Instituto de Computação
Universidade Estadual de Campinas

Atribuição de fonte em imagens provenientes de câmeras digitais

Filipe de Oliveira Costa

Este exemplar corresponde à redação final da Dissertação devidamente corrigida e defendida por Filipe de Oliveira Costa e aprovada pela Banca Examinadora.

Campinas, 6 de julho de 2012.

Prof. Dr. Anderson de Rezende Rocha
(Orientador)

FICHA CATALOGRÁFICA ELABORADA POR
ANA REGINA MACHADO - CRB8/5467
BIBLIOTECA DO INSTITUTO DE MATEMÁTICA, ESTATÍSTICA E
COMPUTAÇÃO CIENTÍFICA - UNICAMP

Costa, Filipe de Oliveira, 1987-
C823a Atribuição de fonte em imagens provenientes de câmeras digitais / Filipe de Oliveira Costa. – Campinas, SP : [s.n.], 2012.

Orientador: Anderson de Rezende Rocha.
Dissertação (mestrado) – Universidade Estadual de Campinas, Instituto de Computação.

1. Processamento de imagens. 2. Computação forense. 3. Mineração de dados (Computação). 4. Inteligência artificial. I. Rocha, Anderson de Rezende, 1980-. II. Universidade Estadual de Campinas. Instituto de Computação. III. Título.

Informações para Biblioteca Digital

Título em inglês: Image source camera attribution
Palavras-chave em inglês:
Image processing
Forensic computing
Data mining
Artificial intelligence
Área de concentração: Ciência da Computação
Titulação: Mestre em Ciência da Computação
Banca examinadora:
Anderson de Rezende Rocha [Orientador]
Ricardo da Silva Torres
Marco Antônio Piteri
Data de defesa: 06-07-2012
Programa de Pós-Graduação: Ciência da Computação

TERMO DE APROVAÇÃO

Dissertação Defendida e Aprovada em 06 de Julho de 2012, pela Banca examinadora composta pelos Professores Doutores:

Prof. Dr. Marco Antonio Piteri
FCT / UNESP

Prof. Dr. Ricardo da Silva Torres
IC / UNICAMP

Prof. Dr. Anderson de Rezende Rocha
IC / UNICAMP

Instituto de Computação
Universidade Estadual de Campinas

Atribuição de fonte em imagens provenientes de câmeras digitais

Filipe de Oliveira Costa[1]

Julho de 2012

Banca Examinadora:

- Prof. Dr. Anderson de Rezende Rocha (Orientador)
- Prof. Dr. Marco Antônio Piteri – UNESP, Presidente Prudente, São Paulo
- Prof. Dr. Ricardo da Silva Torres – UNICAMP, Campinas, São Paulo
- Prof. Dr. Hélio Pedrini – UNICAMP, Campinas, São Paulo (Suplente)
- Prof. Dr. João Paulo Papa – UNESP, Bauru, São Paulo (Suplente)

[1]Projeto financiado pela FAPESP (processo 2011/03808-3)

Dedico este trabalho às pessoas mais importantes da minha vida: meus pais Moisés e Maria José e meus irmãos Mariana e André que sempre contribuíram para minha formação e estiveram, a todo momento, me apoiando nesta caminhada. Amo vocês.

Agradecimentos

Foram dois anos de muito trabalho, muitas noites mal dormidas... mas valeu a pena. E eu não poderia, de forma alguma, deixar de agradecer às pessoas que me ajudaram durante todo esse tempo.

Primeiramente agradeço a Deus por Suas bençãos, pois sem Sua luz, eu não seria nada. Obrigado, Senhor, por me dar força, proteção e nunca me deixar desistir. A ti toda honra e glória para sempre. Amem.

Gostaria muito de agradecer ao meu orientador Anderson Rocha, pela ajuda, por seus conselhos, por me mostrar o caminho a ser seguido e, principalmente, pela incrível paciência que ele teve comigo. Você é um exemplo de pessoa e profissional que quero sempre seguir. Agradeço também ao professor Michael Eckmann, que lá dos Estados Unidos me ajudou muitas vezes. *Thank you very much, Mike!*

Agradeço ao Instituto de Computação, a todos os professores e funcionários, pela oportunidade de realizar uma pós-graduação, pelo espaço físico disponível, e aos amigos novos, pelas ajudas, pelas risadas e pelo santo futebol de toda sexta-feira. Um obrigado também à FAPESP pelo suporte financeiro.

Agradeço a todos os meus familiares (avós, tios, primos e por aí vai... a família é gigante) e a meus grandes amigos: Guizinho, Rafael, Eduardo, Zé Luiz, Gabi, Mia, Bruna e CIA LTDA (infelizmente não cabe o nome de todos aqui): sem vocês não sou ninguém! Obrigado também aos meus queridos "alunos" da Engenharia Química da UNICAMP, pela amizade, pela companhia e pelos vários momentos de muita risada.

Agradeço também aos professores da Universidade Federal de Alfenas - MG, que mesmo depois de eu ter me formado, sempre me apoiaram nesta minha caminhada, e ao pessoal do LInC, por salvar minha vida durante alguns experimentos do meu mestrado. Valeu!

Por fim, agradeço às pessoas que são as mais importantes da minha vida e que contribuíram diretamente com a minha formação profissional e pessoal: minha mãe Maria José, meu pai Moisés e meus irmãos mais novos Mariana e André. Não imagino minha vida sem vocês. Obrigado por tudo! Amo muito vocês.

O sofrimento é passageiro.
Desistir é pra sempre.

Lance Armstrong

Resumo

Verificar a integridade e a autenticidade de imagens digitais é de fundamental importância quando estas podem ser apresentadas como evidência em uma corte de justiça. Uma maneira de se realizar esta verificação é identificar a câmera digital que capturou tais imagens. Neste trabalho, nós discutimos abordagens que permitem identificar se uma imagem sob investigação foi ou não capturada por uma determinada câmera digital. A pesquisa foi realizada segundo duas óticas: (1) *verificação*, em que o objetivo é verificar se uma determinada câmera, de fato, capturou uma dada imagem; e (2) *reconhecimento*, em que o foco é verificar se uma determinada imagem foi obtida por alguma câmera (se alguma) dentro de um conjunto limitado de câmeras e identificar, em caso afirmativo, o dispositivo específico que efetuou a captura. O estudo destas abordagens foi realizado considerando um cenário aberto (*open-set*), no qual nem sempre temos acesso a alguns dos dispositivos em questão. Neste trabalho, tratamos, também, do problema de correspondência entre dispositivos, em que o objetivo é verificar se um par de imagens foi gerado por uma mesma câmera. Isto pode ser útil para agrupar conjuntos de imagens de acordo com sua fonte quando não se possui qualquer informação sobre possíveis dispositivos de origem. As abordagens propostas apresentaram bons resultados, mostrando-se capazes de identificar o dispositivo específico utilizado na captura de uma imagem, e não somente sua marca.

Abstract

Image's integrity and authenticity verification is paramount when it comes to a court of law. Just like we do in ballistics tests when we match a gun to its bullets, we can identify a given digital camera that acquired an image under investigation. In this work, we discussed approaches for identifying whether or not a given image under investigation was captured by a specific digital camera. We carried out the research under two vantage points: (1) *verification*, in which we are interested in verifying whether or not a given camera captured an image under investigation; and (2) *recognition*, in which we want to verify if an image was captured by a given camera (if any) from a pool of devices, and to point out such a camera. We performed this investigation considering an open set scenario, under which we can not rely on the assumption of full access to all of the investigated devices. We also tried to solve the device linking problem, where we aim at verifying if an image pair was generated by the same camera, without any information about the source of images. Our approaches reported good results, in terms of being capable of identifying the specific device that captured a given image including its model, brand, and even serial number.

Sumário

v

Agradecimentos **vi**

vii

Resumo **viii**

Abstract **ix**

1 Introdução **1**

2 Estado da arte **4**
2.1 Análise Forense de Documentos Digitais 4
2.2 Atribuição de fonte de documentos 6
2.2.1 Identificação do modelo de aquisição 8
2.2.2 Identificação do dispositivo específico 10
2.3 Cenário aberto (*open-set*) 23

3 Atribuição de fonte em cenário aberto **26**
3.1 Abordagem Proposta 26

4 Correspondência entre Dispositivos (*Device Linking*) **40**
4.1 Abordagem Proposta 41

5 Experimentos e validação **44**
5.1 Conjunto de dados 44
5.2 Experimentos e Resultados 45
5.2.1 Atribuição de fontes em cenário aberto 45
5.2.2 Correspondência entre dispositivos 56
5.3 Considerações finais 56

6 Conclusão e trabalhos futuros **58**

Bibliografia **60**

Lista de Tabelas

2.1 Comparativo entre técnicas do estado da arte no domínio da atribuição de fonte de imagens. 22

5.1 Câmeras utilizadas em nossos experimentos. 45
5.2 Resultados ($ACC_F \pm$ desvio padrão, em (%)), para 15, 10, 5 e 2 câmeras disponíveis durante o treinamento na análise de verificação. O cenário considera 35 câmeras no total. 49
5.3 Resultados (em %) considerando duas câmeras de mesma marca/modelo como classes conhecidas no treinamento. O cenário considera 35 câmeras no total na etapa de teste. 50
5.4 Comparação de resultados entre os cenários fechado (F) com 15 câmeras e aberto (A) considerando 15 câmeras disponíveis para treinamento e 35 para testes. 50
5.5 Matriz de confusão para a análise de Reconhecimento – Abordagem proposta. Acerto médio: 91.39%. 53
5.6 Matriz de confusão para a análise de Reconhecimento – Lukáš et al. [30]. Acerto médio: 78.18%. 54
5.7 Matriz de confusão para a análise de Reconhecimento – Li [26]. Acerto médio: 74.63 %. 55
5.8 Resultados obtidos com *2-fold cross-validation* executado em cada experimento. 57

Lista de Figuras

2.1 Possível *pipeline* do processo de geração de uma imagem (via câmera digital). FONTE: Rocha e Goldenstein [37]. 7
2.2 Exemplo de arranjo de pixels em mosaico utilizando CFAs e posterior operação de demosaico. No caso, o filtro de mosaico/demosaico apresentado é conhecido como filtro de Bayer [19]. FONTE: Rocha e Goldenstein [37]. . 8
2.3 Presença de poeira em duas imagens diferentes obtidas pela mesma câmera. Ajustes locais de histograma são aplicados para tornar as partículas de poeira visíveis (parte inferior). Os quadrados brancos mostram a localização das partículas. FONTE: Dirik et al. [11]. 11
2.4 Hierarquia do padrão de ruído. FONTE: Rocha e Goldenstein [37]. 12
2.5 Abordagem proposta por Lukáš et al. [30] para a identificação da fonte geradora de uma imagem. 13
2.6 Exemplos de PRNU de uma imagem, onde (a) é a imagem original, (b) é o ruído residual obtido pela abordagem de Lukáš et al. [30] e (c) é o ruído residual aprimorado pela abordagem proposta por Li [26]. 14
2.7 Processo de decisão combinada proposto por Sutcu et al [43]. 17
2.8 Algoritmo proposto por Goljan e Fridrich [15] para verificar se duas imagens foram geradas pela mesma câmera. 18
2.9 Técnica de *Zero padding*. 19
2.10 Teste triângulo (*Triangle Test*). FONTE: Adaptado a partir de Goljan et al. [16] 19
2.11 Exemplo de classificação em cenário aberto. A abordagem em cenário aberto assume que nem todas as classes são conhecidas *a priori*. A figura mostra a classe de interesse ("pentágono"), bem como as classes negativas conhecidas ("triângulo", "quadrado"e "círculo") e desconhecidas ("?"). . . 24

3.1 Regiões de interesse (ROIs) de dimensão 512×512 *pixels*. 28
3.2 Exemplos de imagens sem (a) e com (b) artefato de *vignetting*. 28

3.3 Calculando padrão de ruído para uma região, considerando os canais de cores R, G, B e Y. O processo é realizado para todas as nove regiões marcadas na imagem. 29
3.4 Exemplo de um classificador SVM considerando o caso linear. 31
3.5 Nossa implementação do cenário aberto para atribuição de fonte de imagens utilizando *Decision Boundary Carving* (DBC). 34
3.6 Geração de um hiperplano para OC-SVM. 35
3.7 Definição dos hiperplanos de decisão para OC-SVM. O OC-SVM necessita somente da classe positiva ("círculo") para ser definido (a). Porém, como neste caso conhecemos outras duas classes negativas ("quadrado" e "triângulo"), podemos utilizá-las para melhor definirmos a posição dos hiperplanos de separação (b). 36
3.8 Processo de generalização (b) e especialização (c) para OC-SVM. As classes negativas aqui mostradas são apenas para definir os hiperplanos. O exemplo representa o caso do OC-SVM linear. 37
3.9 Variação dos parâmetros γ (0.0, 1.0 e 11.9) e ν (0.01, 0.52 e 0.99) na geração de hiperplano utilizando OC-SVM com núcleo RBF. 38
3.10 Exemplo de generalização (a) e especialização (b) de margens para OC-SVM. Em azul está representada a generalização da margem; em vermelho, a especialização; em cinza, o hiperplano inicialmente gerado para OC-SVM, com $\nu = 0.25$ e $\gamma = 0.25$. A classe de interesse é representada por +. 39

4.1 Extração de características para correspondência entre dispositivos. O processo é realizado para todas as nove regiões marcadas na imagem. . . . 42

5.1 Exemplos de imagens que compõem nosso conjunto de dados. 46
5.2 Identificação do dispositivo específico — *Verificação*. 47
5.3 Identificação do dispositivo específico — *Reconhecimento*. 51

Capítulo 1

Introdução

Como uma forma de representar unicamente um momento no espaço-tempo, a fotografia tornou-se uma forte aliada na interpretação de crimes e passou, ao longo do tempo, a ser utilizada com frequência em cortes de justiça, tanto como forma ilustrativa de testemunho quanto como evidências de crimes ligados à pornografia infantil e pirataria, por exemplo. Alguns países adotaram um método que afirma que a evidência fotográfica "fala por si" e é admissível a partir de testemunhos que estabelecem a forma como a imagem foi adquirida [3]. Com a evolução da tecnologia e a utilização de imagens digitais nesse cenário, a confiabilidade desse método torna-se questionável a partir da possibilidade de manipulação digital de tais imagens. Uma imagem gerada por meio de manipulação digital poderia ser utilizada, por exemplo, como falsa evidência de que uma pessoa esteve presente no local de um crime mesmo que essa pessoa nunca tenha estado em tal lugar.

Considerando que imagens digitais podem ser apresentadas como evidências de um crime em uma corte de justiça, verificar a sua integridade e sua autenticidade se tornou uma tarefa de fundamental importância. Uma forma de se fazer essa verificação é identificar a fonte geradora da imagem em questão. Da mesma forma que as ranhuras existentes em um projétil encontrado na cena de um crime podem ser utilizadas para identificar se uma determinada arma foi utilizada no momento do disparo [29], técnicas voltadas para a identificação da origem da imagem procuram detectar "marcas" deixadas na imagem pelo dispositivo gerador no momento da captura e geração da imagem. Estas marcas são provenientes de características próprias do dispositivo gerador, como defeitos de fabricação, modo de interação entre os componentes da câmera e a luz, algoritmos de geração de imagem implementados nos componentes do dispositivo, entre outros fatores.

Pesquisas na área de atribuição de fonte em imagens digitais procuram identificar a marca ou fabricante do dispositivo utilizado na geração de uma imagem, bem como o dispositivo exato. Para identificar o modelo do dispositivo utilizado na captura de uma imagem, os pesquisadores procuram estimar os algoritmos utilizados para a geração da

imagem, comparando com os algoritmos implementados em certos modelos de câmera [24, 34, 45].

Em particular, a tarefa de se encontrar o dispositivo específico utilizado na captura de uma imagem é a mais estudada. Existem propostas na literatura voltadas para a identificação da origem de uma imagem por meio do padrão de ruído deixado na imagem pelo dispositivo [30, 26, 18], artefatos gerados por imperfeições dos sensores de captura de um dispositivo [25, 12], e presença de partículas de poeira no sensor [11].

Identificar o dispositivo que gerou uma determinada imagem é uma forma de se garantir, por exemplo, que um documento foi gerado por uma câmera e não é resultado de qualquer manipulação digital; a informação de que uma foto foi obtida por uma câmera digital apreendida sob possessão de um suspeito poderia classificá-lo não mais como um consumidor mas sim como produtor de, por exemplo, fotos de pornografia infantil. A identificação da câmera que gerou uma imagem digital é um dos tópicos abordados no campo de pesquisas denominado *Análise Forense de Documentos Digitais*, cujo objetivo é verificar a integridade e autenticidade de documentos digitais [37].

Embora existam abordagens para a atribuição de fonte em imagens, essas pesquisas são realizadas considerando um cenário fechado (*closed-set*), no qual os autores assumem que uma imagem sob investigação foi gerada por uma entre n câmeras disponíveis durante a etapa de treinamento. Infelizmente, não podemos ter certeza de que uma imagem foi gerada por uma dessas câmeras. Na prática, uma imagem a ser avaliada pode ter sido gerada por uma câmera totalmente desconhecida que não faz parte de nosso grupo de câmeras suspeitas, o que torna importante a identificação deste fato. Portanto, é importante modelar o problema de atribuição de fontes considerando um cenário aberto (*open-set*), no qual temos acesso somente a um conjunto limitado de câmeras suspeitas e temos que treinar o modelo de classificação considerando somente este conjunto enquanto buscamos classificar corretamente imagens geradas por câmeras às quais não necessariamente temos acesso.

Em certos casos, um analista forense pode estar interessado em dizer se duas imagens foram geradas pela mesma câmera. Essa abordagem pode ser útil para decidir se um suspeito é o proprietário de um conjunto de imagens. Adicionalmente, identificar se duas ou mais imagens foram geradas pela mesma câmera poderia levantar suspeita de que algumas imagens desse conjunto foram manipuladas digitalmente (caso se saiba que uma imagem do conjunto é uma falsificação). Em outro cenário mais realista, podemos revelar que outras imagens foram geradas por uma câmera suspeita apenas comparando cada uma dessas imagens com imagens as quais sabemos que foi gerada pela câmera sob investigação, ou seja, sem a necessidade de acesso físico à essa câmera, o que pode ser útil para avaliar, por exemplo, se fotos criminosas postadas na internet foram geradas por uma câmera roubada. Este problema é denominado Correspondência entre Dispositivos

(*Device Linking*).

Considerando a importância do estudo deste tema, nesta dissertação apresentamos abordagens desenvolvidas visando identificar a fonte geradora de uma imagem em um cenário aberto. Estamos interessados em identificar o dispositivo específico utilizado na geração de uma imagem, e não somente sua marca. Além disso, apresentamos também uma abordagem para verificar se um par de imagens sob investigação foi gerado pela mesma câmera, sem saber qual é o dispositivo gerador.

Contribuições científicas As principais contribuições deste trabalho são:

- Implementação de técnicas existentes na literatura e verificação do funcionamento das mesmas;
- Criação de um conjunto de dados comum (que será público) e validação das técnicas da literatura neste conjunto de dados comum;
- Investigação do problema de atribuição de fontes em um cenário aberto, no qual uma imagem sob investigação pode ter sido gerada por um dispositivo desconhecido e não necessariamente por um dispositivo ao qual temos acesso (suspeitos);
- Investigação de novas abordagens para caracterização de imagens e integração das soluções em um cenário de aprendizado de máquina;
- Solução do problema de atribuição de fonte em um cenário aberto.

Organização O texto está organizado da seguinte forma: o Capítulo 2 discute o estado da arte na atribuição de fonte em imagens provenientes de câmeras digitais, além de discutir também, de forma geral, o problema de classificação em um cenário aberto. O Capítulo 3 apresenta a modelagem do problema de atribuição de fontes em cenário aberto. O Capítulo 4 discute o problema da correspondência entre dispositivos. O Capítulo 5 apresenta os experimentos realizados e os resultados obtidos para atribuição e correspondência entre dispositivos. Finalmente, o Capítulo 6 apresenta as considerações finais e os trabalhos futuros.

Capítulo 2

Estado da arte

Neste capítulo, apresentamos os trabalhos relacionados a este trabalho. Primeiramente, discutimos o conceito de *Análise Forense de Documentos Digitais*, seguido de sua sub-área *Atribuição de Fonte*, destacando os trabalhos que visam identificar o modelo da câmera utilizada na aquisição de uma imagem e trabalhos que tem como objetivo identificar o dispositivo específico utilizado para a aquisição. Descrevemos, também, o conceito de classificação em cenário aberto (*open-set*), em que informações sobre a origem de uma imagem a ser classificada podem não estar disponíveis no momento do treinamento do classificador.

2.1 Análise Forense de Documentos Digitais

Com o avanço tecnológico da última década, dispositivos de captura de imagens (e.g., câmeras, *camcorders*, *scanners*, etc.) tornaram-se acessíveis à população em geral. Em conjunto com poderosas ferramentas de edição de imagens (e.g., *Adobe Photoshop*, *Gimp*, *Corel Photo-Paint*, entre outros), esse avanço permitiu a usuários comuns se tornarem especialistas na geração e manipulação de documentos digitais. A partir do momento que essas manipulações deixam de ser inocentes e passam a implicar questões legais, faz-se necessário o desenvolvimento de abordagens para sua detecção [38].

Identificar se houve alteração em um documento digital é uma tarefa de extrema importância. O campo de pesquisa relacionado à verificação de autenticidade e integridade de documentos digitais é denominado *Análise Forense de Documentos Digitais*. Essa é uma área de pesquisa que está cada vez mais presente em nosso dia-a-dia nos auxiliando em questões legais. Por exemplo, o julgamento de um crime pode estar sendo baseado em evidencias fabricadas com o intuito de enganar e mudar a opinião do júri; um banco pode receber como verdadeiro um cheque falsificado de um cliente [37]; uma pessoa pode ser acusada de ter cometido um crime mesmo sem nunca ter estado no local do ocorrido [42].

De forma geral, na análise forense de documentos, dado um objeto (e.g., imagem), visamos obter soluções para as seguintes questões [40]:

- Tal objeto é original ou foi gerado a partir de composição de outros objetos digitais?
- Tal objeto realmente representa um momento único ou foi modificado digitalmente para enganar o visualizador?
- Qual o histórico de processamento de tal objeto?
- Quais partes do objeto sofreram adulterações e qual o impacto dessas modificações?
- O objeto foi adquirido pela câmera do fabricante F_1 ou do fabricante F_2?
- Tal objeto realmente é originário da câmera, filmadora ou *scanner* X como afirmado?

Atualmente, não existem metodologias estabelecidas para verificar a integridade e autenticidade de documentos digitais de maneira automática [40]. Em algumas situações, utilizam-se marcações digitais (*watermarking*), porém sabemos que a maioria das imagens e vídeos não possui marcações. A utilização de marcações digitais em documentos implicaria a implementação de tal abordagem diretamente nos sensores de aquisição dos dispositivos de captura, o que poderia elevar o custo dos dispositivos e tornaria seu uso restritivo. Além disso, devido à inserção das marcações, poderia haver perdas na qualidade do conteúdo das imagens ou vídeos [37]. Com isso, as técnicas propostas na literatura para análise forense de imagens e vídeos são categorizadas, basicamente, em três áreas, de acordo com o seu foco principal:

1. **Distinção entre objetos naturais e sintéticos:** Visa avaliar se um objeto (e.g., imagem) foi obtido a partir de um dispositivo de captura ou se foi gerada em computador.
2. **Identificação de adulterações:** O objetivo dessa área é estabelecer a autenticidade de documentos digitais ou expor quaisquer tipos de adulterações sofridas por ele, desde adulterações visando melhoria na qualidade visual do documento (e.g., ajuste de brilho e contraste em imagens) até modificações realizadas com a intenção de enganar o visualizador ("foto-montagens").
3. **Identificação da origem do objeto:** Tem como objetivo identificar o modelo particular de dispositivo que originou um determinado objeto, ou o dispositivo exato. Essa área é o foco principal deste trabalho.

Segundo Rocha et al. [38], uma característica adicional das técnicas citadas acima é que elas são denominadas técnicas de detecção cega e passiva. A detecção é cega no sentido de que não se faz necessária a presença do conteúdo original para comparação e é passiva no sentido de que não é necessária a utilização de nenhuma outra forma de marcação digital no processo geral.

Existem também algumas técnicas denominadas contra-forenses, que têm por objetivo atacar técnicas forenses já existentes de forma a afetar negativamente a existência e/ou a qualidade de uma evidência de crime ou até mesmo dificultar (ou tornar impossível) a análise de uma evidência a ser realizada. Uma pessoa que pretende atacar uma técnica pode se beneficiar de sua vulnerabilidade. Ao contrário do que se pensa, técnicas contra-forense são importantes, uma vez que, encontrando os pontos fracos de uma determinada técnica, pesquisadores podem se esforçar para estudar esses ataques e encontrar medidas para reduzir a sensitividade das técnicas a eles [37, 38, 4]. Nós não consideramos os efeitos de técnicas contra-forenses para essa área neste trabalho, uma vez que o foco é realizar a identificação da origem de imagens.

A seguir, apresentamos um estudo mais detalhado sobre a atribuição de fonte de documentos digitais.

2.2 Atribuição de fonte de documentos

As técnicas de atribuição de fonte de documentos são abordagens voltadas para a identificação das características do dispositivo de captura de um objeto (e.g., câmeras digitais, *scanners*, *camcorders*). Vale ressaltar que as informações de aquisição (e.g., marca, modelo e número de série do dispositivo gerador) são codificadas no cabeçalho do documento (e.g., cabeçalho EXIF – *Exchangeable Image File Format*). Porém, essas informações não possuem muita utilidade no cenário forense, pois podem facilmente ser destruídas ou alteradas [38]. Portanto, a existência do cabeçalho EXIF não será considerada neste trabalho.

Existem pesquisas na área de atribuição do sensor de captura voltadas para a identificação do *scanner* que capturou uma determinada imagem [22] [23], bem como na identificação da impressora utilizada na geração de um documento [9] [21] e a identificação da câmera digital ou *camcorder* que gerou uma imagem ou vídeo [8]. Neste trabalho, nos focamos na identificação de câmeras digitais a partir de imagens digitais.

A identificação de uma câmera digital que gerou uma determinada imagem requer o conhecimento das propriedades físicas e operacionais de tal dispositivo. Segundo Rocha e Goldenstein [37], o processo de aquisição de uma imagem digital ocorre da seguinte forma: os raios de luz entram na câmera através das lentes e passam por uma combinação de filtros que incluem, pelo menos, os filtros de infravermelho e anti-serrilhamento para garantir uma melhor qualidade visual. A luz então é focada no sensor de captura que nada mais é que uma matriz de *sensels* (elementos foto-sensíveis). Os sensores mais comuns são os baseados em CCDs (*charge-coupled devices*) ou CMOS (*complimentary metal-oxide semiconductor*). Cada ponto da matriz de *sensels* integra a luz incidente no sensor e obtém um sinal elétrico que representa a cena fotografada.

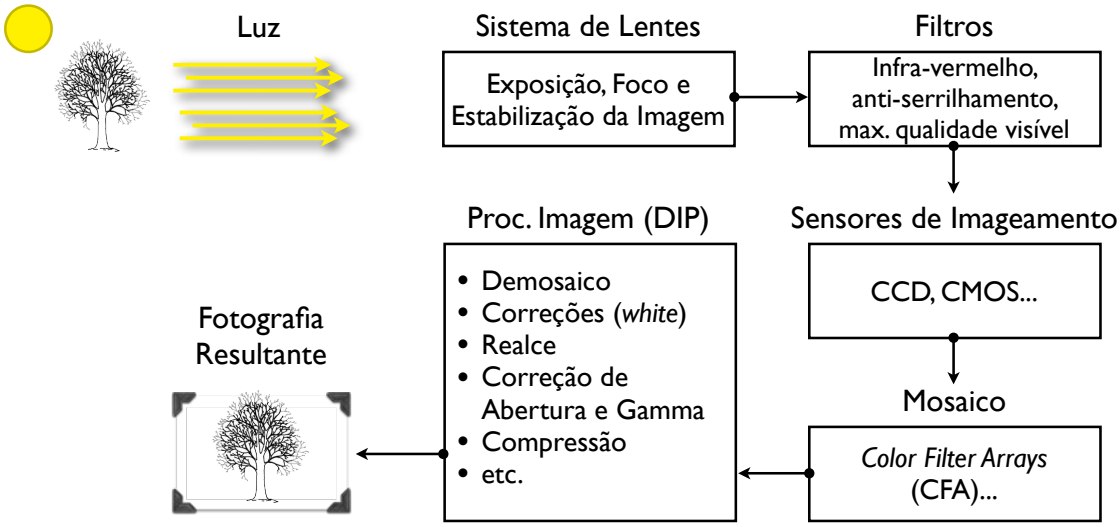


Figura 2.1: Possível *pipeline* do processo de geração de uma imagem (via câmera digital). FONTE: Rocha e Goldenstein [37].

Visando diminuir o custo final do dispositivo, geralmente cada *sensel* é monocromático. Com isso, as câmeras digitais empregam um processo de mosaico para organizar os pixels (*Color Filter Array* – CFA) de modo que cada *sensel* possua um filtro espectral e capte apenas uma banda do comprimento de onda da luz. Os CFAs mais comuns utilizam três sensores: vermelho (***R**ed*), verde (***G**reen*) e azul (***B**lue*). Sabendo que cada ponto possui somente uma cor, as outras duas cores ausentes são estimadas a partir de uma operação de interpolação de pixels denominada demosaico. Feito isto, a imagem entra em uma fase de pós-processamento, na qual passa por outras operações para melhoria da qualidade visual final, como correção pontual, correção de gamma, correção de abertura e compressão. A Figura 2.1 ilustra o processo de aquisição de uma imagem por uma câmera digital, enquanto a Figura 2.2 apresenta um exemplo de arranjo de pixels em mosaico utilizando CFAs. Pesquisas recentes apontam indícios de que as condições climáticas do ambiente capturado podem, também, influenciar a relação entre os sensores de captura da câmera e a luz do ambiente [35, 36].

Na identificação da origem de uma imagem, geralmente temos dois cenários possíveis: (1) a identificação da marca ou modelo do dispositivo utilizado na aquisição da imagem; e (2) as características da fonte específica utilizada. A seguir, apresentamos mais detalhes a respeito de técnicas empregadas na identificação de origem de uma imagem considerando os dois cenários citados acima.

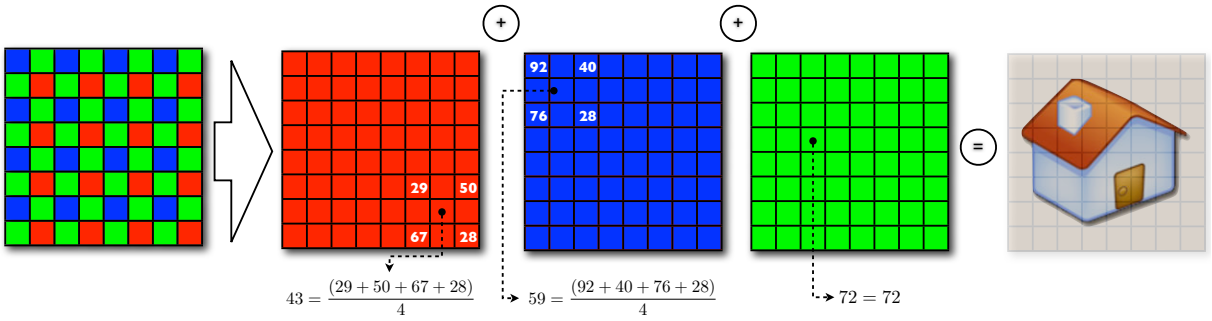


Figura 2.2: Exemplo de arranjo de pixels em mosaico utilizando CFAs e posterior operação de demosaico. No caso, o filtro de mosaico/demosaico apresentado é conhecido como filtro de Bayer [19]. FONTE: Rocha e Goldenstein [37].

2.2.1 Identificação do modelo de aquisição

A meta principal das técnicas voltadas para a identificação do modelo de aquisição é apontar o modelo ou o fabricante do dispositivo que capturou uma determinada imagem. Em se tratando de câmeras digitais, geralmente são levadas em consideração informações relacionadas ao processo de aquisição da imagem tais como: informações das lentes, tipo e tamanho dos sensores de aquisição, tipo de filtro de mosaico/demosaico empregado, algoritmos de processamento de imagens implementados na lógica de processamento da câmera entre outras [38]. A principal dificuldade dessa abordagem é que muitos modelos de marcas de dispositivos utilizam componentes de poucos fabricantes e os métodos de geração e processamento das imagens permanecem os mesmos (ou bastante similares) entre os diferentes modelos da mesma marca [11].

Uma característica que pode fornecer informações para distinguir imagens provenientes de dispositivos de fabricantes ou modelos diferentes é a presença de artefatos de cor inseridos nas imagens durante o processo de demosaico. Baseando-se nesse fator, Kharrazi et al. [24] propuseram um conjunto de 34 características para imagens coloridas que levam a um bom índice discriminatório entre variados modelos de câmeras digitais. Algumas características apresentadas pelos autores são: valores médios de pixels, correlação entre pares das bandas RGB, distribuição do centro de massa de pixels vizinhos, taxa de energia entre pares RGB e estatísticas no domínio Wavelet. Os autores afirmam, ainda, que diferentes câmeras produzem imagens com diferentes qualidades. Com isto, adicionalmente os autores utilizam características obtidas a partir de Métricas de Qualidade de Imagem (*Image Quality Metrics – IQM*), baseando-se em medidas de diferenças de pixels (e.g., erro médio quadrático e erro médio absoluto), medidas de correlação (e.g., correlação cruzada normalizada) e distância espectral entre pixels (e.g., distância de Czenakowski). Por fim, os autores utilizam um classificador de padrões

baseado em Máquina de Vetores de Suporte (*Support Vectors Machine* – SVM) para verificar a efetividade das características propostas. Nesse trabalho, os autores reportam resultados entre 78.7% e 95.2% de acurácia, considerando cinco modelos de câmeras.

Outros fatores que podem fornecer pistas sobre o modelo de câmera utilizado na captura de uma imagem são as operações de mosaico e demosaico empregadas e a escolha do filtro CFA. Para Swaminathan et al. [45], as imagens fornecem informações completas sobre o CFA e a operação de mosaico e demosaico utilizada pela fonte geradora da imagem, pois a entrada e a saída correspondem às imagens amostradas e interpoladas, respectivamente. Com isto, é possível identificar o modelo gerador da imagem, visto que câmeras de mesmo modelo utilizam os mesmos filtros CFA e a mesma operação de demosaico para a geração de uma imagem.

Popescu e Farid propuseram, em [34], a utilização do algoritmo para estimar os componentes de interpolação de cores. Primeiramente, os autores assumem que os pixels da imagem pertencem a um dos seguintes modelos: (1) um pixel é correlacionado com os seus vizinhos e seu valor é obtido a partir de interpolação linear; (2) um pixel não está correlacionado com os seus vizinhos. Partindo desse pressuposto, os autores propõem o algoritmo de Esperança/Maximização (*Expectation Maximization* – EM) para estimar os coeficientes CFA utilizados na geração da imagem em questão. Esse algoritmo é executado em duas etapas. Na primeira etapa (Esperança), estima-se a probabilidade de cada amostra pertencer a um dos dois modelos citados acima. Na segunda etapa (Maximização), a forma específica de correlação entre a amostra e os modelos é encontrada. Feito isto, o algoritmo EM gera duas saídas, sendo elas um mapa de probabilidade de duas dimensões indicando a probabilidade de cada pixel da amostra pertencer a um dos dois modelos, e o coeficiente de ponderação, que define o quão forte é a correlação entre o pixel e os modelos.

A estimativa do coeficiente de interpolação de cores pode auxiliar na distinção entre diferentes algoritmos de interpolação, podendo ser utilizada para determinar o algoritmo de demosaico utilizado em uma câmera em particular. Com isso, algumas extensões desse trabalho foram elaboradas, como a utilização do algoritmo EM para o problema de classificação de modelos de câmeras em conjunto pré-determinado com um classificador SVM [1, 46] ou o cálculo do erro assumindo-se um padrão CFA em uma imagem [44, 38].

As técnicas apresentadas acima são mais voltadas à identificação do modelo da câmera do que à identificação da câmera exata utilizada na obtenção de uma imagem, tendo em vista que essas técnicas examinam atributos que discriminam câmeras entre (e não intra) modelos.

2.2.2 Identificação do dispositivo específico

Para se realizar a identificação da câmera específica utilizada na geração de uma imagem e não do modelo em questão, é necessário analisar e identificar características únicas em relação à câmera empregada no momento da aquisição. Essas características podem ser oriundas de imperfeições dos componentes, defeitos e falhas originadas por efeitos do ambiente e condições de operação [25, 12, 45]. O maior desafio é estimar o dispositivo por meio de uma única imagem, pois o conteúdo da imagem pode dificultar essa estimativa [38, 37].

Kurosawa et al. [25] apresentaram um dos primeiros trabalhos a sugerir a utilização de imperfeições nos sensores para identificar o dispositivo específico que capturou uma imagem. Nesse trabalho, os autores sustentam que a identificação do ruído de padrão fixo causado por *dark currents* em câmeras digitais pode ser útil na identificação do sensor de captura de uma imagem. Um *dark current* é definido como a razão pela qual os elétrons se acumulam em cada pixel devido à ação termal, que é encontrada nas junções inversas dos pinos e independe da luz incidente. Os autores propuseram a detecção de defeitos localizados dos pixels, intensificando os ruídos de padrão fixo da imagem. A utilização dessa técnica é bastante limitada, tendo em vista que câmeras mais sofisticadas possuem sensores especiais e mecanismos para correção automática desses defeitos *on-the-fly*.

Existem, também, abordagens que analisam outros tipos de imperfeições. Dirik et al. [11] apresentam uma abordagem para identificar artefatos originados por presença de poeira nos sensores no momento da aquisição. Os autores afirmam que partículas de poeira presentes nos sensores criam um padrão permanente em todas as imagens capturadas e, com isto, é possível identificar a fonte de uma imagem, como mostrado na Figura 2.3. Experimentos apresentados pelos autores reportaram uma baixa taxa de falsos positivos e uma acurácia de classificação de 94%. No entanto, os autores consideraram um cenário com somente duas câmeras.

A grande limitação das abordagens baseadas em informações de poeira no sensor é o fato de que a característica utilizada para a identificação do dispositivo específico que capturou a imagem é temporal por natureza (pode-se limpar o sensor, por exemplo). Nesse contexto, pesquisadores na área forense devem estar atentos a essas informações sempre que possível [37].

Geradts et al. [12] propõem a análise de defeitos dos componentes como uma possível forma de relacionar uma imagem à sua fonte geradora. Os autores consideram a presença de pixels supersaturados (*hot pixels*), pixels pouco saturados (*dead pixels* ou *cold pixels*) e defeitos agrupados (*pixel traps*).

Pixels supersaturados são pixels individuais do sensor de captura com uma carga maior que a normal. Pixels com pouca saturação são aqueles que apresentam pouquíssima ou nenhuma carga. Defeitos agrupados são definidos como uma interferência com o processo

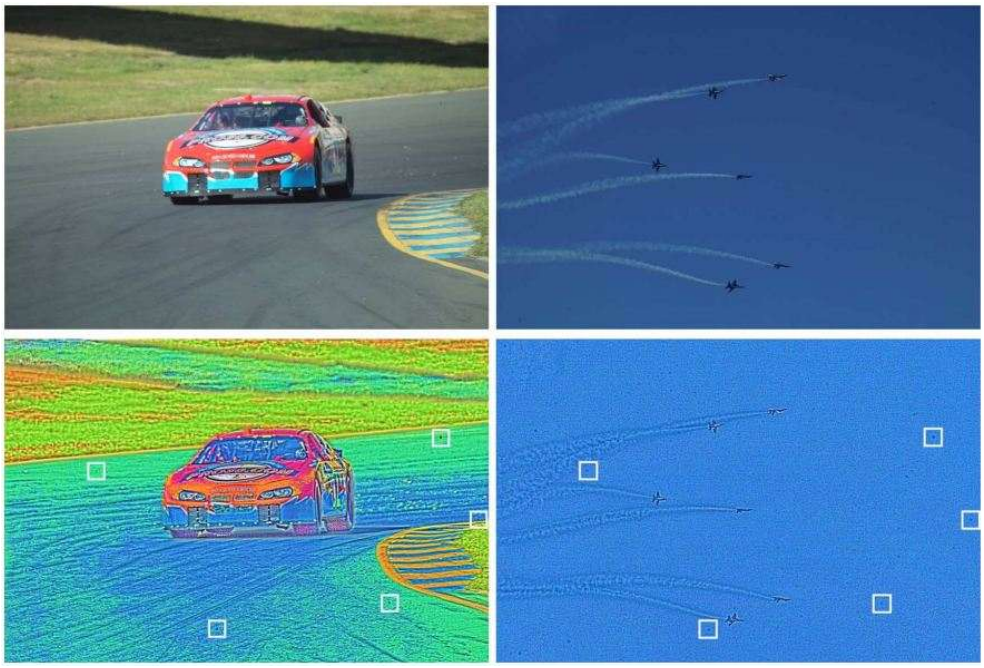


Figura 2.3: Presença de poeira em duas imagens diferentes obtidas pela mesma câmera. Ajustes locais de histograma são aplicados para tornar as partículas de poeira visíveis (parte inferior). Os quadrados brancos mostram a localização das partículas. FONTE: Dirik et al. [11].

de transferência de carga durante a captura levando a uma linha total ou parcialmente danificada da imagem (e.g., toda branca ou toda preta) [37]. Assim como Dirik et al. [11], Geradts et al. [12] consideram que a presença desses defeitos nas imagens torna possível a identificação da fonte geradora, pois criam um padrão permanente em todas as imagens capturadas.

A presença de pixels defeituosos dificilmente é utilizada na prática para identificar a fonte geradora de uma imagem, visto que nem todas as câmeras possuem pixels defeituosos. Além disso, como dito anteriormente, câmeras mais sofisticadas possuem sensores especiais que corrigem esses defeitos internamente no momento da captura da imagem.

Atualmente, as técnicas mais efetivas para a identificação do dispositivo de captura específico analisam os efeitos do ruído inserido no processo de captura de imagens. Nesse contexto, Lukáš et al. [30] propõem uma maneira de se realizar a estimativa do padrão de ruído dos sensores para identificar o dispositivo gerador de uma imagem. De forma geral, podemos categorizar o ruido em imagens de acordo com a Figura 2.4.

Vemos dois tipos de padrões de ruídos: padrão de ruído fixo (causado pelas *dark currents* e descritos anteriormente) e ruído decorrente da foto-responsividade não uniforme

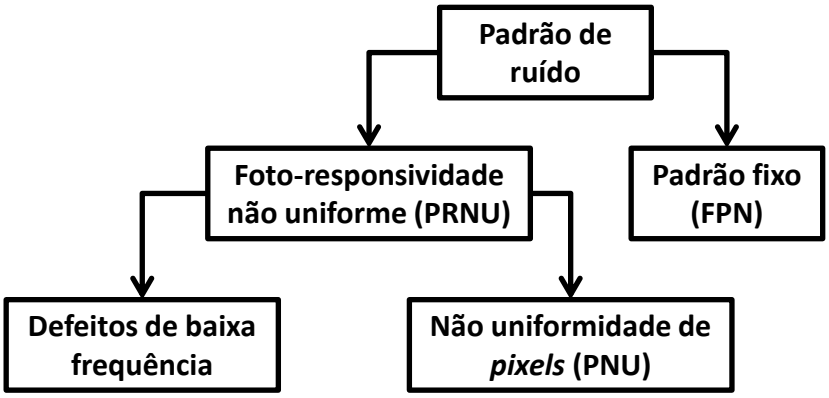


Figura 2.4: Hierarquia do padrão de ruído. FONTE: Rocha e Goldenstein [37].

(*Photo Response Non-Uniformity Noise* – PRNU). A utilização do ruído fixo se torna limitada, conforme esclarecido anteriormente. Os defeitos de baixa frequência são provocados por refração dos raios de luz nas partículas na câmera ou próximas dela, superfícies óticas e configurações de *zoom*. Esses tipos de ruídos geralmente não são considerados devido à sua natureza efêmera [37, 38]. O ruído decorrente do componente PRNU, por outro lado, é calculado pelo ruído não uniforme dos *pixels* (PNU). O PNU é definido como a sensitividade que diferentes *pixels* possuem à luz e é causado basicamente por inconsistências no sensor durante o processo de fabricação.

Para se utilizar o ruído PNU na identificação de sensores de captura, a natureza desse ruído deve ser isolada. Em um cenário forense, não é possível termos acesso à uma imagem de referência que nos permitiria a recuperação do ruído PNU. Assim sendo, um algoritmo de identificação de câmeras que utilize informações baseadas no ruído PNU deve, primeiramente, estabelecer uma aproximação desse ruído. Essa aproximação é definida como padrão de ruído do sensor (*Sensor Pattern Noise* – SPN). O processo de aproximação é feito da seguinte maneira: para cada imagem I pertencente a um conjunto K de imagens calcula-se o ruído residual R_i utilizando um filtro de redução de ruído F.

$$R_i = I_i - F(I_i), \text{ em que } I_i \in K \tag{2.1}$$

Lukáš et al. [30] utilizam um filtro de supressão de ruído baseado na Transformada Wavelet Discreta (*Discrete Wavelet Transform* – DWT) [31]. Os autores mostraram que a utilização desse filtro supressor de ruído possui bons resultados.

Em seguida, é feita a aproximação P_c do SPN, calculada como sendo a média dos

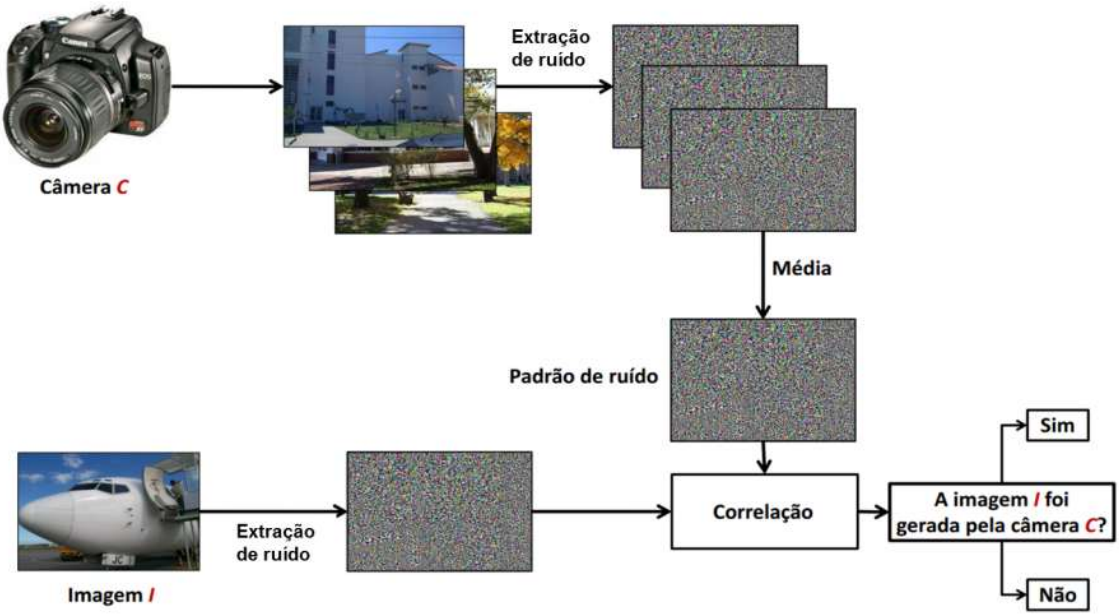


Figura 2.5: Abordagem proposta por Lukáš et al. [30] para a identificação da fonte geradora de uma imagem.

resíduos de ruído das $k = |K|$ imagens.

$$P_c = \frac{1}{k} \sum_{i=1}^{k} R_i, \text{ em que } k = |K| \tag{2.2}$$

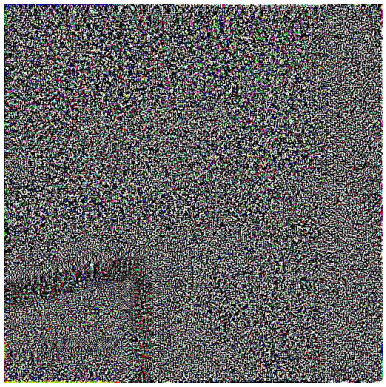
A utilização do ruído das imagens para o cálculo dessa aproximação é feita para reduzir a influência dos detalhes das cenas fotografadas (conteúdo), o que poderia não acontecer caso a aproximação fosse calculada com base na média do conteúdo das próprias imagens.

Por fim, é efetuada a correlação ρ_c entre o padrão residual R da imagem J que se deseja avaliar e o padrão de referência da câmera P_c.

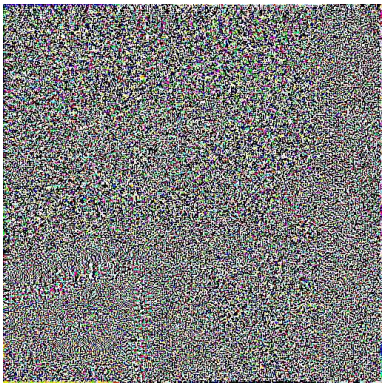
$$\rho_c(J) = \text{corr}(R, P_c) = \frac{(R - \bar{R}) \cdot (P_c - \bar{P}_c)}{||R - \bar{R}|| \cdot ||P_c - \bar{P}_c||} \tag{2.3}$$

em que a barra acima do símbolo denota o valor médio de *pixels*. Assim, um limiar T é calculado utilizando a abordagem de Neyman-Pearson, visando reduzir a taxa de falso-negativos. No momento que um valor de correlação ultrapassa um limiar determinado, os autores consideram que a imagem foi obtida pela câmera referente ao padrão usado naquela correlação. A Figura 2.5 ilustra a sequência de fases da metodologia proposta Lukáš et al. [30]. O trabalho apresenta resultados satisfatórios, chegando a 95% de acerto. Porém, foi reportada uma elevada taxa de falsos positivos, considerando um cenário com nove câmeras.

(a)



(b)



(c)

Figura 2.6: Exemplos de PRNU de uma imagem, onde (a) é a imagem original, (b) é o ruído residual obtido pela abordagem de Lukáš et al. [30] e (c) é o ruído residual aprimorado pela abordagem proposta por Li [26].

Uma melhoria dessa técnica é apresentada no trabalho de Li [26]. O autor assume que o ruído obtido com a técnica proposta por Lukáš et al. [30] sofre forte influência dos detalhes da cena (e.g., as altas frequências nas bordas dos objetos). Nesse contexto, o autor propõe a aplicação de modelos matemáticos de maneira a aprimorar o ruído residual obtido, diminuindo a interferência dos detalhes da cena. A Figura 2.6 ilustra os resultados obtidos com a extração do PRNU por meio da técnica proposta por Lukáš et al. [30] antes (Figura 2.6(b)) e após (Figura 2.6(c)) a aplicação do aprimoramento de ruídos proposta por Li [26]. É possível perceber claramente que o aprimoramento reduz a influência das bordas do objeto presente na cena. Apesar de o autor reportar um aumento da acurácia na atribuição de fonte, o método altera os valores dos pixels, inserindo assim artefatos de ruído na imagem que podem, em um conjunto de dados maior, dificultar a identificação da origem de imagens.

Segundo o trabalho de Li [26], dos modelos matemáticos propostos para melhoria de

ruído, o mais eficaz foi

$$n_e(i,j) = \begin{cases} e^{-0,5n^2(i,j)/\alpha^2}, & \text{se } 0 \leq n^2 \\ -e^{-0,5n^2(i,j)/\alpha^2}, & \text{caso contrário} \end{cases} \tag{2.4}$$

onde α é um valor definido pelo usuário, e o melhor valor encontrado foi $\alpha = 7$.

Técnicas que utilizam informações do ruído PRNU para identificação de um dispositivo específico podem não fornecer bons resultados caso as imagens sofram algum tipo de modificação de escala, rotação ou recorte, por exemplo. Essas operações introduzem distorções na imagem devido à reamostragem e podem provocar uma identificação incorreta do dispositivo por parte de um classificador. Nesse contexto, Goljan et al. [18] apresentam uma extensão do trabalho de Lukáš et al. [30], em que os autores consideram que as imagens sob investigação possam ter sofrido alguma operação de recorte e/ou escala. Primeiramente, os autores realizam uma etapa de busca exaustiva para a identificação dos parâmetros de escala da imagem analisada. Em seguida, são utilizadas métricas de estimativa de correlação de pico (*Peak to Correlation Energy* – PCE) e, então, é efetuada a correlação cruzada normalizada (*Normalized Cross Correlation* – NCC) entre os padrões de referência da câmera e da imagem redimensionada estimando, assim, os parâmetros de recorte. A NCC pode ser calculada utilizando a Transformada Rápida de Fourier (*Fast Fourier Transform* – FFT). O cálculo de NCC é mostrado na Equação 2.5, onde $\mathbf{X}$ e $\mathbf{Y}$ são os ruídos residuais, a barra sobre eles representa média aritmética dos pixels, $[i,j]$ representa o pixel na posição (i,j) dos ruídos e os valores u e v representa a translação do padrão de ruído sobre a imagem redimensionada.

$$\text{NCC}(u,v) = \frac{\sum_{i,j}(\mathbf{X}[i,j] - \bar{\mathbf{X}})(\mathbf{Y}[i+u, j+v] - \bar{\mathbf{Y}})}{\sqrt{\sum_{i,j}(\mathbf{X}[i,j] - \bar{\mathbf{X}})^2}\sqrt{\sum_{i,j}(\mathbf{Y}[i,j] - \bar{\mathbf{Y}})^2}}. \tag{2.5}$$

Os autores reportaram bons resultados para imagens com até 50% de redimensionamento e até 90% de área recortada. Uma desvantagem desse método é o alto custo computacional, tendo em vista que se utiliza um algoritmo de busca exaustiva para a localização dos parâmetros de escala. Para minimizar esse problema, os autores propõem a utilização de uma busca hierárquica ao invés da busca exaustiva, mas não desenvolvem a ideia nem apresentam resultados. Além do custo computacional, é importante notar que a qualidade da resposta depende, em parte, do conteúdo da imagem e do nível de compactação.

Os resultados apresentados em [30] e [18] foram confirmados por Goljan et al. em [17] por meio de um teste em larga escala, considerando 150 câmeras e aproximadamente 7.000 imagens por câmera. As imagens utilizadas nesse experimento foram obtidas no *Flickr*[1].

[1] `http://www.flickr.com`

Os autores consideram que uma imagem é autêntica (i.e., que não sofreu qualquer tipo de modificação) por meio de leitura do cabeçalho EXIF presente nas imagens. No entanto, a confiabilidade desse tipo de verificação é questionável, considerando que esse cabeçalho pode facilmente ser alterado ou removido.

Chen et al. [8] propõem a utilização da técnica apresentada por Lukáš et al. [30] para identificar a origem de vídeos suspeitos (atribuição de fonte em câmeras filmadoras). Segundo os autores, a técnica apresentada em [30] não pode ser aplicada diretamente para atribuição de fonte de vídeos devido ao fato de que sua resolução espacial geralmente é menor, comparando com a resolução de imagens, e cada *frame* do vídeo poder ter níveis de compressão diferentes. Primeiramente, os autores extraem o PRNU da câmera filmadora como apresentado em [30], tendo como vantagem a resolução temporal (considerando, por exemplo, que 1 segundo de vídeo equivale a, aproximadamente, 30 imagens). Em seguida, o PRNU da câmera é filtrado para eliminar artefatos gerados pela compressão do vídeo. Em posse de um vídeo suspeito, o PRNU desse vídeo é calculado da mesma maneira e, em seguida, é feita o cálculo da NCC (Equação 2.5) entre os PRNUs da câmera e do vídeo suspeito. Caso a NCC reporte um valor de energia de correlação alto (acima de um limiar estabelecido), os autores consideram que o vídeo foi gerado pela câmera filmadora avaliada. Nesse trabalho, é afirmado que 40 segundos de vídeo são suficientes para identificar se esse foi ou não gerado pela câmera suspeita. Entretanto, se o vídeo em questão possui baixa qualidade, é necessário mais tempo de vídeo para fazer essa identificação.

Sutcu et al. [43] propõem uma abordagem de classificação que combina características de padrão de ruído propostas por Lukáš et al. [30] e características obtidas de artefatos causados pelo processo de demosaico das imagens. Os autores consideram que informações referentes ao processo de interpolação de pixels são mais robustas e melhores para identificação da origem de imagens, em comparação com informações de padrão de ruído. Primeiramente, é feita a atribuição de fonte conforme proposto em [30]. Caso a classificação apresente valor positivo (classificação correta), os autores realizam, sobre as mesmas amostras, uma classificação SVM considerando características obtidas utilizando-se artefatos resultantes do processo de demosaico. Se a segunda classificação também for positiva, é considerado que a imagem avaliada foi gerada pela câmera sob investigação. A Figura 2.7 apresenta as etapas da abordagem proposta por Sutcu et al. [43].

Os autores reportam bons resultados em um cenário com três câmeras. O grande problema desse trabalho é a falta de detalhes, uma vez que os autores não apresentam informações sobre quais características referentes ao processo de demosaico da câmera são utilizadas e nem como elas são obtidas (e.g., algoritmo EM [34], métricas de qualidade de imagens [24], etc.). Além disso, um segundo teste seguindo o esquema proposto poderia invalidar uma resposta correta do primeiro classificador, reportando assim, ao final, uma

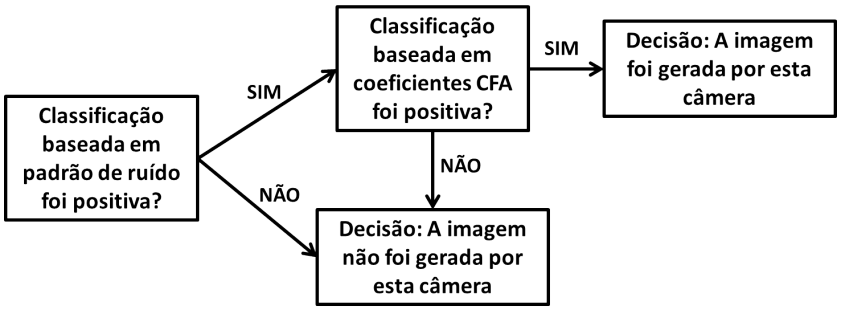


Figura 2.7: Processo de decisão combinada proposto por Sutcu et al [43].

identificação incorreta.

Em [15], Goljan e Fridrich consideram que o PRNU de um dispositivo pode ser extraído por meio de uma única imagem. Nesse trabalho, é apresentado um algoritmo para comparação e avaliação do PRNU de duas imagens visando verificar se ambas foram geradas pela mesma câmera. Essa técnica é denominada Correspondência entre Dispositivos (*Device Linking*). Os autores assumem que as imagens a serem avaliadas não sofreram qualquer tipo de manipulação digital além de recorte (operações como compressão com perdas e filtros são aceitáveis).

A extração do PRNU por meio de uma imagem é realizada como apresentada por Lukáš et al. [30] (Equação 2.1). O próximo passo consiste em comparar duas imagens baseadas em seus PRNU's. Essa comparação é feita por meio do cálculo da NCC (Equação 2.5). É esperado que imagens de uma mesma câmera possuam alto valor de correlação entre seus respectivos ruídos residuais, e imagens de câmeras diferentes possuam um valor baixo de correlação.

Com o NCC, é possível extrair várias métricas. Uma delas é a razão entre o pico primário e o pico secundário (*Primary and Secundary Peaks Ratio* – PSR), que pode ser selecionada para auxiliar na etapa de decisão. Um valor de PSR que está acima de um limiar (estabelecido para minimizar a taxa de falso-positivos na classificação) é um indicativo que ambas as imagens foram geradas pela mesma câmera. A Figura 2.8 ilustra a abordagem apresentada em [15].

A execução desse algoritmo exige que ambas as imagens possuam o mesmo tamanho para que sejam comparadas. Por essa razão, as bordas da menor imagem é preenchida com o valor 0 (quando necessário) antes de calcular a NCC (técnica denominada *zero padding*), conforme apresentada na Figura 2.9. De acordo com os autores, recortar uma região com um tamanho definido no centro de cada imagem é plausível.

Foram realizados experimentos com imagens com e sem compressão JPEG, utilizando-se 80 imagens provenientes de oito câmeras diferentes (10 imagens por dispositivo) e 560 pares de imagens (360 pares de imagens de mesma câmera e 200 pares de imagens

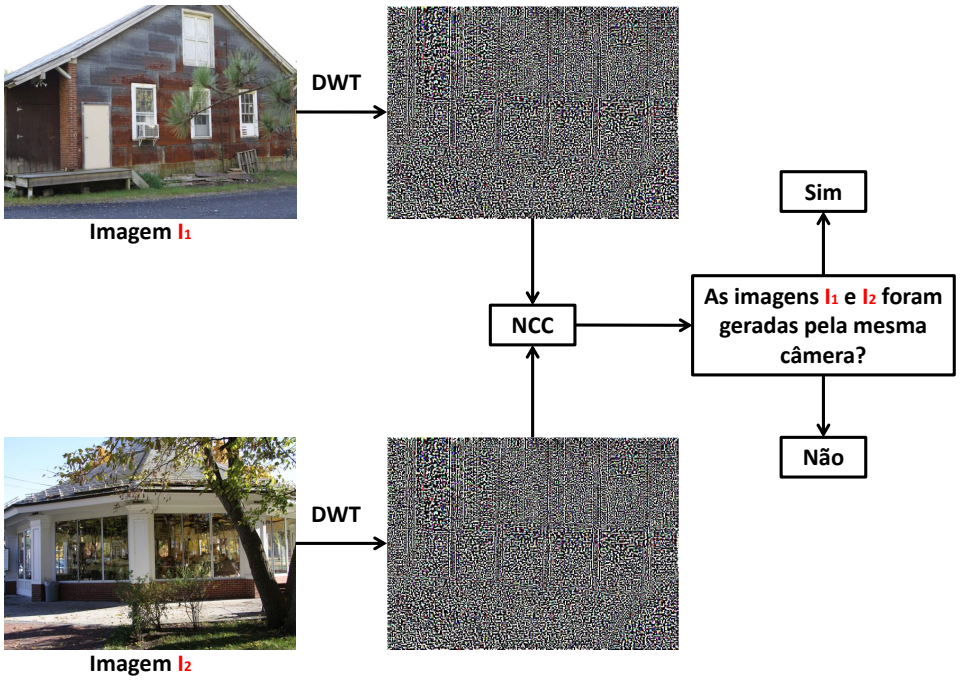


Figura 2.8: Algoritmo proposto por Goljan e Fridrich [15] para verificar se duas imagens foram geradas pela mesma câmera.

geradas por câmeras diferentes entre si). Considerando pares de imagens sem compressão JPEG, os autores reportam 100% de probabilidade de detecção de um par de imagens ser corretamente classificado como "imagens originadas pela mesma câmera" para quase todas as câmeras. Para os mesmos pares de imagens com compressão JPEG com fatores 90 e 75, os autores reportam um decréscimo na probabilidade de detecção correta (em média, 91% e 66%, respectivamente).

Embora o trabalho de Lukáš et al. [30] apresente resultados eficazes para a identificação de origem de imagens, Gloe et al. [13] propõem uma técnica contra-forense que dificulta essa identificação. Basicamente, a técnica proposta denominada *flatfielding* consiste em inserir informações de uma determinada câmera em uma imagem gerada por outra câmera. Para melhor compreensão, considere o seguinte cenário: Maria disponibilizou algumas fotos de sua câmera C_M na internet (redes sociais, em sua página pessoal, etc.). João obtém as fotos que Maria disponibilizou e, por meio delas, estima o padrão de ruído $\hat{K}$ da câmera C_M. Em seguida, João insere o ruído da câmera de Maria em uma imagem suspeita J' (e.g., pornografia infantil) gerada por uma câmera C_J. Assim, no momento da identificação (utilizando a técnica proposta por Lukáš et al. [30]), a origem da imagem gerada pela câmera C_J seria definida, equivocadamente, como sendo a câmera C_M.

Figura 2.9: Técnica de *Zero padding*.

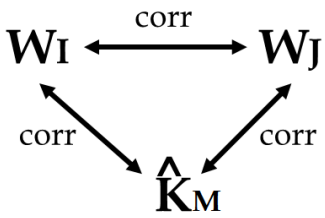


Figura 2.10: Teste triângulo (*Triangle Test*). FONTE: Adaptado a partir de Goljan et al. [16]

Os autores afirmam que um *flatfielding* considerado perfeito necessita, obviamente, de várias outras características além do PRNU, como tempo de exposição dos sensores à luz no momento da captura, velocidade da captura, algoritmo de geração da imagem, entre outros. Porém, os autores reportam resultados interessantes em experimentos considerando duas câmeras e analisando apenas o PRNU.

Considerando um cenário no qual um padrão de ruído de uma câmera pode ser "roubado" por meio de imagens disponíveis na internet e inseridos em uma imagem gerada por uma outra câmera, Goljan et al. [16] propuseram o Teste Triângulo (*Triangle Test*) como um meio de identificar esse tipo de ataque. Esse teste funciona da seguinte maneira: após ter suas imagens roubadas, Maria estima o padrão de ruído $\hat{K}_M$ de sua câmera utilizando imagens que, garantidamente, não foram utilizadas por João no momento do ataque (Maria pode, por exemplo, gerar novas imagens com sua câmera). Em seguida, Maria calcula a correlação $C_{I,J'} = corr(\mathbf{W}_I, \mathbf{W}_{J'})$, em que $\mathbf{W}_I$ é o ruído de uma das imagens a qual Maria desconfia que possa ter sido utilizada por João, e $\mathbf{W}_{J'}$ é o ruído extraído da imagem gerada por João. Além disso, Maria também calcula as correlações $C_{\hat{K}_M,J'} = corr(\hat{K}_M, \mathbf{W}_{J'})$ e $C_{I,\hat{K}_M} = corr(\mathbf{W}_I, \hat{K}_M)$. A Figura 2.10 mostra um diagrama dessas correlações.

Se a imagem I não foi utilizada na geração da imagem J', os valores de $C_{I,J'}$ podem ser estimados por $C_{\hat{K}_M,I}$ e $C_{\hat{K}_M,J'}$. Por outro lado, quando I foi utilizado para gerar a falsificação, o valor de $C_{I,J'}$ é maior do que estimado. O problema do Teste Triângulo é que seu sucesso depende do número de imagens utilizadas na falsificação. Os autores

reportam que, quanto maior o número de imagens utilizadas, pior é o resultado.

Embora técnicas contra-forense sejam importantes para a área de atribuição de fontes, nós não consideramos a existência de tais técnicas neste trabalho.

A Tabela 2.1 apresenta um comparativo de todas as técnicas descritas nesta seção. A principal características de tais técnicas é o fato de trabalharem em um cenário fechado (*closed-set*), ou seja, os pesquisadores consideram que uma imagem foi gerada necessariamente por uma das câmeras disponíveis para experimentos na etapa de treinamento do sistema. Acreditamos que o problema de atribuição de fonte possa ser abordado em um ambiente mais realístico, no qual uma imagem, na etapa de testes, possa ter sido gerada por qualquer dispositivo, seja ele com acesso disponível ou não. Esse cenário é denominado cenário aberto (*open-set*), e será descrito a seguir.

Autor	Abordagem	Pontos Positivos	Pontos negativos
Kharrazi et al. [24]	Definição de 34 características para identificação do modelo de dispositivos	Bom índice discriminatório de classificação	Dificuldade na identificação de dispositivos específicos
Popescu e Farid [34]	Algoritmo EM para estimar componentes CFA da câmera nas imagens	Autores reportam bons resultados	Dificuldade na identificação de dispositivos específicos
Kurosawa et al. [25]	Identificação da origem de imagens a partir de ruído de padrão fixo causado por *dark currents*	É o primeiro trabalho voltado para identificação de dispositivos específicos	Câmeras mais modernas destroem o ruído de padrão fixo no momento da captura
Dirik et al. [11]	Identifica artefatos causados por poeira nas lentes das câmeras no momento da captura	Resultados interessantes em um cenário com duas câmeras	Artefatos podem ser facilmente destruídos
Geradts et al. [12]	Análise de defeitos dos componentes como uma possível forma de relacionar uma imagem à sua fonte geradora.	Câmeras com defeitos gerarão imagens com artefatos característicos (um ponto preto em uma certa posição, por exemplo)	Técnica limitada, pois nem todas as câmeras possuem esse tipo de imperfeição
Lukáš et al [30]	Estimativa de padrão de ruído não uniforme para identificar a origem de imagens	Técnicas que utilizam padrão de ruído atualmente são as mais promissoras para identificar câmeras específicas	Alta taxa de falso-positivos
Li [26]	Aplicação de modelos matemáticos de maneira a aprimorar o ruído residual obtido, diminuindo a interferência dos detalhes da cena.	Autor reporta aumento na acurácia de classificação	O método altera os valores dos pixels, inserindo assim artefatos na imagem que podem, em um conjunto de dados maior, dificultar a identificação de sua origem
Goljan et al. [18]	Extensão do trabalho de Lukáš et al [30], na qual os autores consideram que as imagens sob investigação possam ter sofrido alguma operação de recorte e/ou escala.	Bons resultados para imagens com até 50% de redimensionamento e até 90% de área recortada.	Alto custo computacional (busca exaustiva); a qualidade da resposta depende, em parte, do conteúdo da imagem e do nível de compactação

Chen et al. [7]	Utilização da técnica apresentada por Lukáš et al. [30] para identificar a origem de vídeos suspeitos (atribuição de fonte em câmeras filmadoras)	Os autores afirmam que 40 segundos de vídeo são suficientes para identificar se este foi ou não gerado pela câmera suspeita	A qualidade da técnica depende diretamente da qualidade do vídeo
Sutcu et al. [43]	Utilização da técnica apresentada por Lukáš et al. [30] em conjunto com características do algoritmo de demosaico das câmeras para identificação do dispositivo específico	Proposta de união de classificadores para atribuição de fonte	O resultado do primeiro teste (com ruído) pode ser anulado pelo resultado do segundo (demosaico)
Goljan e Fridrich [15]	Comparação e avaliação do PRNU de duas imagens visando verificar se ambas foram geradas pela mesma câmera	Resultados com alta acurácia	Poucos detalhes de implementação; autores realizam *zero padding* para imagens com resolução diferente entre si
Gloe et al. [13]	Técnica contra-forense denominada *flatfielding*, que consiste em inserir informações de uma determinada câmera em uma imagem gerada por outra câmera	Resultados interessantes em um cenário com duas câmeras	Um *flatfielding* considerado perfeito necessita de várias outras características além do PRNU
Goljan et al. [16]	Identificação de ataque com *flatfielding* por meio de Teste Triângulo	Bons resultados reportados	O sucesso da abordagem depende do número de imagens utilizadas na falsificação

Tabela 2.1: Comparativo entre técnicas do estado da arte no domínio da atribuição de fonte de imagens.

2.3 Cenário aberto (*open-set*)

Em classificação de padrões, muitas vezes não precisamos ou não temos acesso ao conhecimento de todas as classes que possam ser consideradas. Por exemplo, em um problema de reconhecimento de espécies de peixes, uma única espécie pode ser considerada como classe de interesse. Porém, um classificador deve considerar o conjunto de todas as outras espécies que tenham características relevantes como classe negativa. Em muitos casos, a modelagem de toda a classe negativa se torna inviável ou impossível (por exemplo, modelar todas as outras espécies existentes de peixe).

Para muitos problemas na área de visão computacional, pesquisadores assumem que possuem acesso a todas as classes possíveis e que uma determinada amostra pertence a uma de duas classes: positiva ($+1$) ou negativa (-1). Porém, esse não é um cenário tão próximo da realidade. Segundo Zhou e Huang [48], "as amostras da classe positiva são parecidas entre si, porém amostras de diferentes classes negativas são negativas à sua maneira".

Uma abordagem alternativa consideraria um cenário aberto, em que amostras de algumas classes seriam consideradas tanto no treino quanto no teste, porém nem todas as amostra a serem classificadas são pertencentes a essas classes. Isso otimiza a solução das classes desconhecidas, bem como das classes conhecidas. A Figura 2.11 mostra um exemplo de classificação em um cenário aberto.

A classificação em cenário *open-set* tem recebido um tratamento limitado na literatura. Em um estudo de métodos de avaliação para reconhecimento de faces apresentado por Phillips et al. [33], um típico arcabouço para reconhecimento em cenário aberto é proposto. Nesse trabalho, os autores definem um limiar em que todas as identificações de face devem ultrapassar esse valor para serem consideradas corretas. Um sistema incorporado a esse limiar não aceita uma amostra com valor de reconhecimento alto como acerto caso a classe dessa amostra seja desconhecida no momento do treinamento. Obviamente, a escolha do limiar depende dos requerimentos do sistema de reconhecimento e do seu ambiente de operação.

Em atribuição de fontes, Wang et al. [46] realizam a identificação do modelo de câmera que gerou uma determinada imagem por meio da estimativa de coeficientes de CFA, como apresentado na Seção 2.2.1 e utilizando uma combinação de dois classificadores SVM: classificação com duas classes (*Two-class SVM* – TC-SVM) [10] e classificação com uma classe (*One-class SVM* – OC-SVM) [39]. Na classificação com OC-SVM, são utilizadas somente duas das 17 câmeras disponíveis para a definição de *outliers*. O trabalho reporta resultados médios próximos de 91%. A desvantagem dessa abordagem é que, em se tratando de coeficientes de CFA, somente a marca ou modelo da câmera pode ser reconhecida. Caso os autores procurassem identificar o dispositivo específico,

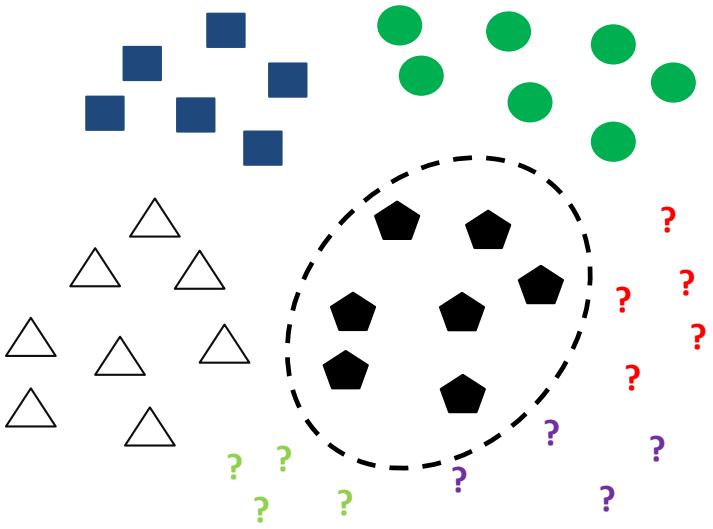


Figura 2.11: Exemplo de classificação em cenário aberto. A abordagem em cenário aberto assume que nem todas as classes são conhecidas *a priori*. A figura mostra a classe de interesse ("pentágono"), bem como as classes negativas conhecidas ("triângulo", "quadrado"e "círculo") e desconhecidas ("?").

imagens geradas por câmeras distintas mas de mesmo modelo poderiam ser classificadas incorretamente.

Li [27] e Caldelli et al. [5] propuseram diferentes abordagens para separar um conjunto em imagens de acordo com suas respectivas fontes. Em ambos os trabalhos é considerado que informações sobre as câmeras que geraram tais imagens não são acessíveis. Embora os autores apliquem em suas abordagens classificação não-supervisionada para a definição dos grupos de imagens, os trabalhos não são considerados como uma classificação em cenário aberto, uma vez que é utilizado um conjunto fechado de câmeras nos experimentos, em que uma imagem necessariamente foi gerada por uma câmera pertencente a esse conjunto (isto é, não é considerada qualquer classe desconhecida durante o treinamento de suas abordagens).

Considerações finais

A área de pesquisa denominada Análise Forense de Documentos Digitais é relativamente nova e tem como objetivo a verificação de autenticidade e integridade de documentos digitais.

Identificar a câmera que gerou uma determinada imagem é uma maneira de determinar se tal imagem foi ou não adulterada. Várias pesquisas focam nesse tipo de trabalho, tanto

para a identificação da marca/modelo da câmera utilizada no momento da captura de uma imagem como na identificação do dispositivo específico utilizado.

As técnicas discutidas trabalham em um cenário fechado, ou seja, os pesquisadores consideram que uma imagem foi gerada necessariamente por uma das câmeras disponíveis para experimentos na etapa de treinamento do sistema. No que tange o problema de atribuição de fontes, acreditamos que a classificação em cenário aberto é uma abordagem que melhor representa um cenário real pois, na prática, uma imagem a ser classificada pode ter sido gerada por qualquer dispositivo e não somente por algum dispositivo ao qual se tem acesso.

Capítulo 3

Atribuição de fonte em cenário aberto

Como dito anteriormente, identificar a origem de documentos digitais é uma tarefa importante no cenário forense, pois pode evidenciar, por exemplo, que uma imagem suspeita não é resultado de manipulação digital. Geralmente, as técnicas de identificação da origem do documento são abordagens voltadas para a identificação das características do dispositivo de captura de um objeto, como defeitos de fabricação, interação do sensor com a luz, artefatos gerados por poeira nas lentes, entre outros.

Atualmente, as técnicas mais efetivas para a identificação do dispositivo de captura específico analisam os efeitos do ruído inserido no processo de captura de imagens. As abordagens apresentadas por Lukáš et al. [30], assim como a melhoria proposta por Li [26] permitiram o desenvolvimento de outras abordagens baseadas em seus conceitos, como abordagens que visam identificar uma fonte comum entre pares de imagens [15], clusterização de imagens [5, 27] e algumas técnicas contra-forense para o problema de atribuição de fonte [4, 13].

Neste capítulo, destacamos as desvantagens de se utilizar a identificação da origem de uma imagem em um cenário fechado e apresentamos nossa abordagem para realizar a atribuição de fonte de imagens em um cenário mais próximo do real (cenário aberto). Utilizamos como base para nossa abordagem o trabalho de Lukas et al [30], o qual visa identificar a câmera geradora de uma imagem por meio de padrão de ruído (PRNU).

3.1 Abordagem Proposta

Embora a abordagem de Lukáš et al. [30] seja eficaz para a identificação da câmera que gerou uma imagem sob investigação, para a estimativa do limiar na etapa de treinamento, os autores assumem que possuem amostras de todas as classes (câmeras) disponíveis, e

assim classificam as imagens como sendo da classe positiva (isto é, se a imagem foi gerada pela câmera sob investigação) ou negativa (caso contrário). Essa abordagem pode não ser eficaz se precisarmos analisar imagens geradas por uma câmera a qual não temos acesso na etapa de treinamento. Trabalhando em um cenário aberto, em que apenas algumas câmeras são consideradas no momento do treinamento, mas nem todas as imagens a serem avaliadas na etapa de teste foram geradas por uma dessas câmeras, acreditamos que a definição de características e a utilização de técnicas de aprendizado de máquina que possam gerar hiperplanos não-lineares que separem as classes positiva e negativa possam ser mais eficazes na classificação de imagens geradas por câmeras desconhecidas, bem como as geradas por câmeras disponíveis no treinamento.

Com base nisso, a abordagem proposta neste trabalho consiste basicamente em três passos:

A. Definição de regiões de interesse;

B. Definição de características;

C. Atribuição de fonte em um cenário aberto.

A. Definição de regiões de interesse (ROIs) : Lukáš et al. [30] consideram em seus experimentos uma região central da imagem para determinar sua origem. Li [26] também considera uma região central da imagem e, em alguns experimentos, toda a imagem. Entretanto, Li [26] realiza seus experimentos em um cenário com seis câmeras de mesma resolução nativa (ou seja, todas as imagens utilizadas pelo autor possuem as mesmas dimensões). Quando se tem imagens com diferentes tamanhos, considerar uma região comum em todas as imagens (por exemplo, a região central) pode ser mais apropriado para atribuição de fontes de imagem.

De acordo com [28], diferentes regiões da imagem podem ter diferentes informações sobre o padrão de ruído da câmera. Neste trabalho para atribuição de fontes em um cenário aberto, visamos considerar várias regiões de uma imagem, e não somente a região central, como é feito em [30, 26]. Para cada imagem, foram extraídas nove regiões de interesse (*Regions of Interest* – ROI) de tamanho 512×512 *pixels*, numeradas de acordo com a Figura 3.1.

Para as ROIs 1-5 (no centro da imagem), assumimos que essas regiões coincidem com o eixo principal das lentes e podem possuir mais detalhes da cena, pois fotógrafos amadores geralmente focam o objeto de interesse da cena no centro das lentes. Consequentemente, essas regiões tendem a possuir mais riquezas de detalhes e, talvez, mais ruídos. As ROIs 6-9 (nos cantos da imagem) também são importantes, pois algumas câmeras deixam nas imagens artefatos causados por *vignetting*, uma queda radial de intensidade partindo

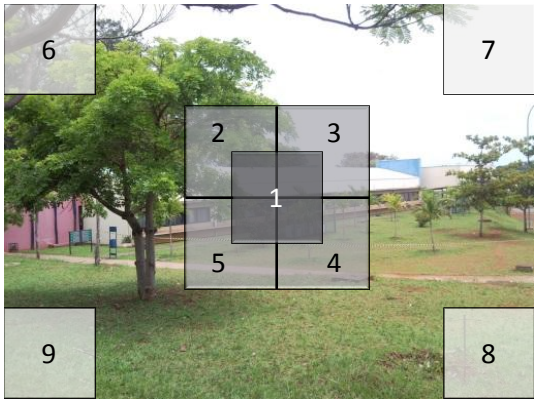


Figura 3.1: Regiões de interesse (ROIs) de dimensão 512×512 *pixels*.

(a) (b)

Figura 3.2: Exemplos de imagens sem (a) e com (b) artefato de *vignetting*.

do centro da imagem, causando uma redução de brilho/saturação nas periferias das imagens [14], como mostrado na Figura 3.2.

B. Definição de características : Para cada região apresentada na Figura 3.1, nós calculamos o padrão de ruído conforme discutido em [30], utilizando o filtro para extração de ruído no domínio Wavelet proposto em [32]. Lukáš et al. calculam o padrão de ruído considerando imagens em tons de cinza, mas essa abordagem pode trivialmente ser expandida para outros espaços de cores. Neste trabalho, consideramos a utilização dos canais R (vermelho), G (verde), B (azul). Foram realizados experimentos com outros espaços de cores e notamos que, para o problema de atribuição de fonte, a utilização do canal Y (luminância, do espaço de cor YCbCr [47]) em conjunto com os canais R, G e B se mostrou eficaz.

Para cada ROI, extraímos o ruído residual de cada canal de cor utilizando um filtro baseado na DWT. Em seguida, calculamos a média entre os ruídos de mesmo canal de

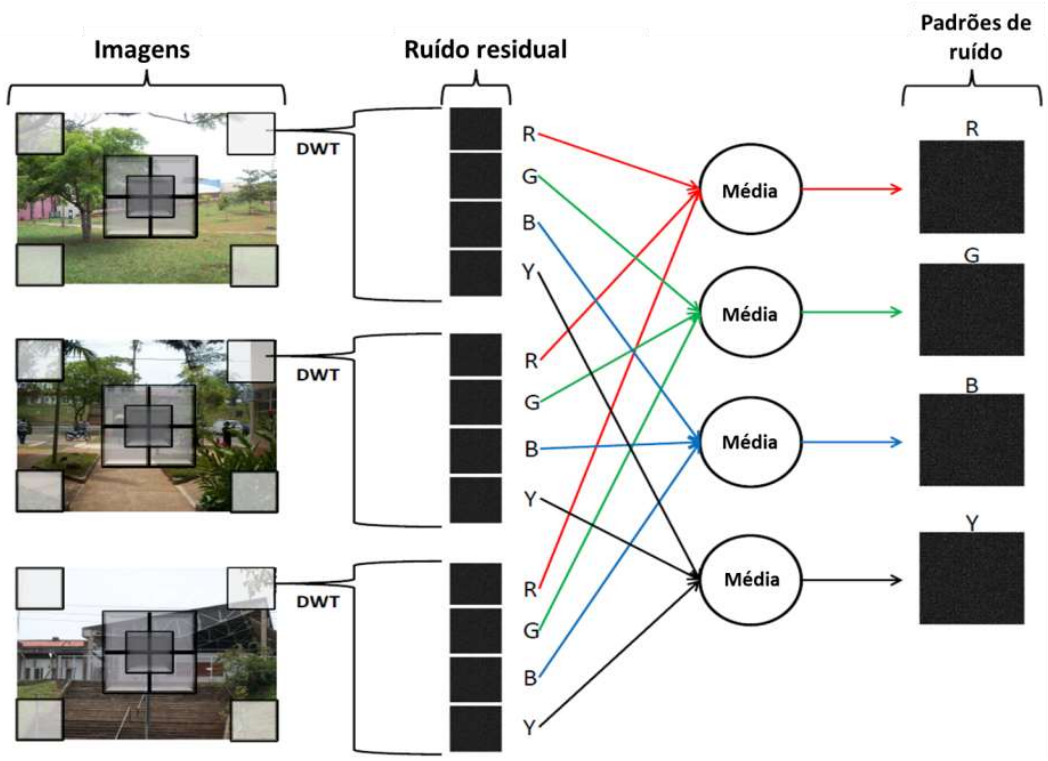


Figura 3.3: Calculando padrão de ruído para uma região, considerando os canais de cores R, G, B e Y. O processo é realizado para todas as nove regiões marcadas na imagem.

várias imagens, gerando o padrão de ruído para cada canal de cor que representa a câmera sob investigação. Com isso, temos 36 padrões de ruído para representar uma câmera, conforme mostrado na Figura 3.3). É importante notar que esse tipo de caracterização por regiões nos permite comparar imagens de diferentes dimensões sem nos preocupar com artefatos de interpolação de cores.

Para cada imagem, calculamos seu ruído residual e criamos um vetor de características considerando a correlação entre cada ROI de uma imagem e o padrão de referência correspondente para cada câmera. Com essas correlações, temos 36 características para cada imagem, considerando uma câmera, rotulando imagens geradas pela câmera sob investigação como a classe positiva e as câmeras restantes disponíveis como classe negativa. Observe que algumas dessas imagens serão consideradas como sendo pertencentes à classe negativa desconhecida, ou seja, são imagens geradas por câmeras às quais não temos acesso na etapa de treinamento.

C: Atribuição de fonte em um cenário aberto A principal contribuição deste trabalho é a utilização de aprendizado de máquina visando fazer a identificação da origem de imagens em um cenário aberto. Para resolver esse problema, primeiramente

encontramos um classificador para treinar um conjunto de amostras considerando a classe de interesse e outras classes as quais temos acesso. Formalmente, dado um conjunto de treinamento $(\mathbf{x}_i, y_i)$ para $i = 1 \ldots N$, com $\mathbf{x}_i \in \Re^d$ e $y_i \in \{-1, 1\}$, precisamos encontrar um classificador $f(\mathbf{x})$ tal que

$$f(\mathbf{x}_i) = \begin{cases} \geq 0, & y_i = +1 \\ < 0, & y_i = -1 \end{cases} \tag{3.1}$$

Seja $\mathbf{X}$ nossa matriz de treinamento a qual a n-ésima linha de $\mathbf{X}$ corresponde ao vetor de características $\mathbf{x}_i^T$. Considere que a classe positiva é representada pelos vetores de características $\mathcal{P} = \{\mathbf{x}_1^p, \mathbf{x}_2^p, \ldots \mathbf{x}_{n_{pos}}^p\}$ e a(s) classe(s) negativa(s) consistem de $\mathcal{K} = \{\mathbf{x}_1^k, \mathbf{x}_2^k, \ldots \mathbf{x}_{n_{neg}}^k\}$ em que $N = n_{pos} + n_{neg}$ é o número total de amostras de treinamento e $\mathbf{X} = \mathcal{P} \bigcup \mathcal{K}$. Podemos encontrar o melhor hiperplano de separação $w^T\mathbf{x} + b = 0$ (caso linear) ou $w^T\phi(\mathbf{x}) + b = 0$ (caso não linear) por meio do clássico algoritmo de Máquina de Vetores de Suporte (*Support Vector Machine* – SVM) [10, 2] que visa encontrar um classificador capaz fazer a separação entre os dados de $\mathcal{P}$ e de $\mathcal{K}$, onde $\mathbf{w}$ é a normal do hiperplano, b é o viés do hiperplano tal que $|b|/||\mathbf{w}||$ é a distância perpendicular da origem do hiperplano e ϕ é a função de mapeamento do espaço original das características para um alto espaço dimensional por meio de uma função *kernel* [2].

Após encontrarmos a maior margem de separação do hiperplano (classificador $f(\cdot)$) entre os dados de treinamento X, nós temos a situação como descrita (somente para o caso linear) na Figura 3.4 em que temos uma classe de interesse como a classe positiva (consistindo em dados de uma câmera sob investigação) e somente uma classe negativa (consistindo de dados proveniente de outras câmeras conhecidas). De acordo com esse modelo, cada ponto $\mathbf{x}_i$ durante o treinamento está a uma distância d_i^m do hiperplano de decisão dado pelo modelo SVM e pode ser classificado como classe $+1$ se $w^T\mathbf{x_i} + b \geq 0$ (caso linear) ou como classe -1, caso contrário.

SVM utiliza minimização de risco estrutural (*Structural Risk Minimization* – SRM) [2] que é um princípio indutivo para seleção de modelo para aprendizado por um conjunto de treinamento finito para resolver o problema de encontrar a maior margem de separação entre duas classes. Entretanto, pode-se notar que o SVM pode minimizar o risco somente no caso baseado em que o classificador "conhece" os dados de treinamento. No caso da classificação em cenário aberto, muito mais classes podem aparecer como sendo classe negativa, o que poderia atrapalhar a operação do classificador durante os testes.

Portanto, neste trabalho nós definimos uma política de minimização do risco para a classe desconhecida para o cenário aberto pela minimização do erro de dados $\mathcal{D}$ durante o treinamento após o cálculo do hiperplano de separação pelo SVM. O erro de dados $\mathcal{D}$ é definido como a inversa da acurácia normalizada da classificação $A(\mathbf{X})$ durante o

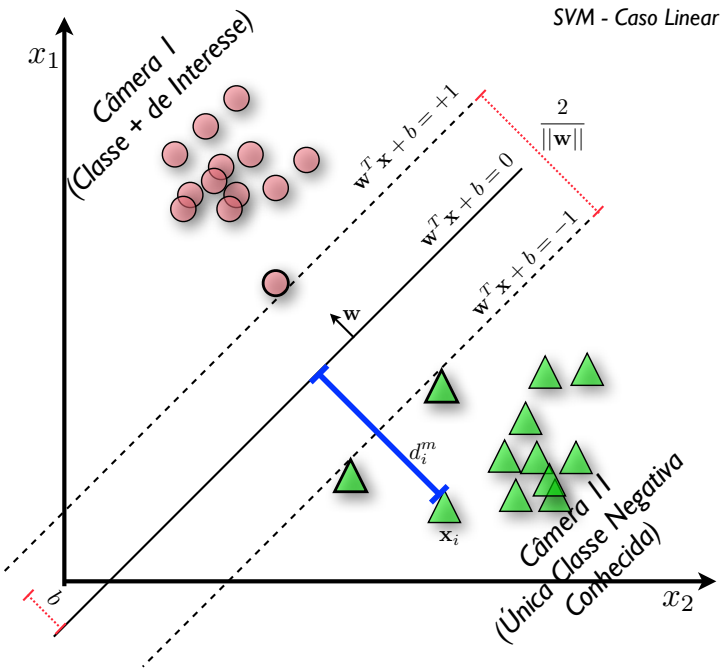


Figura 3.4: Exemplo de um classificador SVM considerando o caso linear.

treinamento

$$A(\mathbf{X}) = \frac{\left(\frac{\sum_{i=1}^{npos}\theta(\mathbf{x_i^p})}{n_{pos}} + \frac{\sum_{j=1}^{nneg}\omega(\mathbf{x_j^k})}{n_{neg}}\right)}{2} \tag{3.2}$$

onde

$$\theta(\mathbf{x_i^p}) = \begin{cases} 1, & \text{se } f(\mathbf{x_i^p}) \geq \mathbf{0} \\ 0, & \text{caso contrário.} \end{cases} ; \tag{3.3}$$

$$\omega(\mathbf{x_j^k}) = \begin{cases} 1, & \text{se } f(\mathbf{x_j^k}) < \mathbf{0} \\ 0, & \text{caso contrário.} \end{cases} \tag{3.4}$$

Para isto, definimos o hiperplano utilizando como *kernel* a função de base radial (*Radial Basis Function* – RBF).

$$\text{RBF}(\mathbf{x_i^p}, \mathbf{x_j^k}) = \exp\left(-\gamma \cdot |\mathbf{x_i^p} - \mathbf{x_j^k}|^{\mathbf{2}}\right) \tag{3.5}$$

A escolha do *kernel* RBF se deve ao fato de que este apresentou melhores resultados durante os experimentos.

Realizamos uma busca em grade (*Grid Search* [20]), visando encontrar os melhores valores de custo (punição para cada erro de classificação) e o parâmetro γ da função RBF (que define a suavidade do hiperplano de separação entre as classes positiva e negativa).

Assim, escolhemos o hiperplano que melhor separa as amostras negativas das amostras positivas durante a etapa de treinamento, definindo o melhor modelo de classificação encontrado neste passo.

A Equação 3.2 mostra que os valores de classificação de todas as amostras de treinamento são analisados para encontrar a acurácia de classificação $A(\mathbf{X})$. Considerando o hiperplano calculado na etapa de treinamento, propomos um meio de considerar as classes desconhecidas na classificação por meio da movimentação do hiperplano de decisão por um valor ε se aproximando da classe positiva ou se afastando da(s) classe(s) negativa(s). A lógica é que, movendo o hiperplano, podemos ser mais restritos para o que nós sabemos como amostras positivas e, portanto, classificar qualquer outra amostra "muito diferente" como negativa, ou podemos ser pouco rigorosos sobre o que sabemos em relação às amostras positivas e aceitar pontos mais distantes do hiperplano como possíveis amostras positivas. Como um primeiro passo visando resolver o problema de atribuição de fontes em um cenário aberto, consideramos ε para mover o plano em um intervalo dado pela amostra "mais positiva" (isto é, pela amostra positiva mais distante do hiperplano de separação) e a amostra "mais negativa" (ou seja, a amostra negativa mais distante do hiperplano de separação).

O valor de ε representa o movimento do hiperplano de decisão $w^T\mathbf{x} + b + \varepsilon = 0$ (caso linear) ou $w^T\phi(\mathbf{x}) + b + \varepsilon = 0$ (caso não-linear). Esse processo é definido como *Decision Boundary Carving* (DBC). O valor de ε é definido a partir de uma busca exaustiva visando minimizar o erro de treinamento $\frac{1}{A(\mathbf{X})}$. Assim sendo, após a movimentação do plano, a classificação definida pelas Equações 3.3 e 3.4 são modificadas para atender a nova posição do plano.

$$\theta'(\mathbf{x_i^p}, \varepsilon) = \begin{cases} 1 & \text{se } f(\mathbf{x_i^p}) \geq \varepsilon \\ 0 & \text{caso contrário.} \end{cases} \tag{3.6}$$

$$\omega'(\mathbf{x_j^n}, \varepsilon) = \begin{cases} 1 & \text{se } f(\mathbf{x_j^n}) < \varepsilon \\ 0 & \text{caso contrário.} \end{cases} \tag{3.7}$$

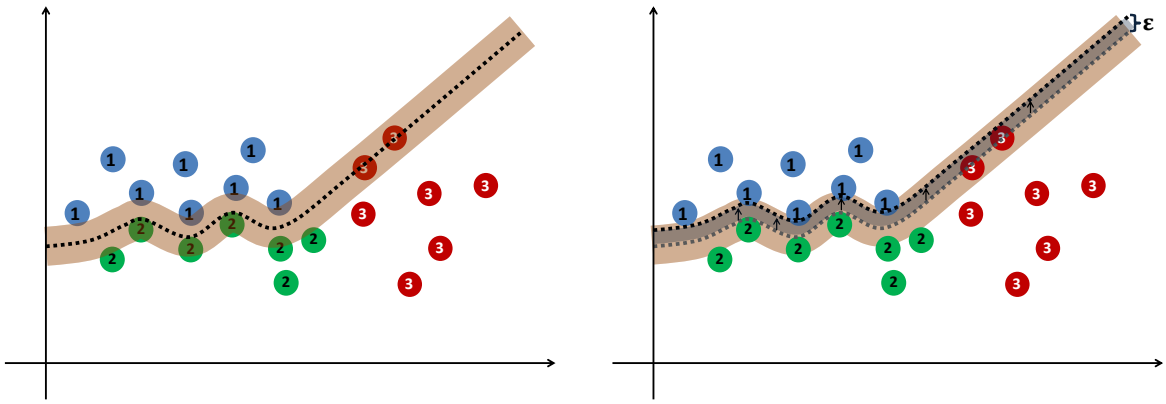
A Figura 3.5 descreve um exemplo para o caso não-linear. A Figura 3.5(a) apresenta o hiperplano de separação calculado, considerando amostras das classes azul e verde como as classes de interesse (1) e a classe negativa conhecida (2), respectivamente, e as amostras em vermelho representam as classes desconhecidas (3). A região laranja representa a distância entre as margens dos vetores de suporte das classes positiva e negativa. A Figura 3.5(b) mostra a operação de DBC sobre o hiperplano calculado, representado pela região azul. Note que o processo de movimentação do hiperplano de decisão procura a minimização do risco da classe desconhecida por meio da minimização de $\mathcal{D}$. Dado qualquer amostra $\mathbf{z}$ durante o teste, esta é classificada como uma amostra da classe positiva se $f(\mathbf{z}) \geq \varepsilon$. O pseudo-código do processo de *Decision Boundary Carving* é apresentado no Algoritmo 1.

Algoritmo 1: Decision Boundary Carving (DBC)

Entrada: $\mathcal{P}, \mathcal{K}$
//$\mathcal{P}$ = Classe de interesse
//$\mathcal{K}$ = Classe negativa conhecida
Saída: H, ε

$\mathrm{H} \leftarrow \mathrm{MelhorHiperplano}(\mathcal{P}, \mathcal{K})$;
//Encontra o melhor hiperplano na etapa de treinamento
$\mathbf{C} \leftarrow$ Classificação $(\mathrm{H}, \mathcal{P}, \mathcal{K})$;
//Encontra os valores de classificação de todas as amostras de treinamento
$\mathrm{MIN} \leftarrow \mathrm{menor}(\mathbf{C})$;
$\mathrm{MAX} \leftarrow \mathrm{maior}(\mathbf{C})$;
Para $\mathrm{n} \leftarrow \mathrm{MIN}$ até MAX **faça**
  $\mathrm{AccPos} \leftarrow 0$
  $\mathrm{AccNeg} \leftarrow 0$
  Para todo $\mathbf{x^p} \in \mathcal{P}$ **faça**
    $\mathrm{AccPos} \leftarrow \mathrm{AccPos} + \theta'(\mathbf{x^p}, \mathrm{n})$;
  Fim para
  Para todo $\mathbf{x^k} \in \mathcal{K}$ **faça**
    $\mathrm{AccNeg} \leftarrow \mathrm{AccNeg} + \omega'(\mathbf{x^k}, \mathrm{n})$;
  Fim para
  $\mathrm{Acc} \leftarrow \mathrm{média}(\mathrm{AccPos}, \mathrm{AccNeg})$;
  $\mathcal{D}' \leftarrow 1/\mathrm{Acc}$;
  Se $\mathcal{D}' < \mathcal{D}$
    $\mathcal{D} \leftarrow \mathcal{D}'$;
    $\varepsilon \leftarrow \mathrm{n}$;
  Fim se
Fim Para

Nós também tentamos fazer a atribuição de fontes considerando um classificador SVM de uma classe (*One-class SVM* – OC-SVM), proposta inicialmente por Schölkopf [39] e que foi adaptada para a classificação SVM em um cenário aberto. Com a ausência da classe negativa no conjunto de treino (ou seja, considerando somente amostras da classe de interesse), a origem definida pela função *kernel* do classificador representa amostras de uma "segunda classe". O objetivo então se torna encontrar a melhor margem que respeite a origem. Como resultado, o classificador a função f retorna, após o treino, o valor $+1$ (classe positiva) para a região que contempla a maioria das amostras de treinamento e -1 na região que não contempla tais amostras. O classificador define como classe negativa o ponto 0, que representa a origem da função *kernel* do classificador, e como classe positiva a região delimitada acima da origem. Assim, podemos traçar um hiperplano H (b) que



(a) (b)

Figura 3.5: Nossa implementação do cenário aberto para atribuição de fonte de imagens utilizando *Decision Boundary Carving* (DBC).

separa a classe de interesse, como mostrado na Figura 3.6.

A fim de determinar a eficácia da classificação utilizando o hiperplano H gerado, realizamos uma classificação considerando as mesmas amostras da classe $\mathcal{P}$ e amostras das outras classes conhecidas $\mathcal{K}$ (classe negativa conhecida). A acurácia normalizada da classificação é calculada e o processo se repete, gerando um novo hiperplano, de forma a maximizar a acurácia relativa, até que seja encontrado o melhor hiperplano de separação ao final do processo.

Como na classificação de uma classe, o hiperplano é gerado considerando somente a classe positiva, pode ocorrer de amostras das classes negativas estarem na região definida para a classe positiva na etapa de testes, sendo classificadas incorretamente como amostras *positivas*. Assim sendo, definimos dois hiperplanos H_1 e H_2 de forma a definir um intervalo entre os hiperplanos onde a classe positiva estará contida, conforme a Figura 3.7. Feito isso, para cada hiperplano é feito o processo de DBC de modo a identificar a melhor posição de ambos os hiperplanos, considerando as amostras de treinamento utilizadas para a geração do hiperplano H e algumas amostras da classe negativa. São feitos os processo de generalização (aumentando a área entre os hiperplanos H_1 e H_2 de forma a considerar mais amostras da classe negativa durante o treinamento) e especialização (diminuindo tal área, tornando o classificador mais restrito em relação à classe positiva). Esses processos são mostrados na Figura 3.8.

Na etapa de testes, uma amostra é classificada como *positiva* se e somente se, após a classificação, a amostra estiver contida entre os hiperplanos H_1 e H_2, definidos na etapa de treinamento. Essa abordagem pode ser bastante útil em um cenário no qual possuímos

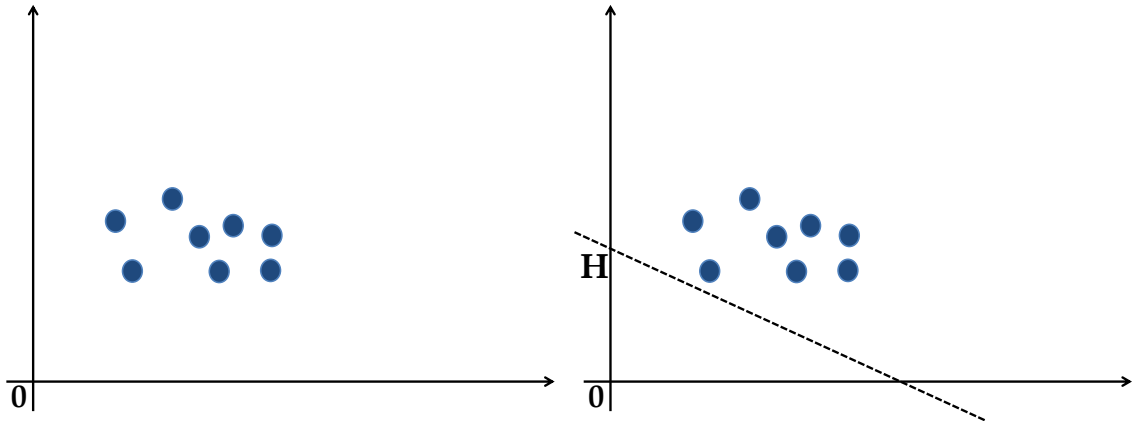


Figura 3.6: Geração de um hiperplano para OC-SVM.

acesso a poucas câmeras (ou somente uma), podendo utilizar imagens as quais se temos certeza de que não foram geradas pela câmera de interesse para definir as posições dos hiperplanos H_1 e H_2 e realizar a exclusão de amostras anômalas (*outliers*).

Além do *kernel* linear (como mostrado na Figura 3.7), também realizamos experimentos utilizando *kernel* RBF para classificação com OC-SVM, sendo necessária a regulagem de seus parâmetros $\gamma \in (0, \infty]$ (que define o quão suave é a borda do hiperplano) e $\nu \in (0, 1)$ (que define um limite superior para a porcentagem de elementos que possam ser classificados incorretamente). Vale lembrar que valores altos para γ forçam um *overfitting* (treinamento viciado) na classificação, enquanto valores altos para ν desconsideram muitos exemplos da classe positiva. Portanto, esses valores são regulados por meio de uma busca em grade, de modo a definir um hiperplano que classifique bem amostras de ambas as classes. A Figura 3.9 mostra a variação desses valores na geração de um hiperplano de decisão em 2D.

A eficácia do hiperplano gerado é calculada da mesma maneira que é feita com o *kernel* linear, conforme as Equações 3.6 e 3.7 para um conjunto de validação com amostras das classes positivas e negativas. Após encontrar o melhor hiperplano H definido pela função ϕ, utilizamos o DBC para movimentar o plano a uma distância ε, de modo que a classificação $C(x)$ ocorra da seguinte maneira:

$$C(x) = \begin{cases} 1 & \text{se } \phi(x) \geq \varepsilon \\ -1 & \text{caso contrário.} \end{cases} \tag{3.8}$$

Essa movimentação é apresentada na Figura 3.10.

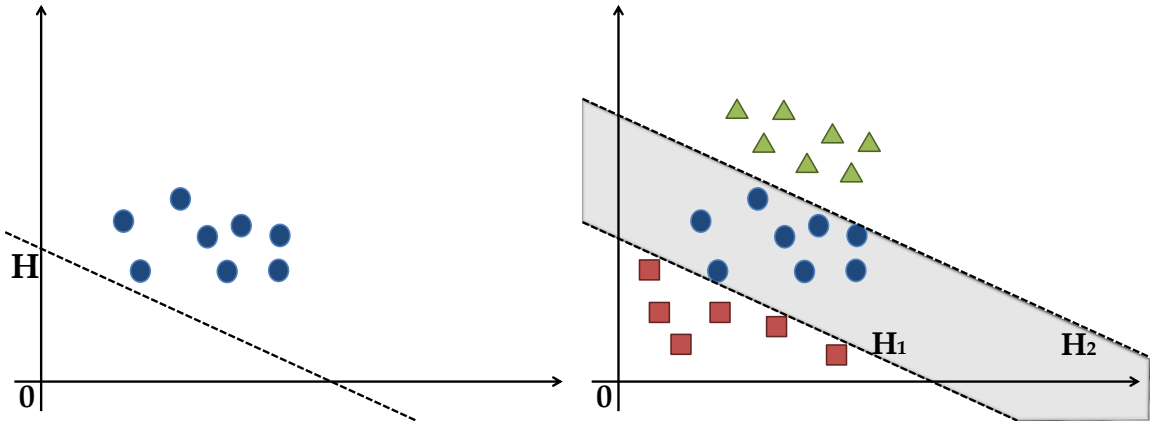
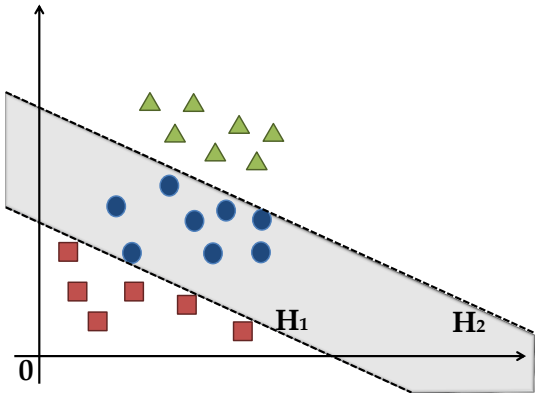


Figura 3.7: Definição dos hiperplanos de decisão para OC-SVM. O OC-SVM necessita somente da classe positiva ("círculo") para ser definido (a). Porém, como neste caso conhecemos outras duas classes negativas ("quadrado" e "triângulo"), podemos utilizá-las para melhor definirmos a posição dos hiperplanos de separação (b).

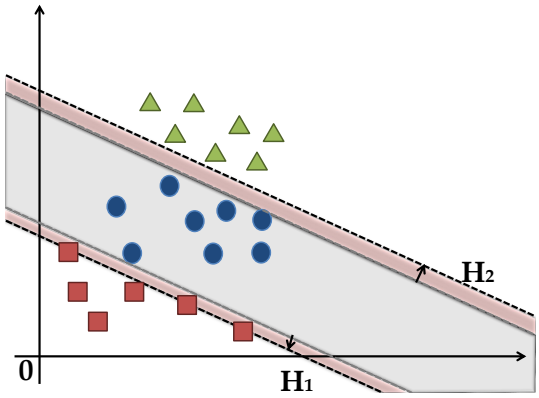
Considerações finais

Como dito anteriormente, a maior contribuição deste trabalho é a realização da atribuição de fontes de imagens em um cenário aberto. Para definir a margem de separação entre a classe de interesse e as demais classes de forma automática, nós utilizamos abordagens baseadas em aprendizado de máquina.

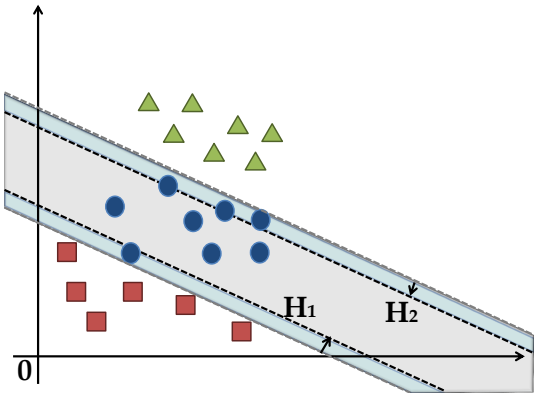
A principal vantagem do conjunto de características escolhido é justamente poder realizar a atribuição de fonte sem se preocupar com imagens de dimensões diferentes. Tanto o *Two-class SVM* (TC-SVM) quanto o *One-Class SVM* (OC-SVM) apresentaram resultados interessantes para a identificação de origem de imagens. Os resultados obtidos por ambas as técnicas serão discutidos no Capítulo 5.



(a) Hiperplanos de decisão

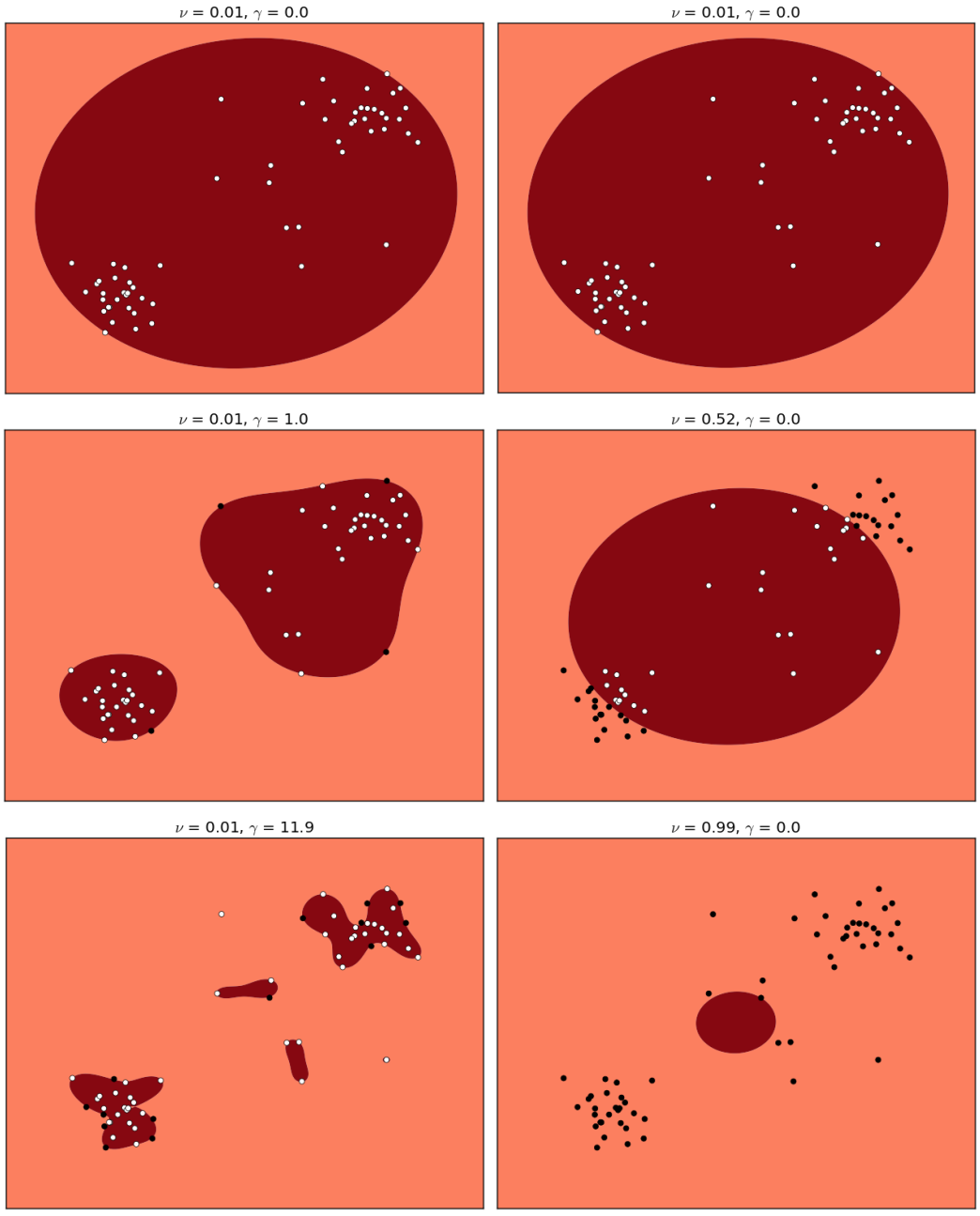


(b)



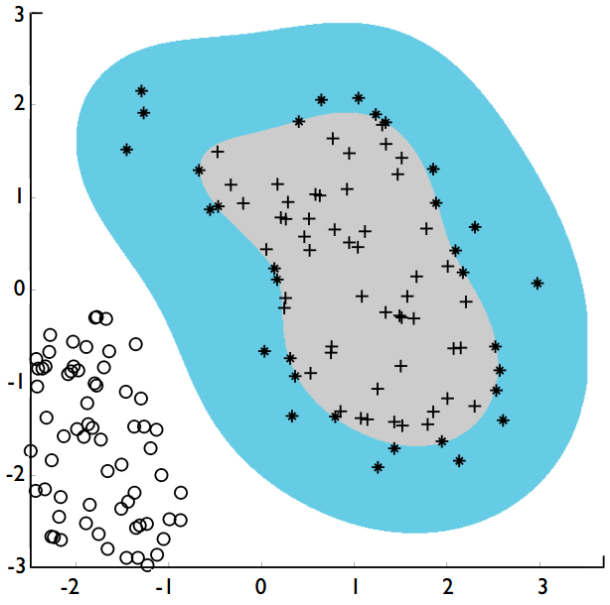
(c)

Figura 3.8: Processo de generalização (b) e especialização (c) para OC-SVM. As classes negativas aqui mostradas são apenas para definir os hiperplanos. O exemplo representa o caso do OC-SVM linear.

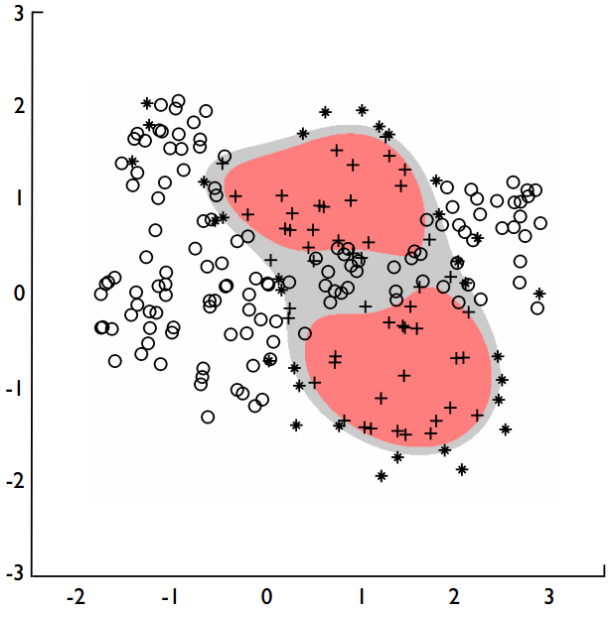


Variando valores de γ Variando valores de ν

Figura 3.9: Variação dos parâmetros γ (0.0, 1.0 e 11.9) e ν (0.01, 0.52 e 0.99) na geração de hiperplano utilizando OC-SVM com núcleo RBF.



(a)



(b)

Figura 3.10: Exemplo de generalização (a) e especialização (b) de margens para OC-SVM. Em azul está representada a generalização da margem; em vermelho, a especialização; em cinza, o hiperplano inicialmente gerado para OC-SVM, com $\nu = 0.25$ e $\gamma = 0.25$. A classe de interesse é representada por $+$.

Capítulo 4

Correspondência entre Dispositivos (*Device Linking*)

Existem situações em que identificar a origem de um conjunto de imagens se torna uma tarefa complicada, principalmente quando não existe qualquer informação sobre o dispositivo que possivelmente tenha gerado tais imagens. Uma possível solução inicial para um analista forense que tenha acesso a esse conjunto de imagens suspeitas é identificar primeiramente se todas foram geradas pelo mesmo dispositivo. Fazendo isso, ele pode primeiramente separar este conjunto de imagens em subconjuntos, conforme a fonte comum das imagens.

Identificar se duas imagens foram geradas pela mesma câmera pode ser útil, por exemplo, para localizar câmeras roubadas. Existem *websites* (e.g., *Stolen Camera Finder* [1], *Camera Trace* [2]) que analisam as informações contidas no cabeçalho EXIF de uma imagem e, com base nessas informações, realizam uma busca em redes sociais como *Facebook* ou *Flickr* para verificar se outras imagens geradas pela mesma câmera foram postadas em alguns desses sites. Com isso, é possível localizar o autor das novas postagens e, muitas vezes, recuperar o dispositivo roubado.

O grande problema desse tipo de solução é justamente a utilização do cabeçalho EXIF, que pode ser facilmente removido ou alterado, o que pode dificultar (ou até mesmo invalidar) essa solução. Uma possível solução é utilizar alguma característica mais robusta do dispositivo de origem ao invés do cabeçalho EXIF. O principal trabalho nesta linha de pesquisa foi proposto por Goljan e Fridrich [15] (descrito na seção 2.2.2), em que os autores verificam se um par de imagens foi gerado pela mesma câmera com base no ruído residual das imagens em questão.

Esse método foi implementado como *baseline* deste trabalho considerado o conjunto

[1] *http://www.stolencamerafinder.com – Último acesso em 1 de Junho de 2012.*

[2] *http://cameratrace.com – Último acesso em 1 de Junho de 2012.*

de imagens descrito no Capítulo 5. Consideramos somente a região central de tamanho 512×512 pixels de cada imagem. No primeiro experimento, examinamos o quão eficaz era o limiar de separação encontrado pelos autores quando aplicado em nosso conjunto de imagens. Foi encontrada uma acurácia de 50%. Em um segundo experimento, nós calculamos o melhor limiar considerando os valores de PSR encontrado na etapa de treinamento e realizamos a etapa de testes considerando esse limiar. Também foi realizada validação cruzada nos conjuntos de treino e teste. O valor médio de acurácia encontrado foi de 51%. Uma possível razão para tais resultados tão divergentes é o fato de não termos efetuado o processo de *zero padding* quando consideramos duas imagens de resoluções diferentes.

Neste capítulo, apresentamos uma proposta para identificar se duas imagens foram geradas pela mesma câmera.

4.1 Abordagem Proposta

Primeiramente, extraímos o ruído residual de vários pares de imagens geradas pelas mesmas câmeras e vários pares de imagens geradas por câmeras diferentes e treinamos um classificador. Fundamentalmente, o método se trata de uma extensão da abordagem apresentada em [15] e, de alguma forma, é fortemente inspirado em alguns trabalhos na área de atribuição de fonte, tais como [24, 43]. Nessas abordagens, os autores adotaram classificação supervisionada como a principal estratégia para determinar que uma câmera sob investigação gerou uma imagem específica. Na realidade, o objetivo principal desta abordagem não é apenas melhorar a técnica de correspondência entre dispositivos proposta em [15], mas sim fornecer um mecanismo automático para calcular um limiar (não-linear) para diferenciar pares de imagens gerados pela mesma câmera de pares de imagens gerados por câmeras diferentes. Assim, esse limiar é calculado por um classificador SVM.

A primeira etapa da abordagem proposta neste trabalho é a extração do PRNU das imagens, similar à técnica apresentada em [15] com a exceção de que utilizamos o filtro de extração de ruído baseado em DWT proposto em [32]. Primeiramente, selecionamos 9 regiões de interesse (ROIs) para cada imagem, conforme apresentado anteriormente na Figura 3.1 (pag. 28). Em seguida, extraímos os ruídos considerando diferentes canais de cor, como foi feito na atribuição de fonte. Para este caso, os canais de cores que se mostraram mais eficazes foram somente os canais R, G e B. Assim, são extraídos 27 ruídos residuais por imagem. Como dito anteriormente, extrair o ruído das imagens considerando ROIs exclui a necessidade de as imagens possuírem a mesma resolução. Feito isso, para cada ROI correspondente entre duas imagens, calculamos a correlação entre os ruídos de acordo com a Equação 4.1. Essa medida é similar à NCC, porém os resultados são calculados de forma mais rápida.

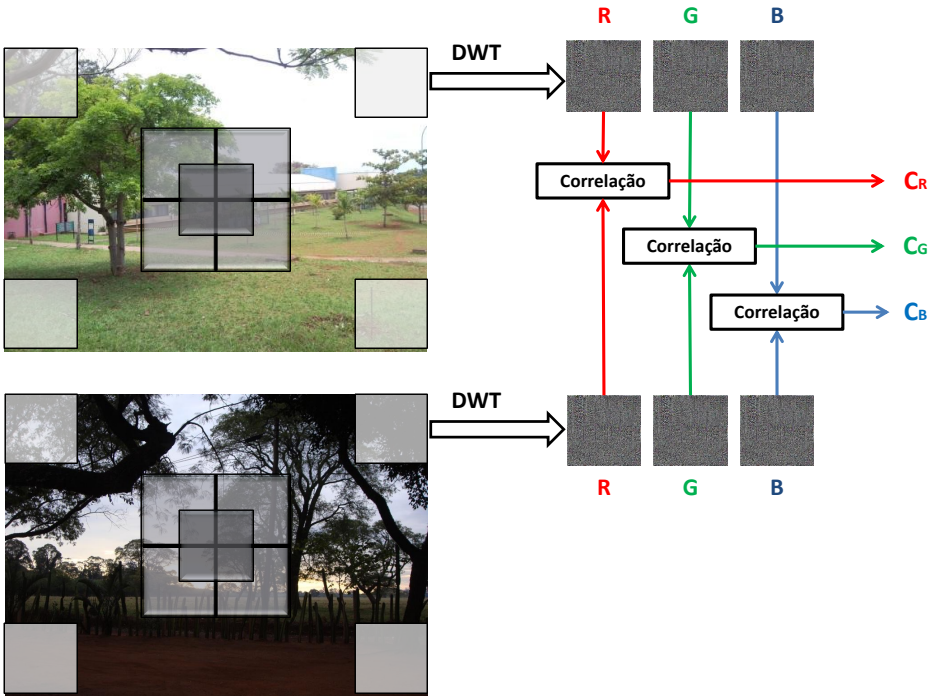


Figura 4.1: Extração de características para correspondência entre dispositivos. O processo é realizado para todas as nove regiões marcadas na imagem.

$$\mathrm{CRL} = \frac{\sum_{i,j}(\mathbf{X}[i,j] - \bar{\mathbf{X}})(\mathbf{Y}[i,j] - \bar{\mathbf{Y}})}{\sqrt{\sum_{i,j}(\mathbf{X}[i,j] - \bar{\mathbf{X}})^2}\sqrt{\sum_{i,j}(\mathbf{Y}[i,j] - \bar{\mathbf{Y}})^2}} \tag{4.1}$$

Para cada região temos os valores C_R, C_G e C_B dadas pela correlação entre ruídos de duas imagens considerando as bandas R, G e B, respectivamente. Dessa forma, temos no total 27 características que representam um par de imagens. Pares de imagens geradas pela mesma câmera tendem a ter um valor de correlação mais alto em relação aos pares de imagens geradas por câmeras diferentes. Como o cálculo dessa função de correlação requer que ambas as imagens tenham a mesma dimensão, acreditamos que esta abordagem possa substituir a abordagem anterior apresentada em [15], onde era realizado um preenchimento com valores zero nas imagens para que essas pudessem ter a mesma dimensão antes do calculo da correlação, pois o cálculo da correlação após esse preenchimento poderia produzir artefatos indesejados. A Figura 4.1 descreve a extração de características proposta neste trabalho para correspondência entre dispositivos.

A etapa final consiste em treinar um classificador TC-SVM com todas as informações obtidas, e encontrar automaticamente um limiar de decisão não-linear (um hiperplano)

Algoritmo 2: Correspondência entre dispositivos

Entrada: Conjunto de pares de imagens C_I
Saída: $\mathcal{H}, \varepsilon$

Para cada par de imagens $(I_i, I_j) \in C_I$ **faça**
  Ruído$_i \leftarrow$ DWT (I_i)
  Ruído$_j \leftarrow$ DWT (I_j)
  $\mathbf{v} \leftarrow \text{Corr}(I_i, I_j)$
  //Considerando cada ROI e canal de cor correspondente
  Se (I_i, I_j) foram geradas pela mesma câmera
    Adiciona ($\mathbf{v}$, M)
    //M = Mesma câmera
  Senão
    Adiciona ($\mathbf{v}$, D)
    //D = Câmeras diferentes
  Fim Se
Fim para
$[\mathcal{H}, \varepsilon] \leftarrow$ DBC (M , D)

sobre todos os pares na etapa de treinamento e, portanto, sobre as classes *positiva* (imagens geradas pela mesma câmera) e *negativa* (caso contrário). Também é realizado a técnica DBC para encontrar a melhor posição do hiperplano de decisão na etapa de treinamento, considerando *kernel* RBF e busca em grade e visando maximizar a acurácia normalizada, similar à técnica proposta para o problema de atribuição de fonte. O pseudo-código da abordagem proposta para resolver o problema de correspondência entre dispositivos é apresentado no Algoritmo 2.

Considerações finais

Neste capítulo, apresentamos uma abordagem para identificar se duas imagens foram geradas por uma mesma câmera, tarefa que pode ser útil, por exemplo, para identificar se fotos de uma câmera roubada foram postadas na internet por meio de imagens as quais se tem certeza que foram geradas por tal câmera.

A abordagem proposta é baseada no trabalho de Goljan e Fridrich. [15]. A metodologia utilizada nos testes desta abordagem, bem como os resultados obtidos (comparando-os com os reportados em [15]) serão discutidos no Capítulo 5.

Capítulo 5

Experimentos e validação

Neste capítulo, apresentamos a base de dados utilizada em ambas as técnicas propostas, bem como a metodologia utilizada na realização de experimentos e os resultados obtidos com cada técnica, comparados com as abordagens da literatura.

5.1 Conjunto de dados

Foi gerado um conjunto de 8500 imagens a partir de 35 câmeras digitais de marcas e modelos diferentes (no mínimo, 150 imagens por câmera)[1]. Todas as imagens foram geradas na resolução e qualidade de compressão JPEG nativas das câmeras, sendo imagens com resolução mínima de três megapixels (dimensão de 1600 × 1200 pixels) e resolução máxima de 15.1 megapixels (dimensão de 4752 × 3168), totalizando 30.4 GB de espaço em disco.

As imagens foram obtidas considerando diferentes condições de iluminação (ambientes claros e escuros, internos e externos, com *flash* e sem *flash*), e diversas configurações de *zoom* óptico e foco (objeto aproximado, afastado, foco no centro da câmera, imagens de horizontes, etc.). Na geração das imagens não utilizamos *zoom* digital. Todas as imagens foram obtidas com a câmera na horizontal e não sofreram qualquer processo de rotação e escala, bem como qualquer operação de pós-processamento após sua geração (correção de brilho, contraste, etc.).

A Tabela 5.1 mostra os modelos de câmera utilizados nos experimentos, enquanto a Figura 5.1 mostra alguns exemplos de imagens utilizadas.

[1] O conjunto de dados está disponível em `http://www.ic.unicamp.br/~rocha/pub/communications.html`.

	Câmera	Resolução Nativa
1	**Canon PowerShot SX1-LS**	3840×2160
2	**Kodak EasyShare c743**	3072×2304
3	**Sony Cybershot DSC-H55**	4320×3240
4	**Sony Cybershot DSC-S730**	2592×1944
5	**Sony Cybershot DSC-W50**	2816×2112
6	**Sony Cybershot DSC-W125**	3072×2304
7	**Samsung Omnia**	2560×1920
8	**Apple iPhone 4**	2592×1936
9	**Kodak EasyShare M340**	3664×2748
10	**Sony Cybershot DSC-H20**	3648×2736
11	**HP PhotoSmart R727**	2048×2144
12	**Canon EOS 50d**	4752×3168
13	**Kodak EasyShare Z981**	4288×3216
14	**Nikon D40**	3008×2000
15	**Olympus SP570UZ**	3968×2976
16	**Panasonic Lumix DMC-FZ35**	4000×3000
17	**Sony Alpha DSLR A500L**	4272×2848
18	**Olympus Camedia D395**	2048×1536
19	**Sony Cybershot DSC-W120**	3072×2304
20	**Nikon Collpix S8100**	4000×3000
21	**Sony Cybershot DSC-W330**	4320×3240
22	**Apple iPhone 4**	2592×1936
23	**Cannon Powershot A520**	1600×1200
24	**Apple iPhone 3**	1600×1200
25	**Samsung Star**	2048×1536
26	**Olympus SP 800UZ**	4288×3216
27	**Nikon d5000**	4288×2848
28	**Panasonic Lumix DMC-FZ35**	4000×3000
29	**Sony Alpha DSLR A500**	4272×2400
30	**Kodak EasyShare Z981**	4288×3216
31	**Nikon Coolpix P100**	3648×2736
32	**Canon PowerShot G12**	3648×2736
33	**Canon EOS 50d**	4752×3168
34	**Sony Cybershot DSC HX1**	3456×2592
35	**Olympus E30**	4032×3024

Tabela 5.1: Câmeras utilizadas em nossos experimentos.

5.2 Experimentos e Resultados

5.2.1 Atribuição de fontes em cenário aberto

Para os experimentos em atribuição de fonte, separamos as imagens do conjunto de dados em cinco grupos de forma randômica, para aplicar a técnica de validação cruzada (*5-fold cross validation* [2]). Para cada execução, utilizamos três desses conjuntos para a geração dos padrões de referência de cada câmera, um conjunto para o treinamento do classificador SVM (considerando somente imagens provenientes das câmeras consideradas disponíveis para treinamento) e o último conjunto para a realização de testes e coleta de resultados (considerando imagens proveniente de todas as câmeras). O processo é repetido cinco vezes, apenas alternando os conjuntos em cada etapa.

O problema de atribuição de fontes em um cenário aberto foi analisado considerando

Figura 5.1: Exemplos de imagens que compõem nosso conjunto de dados.

que, em cada execução, temos acesso a 15, 10, 5 e 2 câmeras suspeitas, e que as imagens podem ter sido geradas por qualquer uma das 35 câmeras apresentadas na Tabela 5.1. Nesses cenários, consideramos que nunca temos acesso às câmeras 16–35. No primeiro caso, consideramos que as câmeras 1–15 são acessíveis, sendo estas utilizadas no treinamento do classificador. Dois experimentos considerando 10 câmeras disponíveis foram executados (câmeras 1–10 e câmeras 6–15). Para os experimentos com cinco câmeras disponíveis, nós consideramos três combinações diferentes com cinco câmeras (1–5; 6–10; 11–15). Por fim, os experimentos com duas câmeras foram executados considerando sete combinações de duas câmeras (1–2,3–4, e assim por diante até o conjunto 13–14).

A identificação da câmera geradora de uma imagem digital foi realizada seguindo dois tipos de análise: verificação e reconhecimento.

Verificação

Na análise de verificação, o objetivo é determinar se uma imagem foi ou não adquirida por uma determinada câmera. A Figura 5.2 ilustra a forma que é realizada esse tipo

de análise. A grande dificuldade nesse caso é a modelagem da classe negativa, isto é, a definição de como serão representados todos os dispositivos existentes exceto a câmera avaliada. A princípio, procuramos resolver esse problema visando modelar a classe que representa as imagens geradas pelo dispositivo em questão e verificar se a imagem avaliada se assemelha com as imagens pertencentes a essa classe.

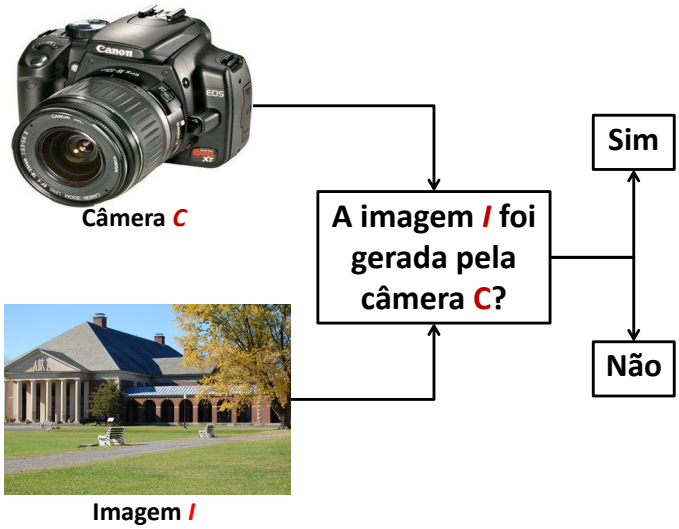


Figura 5.2: Identificação do dispositivo específico — *Verificação*.

A análise de verificação é feita da seguinte maneira: para cada câmera disponível, o modelo de hiperplano é definido por meio da classificação SVM (OC ou TC-SVM). Dada uma amostra de teste, esta é classificada como *positiva* ou *negativa* de acordo com esse modelo. Assim, calculamos a acurácia para cada câmera. Geralmente, pesquisadores consideram, em seus resultados, o cálculo da acurácia absoluta de classificação (ou seja, a porcentagem de acerto total). Devido ao número desbalanceado de amostras entre as classes *positiva* e *negativa* em cada teste, a acurácia para cada câmera é calculada considerando a acurácia relativa de classificação.

$$Acc_{Rel} = \frac{Acc_P + Acc_N}{2}, \tag{5.1}$$

em que Acc_P e Acc_N são as porcentagens de acerto na classificação para as classes positiva e negativa, respectivamente. Vale lembrar que o conjunto de testes contém amostras das classes positiva e negativa (sem sobreposição de amostras utilizadas no treinamento), além de amostras de imagens cujas câmeras geradoras não estão disponíveis durante a etapa de treinamento (as câmeras geradoras de tais imagens são desconhecidas e elas não estão disponíveis como câmeras suspeitas durante uma investigação). Note que as imagens

geradas por câmeras desconhecidas devem ser classificadas como pertencentes à classe negativa no momento do teste.

Em seguida, calculamos a acurácia media Acc_M como

$$Acc_M = \frac{1}{z}\sum_{i=1}^{z} Acc_R^i, \tag{5.2}$$

em que $z = 5$ é o número de *folds* para a validação cruzada.

O resultado reportados neste trabalho corresponde à acurácia final Acc_F, calculada como a média das acurácias obtidas por cada câmera.

$$Acc_F = \frac{1}{n}\sum_{i=1}^{n} Acc_M^i, \tag{5.3}$$

em que n é o número de câmeras disponíveis durante o treinamento do classificador.

A Tabela 5.2 apresenta os resultados obtidos com a abordagem proposta neste trabalho, comparando-os com os resultados obtidos pela implementação do método de Lukáš et al. [30] e da extensão proposta por Li [26] aplicados em cenário aberto, considerando a análise de verificação e utilizando o conjunto de dados proposto. Foram feitos experimentos considerando somente a ROI central (ROI #1) e considerando todas as ROIs, tanto para TC-SVM quanto para OC-SVM. Na utilização do TC-SVM, foram feitos experimentos com e sem a aplicação da técnica de DBC. Para o OC-SVM, foram feitos experimentos considerando tanto *kernel* linear quanto *kernel* RBF.

A Tabela 5.2 mostra uma melhoria significativa na acurácia geral quando comparando o método TC-SVM proposto por este trabalho e as abordagens de Lukáš et al. [30] e Li [26]. Já os experimentos com OC-SVM se mostraram inferiores, considerando tanto um classificador linear quanto um classificador RBF. É possível notar que tanto a utilização de características extraídas de outras ROIs além da ROI central em ambos os casos quanto a aplicação do DBC na classificação TC-SVM se mostraram positivas nos experimentos.

Outras formas de avaliação foram executadas, como por exemplo, a definição de dois hiperplanos para o TC-SVM, bem como a utilização de *kernel* linear para o mesmo. Essas abordagens apresentaram resultados piores, comparados aos resultados reportados, por isso não foram considerados na Tabela 5.2.

A Tabela 5.3 apresenta resultados para experimentos considerando que temos somente duas câmeras disponíveis de mesma marca/modelo (os pares de câmeras {(8,22),(12,33),(16,28)}), mas uma imagem de teste pode ter sido gerada por qualquer uma das 35 câmeras. Para calcular os resultados, obtemos a média dos cinco testes para cada par, obtendo um resultado para cada par de dispositivos e, por fim, calculamos a média desses valores. Nesse caso, consideramos a utilização de todas as ROIs da imagem.

	Nº de câmeras no treinamento			
	15	**10**	**5**	**2**
Lukáš et al. [30]	93,9 ± 2,03	93,2 ± 2,05	93,8 ± 2,16	93,2 ± 2,61
Li [26]	93,4 ± 2,27	92,9 ± 2,36	93,3 ± 2,39	92,2 ± 3,05
TC-SVM Somente ROI #1	91,1 ± 3,13	91,1 ± 2,60	93,2 ± 2,68	93,8 ± 2,60
TC-SVM Somente ROI #1 + DBC	95,1 ± 1,51	94,5 ± 1,77	94,5 ± 1,61	93,8 ± 2,11
TC-SVM Todas as ROIs	96,0 ± 1,61	96,2 ± 1,63	96,2 ± 1,73	**96,0 ± 1,75**
TC-SVM Todas as ROIs + DBC	**97,2 ± 1,52**	**96,9 ± 1,44**	**96,5 ± 1,40**	95,1 ± 2,02
OC-SVM Somente ROI #1 + OC-SVM linear	90,6 ± 5,74	90,0 ± 5,83	88,0 ± 6,25	86,9 ± 8,30
OC-SVM Somente ROI #1 + OC-SVM RBF	90,6 ± 5,74	90,0 ± 5,83	88,0 ± 6,25	86,9 ± 8,30
OC-SVM Todas as ROIs + OC-SVM linear	92,1 ± 1,15	92,1 ± 0,47	92,1 ± 0,89	91,8 ± 2,76
OC-SVM Todas as ROIs + OC-SVM RBF	89,1 ± 1,15	89,1 ± 0,47	89,1 ± 0,89	88,6 ± 2,76

Tabela 5.2: Resultados ($ACC_F \pm$ desvio padrão, em (%)), para 15, 10, 5 e 2 câmeras disponíveis durante o treinamento na análise de verificação. O cenário considera 35 câmeras no total.

A Tabela 5.3 mostra que a abordagem proposta neste trabalho é eficaz em cenários em que possuímos câmeras de mesma marca e modelo, identificando com alta acurácia o dispositivo específico. É possível notar que as abordagens de Lukáš et al. [30] e Li [26] apresentaram resultados semelhantes para esse caso. Podemos notar também pelas Tabelas 5.2 e 5.3 que a técnica DBC para TC-SVM não é eficaz quando temos acesso a somente duas câmeras suspeitas, mas que pode ser útil quando temos acesso a mais câmeras.

A Tabela 5.4 mostra uma comparação de resultados para os cenários fechado (F) e aberto (A) com 15 câmeras, em que o cenário fechado com 15 câmeras consiste em realizar o treinamento com 15 câmeras e testar o classificador utilizando somente imagens proveniente dessas 15 câmeras. São apresentadas as taxas de falso-positivos e falso-negativos (%). Note que os resultados apresentam similaridade entre os cenários aberto e fechado. O método proposto utilizando TC-SVM mostra melhor desempenho em ambos os casos, em comparação com o OC-SVM. Neste experimento, também consideramos a utilização de todas as ROIs obtidas na imagem.

Os resultados obtidos mostram que é possível identificar a origem de imagens em um

Método	Média ± Desvio Padrão
Lukáš et al. [30]	93,1 ± 2,94
Li [26]	92,1 ± 3,08
TC-SVM	**95,7 ± 1,61**
TC-SVM (DBC)	94,4 ± 1,33
OC-SVM (Linear)	91,9 ± 2,70
OC-SVM (RBF)	88,6 ± 3,17

Tabela 5.3: Resultados (em %) considerando duas câmeras de mesma marca/modelo como classes conhecidas no treinamento. O cenário considera 35 câmeras no total na etapa de teste.

Método	TP (%)	TN (%)
Lukáš et al. [30] (F)	92,59 ± 5,15	96,83 ± 2,06
Lukáš et al. [30] (A)	92,59 ± 5,15	95,37 ± 2,73
Li [26] (F)	91,62 ± 5,70	96,87 ± 2,06
Li [26] (A)	91,62 ± 5,70	95,37 ± 3,02
TC-SVM (F)	92,15 ± 3,29	99,9 ± 0,13
TC-SVM (A)	92,15 ± 3,29	99,5 ± 0,26
TC-SVM (DBC) (F)	**95,8 ± 3,46**	**99,2 ± 0,62**
TC-SVM (DBC) (A)	**95,8 ± 3,46**	**98,2 ± 0,96**
OC-SVM-Linear (F)	89,7 ± 7,02	95,2 ± 2,81
OC-SVM-Linear (A)	89,7 ± 7,02	94,6 ± 2,74
OC-SVM-RBF (F)	87,2 ± 7,75	91,4 ± 4,05
OC-SVM-RBF (A)	87,2 ± 7,75	90,7 ± 4,23

Tabela 5.4: Comparação de resultados entre os cenários fechado (F) com 15 câmeras e aberto (A) considerando 15 câmeras disponíveis para treinamento e 35 para testes.

cenário aberto de forma satisfatória.

Reconhecimento

Pesquisadores na área de atribuição de fonte procuram considerar, em suas abordagens, que as imagens sob investigação provêm de uma de n câmeras possíveis, uma suposição claramente não realística. Como o foco é expandir esse tipo de análise e considerar um cenário em que a imagem sob investigação pode pertencer a uma de $n+1$ classes possíveis (n câmeras e o complemento ou negativo dessas), acreditamos que os tipos de abordagens propostos gerarão boas contribuições para o estudo de identificação de fonte de documento, visto que a modelagem do problema considerando o universo de dispositivos aos quais não se tem acesso é algo não muito pesquisado.

Considerando um conjunto com n dispositivos, o objetivo da análise de reconhecimento é verificar se a imagem avaliada foi gerada por algum desses e, em caso afirmativo, apontar

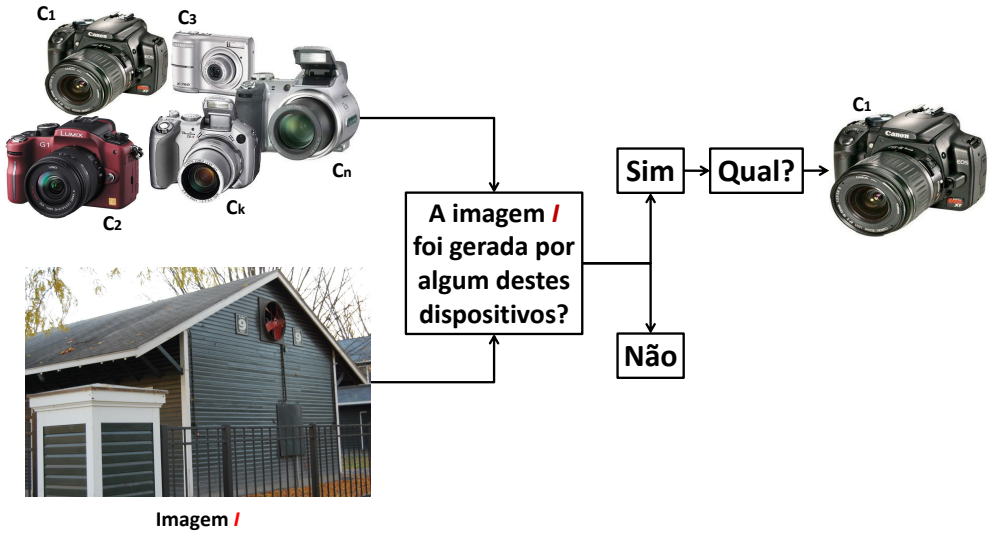


Figura 5.3: Identificação do dispositivo específico — *Reconhecimento*.

a câmera geradora. Caso essa imagem seja pertencente a um dos dispositivos presentes nesse conjunto (o k-ésimo dispositivo, por exemplo, $k \in \{1, 2, ...n\}$), esta deverá ser classificada como *gerada pelo k-ésimo dispositivo*. Caso a imagem tenha sido gerada por um dispositivo diferente dos disponíveis para acesso, esta deverá ser classificada como *gerada por dispositivo indisponível*. Esse tipo de análise é apresentado na Figura 5.3.

A modelagem da classe que representa as câmeras não acessíveis é algo complexo. Esse problema foi resolvido da seguinte maneira: primeiramente, o melhor modelo de hiperplano para cada câmera disponível na etapa de treinamento é encontrado, conforme o Algoritmo 1 apresentado no Capítulo 3 (pag. 33). No momento do teste, dado uma amostra, esta é classificada como *positiva* ou *negativa* em relação a cada uma das câmeras. Isso implica três situações:

1. Se uma imagem foi gerada por uma determinada câmera, ela deverá ser classificada como positiva pelo modelo correspondente a tal dispositivo;

2. Caso uma imagem sob análise tenha sido gerada por uma câmera *desconhecida* na etapa de treinamento, ela deverá ser classificada como *negativa* pelos modelos de hiperplano de **todas** as câmeras disponíveis;

3. Caso uma amostra seja classificada como *positiva* por mais de um modelo, a câmera a ser considerada é aquela para qual a amostra apresenta o maior valor de decisão na classificação (isto é, a maior distância entre seu valor de previsão e o hiperplano gerado antes da etapa de DBC).

No caso da análise de reconhecimento, a acurácia a ser considerada é a acurácia geral Acc_G, dada por

$$Acc_G = \frac{T_{A.C}}{T_A}, \tag{5.4}$$

em que $T_{A.C}$ é o número de amostras classificadas corretamente e T_A é o número total de amostras no teste (todas as classes, incluindo a classe das imagens de origem desconhecida).

Para a análise de reconhecimento, consideramos somente a técnica de TC-SVM com DBC e considerando todas as ROIs da imagem, por ser a mais eficaz no cenário de verificação, como mostrado anteriormente. As Tabelas 5.5, 5.6 e 5.7 apresentam as matrizes de confusão de classificação (acurácia +- desvio padrão) para a abordagem proposta neste trabalho, a abordagem de Lukáš et al. [30] e a abordagem de Li [26], respectivamente, considerando 16 classes (câmeras 1–15 como sendo as classes acessíveis e a classe negativa, ou seja, o conjunto de imagens geradas por todas as câmeras restantes). O resultado apresentado é a acurácia da validação cruzada (*5-fold cross validation*).

As matrizes de confusão apresentadas mostram um bom índice discriminatório para um cenário de classificação multi-classes para a abordagem proposta neste trabalho, comparando com as abordagens de Lukáš et al. [30] e Li [26]. É possível perceber que a classificação ultrapassou a faixa de 75% de acurácia no cenário multi-classe na abordagem proposta, e se mostrou melhor, comparando com os resultados obtidos com as abordagens de Lukáš et al. [30] e Li [26] nesse mesmo cenário.

Câmera	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	NEG
1	**100,00 ±0,00**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00
2	0,00 ±0,00	**98,18 ±1,66**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	1,82 ±1,66
3	0,00 ±0,00	0,00 ±0,00	**98,31 ±1,78**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	1,69 ±1,78
4	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	**100,00 ±0,00**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00
5	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	**93,04 ±3,22**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	6,96 ±3,22
6	0,00 ±0,00	0,66 ±1,44	0,00 ±0,00	0,00 ±0,00	3,29 ±3,33	**82,24 ±7,78**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	13,82 ±7,32
7	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	**98,18 ±2,71**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	1,82 ±2,71
8	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	**79,82 ±5,85**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	20,18 ±5,85
9	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	**84,00 ±4,35**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	16,00 ±4,35
10	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	2,00 ±1,83	2,67 ±4,35	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	**88,00 ±5,06**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	7,33 ±1,49
11	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	**99,33 ±1,49**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,67 ±1,49
12	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	**89,09 ±4,60**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	10,91 ±4,60
13	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	**94,44 ±4,63**	0,62 ±1,40	0,00 ±0,00	4,94 ±4,73
14	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	1,27 ±2,89	0,63 ±1,40	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	**75,32 ±6,62**	0,00 ±0,00	22,78 ±6,09
15	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,61 ±1,36	0,61 ±1,36	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,61 ±1,40	**90,85 ±4,79**	7,32 ±4,58
NEG	0,00 ±0,00	0,00 ±0,00	4,71 ±1,87	0,02 ±0,04	1,00 ±0,63	0,47 ±0,23	0,34 ±0,23	0,29 ±0,56	0,34 ±0,28	0,05 ±0,08	0,02 ±0,04	0,28 ±0,19	0,00 ±0,00	0,71 ±0,19	0,28 ±0,24	**91,50 ±1,93**

Tabela 5.5: Matriz de confusão para a análise de Reconhecimento – Abordagem proposta. Acerto médio: 91.39%.

Câmera	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	NEG
1	**100,00** **±0,00**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00
2	0,00 ±0,00	**94,55** **±3,32**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,61 ±1,36	0,61 ±1,36	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,61 ±1,36	0,00 ±0,00	3,64 ±2,54
3	0,00 ±0,00	1,27 ±1,90	**94,92** **±3,57**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	1,27 ±1,90	0,42 ±0,95	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,85 ±1,17	0,00 ±0,00	1,27 ±1,90
4	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	**98,07** **±1,36**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,39 ±0,86	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	1,54 ±0,86
5	0,00 ±0,00	0,87 ±1,19	0,00 ±0,00	1,30 ±1,94	**93,04** **±4,71**	0,00 ±0,00	0,00 ±0,00	0,43 ±0,97	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	4,35 ±5,10
6	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	1,97 ±1,81	25,00 ±15,65	**65,79** **±10,66**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	7,24 ±7,61
7	0,00 ±0,00	3,64 ±1,36	0,61 ±1,36	1,21 ±1,66	0,00 ±0,00	0,00 ±0,00	**89,09** **±7,30**	1,21 ±2,71	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	4,24 ±5,91
8	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,45 ±1,02	0,00 ±0,00	0,00 ±0,00	1,79 ±1,86	**81,61** **±6,07**	0,90 ±2,03	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	4,93 ±3,97	0,45 ±1,02	9,87 ±4,69
9	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	1,33 ±1,83	0,00 ±0,00	0,00 ±0,00	6,00 ±4,94	6,67 ±7,82	**74,67** **±7,30**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,67 ±1,49	0,00 ±0,00	10,67 ±7,60
10	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	1,33 ±1,83	38,00 ±17,89	6,67 ±4,71	0,67 ±1,49	2,67 ±2,79	2,00 ±1,83	**42,67** **±22,53**	0,00 ±0,00	2,00 ±2,98	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	4,00 ±2,79
11	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	1,33 ±2,98	0,00 ±0,00	0,00 ±0,00	4,00 ±3,65	4,67 ±4,47	2,67 ±4,35	0,00 ±0,00	**76,00** **±9,25**	0,00 ±0,00	0,00 ±0,00	0,67 ±1,49	0,00 ±0,00	10,67 ±8,94
12	0,00 ±0,00	0,61 ±1,36	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	4,24 ±2,71	10,30 ±6,98	2,42 ±2,54	0,00 ±0,00	2,42 ±3,95	**73,94** **±6,28**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	6,06 ±3,71
13	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	3,09 ±5,41	3,70 ±3,95	1,85 ±1,68	0,00 ±0,00	0,62 ±1,36	1,23 ±1,69	**83,95** **±6,95**	1,23 ±2,80	0,00 ±0,00	4,32 ±3,46
14	0,00 ±0,00	0,63 ±1,44	0,63 ±1,40	1,27 ±1,74	0,00 ±0,00	0,00 ±0,00	5,70 ±6,23	9,49 ±7,37	0,63 ±1,40	0,00 ±0,00	1,27 ±1,74	0,63 ±1,40	0,63 ±1,44	**67,09** **±10,28**	0,00 ±0,00	12,03 ±4,67
15	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,61 ±1,36	0,00 ±0,00	0,00 ±0,00	6,10 ±2,14	12,80 ±11,24	1,22 ±1,69	0,00 ±0,00	1,22 ±2,71	0,00 ±0,00	0,00 ±0,00	15,24 ±8,89	**60,98** **±15,75**	1,83 ±2,71
NEG	0,00 ±0,00	0,97 ±0,80	5,50 ±2,26	1,09 ±0,99	10,09 ±4,35	2,62 ±1,95	3,48 ±1,75	6,88 ±5,60	1,66 ±1,10	0,29 ±0,65	1,67 ±2,14	1,41 ±1,49	0,12 ±0,27	7,24 ±4,08	1,48 ±1,30	**55,49** **±6,02**

Tabela 5.6: Matriz de confusão para a análise de Reconhecimento – Lukáš et al. [30]. Acerto médio: 78.18%.

Câmera	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	NEG
1	**100,00 ±0,00**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00
2	0,00 ±0,00	**95,76 ±3,46**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,61 ±1,36	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	3,64 ±3,95
3	0,00 ±0,00	1,69 ±1,78	**95,34 ±4,10**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,85 ±1,15	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,42 ±0,95	1,69 ±3,81
4	0,00 ±0,00	0,77 ±1,06	0,00 ±0,00	**97,68 ±1,61**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	1,54 ±1,61
5	0,00 ±0,00	0,43 ±0,97	0,00 ±0,00	0,43 ±0,97	**94,35 ±6,98**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,43 ±0,97	0,00 ±0,00	0,00 ±0,00	0,43 ±0,97	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	3,91 ±5,41
6	0,66 ±1,49	0,66 ±1,49	0,00 ±0,00	0,66 ±1,44	61,18 ±20,07	**31,58 ±17,78**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	5,26 ±3,80
7	0,00 ±0,00	1,82 ±1,66	0,00 ±0,00	1,21 ±2,71	0,00 ±0,00	0,00 ±0,00	**92,73 ±5,50**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,61 ±1,36	0,00 ±0,00	3,64 ±3,95
8	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,90 ±1,23	**78,92 ±4,91**	0,90 ±1,23	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	3,59 ±3,06	0,00 ±0,00	15,70 ±2,85
9	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,67 ±1,49	0,00 ±0,00	0,00 ±0,00	2,00 ±1,83	5,33 ±5,06	**75,33 ±7,30**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,67 ±1,49	0,00 ±0,00	16,00 ±9,25
10	0,00 ±0,00	0,67 ±1,49	0,00 ±0,00	1,33 ±1,83	63,33 ±21,98	7,33 ±7,23	0,67 ±1,49	1,33 ±1,83	2,67 ±2,79	**18,67 ±15,02**	0,00 ±0,00	0,67 ±1,49	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	3,33 ±4,08
11	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	3,33 ±3,33	2,67 ±3,65	5,33 ±3,80	0,00 ±0,00	**74,00 ±6,83**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	14,67 ±9,31
12	0,00 ±0,00	0,61 ±1,36	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	1,21 ±2,71	7,88 ±4,60	3,64 ±3,95	0,00 ±0,00	1,82 ±4,07	**74,55 ±14,31**	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	10,30 ±10,19
13	0,00 ±0,00	0,00 ±0,00	0,62 ±1,40	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	3,09 ±5,41	3,09 ±2,14	1,85 ±1,69	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	**85,80 ±6,52**	1,23 ±1,71	0,00 ±0,00	4,32 ±6,03
14	0,00 ±0,00	0,63 ±1,44	1,90 ±4,19	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	3,16 ±2,21	11,39 ±2,70	1,90 ±2,81	0,00 ±0,00	0,63 ±1,44	0,63 ±1,40	1,90 ±2,87	**59,49 ±7,35**	0,00 ±0,00	18,35 ±11,26
15	0,00 ±0,00	1,22 ±1,69	1,83 ±4,07	0,61 ±1,40	0,00 ±0,00	0,00 ±0,00	4,27 ±3,45	9,15 ±7,42	1,83 ±1,68	0,00 ±0,00	0,00 ±0,00	0,00 ±0,00	1,22 ±1,69	11,59 ±10,19	**65,85 ±8,67**	2,44 ±2,54
NEG	0,02 ±0,04	0,86 ±0,76	6,29 ±4,71	0,55 ±0,68	16,66 ±6,45	1,26 ±1,24	2,52 ±1,01	4,33 ±3,26	2,52 ±3,45	0,07 ±0,09	1,07 ±1,79	1,50 ±1,71	1,07 ±1,69	5,69 ±4,43	1,62 ±1,17	**53,97 ±3,99**

Tabela 5.7: Matriz de confusão para a análise de Reconhecimento – Li [26]. Acerto médio: 74.63 %.

5.2.2 Correspondência entre dispositivos

Para os experimentos com correspondência entre dispositivos, criamos um conjunto de treino e outro de testes, cada um compostos por 5000 pares de imagens geradas pela mesma câmera e 5000 pares de imagens geradas por câmeras diferentes, sem repetição de pares dentro desses conjuntos.

Nas etapas de treinamento e validação, utilizamos 2500 imagens de cada classe (total de 5000 imagens para cada estágio). Foi aplicada a técnica de validação cruzada em todos os experimentos (*2-fold cross-validation*).

Todos os vetores de características foram normalizados separadamente por meio da Equação 5.5. Uma característica (F_i) de uma amostra simples é normalizada com base nos valores mínimos e máximos entre as demais características do conjunto (F) correspondentes à amostra.

$$F_{i'} = \frac{F_i - min(F)}{max(F) - min(F)}. \tag{5.5}$$

Um primeiro experimento analisa o ruído residual sem qualquer operação de pós - processamento. Nesse caso, nosso método atingiu 84% de acurácia média, considerando pares de imagens gerados ou não pela mesma câmera. É importante ressaltar que sem a normalização da Equação 5.5, nossa técnica não obteve acurácia superior a 52%.

Nosso experimento final consistiu em analisar o hiperplano de decisão gerado (na etapa de treinamento) visando maximizar a acurácia do classificador. Utilizando DBC, conseguimos aumentar a taxa de classificação correta. Assim, obtemos acurácia média de 85%. A Tabela 5.8 apresenta os resultados obtidos (comparando nosso método e a abordagem de Goljan e Fridrich [15]) em termos de acurácia média e desvio padrão para os experimentos com a validação cruzada (*2-fold cross validation*).

Podemos notar pela Tabela 5.8 que a nossa abordagem apresentou resultados superiores, comparando com a técnica proposta Goljan e Fridrich [15]. É possível perceber também que a técnica de DBC também se mostrou bastante eficaz nesse caso.

5.3 Considerações finais

Os resultados obtidos com as abordagens propostas foram satisfatórios, com acurácia relativa elevada. Na atribuição de fonte, a utilização da técnica de movimentação do hiperplano se mostrou bastante eficaz, aumentando a acurácia da classificação. Podemos notar também que a utilização de outras regiões em conjunto com a região central da imagem também apresentou bons resultados. A aplicação da abordagem proposta para atribuição de fonte também se mostrou bastante eficaz em uma análise de reconhecimento, produzindo bons resultados.

Experimento		Acurácia média.	Desvio Padrão
Goljan e Fridrich [15]	*Threshold* sugerido	50,21%	0,71
	Novo cálculo de *threshold*	51,02%	1,28
Abordagem proposta	TC-SVM	83,85%	0,57
	TC-SVM + DBC	86,19%	0,48

Tabela 5.8: Resultados obtidos com *2-fold cross-validation* executado em cada experimento.

A abordagem proposta para resolver o problema de correspondência entre dispositivos também se mostrou bastante eficaz, apresentando resultados satisfatórios. É importante ressaltar que o fato do trabalho proposto em [15] apresentar pouquíssimos detalhes sobre a implementação de sua técnica, os resultados podem não estar corretos em um primeiro momento. Porém, a técnica foi implementada por vários membros do Laboratório de Inferência em Dados Complexos (*Reasoning for Complex Data Laboratory* – RECOD) do Instituto de Computação da UNICAMP e os resultados se mantiveram os mesmos reportados neste trabalho.

Todos os experimentos com SVM foram executados utilizando-se a biblioteca LibSVM [6]. As abordagens propostas neste trabalho, bem como as abordagens propostas por Lukáš et al. [30], Li [26] e Goljan e Fridrich [15] foram implementadas utilizando-se a linguagem MATLAB.

Capítulo 6

Conclusão e trabalhos futuros

Neste trabalho, nós exploramos soluções para o problema de atribuição de fonte de imagens geradas por câmeras digitais, que é uma tarefa de fundamental importância em um cenário criminal, em que imagens falsificadas podem ser utilizadas como falsas evidências de crimes. Foram propostas abordagens com os seguintes objetivos: (1) identificar se um dispositivo sob investigação gerou um conjunto de imagens ilegais; e (2) identificar se um par de imagens foi gerado pelo mesmo dispositivo, sem ter qualquer informação sobre o dispositivo de origem.

Para o primeiro caso, consideramos um cenário mais realístico, denominado cenário aberto, onde uma imagem sob investigação pode ter sido gerada por qualquer dispositivo, e não somente pelos dispositivos disponíveis no momento do treinamento. Essa é somente uma primeira etapa para técnicas de atribuição de fontes. Com a abordagem proposta, é possível analisar imagens de diferentes resoluções sem a necessidade de preenchimento da imagem com valores zero. Além disso, podemos identificar a fonte de imagens considerando métodos de caracterização complementares, tirando vantagem de todos os potenciais métodos de classificação de padrões por aprendizado de máquina.

Com base nos trabalho de Lukáš et al. [30] e Li [26], os experimentos apresentados neste trabalho reportaram resultados com alta taxa de acurácia, tanto com o TC-SVM quanto com o OC-SVM, que pode ser muito útil em um cenário aberto, em que temos acesso a somente à câmera de interesse, considerando a análise de verificação (cenário com duas classes – positiva e negativa). A escolha de outras regiões das imagens além da região central e a utilização da técnica de DBC também se mostraram satisfatórias em cenários onde possuímos acesso a várias câmeras suspeitas.

Este trabalho também apresentou resultados satisfatórios para a análise de reconhecimento, onde temos várias câmeras e o objetivo é identificar se uma imagem suspeita foi gerada por uma dessas câmeras e, em caso afirmativo, apontar qual foi o dispositivo gerador. Como esse tipo de análise é pouco estudado na área de atribuição de

fonte, acreditamos que avaliar a abordagem proposta considerando um ambiente multi-classe pode ser considerada uma contribuição relevante deste trabalho.

No caso da correspondência entre dispositivos, nós expandimos a abordagem proposta por Goljan e Fridrich [15], utilizando o mesmo componente de ruído para realizar tal identificação, aplicando uma caracterização mais robusta das imagens sob investigação e utilizando modelos para melhoria de ruído de forma a reduzir a interferência dos detalhes da cena no ruído das imagens. Por fim, foi utilizada uma classificação supervisionada com o auxilio de DBC para decidir se um par de imagens foi gerado pela mesma câmera. Assim como na atribuição de fontes, o DBC também contribuiu para aprimorar a classificação.

Como resultado final, obtivemos algumas publicações referentes a este trabalho. Foram publicados um resumo estendido[1] no qual é abordada uma visão geral do problema de atribuição de fontes. Também tivemos um trabalho completo aceito para publicação[2], no qual são apresentados os resultados obtidos com a abordagem proposta para o problema de atribuição de fonte em cenário aberto.

Uma grande contribuição deste trabalho é o conjunto de dados gerado, uma vez que este poderá ser utilizado para comparação de outras técnicas existentes. Futuramente, pretendemos ampliar esse conjunto considerando outras categorias de imagens (imagens com baixa resolução, imagens adulteradas digitalmente, etc.).

Assim como todo trabalho, este ainda pode ser melhorado. Experimentos com imagens com baixa qualidade de compressão JPEG devem ser realizados para validar a eficácia das abordagens propostas. Uma outra abordagem que pode melhorar os resultados consiste na extração de características adicionais por meio de PRNU e, também, por meio de CFA (imagens geradas por uma mesma câmera tendem a ter coeficientes de interpolação similar). Também pretendemos investigar novas formas de caracterização de imagens de forma a melhorar os resultados obtidos com a solução proposta para o problema de correspondência entre dispositivos.

Este trabalho pode ser melhorado também para auxiliar o combate contra técnicas contra-forense, como apresentado em [16]. Um possível trabalho futuro é analisar algumas técnicas contra-forense para este trabalho, considerando as técnicas e as características propostas.

Por fim, acreditamos que esforços como a abordagem apresentada por Lukas et al. [30] e outras melhorias como a apresentada neste trabalho tendem a mover as técnicas de atribuição de fonte em direção ao cumprimento dos fortes padrões da Trilogia de Daubert [41], a qual estabelece um alto limiar para aceitação de evidências em um cenário forense (digital e analógico) nas cortes americanas e, possivelmente, em outros países.

[1] Filipe de O. Costa, Anderson Rocha. *Atribuição de fonte em imagens provenientes de câmeras digitais.* II Simpósio de Processamento de Sinais. Campinas, 2011.

[2] Filipe de O. Costa, Michael Eckmann, Walter J. Sheirer, Anderson Rocha. *Open Set Source Camera Attribution.* XXV *Conference on Graphics, Patterns and Images* (SIBGRAPI), Ouro Preto, 2012. (Indicado como *best paper*).

Referências Bibliográficas

[1] S. Bayram, H. Sencar, N. Memon, and I. Avcibas. Source camera identification based on CFA interpolation. In *IEEE International Conference on Image Processing (ICIP)*, pages 69–72, Genova, Italy, 2005.

[2] C. M. Bishop. *Pattern Recognition and Machine Learning, 1st edition.* Springer, 2006.

[3] P. Blythe and J. Fridrich. Secure digital camera. Digital Forensic Research Workshop, 2004.

[4] R. Caldelli, I. Amerini, and A. Novi. An analysis on attacker actions in fingerprint–copy attack in source camera identification. In *IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–6, Foz do Iguaçu.

[5] R. Caldelli, I. Amerini, F. Picchioni, and M. Innocenti. Fast image clustering of unknown source images. In *IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–5, 2010.

[6] C. Chang and C. Lin. LIBSVM: A library for support vector machines. *Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27:1–27:27, 2011.

[7] M. Chen, J. Fridrich, M. Goljan, and J. Lukas. Determining image origin and integrity using sensor noise. *IEEE Transactions on Information Forensics and Security (TIFS)*, 3(1):74–90, 2008.

[8] M. Chen, J. Fridrich, M. Goljan, and J. Lukas. Source digital camcorder identification using sensor photo-response non-uniformity. *In SPIE Photonics West*, pages 1G–1H, 2008.

[9] P. Chiang, N.Khana, A. K. Mikkilineni, M. V. O. Segovia, S. Suh, J. P. Allebach, G. T. C. Chiu, and E. J. Delp. Printer and scanner forensics. *IEEE Signal Processing Magazine*, 72(2):72–83, March 2009.

[10] C. Cortes and V. Vapnik. *Machine Learning*, chapter Support-Vector Networks, pages 273–297. Kluwer, 20 edition, 1995.

[11] A. E. Dirik, H. T. Sencar, and N. Memon. Digital single lens reflex camera identification from traces of sensor dust. *IEEE Transactions on Information Forensics and Security (TIFS)*, 3(3):539–552, September 2008.

[12] Z. J. Geradts, J. Bijhold, M. Kieft, K. Kurosawa, K. Kuroki, and N. Saitoh. Methods for identification of images acquired with digital cameras. *Enabling Technologies for Law Enforcement and Security*, 4232:505–512, 2001.

[13] T. Gloe, M. Kirchner, A. Winkler, and R. Böhme. Can we trust digital image forensics? In *ACM Multimedia*, pages 78–86, 2007.

[14] D. B. Goldman and J. H. Chen. Vignette and exposure calibration and compensation. In *International Conference on Computer Vision (ICCV)*, pages 899–906, 2005.

[15] M. Goljan and J. Fridrich. Identifying common source digital camera from image pairs. In *IEEE International Conference on Image Processing (ICIP)*, pages 14–19, 2007.

[16] M. Goljan, J. Fridrich, and M. Chen. Defending against fingerprint-copy attack in sensor-based camera identification. In *IEEE Transactions on Information Forensics and Security (TIFS)*, pages 227–236, 2011.

[17] M. Goljan, J. Fridrich, and T. Filler. Large scale test of sensor fingerprint camera identification. In *Proc. SPIE*, volume 7254, January 2009.

[18] M. Goljan, J. Fridrich, and J. Lukas. Camera identification from printed images. In *SPIE Conference on Security, Forensics, Steganography, and Watermarking of Multimedia Contents*, volume 6819, 2008.

[19] R. Gonzalez and R. Woods. *Digital Image Processing*. Prentice-Hal, 2007.

[20] C. Hsu, C. Chang, and C. Lin. A practical guide to support vector classification, 2010.

[21] E. Kee and H. Farid. Printer profiling for forensic and balistic. In *ACM Workshop on Multimedia and Security*, volume 10, pages 3–10, 2008.

[22] N. Khanna, A. K. Mikkilineni, G. T. C. Chiu, J. P. Allebach, and E. J. Delp. Scanner identification using sensor pattern noise. *SPIE Security, Steganography and Watermarking of Multimedia Contents (SSWMC)*, 6505, 2007.

[23] N. Khanna, A. K. Mikkilineni, and E. J. Delp. Scanner identification using feature-based processing and analysis. *IEEE Transactions on Information Forensics and Security (TIFS)*, 4(1):123–139, 2009.

[24] M. Kharrazi, H. Sencar, and N. Memon. Blind source camera identification. In *IEEE International Conference on Image Processing (ICIP)*, pages 709 – 712, Singapore, 2004.

[25] K. Kurosawa, K. Kuroki, and N. Saitoh. CCD fingerprint method – identification of a video camera from videotaped images. In *IEEE International Conference on Image Processing (ICIP)*, pages 537–540, 1999.

[26] C.-T. Li. Source camera identification using enhanced sensor pattern noise. *IEEE Transactions on Information Forensics and Security (TIFS)*, 5(2):280–287, June 2010.

[27] C.-T. Li. Unsupervised classification of digital images using enhanced sensor pattern noise. In *IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 3429–3432, 2010.

[28] C.-T. Li and R. Sata. On the location-dependent quality of the sensor pattern noise and its implication in multimedia forensics. In *Proc. IV International Conference on Imaging for Crime Detection and Prevention (ICDP)*, pages 1–6, 2011.

[29] D. Li. Ballistics projectile image analysis for firearm identification. *IEEE Transactions on Image Processing (TIP)*, 15(10):2857–2865, 2002.

[30] J. Lukas, J. Fridrich, and M. Goljan. Digital camera identification from sensor pattern noise. *IEEE Transactions on Information Forensics and Security (TIFS)*, 1(2):205–214, 2006.

[31] S. Lyu. *Natural Image Statistics for Digital Image Forensics*. Phd thesis, Dartmouth College, August 2005.

[32] M. Kivanc Mihcak, Igor Kozintsev, Kannan Ramchandran, and Pierre Moulin. Low-complexity image denoising based on statistical modeling of wavelet coefficients. *IEEE Signal Processing Letters*, 6(12):300–303, December 1999.

[33] P. Phillips, P. Grother, and R. Micheals. *Handbook of Face Recognition*, chapter Evaluation Methods on Face Recognition, pages 329–348. Springer, 2005.

[34] A.C. Popescu and H. Farid. Exposing digital forgeries in color filter array interpolated images. *IEEE Transactions on Signal Processing (TSP)*, 53(10):3948–3959, 2005.

[35] T. Riopka and T. Boult. The eyes have it. In *ACM workshop on Biometric methods and applications*, volume 3, pages 33–40, 2003.

[36] T. Riopka and T. Boult. Classification enhancement via biometric pattern pertubation. In *Audio-Video Based person Authentication (AVBPA)*, pages 850 – 859, 2005.

[37] A. Rocha and S. Goldenstein. *Atualizações em Informática (2010)*, chapter CSI: Análise Forense de Documentos Digitais, pages 263–317. Sociedade Brasileira de Computação (SBC), Belo Horizonte, Brazil, 1 edition, July 2010.

[38] A. Rocha, W. Scheirer, T. E. Boult, and S. Goldenstein. Vision of the unseen: Current trends and challenges in digital image and video forensic. *ACM Computing Surveys (CSUR)*, 42(26):26:1–26:42, October 2011.

[39] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson. Estimating the support of a high-dimensional distribution. *Neural Computation*, 13(7):1443–1471, 2001.

[40] T. Sencar and N. Memon. *Algorithms, architectures and information systems security*, volume 3, chapter Overview of State-of-the-art in Digital Image Forensic, pages 325–347. Sociedade Brasileira de Computação (SBC), Indian Statistical Institute, India, 1 edition, November 2008.

[41] Donald E. Shelton. *Forensic Science in Court - Challenges in the 21st Century*. Rowman & Littlefield Publishers, 2011.

[42] S.N. Srihari. Beyond C.S.I.: The rise of computational forensic. Online at http://spectrum.ieee.org/computing/software/beyond-csi-the-rise-of-computational-forensics – Last Access in July 10Th, 2012, December 2010.

[43] Yagiz Sutcu, Sevinc Bayran, Husrev Sencar, and Nassir Memon. Improvements on sensor noise based source camera identification. In *IEEE International Conference on Multimedia and Expo*, pages 24–27, 2007.

[44] A. Swaminathan, M. Wu, and K.J.R. Liu. Non-intrusive forensic analysis of visual sensors using output images. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 401–404, Atlanta, USA, 2006.

[45] A. Swaminathan, M. Wu, and K.J.R. Liu. Component forensic - theory, methodologies e applications. *IEEE Signal Processing Magazine*, 26(2):38–48, March 2009.

[46] B. Wang, X. Kong, and X. You. Source camera identification using support vector machines. In *Advances in Digital Forensics V*, volume 306 of *IFIP Advances in Information and Communication Technology*, pages 107–118. Springer Boston, 2009.

[47] X. Wang and Z. Weng. Scene abrupt change detection. In *Canadian Conference on Electrical and Computing Engineering*, pages 880–883, 2000.

[48] X. S. Zhou and T. S. Huang. Relevance feedback in image retrieval : A comprehensive review. *Multimedia Systems*, 8(6):536–544, 2003.