Moore

O algoritmo da Vizinhança de Moore, é um algoritmo simples que consiste em, considerando um pixel que se sabe pertencer ao contorno de um objecto, fazer uma pesquisa aos pixels vizinhos. Esta pesquisa é realizada a partir do pixel de onde se chegou ao pixel considerado e no sentido dos ponteiros do relógio.

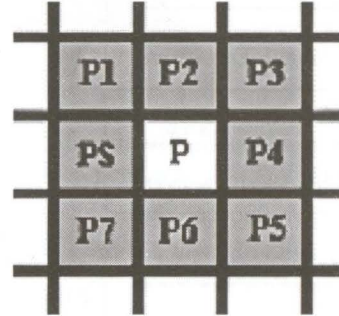


Figura 6: Vizinhança de Moore [Ghuneim00]

Seguindo a representação e notação da figura 6, PS é o pixel que foi pesquisado antes de se chegar a P, pixel que constitui o contorno da imagem, portanto sabe-se que PS não faz parte do contorno ou que já foi processado. De seguida são pesquisados os pixels P1 a P7, parando se algum for detectado como pertencendo à imagem (contorno). Para o novo ponto encontrado repete-se este processo, tendo em conta o pixel pesquisado anteriormente. Por exemplo, se o ponto encontrado como constituinte do contorno for P2, este será o próximo P e P1 será o próximo PS.

6. CRIAÇÃO DE DESCRITORES

Depois da detecção de contornos de uma imagem, calculam-se os descritores que identificam essa imagem. Como já foi referido atrás, foram utilizados dois métodos para descrever o contorno dos objectos, a biblioteca CALI e os Momentos de Zernike.

6.1 Biblioteca CALI

Esta biblioteca recebe um conjunto de contornos e devolve um outro conjunto de valores que representam esses contornos, os chamados descritores. A biblioteca CALI utiliza onze características geométricas que descrevem formas, independentemente da rotação, translação, escala ou tipo de traço:

Características	Observação
$\frac{\text{Perímetro}_{\text{PolígonoConvexo}}^2}{\text{Área}_{\text{PolígonoConvexo}}}$	Esta característica mede a estreiteza de um objecto, indicada para identificar objectos circulares, visto que formas circulares têm menor perímetro para uma dada área.

$\frac{Altura_{\text{RectânguloEnvolvente}}}{Largura_{\text{RectânguloEnvolvente}}}$	A relação de aspecto é uma boa forma de identificar linhas e outras formas estreitas.
$\frac{Área_{\text{PolígonoConvexo}}}{Área_{\text{RectânguloEnvolvente}}}$	Este conjunto de características é útil para extrair informação rectangular das formas.
$\frac{Perímetro_{\text{PolígonoConvexo}}}{Perímetro_{\text{RectânguloEnvolvente}}}$	
$\frac{Área_{\text{MaiorQuadrilátero}}}{Área_{\text{RectânguloEnvolvente}}}$	
$\frac{Área_{\text{MaiorQuadrilátero}}}{Área_{\text{PolígonoConvexo}}}$	
$\frac{Perímetro_{\text{MaiorQuadrilátero}}}{Perímetro_{\text{PolígonoConvexo}}}$	
$\frac{Área_{\text{MaiorTriângulo}}}{Área_{\text{PolígonoConvexo}}}$	
$\frac{Perímetro_{\text{MaiorTriângulo}}}{Perímetro_{\text{PolígonoConvexo}}}$	
$\frac{Área_{\text{MaiorTriângulo}}}{Área_{\text{MaiorQuadrilátero}}}$	
$\frac{ComprimentoTotal_{\text{Forma}}}{Perímetro_{\text{PolígonoConvexo}}}$	Medida da complexidade de uma forma, i.e., medida do comprimento de um contorno dentro de uma área limite (<i>convex hull</i>).

6.2 Momentos de Zernike

Momentos de Zernike são uma classe de momentos ortogonais, cuja eficiência ao descrever imagens tem sido aproveitada para aplicações de visão assistida por

computador. Uma grande vantagem desta aproximação é a invariância dos descritores calculados, em função da rotação de uma forma. Apesar disto, os descritores dependem da translação e da escala da forma, pelo que é necessário um pré-processamento dos pontos que se passa ao algoritmo, para que todas as formas processadas sejam centradas na origem e tenham um mesmo tamanho de imagem (neste caso são criadas formas num tamanho 100x100). Através desta normalização chega-se a uma situação em que os descritores são também independentes destes dois factores (escala e translação).

Polinómios de Zernike são um conjunto de polinómios ortogonais complexos, definidos para o interior de um círculo unitário [Hse03a]. A partir destes polinómios são criados os descritores que vão ser utilizados para descrever a forma de cada objecto ou esboço, permitindo identificar cada um deles.

A ordem destes momentos influencia os resultados de uma pesquisa, pois se a ordem for baixa, então os objectos são pobremente descritos, mas se a ordem for demasiado alta, o ruído que possa haver no objecto, vai também prejudicar a busca, e sendo assim, nesta abordagem foram utilizados Momentos de Zernike de oitava ordem. Isto significa que serão gerados vinte e três números de virgula flutuante que são os descritores a utilizar. Estes valores são a magnitude dos números complexos criados pelos polinómios que compõem este método e são normalizados por uma questão de compatibilidade com a estrutura de indexação, ficando estes valores entre 0 e 1.

6.3 Extracção de resultados

Como foi dito na introdução, esta aproximação é uma redução do problema 3D ao 2D, pois o que é comparado com o esboço criado pelo utilizador são as vistas dos objectos (imagens bidimensionais) e não o objecto em si ou informação baseada nas suas coordenadas tridimensionais.

A decisão sobre quais objectos apresentam maiores semelhanças com um dado esboço, é realizada a partir dos descritores do esboço e das diferentes vistas de cada objecto.

Na base de dados estão guardados os 10 vectores de descritores de cada objecto (1 vector por cada vista que foi extraída). Assim, a decisão final sobre se um objecto vai estar entre os resultados, baseia-se na proximidade entre o vector de descritores do esboço e o vector de descritores de uma das vistas. Para este efeito foi utilizado o algoritmo de *K Nearest Neighbours* (KNN), para obter os K elementos na base de dados mais próximos do elemento da interrogação. Neste projecto foi utilizado um K igual a 40, pois, entre os resultados estarão, por vezes, várias referências ao mesmo objecto (cada objecto tem 10 vectores de descritores que o referenciam), pelo que um K de 40 permite uma boa quantidade de resultados diferentes.

7. RESULTADOS E CONCLUSÕES

Para comparar os métodos de reconhecimento geométrico, foi utilizado um processo descrito no trabalho de Safar [Safar00]. Nesta experiência usou-se uma base de dados com 100 contornos de diferentes peixes, de onde foram escolhidos 5 como interrogações. De seguida, o utilizador, baseado na sua própria percepção, escolheu as 10 formas mais parecidas com as 5 escolhidas anteriormente.

Esta experiência foi realizada recorrendo a uma estrutura de indexação, onde são inseridos os descritores das 100 formas. Para cada uma das 5 interrogações calcularam-se os seus descritores e utilizou-se KNN para obter as 100 formas (K=100) por ordem decrescente de semelhança em relação a cada uma das interrogações, segundo cada um dos algoritmos de reconhecimento geométrico.

Utilizando esta metodologia comparou-se os dois métodos e obtivemos os seguintes resultados:

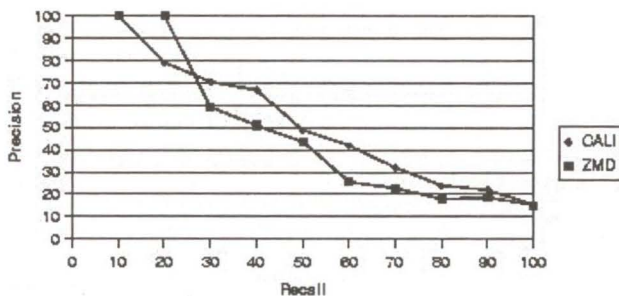


Gráfico 1: Resultados obtidos com CALI e Momentos de Zernike

O gráfico 1 representa as medidas de *Precisão/Recall* para os dois métodos, e cuja definição se apresenta a seguir:

- *Recall* - percentagem de formas semelhantes que a pesquisa retornou, em relação ao número total de formas semelhantes existentes na base de dados;
- *Precision* - percentagem de formas semelhantes que resultaram da pesquisa, em relação ao número total de formas que se obteve nesses resultados.

O CALI demonstrou um melhor comportamento em relação aos Momentos de Zernike de oitava ordem, tendo em conta a utilização do algoritmo KNN para pesquisar a estrutura de indexação (a base de dados onde são guardados os descritores dos objectos).

Por fim vamos mostrar os resultados obtidos pelo VRML Finder usando os dois métodos com uma única interrogação.

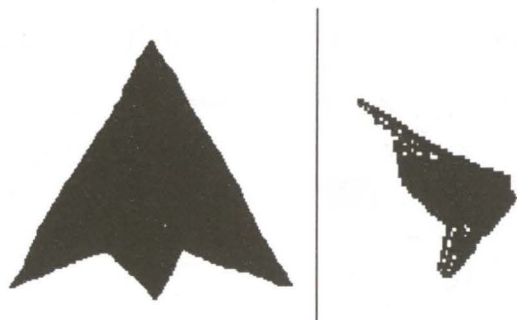


Figura 7: Interrogação utilizada e imagem do objecto a procurar (F-117)

Note-se que a imagem do objecto na região de resultados não corresponde à vista do objecto que foi referenciada como a mais semelhante, quando comparada com o esboço, mas sim à disposição do objecto no respectivo ficheiro VRML, ou seja, existe na base de dados uma vista igual à que foi esboçada, mas não é a que está nos resultados.

Com este esboço, os resultados foram os seguintes:



Figura 8: Resultados obtidos utilizando CALI

Os resultados obtidos utilizando a biblioteca CALI incluem, além do F-117 pretendido, por ser o mais semelhante, outros três aviões entre os resultados. Estes poderão ter sido seleccionados, por exemplo, pela semelhança de uma das vistas superiores do objecto, com o esboço: uma forma estreita numa ponta; alargada no meio, e de novo mais estreita na outra extremidade.

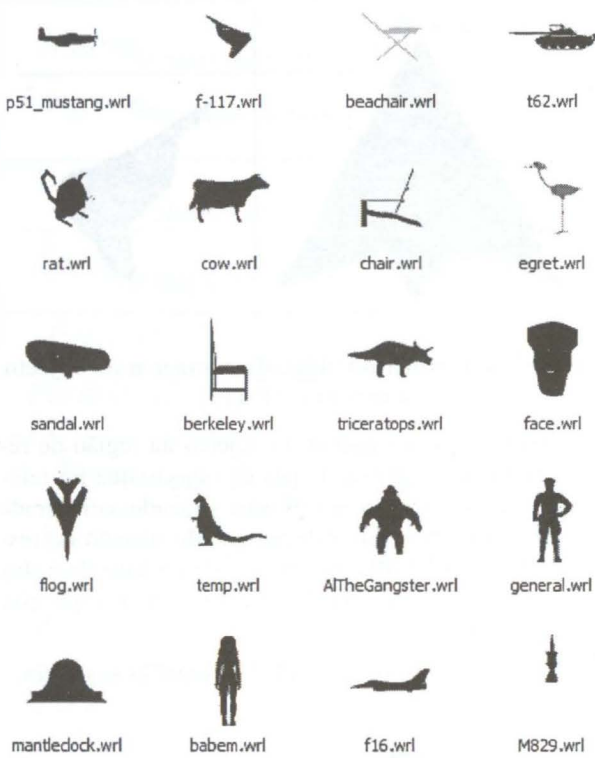


Figura 9: Resultados obtidos utilizando Momentos de Zernike

Usando os Momentos de Zernike, os resultados incluem mais objectos considerados como semelhantes. O F-117 não se encontra em primeiro lugar, mas sim em segundo, o que continua a ser um bom resultado. Os três aviões são também retornados, mas agora um deles foi reconhecido como o mais parecido com o esboço.

Entre os resultados, estão objectos que em nada se parecem com o esboço. Isto deve-se, possivelmente, ao facto de:

1. Embora objectos semelhantes tenham descritores semelhantes, há objectos diferentes, que podem ter descritores semelhantes;
2. Um esboço não é exactamente igual a uma dada vista de um objecto, e por consequência, pode ter características presentes em objectos inesperados.

Existe ainda espaço para progressão no VRML Finder. Poder-se-ia, por exemplo, aplicar a abordagem usada em [Elad02] para refinar as pesquisas, eliminando objectos indesejados ao longo de várias. Outra opção podia ser combinar de mais que um método de descrição de cada vez, avaliando os resultados em cada situação, aplicando no final uma heurística para a classificação final dos resultados.

8. REFERÊNCIAS

- [Chen03] Chen, Ding-Yun and Tian, Xiao-Pei and Shen, Yu-Te and Ouhyoung, Ming. On Visual Similarity Based 3D Model Retrieval. *Computer Graphics Forum* 22(3), 223-232, 2003.
- [Hse03a] Hse, Heloise and Newton, A. Richard. Sketched Symbol Recognition using Zernike Moments, Technical Memorandum UCB/ERL M03/49. Electronics Research Lab, Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, 2003.
<http://www-cad.eecs.berkeley.edu/~hwawen/research/papers/techReport.pdf>
- [Hse03b] Hse, Heloise. HHReco. 2003.
<http://www-cad.eecs.berkeley.edu/~hwawen/research/hhresco/>
- [Safar00] Safar, M., Shahabi, C., and Sun, X. Image retrieval by shape: A comparative study. *Proceedings of International Conference on Multimedia and Exposition (ICME)*, 2000.
- [Fonseca02a] Fonseca, Manuel J. da and Pimentel, César F. and Jorge, Joaquim A. Experimental Evaluation of a Trainable Scribble Recognizer for Calligraphic Interfaces. In D. Blostein and Y-B Kwon (Editors), *Graphics Recognition - Algorithms and Applications*, Lecture Notes in Computer Science, Vol. 2390, Springer Verlag, 2002.
- [Fonseca02b] Fonseca, Manuel J. and Pimentel, César and Jorge, Joaquim A. CALI: An Online Scribble Recognizer for Calligraphic Interfaces. *Proceedings of the 2002 AAAI Spring Symposium - Sketch Understanding*, pages 51-58, Palo Alto, USA, Mar 2002.
- [Fonseca03a] Fonseca, Manuel J. e Jorge, Joaquim A. Towards Content-Based Retrieval of Technical Drawings through High-Dimensional Indexing. *Computers and Graphics*, 27(1), pages 61-69, Jan 2003.
- [Fonseca03b] Fonseca, Manuel J. e Ferreira, Alfredo. Sketch Based Retrieval, 2003.
http://immi.inesc.pt/project_page.php?project_id=3
- [Ghuneim00] Ghuneim, Abeer George. Contour Tracing, 2000.
http://www.imageprocessingbook.com/DIP2E/dip2e_downloads/tutorials/contour_tracing_Abeer_George_Ghuneim/index.html
- [Elad02] Elad, Michael and Tal, Ayellet and Ar, Sigal. Content based retrieval of VRML objects: an iterative and interactive approach. *Proceedings of the sixth Eurographics workshop on Multimedia 2001*, 107-118. 2003.