Universidade Federal de São Carlos
Bacharelado em Ciência da Computação
Processamento de Imagens
Prof. Siovani Cintra Felipussi

Proposta de Projeto Final

Detecção de Manipulação de Imagens

Bruno Frítoli Carraza – 770993
Jade Manzur de Almeida – 771025

Tema da Proposta

O tema que será trabalho em nosso projeto final é a detecção de manipulação em imagens, ou seja, um algoritmo capaz de analisar e identificar imagens digitais que sofreram alterações ou modificações, para garantir a autenticidade do conteúdo da imagem.

Introdução e Motivação

A manipulação e edição de imagens não é um assunto novo. O processo de retocar e alterar uma imagem data desde a metade do século IX, época em que as imagens eram pintadas e recortadas manualmente para remover ou adicionar detalhes que antes não pertenciam à imagem.

Atualmente, qualquer um tem o poder de editar suas fotos usando de softwares próprios para a edição. como simples editores que só aplicam filtros, aplicativos voltados para usuários comuns, como o *Facetune* ou softwares para uso profissional, como é o poderoso *Photoshop*. Simplesmente ter à disposição um software capaz de modificar todo e qualquer aspecto de uma imagem é uma ferramenta muito poderosa que muitas vezes não é usada com sabedoria.

Por exemplo, nas redes sociais, muitas celebridades e usuários editam suas fotos para alterar suas características físicas, acrescentando músculos aonde não existem ou removendo imperfeições que não são consideradas atraentes. A exposição de outros usuários à essas imagens pesadamente editadas gera um impacto negativo ao favorecer a ideia de um corpo perfeito inatingível. Um exemplo ainda mais grave é a manipulação de imagens para espalhar *fake news*, usando imagens fora de contexto ou alterando-as para reforçar uma ideia falsa. Uma imagem existe como uma evidência, e para espalhar uma notícia fraudulenta, não faz diferença se as evidências forem falsas ou tiradas de contexto. A presença de uma imagem adiciona legitimidade a qualquer coisa simplesmente só de estar lá, e isso é usado para espalhar facilmente *fake news* e provocar reações do público.

A edição e manipulação de imagens, quando não usada com cautela possui um impacto negativo grande, e por isso analisar e poder detectar uma imagem que foi alterada é bastante valioso para garantir a autenticidade de seu contéudo, pois já que uma imagem vale mais do que mil palavras, nenhuma delas deveria ser mentira.

Proposta

Quando uma imagem sofre uma edição, alguns rastros são deixados para trás, já que no processo de edição, alguns pixels da imagem sofrem modificações drásticas e contrastam com os outros que não sofreram mudanças. Por exemplo, uma edição muito comum que é feita em imagens é o uso de uma ferramenta para retocar imperfeições, bastante usada em publicidades e anúncios em revistas, essa ferramenta normalmente funciona borrando os pixels em um certo ponto da imagem e isso causa uma perturbação aos pixels adjacentes e remove ruído dessa parte da imagem, então, em comparação à todo resto da imagem, que não foi modificado, a área alterada terá menos ruído e apresentará um borramento perto da edição.

O projeto consiste em aplicar algoritmos para analisar e detectar os rastros que são deixados para trás na edição de uma imagem e verificar se houve ou não uma falsificação desta.

Mesmo assim, nem toda modificação consiste em uma falsificação da imagem. Procedimentos como cortar, rotacionar ou aplicar um filtro de correção não forjam uma imagem e nem alteram o seu conteúdo, por isso, identificar esses procedimentos não é o foco do projeto.

Existem muitas técnicas para se identificar uma modificação em uma imagem, e é do interesse do projeto testar e implementá-las, considerando tudo que é ensinado na disciplina de Processamento de Imagens.

Para meios de implementação o escopo da proposta foi reduzido para analisar apenas imagens com extensão JPEG, já que os algoritmos estudados funcionam para esse formato de imagem.

Resultados Esperados

Os nossos objetivo com este projeto é criar uma aplicação capaz de identificar se uma imagem sofreu alteração através de técnicas forenses que identicam rastros deixados pelos métodos de edição ou filtros. Ainda estamos explorando as possibilidades para realizar o projeto e a melhor técnica que podemos implementar para que o trabalho não se torne muito complexo. Uma das técnicas que parece mais atrativa é a Análise de Nível de Erro (**Error level analysis**), um algoritmo que indentifica se um objeto foi inserido em uma imagem. Outras técnicas que estão sendo pesquisadas são: análise de inconsistência na qualidade da imagem, análise de metadados, efeito da dupla quantização (*double quantization effect*), detecção de clonagem/cópia de objetos em imagens.

Esperamos ter um projeto que tenha como entrada uma imagem e, através dos procedimentos implementados, seja capaz de informar se a imagem foi ou não forjada, mostrando o que levou à esse resultado.

Dificuldades Encontradas

Como o tema é bastante complexo, tivemos dificuldade em encontrar boas explicações sobre o básico do assunto. Mesmo entendendo como alguns algoritmos funcionam em alto nível, muitas vezes, a matemática se apresentava confusa e isso dificultou muito a implementação de vários dos algoritmos.

Muito tempo foi gasto tentando debugar e entender o que as milhares de variáveis significavam nos artigos cientificos. Muitas vezes, confundia-se a notação de função com a de matriz, variáveis

surgiam sem explicação de sua definição, muitas provas e etapas importantes que eram dadas como "exercício para o leitor", conceitos que por si só demorariam dias para entender o funcionamento eram explicados em pequenas frases com vocabulário confuso e ambíguo, etc... A parte matemática que deveria, então, conter as respostas de como tudo funciona acabava virando mais um empecilho

Quando conseguíamos fazer uma implementação não tinhamos noção de como checar se um número estava correto ou não depois de executar a conta. Tentamos ver implementações em outras linguagens também com uma tentativa até de transcrever um algoritmo de uma linguagem para outra sem sucesso. Então, após tudo isso, foi decidido usar algoritmos mais simples.

Para tentar encontrar algum ponto de partida, usamos algoritmos já conhecidos na área forense de imagens mas mesmo assim tivemos problemas. Várias das implementaçãos que tentamos usar acabaram não funcionando e grande parte desse tempo que usávamos para entender o funcionamento do código acabava sendo desperdiçado já que a teoria matemática e lógica por trás de um algoritmo (na nossa experiência) acabava sendo diferente da lógica e matemática usada em outras implementações. Outra coisa que não ajudou foi o fato dos códigos demorarem muito para rodar, então, a cada vez que tentavamos arrumar parâmetros ou semelhantes, nos custava de 10 até 15 minutos para checar se teríamos resultados.

Resultados Obtidos

Foram selecionados os seguintes algortimos:

- Análise de metadados
- Error Level Analysis

O primeiro, bem simples, apenas recupera os metadados de uma imagem e retorna no terminal do Octave. Se a imagem foi editada, será possível ver nos metadados o último programa em que ela foi aberta.

O segundo algoritmo, embora tenha uma implementação não tão complicada, se baseia profundamente em conceitos de processamento de imagem. Um algoritmo de *Error Level Analisys* só é aplicável à imagens com formato JPEG, isso porque ele se baseia na perda de dados que ocorre toda vez que uma imagem JPEG é aberta e salva novamente, devido à usar uma compressão com perda de dados (*lossy compression*). Uma imagem não editada possui uma compressão basicamente uniforme. Quando um pedaço de uma imagem JPEG é copiado e colado em outra, o pedaço apresenta uma compressão diferente do resto da imagem em que foi colocada. Devido à essa ocorrência, é possível identificar partes da imagem que não apresentam a mesma compressão. O algoritmo comprime a imagem para enfatizar essa diferença e então as imagens são subtraídas e o resultado é multiplicado por um valor para aumentar suas diferenças.

Como muitos dos demais algoritmos encontrados eram bastante difíceis de implementar, escolhemos um algoritmo bem interessante, baseado no artigo *A robust detection algorithm for copy-move forgery in digital images* escrito por Yanjun Cao, Tiegang Gao,Li Fan, Qunting Yanga .

Após realizar uma leitura a fundo, nos baseamos em uma implementação encontrada sobre o artigo para testar e implementar nossa versão do algoritmo. Fizemos uma implementação de acordo

com o proposto pelos autores e colocamos o algoritmo para teste com algumas imagens forjadas, para comparar seu desempenho em diferentes imagens.

Para realizar os testes, foi criado uma base com imagens retiradas do site https://www.forestryimages.org/wildlife.cfm, todas JPEG. Foram retiradas 10 imagens, e elas imagens foram editadas para gerar 5 imagens para serem testadas com *Error Level Analysis* e 5 que possuem uma edição do tipo *Copy-Move* para serem testadas com o algoritmo implementado do artigo.

É importante notar que as diferentes imagens possuem resultados interessantes quando passadas pelos diferentes algoritmos, por exemplo, quando utilizamos o algoritmo de ELA para analisar uma imagem editada com *Copy-Move*, o algoritmo basicamente não acusa edição alguma. O mesmo serve para o algoritmo de *Copy-Move*, já que é especializado em procurar similaridades na mesma imagem.

Conclusão e Perspectivas

A área de computação forense é muito abrangente e a análises de imagens possui um papel muito importante nela. Embora não tenha sido possível realizar nossos objetivos por completo neste projeto, foi possível esquematizar um esqueleto para futuras implementações.

Nosso projeto consegue facilmente identificar imagens JPEG que foram editadas de forma que formam uma imagem composta (um pedaço de uma imagem colocado em outra), e por meio de análise de metadados e observando os resultados nas imagens, é possível tirar conclusões a respeito da integridade da imagem. Também não conseguimos ter uma implementação única de um algoritmo de *Copy-Move Detection*, mas através da análise feita do algoritmo proposto, pudemos entender muito melhor como ele funciona e testar seus limites, o que consideramos uma vitória.

Para ter um software completamente funcional e que correspondesse à nossas expectativas iniciais, precisaríamos de muito mais estudo e mais experiência na área. Gostaríamos de expandir o projeto de modo que possamos implementar nossos próprios algoritmos e ainda testar outros, de forma que ele não se limite à imagens JPEG e possa até ser aplicado à *Deepfakes*. Uma abordagem interessante, não explorada aqui, seria o uso de inteligência artifical para detectar imagens manipuladas. O que temos é suficiente para identificar certos tipos de edições e serve como uma base para construir em cima, embora não seja, de todo, muito complexou ou como imaginamos

Felizmente, conseguimos implementar um projeto básico, que contribuiu para nosso aprendizado e possui um resultado satisfatório.