

Comparação de Algoritmos de Suavização Espacial

EDUARDO ROBERTO ARECO
 ARNALDO DE ALBUQUERQUE ARAÚJO
 MÁRIO FERNANDO MONTENEGRO CAMPOS

DCC—Departamento de Ciência da Computação
 UFMG—Universidade Federal de Minas Gerais
 Caixa Postal 702
 30161-970 Belo Horizonte, MG, Brasil
 {areco,arnaldo,mario}@dcc.ufmg.br

Abstract. This work presents the results of the evaluation of some spatial-domain noise-smoothing techniques, focusing the edge-preserving characteristics and the power of the algorithms, when used iteratively. Some comments in evaluation noise-smoothing techniques are also included. The algorithms implementation have been executed within the Khoros environment.

1 Introdução

O principal objetivo das técnicas de realce é processar uma imagem para que o resultado seja mais apropriado que a imagem original, para uma aplicação específica. Isto significa que determinada técnica pode ser altamente indicada para um tipo de aplicação e possuir um desempenho insatisfatório com outra classe de aplicação.

As técnicas de realce de imagens têm o propósito de obter algum tipo de melhora na imagem e para isso se valem do conhecimento de algumas características do sistema visual humano, visto que as imagens realçadas para melhorar a interpretação humana geralmente também melhoram o desempenho de aplicações automatizadas de níveis mais altos da área de processamento digital de imagens. As principais tarefas classificadas como realce de imagens são as operações relativas a mudança de histograma, as quais afetam o contraste e o brilho da imagem, e as operações sobre pontos da imagem, como suavização e realce de bordas [2]. Quando uma imagem é processada apenas para interpretação visual, o observador é quem deve julgar se os métodos empregados foram ou não satisfatórios, sendo este processo muito subjetivo e dificultando o estabelecimento de um procedimento padrão para a comparação do desempenho dos algoritmos. Já no caso de imagens processadas para servirem de entrada a sistemas de visão computacional ou reconhecimento automático, o processo de avaliação é mais simples, baseando-se no resultado destes sistemas.

As técnicas de suavização têm o objetivo de tornar as regiões de uma imagem mais homogêneas, ou seja, fazer com que os todos os *pixels* representantes de alguma característica de determinado objeto

venham a ter intensidades semelhantes. Geralmente os procedimentos de suavização procuram identificar pontos que não possuam nenhuma relação com a sua vizinhança, classificando os mesmos como pontos ruidosos e eliminando-os através de algum operador de vizinhança, como média ou mediana. Seus objetivos principais são a remoção de ruído e a uniformização dos níveis de cinza dos *pixels* nas regiões presentes na imagem.

Com o maior desenvolvimento da área de processamento digital de imagens, o número de artigos descrevendo técnicas de suavização espacial também tem crescido muito, o que leva o usuário destes algoritmos a encontrar um problema na modelagem de sua aplicação: a escolha do algoritmo mais indicado para o seu caso. Somente uma análise criteriosa dos algoritmos disponíveis combinada com um bom procedimento de testes feitos com imagens representativas da classe a ser utilizada é que poderiam auxiliar o pesquisador na escolha de determinado filtro. Mas isto nem sempre é possível, devido ao grande número de algoritmos disponíveis e a dificuldade de se conseguir definir exatamente quais as características das imagens que serão utilizadas.

Este trabalho se propõe a auxiliar na escolha de um filtro de suavização espacial, sendo que para isso serão apresentados alguns dados comparativos referentes aos filtros estudados. Todos os testes e as implementações dos algoritmos foram feitos no ambiente *Khoros*, sendo que está sendo planejada a colocação dos algoritmos implementados a disposição da comunidade acadêmica através de uma *toolbox* para o sistema *Khoros*.

2 O Ambiente Khoros

Khoros [7] é um ambiente para pesquisa desenvolvido na Universidade do Novo México. O sistema *Khoros* integra múltiplos modos de interface com o usuário, geradores de código, visualização de dados, computação distribuída e processamento de informações. O resultado é um ambiente único com ferramentas para pesquisa e desenvolvimento de sistemas computacionais. O sistema foi desenvolvido para ambiente *Unix* que utilizam *X Window System*, e portanto pode ser transportado para uma ampla faixa de estações de trabalho que utilizam sistema operacional semelhante e é composto de diversos aplicativos, que podem ser classificados como ferramentas para o auxílio no desenvolvimento de sistemas e também aplicações para o usuário final. A utilização deste sistema para o desenvolvimento do trabalho se deve ao fato do mesmo ser de domínio público e de estar conseguindo uma boa aceitação tanto no meio universitário quanto em outras instituições de pesquisa.

3 Descrição dos Filtros Implementados

Os filtros selecionados para este trabalho são aqueles citados pela literatura básica da área e também vários filtros mais recentes que têm aparecido nos periódicos relacionados com processamento de imagens.

A seguir serão citados os parâmetros utilizados, quando os mesmos existirem, e o nome dado para a implementação de cada filtro: filtro da média, *vavg*, janela 3x3; mediana, *vmedian*, janela 3x3; mediana adaptativo [4], *vadaptmed*, limiares T1 e T2 iguais a 16; mediana adaptativo com pesos [5], *vawmed*, $c = 5$ e $w = 20$; filtro sigma, *vsigma*, janela 5x5, $K = 2$ e $\sigma = 20$; sigma adaptativo, *vadapsig*, janela 5x5, $K = 2$, $\alpha = 2.0$; filtro Nopel [3], *vnopel*; suavização por soma de diferenças absolutas, *vsadvs*; suavização por região mais homogênea, *vmhns*; média com os K vizinhos, *vknn*, janela 3x3, $k = 7$; suavização pelo inverso do gradiente, *vgradinv*; suavização logarítmica, *suavlog*, $M = 16$; suavização pelo modelo de facetas, *vfacet*; mediana cruzada, *vcrossmed*, janela 3x3, mediana no final.

4 Comparação dos algoritmos

Para o desenvolvimento dos testes comparativos, têm-se adotado o procedimento mais comum existente na literatura da área, que é a geração de imagens com ruído a partir de imagens já existentes, para se ter a possibilidade de fazer uma comparação entre uma imagem com ruído depois da filtragem com a imagem *ideal*, que seria a própria imagem original, ou seja, antes da aplicação do ruído. Desta forma,



Figura 1: Imagem Lena

é possível se extrair informações quantitativas dos procedimentos de testes, além da informação qualitativa, que neste caso pode ser apenas um critério subjetivo do observador ao inspecionar visualmente o resultado de determinado processo de suavização espacial. Outra categoria de procedimentos de teste pode ser definida como aquela em que as imagens filtradas servem de entrada para algum mecanismo de análise de imagens. Neste caso, não há a obrigatoriedade de se ter uma imagem original para se adicionar ruído artificialmente, pois os critérios de classificação dos algoritmos seriam baseados na resposta do mecanismo analisador de imagens. Como um exemplo, poderíamos ter uma imagem ruidosa que possuísse n artefatos de determinada forma, portanto o procedimento de comparação dos algoritmos seria apenas verificar a saída de um analisador de artefatos daquela forma. O algoritmo que conseguisse se aproximar mais do número exato de artefatos seria considerado melhor. Também poderiam ser acrescentados outros dados, como o tamanho do artefato e seu alinhamento, utilizando-se todos estes dados de forma ponderada para classificar os algoritmos. Podemos notar que, embora nem sempre seja necessário, em alguns casos pode ser utilizada a técnica da imagem original com ruído adicionado, neste caso a classificação dos algoritmos se daria pela proximidade da resposta dada pelo analisador para cada imagem ruidosa filtrada em relação à resposta dada pela imagem dita original.

Para este trabalho foram utilizados dois conjuntos de imagens. As imagens das Figuras 1 e 2 são as imagens originais. Para estas duas imagens foram geradas imagens com ruído gaussiano ($\mu = 0$, $\sigma = 20$) e ruído sal e pimenta (10 %).

4.1 Descrição dos Métodos Utilizados

No processo de comparação dos algoritmos em relação a suas características de remoção de ruído com preservação de bordas comumente se utilizam medidas como o desvio padrão em área homogênea para me-

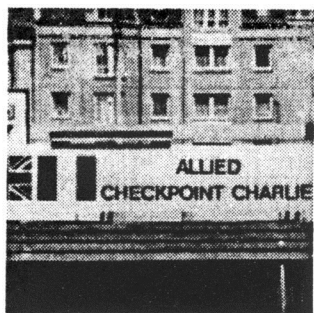


Figura 2: Imagem Checkpoint



Figura 3: Regiões de alta atividade

dir o poder de suavização e medidas de erro, como relação sinal/ruído e erro médio quadrático para verificar a distorção das bordas. Estas medidas de erro são sempre calculadas em relação a alguma outra imagem, que no caso é a imagem dita original. O desvio padrão ou outra medida de homogeneidade pode ser calculado apenas nas imagens com ruído após a filtragem, mas implica na existência de uma área na imagem original que já apresentasse altos índices de homogeneidade, para que seja possível medir apenas o poder de eliminação de elementos ruidosos, sem interferência de possíveis bordas ou texturas.

O método descrito acima funciona bem em imagens com poucos objetos, como por exemplo uma imagem de teste composta apenas de um círculo claro sobre fundo escuro, pois podem ser escolhidas áreas bem definidas para o teste de poder de suavização, bastando apenas que as mesmas não sejam próximas das bordas do círculo. Já para a interpretação dos resultados das medidas de erro, consegue-se identificar facilmente quando o erro tornou-se grande devido à distorção das bordas ou à baixa capacidade de filtragem, analisando-se estes dados em conjunto com a informação de homogeneidade.

Para imagens mais complexas, como por exemplo imagens naturais onde existem muitos objetos, um outro método foi empregado. Neste método, procura-se desenvolver uma classificação dos pontos das imagens pela atividade espacial, ou seja, pela

presença ou não de bordas ou texturas mais profundas. A princípio podem ser definidas várias classes, mas a definição de apenas duas tem se mostrado eficaz [1]. Para obter esta classificação foi utilizado um procedimento baseado nas máscaras direcionais de Prewitt [6], onde a soma da saída das máscaras foi confrontada com um limiar escolhido, para ser tomada a decisão de se classificar o ponto em alta ou baixa atividade espacial. Posteriormente, foi aplicado um filtro para nublur a imagem seguido de uma nova aplicação de limiar, desta vez gerando as máscaras binarizadas correspondentes às regiões de alta e baixa atividade espacial. Esta último filtro aplicado se deve ao fato de que é interessante ter não só os pontos exatos em que a atividade espacial é mais alta, mas também uma faixa de sua vizinhança, podendo assim serem observados mais facilmente os fenômenos de degradação de bordas, já que os mesmos afetam tanto a borda quanto os seus *pixels* mais próximos. Na Figura 3 temos a máscara utilizada para definir as regiões de alta atividade espacial para uma das imagens de teste. As máscaras para as regiões de baixa atividade espacial foram obtidas com a inversão das já obtidas.

O procedimento básico para a realização dos testes foi a filtragem das imagens com ruído e posterior cálculo de uma medida de erro, no caso o erro médio quadrático, que é definido como sendo a média das diferenças entre as duas imagens elevadas ao quadrado, para todos os pontos da imagem. Além do erro médio quadrático comum, foram efetuadas medidas do erro médio quadrático em regiões de alta e de baixa atividade espacial, onde a definição de erro médio quadrático foi estendida adicionando-se uma terceira imagem binária para funcionar como máscara de ativação da função de erro. No caso, para um ponto ser computado para o cálculo do erro, o valor de sua posição correspondente na terceira imagem deverá ser diferente de zero. Utilizando-se as máscaras que definem as regiões de alta e de baixa atividade espacial, conseguiu-se gerar estas outras duas medidas de erro.

4.2 Testes comparativos

Os testes comparativos foram realizados nos filtros citados anteriormente e constaram da aplicação iterativa dos mesmos nas imagens de teste, com a medição dos três erros já definidos. Procurou-se utilizar os parâmetros indicados pelo autor de cada algoritmo quando estes dados eram disponíveis, mas mesmo assim uma mudança neste parâmetros em alguns casos pode vir a melhorar o desempenho de alguns algoritmos, o que não foi possível testar completamente, devido ao grande número de possibilidades envolvi-

das.

Nas tabelas 1,2,3 temos o erro médio quadrático total (EMQ), o erro médio quadrático em regiões de alta atividade (EMQRA) e o erro médio quadrático em regiões de baixa atividade (EMQRB), respectivamente, para a imagem Lena com ruído gaussiano, após as três primeiras aplicações de cada algoritmo. Os erros iniciais, ou seja, antes da aplicação dos filtros, eram 398.84 o total, 397.67 nas regiões de alta atividade e 400.01 nas de baixa.

| filtro | iter. 1 | iter. 2 | iter. 3 |
|-----------|---------|---------|---------|
| vsadvs | 139.24 | 128.45 | 133.00 |
| vmhns | 161.97 | 168.95 | 183.89 |
| vavg | 114.35 | 114.57 | 133.28 |
| vmedian | 128.21 | 109.98 | 110.66 |
| vadapmed | 140.96 | 137.21 | 140.76 |
| vnopel | 283.12 | 206.13 | 168.06 |
| vfacet | 184.23 | 173.93 | 183.29 |
| vsigma | 128.21 | 95.11 | 96.34 |
| vknn | 123.16 | 102.14 | 102.08 |
| vgradinv | 227.62 | 181.28 | 160.89 |
| vlogsmoo | 166.76 | 165.06 | 177.96 |
| vadapsig | 189.48 | 141.14 | 130.36 |
| vcrossmed | 158.30 | 129.42 | 122.65 |
| vawmed | 303.33 | 266.84 | 249.96 |

Tabela 1: EMQ lena gaussiano

| filtro | iter. 1 | iter. 2 | iter. 3 |
|-----------|---------|---------|---------|
| vsadvs | 213.03 | 226.01 | 243.84 |
| vmhns | 261.58 | 304.80 | 339.64 |
| vavg | 191.53 | 216.25 | 261.27 |
| vmedian | 196.41 | 191.58 | 203.30 |
| vadapmed | 242.85 | 257.30 | 268.64 |
| vnopel | 300.50 | 241.77 | 218.85 |
| vfacet | 316.38 | 324.39 | 347.17 |
| vsigma | 182.76 | 170.33 | 181.28 |
| vknn | 159.25 | 158.78 | 171.78 |
| vgradinv | 240.86 | 202.37 | 186.67 |
| vlogsmoo | 260.36 | 299.15 | 333.68 |
| vadapsig | 232.14 | 212.25 | 219.72 |
| vcrossmed | 192.66 | 180.14 | 180.31 |
| vawmed | 347.87 | 330.36 | 321.78 |

Tabela 2: EMQRA lena gaussiano

5 Conclusões

O processo de seleção de algoritmos de suavização espacial não é uma tarefa muito simples. Este trabalho tentou mostrar algumas técnicas utilizadas para auxiliar neste processo, técnicas estas que podem ser estendidas e aplicadas a novos conjuntos de imagens, para que assim se tenha uma base maior de informações para auxiliar na escolha do melhor algoritmo para cada aplicação. Com base nos testes

| filtro | iter. 1 | iter. 2 | iter. 3 |
|-----------|---------|---------|---------|
| vsadvs | 84.63 | 56.24 | 50.96 |
| vmhns | 88.26 | 68.41 | 68.62 |
| vavg | 57.22 | 39.32 | 38.55 |
| vmedian | 77.74 | 49.58 | 42.10 |
| vadapmed | 65.54 | 48.33 | 46.11 |
| vnopel | 270.33 | 179.76 | 130.46 |
| vfacet | 86.43 | 62.57 | 62.00 |
| vsigma | 87.84 | 39.44 | 33.48 |
| vknn | 96.45 | 60.21 | 50.49 |
| vgradinv | 217.81 | 165.68 | 141.81 |
| vlogsmoo | 97.49 | 65.82 | 62.71 |
| vadapsig | 157.91 | 88.52 | 64.22 |
| vcrossmed | 132.86 | 91.87 | 79.97 |
| vawmed | 270.49 | 219.86 | 196.81 |

Tabela 3: EMQRB lena gaussiano

efetuados foi constatado que alguns algoritmos só possuem desempenho razoável em um dos tipos de ruídos utilizados e também que algoritmos com desempenho global equivalente podem conseguir dados diferentes nas regiões de alta ou de baixa atividade espacial.

Referências

- [1] Roland T. Chin and Chia-Lung Yeh. Quantitative evaluation of some edge-preserving noise-smoothing techniques. *Computer Vision, Graphics, and Image Processing*, 23:67-91, 1983.
- [2] Rafael C. Gonzalez and Paul Wintz. *Digital Image Processing*. Addison-Wesley Publishing Company, 1977.
- [3] M. Imme. A noise peak elimination filter. *CVGIP: Graphic Models and Image Processing*, 53(2):204-211, March 1991.
- [4] Ho-Ming Lin and Alan N. Wilson, Jr. Median filters with adaptive length. *IEEE Trans. on Circuits and Systems*, 35(6):675-689, 1988.
- [5] T. Loupas, W. N. McDicken, and P. L. Allan. An adaptive weighted median filter for speckle suppression in medical ultrasonic images. *IEEE Trans. on Circuits and Systems*, 36(1):129-135, 1989.
- [6] J. M. S. Prewitt. Object enhancement and extraction. In *Picture Processing and Psychopictoris*, pages 75-149. Academic Press, 1970.
- [7] John Rasure and Mark Young. An open environment for image processing and software development. In *SPIE/IS&T Symposium on Electronic Imaging*, volume 1659. SPIE, 1992.