

## **Exposing Digital Forgeries by Detecting Duplicated Image Regions**

by Alin C. Popescu and Hany Farid

11 de Outubro de 2010

**Revisores:** Ewerton Almeida Silva — RA 100588

Filipe de Oliveira Costa — RA 109230

### **1 Visão global**

Popescu e Farid [3] propõem um método para detecção de manipulações de cópia-colagem em imagens digitais baseado em comparação e casamento de blocos de *pixels*. O grande diferencial do algoritmo em relação a outras abordagens da literatura é a utilização da Análise de Componentes Principais (*Principal Component Analysis* – PCA) para redução da dimensionalidade dos blocos de *pixels*.

### **2 Resumo**

Os autores descrevem um algoritmo para identificação de regiões duplicadas em imagens digitais. O método consiste, inicialmente, na determinação de um bloco de tamanho  $b \times b$  que é posteriormente deslizado (com sobreposição) sobre a imagem de tamanho  $M \times N$  em ordem *raster* (de cima para baixo e da esquerda para direita). Cada bloco é, então, linearizado e armazenado numa matriz contendo  $(M - b + 1)(N - b + 1)$  linhas. Entretanto, previamente ao armazenamento desses blocos, a quantidade de informações presente em cada um deles é reduzida com o auxílio de PCA.

A motivação para o emprego de PCA é a eliminação de pequenas variações nos blocos ocasionadas por operações de pós-processamento nas regiões duplicadas, tais como introdução de ruídos aditivos e compressão JPEG. As variações provocadas nas regiões clonadas por meio dessas operações dificultam a busca por blocos idênticos e, portanto, devem ser atenuadas. Com a Análise de Componentes Principais é possível encontrar uma nova representação para cada bloco a partir da projeção destes nos  $n$  primeiros autovetores do espaço formado. Assim, apenas as “componentes principais” de cada bloco persistem na nova representação, e a variabilidade causada pelas operações de pós-processamento é reduzida.

O próximo passo do algoritmo é a quantização de cada bloco em  $Q$  bins. Isso aprimora o bloco no sentido de que também reduz variações pequenas na sua representação. Em seguida, a matriz de blocos linearizados é preenchida e ordenada lexicograficamente. Esse procedimento garante que linhas similares fiquem mais próximasumas das outras.

Cada bloco tem sua localização na imagem dada pela coordenada do seu *pixel* do canto superior esquerdo. Assim, para cada linha e suas  $t$  linhas subsequentes, calcula-se o *offset* entre elas, isto é, o vetor de desloca-

mento entre dois blocos na imagem. Um contador armazena o número de ocorrências de cada *offset*. Além disso, calcula-se a magnitude de cada *offset* e um limiar é aplicado sobre esse valor, indicando que regiões com magnitude de *offset* pequena (abaixo do limiar) tendem a estar muito próximas e, por isso, têm grandes chances de serem partes de uma região homogênea, como, por exemplo, céu azul. Os respectivos blocos são, então, descartados.

Finalmente, um novo limiar é aplicado sobre o número de cada *offset* computado: apenas os blocos que contribuíram para os vetores de deslocamento com frequência (de ocorrência) maior que tal limiar são considerados como duplicações na imagem.

Os autores efetuaram experimentos com 100 imagens de tamanho  $512 \times 512$  pixels comprimidas em JPEG com fatores de qualidade variando de 50% a 100% ou com ruídos aditivos Gaussianos; no que diz respeito a este último cenário, a métrica adotada foi a relação sinal ruído (*Signal Noise Ratio – SNR*), com variação de 24db a 40db. Entretanto, os autores não informam se foram executados testes nesses dois cenários combinados. Ademais, as regiões duplicadas tiveram tamanhos mínimos de  $32 \times 32$  e máximos de  $160 \times 160$  pixels. O tamanho do bloco de inspeção (janela deslizante) foi igual a  $8 \times 8$ , tendo sido reduzido à metade com o emprego de PCA.

Os resultados foram reportados em termos da taxa de acertos e da taxa de falso-positivos. Com relação à primeira, os resultados obtidos levaram a concluir que o método é mais preciso na detecção de regiões duplicadas maiores e quando a qualidade de compressão e a SNR são elevadas, visto que os blocos possuem variabilidade menor nesses casos. Em contrapartida, o número de falso-positivos esteve entre 15% a 20% na maioria dos testes, tanto para diferentes tamanhos de duplicação quanto para fatores de compressão e de SNR variáveis.

### 3 Contribuições

Os autores apresentam uma técnica eficiente e robusta que detecta automaticamente regiões duplicadas em imagens. Este tipo de detecção se faz bastante importante em um cenário forense, cujo maior objetivo consiste na detecção de adulterações em imagens digitais. Com esta técnica, se torna mais difícil a criação de falsificações digitais com credibilidade.

A técnica apresentada pelos autores se torna uma ferramenta útil para identificar, por exemplo, se a imagem da cena de um crime foi adulterada com o intuito de mascarar alguma informação importante a fim de influenciar o julgamento de algum indivíduo.

### 4 Defeitos/Desvantagens

O artigo parece correto tecnicamente. De modo geral, o trabalho está coeso e de fácil compreensão. Porém, é possível notar alguns pontos fracos na técnica proposta e nos resultados apresentados, como segue abaixo.

1. A técnica apresentada se mostra eficiente em imagens digitais, porém possui uma alta complexidade para ser utilizada diretamente na detecção de regiões duplicadas em vídeos.
2. Os autores consideram que a região duplicada sofre somente operações geométricas de translação. Portanto, o algoritmo não se faz eficaz caso a duplicação de uma região da imagem seja seguida de outras operações geométricas, como rotação, redimensionamento e espelhamento.
3. Os autores apresentam a taxa de falso-positivos obtida com os experimentos. Porém, na detecção de manipulação em imagens, a taxa de falso-negativos possui maior relevância, considerando que, neste

caso, o sistema assume que não houve manipulação na imagem, quando na realidade ela ocorreu. E neste trabalho os autores não apresentam a taxa de falso-negativos obtida.

## 5 Trabalhos correlatos

### 5.1 Detection of Copy-Move Forgery in Digital Images – Jessica Fridrich et al. [1]

**Relação com o artigo avaliado:** O artigo apresenta dois métodos para detecção de cópia-colagem baseados em comparação entre blocos de *pixels*, tal como é efetuado por Popescu e Farid [3]

**Descrição:** O primeiro método apresentado no artigo se baseia na comparação exata entre blocos de *pixels* da imagem. Primeiramente, a imagem é dividida em blocos. Em seguida, os *pixels* de cada bloco são salvos em um vetor na ordem lexicográfica. Feito isto, é feita a comparação entre cada vetor que representa os blocos, e aqueles que possuírem uma alta semelhança podem ser considerados cópias.

A desvantagem do primeiro método é que regiões homogêneas da imagem podem ser consideradas como regiões onde houve manipulação. Além disso, pequenas alterações nas regiões clonadas são suficientes para que o método se torne ineficaz. A segunda abordagem apresentada (casamento aproximado) visa minimizar este problema. Primeiramente é calculado, para cada bloco, o coeficiente da Transformada Discreta do Cosseno (*Discrete Cosine Transform* – DCT), que são salvos em uma matriz. Em seguida, as linhas desta matriz são ordenadas lexicograficamente e comparadas, como na abordagem anterior. Dado que os coeficientes DCT quantizados estão sendo comparados ao invés da representação do *pixel*, o algoritmo pode encontrar muitas falsas correspondências. Para evitar este problema, os autores consideram que, se duas linhas consecutivas da matriz forem encontradas pelo algoritmo, estas são salvas em uma lista separada. Um contador é incrementado para cada par de fileiras consecutivas encontrado na correspondência da matriz. Caso o número de fileiras consecutivas ultrapasse um limiar definido pelo usuário, os respectivos blocos são considerados como pertencentes a uma cópia-colagem.

Uma dificuldade deste método é a escolha do limiar. Valores altos podem levar o algoritmo a não reconhecer blocos que seriam resultado de clonagens, enquanto valores baixos podem introduzir muitas falsas correspondências.

### 5.2 Detecting Image Region Duplication Using SIFT Features – Pan and Lyu [2]

**Relação com o artigo avaliado:** O artigo apresenta um método para detecção de cópia-colagem fundamentado na correspondência de características SIFT da imagem, em oposição ao método tradicional baseado na comparação entre blocos de *pixels*, tal como é efetuado por Popescu e Farid [3] e Fridrich et al. [1]. Segundo Pan e Lyu, o método baseado na comparação de blocos de *pixels* é ineficiente na tarefa de identificação das regiões duplicadas quando estas são alvo de operações geométricas como escala e rotação.

**Descrição:** Visando produzir uma técnica para identificação de regiões duplicadas robusta a diversos tipos de manipulações, os autores propõem utilizar as características SIFT (*Scale Invariant Features Transform*) da imagem. SIFT é um algoritmo capaz de detectar pontos-chave (extremos locais estáveis) na imagem, isto é, invariantes a alguns tipos de operações como por exemplo, redimensionamentos (escala) e rotações. Ainda, segundo os autores, além de tratar com eficiência a presença de operações geométricas, o algoritmo proposto é robusto a operações como compressão JPEG, introdução de ruídos Gaussianos e mudança global de contraste e iluminação.

O método apresentado é constituído de quatro etapas. Na primeira, o algoritmo SIFT é usado para detectar pontos invariantes às operações supracitadas e na geração, para cada ponto, de uma representação

128-dimensional (um vetor de características). Em seguida, a imagem é dividida em vários blocos de pontos-chave sem sobreposição. Nesta etapa, o objetivo é encontrar a correspondência de cada ponto-chave na imagem (o vizinho mais próximo de cada um deles). Ainda, correspondências influenciadas por ruídos (que tendem a se corresponder, já que o SIFT é sensível a este tipo de artefato) são desconsideradas. Os blocos que possuem as maiores correspondências são candidatos a fazerem parte das regiões original e duplicada.

A ideia da terceira etapa do método é estimar as transformações geométricas efetuadas nas regiões clonadas. Considera-se que os dois blocos com maior correspondência entre pontos-chave são partes das regiões original e duplicada. Os pontos-chave desses blocos são examinados em busca de uma estimativa para o fator de escala e para o ângulo de rotação que podem ter sido aplicados. Finalmente, na última etapa a imagem é dividida em vários blocos de tamanho  $4 \times 4$ . Em seguida, a correlação entre cada região é computada efetuando-se as transformações encontradas. Se o valor da correlação for maior do que um limiar preestabelecido, o bloco corresponde a um segmento duplicado.

Os autores efetuam testes muito parecidos com os de Popescu e Farid [3], à exceção de que são efetuados testes com rotações e redimensionamentos. Apesar de altas taxas de precisão e do baixo número de falsos-positivos, os autores não exemplificam o funcionamento do método para rotações e escala, apenas mostram um exemplo visual no qual a região duplicada aparenta ter sido corretamente identificada nesses cenários. Além disso, não mostram resultados para a utilização dessas operações em conjunto.

## 6 Extensões

Esta pesquisa poderia ser aplicada juntamente com a técnica proposta por Pan and Lyu [2] para identificar possíveis transformações geométricas nas regiões duplicadas, considerando também a transformação de espelhamento, além de considerar composições de operações geométricas.

## 7 Notas

1. Relevância: 8.5
2. Originalidade: 9.5
3. Qualidade científica: 9.0
4. Apresentação: 8.0
5. Nota final: 8.5

## Referências

- [1] Jessica Fridrich, David Soukal, and Jan Lukas. Detection of copy-move forgery in digital images. In *Digital Forensic Research Workshop (DFRWS)*, Cleveland, USA, 2003.
- [2] Xunyu Pan and Siwei Lyu. Detecting image region duplication using sift features. In *IEEE Intl. Conference on Acoustics, Speech and Signal Processing (ICASSP 2010)*, pages 1706–1709, 2010.
- [3] Alin C. Popescu and Hany Farid. Exposing digital forgeries by detecting duplicated image regions. Technical Report TR 2004-515, Dept. of Computer Science – Dartmouth College, Hanover, USA, 2004.