

Uso de técnicas de paralelização em GPGPU e Multirresolução para Suavização de Imagens

Carlos A. S. J. Gulo, Antonio C. Sementille
Departamento de Computação
Universidade Estadual Paulista (UNESP)
Bauru, Brasil

Email: sander@unemat.br, sementille@fc.unesp.br

Alex F. de Araujo, João Manuel R. S. Tavares
Instituto de Engenharia Mecânica e Gestão Industrial
Faculdade de Engenharia, Universidade do Porto
Porto, Portugal

Email: fa.alex@gmail.com, tavares@fe.up.pt

Abstract—Around the world, a lot of researchers have been developed unnumbered contributions to the image processing area, among them, methods have been proposed to reduce the required time to processing large images or achieve real time processing. In this work, it is combined GPGPU parallelization and multiresolution techniques to reducing the processing time required to removing multiplicative noise in images. The experiments have revealed very promising results regarding the achieved computational speedup.

Keywords—Image Processing; image smoothing; noise removal; multiresolution

I. INTRODUÇÃO

Avanços recentes na área de processamento de imagem têm obtido destaques devido à grande utilização de imagens em áreas como medicina [1], segurança [2] e inspeção industrial [3]. Entre as diferentes tarefas realizadas no processamento de imagem pode-se destacar a suavização para redução de artefatos em imagens causados por diferentes efeitos [4], [5]. Imagens com altas resoluções, isto é, com grandes dimensões, implicam o processamento de grandes volumes de dados, assim, a computação de alto desempenho pode contribuir na redução do tempo de processamento, especialmente para aplicações em tempo real. Além disso, existem diversos casos onde o tempo gasto para tal processamento deve ser o menor possível [6], como no estudo de grandes sequências de imagens 2D ou 3D, como de ultrassonografias. Adicionalmente, tais métodos costumam possuir complexidade computacional elevada, uma vez que manipulam matrizes multidimensionais, resultando, conseqüentemente, em tempo de processamento elevado. A redução do custo computacional, em processamento de imagem, pode ser realizada processando apenas as informações de interesse ou reduzindo o tamanho do conjunto de dados de entrada, através da técnica de multirresolução [7]. Uma abordagem comum para este tipo de processamento é a aplicação da transformada Wavelet [7], que pode ser utilizada para separar informações específicas da imagem em “sub-imagens”, geralmente com dimensões reduzidas. Assim, pode-se escolher e manipular apenas algumas destas “sub-imagens”, evitando processar todos os dados de entrada, minimizando o tempo de processamento.

Além das estratégias de processamento de imagem com multirresolução [7], a utilização de técnicas de programação

massivamente paralelas em *General Purpose Computing on Graphics Processing Units* (GPGPUs) [8] pode também contribuir efetivamente na redução do tempo de processamento. O modelo de paralelização da NVIDIA, denominado *Single Instruction, Multiple Thread* (SIMT) foi desenvolvido com base na técnica de decomposição de dados. Neste modelo, os dados de entrada são representados como um fluxo de dados, e este fluxo é classificado como sendo um conjunto estruturado de *arrays*, e suas instruções são definidas como *kernels*. Um *kernel* por sua vez, realiza operações paralelamente ao longo do fluxo de entrada, sendo dividido em blocos independentes, garantindo escalabilidade transparente e permitindo sua execução em qualquer número de núcleos de processamento [9].

Contribuições: Neste trabalho são combinadas as técnicas de processamento em multirresolução e programação paralela em GPGPU para reduzir o tempo de processamento do algoritmo de redução de ruído multiplicativo baseado num modelo variacional proposto por Jin e Yang [4]. A abordagem proposta mostra-se promissora, apresentando uma redução considerável no tempo de processamento do algoritmo considerado.

Este artigo está organizado da seguinte forma: na seção seguinte é apresentada uma descrição sobre a abordagem proposta, seguido pelas seções de experimentos, resultados e discussão. Por fim, na última seção, apresentam-se as considerações finais e a indicação de trabalhos futuros.

II. METODOLOGIA PROPOSTA

Para obter uma redução no tempo de processamento do algoritmo de suavização baseado no modelo variacional, proposto por Jin e Yang [4], adotou-se a combinação das técnicas de processamento em multirresolução e paralelização em GPGPU. Desta forma a suavização da imagem original foi dividida em 3 etapas. Na primeira, a imagem original é transformada do domínio espacial para o domínio da frequência através da aplicação da transformada Wavelet de nível 1. Desta forma, os dados da imagem original podem ser divididos em 4 “sub-imagens”, onde cada uma possui metade da resolução da imagem de entrada e representam respectivamente, as baixas frequências, as altas frequências nas direções vertical, horizontal e diagonal. Após testes realizados, utilizando uma

máscara de 3x3, verificou-se que a maioria dos detalhes da imagem original de teste é representada no quadrante das baixas frequências. Assim, percebeu-se que descartando os demais quadrantes e processando apenas as baixas frequências, o resultado da suavização é próximo da suavização da imagem original como um todo no domínio espacial, como pode ser observado na Fig. 1.

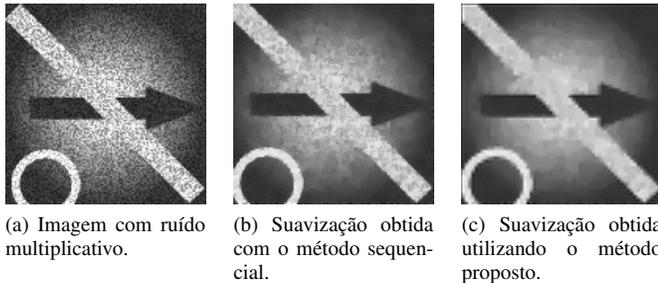


Fig. 1. Suavização de imagem (128x128 pixels).

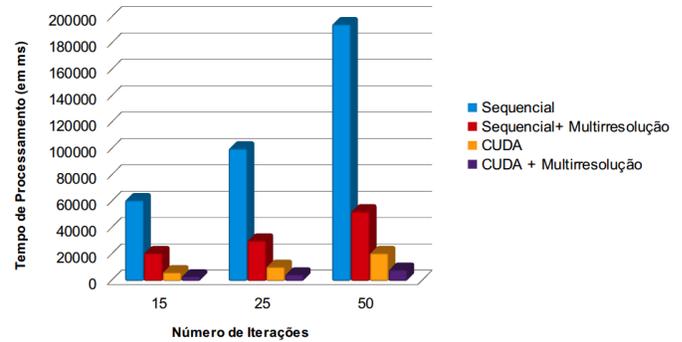
A redução de ruído é o segundo passo da metodologia desenvolvida, sendo obtida através da paralelização em GPGPU do método proposto por Jin e Yang [4]. Mais detalhes sobre a paralelização implementada podem ser obtidos em [8]. O terceiro e último passo foi aplicar a transformada Wavelet inversa, obtendo assim a imagem resultante novamente no domínio espacial.

III. EXPERIMENTOS

Para testar a abordagem proposta, foram utilizadas imagens sintéticas em níveis de cinza, criadas a partir de um editor de imagem, e afetadas com ruído multiplicativo sintético e aleatório com variância 0.05. Os experimentos foram realizados em um computador portátil Dell XPS 15.4", equipado com processador Intel i7-2630QM de 2GHz, com 8GB de memória RAM, placa gráfica GeForce GT 540M (NVIDIA) com 96 CUDA cores e 2GB de memória, sistema operacional Linux Ubuntu 11.04 - 64 bits, compilador CUDA nvcc 4.2 release, e compilador g++ versão 4.4.5. Foram comparados o desempenho entre a implementação do algoritmo de suavização sequencial, sequencial combinando abordagem multirresolução, paralelizado em GPGPU utilizando a linguagem CUDA e paralelizado combinando a abordagem multirresolução. Foram utilizadas 6 imagens com resoluções a partir de 128x128 até 4096x4096 pixels, e os parâmetros para suavização foram: 15, 25 e 50 iterações, com o valor de $\Delta t = 1.0$.

IV. RESULTADOS E DISCUSSÃO

Os experimentos demonstraram o ganho de desempenho do algoritmo de suavização que combina as técnicas de multirresolução e arquitetura paralela em GPGPU, ilustrado na Fig. 2. Os resultados podem ser considerados de grande relevância, permitindo a utilização da abordagem proposta em problemas que exigem o processamento de imagens com altas resoluções, bem como processamento das mesmas em tempo real.



(a) Imagem com 4096x4096 pixels.

Fig. 2. Comparação de desempenho dos algoritmos de suavização.

V. CONSIDERAÇÕES FINAIS

Apesar dos resultados preliminares serem promissores, alguns estudos sobre esta abordagem precisam ser realizados, tais como: verificar se a abordagem não gera perda excessiva de informação presente na imagem original, o que pode causar redução da precisão em etapas seguintes, como de segmentação; realizar testes com diferentes famílias Wavelet com o objetivo de detectar qual delas melhor se adapta; e repetir os experimentos usando imagens reais de diferentes tipos.

AGRADECIMENTOS

O primeiro autor agradece à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), por sua bolsa de Mestrado, e à UNEMAT, pelo suporte dado. O autor Alex F. Araujo agradece à Fundação para a Ciência e a Tecnologia (FCT) em Portugal, por sua bolsa de Doutorado, com referência SFRH/BD/61983/2009.

REFERÊNCIAS

- [1] Z. Ma, J. M. R. S. Tavares, T. Mascarenhas, and R. Jorge, "A review of algorithms for medical image segmentation and their applications to the female pelvic cavity," in *Computer Methods in Biomechanics and Biomedical Engineering*, vol. 13, no. 2, 2010, pp. 235–246.
- [2] S. Chang, Y. Cheng, K. Larin, Y. Mao, and S. Sherif, "Optical coherence tomography used for security and fingerprint-sensing applications," *IET - Image Processing*, vol. 2(1), pp. 48–58, 2008.
- [3] I. Kunttu and L. Lepisto, "Shape-based retrieval of industrial surface defects using angular radius fourier descriptor," *IET - Image Processing*, vol. 1(2), pp. 231–236, 2007.
- [4] Z. Jin and X. Yang, "A variational model to remove the multiplicative noise in ultrasound images," *Journal of Mathematical Imaging and Vision*, vol. 39, no. 1, pp. 62–74, 2011.
- [5] F. N. S. Medeiros, N. D. A. Mascarenhas, and L. d. F. Costa, "Speckle noise MAP filtering based on local adaptive neighborhood statistics." XII SIBGRABI, 1999, pp. 347–354.
- [6] M. Celebi, "Real-time implementation of order-statistics based directional filters," *IET - Image Processing*, vol. 3(1), pp. 1–9, 2009.
- [7] J. M., *Noise reduction by wavelet thresholding*, ser. Lecture Notes in Statistics. Springer, 2001.
- [8] C. A. S. J. Gulo, H. F. de Arruda, A. C. Sementille, A. F. de Araujo, and J. M. R. S. Tavares, "Método de suavização de imagem baseado num modelo variacional paralelizado em arquitetura CUDA," in *GPU Computing Developer Forum*. XXXII Congresso da Sociedade Brasileira de Computação, 2012.
- [9] NVIDIA, *GPU Tutorial: Build environment, Debugging/Profiling, Fermi, Optimization/CUDA 3.1 and Fermi advice*, NVIDIA, 2010.