

# Evolução de filtros ótimos com algoritmo fireworks para análise por níveis de erro em imagens comprimidas com perdas

**Daniel Cavalcanti Jeronimo**

Departamento de Engenharia Elétrica, Universidade Federal do Paraná, UFPR, C. P. 19011, Centro Politécnico,  
CEP 81531-970, Curitiba, Paraná, Brasil.  
dcavalcanti@bigfoot.com

**Yuri Cassio Campbell Borges**

Departamento de Engenharia de Telecomunicações e Controle, Universidade de São Paulo, USP, Escola  
Politécnica, CEP 05508-900, São Paulo, São Paulo, Brasil.  
yuri.campbell@gmail.com

**Leandro dos Santos Coelho**

Departamento de Engenharia Elétrica, Universidade Federal do Paraná, UFPR, C. P. 19011, Centro Politécnico,  
CEP 81531-970, Curitiba, Paraná, Brasil.  
Programa de Pós-Graduação em Engenharia de Produção e Sistemas, LAS/PPGEPS, Pontifícia Universidade  
Católica do Paraná, PUCPR, Imaculada Conceição, 1155, CEP 80215-901, Curitiba, Paraná, Brasil.  
leandro.coelho@pucpr.br

**Resumo** – A análise por níveis de erro (*error level analysis - ELA*) é um método para detecção de falsificação em imagens comprimidas com perdas. Este método é degradado se o ruído gerado pelo algoritmo de compressão for intenso, criando a necessidade de um filtro. Neste trabalho, uma metaheurística de otimização conhecida como Algoritmo *Fireworks* (AF) é usada para a evolução de filtros passa-baixa, atenuando níveis de ruído. Imagens padrões de teste foram editadas para simular falsificações. Para simular esquemas de pós-processamento com efeitos em toda a imagem, as imagens foram redimensionadas e recomprimadas, o que significativamente aumenta o ruído na análise por níveis de erro. A combinação da otimização estocástica com a análise por níveis de erro resulta em diversos filtros, variando parâmetros como método de equalização, método de filtragem e parâmetros da filtragem, procurando pela configuração de filtro que resulta nos melhores resultados, dentro das metas estipuladas. Os filtros otimizados são então utilizados para remoção de ruído e melhoramento das imagens. Resultados são apresentados mostrando que o Algoritmo *Fireworks* é bem sucedido em encontrar um filtro que adequadamente atenua ruído enquanto preserva os níveis de erro necessários.

**Palavras-chave** – metaheurísticas, *fireworks*, análise por níveis de erro, falsificação de imagens

**Abstract** – Error level analysis (ELA) is a method for detection of image forgery in lossy compressed digital images. This method might be impaired if the image noise generated by the compression scheme is too intense, creating the necessity of noise filtering. In this paper an optimization metaheuristic known as Fireworks Algorithm (FA) is used to enhance low-pass filters, smoothing error level noises. Standard test images have been edited as to simulate image tampering. In order to simulate post-processing full image tampering, the tampered images are resized and recompressed, which significantly increases noise in ELA. Using ELA, these lossy compressed images are once again, intentionally, recompressed at a known error rate and the absolute differences, known as error levels, computed. Fireworks Algorithm is then used to setup different filters by varying parameters such as equalization method, filter method, filter parameters, and sharpness, searching for the filter setup which results in optimal performance. The optimized filters are then used for noise removal and image enhancement. Results are presented showing Fireworks Algorithm is successful in finding a filter which adequately attenuates noise while preserving the necessary error levels.

**Keywords** – metaheuristics, fireworks, error level analysis, image tampering, image forgery, image forensics

## 1 INTRODUÇÃO

As novas técnicas de edição de imagens, cada vez mais disponíveis através de *softwares* poderosos de edição e *hardwares* de alto desempenho, disponíveis até mesmo para os não-profissionais, tem tornado cada vez mais difícil a distinção visual de imagens falsificadas. As falsificações são por definição imagens cujo conteúdo original foi alterado, removido ou combinado com outras imagens, texturas sintéticas ou gráficos renderizados por computador [1]. As técnicas para a construção destas falsificações são diversas. Em [1] algumas destas são citadas, como cortar e colar, conhecida como clonagem [2], *matting* para ambientar mudanças [3], cortes em grafos para descoberta de limites ótimos de composição [4], síntese de texturas [4] e abordagens variadas para síntese de conteúdo [5, 6].

Estudos sobre detecção de fraudes em imagens são divididos em abordagens ativas ou passivas. As abordagens ativas dependem de pré-processamento, como o uso de marcas d'água ou assinaturas digitais [7]. Por outro lado, as abordagens passivas não

fazem uso de nenhum conhecimento *a priori* sobre a imagem, sendo esta a principal direção de interesse nas pesquisas forenses em imagens [8]. Entretanto, mesmo considerando o estado da arte atual, não há nenhuma solução completa para automaticamente determinar falsificações [1].

Análise por níveis de erro (*Error Level Analysis - ELA*) é um método passivo-cego para análise forense de imagens, criado por Krawetz [9]. Há uma falha na literatura recente [10, 11] em reconhecer e citar apropriadamente o autor original deste método.

Uma imagem original, comprimida por algoritmos com perdas, possui níveis únicos de qualidade, uma propriedade originada da sua aquisição. Quando ocorrem alterações nesta imagem, seja através de clonagem, *splicing* ou *matting*, o conteúdo original é combinado com conteúdo de outras fontes, exibindo variações nos níveis de qualidade ao longo da imagem.

A *ELA* explora essas diferenças nos níveis de qualidade ao computar as diferenças entre níveis de pixels em uma imagem e entre a mesma imagem recomprimida. Esta diferença é associada com o conteúdo do pixel analisado, se a diferença for grande, a taxa de compressão foi alta e o pixel é provavelmente de outra fonte, por outro lado, se a diferença for baixa, o pixel já estava em seu mínimo local de compressão e é provavelmente original [9]. Pós-processamento com efeitos em toda a imagem, como operações de redimensionamento, aumentam significativamente o ruído nos níveis de erro, degradando a eficiência da *ELA*. Neste contexto, remoção de ruído e melhoramento da imagem são necessários.

Em anos recentes, a computação evolucionária tem sido um campo extenso de pesquisa devido a sua variedade de aplicações e métodos. A inteligência de enxames é, em partes, responsável por esta crescente atividade neste campo de pesquisa desde os anos 90, principalmente devido a sua universalidade quando se trata de aplicações [12].

Seguindo o mesmo princípio básico, alguns algoritmos tem sido bem sucedidos em suas propostas e desenvolvimentos, inspirados por estes comportamentos coletivos e sociais de grupos como enxames de pássaros, cardumes de peixes ou enxames de insetos [13]; otimização por enxame de partículas (*Particle Swarm Optimization - PSO*) [14], otimização por colônia de formigas (*Ant Colony Optimization - ACO*) [15], colônia artificial de abelhas (*Artificial Bee Colony - ABC*) [16] e forrageamento de bactérias (*Bacterial Foraging - BF*) [17] são apenas alguns exemplos.

Continuando na mesma direção de outros algoritmos de otimização, o Algoritmo Fireworks (AF) [18], é baseado na imitação do comportamento emergente das explosões de fogos de artifício. No trabalho original [18], o algoritmo foi contrastado contra o PSO e o Clonal PSO [19], exibindo melhores resultados que ambos tanto em convergência quanto em velocidade e precisão.

Neste trabalho, imagens padrões como a *lena* e *farm*, foram originalmente obtidas de um formato sem compressão, TIFF, em resolução de 512x512 e convertidas para um formato de compressão com perdas, JPEG, para simular imagens originais (i.e., de uma câmera). Estas imagens foram então editadas para representar falsificações. Finalmente, as imagens editadas foram redimensionadas, para 256x256, para simular pós-processamento, distanciando a imagem editada da imagem original e assim aumentando os níveis de ruído na *ELA*. Níveis de erro foram então computados e através da combinação linear das diferenças absolutas com as imagens editadas, resulta-se numa imagem de referência. Esta imagem foi então utilizada como a base para a medida de *peak-signal-to-noise-ratio* (PSNR) da imagem filtrada. O AF foi então utilizado para encontrar a configuração ótima, isto é, a configuração com a melhor resultante PSNR, através da combinação estocástica de filtros passa-baixa como a mediana, Wiener ou NCDF, com parâmetros dos filtros, métodos de equalização e *sharpening*, um filtro de nitidez. Um sumário do método é apresentado na Figura 1.

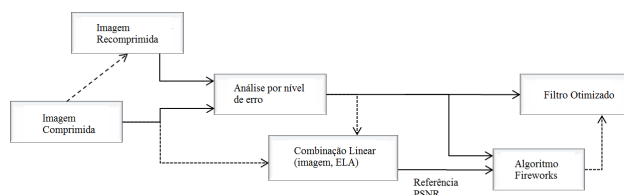


Figura 1: Sumário e fluxo do método apresentado.

O restante deste artigo está organizado da seguinte maneira: a seção 2 sumariza o método proposto, detalhando a aplicação da análise por níveis de erro e apresenta as imagens padrões de teste, suas versões editadas para este trabalho e os níveis de erro associados; a seção 3 apresenta os filtros utilizados neste trabalho e suas equações; a seção 4 descreve o Algoritmo Fireworks e suas propriedades; a seção 5 apresenta o problema de encontrar o filtro ótimo como um problema de otimização, especificando a função objetivo e o espaço de busca; a seção 6 apresenta os resultados obtidos pela otimização da configuração do filtro; finalmente, na seção 7, são apresentadas as etapas do método, sua eficiência é discutida e trabalhos futuros são mencionados.

## 2 ANÁLISE POR NÍVEIS DE ERRO

Análise por níveis de erro é um método passivo-cego para análise forense de imagens criado por Krawetz [9]. Há uma falha na literatura recente [10, 11] em identificar apropriadamente o autor original do método. Esquemas de compressão com perdas efetuam uma troca entre qualidade de dados e tamanho dos dados comprimidos, em primeira instância, isto pode parecer como um empecilho para o analista forense devido a perda de evidência associada com a troca, entretanto, diferentes níveis de qualidade em uma imagem são evidências por si mesmos.

A análise dos coeficientes DCT para detectar fraudes, como em [1], tende a múltiplos mínimos, devido a comparação com múltiplos valores inteiros de quantização [10], e dependente do formato JPEG. A análise por níveis de erro foca nas diferenças absolutas entre os pixels de uma imagem e os mesmos pixels recomprimidos, tornando-a menos tendenciosa a encontrar múltiplos mínimos, a não ser que os valores de quantização na compressão sejam múltiplos uns dos outros [10], e efetivo sempre que perdas de compressão existam.

O método consiste em obter uma imagem comprimida por um algoritmo com perdas, intencionalmente recomprimir a imagem a uma taxa conhecida de erro e então computar as diferenças absolutas entre a imagem original e a imagem recomprimida. Em uma representação matemática, os níveis de erro  $ELA(n_1, n_2)$ , onde  $n_1$  e  $n_2$  são índices de linha e coluna, podem ser representados por

$$ELA(n_1, n_2) = |X(n_1, n_2) - X_{rc}(n_1, n_2)|, \quad (1)$$

para cada canal de cor, onde  $X$  é a imagem suspeita de falsificação e  $X_{rc}$  é a imagem recomprimida. Níveis totais de erro são níveis de erro normalizados por todos os canais de cor, como em

$$ELA(n_1, n_2) = \frac{1}{3} \sum_{i=1}^3 |X(n_1, n_2, i) - X_{rc}(n_1, n_2, i)|, \quad (2)$$

onde  $i = 1, 2, 3$ , para uma imagem RGB.

Essas diferenças entre imagens são os níveis de erros associados com os pixels originais. Estes níveis de erro, vistos como uma quantidade de mudança, são diretamente associados com a perda de compressão. Se a quantidade de mudança for pequena, o pixel atingiu seu mínimo local para erro na determinada taxa de erro. Entretanto, se há uma grande quantidade de mudança, então os pixels não estão no seu mínimo local e são provavelmente advindos de fonte externa [9]. É importante ressaltar que os níveis de erro são apresentados imitando as frequências espaciais locais, o que pode dificultar a análise forense, de tal maneira que altas frequências também implicam em grandes quantidades de mudança e baixas frequências implicam em pouca quantidade de mudança.

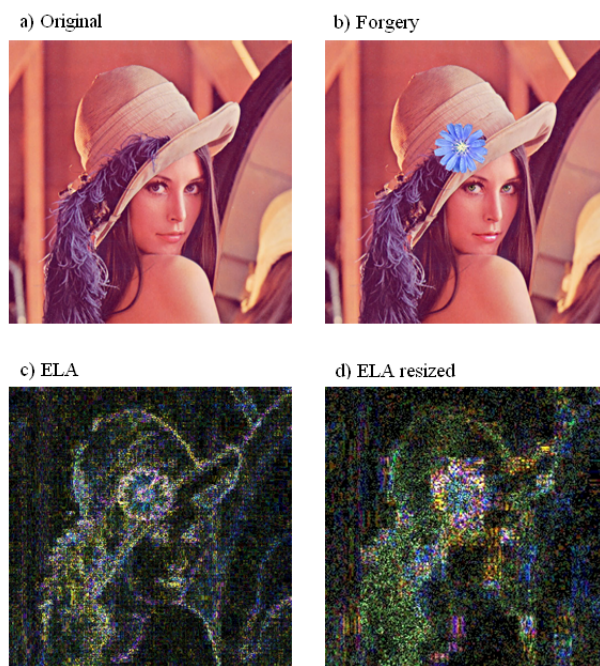


Figura 2: a) a imagem *lena* original, b) a falsificação, com uma flor no seu chapéu e com olhos e lábios contrastados, c) ELA para a imagem editada, d) ELA para a imagem editada e redimensionada para 256x256, mostrando um aumento substancial em ruído.

As Figuras 2 e 3 apresentam as imagens *lena* e *farm*, respectivamente. Exibidas nestas figuras estão as imagens originais e as editadas, assim como a análise por níveis de erro para as versões sem pós-processamento, em dimensões 512x512, e para as versões com pós-processamento, em dimensões 256x256, mostrando aumentos significativos de ruído. É facilmente visível como o aumento no ruído é prejudicial à ELA, particularmente na Figura 3 d), onde o ruído torna extremamente difícil detectar a falsificação.

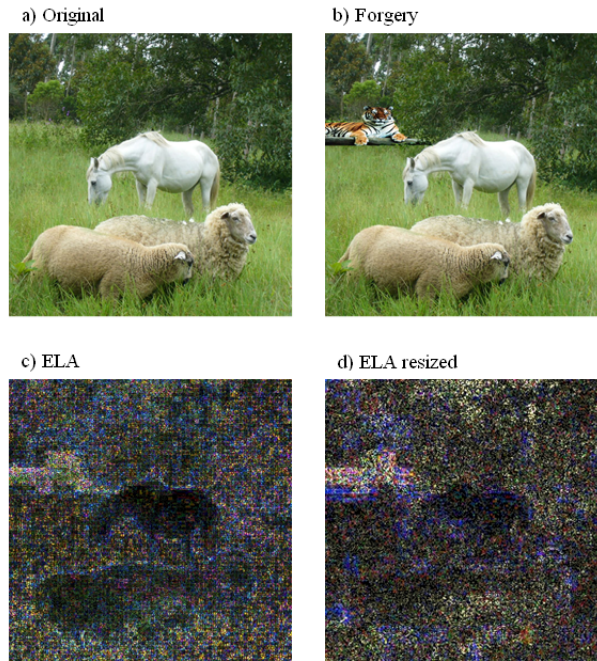


Figura 3: Figura a) a imagem *farm* original, b) a falsificação, com um tigre no fundo, c) ELA para a imagem editada, d) ELA para a imagem editada e redimensionada para 256x256, mostrando um grande aumento substancial em ruído.

### 3 FILTROS PASSA-BAIXA

Alterações súbitas na intensidade de uma imagem, observadas como bordas e ruídos, são vistas como alta frequência espacial. Disso, vem a troca que filtros passa-baixa devem fazer entre borramento da imagem e atenuação de ruído. De um lado, através da escolha de parâmetros, o filtro pode degradar a qualidade da imagem e do outro, pode melhorá-la, implicando na existência de uma configuração ótima dependendo do sinal em mãos.

O filtro não-linear da mediana [20,21] é uma particularidade dos filtros de posição [22,23]. Ele substitui o valor de cada pixel pela mediana dos pixels vizinhos. É notavelmente eficiente com ruídos de impulso, como ruído sal e pimenta ou speckle [24]. Considerando um pixel  $X(n_1, n_2)$ , onde  $n_1$  e  $n_2$  são índices de linha e coluna, e uma vizinhança  $MN$ , conhecida como a janela, o filtro da mediana pode ser representado por:

$$\hat{X}(n_1, n_2) = \text{median}\{X(\hat{n}_1, \hat{n}_2) \mid (\hat{n}_1, \hat{n}_2) \in MN\} . \quad (3)$$

O filtro adaptativo de Wiener [25–27] considera o sinal corrompido por um processo Gaussiano branco, sendo excelente em filtrar sinais Gaussianos estacionários. O filtro de Wiener é descrito por:

$$\hat{X}(n_1, n_2) = \mu + \frac{\sigma^2 - v^2}{\sigma^2} (X(n_1, n_2) - \mu) , \quad (4)$$

onde  $\mu$  and  $\sigma$  são a média e variância locais,  $v$  é a variância do ruído estimada pela média de todas as variâncias locais estimadas. Este filtro não é adaptativo por definição, já que considera o ruído Gaussiano como estacionário. Entretanto, pode ser facilmente tornado um filtro adaptativo ao considerar a média e variância locais na vizinhança de cada pixel, tomando uma janela de tamanho  $mn$ :

$$\mu = \frac{1}{mn} \sum_{n_1, n_2 \in MN} X(n_1, n_2) , \quad (5)$$

$$\sigma^2 = \frac{1}{mn} \sum_{n_1, n_2 \in MN} X(n_1, n_2)^2 - \mu^2 . \quad (6)$$

O filtro não-linear adaptativo de difusão complexa (*adaptive nonlinear complex diffusion filter - NCDF*) [28–31] apresenta características úteis para atenuar ruído speckle enquanto preserva características da imagem. Embora sua definição esteja muito acima do escopo deste trabalho, o método é delineado pela atualização iterativa de uma imagem de acordo com:

$$\hat{X} = X + \Delta t (D \Delta_h X + \nabla_h D \cdot \nabla_h X) , \quad (7)$$

onde  $\Delta_h$  e  $\nabla_h$  são os operadores discretos laplaciano e gradiente,  $\Delta t$  é o passo no tempo para um tempo máximo  $TMAX$  e  $D$  é o coeficiente de difusão is the diffusion coefficient. As definições exatas destes parâmetros podem ser encontradas em [31] assim como uma discussão profunda do NCDF pode ser encontrada em [28]. A idéia por trás deste método de filtragem é utilizar equações derivativas parciais (*partial derivative equations - PDE*), conhecidas como modelos de difusão, para detecção de bordas, as quais são iterativamente amaciadas. Bordas pequenas, como ruído speckle, são diluídas e bordas grandes, como características da imagem, não serão tão degradadas devido a suas características maiores.

Tanto o filtro da mediana e o filtro Wiener dependem da escolha de dois parâmetros, que formam o tamanho da janela,  $M-by-N$  e o NCDF depende da escolha de  $TMAX$ , que determina o nível de diluição das bordas.

## 4 ALGORITMO FIREWORKS

O algoritmo proposto por Tan e Zhu é chamado Algoritmo Fireworks, em homenagem a seu processo inspirador, um show de fogos de artifício. Na visão de seus criadores, a explosão de um único fogo de artifício pode ser vista como uma busca local no espaço diretamente em volta, enquanto a explosão de diversos fogos de artifício se assemelham a uma busca global. No trabalho original, o algoritmo é comparado contra a otimização por enxame de partículas e a otimização por enxame de partículas clonal, mostrando melhores resultados em funções de teste tanto em velocidade de convergência quanto melhor solução global.

A Figura 4 apresenta o pseudo-código do AF, onde a variável  $D$  apresenta o número de dimensões no espaço de busca,  $VR$  é uma matriz de limites superiores e inferiores para cada dimensão,  $n$  apresenta a contagem de fogos de artifício,  $m$  é o limite de faíscas criadas pelo algoritmo 1,  $m2$  é o limite de faíscas criadas pelo algoritmo 2,  $a$  é o parâmetro de limite inferior para arredondar a quantidade de faíscas do algoritmo 1,  $b$  o limite superior para o mesmo propósito  $A$  é a amplitude máxima das explosões.

```

BEGIN FA(D, VR, n, m, m2, a, b, A)
  fireworks = create n random fireworks of dimension D respecting VR
  stop_criteria = 0

  while stop_criteria == 0
    fx = evaluate fireworks

    si = calculate spark count for each firework with m
    si = round si with minimum m*a and maximum m*b

    Ai = calculate explosion amplitudes with A

    sparks1 = generate si sparks with algorithm 1 for each firework using amplitude Ai
    sparks2 = generate m2 sparks with algorithm 2 for random fireworks using amplitude Ai
    fsparks = evaluate sparks1 and sparks2

    best = select best from fireworks and sparks

    weights = evaluate distances between all fireworks and sparks
    selected = random sample n-1 from (fireworks, sparks1, spark2) with weights

    fireworks = (best, selected)

    if best is optimal
      stop_criteria = 1
    end if
  end while
END FA

```

Figura 4: Pseudo-código do Algoritmo Fireworks original.

Detalhes dos algoritmos 1 e 2 fogem do escopo deste trabalho e podem ser encontrados facilmente no trabalho original. Em uma abordagem literal ao pseudo-código, é dito que inicialmente,  $n$  fogos de artifício são aleatoriamente colocados no espaço de busca respeitando os limites definidos por  $VR$ . Após, até que um critério de parada seja satisfeito, os fogos de artifício são avaliados.

A quantidade de faíscas,  $si$  para cada explosão é diretamente dependente da qualidade da solução, enquanto a amplitude da explosão,  $Ai$ , é inversamente dependente da qualidade da explosão. Isto é, melhores soluções geram mais faíscas em um espaço menor, concentrando a busca local, enquanto soluções piores criam menos faíscas em um espaço maior, privilegiando a busca global. Isso reduz o esforço computacional sem denegrir a eficiência do algoritmo.

A quantidade de partículas, e avaliações da função objetivo, são dependentes da qualidade das soluções encontradas. No pior caso, há no máximo um total de  $n + am \cdot n + m2$  partículas ao final de cada iteração e apenas  $n$  partículas procedem à próxima iteração, através de um processo de seleção por amostragem, privilegiando a diversidade das soluções.

## 5 OTIMIZAÇÃO DE FILTROS

O problema neste trabalho é remover o ruído de um mapa de níveis de erro enquanto são preservadas características significativas. Primeiramente, uma métrica deve ser criada para o que deve ser considerada uma "boa" imagem filtrada. Devem ser preservadas características tanto da imagem original quanto dos níveis de erro, logo, é interessante definir uma imagem de referência como a combinação linear de ambos, como em

$$X^* = \frac{X + ELA}{2} \quad (8)$$

PSNR pode então ser empregado como uma métrica entre as imagens filtradas e a imagem de referência  $X^*$ . O problema se torna um de maximização, descrito por

$$\max_{f, fp, e, s} PSNR(X^*, X_h), \quad (9)$$

onde  $X_h$  é a imagem filtrada,  $f, fp, e, s$ , são métodos de filtragem, parâmetros do filtro, método de equalização e *sharpness*, respectivamente.

A imagem filtrada é criada por um filtro,  $h$ , usado para remover ruído, ajustando níveis de imagem e *sharpening*, de acordo com

$$X_h = h(X, f, fp, e, s). \quad (10)$$

Variar parâmetros na equação 10 resulta em diferentes filtros. Estes parâmetros determinam o espaço de busca. As variáveis  $f, fp, e, s$ , são mantidas no domínio real para uma melhor eficiência do AF porém são arredondas conforme necessário para a função do filtro.

O método do filtro,  $f$ , é selecionado como um de: Mediana, Wiener ou NCDF. Os parâmetros do filtro,  $fp$ , são definidos por duas variáveis,  $m$  e  $n$ . Estas são mantidas no espaço  $[3, 25]$ , implicando numa janela mínima de  $3 \times 3$  e uma janela máxima de 10% do tamanho da imagem,  $25 \times 25$ .  $m$  e  $n$  são usados apenas para os filtros da mediana e Wiener, no caso do filtro NCDF estes parâmetros são utilizados para calcular *TMAX*:

$$TMAX = 0 + (1 - 0) \cdot \frac{m + n - 2 * 3}{2 * 25 - 2 * 3} + \xi, \quad (11)$$

onde  $\xi$  é uma pequena constante para evitar *TMAX* zero. Desta maneira, o tempo máximo de difusão é mantido no espaço  $(0, 1]$ .

O método de equalização  $e$ , é um de: melhora de contraste, equalização do histogram ou normalização dos níveis de intensidade.

Já o filtro de nitidez, *sharpness*, é apenas determinado por um booleano: desligado ou ligado.

## 6 RESULTADOS

Os parâmetros do AF foram escolhidos empiricamente, baseados em uma região de operação aparentemente ótima, 5 fogos de artifício, 5 faíscas por explosão, redimensionamento mínimo de 0.04 e máximo de 0.8, amplitude 10 e 5 faíscas Gaussianas por explosão. O processo de otimização totalizou 30 experimentos, gerando diferentes filtros, cada experimento com 100 iterações. Os melhores três filtros, de acordo com a PSNR, para cada mapa de níveis de erro são apresentados.

Para a imagem *lena*, resultados são apresentados na figura 5. O primeiro filtro utiliza o filtro Wiener com janela  $5 \times 24$ , normalização da intensidade e *sharpening*. O segundo filtro utiliza o filtro Wiener com janela  $17 \times 7$ , melhora do contraste e sem *sharpening*. Já o terceiro filtro utiliza o filtro da mediana com janela  $11 \times 15$ , melhora do contraste e *sharpening*.

For the farm image, results are presented in figure 6. The first filter uses the Wiener filter with window  $18 \times 6$ , intensity normalization and sharpening. The second filter uses the median filter with window  $12 \times 15$ , histogram equalization and sharpening. The third filter uses the Wiener filter with window  $19 \times 21$ , contrast enhancement and sharpening.

Para a imagem *farm*, resultados são apresentados na figura 6. O primeiro filtro utiliza o filtro Wiener com janela  $18 \times 6$ , normalização da intensidade e *sharpening*. O segundo filtro utiliza o filtro da mediana com janela  $12 \times 15$ , equalização do histograma e *sharpening*. Finalmente, o terceiro filtro utiliza o filtro Wiener com janela  $19 \times 21$ , melhora do contraste e *sharpening*.

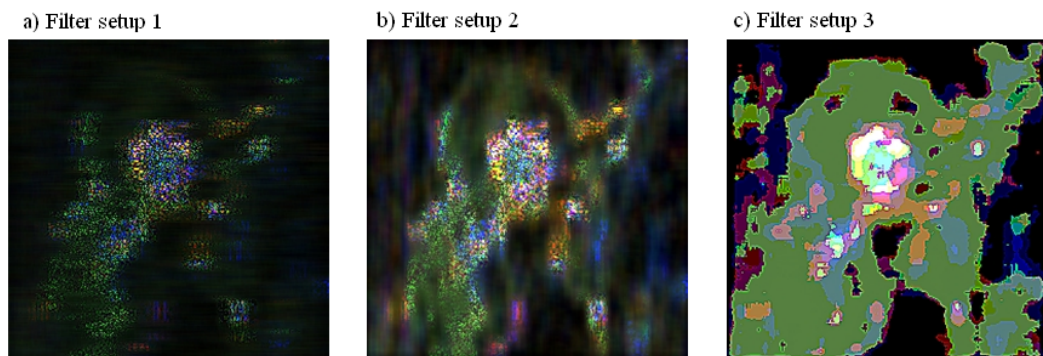


Figura 5: A flor apresenta níveis de erro de alta intensidade, indicando edição, entretanto, regiões de altas frequências como as plumas também aparentam ser níveis de erro de alta intensidade. As pequenas edições nos olhos e lábios são perdidas pela magnitude do ruído. Figura a) apresenta resultados para a primeira configuração de filtro, b) para a segunda e c) para a terceira.

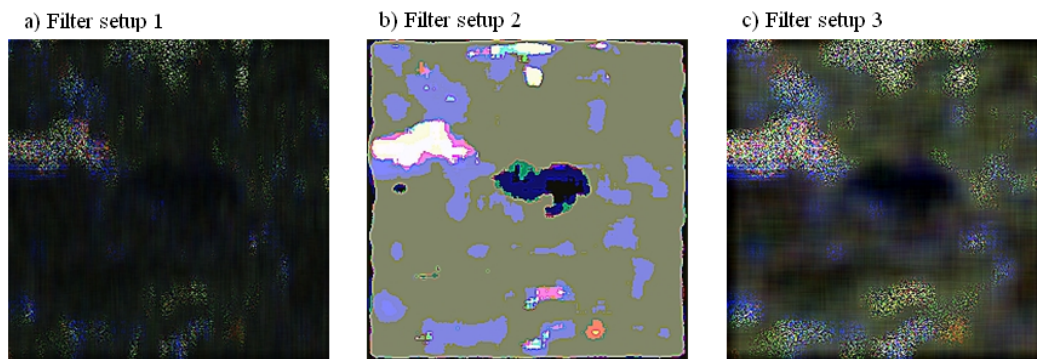


Figura 6: O tigre destaca-se como uma massa de níveis de erro de alta intensidade, indicando edição. The tiger stands out as a tight mass of high intensity error levels, indicating tampering. Figura a) apresenta resultados para a primeira configuração de filtro, b) para a segunda e c) para a terceira.

## 7 Conclusão

Este trabalho apresenta um método não tão conhecido, a análise por níveis de erro, corretamente identificando seu autor original apesar da omissão da literatura recente, e investiga a utilização do AF na detecção automática de configurações ótimas para filtros de ruído e aperfeiçoadores de imagem.

Imagens padrões como a *farm* e *lena*, foram editadas para representar falsificações. Estas imagens foram então estudadas com a análise por níveis de erro e comparadas com a análise por níveis de erro de versões com menor resolução destas mesmas imagens. Foi observado que o redimensionamento, análogo a operações de pós-processamento, degrada os níveis de erro, efetivamente aumentando o ruído observado nos mesmos.

Após, o AF foi utilizado para automaticamente, e de maneira estocástica, criar filtros, empregando métodos de atenuação de ruído em imagens como o filtro da mediana, filtro Wiener e filtro NCDF, assim como aperfeiçoadores de imagem como ajuste de contraste, equalização do histograma, normalização de intensidades e filtro de nitidez, *sharpening*.

Empiricamente, resultados mostram que esta abordagem atenua com sucesso ruído e melhora níveis de erro, melhor identificando regiões da imagem onde edições possam ter ocorrido.

## AGRADECIMENTOS

Os autores agradecem o apoio financeiro do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq, processo: 303963/2009-3/PQ). Além disso, o primeiro autor agradece a UFPR por uma bolsa de mestrado associada ao programa REUNI e o segundo autor agradece a USP por uma associada ao CNPq (processo: 134458/2010-9).

## REFERÊNCIAS

- [1] Z. Lin, J. He, X. Tang and C.-K. Tang. “Fast, automatic and fine-grained tampered JPEG image detection via DCT coefficient analysis”. *Pattern Recognition*, vol. 42, no. 11, pp. 2492–2501, 2009.
- [2] T.-T. Ng and S.-F. Chang. “Blind Detection of Photomontage Using Higher Order Statistics”. In *IEEE International Symposium on Circuits and Systems*, pp. 688–691, Vancouver, Canada, 2004.
- [3] Y.-Y. Chuang, A. Agarwala, B. Curless, D. H. Salesin and R. Szeliski. “Video matting of complex scenes”. In *Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '02*, pp. 243–248, San Antonio, Texas, 2002. ACM.
- [4] V. Kwatra, A. Schödl, I. Essa, G. Turk and A. Bobick. “Graphcut textures: image and video synthesis using graph cuts”. *ACM Trans. Graph.*, vol. 22, pp. 277–286, July 2003.
- [5] M. Bertalmío, G. Sapiro, V. Caselles and C. Ballester. “Image inpainting”. In *Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '00*, pp. 417–424, New Orleans, Louisiana, USA, 2000.
- [6] A. Bugeau, M. Bertalmío, V. Caselles and G. Sapiro. “A Comprehensive Framework for Image Inpainting”. *IEEE Transactions on Image Processing*, vol. 19, no. 10, pp. 2634–2645, 2010.
- [7] M. A. Savelonas and S. Chountasis. “Noise-resistant watermarking in the fractional Fourier domain utilizing moment-based image representation”. *Signal Processing*, vol. 90, no. 8, pp. 2521–2528, August 2010.
- [8] B. Mahdian and S. Saic. “Using noise inconsistencies for blind image forensics”. *Image Vision Computing*, vol. 27, pp. 1497–1503, September 2009.

- [9] N. Krawetz. “A Picture’s Worth... Digital image analysis and forensics”. Technical report, Black Hat Briefings, USA, 2007.
- [10] H. Farid. “Exposing digital forgeries from JPEG ghosts”. *Trans. Info. For. Sec.*, vol. 4, pp. 154–160, March 2009.
- [11] Y. Q. Zhao, F. Y. Shih and Y. Q. Shi. “Passive detection of paint-doctored JPEG images”. In *Proceedings of the 9th International Conference on Digital Watermarking, IWDW’10*, pp. 1–11, Berlin, Heidelberg, 2011. Springer-Verlag.
- [12] A. E. Eiben and J. E. Smith. *Introduction to Evolutionary Computing*. Springer Heidelberg, 2010.
- [13] S. Das, A. Abraham and A. Konar. *Swarm Intelligence Algorithms in Bioinformatics*, volume 94 of *Studies in Computational Intelligence*, chapter 4, pp. 113–147. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.
- [14] J. Kennedy, R. Eberhart and Others. “Particle swarm optimization”. In *Proceedings of IEEE International Conference on Neural Networks*, volume 4, pp. 1942–1948, Perth, Western Australia, 1995. Piscataway, NJ: IEEE.
- [15] M. Dorigo. “Optimization, Learning and Natural Algorithms”. Ph.D. thesis, Politecnico di Milano, 1992.
- [16] D. Karaboga and C. Ozturk. “A novel clustering approach: Artificial Bee Colony (ABC) algorithm”. *Applied Soft Computing (in press)*, 2009.
- [17] H. Chen, Y. Zhu and K. Hu. “Multi-colony bacteria foraging optimization with cell-to-cell communication for RFID network planning”. *Applied Soft Computing*, vol. 10, no. 2, pp. 539–547, March 2010.
- [18] Y. Tan and Y. Zhu. “Fireworks Algorithm for Optimization”. *Advances in Swarm Intelligence*, pp. 355–364, 2010.
- [19] Y. Tan and Z. M. Xiao. “Clonal particle swarm optimization and its applications”. In *2007 IEEE Congress on Evolutionary Computation*, pp. 2303–2309, Tokyo, Japan, September 2007.
- [20] T. Sun, M. Gabbouj and Y. Neuvo. “Center weighted median filters: Some properties and their applications in image processing”. *Signal Processing*, vol. 35, no. 3, pp. 213–229, 1994.
- [21] T. Nides and N. Gallagher Jr. “Center weighted median filters: Some properties and their applications in image processing”. *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 30, no. 5, pp. 739–746, 1982.
- [22] Y. Nakagawa and A. Rosenfeld. “A note on the use of local min and max operations in digital picture processing”. Technical report, Maryland University College Park, Computer Science Center, 1978.
- [23] G. Heygster. “Rank filters in digital image processing”. *Computer Graphics and Image Processing*, vol. 19, no. 2, pp. 148–164, 1982.
- [24] G. R. Arce. *Nonlinear Signal Processing: A Statistical Approach*. John Wiley & Sons Inc., New York, 2004.
- [25] R. Dugad and N. Ahuja. “Video Denoising by Combining Kalman and Wiener Estimates”. In *Proc. of the International Conference on Image Processing(ICIP)*, pp. 152–156, Kobe, Japan, 1999.
- [26] R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, second edition, 2001.
- [27] J. S. Lim. *Two-dimensional signal and image processing*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1990.
- [28] G. Gilboa, S. Member, N. Sochen and Y. Y. Zeevi. “Image enhancement and denoising by complex diffusion processes”. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 8, pp. 1020–1036, 2004.
- [29] H. M. Salinas and D. C. Fernández. “Comparison of PDE-Based Nonlinear Diffusion Approaches for Image Enhancement and Denoising in Optical Coherence Tomography”. *IEEE Transactions on Medical Imaging*, vol. 26, no. 06, June 2007.
- [30] D. C. Fernández, H. M. Salinas and C. A. Puliafito. “Automated detection of retinal layer structures on optical coherence tomography images”. *Optics Express*, vol. 13, no. 25, pp. 10200–10216, Dec 2005.
- [31] R. Bernardes, C. Maduro, P. Serranho, A. Araújo, S. Barbeiro and J. Cunha-Vaz. “Improved adaptive complex diffusion despeckling filter”. *Optics Express*, vol. 18, no. 23, pp. 24048–24059, Nov 2010.