

Projeto e Desenvolvimento de Técnicas Forenses para Identificação de Imagens Sintéticas

Este exemplar corresponde à redação final da Dissertação devidamente corrigida e defendida por Éric Keiji Tokuda e aprovada pela Banca Examinadora.

Campinas, Julho de 2012.

Prof. Dr. Hélio Pedrini (Orientador)

Prof. Dr. Anderson de Rezende Rocha
(Co-orientador)

Dissertação apresentada ao Instituto de Computação, UNICAMP, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação.

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

Tokuda, Eric Keiji, 1984-
T573p Projeto e desenvolvimento de técnicas forenses para
identificação de imagens sintéticas / Eric Keiji Tokuda. – Campinas,
SP : [s.n.], 2012.

Orientador: Hélio Pedrini.

Coorientador: Anderson de Rezende Rocha.

Dissertação (mestrado) – Universidade Estadual de Campinas,
Instituto de Computação.

1. Análise forense de imagem. 2. Computação forense. 3.
Fusão de classificadores. 4. Fusão de características. 5.
Aprendizado do computador. I. Pedrini, Hélio, 1963-. II. Rocha,
Anderson de Rezende. III. Universidade Estadual de Campinas.
Instituto de Computação. IV. Título.

Informações para Biblioteca Digital

Título em inglês: Design and development of forensic techniques for synthetic
image identification

Palavras-chave em inglês:

Digital forense

Forensic computing

Classifier fusion

Feature fusion

Machine learning

Área de concentração: Ciência da Computação

Titulação: Mestre em Ciência da Computação

Banca examinadora:

Hélio Pedrini [Orientador]

Moacir Pereira Ponti Junior

Ricardo da Silva Torres

Data de defesa: 24-07-2012

Programa de Pós-Graduação: Ciência da Computação

TERMO DE APROVAÇÃO

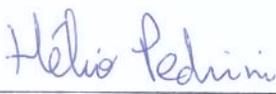
Dissertação Defendida e Aprovada em 24 de Julho de 2012, pela
Banca examinadora composta pelos Professores Doutores:



Prof. Dr. Moacir Pereira Ponti Junior
ICMC / USP



Prof. Dr. Ricardo da Silva Torres
IC / UNICAMP



Prof. Dr. Hélio Pedrini
IC / UNICAMP

Projeto e Desenvolvimento de Técnicas Forenses para Identificação de Imagens Sintéticas

Éric Keiji Tokuda¹

24 de Julho de 2012

Banca Examinadora:

- Prof. Dr. Hélio Pedrini (Orientador)
- Prof. Dr. Moacir Pereira Ponti Junior (ICMC/USP)
- Prof. Dr. Ricardo da Silva Torres (IC/UNICAMP)

¹Suporte financeiro de: FAPESP (processo 10/13745-6) 2011–2012) e CNPq

Resumo

O grande investimento de companhias de desenvolvimento de *software* para animação 3D nos últimos anos tem levado a área de Computação Gráfica a patamares nunca antes atingidos. Frente a esta tecnologia, torna-se cada vez mais difícil a um usuário comum distinguir fotografias reais de imagens produzidas em computador. Mais do que nunca, a fotografia, como meio de informação segura, passa a ter sua idoneidade questionada. A identificação de imagens geradas por computador tornou-se uma tarefa imprescindível. Existem diversos métodos de classificação de imagens fotográficas e geradas por computador na literatura. Todos trabalhos se concentram em identificar diferenças entre imagens fotográficas e imagens geradas por computador. Contudo, no atual estágio da Computação Gráfica, não há uma caracterização isolada que resolva o problema. Propomos uma análise comparativa entre diferentes formas de combinação de descritores para abordar este problema. Para tanto, criamos um ambiente de testes com diversidade de conteúdo e de qualidade; implementamos treze métodos representativos da literatura; criamos e implementamos quatro abordagens de fusão de dados; comparamos os resultados dos métodos isolados com o resultado dos mesmos métodos combinados. Realizamos a implementação e análise de um total de treze métodos. O conjunto de dados para validação foi composto por aproximadamente 5.000 fotografias e 5.000 imagens geradas por computador. Resultados isolados atingiram acurácias de até 93%. A combinação destes mesmos métodos atingiu uma precisão de 97% (uma redução de 57% no erro do melhor método de maneira isolada).

Abstract

The development of powerful and low-cost hardware devices allied with great advances on content editing and authoring tools have pushed the creation of computer generated images (CGI) to a degree of unrivaled realism. Differentiating a photorealistic computer generated image from a real photograph can be a difficult task to naked eyes. Digital forensics techniques can play a significant role in this task. Indeed, important research has been made by our community in this regard. The current approaches focus on single image features aiming at spotting out differences between real and computer generated images. However, with the current technology advances, there is no universal image characterization technique that completely solves this problem. In our work, we present a complete study of several current CGI vs. Photograph approaches; create a big and heterogeneous dataset to be used as a training and validation database; implement representative methods of the literature; and devise automatic ways to combine the best approaches. We compare the implemented methods using the same validation environment. Approximately 5,000 photographs and 5,000 CGIs with large diversity of content and quality were collected. A total of 13 methods were implemented. Results show that this set of methods, in an integrated approach, can achieve up to 93% of accuracy. The same methods, when combined through the proposed fusion schemes, can achieve an accuracy rate of 97% (a reduction of 57% of the error over the best result alone).

Agradecimentos

Os autores agradecem ao Dr. Gamhewage Chaminda de Silva por gentilmente fornecer as imagens de seu conjunto de dados, ao Dr. William Robson Schwartz por sua assistência com a implementação dos Histograma de Coeficientes Shearlets e ao Ricardo Dutra por nos auxiliar com a implementação de métodos de detecção de bordas. Os autores agradecem à FAPESP (processo número 2010/13745-6), ao CNPq e à CAPES pelo apoio financeiro.

Sumário

Resumo	v
Abstract	vi
Agradecimentos	vii
1 Introdução	1
2 Conceitos	5
2.1 Processo de Formação de Imagens	5
2.1.1 Formação de IFGs	5
2.1.2 Formação de ICGs	7
2.2 Classificadores	9
2.3 Curvas ROC	10
3 Trabalhos Relacionados	11
3.1 Métodos do Estado da Arte	12
3.2 Métodos de Áreas Relacionadas	19
3.3 Combinação de Métodos	25
4 Metodologia	27
4.1 Escolha das Imagens	29
4.2 Implementação dos Métodos do Estado da Arte	30
4.3 Implementação de Métodos de Áreas Relacionadas	31
4.4 Implementação de Métodos de Combinação	35
5 Experimentos e Validação	41
5.1 Configuração do Sistema	41
5.2 Coleta das Imagens	42
5.3 Validação de Métodos do Estado da Arte	43

5.4	Validação de Métodos de Áreas Relacionadas	44
5.5	Validação dos Métodos de Combinação	50
5.6	Análise Comparativa	52
6	Conclusões e Trabalhos Futuros	57
	Bibliografia	59

Lista de Tabelas

3.1	Comparativo entre os principais métodos existentes na literatura.	18
4.1	Conceitos utilizados pelos métodos implementados.	30
5.1	Resultados do método LYU.	44
5.2	Resultados do método LI.	44
5.3	Resultados do método BOX.	45
5.4	Resultados do método CON.	45
5.5	Resultados do método CUR.	46
5.6	Resultados do método GLCM.	46
5.7	Resultados do método HOG.	47
5.8	Resultados do método HSC.	47
5.9	Resultados do método LBP.	48
5.10	Resultados do método LSB.	48
5.11	Resultados do método SHE.	49
5.12	Resultados do método SOB.	49
5.13	Resultados do método POP.	49
5.14	Resultados do método de entropia.	50
5.15	Comparação entre os resultados dos métodos de combinação.	51
5.16	Comparação entre os resultados de todos os métodos implementados.	54

Lista de Figuras

1.1	Amostra de imagens.	2
2.1	Diagrama do processo de formação de uma imagem.	6
2.2	Amostra de ICGs.	8
3.1	Operadores da matriz de coocorrência.	22
3.2	Diagrama do cálculo do LBP.	24
4.1	Fluxograma da extração e validação de uma abordagem IFG versus ICG.	28
4.2	Representação da matriz de características.	28
4.3	Fluxograma do método de Concatenação.	36
4.4	Fluxograma do método de Votação Simples.	37
4.5	Fluxograma do método de Votação Ponderada.	38
4.6	Fluxograma do método de Metaclassificação.	40
5.1	Amostra de imagens corretamente identificadas.	52
5.2	Diagrama das possibilidades de acertos por um conjunto de classificadores.	53
5.3	Amostra de imagens incorretamente identificadas.	54
5.4	Curvas ROC dos métodos isolados implementados.	55
5.5	Curva ROC do Metaclassificador.	56
5.6	Amostra de imagens do <i>Fake of Foto</i>	56

Lista de Abreviaturas e Siglas

AUC	<i>Area Under the Curve</i>	Área Sob a Curva
GLCM	<i>Gray Level Co-occurrence Matrix</i>	Matriz de Coocorrência
HOG	<i>Histogram of Oriented Gradients</i>	Histograma de Gradientes Orientados
HSC	<i>Histogram of Shearlet Coefficients</i>	Histograma de Coeficientes <i>Shearlets</i>
HSV	<i>Hue Saturation Value</i>	Matiz Saturação Intensidade
LBP	<i>Local Binary Patterns</i>	Padrões Binários Locais
LDA	<i>Local Discriminant Analysis</i>	Análise Discriminante Linear
LSB	<i>Least Significant Bits</i>	Bits Menos Significativos
PCA	<i>Principal Component Analysis</i>	Análise de Componentes Principais
RGB	<i>Red Green Blue</i>	Vermelho Verde Azul
ROC	<i>Receiver Operation Characteristic</i>	Característica de Operação do Receptor
SVM	<i>Support Vector Machines</i>	Máquinas de Vetores de Suporte

BOX	<i>Método baseado em contagem de caixas</i>
CON	<i>Método baseado em coeficientes contourlets</i>
CUR	<i>Método baseado em coeficientes curvelets</i>
ENT	<i>Método baseado em entropia</i>
FUS _i	<i>Método de combinação i</i>
ICG	<i>Imagem gerada por Computação Gráfica</i>
IFG	<i>Imagem Fotográfica</i>
LI	<i>Método baseado em diferenças de segunda ordem</i>
LYU	<i>Método baseado em coeficientes wavelets</i>
POP	<i>Método baseado em mosaico</i>
SHE	<i>Método baseado em coeficientes shearlets</i>

Capítulo 1

Introdução

A redução dos custos dos equipamentos de aquisição de imagens digitais e o constante avanço da tecnologia impulsionaram a popularização das fotografias digitais. Mesmo um usuário de classe econômica média tem acesso a câmeras de alta definição, filmadoras e *scanners*. Atualmente, há grande quantidade de Imagens Fotográficas Digitais (IFG) *online*, de diversos conteúdos, origens e níveis de profissionalismo. O repositório de imagens *Flickr*¹, por exemplo, de acordo com o *blog* dos próprios autores, atingiu em 2011, a marca de 6 bilhões de *uploads* de usuários desde sua criação.

Paralelamente, os constantes avanços tecnológicos de *hardware* e *software* gráficos impulsionam a Computação Gráfica. Empresas como a *Nvidia* produzem unidades de processamento gráfico (*Graphics Processing Unit – GPU*) projetadas para lidar paralelamente com grande quantidade de tarefas. Estas empresas reportam² que seus produtos já atingem mais de 500 bilhões de instruções lógicas de ponto flutuante por segundo (*Float Point Operations per Second – FLOP*). Paralelamente, ferramentas de *software* gráfico como *Maya*³ e *3ds Max*⁴ estão constantemente sendo melhoradas para acompanhar o alto desempenho de *hardware*.

A área de Computação Gráfica tem uma diversidade de aplicações industriais e científicas, como a criação de animações gráficas cinematográficas, modelagem assistida por computador (*Computer Aided Design – CAD*), simulação de sistemas, reconstituição de cenas criminais, entre outras. Os avanços da área permitiram o surgimento de modelos anteriormente inconcebíveis, como o de representações de cenas reais com alto grau de realismo (fotorrealismo).

Em meio a tanto avanço na qualidade das imagens, identificar uma imagem fotorrealista gerada por Computação Gráfica (ICG) pode representar uma tarefa complexa a olho

¹<http://www.flickr.com> (último acesso em 23/07/2012)

²<http://www.nvidia.com/object/why-choose-tesla.html> (último acesso em 23/07/2012)

³<http://usa.autodesk.com/maya/> (último acesso em 23/07/2012)

⁴<http://usa.autodesk.com/3ds-max/> (último acesso em 23/07/2012)

nu. Embora o sistema visual humano seja incomparável em aspectos como alta precisão, rápido entendimento da cena e reconhecimento de padrões em alto nível, ele é ineficiente em tarefas como identificação de inconsistências de iluminação, de posição de observação e de cores [25]. A Figura 1.1 apresenta uma amostra de imagens que pode apresentar dificuldades ao usuário que tentar identificar se alguma delas é uma ICG.



Figura 1.1: Exemplos de uma (a) imagem fotorrealista gerada por Computação Gráfica (ICG) e uma (b) imagem fotográfica (IFG).

No cenário jurídico, uma imagem fotográfica pode ser considerada uma importante evidência criminal. Por exemplo, os Estados Unidos têm uma política rigorosa de combate ao abuso infantil. Em 1996, foi estabelecida a emenda *Child Pornography Prevention Act (CPPA)* [23], a qual previa combater a pornografia infantil virtual. Esta emenda incriminava quaisquer informações multimídia que aparentassem ter conteúdo pornográfico infantil. A medida incriminava, inclusive, ICGs. Seis anos depois, a corte norte-americana reanalisou a emenda e a considerou como uma violação constitucional. A partir de então, a criação e distribuição de imagens sintéticas com conteúdo pornográfico foi descriminalizada. Atualmente, embora no Brasil não hajam leis que rejam o assunto, nos Estados Unidos, imagens associadas a abuso infantil são consideradas legalmente protegidas se for provado o caráter sintético da imagem. Este é um cenário forense em que é fundamental distinguir de forma clara ICGs de IFGs.

A imagem tem importante papel como documento. Alterações da imagem que não afetam o entendimento da cena raramente constituem-se em uma adulteração criminosa, como ocorre nos casos de alterações estéticas de fotografias. Se descobertos, contudo, podem levar a uma situação constrangedora pela fonte de dados, o que é frequente em revistas de moda. Alterações de conteúdo têm, em grande parte das vezes, motivação política, como foi o caso, em 2008, da imagem dos mísseis adulterada, a qual dava a ideia de uma forte ofensiva iraniana [66]. Alterações em níveis de bits também têm grande interesse forense, pois se constituem como um simples e eficiente meio de transmissão de informações utilizado pela esteganografia. Diversos são os casos que utilizam uma

adulteração maliciosa da imagem. A validação da autenticidade de imagens, portanto, é um importante desafio forense.

A Computação Forense pode ser definida como a utilização de métodos científicos para preservação, coleção, validação, identificação, análise, interpretação e documentação de evidências digitais como suporte a processos criminais [54]. Numa abordagem minimalista, todo equipamento digital armazena informações digitais e, portanto, contém dados que podem ser utilizados num processo criminal. Como estamos rodeados por aparatos eletrônicos, é de se esperar que a área de Computação Forense ganhe cada vez mais importância. Desde sequências numéricas em memória dinâmica até discos rígidos inteiros são alvos desta seção criminalística.

Os desafios da computação forense tem tanto interesse governamental como interesse comercial. Um exemplo de interesse comercial é a busca de serviços forenses para auditoria interna, visando evitar um escândalo gerado pelo vazamento de informações. Empresas como a *Computer Forensics Solutions*⁵ oferecem serviços forenses e garantem sigilo. Outra categoria de empresas, como a da *EnCase*⁶, oferece aplicativos de análise de documentos digitais.

Uma solução ao problema de distinção entre IFG e ICG seria confiar a tarefa a um perito no assunto. O profissional deveria ter uma visão analítica e minuciosa sobre as imagens. Esta solução, contudo, além de exigir a existência de pessoas com tais virtudes visuais, teria o risco de julgamentos subjetivos (por exemplo, baseado no conteúdo da imagem).

Se cada IFG tivesse um identificador não burlável que identificasse sua origem, como uma *marca d'água*, então o problema estaria resolvido [35]. Porém, esta abordagem exigiria que a marca d'água fosse inserida em todas as IFGs existentes. Mesmo que todos os equipamentos de aquisição de imagem digital, incluindo câmeras fotográficas digitais, câmeras de filmagem e telefones móveis passassem a ter este marcador, as imagens antigas ainda não o teriam. Portanto, abordagens não-intrusivas, ou seja, que não precisam alterar o conjunto de dados alvo, constituem-se como as opções factíveis para resolver o problema.

Pesquisadores abordaram de diversas formas o problema de IFG versus ICG. O sistema visual humano é bastante complexo e utiliza diversos recursos visuais para classificar uma cena. Utilizando-o como inspiração, podemos tentar identificar os traços visuais que o fazem distinguir uma IFG de uma ICG e utilizá-los em nossa abordagem. Bordas, cores e formas são alguns dos exemplos de características visuais que podem ser utilizadas [59]. Para aumentar a acurácia dos resultados obtidos a partir dos métodos do Estado da Arte, uma solução seria propor uma abordagem que conseguisse extrair informações relevantes para a classificação ainda não exploradas por outros métodos. Uma segunda solução seria

⁵<http://www.computerforensicssolutions.com/> (último acesso em 23/07/2012)

⁶<http://www.guidancesoftware.com/forensic.htm> (último acesso em 23/07/2012)

combinar os métodos existentes para criar um novo classificador.

Embora trabalhos anteriores reportem acurácias de classificação superiores a 95% [17, 74], Gloe et al. [29] criaram cenários nos quais os desempenhos de determinados métodos foram significativamente inferiores às acurácias reportadas. Neste contexto, em situações reais de classificação de imagens geradas por computador, os métodos existentes não apresentam os níveis de acurácia reportados. De acordo com os teoremas “Não há almoço grátis”, não existe um único classificador que seja ótimo em todos os cenários de classificação [76]. Para isso, deveríamos ter uma regra determinística para saber qual classificador é o ideal para cada natureza de dados. Esta natureza dos dados, contudo, é na maior parte das vezes desconhecida.

A combinação de abordagens tem como um dos intuitos melhorar a acurácia quando comparada a cada abordagem isolada. Propomos a combinação de características aplicado ao cenário de identificação de ICG.

Deste modo, na tarefa de discernimento entre IFGs e ICGs, temos como contribuições neste trabalho:

- criação de um *cenário* mais próximo do real em termos de complexidade e diversidade, pela composição de um banco de imagens heterogêneo e que inclua IFGs e ICGs de alta complexidade. Utilização deste cenário para uma análise comparativa entre todos os métodos implementados;
- implementação de *métodos da literatura* que tratam do mesmo problema, para termos um critério de comparação com os métodos criados;
- criação e implementação de novos métodos, por meio da extensão de descritores de *outras áreas* ao nosso escopo;
- proposição e implementação de formas de *combinação de características*, com o intuito de diminuir a dimensionalidade do espaço de características e aumentar a acurácia dos métodos existentes.

O restante do texto é dividido da seguinte forma: no Capítulo 2, analisamos os métodos da literatura que abordam o problema de IFG versus ICG e, adicionalmente, expomos descritores que podem ser utilizados na criação de novos métodos; no Capítulo 3, apresentamos nossas propostas e implementações de cada uma delas; no Capítulo 4, expomos e analisamos os resultados de cada experimento realizado. Por fim, no último capítulo, apresentamos considerações finais sobre o trabalho, como um todo. Cada capítulo leva em conta a divisão por contribuições do trabalho supracitada.

Capítulo 2

Conceitos

Este capítulo apresenta alguns conceitos básicos relacionados ao processo de formação de imagens e descreve brevemente alguns tipos de classificadores e uma forma gráfica para avaliação de resultados de classificação.

2.1 Processo de Formação de Imagens

Conhecer os processos de formação de ICG e IFG é um ponto de partida para conseguirmos distingui-las, pois nos possibilita identificar as diferenças entre as classes que são devidas às diferenças entre os processos de criação. Identificar com precisão o processo de formação de uma imagem implica a classificação de uma imagem em IFG ou ICG. Esta seção divide-se nas descrições sucintas de ambos os processos de formação.

2.1.1 Formação de IFGs

Câmeras fotográficas são responsáveis pela representação da cena 3D por meio de uma imagem. O processo de captura se inicia com o registro de uma amostra da luz incidente. A luz emitida por uma fonte luminosa é refletida nos objetos da cena e incidida na câmera. Há, adicionalmente, interação da luz entre as fontes luminosas, refração na mudança de meios e inter-reflexão entre os objetos da cena [50].

Inicialmente, o cone de luz da cena atinge o sistema de lentes. O sistema de lentes registra uma porção limitada do cone de luz da cena, devido à sua limitação de campo visual, insere distorções geométricas de acordo com as características físicas das lentes e foca a luz nos fotossensores.

O sensores convertem a energia luminosa em níveis de tensão elétrica. A representação da luz por meio de bits adotada pelas câmeras está associada à fisiologia do olho humano: um bom conjunto de cores primárias representa estímulos que maximizam a diferença

entre as respostas dos cones da retina para diferentes comprimentos de onda de luz [34]. O sistema de cores mais utilizado é o composto por vermelho, verde e azul (RGB).

Uma aquisição precisa das cores de um *pixel* por câmeras com representação de cores em RGB exigiria medir efetivamente as três cores a partir do espectro da luz incidente. No entanto, a maior parte destas câmeras emprega um mosaico de filtros espectrais (*Color Filter Arrays – CFA*) para medir apenas uma faixa do espectro em cada *pixel*. Cada *pixel* fica associado a apenas uma cor do sistema tricromático.

Na etapa de demosaico, as duas cores restantes de cada *pixel* são estimadas. O conhecimento da disposição dos filtros espectrais possibilita estimar as cores restantes. O CFA mais comum entre as câmeras populares é o Bayer [3].

O *software* interno à câmera ainda aplica operações sobre a imagem, incluindo: redução de ruído, correção gama, realce de características da imagem, conversão do espaço de cores e compressão.

A Figura 2.1 resume o processo de formação de uma IFG por uma câmera com CFA.

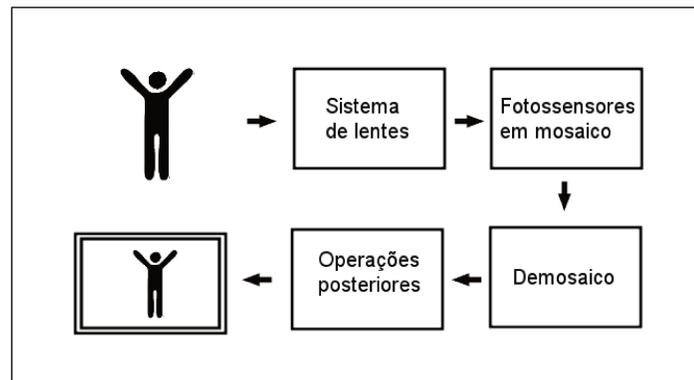


Figura 2.1: Diagrama ilustrativo do processo de formação de uma IFG por uma câmera com CFA. (1) A luz passa através do sistema óptico da câmera e incide sobre os fotossensores. (2) Cada sensor captura apenas uma das cores primárias. (3) Na etapa de demosaico, as cores restantes são estimadas. (3) Operações posteriores de melhoria e compressão da imagem são aplicados sobre a imagem resultante.

A maior parte dos sensores é construída a partir de semicondutores metal-óxido [50]. A utilização deste material está associada a tipos específicos de ruídos [32]. O ruído de uma câmera pode ser definido como qualquer variação randômica no brilho ou na cor da imagem causado por algum processo interno à câmera. O ruído, portanto, pode ser de vários tipos [59]:

- ruído de padrão fixo: heterogeneidade dos elementos dos sensores sob um mesmo nível de iluminação;

- ruído térmico: passagem de elétrons sobre o sensor, mesmo na ausência de iluminação, causado pela alimentação elétrica do conjunto de sensores;
- ruído de aquisição do fóton: o nível de luz coletado por um único sensor não é constante e se dá por um processo aleatório de *Poison*.

2.1.2 Formação de ICGs

A síntese de uma ICG envolve duas fases consecutivas: a fase de *modelagem* e a de *renderização*.

A modelagem é uma etapa assistida por ferramentas de criação. Nesta fase, o acesso às ferramentas, a experiência e o conhecimento do autor sobre elas são os principais diferenciais na qualidade da imagem criada.

Há uma grande quantidade de ferramentas dedicadas à computação gráfica, as quais incluem as de uso geral ou de uso específico. Dentre as de uso geral, estão as populares *Maya*¹, *Blender*² e *3ds Max*³, com as quais é produzida a maior parte das ICG disponíveis publicamente. As ferramentas de *software* de uso específico simplificam e minimizam o trabalho em um determinado escopo: *scanners* 3D como o da *Next Engine*⁴ criam o modelo 3D de um objeto a partir de uma simples varredura a *laser* e são frequentemente usados em engenharia de produtos. Na modelagem da anatomia de um ser humano, a *Cyberware Body Scanner*⁵ reporta a capacidade de gerar o modelo de uma pessoa em menos de 17 segundos, por meio de escaneamento a *laser*. Ferramentas como estas geram modelos que podem ser exportados para ferramentas de edição, tais como aplicação e refinamento de textura, de animação, entre outras.

O modelo previamente gerado é utilizado na fase de renderização, quando é criada a representação da cena por uma imagem. Nesta fase, o usuário escolhe o nível de detalhamento a ser aplicado pelo computador, o qual pode variar drasticamente o tempo de processamento.

Dentro da classe das imagens geradas por computador, há diversos tipos de imagens:

- representação fictícia da realidade (principalmente utilizada em animações gráficas e em simulação de sistemas);
- imagens gráficas como mapas e diagramas, em que nos focamos em determinados aspectos e abstraímos o restante;

¹<http://usa.autodesk.com/maya/> (último acesso em 23/07/2012)

²<http://www.blender.org/> (último acesso em 23/07/2012)

³<http://usa.autodesk.com/3ds-max/> (último acesso em 23/07/2012)

⁴<http://www.nextengine.com/> (último acesso em 23/07/2012)

⁵<http://www.cyberware.com/> (último acesso em 23/07/2012)

- representação fotorrealista, em que o principal objetivo da imagem é o de se aproximar de uma imagem real.

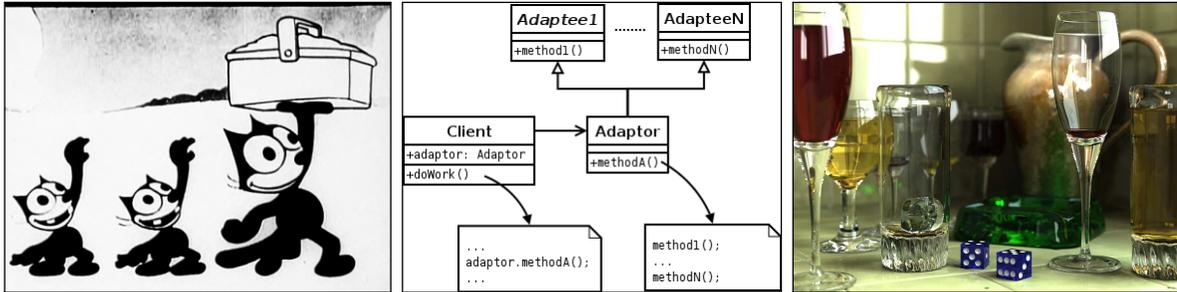


Figura 2.2: Amostra de imagens geradas por computador: (a) uma imagem de desenho animado, (b) um diagrama de classes e (c) uma imagem fotorrealista.

Neste trabalho, quando falamos em ICG, estamos nos referindo às imagens *fotorrealistas* geradas por computação gráfica. Métodos que abordam o nosso problema de maneira mais genérica, incluindo todas as classes acima, foram explorados por trabalhos como o de Athitsos et al. [2, 13].

O nível de realismo de uma cena está associado a vários aspectos visuais como textura, sombreamento, reflectância e profundidade da cena. Alguns aspectos diminuem o caráter fotorrealista da imagem:

- uma baixa resolução das malhas poligonais deixa traços visíveis de poligonalidade;
- a iluminação de fontes pontuais e isotrópicas pode não simular efeitos como transbordamento de cores;
- modelos clássicos de reflectância, como Phong [55], não possibilitam efeitos como a translucidez;
- métodos de renderização como o traçado de raios, podem não simular profundidade de campo (*depth of field*).

O desenvolvimento de técnicas que simulam esses efeitos, em conjugação com o aumento de poder computacional, impulsionaram a área de criação de imagens fotorrealistas.

A simulação do transporte da luz da cena, presente na etapa de renderização, é um processo de grande carga de processamento computacional. A simulação de efeitos como inter-reflexão entre os objetos e a utilização de traçado de raios a partir de múltiplas fontes aumentam ainda mais esta carga. Para amenizar este problema, abordagens de otimização de desempenho como renderização independente nos três canais de cores são normalmente

aplicadas. Uma etapa adicional, posterior à renderização, é a simulação de operações aplicadas por câmeras digitais. Equipamentos de alto desempenho são determinantes na diminuição do tempo e na qualidade do produto final.

2.2 Classificadores

Uma ferramenta de reconhecimento de padrões, ou simplesmente classificador, tem como objetivo estabelecer um critério de classificação baseado num conjunto de dados de referência. Diferentemente do casamento de padrões, o classificador não busca identificar igualdade e, sim, similaridade estatística entre os dados. Enquanto o olho humano é eficaz em determinados aspectos [25], os classificadores têm papel importante na classificação automática de dados.

Basicamente, os classificadores podem ser divididos em dois tipos: *supervisionados* e *não supervisionados*. No caso em que a classificação é *não-supervisionada*, o classificador recebe o conjunto de dados e tenta identificar as similaridades presentes com o intuito de formar grupos. O classificador tem pouca ou nenhuma informação *a priori* sobre os dados. Abordagens conhecidas incluem *k*-médias, agrupamento hierárquico e mapas auto-organizados. Um sistema de aprendizado *supervisionado*, por outro lado, age sob o suporte de um *oráculo* o qual identifica previamente os objetos. A escolha dos dados na etapa de treinamento é fundamental para a criação de um classificador eficaz. A obtenção de um conjunto de dados que se aproximem do real possibilita que o classificador possa construir uma função discriminante que se adapte ao cenário real. Na literatura, há uma série de classificadores supervisionados, entre eles a Máquina de Vetores de Suporte (*Support Vector Machines – SVM*), a árvore de decisão e o *Bayes* ingênuo.

Dois métodos de Reconhecimento de Padrões são largamente utilizados em trabalhos na área de identificação de ICGs: Análise Discriminante Linear (*Linear Discriminant Analysis – LDA*) [20] e SVM. A LDA cria uma superfície de classificação linear e sua implementação é relativamente simples. O SVM utiliza superfícies de classificação *n*-dimensionais e sua implementação é mais complexa. O SVM é um método de classificação supervisionado que busca o hiperplano ótimo de separação entre duas classes. Elementos que ficam na borda são chamadas de *vetores de suporte* e a região entre essas margens é o hiperplano ótimo [49]. Elementos no lado *errado* da margem são ponderados negativamente para reduzir sua influência. Quando não é possível encontrar um hiperplano linear, os dados são projetados de forma implícita em espaços de maiores dimensões até que os elementos tornem-se linearmente separáveis.

2.3 Curvas ROC

As curvas ROC (*Receiver Operation Characteristic*) representam um modo gráfico de avaliação, organização e seleção de sistemas de diagnóstico e/ou predição e permitem o cálculo de diversas medidas para avaliação. Gráficos ROC foram originalmente utilizados em detecção de sinais para se avaliar a qualidade de transmissão de um sinal em um canal com ruído.

A análise ROC foi introduzida em Aprendizagem de Máquina como uma ferramenta útil para a avaliação dos modelos de classificação [60], particularmente útil em domínios nos quais existe uma desproporção entre as classes ou quando se deve levar em consideração diferentes custos/benefícios para os diferentes erros/acertos de classificação [60]. Alguns autores da área de identificação de imagens geradas por computador utilizam as curvas ROC na etapa de validação dos métodos propostos.

Capítulo 3

Trabalhos Relacionados

of de alta qualidade emergiram com a evolução da Computação Gráfica. Métodos para distinguir IFGs e ICGs são de meados de 2000 [51, 51] e frequentemente são extensões de conceitos de áreas correlatas [11, 39, 44, 51].

A maior parte das propostas existentes para distinguir ICGs e IFGs é dividida em duas etapas:

- identificação e extração de características que evidenciem as diferenças entre as duas classes;
- classificação das imagens baseada no conjunto de características obtido.

Além dos conceitos já explorados no problema de identificação de IFG e ICG, há ainda muitos descritores de áreas correlatas que são promissores na aplicação em nosso problema. Como buscamos uma gama variada de descritores neste trabalho, optamos por prospectar descritores de diferentes áreas de Processamento de Imagens.

A principal diferença entre os diversos métodos da literatura consiste na escolha das características para descrever uma imagem (descriptor). A eficácia deste processo é fundamental para a boa acurácia de um método. Analisamos descritores já existentes quanto à aplicabilidade e a eficácia no problema de identificação de ICGs e IFGs. Áreas como Recuperação de Imagens por Conteúdo (*Content-Based Image Retrieval – CBIR*) apresentam ampla pesquisa na área de extração de características [38, 67, 71] e, portanto, apresentam potenciais fontes para o nosso trabalho. A caracterização de uma imagem pode se basear em diversos critérios: em histogramas de cores [9], textura [21, 22], forma [41], bordas [37], malhas (*patches*) [37], superfície [72], entre outros.

Apresentamos, a seguir, a seção *Métodos do Estado da Arte*, onde analisamos trabalhos relevantes de distinção entre IFG e ICG. Na seção *Métodos de Áreas Relacionadas*, expomos conceitos já explorados em outras áreas e ainda não aplicados no problema de IFG

versus ICG. Por último, na seção *Combinação de métodos*, analisamos uma abordagem existente de combinação de descritores.

3.1 Métodos do Estado da Arte

Lyu e Farid (2005) [44]

Os autores propuseram a decomposição da imagem utilizando transformada *wavelet*. De acordo com [6], os coeficientes *wavelets* de subbandas de imagens naturais apresentam uma distribuição Laplaciana Generalizada, a qual é caracterizada por um grande pico em zero e caudas longas e simétricas. Uma interpretação para esta distribuição em uma imagem real pode ser entendida pela presença de grandes áreas de regiões homogêneas e transições abruptas (bordas). Estas informações são utilizadas para verificar se uma imagem segue o padrão de uma IFG.

A decomposição segue o método Filtros Espelho de Quadratura, de Vaidyanathan [73] e propõe a decomposição em n escalas, três orientações (vertical, horizontal e diagonal) e três cores (vermelho, verde e azul).

Para caracterizar a distribuição dos coeficientes, os autores obtiveram quatro estatísticas de ordem para cada subbanda (média, variância, obliquidade e curtose) da imagem original. Além das tradicionais média e variância para caracterização de uma distribuição, a utilização de curtose visa medir o quanto os valores dos picos diferem do restante da distribuição. A obliquidade avalia a assimetria da distribuição de probabilidade.

Estas características, no entanto, não capturam adequadamente a correlação dos coeficientes nas escalas, bandas de cor e orientações. A correlação dos coeficientes da imagem através de várias escalas e orientações reflete características de bordas, por exemplo. O trabalho propõe a identificação desta correlação através de um preditor linear de erros, o qual leva em conta os *pixels* adjacentes da mesma subbanda, subbandas de outras escalas e de outras orientações. Cada coeficiente é associado a um erro, novamente caracterizado pelas quatro estatísticas de ordem superior: média, variância, curtose e obliquidade. A dimensão do vetor de características é dado pelo produto do número de bandas de cor, número de escalas, número de orientações e pelo número de estatísticas de ordem.

A partir deste modelo preditivo de coeficientes, os autores estimaram o coeficiente de cada subbanda e os erros associados a cada coeficiente para cada canal de cor e obtiveram um conjunto de 216 estatísticas. Para uma taxa de falsos negativos fixada em 1,2%, os resultados atingiram uma acurácia de 54,6% e 66,8% com os classificadores LDA e SVM, respectivamente.

Chen et al. (2007) [11]

Chen et al. seguiram a linha da decomposição *wavelet*. O método, diferentemente de [44], não calcula as quatro estatísticas da distribuição dos coeficientes *wavelets* e os erros associados a cada estimativa. Para caracterizar as distribuições, apenas os momentos estatísticos das funções características de ambas distribuições são obtidos. Um total de 78 características é extraído de cada componente de cor da imagem, resultando num vetor de características de $78 \times 3 = 234$ dimensões. Uma das propostas do trabalho é investigar a influência do sistema de cores da imagem na extração de características, em particular, o espaço de cores HSV (*Hue Saturation Value*).

Os autores utilizaram um classificador SVM na validação dos métodos e os dados utilizados foram de fontes diversas. Para a análise dos resultados, os autores construíram uma curva ROC. A acurácia reportada é de 82,1% no espaço de cores HSV e de 76,9% no espaço de cores RGB. A precisão em HSV é de cinco pontos percentuais superior à precisão em RGB. O resultado evidencia a importância da representação de cores para os descritores.

Wang e Moulin (2006) [74]

Posteriormente, Wang e Moulin seguiram a mesma linha e propuseram um modelo baseado nos histogramas dos coeficientes *wavelets* da imagem. Para melhorar o desempenho, filtros passa-banda são aplicados ao histograma normalizado, de modo a caracterizar o histograma com um menor número de informações. Para a validação do método, os autores utilizaram um classificador LDA e uma base de imagens pessoais. Por critério de comparação, Wang e Moulin testaram o mesmo conjunto de dados com o método de [44]. O resultado atingido foi próximo de 100% de acurácia, para ambas as classes de imagens, a uma taxa de falsos negativos de 0,1%. Os resultados obtidos pelo método de [44] sobre a mesma base de imagens foram de mesma acurácia para IFGs e acurácia inferior para ICGs. De acordo com a análise de [50], o método é mais preciso que a abordagem de [44] e obteve uma significativa redução no tempo computacional nos tempos reportados.

Ng et al. (2005) [51]

Ng et al. propuseram um modelo de imagem baseado em propriedades geométricas, o qual leva em conta os processos físicos de formação das IFGs e ICGs. O trabalho define dois conceitos distintos e independentes para avaliação de seu método: (1) autenticidade do processo de aquisição; (2) autenticidade da cena. A autenticidade da *aquisição* visa determinar se a imagem foi obtida por equipamento de aquisição digital ou foi gerada por computador, independente do conteúdo da imagem. A autenticidade da *cena*, por sua vez, busca a origem do conteúdo, independente da imagem final. O objetivo da definição

destes conceitos é a categorização de classes que não se encaixam adequadamente na divisão tradicional IFG versus ICG. Por exemplo, uma dessas classes é a de IFG com conteúdo fotorrealista, isto é, uma fotografia de uma imagem gerada por computador (recaptura), que pode ser isolada com as definições propostas pelos autores.

Baseando-se nessas diferenças, o trabalho propôs a análise da imagem em duas escalas distintas. Na melhor escala, a geometria da cena pode ser descrita por uma dimensão fractal local e por retalhos (*patches*) locais. Na outra escala, a geometria é melhor descrita por uma linguagem de geometria diferencial. O gradiente de superfície, a segunda forma fundamental e os vetores de fluxo Beltrami são calculados. Tais características são extraídas pelo método de *Momento dos Corpos Rígidos*, o qual caracteriza a distribuição conjunta de várias dimensões. Esta informação é utilizada por um classificador SVM na etapa de treinamento. O método foi validado levando-se em conta as definições propostas de autenticidade.

O mesmo conjunto de dados foi utilizado para avaliar o método de Lyu e Farid [44], para validação e comparação com outros resultados. A média das acurácias foi de 83,5%, superior à média dos resultados obtidos por [44] aplicado ao mesmo conjunto de dados (80,3%). De acordo com os próprios autores, os resultados indicam que o método é eficaz na classificação de *aquisição*, embora não seja na classificação de *cena*.

Rocha e Goldenstein (2008) [63]

Rocha e Goldenstein propuseram uma abordagem baseada nos bits menos significativos (*Least Significant Bits – LSB*) de imagens. O bit menos significativo de um número, em sua representação binária, é o dígito binário da extrema direita. Os LSBs de uma imagem são os LSBs dos *pixels* da imagem. Este conjunto constitui um canal de mensagens popular na área de esteganografia [66]. Os bits menos significativos podem guardar dados que desejamos transmitir camufladamente. Em uma imagem de formato JPEG, trocar os bits menos significativos pode mudar a intensidade de um *pixel* em no máximo 1% [66]. Estatísticas sobre os LSBs de uma imagem podem ser utilizadas como descritores.

A principal hipótese do método em [66] é a de que as duas classes de imagens, quando “perturbadas” a partir de uma determinada forma de alteração dos bits menos significativos (*Progressive Randomization – PR*), apresentam padrões de comportamento distintos. Isto é suficiente para a distinção entre as classes de imagens. O método cria, progressivamente, n imagens a partir da imagem original, diferindo no grau de perturbação (taxa de LSB a serem alterados). Um vetor de características k -dimensional (no caso, 96-D) é formado e fornecido a um classificador SVM. Os resultados foram expressos em frequência de acertos por categoria: 98,7% de acurácia na identificação de IFG e 95,7% na de ICGs. Os autores salientam que os resultados foram significativamente melhores que os de [44], em particular, na identificação de imagens fotográficas (32 pontos percentuais acima).

Duas limitações do método, de acordo com os autores, são a exigência de muitos exemplos de treinamento (~40.000 imagens de exemplo) e a ineficácia em casos de imagens de recaptura (fotografias de imagens geradas por computador).

Dehnie et al. (2006) [15]

Dehnie et al. basearam-se na hipótese de que IFGs obtidas por um mesmo processo de aquisição preservam traços comuns [43]. Estes traços residuais de equipamentos de aquisição de imagem podem ser utilizados para distinguir uma IFG de uma ICG. Para obter o ruído presente na imagem, aplica-se o filtro adaptativo local proposto por Lukas et al. [43] e subtrai-se a imagem filtrada da imagem original. O filtro é derivado de um modelo estatístico que leva em conta a dependência entre coeficientes *wavelets* de *pixels* adjacentes presente em imagens fotográficas.

A média dos ruídos de todas as imagens de uma mesma classe dá origem ao padrão de ruídos de referência daquela classe. A classificação é feita pelo valor da correlação entre a imagem residual e o padrão de referência. Parte das imagens fotográficas foi proveniente de duas câmeras de modelos conhecidos e o restante, de modelos diversos. As ICGs foram obtidas na *Internet* e foram divididas em dois grupos: o primeiro, de imagens geradas por *Maya* e o segundo por *3ds Max*. Os resultados indicam que as fotografias digitais apresentam baixa correlação com os erros padrões de ambos os produtos de *software*. O desempenho médio do método foi de 72% [50].

Dirik et al. (2007) [17]

Dirik et al. propuseram um método baseado no processo de interpolação de cores presente em câmeras digitais. Embora processamentos posteriores sejam aplicados, a identificação do algoritmo aplicado na etapa de mosaico ainda é possível. Os autores propuseram que, para classificar uma imagem como IFG, é suficiente identificar algum processo de interpolação de cores.

O método admite como hipótese que, se uma IFG interpolada por um filtro Bayer [3] for reinterpolada pelo mesmo filtro, ela terá significativamente menos mudanças do que se fosse reinterpolada por um outro filtro. Ou seja, se uma das reinterpolações tiver um erro quadrático médio discrepante do restante, a imagem é considerada ICG e, caso contrário, ela pode ser caracterizada como uma IFG.

Dirik et al. exploraram, adicionalmente, a interferência da lente no processo de aquisição de uma fotografia digital. A aberração cromática está associada à diferença entre os índices de refração para os diferentes comprimentos de onda da luz incidente na lente de aquisição. Ela causa um desalinhamento entre os canais de cores da imagem. Quando há alinhamento entre os canais de cores, existe dependência entre eles. Uma

imagem com informações mútuas altas preserva um alinhamento entre os canais de cores e, portanto, pode ser identificada como não tendo traços de aberração cromática.

Um total de $1.200 \times 3 = 3.600$ imagens foi reunido. Metade do conjunto de dados foi utilizada para treinamento por um classificador SVM e a outra metade foi usada para testes. Os resultados informados têm acurácias médias de 98,1%, 89,3% e 99,6% utilizando características de interpolação de cores, aberração cromática e características *wavelets*, respectivamente. Fazendo a junção das características de demosaico e *wavelets*, o método obteve uma acurácia de 99,9%. Para testar a invariância das características de aberração cromática de uma imagem perante compressões JPEG, as imagens foram comprimidas a taxas de fator de qualidade de 50% e 75%, resultando em 6.000 imagens. O teste foi feito utilizando evidências de aberração cromática e atingiu uma acurácia de 90%. O resultado valida a hipótese de que as características de aberração cromática são relevantes, mesmo para imagens com altas taxas de compressão.

Gallagher e Chen (2008) [28]

Gallagher e Chen também utilizaram traços de interpolação de cores Bayer para distinguir imagens fotográficas digitais e imagens geradas por computador. O algoritmo não faz nenhuma hipótese sobre a identidade, ou ainda, sobre a linearidade do algoritmo de demosaico, apenas baseia-se na hipótese de que o conjunto dos *pixels* interpolados de uma imagem apresenta variâncias distintas do conjunto de *pixels* originais.

Para verificar sua hipótese, o método busca uma periodicidade nas variâncias das diagonais da imagem. Inicialmente, um filtro passa-altas é aplicado à imagem para realçar a periodicidade, quando presente. Espera-se que, no caso de uma amostra de *pixels* que sofreram interpolação, as variâncias sejam periódicas entre diagonais diferentes e constantes ao longo delas. Posteriormente, a variância de cada diagonal é estimada pelo método da Máxima Verosimilhança. Ao final, tem-se um vetor com o valor da variância para cada diagonal, cujo sinal é analisado no domínio da frequência (transformada de Fourier) quanto à periodicidade das variâncias. Um valor de corte é estabelecido.

Um conjunto de 2.400 imagens foi utilizado, do qual 1.600 eram IFGs e 800 ICGs. A abordagem conseguiu um resultado de 98,4% de acurácia. Aplicando pós-compressão JPEG, o método apresentou um resultado de aproximadamente 82%. De acordo com os autores, esperava-se que o método necessitasse de imagens grandes para um maior espaço amostral (maior número de diagonais), porém, mesmo com imagens pequenas, de 64×64 *pixels*, o método obteve uma acurácia de 66%. A eficácia do método, contudo, restringe-se a imagens obtidas diretamente de câmeras digitais, isto é, que não sofreram processamento posteriores à aquisição (como reescala), pois estas destruiriam as características associadas à interpolação de cores.

Li et al. (2010) [39]

Li et al. propuseram um método que utiliza importantes ideias dos métodos de [11] e de [44]. Os autores optaram por trabalhar com o espaço de cores HSV. Em sua implementação, contudo, os autores não utilizam a decomposição *wavelet*.

Cada imagem é convertida para o espaço de cores HSV por uma transformação linear dada como

$$H = \begin{cases} 60 \left(\frac{G-B}{\delta} \right) + 0, & \text{se } MAX = R \text{ e } G \geq B \\ 60 \left(\frac{G-B}{\delta} \right) + 360, & \text{se } MAX = R \text{ e } G < B \\ 60 \left(\frac{B-R}{\delta} \right) + 120, & \text{se } MAX = G \\ 60 \left(\frac{R-G}{\delta} \right) + 240, & \text{se } MAX = B \end{cases}$$

$$S = \begin{cases} \frac{\delta}{MAX} & \text{se } MAX \neq 0 \\ 0, & \text{se } MAX = 0 \end{cases} \quad (3.1)$$

$$V = MAX$$

em que R , G e B são as intensidades de vermelho, verde e azul respectivamente e $MIN = \min(R, G, B)$, $MAX = \max(R, G, B)$ e $\delta = (MAX - MIN)$.

Em seguida, os autores propõem a análise da imagem em duas escalas: original e reduzida. Em cada uma das escalas, aplica-se uma aproximação para derivada de segunda ordem. Esta diferenciação é obtida em quatro direções distintas. Estatísticas (variância e curtose) sobre estes resultados são calculados. Embora esta diferenciação aproximada avalie a correlação em uma dada direção, ela não captura a diferença entre direções distintas. Para tanto, eles utilizam um preditor linear, análogo ao proposto por [44]. A partir de cada valor predito, o logaritmo do erro é calculado.

Os autores partem da hipótese que, se IFGs seguem o modelo linear proposto, a distribuição destes erros serve como critério de categorização das imagens. As quatro primeiras estatísticas de ordem são utilizadas para caracterizar esta distribuição. Para cada escala, canal de cor RGB e orientação, duas estatísticas da matriz de diferenciação e quatro estatísticas da distribuição de erros são obtidas. O método fornece um total de 144 características.

A Tabela 3.1 compara os métodos expostos.

Publicação	Metodologia	Acurácia	Vantagens	Desvantagens
------------	-------------	----------	-----------	--------------

Lyu e Farid (2005) [44]	Decomposição <i>wavelet</i> .	~80,3%	Implementação publicamente conhecida.	Características extraídas não se baseam em aspectos visuais [51].
Ng et al. (2005) [51]	Características geométricas.	~83,5%	Análise estatística detalhada.	Baixa eficácia em distinguir imagens fotorealistas de paisagens;
Wang e Moulin (2006) [74]	Decomposição <i>wavelet</i> .	~100,0%	Redução do tempo computacional com relação a [44].	Resultados disponibilizados apenas para uma taxa de falsos positivos em específico.
Dehnie et al. (2006) [15]	Detecção do padrão de ruído da câmera.	~72,0%	Implementação simples.	Validação do método com banco de dados pequeno e de pouca heterogeneidade.
Chen et al. (2007) [11]	Decomposição <i>wavelet</i> .	~82,1%	Conversão entre espaços de cores é simples.	Vetor de características redundante.
Dirik et al. (2007) [17]	Detecção de demosaico e aberração cromática.	~99,9%	Invariância à compressão JPEG.	Resultados obtidos com um conjunto de dados “simples”.
Gallagher et al. (2008) [28]	Detecção de demosaico.	~98,4%	Tolera compressão JPEG.	Variante à mudança de escala.
Rocha e Goldenstein (2008) [61, 62, 64]	Randomização Progressiva.	~97,2%	Metodologia clara.	Exige grande quantidade de dados na fase de treinamento;
Sankar et al. (2009) [68]	Interpolação de textura; Combinação de características.	~90,3%	Identifica simulação de uma IFG em ICGs.	Precisão reportada tem análise estatística não robusta.

Tabela 3.1: Comparativo entre os principais métodos existentes na literatura.

Todos os métodos citados foram estudados na literatura de forma individual. Uma comparação entre diversas abordagens de distinção entre IFG e ICG, no qual o ambiente de execução e os dados de treinamento/teste são os mesmos, é inexistente até o momento.

3.2 Métodos de Áreas Relacionadas

Há uma série de trabalhos de áreas correlatas ao problema de identificar IFGs e ICGs que podem ser diretamente estendidos em nosso escopo. Neste sentido, analisamos trabalhos que utilizamos como base para proposição de novos descritores aplicados ao problema de IFG versus ICG.

Fractalidade [40]

Um conceito importante na área de segmentação de imagens fotográficas é a fractalidade, proposto por Mandelbrot [48]. A partir da técnica de contagem de caixas [40], podemos estimar a dimensão fractal de uma imagem. Cobrimos um conjunto limitado $E \in \mathbb{R}^n$ com caixas (quadrados) de lado ε e interiores disjuntos. Seja $N_\varepsilon(E)$ a quantidade destes quadrados. Suponha que E contém um número infinito de pontos como em uma curva ou uma superfície, e que $N_\varepsilon(E)$ tende a $+\infty$ quando ε tende a 0. A dimensão da caixa Δ caracteriza a taxa deste crescimento. Δ é o número real tal que

$$N_\varepsilon(E) \simeq \left(\frac{1}{\varepsilon}\right)^\Delta \quad (3.2)$$

assumindo que este número existe. Genericamente, para todo E limitado, temos

$$\Delta(E) = \limsup_{\varepsilon \rightarrow \infty} \frac{\log N_\varepsilon(E)}{|\log \varepsilon|} \quad (3.3)$$

como limite superior e

$$\Delta(E) = \liminf_{\varepsilon \rightarrow \infty} \frac{\log N_\varepsilon(E)}{|\log \varepsilon|} \quad (3.4)$$

como o limite inferior. Alguns autores definem que a dimensão por contagem de caixas só é definida quando os limites superior e inferior coincidem. A definição dos autores não exige a utilização de caixas como elemento de medida. Uma definição alternativa utiliza círculos de raio ϵ , ao invés das caixas.

X-lets [7, 19, 47]

Na área de Processamento de Imagens, há uma série de transformações aplicáveis sobre um conjunto de dados. Transformadas bidimensionais podem ser analisadas sob diversos aspectos:

- multirresolução: decomposição em múltiplas resoluções;
- localidade: localização em espaço e frequência;

- redundância: os coeficientes apresentam informações redundantes;
- direcionalidade: as direções vertical, horizontal e diagonal podem não capturar de forma eficaz os detalhes em todas direções da imagem;
- anisotropia: as janelas devem ter diferentes formatos para capturar todas as nuances de uma imagem.

As transformadas *wavelets* são carentes das duas últimas características, as quais também são responsáveis pela distinção entre as transformadas *curvelets* e *contourlets*. As transformadas *curvelets* [7] foram desenvolvidas inicialmente no domínio contínuo. Filtros multiescalas são aplicados, seguidos por transformadas *ridgelets* [18] em cada subbanda. Pouco depois, uma adaptação foi proposta para o domínio discreto. Uma nova implementação das *curvelets* foi posteriormente proposta pelos mesmos autores sem utilizar as transformadas *ridgelets*. Logo após a introdução das *curvelets*, alguns pesquisadores desenvolveram algoritmos numéricos para a sua implementação e relataram uma série de sucessos práticos [7]. Essas implementações são baseadas na construção original, a qual utiliza um passo de pré-processamento envolvendo um particionamento fase-espaco seguido por uma transformada *ridgelet* aplicada a blocos de dados no domínio espaço/frequência.

Em [7], os autores redesenharam as *curvelets* de modo a tornar sua implementação mais simples. Dois métodos foram propostos: Transformada Rápida de Fourier não igualmente espaçada e via Envoltório Fourier. Os métodos diferem essencialmente pela escolha da grade espacial utilizada em cada escala e ângulo. Ambas retornam uma tabela de coeficientes *curvelets* indexados por um parâmetro de escala, um parâmetro de orientação e um parâmetro de localização espacial. Para uma matriz de tamanhos $n \times n$, ambas implementações têm complexidade $O(n^2 \log n)$. As transformadas são invertíveis, com algoritmos de inversão de semelhante complexidade.

As transformadas *contourlets* [19], assim como *curvelets*, são transformadas multirresolução e direcionais. Diferentemente das *curvelets*, foram formuladas diretamente no domínio discreto. A decomposição piramidal por banco de filtros é obtida pela combinação da pirâmide Laplaciana e um banco de filtros direcionais.

As transformadas *shearlets* são mais recentes [36] e são bastante similares em construção às *curvelets*. Sua base matemática é sólida e, dada sua construção, também possibilita uma análise multirresolução.

A transformada *shearlets* contínua utiliza uma dilatação com dois parâmetros. Esta dilatação consiste no produto de uma matriz de escala parabólica por uma de cisalhamento. Os coeficientes *shearlets*, portanto, dependem do parâmetro de escala, $a > 0$, de parâmetros de cisalhamento, $s \in \mathbb{R}$, e de um parâmetro de translação $t \in \mathbb{R}^2$. Definimos

os coeficientes *shearlets* por

$$Sh_f(a, s, t) = \langle f, \Psi_{a,s,t} \rangle \quad (3.5)$$

A transformada é definida por

$$\Psi_{a,s,t}(x) = a^{-3/4} \Psi((M_{a,s}^{-1}(x - t))), \quad (3.6)$$

$$\text{e } M_{a,s} = \begin{bmatrix} a & -sa^{1/2} \\ 0 & a^{1/2} \end{bmatrix} \quad (3.7)$$

A *shearlet* mãe Ψ é definida pelo produto tensorial

$$\Psi(\xi_1, \xi_2) = \Psi_1(\xi_1) \Psi_2(\xi_2/\xi_1) \quad (3.8)$$

em que Ψ_1 é uma *wavelet* e Ψ_2 é uma função suave e de suporte compacto. Para obter as *shearlets* discretas, amostramos os parâmetros a , s e t

$$a_j = 2^j \quad (j \in \mathbb{Z}), \quad (3.9)$$

$$s_{j,k} = ka_j^{1/2} = k2^{j/2} \quad (k \in \mathbb{Z}) \quad (3.10)$$

$$t_{j,k,m} = M_{a_j, s_{j,k}} \quad (m \in \mathbb{Z}^2) \quad (3.11)$$

e escolhemos funções Ψ_1 e Ψ_2 discretas.

Matriz de Coocorrência [31]

A matriz de coocorrência (*Gray Level Cooccurrence Matrix – GLCM*) [31] é um descritor de textura amplamente utilizado na área de Análise de Imagens. Diferentemente das medidas que são calculadas imediatamente a partir dos valores da imagem original (medidas de primeira ordem), a GLCM considera a relação entre grupos de *pixels*.

Descritores baseados em matrizes de coocorrência são obtidos em duas fases. Primeiramente, a matriz de coocorrência propriamente dita é calculada. Definem-se inicialmente operadores direcionais (vide Figura 3.1). Podem-se escolher operadores de modo que sejam não-redundantes, isto é, que a partir da matriz de ocorrência de um operador não seja possível obter os dados de outro operador. Os operadores definem como a imagem deverá ser percorrida. Uma configuração “*pixel* central - *pixel* vizinho” define uma entrada da matriz. O número de linhas e colunas da matriz é igual ao número de tons de cinza da imagem.

Se cada *pixel* da imagem puder assumir n valores em cada canal de cor, então cada matriz resultante terá tamanho $n \times n$. Por exemplo, imagens que assumem valores discretos de 1 a 256 estão associadas a matrizes resultantes de tamanho de $256 \times 256 = 65.536$.

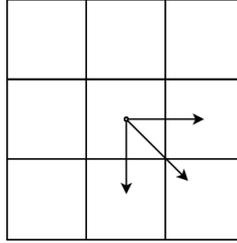


Figura 3.1: Três possíveis operadores direcionais da matriz de coocorrência. Pixels central-direita, central-diagonal e central-abaixo. Para cada *pixel* da imagem, computam-se os valores dos *pixels* de acordo com estes operadores.

Na segunda etapa, as matrizes derivam estatísticas que são utilizadas como descritores. Existem várias medidas que são utilizadas ao invés de utilizar a matriz inteira. Haralick et al. [31] propuseram 14 medidas distintas. Clausi [12] analisou a correlação entre texturas propostas por Haralick et al. e concluiu que algumas das propriedades são correlacionadas. Ele concluiu ainda que, entre as medidas propostas, há no máximo cinco independentes: contraste, dissimilaridade, momento da diferença inversa normalizada e diferença inversa normalizada. Clausi verificou também que as características mais eficazes em seus conjuntos de testes eram entropia, contraste e correlação e homogeneidade.

Histograma de Gradientes Orientados [14]

A técnica de Histograma de Gradientes Orientados (*Histogram of Oriented Gradients – HOG*) conta as ocorrências de gradientes orientados nas regiões de uma imagem [14]. HOG difere das outras técnicas por ser calculado sobre uma malha de células uniformemente espaçadas e utilizar normalização sobre regiões com sobreposição.

Na aplicação do operador HOG, o primeiro passo é a correção gama. Aplica-se um detector de gradientes e, para cada *pixel* da imagem, determina-se a direção do gradiente. Para cada região de tamanho fixo da imagem (*célula*), calcula-se a frequência intervalar de ocorrências de cada direção do gradiente e constrói-se um histograma de orientações de gradientes. As células podem ter forma arbitrária. Os autores implementaram o método com células em forma circular e retangular. Cada *pixel* da célula tem sua direção ponderada pela intensidade do gradiente. As células são agrupadas em regiões maiores, os blocos, etapa na qual os histogramas são concatenados e normalizados. O vetor de características resultante é a concatenação dos histogramas dos blocos.

Histograma de Coeficientes *shearlets* [70]

O histograma de coeficientes *shearlets* (*Histogram of Shearlets Coefficients – HSC*) [70] é uma abordagem que tira proveito das virtudes de direcionalidade da transformada *shear-*

lets para determinar a distribuição de bordas da imagem. O trabalho tem grande analogia com o descritor HOG [14].

Primeiramente aplica-se uma decomposição multiescala em coeficientes *shearlets*, com o intuito de capturar informações da imagem em diferentes orientações e escalas.

Em seguida, para cada escala, um histograma com o mesmo número de intervalos que o número de orientações é computado. O valor referente a cada intervalo do histograma é obtido a partir da soma dos valores absolutos dos coeficientes *shearlets*. Por fim, os histogramas de todos níveis são concatenados e normalizados.

De acordo com os autores, o descritor proposto apresenta melhores resultados que HOG em, pelo menos, duas áreas: classificação de texturas e identificação de faces.

Padrões Binários Locais [52]

O sistema visual humano é capaz de interpretar cenas praticamente acromáticas, como por exemplo, a baixos níveis de iluminação. A cor atua apenas como uma sugestão para interpretações mais ricas. Mesmo quando a informação de cor é distorcida, por exemplo, devido à cegueira de cor, o sistema visual ainda funciona satisfatoriamente [24]. Intuitivamente, isso sugere que, pelo menos, para o nosso sistema visual, o contraste e textura são fenômenos distintos. No entanto, a utilização conjunta de características de contraste e de textura é popular na Análise de Imagens.

O operador de Padrões Binários Locais (*Local Binary Patterns – LBP*) foi introduzido como uma medida complementar ao contraste de uma imagem [52]. O valor é obtido pela soma dos valores limiarizados pelo *pixel* central (Equação 3.13), ponderados pela sua posição em relação ao *pixel* central (Equação 3.12). O algoritmo inicialmente proposto trabalha com raio 1 e vizinhança 8-conexa. Veja a Figura 3.2.

$$Y_i = \begin{cases} 1, & \text{se } X_i > X_0 \\ 0, & \text{caso contrário} \end{cases} \quad (3.12)$$

$$LBP(X_0) = 1y_1 + 2y_2 + 4y_3 + 16y_4 + 128y_5 + 64y_6 + 32y_7 + 8y_8 \quad (3.13)$$

A maior limitação a esta proposta de padrões binários locais é o suporte a apenas áreas pequenas e fixas ao redor da imagem. Imagens capturadas numa vizinhança 8-conexa, por exemplo, podem não capturar detalhes estruturais de escalas maiores. Além disso, o operador não é robusto a mudanças sutis como mudanças na direção da iluminação [52]. Posteriormente, os autores de [46] propuseram um operador de padrões binários multi-escala. Sua implementação é simples e consiste na aplicação do operador em múltiplas vizinhanças.

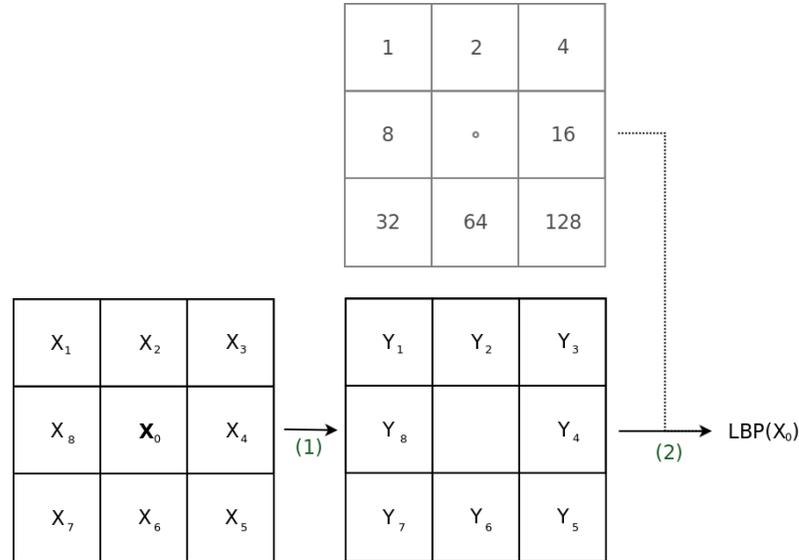


Figura 3.2: Cálculo do Padrão Binário Local de um *pixel* X_0 e uma vizinhança de raio 1. (1) Limiarização dos vizinhos de acordo com o valor de X_0 (Equação 3.12). (2) Soma ponderada pelos pesos pré-definidos (Equação 3.13).

Demoaico [58]

Popescu e Farid [58] propõem uma forma de detectar demoaico e de discernir diferentes tipos de interpolação empregados por modelos diferentes de câmeras. Os autores utilizam este método para identificação de adulteração em uma imagem.

Suponha que uma imagem tenha sido interpolada por um determinado algoritmo de interpolação. Os *pixels* de um mesmo canal de cor apresentarão um padrão de correlação diferente naquela região adulterada. Desta forma, a amostragem por blocos pode ser utilizada para identificar regiões que sofreram alterações.

Por simplicidade, os autores assumem que o algoritmo de interpolação, embora desconhecido, seja linear. Se o método for capaz de descobrir qual tipo de correlação existe entre os *pixels*, o que no método significa determinar os parâmetros, então será possível determinar quais *pixels* foram interpolados para obtenção de seus valores. Por outro lado, se for possível identificar quais são os *pixels* interpolados, pode-se determinar diretamente os parâmetros de correlação entre eles. Como num caso real não se tem acesso a nenhum deles, o método utiliza o algoritmo de Esperança/Maximização (EM) iterativo para estimar os parâmetros desconhecidos. O processo envolve duas etapas:

- etapa E: cálculo da probabilidade de que cada amostra pertença ao modelo;
- etapa M: estimativa de um modelo de interpolação.

Valores iniciais são utilizados e as iterações são executadas até que o critério de estabilidade seja atingido.

O método apresentou resultados da ordem de 97% de acurácia na identificação de modelos de interpolação. A classificação foi feita por LDA. Cada rodada particionou o conjunto em duas partes, sendo 90% para treinamento e 10% para testes.

3.3 Combinação de Métodos

A combinação de abordagens tem como motivação melhorar algum aspecto, como desempenho ou acurácia, quando comparado à cada método isolado. De acordo com o teorema “*There is no free lunch*”, não existe um único classificador que seja ótimo em todos os cenários de classificação [57, 76]. Para isso, deveríamos ter uma regra determinística para identificar o classificador ideal para natureza de dados.

A seguir, apresentamos abordagens de combinação [5] de características e de classificadores.

Concatenação

A concatenação é um método de fusão de características e consiste na junção das características de cada abordagem num único conjunto de características resultante.

Bagging [4]

O método *bagging* cria, a partir de um conjunto de dados, um grupo de modelos de classificação a serem combinados. O método se baseia na hipótese de que amostragens do conjunto de treinamento possibilitam classificadores mais específicos. Os dados de treinamento são obtidos através da amostragem aleatória com reposição do conjunto de treinamento. As diversas classificações assim obtidas são reunidas através de algum critério, como a votação.

Adaboost [27]

O método *Adaboost* é similar ao *bagging*, pois constrói diversos modelos a partir de diferentes conjuntos de treinamento. Porém, *adaboost* combina os classificadores através de uma abordagem serial: ao fim de cada iteração, os padrões classificados incorretamente recebem maior ponderação para iteração seguinte.

Random Subspace Method [33]

O método RSM (*Random Subspace Method*) cria aleatoriamente subespaços do conjunto de características original e constrói um classificador para cada um destes cenários. Em cada subamostragem, o algoritmo extrai um subconjunto do espaço de características restantes (amostragem sem reposição) de tamanho aleatório. Em cada etapa, um classificador é treinado. Evidências experimentais indicam que RSM tem bons resultados com espaços de características redundantes e de altas dimensionalidades.

Error-Correcting Output Coding [1]

O método ECOC (*Error-Correcting Output Coding*) transforma o problema de classificação multiclases em um conjunto de classificações binárias. O método define uma palavra (*code word*) por classe, onde o tamanho de cada palavra é dado pelo número de classificadores. Com o modelo criado, ao receber um objeto a ser classificado, o método gera a palavra correspondente e calcula as distâncias às palavras de referência (classes). A classe cuja palavra fosse mais próxima, sob algum critério de distância, é considerada a classe do objeto.

Behaviour Knowledge Space [1]

O método BKS (*Behaviour Knowledge Space*) armazena as frequências para cada configuração de votos numa tabela de referência. Ao receber um dado a ser classificado, o algoritmo busca na tabela a configuração de classificação do dado e determina a classe com maior frequência. De acordo com os autores, o método garante erro mínimo no conjunto de treinamento. Por outro lado, o algoritmo exige um grande conjunto de treinamento para popular a tabela.

Stacked Combination [75]

A metaclassificação (*Stacked Combination*) utiliza as *probabilidades* das classificações como dados de entrada do classificador resultante. Na fase de treinamento, o classificador recebe as probabilidades obtidas pelos classificadores obtidos a partir dos dados de treinamento. Quando o método recebe um novo objeto, cada um dos classificadores fornece uma probabilidade, a qual representa um elemento do conjunto de características. A partir do conjunto de todas as probabilidades, o metaclassificador realiza sua decisão.

Capítulo 4

Metodologia

A criação de um método de classificação que utiliza reconhecimento de padrões normalmente apresenta duas fases: fase de extração de características e fase de treinamento do classificador. A validação do método envolve uma terceira etapa, ilustrada na Figura 4.1.

Um descritor representa cada imagem por um conjunto ordenado de características. A estrutura de arquivos utilizada é uma matriz de $n \times m$ dimensões, em que n é o número de imagens e m é o número de características (veja Figura 4.2). Como todos os descritores são aplicados sobre o mesmo conjunto de imagens (mesmo n), a dimensão que difere um método de outro é o número de colunas. Considerando os descritores implementados neste trabalho, a menor matriz resultante foi a obtida a partir da fractalidade, com $m = 3$ e a maior foi a obtida a partir de coeficientes *curvelets*, com $m = 2.328$.

A utilização de um grande número de parâmetros enfrenta a chamada *maldição da dimensionalidade* [69], em que todos os dados parecem ser esparsos e o volume de dados necessário para uma análise estatisticamente significativa cresce exponencialmente com o número de dimensões dos dados. Por outro lado, um número pequeno de parâmetros pode resultar numa caracterização precária e ineficaz no contexto de distinção entre ICG e IFG.

Utilizamos ferramentas de reconhecimento de padrões em nosso trabalho, em particular, o SVM. Temos consciência, no entanto, dos riscos associados à utilização de tais ferramentas “caixa-preta” [65]. Para aumentar a robustez dos resultados, aplicamos validações cruzadas em cada método. Uma vez definido o conjunto de imagens a ser utilizado, particionamos nossos conjuntos de entrada em cinco partes. Todas as etapas de classificação utilizaram as mesmas partições de imagens, pois deste modo eliminamos a possibilidade de que a *dificuldade* de um dado conjunto de imagens possa ter favorecido algum método. Como tínhamos $4.850 \times 2 = 9.700$ dados de entrada, criamos cinco partições de tamanhos iguais a 1.940. Não garantimos que cada partição tivesse um mesmo número de ICG e de IFG, uma vez que o processo de criação de partições foi aleatório.

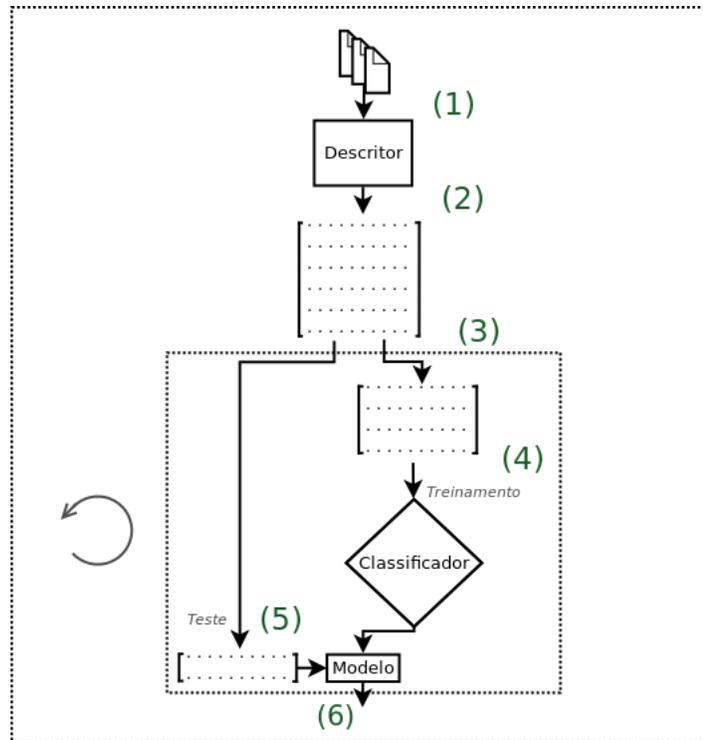


Figura 4.1: Fluxograma de extração e avaliação de um método com validação cruzada. (1) O descritor representa as informações do conjunto de imagens. (2) Obtemos um conjunto de características, representado por uma matriz. (3) A matriz é particionada e separada em dados de treinamento e teste. (4) O conjunto de treinamento é utilizado na criação do modelo. (5) O conjunto de teste é utilizado na validação do modelo. No caso de um método de validação cruzada com k partições, repetimos k vezes as etapas 3, 4 e 5. (6) Por fim, obtemos a acurácia média do classificador.

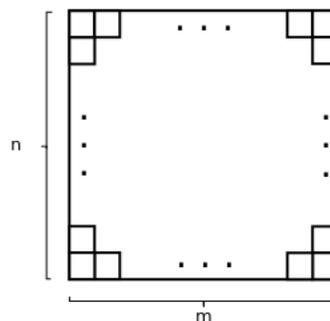


Figura 4.2: Representação da *matriz de características*. m representa o número de dimensões do descritor e n representa o tamanho do conjunto de imagens.

Em cada etapa da validação cruzada, quatro partições são utilizadas como treinamento e a partição restante para teste. Cada conjunto treinamento-teste serve de entrada a um classificador. Utilizamos SVM com núcleo radial.

Com o modelo criado, validamos os métodos com o conjunto disjunto restante (1.940 imagens).

A normalização dos dados é um passo fundamental ao se lidar com grande quantidade de dados heterogêneos [69]. A escolha de qual estratégia de normalização usar em métodos de fusão pode ser um desafio quando não se tem muita informação da distribuição dos dados. Aplicamos a normalização por *z-score* para todos os dados. Comparamos os resultados do último método de combinação com a normalização *z-score* e sem normalização alguma.

Há grande divergência entre as formas de análise dos resultados adotadas pelos trabalhos da área. Assim como no trabalho de Ng et al. [51], propomos a avaliação dos métodos a partir de curvas ROC [26] ao invés da simples avaliação com tabelas de contingência contendo as taxas de falsos positivos e falsos negativos.

A curva ROC apresenta informações visuais importantes ao se tentar determinar o comportamento do classificador em cenários de borda. Na área de identificação de IFGs e ICGs, podemos necessitar de métodos que sejam mais robustos em determinados cenários (extremos das curvas ROC). Por exemplo, no caso de uma evidência criminal, é mais grave utilizar ICG se passando por IFG que utilizar IFG como se fossem ICG, por constituir uma fraude. Por outro lado, no caso norte-americano de pornografia (Capítulo 1), é mais importante a identificação de IFG se passando por ICG.

Implementamos e validamos um total de 17 abordagens (veja Tabela 4.1). O restante desta seção explica como foram realizados: a aquisição das imagens (Seção 4.1), a implementação de métodos do estado da arte (Seção 4.2), a implementação dos métodos estendidos de áreas correlatas (Seção 4.3) e a implementação das formas de combinação de características (Seção 4.4).

4.1 Escolha das Imagens

O projeto contempla uma etapa de coleta de imagens. Para uma maior robustez das análises estatísticas, buscamos criar um espaço amostral grande e heterogêneo, o que se traduz por um grande número de imagens e diversidade de conteúdo.

Dentre as imagens fotográficas, buscamos tanto cenas de ambientes internos como de ambientes externos e de multiplicidade de equipamentos de origem. Dentre as geradas por computador, buscamos imagens fotorrealistas, apenas. Adicionalmente, utilizamos imagens de alto grau de realismo como as de desafios *online*¹ para testar nossas abordagens.

¹<http://www.fakeorfoto.com/>

Índice	Método	Base	Característica
1	LI	Derivadas de segunda ordem [39]	Bordas/Textura
2	LSB	Ruído da câmera [50]	Aquisição
3	LYU	Transformadas <i>wavelets</i> [44]	Bordas/Textura
4	POP	Preditor de interpolação [58]	Aquisição
5	BOX	Contagem de caixas [40]	Auto-similaridade
6	CON	<i>Contourlets</i> [19]	Bordas/Textura
7	CUR	<i>Curvelets</i> [8]	Bordas/Textura
8	GLC	Matriz de coocorrência [31]	Textura
9	HOG	Histograma de gradientes orientados [14]	Forma
10	HSC	Histograma de coeficientes <i>shearlets</i> [70]	Curvas
11	LBP	Padrões binários locais [52]	Bordas/Textura
12	SHE	Transformadas <i>shearlets</i> [36]	Bordas/Textura
13	SOB	Operador sobel [30]	Bordas
14	FUS1	Concatenação	Combinação
15	FUS2	Votação Simples	Combinação
16	FUS3	Votação Ponderada	Combinação
17	FUS4	Metaclassificação	Combinação

Tabela 4.1: Conceitos utilizados pelos métodos implementados. Na primeira coluna, exibimos os índices. Na segunda, o identificador utilizado em nosso trabalho. Na terceira, o principal conceito utilizado no método com a respectiva referência. Na última coluna, as características associadas a cada método.

4.2 Implementação dos Métodos do Estado da Arte

Implementamos dois métodos da literatura: LYU [44] e LI [39]. Ambos são descritos a seguir.

LYU

O primeiro trabalho implementado foi o de Lyu e Farid [44]. A implementação foi baseada no artigo, o qual apresenta uma descrição detalhada da implementação dos próprios autores.

Na primeira abordagem, utilizamos a decomposição piramidal com uma escala a mais (cinco escalas). Nossa hipótese era de que com as apenas quatro escalas do trabalho original, o método estaria ignorando características relevantes. Numa segunda abordagem, alteramos as estatísticas utilizadas em cada subbanda (estatísticas de ordem superior) para as características de Haralick et al. [31], porém, aplicadas sobre os coeficientes no domínio da frequência.

LI

O trabalho de Li et al. [39], assim como em [44], apresentou uma caracterização da implementação bastante detalhada, o que possibilitou implementar um algoritmo idêntico ao publicado.

Aplicamos sobre a imagem de entrada a conversão para o sistema de cores HSV e, em seguida, calculamos as aproximações presentes em [39] para estimar as diferenças de segunda ordem. Calculamos a curtose e a obliquidade da distribuição destes valores. Em seguida, reduzimos a escala da imagem em 50% e calculamos as diferenças de segunda ordem novamente. Calculamos uma predição linear dos coeficientes e seu erro associado, assim como no trabalho anterior de Lyu e Farid [44]. Novamente extraímos a curtose e a obliquidade dos coeficientes.

4.3 Implementação de Métodos de Áreas Relacionadas

Propomos e implementamos onze métodos a partir de áreas correlatas a de identificação de IFGs e ICGs: BOX, CUR, CON, SHE, HSC, GLCM, LBP, HOG, SOB, LSB e POP.

BOX

O método utilizando a fractalidade de uma imagem baseou-se na técnica de Contagem de Caixas [40]. O primeiro passo foi aplicar um detector de bordas limiarizado. Em seguida, realizamos a contagem do número de caixas ($N(\epsilon)$) necessário para cobrir todas as bordas da imagem, para diferentes tamanhos de caixa ϵ . A partir dos pares de valores, ϵ e $\frac{1}{N(\epsilon)}$, ajustamos uma reta e computamos seu coeficiente angular. Para cada imagem, obtivemos um vetor resultante de tamanho três. O coeficiente angular obtido foi utilizado como descritor da imagem como forma de estimar a fractalidade da imagem.

CUR

As transformadas *curvelets*, assim como as *wavelets*, têm sido empregadas em diversas áreas. Os próprios autores de [8] disponibilizaram uma implementação completa das transformadas *curvelets*, a *CurveLab*². Esta ferramenta serviu de base para um novo descritor. Utilizamos a transformada *curvelet* baseada em transformada Fourier heterogeneamente espaçada da versão 2.1.2 da ferramenta.

Inicialmente, utilizamos todos os coeficientes da transformada *curvelet* como entrada do classificador. A decomposição *curvelet*, pela sua própria característica de direciona-

²<http://www.curvelet.org> (último acesso em 23/07/2012)

lidade e localidade em espaço e frequência, apresenta um grande número de coeficientes (2.270.020 em nossa implementação). Devido à impossibilidade computacional em lidar com tantos dados (nem mesmo com PCA), para cada subbanda de escala, orientação e cor, computamos as quatro estatísticas de ordem superior. O número de características resultantes foi de 2.328 características por imagem.

CON

As transformadas *contourlets* [19] também têm uma implementação disponível na *Internet*, a *Contourlet Toolbox*³. Utilizamos tal implementação como base em nossa abordagem. Inicialmente, calculamos a decomposição Laplaciana padrão da imagem e, em seguida, aplicamos uma decomposição por banco de filtros passa-altas direcionais a cada escala. Podemos decompor cada escala em um número arbitrário de direções. Desta forma, dois níveis diferentes de quantização são utilizados para subbandas, mais níveis para escalas de maiores detalhamentos e menos níveis para as escalas restantes. A implementação tem como parâmetros o filtro da decomposição piramidal, o filtro direcional e o número de níveis de decomposição direcional em cada nível da pirâmide.

Utilizamos, em todos os casos, o filtro direcional *pkva* [56]. O filtro *pkva* é biortogonal e tem forma *quincunx* [19]. A implementação da transformada *contourlet* utilizada otimiza este filtro para obter respostas de frequência ideais.

A saída da implementação é uma matriz tridimensional, indexada por posição, escala e direção. Em seguida, para cada subbanda de escala, orientação e cor, extraímos as estatísticas de ordem para caracterização da distribuição dos coeficientes. Variamos o número de níveis de decomposição, o filtro piramidal e o filtro direcional de modo a identificar os parâmetros mais adequados.

SHE

Implementamos também um descritor baseado na transformada *shearlet* com auxílio da ferramenta *Shearlab*⁴, versão 1.1. Calculamos os coeficientes *shearlets* e, novamente, ao invés de utilizar toda a matriz de coeficientes, utilizamos quatro estatísticas de ordem superior para caracterizar a distribuição dos coeficientes.

Testamos o descritor com vários núcleos das transformadas *shearlets*:

- Daubechies8
- Daubechies16

³<http://www.ifp.illinois.edu/~minhdo/software/> (último acesso em 23/07/2012)

⁴http://www.shearlab.org/index_software.html

- Symmlet4
- Symmlet8

HSC

Utilizamos o trabalho de [70] para implementar um descritor baseado na transformada *shearlet*. O autor gentilmente cedeu o código fonte⁵, o qual utilizamos como base para extração de características.

Calculamos uma decomposição multiescala utilizando a transformada *shearlets* em 3 níveis e 8 direções. Em seguida, calculamos o histograma dos coeficientes em cada escala com 8 intervalos. Por fim, concatenamos e normalizamos estes histogramas.

Testamos o método com alteração no número de ângulos e no tamanho do bloco deslizando.

GLCM

O cômputo da matriz de coocorrência [31] é custoso, porém, simples. Utilizamos a GLCM para obter um descritor de textura. Estabelecemos direções e calculamos a quantidade de determinadas configurações dois a dois. A matriz de ocorrência resultante foi de 256×256 para cada canal de cor, pois utilizamos imagens de cores com sistemas de 8 *bits*. Numa primeira abordagem, aplicamos o descritor em cada canal de cor e extraímos as características não redundantes de Haralick et al. [31] (homogeneidade, energia, contraste e correlação). Obtivemos um total de 12 valores para cada imagem. Numa segunda abordagem, aplicamos a decomposição *wavelet* piramidal e, para cada subbanda de escala, orientação e cor, calculamos a matriz de coocorrência e calculamos as características de Haralick et al. Para cada imagem, obtivemos um total de 144 características.

LBP

Utilizamos os padrões binários locais como base para um outro descritor. Implementamos LBP invariante à rotação [53] para cada raio 1, 2 e 3 isoladamente.

O padrão binário local de um *pixel* é obtido pela limiarização dos valores dos *pixels* vizinhos pelo valor do *pixel* central. Estes valores limiarizados são ponderados pela máscara da Figura 3.2. Em seguida, calculamos os LBPs de toda a imagem e calculamos a frequência intervalar. A partir do histograma destes valores, obtemos o descritor.

⁵<http://homepages.dcc.ufmg.br/~william/software.html>

HOG

O Histograma de Gradientes Orientados [42] foi outro descritor utilizado. Para cada região de tamanho fixo da imagem (*célula*), calculamos a frequência intervalar de ocorrências de cada direção do gradiente e construímos um histograma de orientações de gradientes. As células utilizadas foram de tamanho 3×3 , retangulares. A direção de cada *pixel* da célula foi ponderada pela intensidade do gradiente. As células foram agrupadas em regiões maiores, os blocos. Por fim, os histogramas dos blocos são concatenados e normalizados.

Utilizamos blocos de tamanho 3×3 e histogramas divididos em 9 intervalos. Obtivemos um vetor de características resultante de tamanho 81.

SOB

Propusemos uma implementação para detecção de bordas. Primeiramente, aplicamos o operador Sobel [30] sem limiarização. Escolhemos, dentre todos os máximos locais no mapa de intensidade, os 50 maiores em magnitude. Buscamos, dentre as centenas de máximos locais encontrados para cada imagem, o conjunto de bordas de maior intensidade. Para cada um destes 50 pixels, fizemos um ajuste Gaussiano 2D centrado no ponto, num bloco quadrado de tamanho l (parâmetro que variamos). Excluimos as Gaussianas que tiveram ajustes ruins e calculamos as variâncias σ_x e σ_y de cada Gaussiana. Calculamos as variâncias médias ($\frac{\sigma_x + \sigma_y}{2}$) com o intuito de eliminar a variação à rotação. Obtivemos um total de $50 \times 3 = 150$ valores como conjunto de características.

LSB

Primeiramente, obtemos um mapa dos *bits* menos significativos da imagem, por meio da divisão do valor de cada *pixel* da imagem por 2. O conjunto de restos da divisão representam os *bits* menos significativos da imagem. Em seguida, calculamos as características de Haralick et al. [31] sobre estes coeficientes.

POP

O trabalho de Popescu e Farid [58] propõe uma forma de identificar o processo de demosaico. Ele foi utilizado como base para a implementação de um descritor do processo de interpolação de cores.

Seja $f(x, y)$ a intensidade do *pixel* na posição (x, y) em um determinado canal de cor da imagem. O trabalho parte da hipótese que $f(x, y)$ pertence a um destes dois modelos de correlação: M_1 , se ele está linearmente correlacionado aos seus vizinhos, ou M_2 , caso contrário. Desta forma, se $f(x, y)$ estiver linearmente correlacionado aos seus vizinhos,

sendo $\alpha = \{\alpha_{u,v} | -N \leq u, v \leq +N\}$ o conjunto de parâmetros de M_1 , então

$$f(x, y) = \sum_{u,v=-N}^N (\alpha_{u,v} f(x+u, y+v) + n(x, y))$$

$$r = \left| \frac{(f(x, y) - \sum_{u,v=-N}^N \alpha_{u,v} f(x+u, y+v))^2}{2\sigma^2} \right|$$

Então na etapa de Esperança, calculamos

$$Pr(f(x, y) \in M_1 | f(x, y)) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{r^2(x, y)}{2\sigma^2}\right)$$

Seja $\varepsilon(\alpha)$ o erro tal que, na etapa de Maximização, minimizamos seu erro quadrático $\frac{\partial \varepsilon}{\partial \alpha_{x,y}} = 0$ e obtemos

$$\sum_{u,v=-N}^{+N} \left(\sum_{x,y} w(x, y) f(x+s, y+t) f(x+u, y+v) \right) = \sum_{x,y} w(x, y) f(x+s, y+t) f(x, y)$$

Pode-se observar que esses passos são iterativos e, na primeira iteração, α é inicializado com o valor 0,5. A etapa de Esperança e de Maximização são repetidas até que a diferença entre os parâmetros α da iteração (i) e ($i-1$) seja menor que 1. A partir do mapa de probabilidades, extraímos as quatro estatísticas de ordem superior.

4.4 Implementação de Métodos de Combinação

Com o conhecimento das peculiaridades dos diferentes descritores explorados, analisamos a possibilidade de fusão de descritores e de classificadores.

FUS1

A concatenação de características foi nossa primeira abordagem para combinar os vetores de características. Trabalhos anteriores como [16] já exploraram esta idéia, contudo, nenhum deles combinou mais de três métodos distintos. Cada método de extração de características fornece, independentemente, um conjunto de características, os quais são reunidos num único vetor de características. O vetor resultante é o dado de entrada do classificador (veja Figura 4.3).

Uma possível vantagem da abordagem de concatenação é que não “perdemos” nenhum dos dados antes do julgamento final e todos os dados são fornecidos ao classificador. Uma outra característica é que apenas uma etapa de reconhecimento de padrões é realizada.

Como cada imagem é representada por um conjunto de características, a matriz resultante pode tornar-se grande (no nosso caso, 9.700×4.100). O classificador tem a tarefa de encontrar uma superfície de separação num espaço n -dimensional, em que n é grande. Neste caso, temos que lidar com a maldição da dimensionalidade. Além disso, a manipulação dinâmica deste volume de dados pode ser limitado pelo poder de processamento de um computador. Embora os avanços em *hardware* possibilitem realizar este processamento, outras abordagens demandam menor poder computacional e atingem resultados similares, como é o caso de FUS2, FUS3 e FUS4.

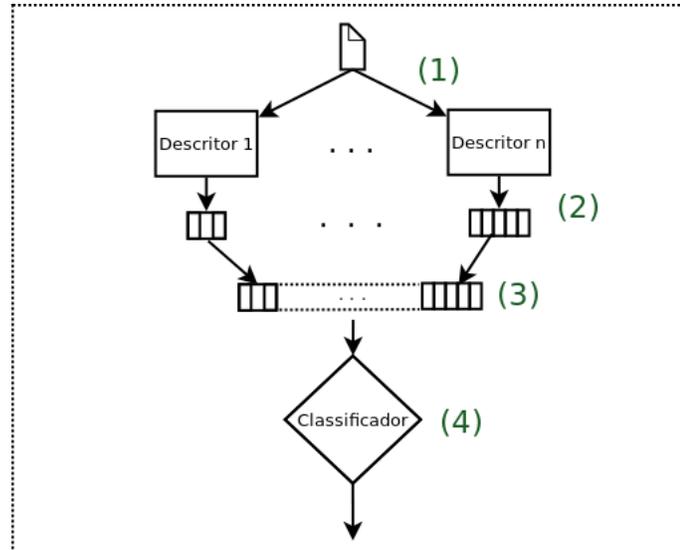


Figura 4.3: Fluxo do *Método da Concatenação*. (1) Uma imagem é descrita por n diferentes descritores. (2) Os n vetores de características são extraídos. (3) O vetor resultante é formado pela união de n vetores de características. (4) O Conjunto de dados resultante serve de entrada a um classificador final.

FUS2

O sistema de votação simples (sem ponderação) foi a implementação mais simples entre as formas de combinação. Uma vez implementados os métodos individualmente, realizamos a classificação de cada um dos métodos.

Uma imagem é classificada como IFG ou ICG por cada um dos n métodos e cada uma destas classificações é chamada de voto. A classe com maior número de votos é considerada a classe eleita do nosso classificador (veja Figura 4.4). No caso de empate, consideramos erro, isto é, o classificador não conseguiu identificar corretamente a classe. Sua implementação é simples e consiste numa contagem de votos de cada uma das classes. Uma desvantagem é a de que um dado método apenas elege uma classe, sem definir o

nível de confiança de sua resposta. Uma outra desvantagem é a de que métodos com baixa acurácia possuem o mesmo poder de voto que um método que produz ótimos resultados.

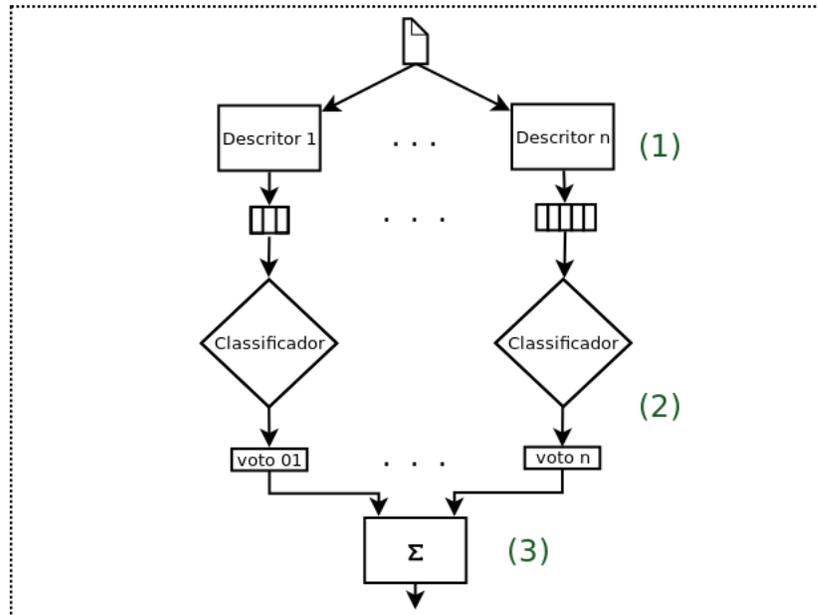


Figura 4.4: Fluxo do *Método de Votação Simples*. (1) n descritores são aplicados sobre a imagem. (2) Cada um dos classificadores elege uma classe (voto). (3) Soma dos valores das classes (apuração de votos).

Seja $vote_i(Im)$ o voto do classificador i sobre a imagem Im . Definimos

$$vote_i(Im) = \begin{cases} +1, & \text{se o voto for pela classe ICG} \\ -1, & \text{se o voto for pela classe IFG} \end{cases} \quad (4.1)$$

Definimos a regra 4.2 do nosso classificador como

$$H = \begin{cases} \text{CG}, & \text{se } \sum vote_i(Im) > 0 \\ \text{PG}, & \text{se } \sum vote_i(Im) < 0 \\ \text{Indefinido}, & \text{caso contrário} \end{cases} \quad (4.2)$$

FUS3

O sistema de votação ponderada utiliza como ponderação a acurácia obtida na fase de treinamento do classificador. Métodos com maior acurácia na fase de treinamento terão maior peso na apuração dos votos. Seja $acc(i)$ a acurácia média de um método i , e seja $A = 1, 2, \dots, n$ o conjunto dos índices dos métodos utilizados. Então, o peso w de um

método i é dado por

$$w(i) = \frac{acc(i)}{\sum_{j \in A} acc(j)} \quad (4.3)$$

Deve-se observar que, na fase de teste, já teremos $acc(i)$ de cada método e, portanto, $w(i)$ poderá ser computado.

Seja uma imagem Im a ser classificada e $vot_i(Im)$ o voto do método i . Utilizando o método de votação ponderada, realizamos a classificação por meio de cada método i e obtemos a votação ponderada $w_i \times vot_i(Im)$. O resultado final é obtido da seguinte forma

$$H = \begin{cases} \text{CG,} & \text{se } \sum_{i \in A} (w_i \times vot_i(Im)) < 0 \\ \text{PG,} & \text{se } \sum_{i \in A} (w_i \times vot_i(Im)) > 0 \\ \text{Indefinido,} & \text{caso contrário} \end{cases} \quad (4.4)$$

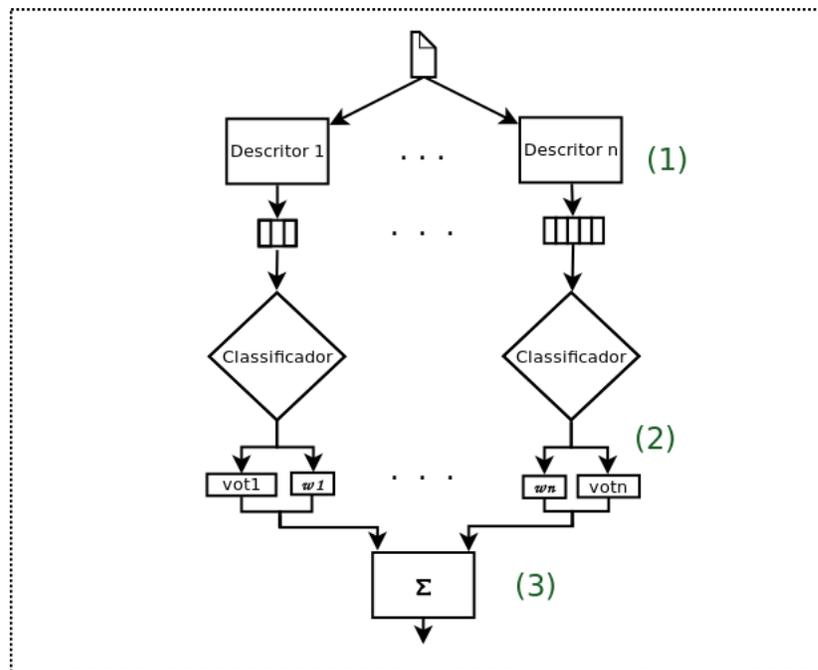


Figura 4.5: Fluxo do *Método de Votação Ponderada*. (1) n descritores são aplicados sobre a imagem. (2) Cada um dos n classificadores elege uma classe (voto) e cada classificador tem um peso w_i obtido na fase de treinamento. (3) Soma dos valores das classes, ponderada por w_i .

O fluxo do Método de Votação Ponderada, ilustrado na Figura 4.5, é muito similar ao do Método da Votação Simples. O que muda é a regra de decisão (passo (3)), a qual é

dada pela Equação 4.4.

FUS4

A motivação para criação de um outro método de votação é a de que cada método pode fornecer mais do que uma simples classificação binária (voto). Ao invés de utilizar um voto, utilizamos uma classificação com nível de confiança, o que chamamos de opinião.

No caso do SVM, por exemplo, ao invés de fornecer um valor discreto (-1 ou $+1$) associado a uma das classes, retorne um valor contínuo ($\{x|x \in \mathbb{R}\}$). O SVM estabelece um hiperplano de separação entre as classes e o valor retornado indica “a qual lado” do hiperplano, o dado se encontra. A magnitude, por sua vez, indica o quão “distante” do hiperplano está o dado. Esta informação é utilizada pelo classificador para definir a classe de saída. A opinião do classificador, portanto, tem importante semântica:

- seu sinal indica qual das duas classes foi escolhida pelo classificador;
- sua magnitude indica *quão certo* o classificador está de que sua classificação é confiável.

Por exemplo, uma opinião com valor $-1,34$ significa que a resposta do classificador é IFG e que seu “grau de confiança” é $1,34$.

Assim como no método anterior, realizamos um processo de extração de características e classificação para cada método. Ajustamos cada um dos classificadores para retornar seu grau de confiança (opinião). As opiniões são agrupadas num vetor de características de tamanho n , em que n é o número de métodos utilizados. Este é o vetor de características utilizado por um último classificador (veja Figura 4.6). Este último classificador é do tipo *meta*, pois utiliza dados das classificações para realizar a classificação final.

Assim como o método previamente apresentado, ele pondera os métodos de acordo com sua acurácia na fase de treinamento. Quem se encarrega da regra de decisão final é um classificador. Escolhemos SVM como nosso classificador final, embora qualquer abordagem de aprendizado supervisionado pudesse ter sido utilizada. Nossa escolha foi motivada pelo fato de SVM ser bem conhecido, ser mais robusto quanto à dimensionalidade do espaço de entrada, usar minimização de risco estrutural e permitir o teste rápido (sublinear do tempo) de uma entrada.

Com o método de Metaclassificação, esperamos um resultado melhor que o método de Votação Ponderada, pois a ponderação neste caso é feita utilizando um classificador.

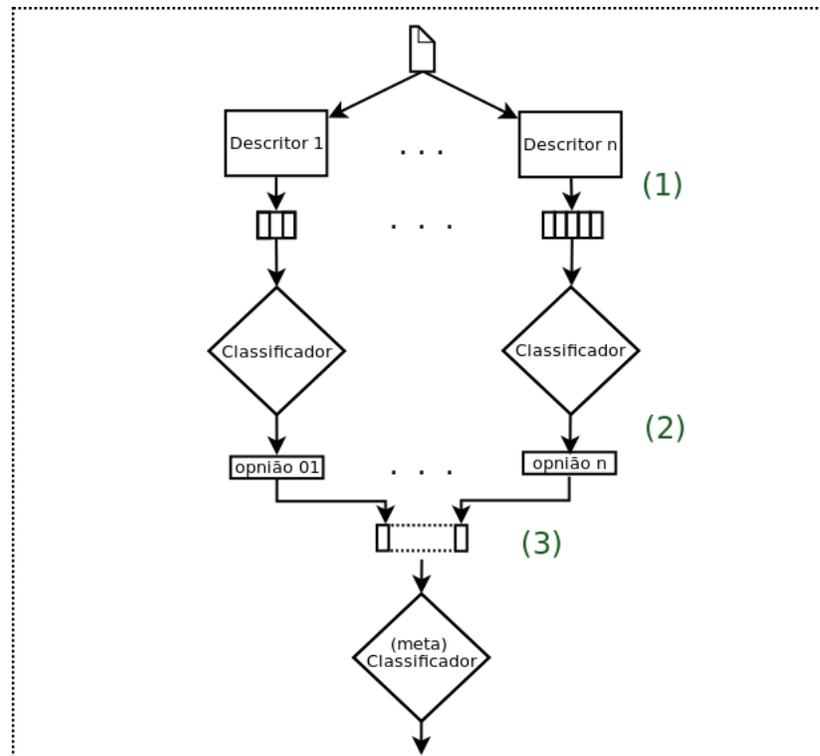


Figura 4.6: Fluxo do Método de Metaclassificação. (1) n extrações independentes de características. (2) n classificações e obtenção de n níveis de confiança (opiniões). Vetor resultante com tamanho n . (3) O vetor resultante é entrada de um último classificador.

Capítulo 5

Experimentos e Validação

A validação dos experimentos pode ser dividida em quatro seções: Configuração do Sistema, Coleta das Imagens, Validação dos Métodos do Estado da Arte, Validação dos Métodos de Áreas Relacionadas e Validação dos Métodos de Combinação. Na Seção Configuração do Sistema, detalhamos o ambiente de desenvolvimento e de testes. Na Seção de Coleta de Imagens, expomos como é o conjunto de dados a ser utilizado na validação. Em Métodos do Estado da Arte e em Métodos de Áreas Relacionadas, apresentamos os resultados dos métodos da literatura e dos métodos estendidos de áreas correlatas, respectivamente. Em Métodos de Combinação, utilizamos os métodos isolados com melhores resultados obtidos e validamos abordagens de combinação. Por fim, na Seção Análise Comparativa, fazemos uma análise comparativa das 17 abordagens implementadas.

5.1 Configuração do Sistema

A implementação dos descritores foi realizada em Matlab¹ devido à nossa familiaridade com a linguagem; devido ao grande número de bibliotecas existentes na área de reconhecimento de padrões e de visão computacional²³; devido à extensa documentação existente; e devido à popularidade do manuseio entre peritos criminais, aos quais a ferramenta poderá ser útil.

As classificações foram feitas em linguagem R⁴, devido à grande quantidade de trabalhos já desenvolvidos na área de reconhecimento de padrões. A biblioteca *e1071* é uma implementação da LIBSVM [10] e foi utilizada em conjunto com uma biblioteca de reconhecimento de padrões (*ipred*) da linguagem R.

¹<http://www.mathworks.com/> (último acesso em 23/07/2012)

²http://www.shearlab.org/index_software.html (último acesso em 23/07/2012)

³<http://www.curvelet.org/software.html> (último acesso em 23/07/2012)

⁴<http://www.r-project.org/> (último acesso em 23/07/2012)

Numa primeira abordagem, verificamos que os resultados foram drasticamente alterados (diferença de até 10 pontos percentuais) pela escolha de parâmetros. A escolha dos parâmetros influencia na acurácia da classificação. Por este motivo, utilizamos busca em malha (*grid search*) sobre todas as classificações do trabalho.

A máquina utilizada para o processamento foi um Pentium Dual Core 2,7 GHz com 2 GBytes de RAM.

5.2 Coleta das Imagens

Todas imagens, tanto ICG quanto IFG, foram extraídas da *Internet*. Uma etapa inicial do trabalho foi prospectar *websites* que oferecessem o conteúdo que buscávamos. Há abundância de IFG na *Internet*, sejam de *websites* pessoais ou de bancos de imagens públicos. As ICGs existem em menor quantidade, o que se justifica pela maior dificuldade de criação deste tipo de imagem. Para o nosso fim, a quantidade é ainda menor, pois buscamos ICG *fotorrealistas*. Avaliamos dezenas de fontes e elegemos apenas uma parcela para ser utilizada como nossa fonte de dados. Ícones gráficos foram excluídos. Das mais de 7.000 imagens geradas em computador coletadas, utilizamos 4.850. Para manter um mesmo número de imagens em ambas as classes, de um conjunto de 60.000 IFG, escolhemos aleatoriamente 4.850 para comporem nosso conjunto de dados.

Todas imagens coletadas têm formato de compressão JPEG e tamanho físico entre 12 KBytes e 1,8 MBytes.

Câmeras atuais possibilitam a captura de imagens de dimensões maiores que as obtidas por câmeras antigas. A nossa abordagem para manter consistente a análise dos dados foi a de padronizar o tamanho das imagens de entradas, por meio de um corte quadrado da imagem, de tamanho 512×512 *pixels*.

As imagens fotorrealistas geradas por computador utilizados têm fonte em: *3dcommunity*⁵, *3dshop*⁶, *3dtotal*⁷, *Artlantis*⁸, *Caligari*⁹, *CG Channel*¹⁰, *Creative Crash*¹¹, *Exchange*

⁵<http://www.3d-community.com/> (último acesso em 23/07/2012)

⁶<http://www.3dshop.com/> (último acesso em 23/07/2012)

⁷<http://www.3dtotal.com/> (último acesso em 23/07/2012)

⁸<http://www.artlantis.com/> (último acesso em 23/07/2012)

⁹<http://www.caligari.com/> (último acesso em 23/07/2012)

¹⁰<http://www.cgchannel.com/> (último acesso em 23/07/2012)

¹¹<http://www.creativecrash.com/> (último acesso em 23/07/2012)

*3D*¹², *IRTC*¹³, *Maxon*¹⁴, *Mental Images*¹⁵, *NPower*¹⁶, *Pandm Studio*¹⁷, *Spatarozliev*¹⁸, *Surreal Structures*¹⁹, *Xfrog*²⁰. As fotografias digitais foram extraídas de *FreePhoto*²¹ e de *Pics n' Trails*²².

Diversas outras fontes de ICGs disponíveis na rede, como *Accurender*²³, *Design Community*²⁴, *Virtual Museum*²⁵, *Digital Art*²⁶, *Pandromeda*²⁷, não foram utilizadas dado que suas imagens não se enquadram como fotorrealistas.

Uma categorização das imagens por autenticidade de *conteúdo* e *aquisição*, como foi feita por [51] seria mais apropriada, pois poderíamos avaliar casos como o de recaptura (fotografias de ICG). Contudo, para estabelecer um cenário de testes robusto, necessitaríamos ter um número significativo de imagens de cada classe de conteúdo e aquisição e, na prática, é difícil preencher todas as categorias. Os temas mais frequentes das imagens encontradas publicamente na *Internet* são personagens e arquitetura; categorias ICG de paisagens que são pouco frequentes. Desta forma, optamos por manter nossas 9.700 imagens, com a categorização tradicional IFG versus ICG.

5.3 Validação de Métodos do Estado da Arte

Validamos duas abordagens nesta seção, LYU [44] e LI [39].

LYU

A implementação do método de Lyu et al. [44] (implementação 1) foi idêntica à reportada no trabalho original: primeiramente, calculamos uma decomposição *wavelet* piramidal em 4 escalas com os filtros utilizados pelos autores; a partir da transformada, obtivemos as quatro estatísticas de ordem superior (média, variância, curtose e obliquidade); em seguida, calculamos um preditor de erros e extraímos novamente as quatro estatísticas

¹²<http://www.exchange3d.com/> (último acesso em 23/07/2012)

¹³<http://www.irtc.org/> (último acesso em 23/07/2012)

¹⁴<http://www.maxon.net/> (último acesso em 23/07/2012)

¹⁵<http://www.mentalimages.com/> (último acesso em 23/07/2012)

¹⁶<http://www.npowersoftware.com/> (último acesso em 23/07/2012)

¹⁷<http://www.pandm-studio.com/> (último acesso em 23/07/2012)

¹⁸<http://spatarozliev.co.uk> (último acesso em 23/07/2012)

¹⁹<http://surrealstructures.com/> (último acesso em 23/07/2012)

²⁰<http://xfrog.com> (último acesso em 23/07/2012)

²¹<http://www.pandm-studio.com/> (último acesso em 23/07/2012)

²²http://www.hal.t.u-tokyo.ac.jp/~chamds/pics_n_trails/ (último acesso em 23/07/2012)

²³<http://www.accurender.com/> (último acesso em 02/09/2011)

²⁴<http://gallery.designcommunity.com/> (último acesso em 02/09/2011)

²⁵<http://moca.virtual.museum/> (último acesso em 02/09/2011)

²⁶<http://digitalart.org/> (último acesso em 02/09/2011)

²⁷<http://www.pandromeda.com/> (último acesso em 02/09/2011)

sobre o “erro”. Obtivemos 72 características para cada canal de cor. O resultado atingiu um acerto médio de 92% (veja Tabela 5.1).

Realizamos alterações no método original para verificar uma possível melhoria apenas com alteração de parâmetros. Numa primeira variação (implementação 2), utilizamos uma decomposição piramidal em 5 escalas (ao invés de 4 escalas do método original). O resultado foi alterado em menos de um ponto percentual com relação ao resultado original.

Numa segunda alteração (implementação 3), ao invés de utilizar as quatro estatísticas de ordem superior para cada escala de decomposição piramidal, utilizamos as características da matriz de cocorrência como descritor final. Obtivemos, neste caso, um vetor de características de 144 dimensões. O resultado foi de aproximadamente 88% de acurácia.

Coeficientes Wavelets	Número de Características	Acurácia		Variância		Acurácia Média
		ICG	IFG	ICG	IFG	
Implementação 1	216	94,2%	89,9%	2,40E-04	2,12E-04	92,0%
Implementação 2	240	93,8%	90,1%	5,21E-04	3,24E-04	91,9%
Implementação 3	144	91,2%	85,3%	5,10E-04	2,64E-04	88,2%

Tabela 5.1: Resultados do método baseado em decomposição em coeficientes *wavelets*.

LI

O método de Li et al. [39] obteve a maior acurácia deste grupo. A classificação de ICG apresentou acurácia da ordem de 95% (veja Tabela 5.2).

Obtivemos um vetor de características de tamanho 48 para cada canal de cor da imagem, totalizando 144 características.

Diferenciação de segunda ordem	Número de Características	Acurácia		Variância		Acurácia Média
		ICG	IFG	ICG	IFG	
	144	94,8%	91,1%	1,07E-04	5,84E-05	93,0%

Tabela 5.2: Resultados do método baseado em diferenciação de segunda ordem.

5.4 Validação de Métodos de Áreas Relacionadas

Implementamos onze métodos: BOX, CON, CUR, GLCM, HOG, HSC, LBP, LSB, SHE, SOB e POP [58].

BOX

A técnica baseada em Contagem de Caixas não obteve boa acurácia (Tabela 5.3). O resultado não ficou longe do esperado, pois utilizamos apenas as dimensões fractais (uma dimensão para cada canal de cor). Deste modo, o vetor de características obtido teve tamanho 3.

Embora a técnica tenha baixa acurácia isoladamente, ela pode aumentar a acurácia se combinada a outros métodos.

Contagem de Caixas	Número de Características	Acurácia		Variância		Acurácia Média
		ICG	IFG	ICG	IFG	
Implementação 1	3	54,1%	56,8%	8,98E-04	2,00E-03	55,4%

Tabela 5.3: Resultados do método baseado em contagem de caixas.

CON

A implementação da transformada *contourlet* tem como parâmetros de programa o número de níveis na decomposição, o filtro da decomposição piramidal e o filtro direcional.

Numa primeira implementação, utilizamos filtros *pkva* e níveis de decomposição (0, 0, 0, 4, 5), ou seja, as três piores escalas sem decomposição (0), a penúltima com 4 níveis e a última com 5 níveis. Numa segunda implementação, alteramos apenas o nível de decomposição da terceira escala, em que utilizamos 4 níveis - (0, 0, 4, 4, 5). Utilizamos este conjunto de parâmetros porque eram os utilizados em exemplos dos próprios autores da *toolbox*.

Em ambos os casos, utilizamos o descritor da matriz de coocorrência para extração de características. Um total de 232 e 254 características para a primeira abordagem e segunda abordagem, respectivamente, foram obtidas para cada canal de cor (Tabela 5.4).

A primeira abordagem utilizou um número menor de características e foi a que obteve maior acurácia (~90%).

Contourlets	Níveis na terceira escala	Número de Características	Acurácia		Variância		Acurácia Média
			ICG	IFG	ICG	IFG	
Implementação 1	0	696	91,8%	88,7%	3,09E-04	2,97E-04	90,2%
Implementação 2	4	852	90,4%	88,5%	2,25E-04	6,89E-04	89,4%

Tabela 5.4: Resultados do método baseado em transformadas *contourlets*.

CUR

Embora não tenhamos utilizado os coeficientes *curvelets* diretamente como descritor, realizamos a extração integral dos coeficientes. Para imagens com 512×512 *pixels*, obtivemos como saída matrizes tridimensionais de 556.032 coeficientes. Lidar com grande quantidade de dados, particularmente em Matlab²⁸, exige dedicação exclusiva do computador, no caso de um máquina sem grandes recursos. O vetor de características resultante, após extração das características de Haralick et al. [31] atinge o tamanho 2.328.

A classificação utilizando a implementação de *curvelets* obteve uma acurácia média de aproximadamente 80% (veja Tabela 5.5). A maior desvantagem da abordagem é a manipulação de um grande número de dados, o que é um gargalo de desempenho do método.

Curvelets	Número de Características	Acurácia		Variância		Acurácia Média
		ICG	IFG	ICG	IFG	
Implementação 1	2.328	80,6%	80,5%	3,57E-04	1,52E-03	80,5%

Tabela 5.5: Resultados do método baseado em *curvelets*.

GLCM

Uma abordagem para utilização das características de Haralick et al. [31] foi a aplicação do método sobre a imagem inteira (em outras implementações, utilizamos a matriz de co-ocorrência sobre porções da imagem ou até sobre dados no domínio de frequência). Inicialmente, computamos a matriz de coocorrência e, em seguida, extraímos as características de Haralick et al. para cada canal de cor. Utilizamos um vetor de 12 dimensões para representar cada imagem. Comparando as dimensões do método com a saída da transformada *contourlet* (696), as quais são subótimas [19], já era esperado que o resultado fosse inferior ao de outros métodos, particularmente ao de *contourlet* (veja Tabela 5.6).

Matriz de Coocorrência	Número de Características	Acurácia		Variância		Acurácia Média
		ICG	IFG	ICG	IFG	
Implementação 1	12	64,0%	63,0%	1,14E-03	8,88E-04	63,5%

Tabela 5.6: Resultados do método baseado em matriz de coocorrência.

²⁸<http://www.mathworks.com/>

HOG

Na implementação baseada nos histogramas de gradientes orientados, utilizamos 9 e 16 orientações. A partir da Tabela 5.7, verificamos uma acurácia média de 74% do método na melhor implementação.

HOG	Número de orientações	Número de Características	Acurácia		Variância		Acurácia Média
			ICG	IFG	ICG	IFG	
Implementação 1	9	81	73,5%	70,0%	1,02E-03	5,33E-04	71,8%
Implementação 2	16	256	75,4%	72,5%	1,64E-04	8,77E-04	74,0%

Tabela 5.7: Resultados do método baseado em Histograma de gradientes orientados.

HSC

Na abordagem baseada em HSC [70], há uma série de parâmetros. É possível alterar o tamanho do bloco, o deslocamento do bloco deslizante, o número de níveis e o número de ângulos. Testamos três conjuntos de parâmetros baseados em nosso conjunto de dados. Em todas as abordagens, utilizamos deslocamentos de 256 *pixels* e 8 níveis. Numa primeira abordagem, utilizamos blocos de tamanho 256 × 256 *pixels* e 8 ângulos; numa segunda abordagem, mantivemos o tamanho 256 × 256 *pixels* e alteramos o número de ângulos para 16; finalmente, numa última abordagem, utilizamos 8 ângulos, porém utilizamos blocos de tamanho 512 × 512 *pixels*. Na primeira e na segunda abordagem, obtivemos quatro blocos justapostos, enquanto na última obtivemos apenas 1 bloco. O método foi aplicado independentemente para cada canal de cor.

Verificamos pela Tabela 5.8 que a variação dos parâmetros praticamente não alterou a acurácia do método. Obtivemos um acurácia média de 80%.

Histograma shearlets	Parâmetros		Número de Características	Acurácia		Variância		Acurácia Média
	N	Níveis		ICG	IFG	ICG	IFG	
Implementação 1	256	8	96	81,8%	78,7%	2,24E-04	4,17E-04	80,2%
Implementação 2	256	16	192	83,0%	78,5%	5,09E-04	7,37E-04	80,8%
Implementação 3	512	8	24	81,5%	78,3%	7,97E-04	2,89E-04	79,9%

Tabela 5.8: Resultados do método baseado em histograma de coeficientes *shearlets*. Em cada um dos métodos, decompomos as imagens em três níveis. Os coeficientes foram obtidos para cada bloco, com centros do bloco deslocados em 256 *pixels*.

LBP

No método baseado em Padrões Binários Locais, o principal parâmetro é a definição do raio de vizinhança. Definimos três raios de vizinhança para nosso método, 1, 2 e 3, o que resultou em 8, 16 e 24 vizinhos, respectivamente. A melhor acurácia (~87%) foi obtida com o raio 2 (veja Tabela 5.9).

Padrões binários Locais	Raio	Número de Características	Acurácia		Variância		Acurácia Média
			ICG	IFG	ICG	IFG	
Implementação 1	1	30	88,5%	82,4%	1,01E-04	8,72E-04	85,5%
Implementação 2	2	54	90,4%	83,8%	5,64E-04	1,71E-04	87,1%
Implementação 3	3	78	89,4%	83,1%	8,94E-04	4,30E-04	86,3%

Tabela 5.9: Resultados do método baseado em Padrões Binários Locais.

LSB

Calculamos o canal de bits menos significativos para cada canal de cor da imagem, assim como em [63]. A partir de cada mapa de bits menos significativos, extraímos a matriz de coocorrência e suas características [31]. A extração de características é rápida e tem como saída 4 características para cada mapa de bits menos significativos. A acurácia média foi de 66%, como verificado pela Tabela 5.10.

Bits menos Significativos	Número de Características	Acurácia		Variância		Acurácia Média
		ICG	IFG	ICG	IFG	
Implementação 1	12	67,2%	65,1%	5,79E-04	9,26E-04	66,2%

Tabela 5.10: Resultados do método baseado em bits menos significativos.

SHE

Implementamos um método baseado no trabalho de [45], utilizando diferentes filtros *wavelets* (como consta na Tabela 5.11). Utilizamos filtros Daubechies e Symmlets, com diferentes tamanhos de suporte.

Após a extração dos coeficientes *shearlets*, para cada nível de cor, escala e direção, utilizamos média, variância, curtose e obliquidade dos coeficientes como descritores.

A acurácia obtida por cada uma das implementações variou pouco e ficou em torno de 71%.

Shearlets	Tamanho do suporte	Número de Características	Acurácia		Variância		Acurácia Média
			ICG	IFG	ICG	IFG	
Implementação 1	8	656	74,8%	67,7%	6,75E-04	1,73E-04	71,3%
Implementação 2	16	656	74,8%	67,6%	1,57E-03	5,49E-04	71,2%
Implementação 3	4	656	75,2%	67,7%	1,24E-03	3,28E-04	71,5%
Implementação 4	8	656	74,7%	67,4%	7,05E-04	1,16E-03	71,0%

Tabela 5.11: Resultados do método baseado em coeficientes *shearlets*. As duas primeiras linhas foram feitas utilizando filtros *Daubechies* e a terceira e quarta foram feitas utilizando filtros *Symmlets*.

SOB

O operador de Sobel é um eficiente detector de bordas. Em nossa implementação, utilizamos duas variações: Gaussiana em janela de lado 7 e Gaussiana em janela de lado 9. Os resultados diferiram pouco e ficaram em torno de 55% (vide Tabela 5.12).

Bordas	Tamanho da janela	Número de Características	Acurácia		Variância		Acurácia Média
			ICG	IFG	ICG	IFG	
Implementação 1	7	150	55,4%	55,2%	8,20E-04	1,20E-03	55,3%
Implementação 2	9	150	56,4%	56,1%	1,35E-03	7,50E-04	56,2%

Tabela 5.12: Resultados do método baseado em detecção de bordas.

POP

Para tentar identificar traços do processo de aquisição de uma IFG, utilizamos a técnica de [58] como base. Como o método tem como saída um grande conjunto de dados, utilizamos as 4 estatísticas de ordem superior para extrair informações. Nossa implementação tem como saída um descritor de tamanho 4 para cada canal de cor. O método atingiu uma acurácia média de 57% (veja Tabela 5.13).

Processo de Interpolação	Número de Características	Acurácia		Variância		Acurácia Média
		ICG	IFG	ICG	IFG	
	12	57,0%	57,5%	3,15E-04	8,74E-04	57,3%

Tabela 5.13: Resultados do método baseado em processo de interpolação.

ENT

Um outro descritor seria um método que utilizasse entropia. Numa etapa de pré-extração de características, prospectamos a eficácia da entropia como um descritor. Calculamos a

entropia de todas as imagens e calculamos a entropia média e a variância dos valores para cada classe (IFG e ICG) e cada canal de cor (Tabela 5.14).

Canal de cor	Entropia		Variância	
	ICG	IFG	ICG	IFG
Vermelho	7,1332	7,2690	0,4734	0,3041
Verde	6,9514	7,3222	0,6908	0,2711
Azul	7,3158	7,3678	0,2162	0,2622

Tabela 5.14: Resultados da tentativa de implementação de um método em Entropia.

Vemos que, no canal verde, a entropia consegue apresentar uma boa distância entre as classes, mas a variância alta impossibilita estabelecer uma superfície de separação. Concluímos que a entropia, isoladamente, *não* consegue distinguir as classes. Por este motivo, não incluímos algum método de entropia entre os nossos métodos. Temos consciência, contudo, da sua eventual eficácia ao ser utilizada em conjunto com outro descritor, mas deixamos isso como um trabalho futuro.

5.5 Validação dos Métodos de Combinação

Uma vez implementadas diversas abordagens individuais para resolução do nosso problema, implementamos diferentes métodos de fusão. Nossa hipótese é a de que resultados individuais podem ser melhorados com a coalizão de diferentes e complementares descritores. Implementamos quatro formas de fusão de descritores:

- *Concatenação* (FUS1): união dos vetores de características;
- *Votação Simples* (FUS2): cada método classifica independentemente e a classificação final é feita através da contagem de votos;
- *Votação Ponderada* (FUS3): cada método realiza uma classificação isolada e a decisão final é feita de acordo com uma regra de decisão;
- *Metaclassificação* (FUS4): além de cada classificação, uma última classificação (nível meta) é avaliada como regra de decisão final.

FUS1

O método de Concatenação atingiu uma precisão de $\sim 93\%$ (veja Tabela 5.15), muito próxima a do melhor método individual (LI). A maior desvantagem da utilização do

Método de combinação	Probabilidade	Número de características	Acurácia		Variância		Acurácia média
			ICG	IFG	ICG	IFG	
FUS1	não	4.011	93,4%	92,2%	1,75E-01	0,17E-01	92,8%
FUS2	não	13	93,4%	96,0%	2,22E-04	3,43E-04	94,7%
FUS3	sim	13	94,9%	96,5%	2,29E-04	5,42E-04	95,7%
FUS4	sim	13	96,6%	98,0%	7,70E-04	4,42E-04	97,3%

Tabela 5.15: Comparação entre os resultados dos métodos de combinação. Cada uma das abordagens de combinação utiliza os 13 descritores previamente implementados.

método de Concatenação é a dificuldade encontrada pelo classificador em lidar com dados de alta dimensionalidade. Tentamos aplicar PCA sobre o conjunto de dados para evitar o consumo de memória na etapa de classificação mas não conseguimos realizar a redução devido à incapacidade física de nossa máquina em realizar os cálculos matriciais do método.

FUS2

Diferentemente da Concatenação, no método de Votação Simples cada método realiza um processo de classificação independente e, no final, há apenas uma contagem de votos. Num cenário em que há um poder de processamento paralelo, os dois pontos percentuais acima do método de combinação justificam a utilização desta abordagem em relação à anterior.

FUS3

O método de Votação Ponderada apresentou um aumento de dois pontos percentuais sobre o resultado da Votação Simples. Sua preferência sobre a Votação Simples só não se justifica se as classificações já tivessem sido feitas e não tivéssemos acesso à ponderação de cada método.

FUS4

O método de Metaclassificação apresentou o melhor resultado das abordagens de combinação, uma precisão média de 97%, quatro pontos percentuais acima do melhor método individual. Em termos de complexidade de implementação, é muito próximo do método de Votação Simples e de Votação Ponderada, sendo que a diferença é apenas uma meta-classificação adicional num espaço de dimensão 13-D.

5.6 Análise Comparativa

Normalização é um passo fundamental ao se lidar com dados de alta dispersão. Aplicamos a normalização por z -score para todos os dados. Comparamos os resultados do metaclasificador com a normalização z -score e sem normalização alguma e verificamos que não há diferença significativa entre a normalização (96,8%). Acreditamos que a razão desta inocuidade da normalização (diferença de 0,5%) deva-se ao conjunto de dados utilizado, o qual sempre esteve num intervalo bem definido de valores (no caso do metaclasificador, os valores estão num intervalo menor que $(-10, +10)$).

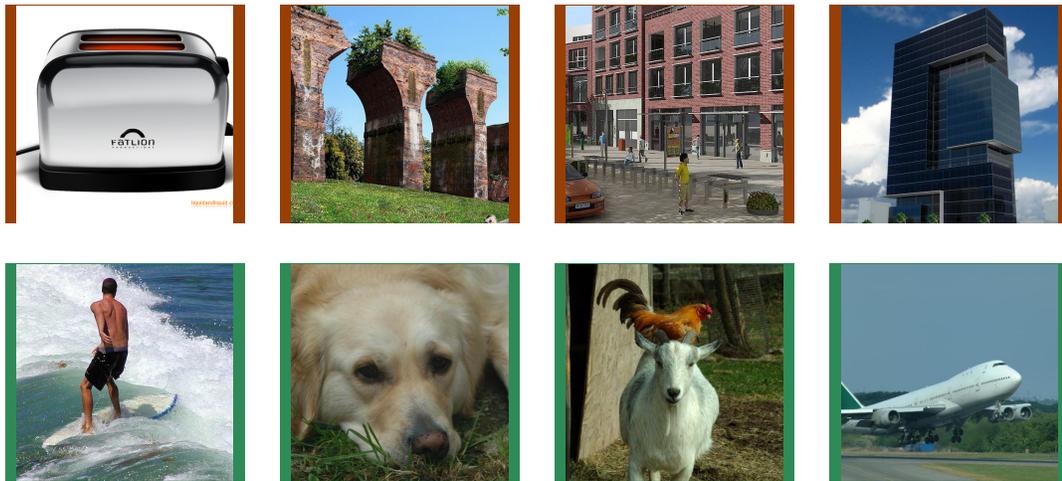


Figura 5.1: Amostras de imagens *corretamente* identificadas por, pelo menos, 9 dos 13 métodos individuais. Na primeira linha ICG e, na segunda, IFG.

Não houve nenhuma imagem que tenha sido classificado incorretamente por todos os métodos isolados e que tenha sido corretamente classificada por algum método de combinação. Os métodos de combinação não levam vantagem neste caso. Isto se justifica pelo fato de os métodos de combinação utilizarem unicamente os resultados dos métodos individuais utilizados na combinação como suas regras de decisão e estarem de acordo com o consenso dos métodos. Os métodos de combinação tampouco almejam alterar os resultados de imagens que foram corretamente classificadas por cada método individualmente.

Abordagens de combinação buscam identificar quais métodos são mais eficazes em determinados cenários. Assim, suponha os dados x , y e z , classificados pelos métodos A , B e C . Suponha que x foi corretamente classificada apenas pelo método A , assim como y por B e z por C (veja Figura 5.2). Métodos de combinação buscam utilizar as informações conjuntas de A , B e C para classificar corretamente tanto x , quanto y , quanto z . No caso de uma imagem que todos classificadores isolados a classificassem

incorretamente, provavelmente o método combinado também erraria.

Na Figura 5.1 verificamos imagens classificadas corretamente pela maior parte dos métodos individuais e corretamente classificada pelo método de Metaclassificação. Na Figura 5.3 verificamos imagens classificadas incorretamente por pelo menos 9 dos 13 métodos individuais e corretamente classificada pelo método de Metaclassificação.

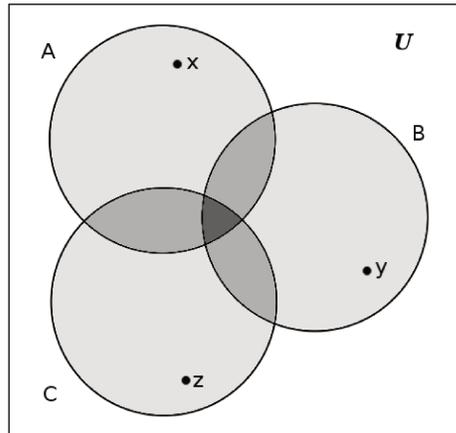


Figura 5.2: Diagrama de Venn dos acertos das imagens de um conjunto U por três métodos: A , B e C . A região mais clara representa o acerto por só um método, e as regiões escuras representam o acerto por mais de um método. Há dados que não foram corretamente classificados por nenhum dos métodos e estão na região externa ($U - (A \cup B \cup C)$).

A Tabela 5.16 mostra todos métodos implementados. Inserimos na tabela o melhor resultado dentre as implementações de cada método. Cada uma das linhas representa um algoritmo e cada coluna representa uma descrição do método. Pela tabela, verificamos que as acurácias têm um intervalo grande de valores. A maior acurácia entre os métodos individuais foi de 93,0%, obtida pelo método LI e os piores resultados foram aproximadamente 55%, obtidos pelos métodos BOX e SOB. A maior acurácia entre os métodos de combinação foi de aproximadamente 97%. Quanto à dimensão do espaço de características, também há grande diferença entre os métodos: a menor foi de $n = 3$, de BOX e a maior foi o de CUR, que utilizou 2.328 características.

As curvas ROC foram construídas da seguinte forma: definimos uma malha de valores entre o menor e o maior valores marginais. Cada ponto da malha foi utilizado como valor de corte. Para cada ponto desta malha, portanto, consideramos todos valores acima dele como a classe positiva e valores abaixo como a classe negativa.

A Figura 5.4 apresenta as curvas ROC de cada um dos métodos individuais implementados. Há grande divergência das curvas. Nos casos marginais, próximos aos eixos, LI,



Figura 5.3: Amostra de imagens *erroneamente* identificadas pela maior parte dos métodos individuais (pelo menos 9 dos 13 erraram) e corretamente classificada pelo método de Metaclassificação. Na primeira linha, ICG e na segunda, IFG.

Índice	Método	n	ICG	IFG	Acurácia média
1	BOX	3	54,1%	56,8%	55,4%
2	CON	696	91,8%	88,7%	90,2%
3	CUR	2,328	80,6%	80,5%	80,5%
4	GLC	12	64,0%	63,0%	63,5%
5	HOG	256	75,4%	72,0%	74,0%
6	HSC	96	81,8%	78,7%	80,2%
7	LBP	78	90,4%	83,8%	87,1%
8	LI	144	94,8%	91,1%	93,0%
9	LSB	12	67,2%	65,1%	66,2%
10	LYU	216	94,2%	89,9%	92,0%
11	POP	12	57,0%	57,5%	57,3%
12	SHE	60	74,8%	67,7%	71,3%
13	SOB	150	55,4%	55,2%	55,3%
14	FUS1	4.011	93,4%	92,2%	92,8%
15	FUS2	13	93,4%	96,0%	94,7%
16	FUS3	13	94,9%	96,5%	95,7%
17	FUS4	13	96,6%	98,0%	97,3%

Tabela 5.16: Comparação entre as abordagens implementadas para resolução do problema de identificação de IFG e ICG. Na primeira e na segunda colunas, apresentamos o índice e o método utilizado, respectivamente. Na terceira coluna, o número de dimensões do espaço de características. Na quarta e na quinta colunas, as acurácias em cada uma das classes. Na última coluna, a acurácia média.

LYU e CON apresentam os melhores resultados. As curvas, em geral, estão mais próximas ao eixo vertical, o que significa que os classificadores conseguem diminuir muito a taxa de

falsos positivos sem, necessariamente, diminuir muito a taxa de verdadeiros positivos.

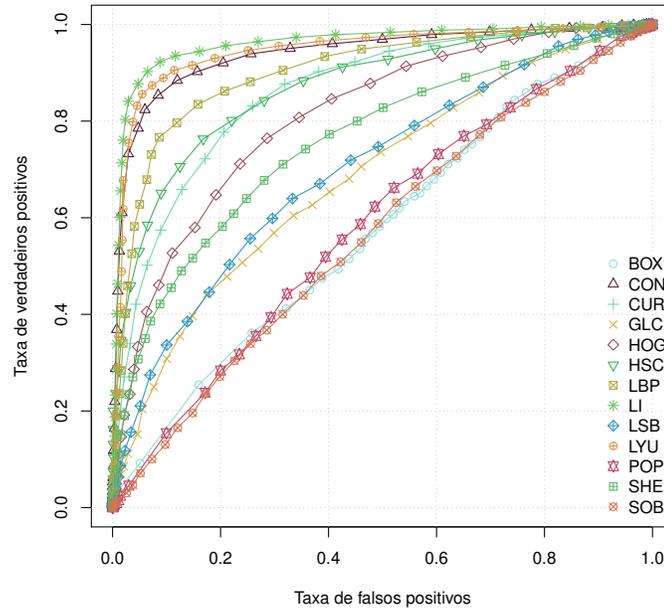


Figura 5.4: Curvas ROC dos métodos isolados implementados.

A Figura 5.5 apresenta a curva ROC do melhor resultado dentre todos implementados em nosso trabalho. A área sob a curva (AUC) é de 0,98.

Fake or Foto

O *website Fake or Foto*²⁹ apresenta um desafio que consta, entre 12 imagens, em identificar quais delas são IFG e ICG. Aplicamos cada um dos métodos isolados sobre as imagens da Tabela 5.6. O método com o maior número de acertos foi o de LI, ou seja, 8 acertos em 12. Os métodos de Concatenação, Votação Simples, Votação Ponderada e Metaclassificação atingiram 9, 8, 9 e 10, respectivamente. Os resultados mostram que, numa aplicação real, métodos de combinação de descritores podem melhorar a acurácia da classificação final.

²⁹<http://area.autodesk.com/fakeorfoto> (último acesso em 23/07/2012)

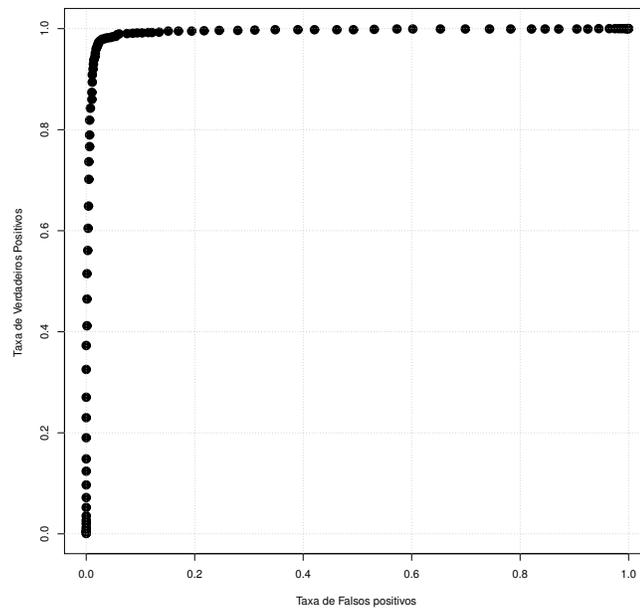


Figura 5.5: Curva ROC do Metaclassificador. $AUC=0,98$.

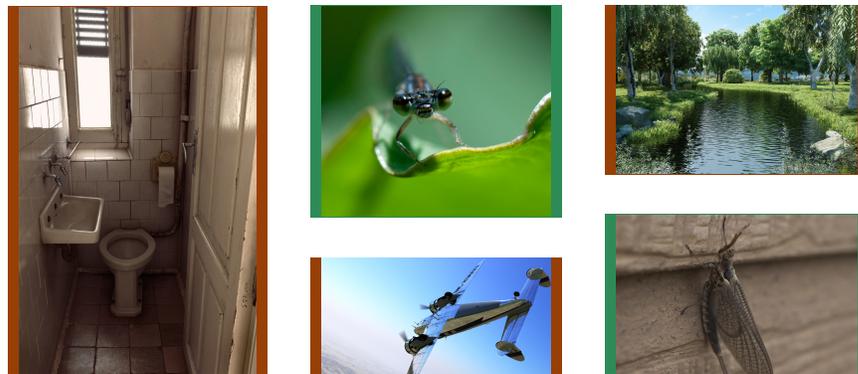


Figura 5.6: Conjunto de imagens do website *Fake of Foto*. As imagens com bordas verdes são IFG. As restantes são ICG.

Capítulo 6

Conclusões e Trabalhos Futuros

A tecnologia modificou significativamente a forma como capturamos, criamos e armazenamos imagens. Câmeras atuais permitem imagens rápidas (instantâneas), com o suporte de ferramentas de aquisição e de tratamento de imagens inseridas no próprio dispositivo. Alterar uma fotografia e distribuí-la também se tornou uma tarefa simples, graças aos poderosos editores de imagens e à Internet.

Nesse contexto, a imagem se consolidou como um meio de comunicação acessível popularmente. Paralelamente, os avanços em Computação Gráfica permitiram a criação de imagens de níveis de complexidade, antes nunca atingidos. Técnicas modernas e máquinas mais potentes possibilitam a criação de cenas complexas, visualmente muito próximas do real. Imagens deste cunho podem confundir um usuário leigo. A conjugação do fato de as IFGs terem seu papel como documento digital, acessível e confiável e do fato de ICGs terem um nível de complexidade a ponto de confundir o visualizador pode gerar problemas no âmbito criminal. A comprovação da autenticidade de uma imagem por um júri pode validar uma evidência criminal a qual, por sua vez, pode incriminar/absolver um indivíduo.

A maior parte dos trabalhos existentes que buscam distinguir IFGs e ICGs não realizam uma comparação completa e robusta com métodos anteriores. Os trabalhos geralmente fazem comparações com outros métodos a partir das acurácias reportadas. Utilizar valores reportados por outros métodos leva a comparações inconsistentes. Os ambientes de testes podem divergir como, por exemplo, quanto ao conjunto de dados de treinamento-teste. Os métodos assumem várias hipóteses a tal ponto que, num cenário real, esperamos que os resultados sejam significativamente piores [29]. Num contexto em que os objetos de estudo são potenciais evidências criminais, precisamos de métodos robustos em todo tipo de cenário.

Buscamos neste trabalho: criar um cenário de testes comum; implementar métodos do estado da arte; propor e implementar novos métodos; realizar uma comparação consistente

dos métodos existentes; e discutir suas complementaridades quando utilizados em conjunto em técnicas de fusão de dados. Todos os nossos testes utilizaram o mesmo conjunto de dados, treinamento/teste, bem com as mesmas condições de execução. Os métodos implementados variaram quanto ao número de características utilizadas (de 3 a 2.328) e desempenho computacional. Dentre os métodos avaliados, os métodos dos trabalhos [39] e [44] apresentaram os melhores resultados individualmente, aproximadamente 93% de acurácia.

Há uma série de abordagens para tratar o problema IFG versus ICG. Até o momento, contudo, houve pouco esforço para combinar as técnicas já existentes. Neste trabalho, discutimos quatro métodos de combinação de descritores. O método da Concatenação realiza uma fusão simples das características de cada método isolado. O método da Votação Simples realiza a extração e classificação individualmente por cada método e a classe mais escolhida é considerada a classificação final. O método da Votação Ponderada utiliza informações da fase de treinamento para ponderar a escolha final. O método da Metaclassificação utiliza as distâncias às margens do SVM como características a serem reunidas num vetor de características resultante. O melhor resultado foi o do Metaclassificador, com (97% de acurácia), com mais de quatro pontos percentuais sobre o resultado do melhor método isolado.

Para um cenário real de classificação, se o usuário já tem seus descritores, então a implementação de métodos de combinação é simples. A utilização conjunta de descritores é geralmente feita por concatenação, porém, nossos resultados indicam que uma Votação Simples dos métodos pode obter resultados melhores.

Trabalhos futuros incluem a extensão dos métodos a regiões locais das imagens (por blocos), aplicação de técnicas de normalização mais sofisticadas como *w-score* [69] e a inclusão de outros, entre tantos, descritores disponíveis. Uma outra abordagem seria, ao invés de estudar recortes de imagens, estudar imagens redimensionadas, já que esta operação é muito frequente em imagens na Internet. Técnicas contraforenses também são alvos de estudo, pois podemos utilizar descritores que sejam mais robustos a determinados tipos de ataques. O caso de imagens obtidas por recaptura, como abordado no trabalho [51], representa um problema de relevância que poderia ser explorado num trabalho futuro e, neste caso, a separação dos conceitos *conteúdo* e *cena* [51] poderia ser considerada.

Bibliografia

- [1] S. G. art, E. Bienenstock, and R. Doursat. Neural Networks and the Bias/Variance Dilemma. *Neural Computation*, 4(1):1–58, 1992.
- [2] V. Athitsos, M. J. Swain, and C. F. Athitsos. Distinguishing Photographs and Graphics on the World Wide Web. In *Content-Based Access of Image and Video Libraries (CBAIVL)*, pages 10–17, USA, 1997.
- [3] B. E. Bayer. Color Imaging Array. U.S. Patent 3971065, 1976.
- [4] L. Breiman and L. Breiman. Bagging predictors. In *Machine Learning Journal*, pages 123–140, 1996.
- [5] G. Brown, J. Wyatt, R. Harris, and X. Yao. Diversity creation methods: A survey and categorisation. *Journal of Information Fusion*, 6:5–20, 2005.
- [6] R. W. Buccigrossi and E. Simoncelli. Image Compression via Joint Statistical Characterization in the Wavelet Domain. *IEEE Transactions on Image Processing (TIP)*, 8(12):1688–1701, 1999.
- [7] E. Candes, L. Demanet, D. Donoho, and L. Ying. Fast Discrete Curvelet Transforms. *Multiscale Modeling Simulation*, 5(3):861–899, 2006.
- [8] E. J. Candes and D. L. Donoho. *Curvelets –A Surprisingly Effective Nonadaptive Representation for Objects with Edges*. Vanderbilt University Press, 2000.
- [9] R. Chakravarti and X. Meng. A Study of Color Histogram Based Image Retrieval. In *IEEE Intl. Conference on Information Technology (CIT)*, pages 1323–1328, USA, 2009.
- [10] C. Chang and C. Lin. LIBSVM: A Library for Support Vector Machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2:1–27, 2011.

- [11] W. Chen, Y. Shi, and G. Xuan. Identifying Computer Graphics Using HSV Color Model and Statistical Moments of Characteristic Functions. In *IEEE Intl. Conference on Multimedia and Expo (ICME)*, pages 1123–1126, China, 2007.
- [12] D. A. Clausi. An Analysis of Co-occurrence Texture Statistics as a Function of Grey Level Quantization. *Canadian Journal of Remote Sensing (CJRS)*, 28(1):45–62, 2002.
- [13] F. Cutzu, R. Hammoud, and A. Leykin. Estimating the Photorealism of Images: Distinguishing Paintings from Photographs. *IEEE Intl. Conference on Computer Vision and Pattern Recognition (CVPR)*, 2:305, 2003.
- [14] N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. In *IEEE Intl. Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 886–893, USA, 2005.
- [15] S. Dehnie, T. Sencar, and N. Memon. Identification of Computer Generated and Digital Camera Images for Digital Image Forensics. In *IEEE Intl. Conference on Image Processing (ICIP)*, USA, 2006.
- [16] A. Dirik, S. Bayram, H. Sencar, and N. Memon. New Features to Identify Computer Generated Images. In *IEEE Intl. Conference on Image Processing (ICIP)*, volume 4, pages 433–436, USA, 2007.
- [17] E. Dirik, H. Sencar, and N. Memon. Source Camera Identification Based on Sensor Dust Characteristics. In *IEEE Signal Processing Applications for Public Security and Forensics (SAFE)*, pages 1–6, USA, 2007.
- [18] M. Do and M. Vetterli. The Finite Ridgelet Transform For Image Representation. *IEEE Transactions on Image Processing (TIP)*, 12(1):16–28, 2003.
- [19] M. N. Do and M. Vetterli. Contourlets: A Directional Multiresolution Image Representation. In *IEEE Intl. Conference on Image Processing (ICIP)*, volume 1, pages I-357–360, USA, 2002.
- [20] R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis*. John Wiley & Sons Inc, 2 edition, 1973.
- [21] G. Elkharraz, S. Thumfart, D. Akay, C. Eitzinger, and B. Henson. Texture Features Corresponding to Human Touch Feeling. In *IEEE Intl. Conference on Image Processing (ICIP)*, pages 1341–1344, Egypt, 2009.

- [22] W. Equitz and W. Niblack. Retrieving Images From a Database Using Texture-Algorithms From the QBIC System. Technical Report RJ 9805, IBM Research, 1994.
- [23] H. Farid. Creating and Detecting Doctored and Virtual Images: Implications to the Child Pornography Prevention Act. Technical Report 2004-518, Dartmouth College, USA, 2004.
- [24] H. Farid. A 3-D Photo Forensic Analysis of the Lee Harvey Oswald Backyard Photo. Technical Report TR2010-669, Dartmouth College, USA, 2010.
- [25] H. Farid and M. J. Bravo. Image Forensic Analyses that Elude the Human Visual System. In *SPIE Symposium on Electronic Imaging (SEI)*, CA, 2010.
- [26] T. Fawcett. An Introduction to ROC Analysis. *Pattern Recognition Letters*, 27(8):861–874, 2006.
- [27] Y. Freund and R. E. Schapire. Experiments with a new boosting algorithm. In *International Conference on Machine Learning*, pages 148–156, 1996.
- [28] A. C. Gallagher and T. Chen. Image Authentication by Detecting Traces of Demosaicing. In *IEEE Intl. Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, USA, 2008.
- [29] T. Gloe, M. Kirchner, A. Winkler, and R. Bohme. Can We Trust Digital Image Forensics? In *ACM Multimedia (ACMMM)*, pages 78–86, Germany, 2007.
- [30] R. Gonzalez and R. Woods. *Digital Image Processing*. Prentice-Hall, 3 edition, 2007.
- [31] R. M. Haralick, K. Shanmugam, and I. Dinstein. Textural Features for Image Classification. *IEEE Transactions on Systems, Man, and Cybernetics (SMC)*, 3(6):610–621, 1973.
- [32] G. E. Healey and R. Kondepudy. Radiometric CCD Camera Calibration and Noise Estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 16(3):267–276, 1994.
- [33] T. K. Ho. The random subspace method for constructing decision forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 20(8):832–844, 1998.
- [34] R. W. G. Hunt. *The Reproduction of Colour*. Voyageur Press, 6 edition, 1975.

- [35] M. K. Johnson and H. Farid. Detecting Photographic Composites of People. In *Intl. Workshop on Digital Watermarking (IWDW)*, China, 2007.
- [36] G. Kutyniok and W.-Q. Lim. Compactly Supported Shearlets are Optimally Sparse. *Journal of Approximation Theory*, pages 1564–1589, 2011.
- [37] A. B. Lee, K. S. Pedersen, and D. Mumford. The Nonlinear Statistics of High-Contrast Patches in Natural Images. *Intl. Journal of Computer Vision (IJCV)*, 54:83–103, 2003.
- [38] M. S. Lew, N. Sebe, C. Djeraba, and R. Jain. Content-based multimedia information retrieval: State of the art and challenges. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 2(1):1–19, 2006.
- [39] W. Li, T. Zhang, E. Zheng, and X. Ping. Identifying Photorealistic Computer Graphics Using Second-Order Difference Statistics. In *Intl. Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, volume 5, pages 2316–2319, China, 2010.
- [40] L. S. Liebovitch and T. Toth. A Fast Algorithm to Determine Fractal Dimensions by Box Counting. *Physics Letters A*, 141:386–390, 1989.
- [41] S. Loncaric. A survey of Shape Analysis Techniques. *Pattern Recognition*, 31(8):983–1001, 1998.
- [42] O. Ludwig, D. Delgado, V. Goncalves, and U. Nunes. Trainable Classifier-Fusion Schemes: An Application to Pedestrian Detection. In *IEEE Intl. Conference on Intelligent Transportation Systems (ITSC)*, pages 1–6, 2009.
- [43] J. Lukas, J. Fridrich, and M. Goljan. Digital Camera Identification from Sensor Noise Sensor. *IEEE Transactions on Information Forensics and Security (TIFS)*, 1(2):205–214, 2006.
- [44] S. Lyu and H. Farid. How Realistic is Photorealistic? *IEEE Transactions on Signal Processing (TSP)*, 53(2):845–850, 2005.
- [45] J. Ma and G. Plonka. The Curvelet Transform, a Review of Recent Applications. *IEEE Signal Processing Magazine (SPM)*, 27(2):118–133, 2010.
- [46] T. Mäenpää and M. Pietikäinen. Multi-Scale Binary Patterns for Texture Analysis. In *Scandinavian Conference on Image analysis (SCIA)*, pages 885–892, Sweden, 2003.
- [47] S. Mallat. *A Wavelet Tour of Signal Processing*. Academic Press, 1999.

- [48] B. Mandelbrot. How Long Is the Coast of Britain? Statistical Self-Similarity and Fractional Dimension. *Science*, 156(3775):636–638, 1967.
- [49] D. Meyer, F. Leisch, and K. Hornik. The Support Vector Machine Under Test. *Neurocomputing*, 55(1–2):169–186, 2003.
- [50] T.-T. Ng and S.-F. Chang. Identifying and Prefiltering Images – Distinguishing Between Natural Photography and Photorealistic Computer Graphics. *IEEE Signal Processing Magazine (SPM)*, 26(2):49–58, 2009.
- [51] T.-T. Ng, S.-F. Chang, J. Hsu, L. Xie, and M.-P. Tsui. Physics-Motivated Features for Distinguishing Photographic Images and Computer Graphics. In *ACM Multimedia (ACMMM)*, pages 239–248, Singapore, 2005.
- [52] T. Ojala, M. Pietikäinen, and T. Mäenpää. A Generalized Local Binary Pattern Operator for Multiresolution Gray Scale and Rotation Invariant Texture Classification. In *Intl. Conference on Advances in Pattern Recognition (ICAPR)*, pages 399–408, Brazil, 2001.
- [53] T. Ojala, M. Pietikäinen, and T. Mäenpää. Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns. In *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, volume 24, 2002.
- [54] G. Palmer. A Road Map for Digital Forensic Research. Technical Report DTR-T001-01, Air Force Research Laboratory, 2001.
- [55] B. T. Phong. Illumination for Computer Generated Pictures. *Communications of the ACM (CACM)*, 18(6):311–317, 1975.
- [56] S.-M. Phoong, C. Kim, P. Vaidyanathan, and R. Ansari. A New Class of Two-Channel Biorthogonal Filter Banks and Wavelet Bases. *IEEE Transactions on Signal Processing (TSP)*, 43(3):649–665, 1995.
- [57] M. Ponti. Combining classifiers: From the creation of ensembles to the decision fusion. In *Intl. Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 1–10, 2011.
- [58] 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.
- [59] T. Pouli and E. Reinhard. Image Statistics and their Applications in Computer Graphics. Technical report, Eurographics State of the Art Report (STAR), 2010.

- [60] R. C. Prati, G. E. Batista, and M. C. Monard. Curvas ROC para Avaliação de Classificadores. *Revista IEEE América Latina*, 6(2), 2008.
- [61] A. Rocha and S. Goldenstein. Is it Fake or Real? In *Intl. Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 1–2, Brazil, 2006.
- [62] A. Rocha and S. Goldenstein. PR: More than Meets the Eye. In *Intl. Conference on Computer Vision (ICCV)*, pages 1–8, Brazil, 2007.
- [63] A. Rocha and S. Goldenstein. Progressive Randomization Process and Equipment for Multimedia Analysis and Reasoning. Patent PCT/BR2007/000156, World Intellectual Property Organization (WIPO), 2008.
- [64] A. Rocha and S. Goldenstein. Progressive Randomization: Seeing the Unseen. *Elsevier Computer Vision and Image Understanding (CVIU)*, 114(3):349–362, 2010.
- [65] A. Rocha, J. P. Papa, and L. A. Meira. How Far You Can Get Using Machine Learning Black-Boxes. In *Intl. Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 193–200, Brazil, 2010.
- [66] A. Rocha, W. Scheirer, T. Boult, and S. Goldenstein. Vision of the Unseen: Current Trends and Challenges in Digital Image and Video Forensics. *ACM Computing Surveys (CSUR)*, 43:26:1–42, 2011.
- [67] Y. Rui, T. S. Huang, and S.-F. Chang. Image Retrieval: Current Techniques, Promising Directions, and Open Issues. *Journal of Visual Communication and Image Representation*, 10(1):39–62, 1999.
- [68] G. Sankar, V. Zhao, and Y.-H. Yang. Feature Based Classification of Computer Graphics and Real Images. In *IEEE Intl. Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 1513–1516, USA, 2009.
- [69] W. Scheirer, A. Rocha, R. Micheals, and T. Boult. Robust Fusion: Extreme Value Theory for Recognition Score Normalization. In *European Conference on Computer Vision (ECCV)*, volume 6313, pages 481–495, USA, 2010.
- [70] W. R. Schwartz, R. D. da Silva, L. S. Davis, and H. Pedrini. A Novel Feature Descriptor Based on the Shearlet Transform. In *IEEE Intl. Conference on Image Processing (ICIP)*, pages 1053–1056, Belgium, 2011.
- [71] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-based image retrieval at the end of the early years. In *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, volume 22, pages 1349–1380, 2000.

- [72] N. Sochen, R. Kimmel, and R. Malladi. A General Framework for Low Level Vision. *IEEE Transactions on Image Processing (TIP)*, 7(3):310–318, 1998.
- [73] P. P. Vaidyanathan. Quadrature Mirror Filter Banks, M-band Extensions and Perfect-Reconstruction Techniques. In *IEEE Intl. Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 4, pages 4–20, USA, 1987.
- [74] Y. Wang and P. Moulin. On Discrimination Between Photorealistic and Photographic Images. In *IEEE Intl. Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 2, page II, France, 2006.
- [75] D. H. Wolpert. Stacked generalization. *Neural Networks*, 5:241–259, 1992.
- [76] D. H. Wolpert. The Lack of A Priori Distinctions Between Learning Algorithms. *Neural Computation*, 8(7):1341–1390, 1996.