

Segmentação de Imagens através de Rede Neuronal por Satisfação de Restrições em Ambiente Paralelo

MARCOS CARNEIRO DE ANDRADE ^{1,2}
FABIANO CRUZ PEIXOTO ¹
ARNALDO DE ALBUQUERQUE ARAÚJO ¹

¹ DCC-Departamento de Ciência da Computação
UFMG-Universidade Federal de Minas Gerais
Caixa Postal 702
30161-970 Belo Horizonte, MG, Brasil
andrade@dcc.ufmg.br, fabinho@dcc.ufmg.br, arnaldo@dcc.ufmg.br

² CDTN-Centro de Desenvolvimento da Tecnologia Nuclear
CNEN-Comissão Nacional de Energia Nuclear
Caixa Postal 1941
30161-970 Belo Horizonte, MG, Brasil

Abstract. The constraint satisfaction neural network - CSNN, proposed by Chen is here implemented in a massively parallel SIMD machine. The CSNN can be viewed as a set of interconnected neurons, whose topology and connections are used to represent constraints in a Constraint Satisfaction Problem - CSP. The neural network iterates until it converges to a consistent state. In this state all constraints are satisfied and the solution outlines the segmented areas. This region segmentation technique has been applied to images and is very promising.

1. Introdução

A segmentação de imagens é o processo de divisão da imagem em segmentos com atributos uniformes, interpretados como um conjunto de pixels que satisfazem a certas condições de adjacência e similaridade. A segmentação, cujo propósito é identificar entidades semanticamente distintas, é uma etapa importante para as fases subsequentes de interpretação e reconhecimento de imagens.

A maioria dos algoritmos convencionais de segmentação implementados em máquinas seriais, determinam relações de adjacência entre os pixels a partir de uma convolução da imagem com uma máscara. Neste estudo, o problema da segmentação de imagens foi implementado em uma máquina paralela sem fazer uso de máscaras de convolução. A segmentação é, aqui, modelada como um problema de satisfação de restrições (PSR) resolvível através de uma rede neuronal sujeita a certas restrições espaciais (*constraint satisfaction neural network* ou, abreviadamente, CSNN) [Chen]. O problema de satisfação de restrições é, neste caso, interpretado como o processo de atribuição de rótulos aos pixels da imagem. Como resultado, os pixels pertencentes a cada

região homogênea da imagem, segundo algum critério de similaridade previamente definido, recebem rótulos iguais por região.

O algoritmo encontra uma solução sub-ótima em tempo limitado, é ideal para máquinas paralelas e viabiliza a tarefa de segmentação em tempo real [Huang].

O procedimento foi aplicado a imagens monocromáticas com 256 níveis de cinza, mas pode ser facilmente estendido a imagens coloridas.

Implementou-se o algoritmo na máquina paralela DTC Zephir [Jackson], do tipo SIMD, disponível no Departamento de Ciência da Computação (DCC) da Universidade Federal de Minas Gerais (UFMG). Partiu-se de uma versão serial, escrita em C, equivalente a implementação realizada por Chen [Chen]. A partir do algoritmo serial, desenvolvida com o propósito de facilitar a depuração e a verificação do algoritmo, desenvolveu-se a versão paralela em multiC (versão paralela da linguagem de programação C, para a Wavetracer).

2. Descrição do algoritmo paralelo de segmentação

Uma CSNN consiste de um conjunto de objetos, um conjunto de rótulos para cada objeto e uma coleção de

relações de restrição (pesos) que estabelecem vínculos entre os rótulos dos objetos. Cada rótulo para um objeto em particular indica uma possibilidade de interpretação daquele objeto como pertencente a uma classe distinta. A CSNN pode ser vista como uma coleção de neurônios interconectados. A Figura 1, abaixo, mostra as conexões da rede. U... representa os neurônios e W... representa os pesos entre os neurônios.

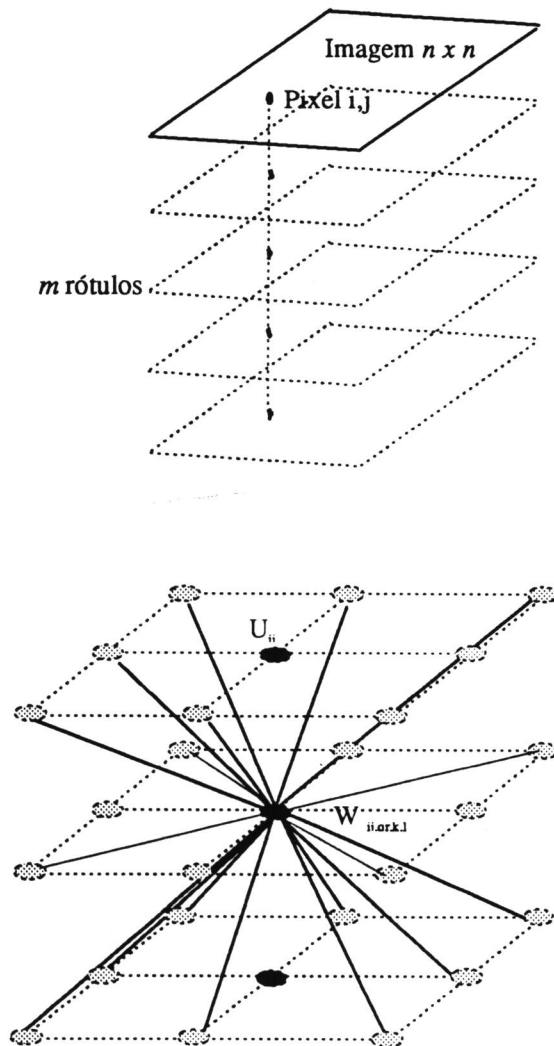


Figura 1[Chen].Conexões da rede neuronal CSNN

Assim, cada pixel de uma imagem $n \times n$ é considerado como um objeto. A cada objeto estão associados m neurônios de forma que o conjunto constitui uma matriz tridimensional contendo $n \times n \times m$ neurônios. Os neurônios interagem entre si até que a CSNN chegue a um estado estável, que representa o resultado

da segmentação, através de computações paralelas e iterativas.

Inicialmente devem ser estabelecidas as condições de partida, que representam as probabilidades de atribuição de um determinado rótulo a cada pixel. Essa atribuição é feita a partir da distribuição dos níveis de cinza da imagem. O seguinte procedimento foi utilizado:

- Determinam-se os níveis de cinza máximo (max) e mínimo (min) da imagem.
- Obtém-se a faixa correspondente a cada intervalo dos níveis de cinza, considerados os m rótulos.
- Determinam-se as probabilidades iniciais dos neurônios associados a um pixel de acordo com o nível de cinza daquele pixel. Uma vez que o nível de cinza de um dado pixel cai no intervalo que corresponde ao rótulo L , atribui-se uma probabilidade maior ao neurônio que representa o rótulo L e probabilidades menores aos outros neurônios, segundo alguma distribuição.
- Os pesos são interpretados como restrições e adota-se uma heurística para sua determinação. Um neurônio excita outros neurônios que representam rótulos similares e inibem aqueles que representam rótulos de intensidades diferentes. Os pesos são atribuídos através da fórmula:

$$W_{ij,qr,k,l} = \frac{1}{p} \left(1 - \frac{2|k-l|}{m} \right)$$

onde $W_{ij,qr,k,l}$ representa o peso entre o neurônio do k -ésimo rótulo do pixel (i,j) e o neurônio do l -ésimo rótulo do pixel (q,r) , (q,r) são as coordenadas de um neurônio pertencente a vizinhança centrada no pixel (i,j) , p é tamanho da vizinhança, m é o número de rótulos. Como observado, os pesos atribuídos aos neurônios são simétricos [Hopfield]. O parâmetro de convergência é definido de tal forma que a rede itera até que a diferença entre a saída no instante t e no instante $t+1$, seja menor que um valor arbitrariamente pequeno.

3. A máquina DTC Zephyr da WaveTracer

A Zephyr da WaveTracer [Zephyr] é uma máquina construída com o objetivo de dar suporte à simulação de fenômenos físicos e a aplicações que requeiram modelagens tridimensionais. A Zephyr é uma máquina paralela do tipo SIMD em malha, cujos elementos de processamento são fracamente acoplados. Esta malha pode assumir configurações uni, bi ou tridimensionais, controladas por software, de acordo com a especificação do programador.

As estruturas de dados manipuladas são matrizes de quaisquer dimensões, sendo criado um processo para gerenciar cada uma das posições desta matriz. Se o número de processos excede o número de processadores da malha, um mapeamento é feito de modo transparente, aproveitando a estrutura de interconexão da máquina. A máquina é composta de quatro módulos principais [Jackson].

- **Malha de processadores**

Cada elemento de processamento é um processador de 1 bit com 2Kbits de memória local e 203Kbits de memória estendida. Na configuração bidimensional, consiste de uma malha de dimensões 128 x 64.

- **Processador de Controle**

Responsável pela comunicação com a estação de trabalho, através de interface SCSI, e com a malha, através de Edge FIFO. Controla a execução das macroinstruções e gerencia os dados compartilhados pelos processadores.

- **Serializador**

Responsável pela tradução das microinstruções em nanoinstruções e pela distribuições dessas para a malha de processadores. Há uma implementação em pipeline que aumenta ainda mais o paralelismo da máquina.

- **Edge FIFO**

Interface bidirecional de comunicação entre o processador de controle e a malha de processadores, permite a transmissão simultânea de dados para 64 processadores (coluna da matriz), aumentando a rapidez na comunicação de dados.

Se qualquer das dimensões do espaço de processamento exceder a dimensão correspondente na malha, é necessário um mecanismo de mapeamento, associando mais de um processo a cada processador. A memória física disponível é dividida em blocos de mesmo tamanho, de acordo com o número de processos. O processador, ao mudar de um bloco para outro, altera seu contexto, executando um processo de cada vez. Dadas as características de sincronização de uma máquina SIMD, cada processador tem que gerenciar o mesmo número de processos.

As facilidades que podemos encontrar na máquina Zephyr são: um compilador, multiC [Wavetracer 91], e um software de monitoração, o dtcmonitor. Esse último permite verificar o número de bits que um processo está utilizando, quantos processos estão associados a cada processador e as dimensões do espaço de processamento. O multiC é um super-conjunto do C padrão. A linguagem permite a utilização de todas as

facilidades de programação do C, além de fornecer extensões para programação SIMD.

Fica a cargo programador estabelecer as dimensões do espaço de processamento e o algoritmo em alto nível, baseado nas seguintes extensões fornecidas:

- O especificador multi que declara um valor independente de variável para cada processo. Por exemplo, a declaração multi int a associa uma variável inteira a para cada processo.
- O operador de comunicação entre processos, definindo a distância em cada uma das dimensões (número de passos de comunicação). O formato do operador é [dx, dy, dz]
- Operadores de redução, que retornam um valor uni, variável única global a todos os processos, depois de uma operação sobre todos os valores de uma variável multi.
- Funções para determinação da localização lógica do processo no espaço de processamento
- Função de inicialização do processamento em paralelo, multi_perform, que determina as dimensões do espaço de processamento e executa uma rotina na máquina Zephyr.

A execução de um programa na máquina Zephyr é totalmente transparente ao programador. A máquina pode ser vista como um coprocessador de uma estação de trabalho e as funções executadas por ela como extensões ao C padrão. Além disso, o programador não precisa se preocupar com o mapeamento de processos por processador, que fica completamente a cargo do processador de controle.

4. Resultados experimentais

Imagens típicas, provenientes de diferentes áreas de aplicação, foram escolhidas para exemplificar os resultados obtidos com a CSNN. A Figura 2 mostra uma imagem de óxido de urânio de uso freqüente em ceramografia. A segmentação obtida com 3 níveis produziu ótimos resultados, eliminando as irregularidades no interior de cada grão, o que facilita a análise automática de imagens ceramográficas. A Figura 3 é uma imagem de cromossomo, também segmentada com 3 planos de neurônios. A Figura 4 exhibe um fragmento de tecido renal característico de análises histopatológicas. Em (b), (c) e (d) são mostrados os resultados obtidos com 3, 4 e 5 planos neuronais respectivamente. Nesta imagem procura-se obter a área total do glomérulo e a área da parte clara, que corresponde à luz do espaço entre a cápsula e o tufo de capilares, na parte central da imagem. A Figura 5 ilustra uma imagem do nervo ótico. Em (a) aparece a imagem original bastante ampliada, Em (b) e (c) estão

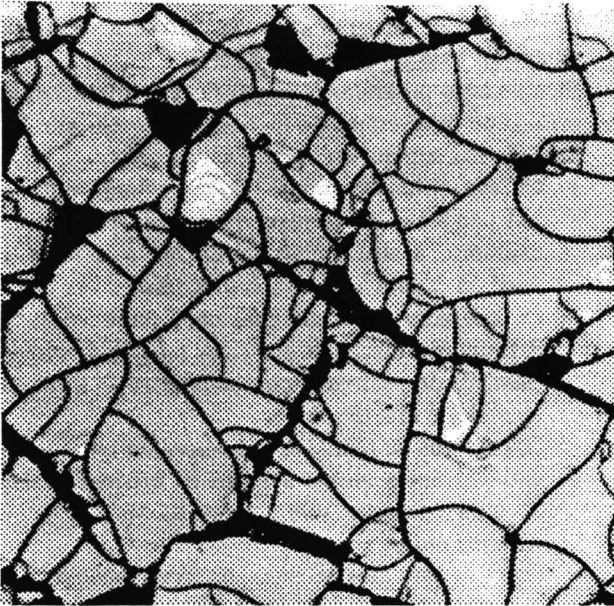


Figura 2a. Imagem original de óxido de urânio



Figura 3a. Imagem original de cromossomos

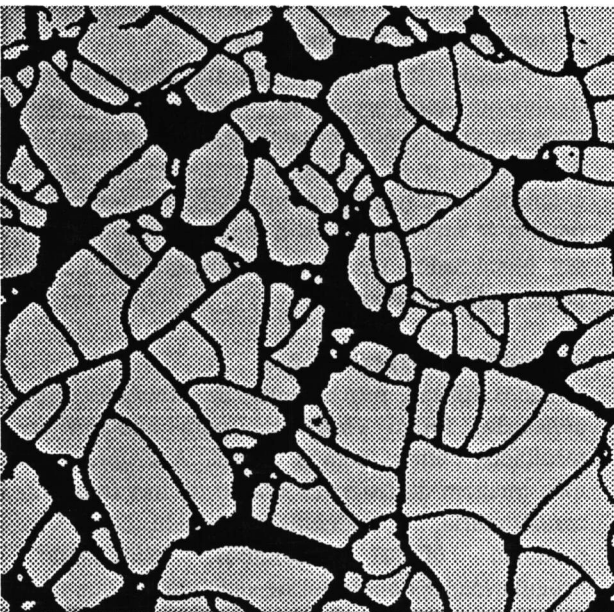


Figura 2b. Imagem segmentada de óxido de urânio



Figura 3b. Imagem segmentada de cromossomos

os resultados obtidos com 3 e 5 planos de neurônios respectivamente. Nesta imagem procura-se determinar o diâmetro da região circular que aparece como a parte central mais clara. Para todas as imagens o número de iterações foi determinado de forma experimental. Implementou-se o algoritmo de forma a permitir que tanto o número de rótulos como o número de iterações, fossem deixados como parâmetros de entrada. Valores adequados para estes parâmetros foram determinados experimentalmente para cada classe de imagem. O número de rótulos utilizados para as imagens aqui

apresentadas variou de 3 a 5, enquanto o número de iterações ficou entre 100 e 300, suficiente para que a rede convergisse ou apresentasse resultados visualmente satisfatórios. Empregou-se uma vizinhança fixa de tamanho 3×3 e probabilidades iniciais também fixas (0.4 no nível mais ativado, 0.2 nos níveis vizinhos imediatos e 0.1 nos vizinhos com distância 2 em relação ao plano mais ativado). Os pesos entre os neurônios foram determinados em função das distâncias relativas entre um neurônio e seus vizinhos para cada um dos diversos planos



Figura 4a. Imagem original de um glomérulo



Figura 4c. Imagem segmentada com 4 níveis

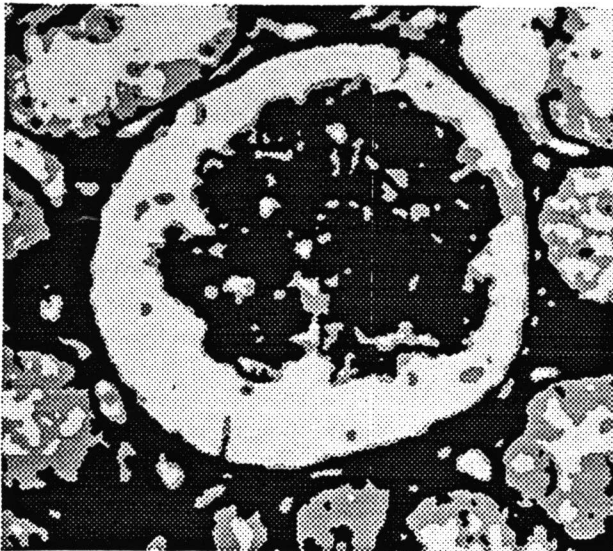


Figura 4b. Imagem segmentada com tres níveis



Figura 4d. Imagem segmentada com 5 níveis

neuronal. Tanto os pesos quanto as probabilidades foram calculados empregando-se números em ponto flutuante.

5. Conclusões

O algoritmo paralelo de segmentação sugerido por Chen foi implementado na máquina paralela DTC, do tipo SIMD, e apresentou resultados satisfatórios, comparáveis aos obtidos com técnicas convencionais de segmentação. Este algoritmo produz regiões bem delineadas, homogêneas e com pouco ruído. A redução do ruído deve-se a competição entre cada neurônio e sua vizinhança que produz, como resultado colateral, uma suavização da imagem.

O número de neurônios por pixel, o tamanho da vizinhança, as probabilidades iniciais atribuídas a cada rótulo e os pesos estabelecidos entre os neurônios são parâmetros ajustáveis. A escolha adequada dos parâmetros da rede, para cada classe de imagem, é importante para que se otimize os resultados obtidos. A escolha de um número menor de rótulos, por exemplo, permite que se obtenha regiões maiores e que pequenos detalhes desapareçam. Teóricamente, a complexidade de tempo do algoritmo paralelo é independente do tamanho da imagem, do número de rótulos e do tamanho da vizinhança. Entretanto, na prática, o desempenho fica dependente da relação entre o número

de pixels da imagem e do número de processadores disponíveis, da carga da rede, da utilização de memória estendida e do *overhead* adicional de controle de mapeamento. Também a determinação das estruturas de dados mais convenientes para representar as imagens, na máquina DTC, é crítica no que concerne ao desempenho final dos algoritmos paralelos implementados [Peixoto].

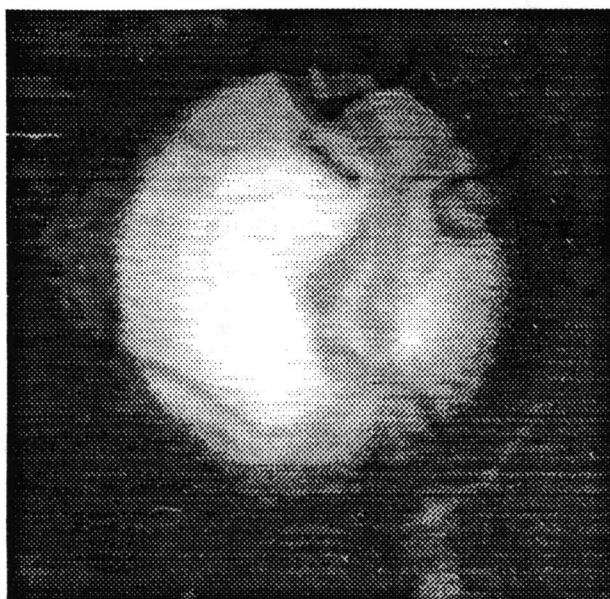


Figura 5a. Imagem original de nervo óptico

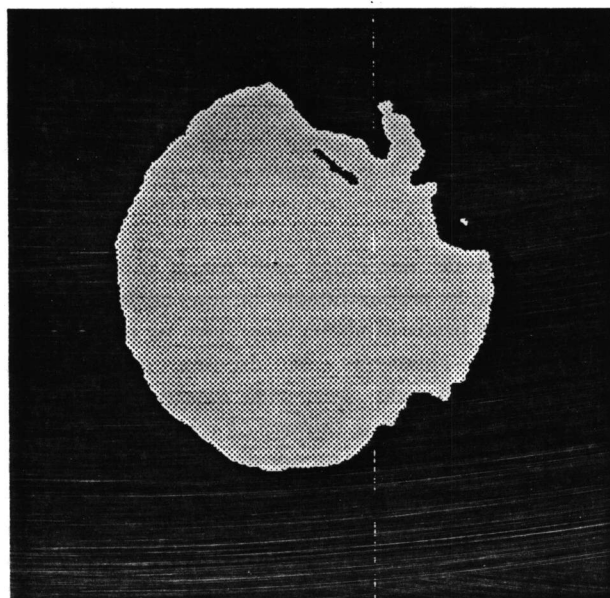


Figura 5b. Imagem segmentada com dois níveis

6. Agradecimentos

Esse projeto foi parcialmente financiado pelos órgãos FAPEMIG, CNPq e CAPES.

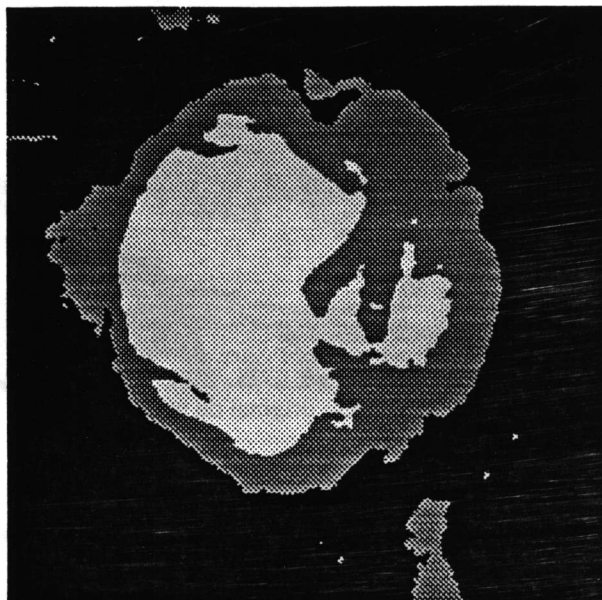


Figura 5c. Imagem segmentada com três níveis

7. Referências

- C.T.Chen et al., Medical image segmentation by a constraint satisfaction neural network. *IEEE Transactions on Nuclear Science*, v.38, n.2, p. 678-686, 1991.
- C. L. Huang, Parallel image segmentation using modified Hopfield model. *Pattern Recognition Letters*, v. 13, p. 345-353, 1992.
- J. J. Hopfield, D. W. Tank, *Computing with neural circuits: a model*. Science, v.233, p. 625-633, 1986.
- J. H. Jackson, The data transport computer: a 3-dimensional massively parallel SIMD computer. Wavetracer Inc., 1991.
- F. C. Peixoto et al., Mapeamento de vetores em malhas bidimensionais. Submetido para publicação no VI SBAC-PAD, 1994.
- Wavetracer Inc. 91, The multiC Programming Language, manual do usuário, 1991.
- Wavetracer Inc.92, The multiC Programming Language, manual de instalação e operação, 1992.