

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DAINF - DEPARTAMENTO ACADÊMICO DE INFORMÁTICA
BACHARELADO EM SISTEMAS DE INFORMAÇÃO

JORDY JACKSON ANTUNES DA ROCHA
JULIA ULSON TRETEL
RODRIGO MACHADO

**ANÁLISE COMPARATIVA DE ASSINATURAS DIGITAIS
PARA VÍDEOS**

TRABALHO DE CONCLUSÃO DE CURSO

CURITIBA
2018

JORDY JACKSON ANTUNES DA ROCHA
JULIA ULSON TRETEL
RODRIGO MACHADO

**ANÁLISE COMPARATIVA DE ASSINATURAS DIGITAIS
PARA VÍDEOS**

Trabalho de Conclusão de Curso apresentado ao curso de Bacharelado em Sistemas de Informação da Universidade Tecnológica Federal do Paraná, como requisito parcial para a obtenção do título de Bacharel.

Orientador: Prof. Dr. Ricardo Dutra da Silva

Orientador: Prof. Dr. Rodrigo Minetto

CURITIBA
2018



TERMO DE APROVAÇÃO

“ANÁLISE COMPARATIVA DE ASSINATURAS DIGITAIS PARA VÍDEOS”

por

“JORDY JACKSON ANTUNES DA SILVA, JULIA ULSON TRETEL e RODRIGO MACHADO”

Este Trabalho de Conclusão de Curso foi apresentado no dia **27 de NOVEMBRO de 2018** como requisito parcial à obtenção do grau de Bacharel em Sistemas de Informação na Universidade Tecnológica Federal do Paraná - UTFPR - Câmpus Curitiba. O(a)s aluno(a)s foi(ram) arguido(a)s pelos membros da Banca de Avaliação abaixo assinados. Após deliberação a Banca de Avaliação considerou o trabalho

<hr/> <p><Prof. Ricardo Dutra da Silva> (Presidente - UTFPR/Curitiba)</p>	<hr/> <p><Prof. Rodrigo Minetto> (Orientador - UTFPR/Curitiba)</p>
<hr/> <p><Prof. Bogdan Tomoyuki Nassu> (Avaliador 1 - UTFPR/Curitiba)</p>	<hr/> <p><Profa. Leyza Baldo Dorini> (Avaliadora 2 e Professora Responsável pelo TCC – UTFPR/Curitiba)</p>
<hr/> <p><Prof. Leonelo Dell Anhol Almeida> (Coordenador do curso de Bacharelado em Sistemas de Informação – UTFPR/Curitiba)</p>	

“A Folha de Aprovação assinada encontra-se na Coordenação do Curso.”

Agradecimentos

Agradecemos especialmente às nossas famílias pelo suporte e incentivo durante o curso, aos nossos orientadores Ricardo Dutra Da Silva e Rodrigo Minetto por toda a dedicação, paciência e zelo, e também a todos os professores que foram tão importantes para que pudéssemos chegar até aqui.

RESUMO

Rocha, J. J. A, Tretel, J. U, Machado, R.. Análise Comparativa de Assinaturas Digitais para Vídeos. 2018. 53 f. Trabalho de Conclusão de Curso – Bacharelado em Sistemas de Informação, Universidade Tecnológica Federal do Paraná. Curitiba, 2018.

Em plataformas digitais de compartilhamento de vídeo, é comum a violação de direitos autorais através do upload de material com *copyright*. Muitos fraudadores dificultam a identificação, aplicando modificações nos vídeos originais para que não sejam identificados tão facilmente — modificações essas conhecidas como ataques. Devido ao número elevado de vídeos na internet, a identificação manual de fraudes é inviável, abrindo espaço para processos automatizados de reconhecimento de cópias. Para realizar estes processos, diversos algoritmos para cálculo de assinaturas digitais foram desenvolvidos. Neste trabalho, foram selecionados e comparadas sete assinaturas de vídeo, baseadas em informações temporais e espaciais, para análise da capacidade de reconhecimento de fraudes. Em nossos experimentos, verificamos que assinaturas com características temporais são menos eficientes, sendo sensíveis a alterações temporais e fotométricas, enquanto que algoritmos espaciais são sensíveis a alterações temporais.

Palavras-chave: Assinatura Digital de Vídeo, Descritor de vídeo, Cópias de Vídeos

LISTA DE FIGURAS

Figura 1 – Composição hierárquica de um vídeo.	15
Figura 2 – Sequência com cinco quadros representando uma cena, retirado de um dos vídeos disponibilizados por (REDDY; SHAH, 2013).	16
Figura 3 – Exemplos de ataques em uma imagem.	17
Figura 4 – Exemplo de divisão em blocos, cálculo das intensidades médias e ordem atribuída a cada valor.	20
Figura 5 – Diagrama do algoritmo baseado em medida ordinal.	21
Figura 6 – Diagrama do algoritmo baseado em gradientes.	22
Figura 7 – Linha azul mostra os valores originais do vetor dY e a linha alaranjada mostra os valores pós filtro passa-baixa.	23
Figura 8 – Diagrama do algoritmo baseado na diferença entre quadros.	24
Figura 9 – Diagrama do algoritmo baseado em quadros de cena.	24
Figura 10 – Divisão da imagem para cálculo dos elementos diferenciais.	25
Figura 11 – Divisão de um quadro em regiões circulares.	26
Figura 12 – Divisão de um quadro em regiões fatias.	27
Figura 13 – Diagrama do algoritmo baseado padrões binários por região.	28
Figura 14 – Diagrama do algoritmo baseado em wavelets.	29
Figura 15 – Movimento da câmera entre dois quadros consecutivos. Os pontos as direita representam os vetores de movimento. Neste caso há uma combinação de <i>pan</i> para a esquerda e <i>tilt</i> para cima.	29
Figura 16 – Diagrama do algoritmo de Camera Motion.	30
Figura 17 – Diagrama das etapas de desenvolvimento.	31
Figura 18 – Exemplos de vídeos retirados da base. O primeiro mostra um jogo de beisebol, o segundo mostra uma criança fazendo malabarismo, o terceiro mostra uma competição de levantamento de pesos.	32
Figura 19 – Distância entre duas sequências temporais medida usando distância Euclidiana (na imagem de cima) e o DTW (na imagem de baixo).	35
Figura 20 – a) Matriz de custo formada comparando duas sequências temporais. b) Caminho com menor custo.	35
Figura 21 – Diagrama das comparações dos casos de teste.	37
Figura 22 – Diagrama da divisão dos casos de teste para parametrização e testes.	37

Figura 23 – Exemplo de simulação de classificação para um tipo de assinatura. O eixo x é composto dos limiares testados para a assinatura. O ponto vermelho indica o valor máximo de <i>F-measure</i> , ponto em que o limiar apresenta o melhor resultado. F1 representa o <i>F-measure</i>	40
Figura 24 – Mapa de calor de revocação, precisão e fmeasure de cada tipo de assinatura com cada tipo de distorção. Revocação é definida na Seção 4.1, precisão é definida na Seção 4.2 e <i>F-measure</i> é definido na Seção 4.3.	41
Figura 25 – Histogramas de resultados de comparações para a assinatura Camera Motion.	44
Figura 26 – Histogramas de resultados de comparações para a assinatura FrameDiff.	44
Figura 27 – Simulação de classificação para cada um dos tipos de assinatura. A medida que o limiar se afasta do zero, o valor de precisão diminui e o de revocação aumenta. Cada tipo de assinatura tem uma proporção diferente para essa variação.	45
Figura 28 – Exemplo de combinação de assinaturas.	46
Figura 29 – Histograma com valores de fmeasure para cada tipo de assinatura. Em cima, resultados da combinação da assinatura Camera Motion com as demais, embaixo, a diferença entre os resultados com e sem a combinação das assinaturas.	48

LISTA DE TABELAS

Tabela 1 – Classificação de ataques em vídeo.	18
Tabela 2 – Parâmetros usados na aplicação das distorções.	33
Tabela 3 – Assinaturas utilizadas para comparação.	34
Tabela 4 – Limiares obtidos nas simulações com cada subconjunto de parametrização.	39

Sumário

1 – Introdução	11
1.1 Objetivo Geral	12
1.2 Objetivos Específicos	12
1.3 Estrutura do trabalho	13
2 – Revisão da Literatura e Conceitos Base	14
2.1 Vídeo Digitais	15
2.2 Técnicas de detecção de cópias de vídeo	16
2.3 Tipos de ataques em vídeos	16
2.4 Assinaturas de vídeo	18
2.5 Algoritmos para geração de assinaturas	19
2.5.1 Assinatura baseada na medida ordinal	19
2.5.2 Assinatura baseada em gradientes	20
2.5.3 Assinatura baseada na diferença entre quadros	21
2.5.4 Assinatura baseada em quadros de cena	22
2.5.5 Assinatura baseada em padrões binários por região	26
2.5.6 Assinatura baseada em wavelets	27
2.5.7 Assinatura baseada no movimento da câmera	28
3 – Metodologia	31
3.1 Criação da Base de Vídeos	31
3.2 Geração das Assinaturas	32
3.3 Comparação de Assinaturas	34
3.4 Experimentos	36
4 – Análise e Discussão dos Resultados	39
4.1 Robustez	40
4.2 Unicidade	43
4.3 Peso da escolha do limiar sobre a detecção de cópias	43
4.4 Combinação de Assinaturas	45
5 – Conclusão	49
5.1 Trabalhos Futuros	49

Referências	51
--------------------	-------	-----------

1 Introdução

A cada minuto é realizado o *upload* de aproximadamente 300 horas de vídeo apenas na plataforma *YouTube*¹, segundo os dados da pesquisa realizada pelo site *Statistic Brain*², em 2016. É provável que essa estatística seja ainda mais expressiva atualmente, devido à popularização cada vez maior de dispositivos móveis e à democratização do acesso à internet. O *YouTube* é um dos serviços de hospedagem de vídeos mais utilizados na internet, em conjunto com diversas outras empresas que oferecem serviços similares, como, por exemplo, *Vimeo*, *Twitch*, *DailyMotion* e, mais recentemente o *Facebook*.

É comum que essas plataformas de compartilhamento de vídeos recebam pedidos vindos de criadores de conteúdo digital (como estúdios de cinema, cineastas independentes ou “vloggers”) solicitando a retirada de determinados vídeos alegando o infringimento de direitos autorais. Segundo estes criadores, seus vídeos estão sendo reproduzidos parcial ou integralmente sem autorização, caracterizando uma situação de pirataria e apropriação indevida de conteúdo.

Enquanto o número de pedidos de retirada é baixo não há grandes problemas em definir se o caso realmente se trata de uma cópia, pois um humano pode analisar o vídeo e definir se realmente se trata de um caso de plágio. Entretanto, à medida que o número de pedidos cresce, levando em conta a quantidade de *uploads*, torna-se inviável a realização desse processo de forma manual, gerando um problema para as plataformas, que precisam buscar métodos automáticos para identificação de vídeos.

Uma das técnicas já adotadas pelas empresas de compartilhamento de vídeos é a identificação de vídeos via áudio. Um exemplo é o serviço chamado *Content ID*, da Audible Magic Corporation, que disponibiliza comercialmente uma base de dados global para verificação de conteúdos protegidos por direitos autorais ([AUDIBLE MAGIC CORPORATION, 2018](#)). Tomando por exemplo o *YouTube*, um dos seus sistemas de prevenção de cópias analisa o áudio de cada vídeo enviado ([KING, 2010](#)). O áudio é então comparado com uma base de dados para verificar se aquele conteúdo está de acordo com as políticas anti-pirataria da plataforma e, caso seja encontrada alguma irregularidade, o vídeo é rejeitado e a pessoa que realizou o *upload* é notificada que está infringindo os termos de uso. Essa abordagem é eficaz para casos específicos, como a reprodução ilegal de um videoclipe ou a utilização ilícita de trilhas sonoras protegidas por lei. Porém, não

¹www.youtube.com/

²www.statisticbrain.com/youtube-statistics/

é totalmente satisfatória para reconhecer se o conteúdo do vídeo está sendo utilizado indevidamente, justamente por utilizar o áudio para detectar duplicatas, não o vídeo. Outra característica negativa dessa técnica é que longas-metragens normalmente têm versões dubladas para cada país onde são distribuídas, aumentando o problema para detectar cópias.

O problema de detecção de duplicatas se torna mais complexo quando os piratas utilizam técnicas de modificação nos vídeos, também conhecidas como ataques, fazendo com que sistemas de identificação mais especializados sejam necessários. Essas modificações podem ser sutis, como a remoção de alguns quadros do vídeo ou a alteração do formato de compressão, ou mais agressivas, como a modificação das cores, espelhamento, rotação e inserção de bordas nos vídeos. O desafio, entretanto, é que devido à natureza de cada ataque, pode-se fazer necessária a utilização de diferentes formas de análise, já que um sistema para identificar um ataque de rotação pode ter dificuldade em detectar um ataque de remoção de quadros, por exemplo.

Como uma possível solução complementar para esse problema, propomos um estudo de sete algoritmos para identificar um vídeo em relação ao seu conteúdo ([HUA; CHEN; ZHANG, 2004](#)), ([LEE; YOO, 2008](#)), ([COOK, 2011](#)), ([MAO et al., 2016](#)), ([KIM; LEE; RO, 2014](#)), ([DUTTA; SAHA; CHANDA, 2013](#)), ([MINETTO; LEITE; STOLFI, 2007](#)). Essa identificação é feita através da extração de uma assinatura do vídeo, que deve ser robusta aos ataques mais comuns que um vídeo pode sofrer. O processo de produção das assinaturas varia de acordo com o algoritmo utilizado e as assinaturas são basicamente características locais, globais, espaciais ou temporais dos vídeos.

1.1 Objetivo Geral

Este trabalho tem como objetivo comparar descritores para assinatura digital de vídeos, e investigar a complementaridade de descritores espaciais em conjunto com uma proposta de descritor temporal.

1.2 Objetivos Específicos

- Implementar sete algoritmos para gerar assinaturas digitais de vídeos.
- Criar um repositório com 1265 vídeos que sofrerão 14 tipos de ataques diferentes para testar a robustez dos algoritmos.
- Analisar a eficiência de cada tipo de assinatura quanto a robustez e unicidade.

1.3 Estrutura do trabalho

Este trabalho está estruturado como segue. No Capítulo 2 são apresentados conceitos básicos, uma introdução aos métodos de detecção de conteúdo em vídeos, os sete algoritmos selecionados para a geração de assinaturas e trabalhos relacionados. O Capítulo 3 detalha a base de vídeos utilizada, a forma como o experimento foi realizado e as métricas de comparação. No Capítulo 4 são apresentados os resultados experimentais. Finalmente, são apresentadas as conclusões do estudo e sugestões de trabalhos futuros.

2 Revisão da Literatura e Conceitos Base

Em 1999, o compartilhamento de vídeos na Internet ainda era uma realidade distante, entretanto, [Indyk, Iyengar e Shivakumar \(1999\)](#) já buscavam métodos para encontrar vídeos pirateados e duplicatas na rede. Os autores então propuseram um algoritmo para geração de assinaturas temporais baseadas na transição de tomadas de um vídeo. Embora este método seja bom para encontrar filmes e vídeos longos, ele não é apropriado para identificar vídeos curtos e cenas isoladas, com duração de até quatro minutos. Esse tipo de produção é a mais comum nas plataformas de compartilhamento de vídeos atualmente ([LELLA, 2018](#)).

Desde então, várias técnicas têm sido usadas para a geração de assinaturas de vídeos. [Coskun, Sankur e Memon \(2006\)](#) propuseram o conceito de funções de *hash* como uma ferramenta para identificação de vídeos, criando um algoritmo que utiliza as informações espaço-temporais do vídeo baseado na diferença da luminância entre regiões de quadros. Outras abordagens incluem o uso de descritores globais que utilizam direção de movimento, distribuição de cor e medida ordinal ([HAMPAPUR; HYUN; BOLLE, 2001](#)), além de medidas ordinais ([HUA; CHEN; ZHANG, 2004](#)), que se provaram robustas para variadas resoluções, mudanças de iluminação e formatos de vídeos.

Assinaturas baseadas em características locais também foram pesquisadas, como em ([JOLY; BUISSON; FRELICOT, 2007](#)), cujo algoritmo apresentado procura ser eficiente para buscas em grandes bases de dados, eficiência essa, tanto na velocidade da busca quanto na qualidade dos resultados. Há também a pesquisa de [Law-To et al. \(2006\)](#), que usa o algoritmo de Harris para encontrar pontos de interesse no vídeo e criar uma assinatura compacta.

[Andrade et al. \(2012\)](#) fizeram um estudo comparativo entre assinaturas globais e locais e mostraram como unir os dois tipos de assinaturas utilizando algoritmos genéticos. [Andrade et al. \(2012\)](#) também fizeram experimentos mostrando que a combinação de descritores globais e locais são complementares e, se usados em conjunto, produzem resultados superiores quando comparados com o uso individual de cada tipo de descritor.

[Hu et al. \(2011\)](#) discorrem sobre a indexação e recuperação de conteúdo em vídeos. O trabalho apresenta métodos para analisar a estrutura de vídeos, segmentação de cenas, extração de quadros-chave, características de movimento, mineração de informações em vídeos, mensuramento de similaridade e relevância entre assinaturas digitais, pesquisa de conteúdo em vídeos, entre outros.

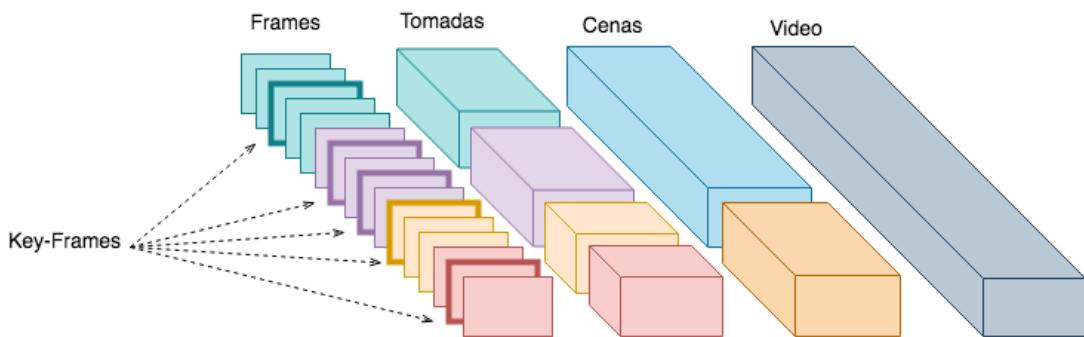
As Seções 2.1 a 2.4 apresentam definições úteis para melhor compreensão do trabalho, como a estrutura de um vídeo, técnicas de detecção de cópias e conteúdo, principais tipos de ataque, e, finalmente, a definição de assinatura digital para vídeos e os algoritmos selecionados para este trabalho.

2.1 Vídeo Digitais

Um vídeo pode ser descrito quanto ao seu conteúdo em quatro níveis de detalhe, sendo o nível mais baixo um conjunto de quadros ([LIENHART; PFEIFFER; EFFELSBERG, 1997](#)). Um quadro (ou *frame*) I é uma imagem representada por uma matriz de altura h e largura w em que cada ponto $I(x,y)$ representa a intensidade de um pixel. Além disso, um quadro possui um determinado tempo, que representa o instante em que é exibido no vídeo. De maneira resumida, um vídeo é uma sequência de imagens. Normalmente são exibidas 30 imagens por segundo, sendo esse o conceito de quadros por segundo, ou FPS (*frames per second*).

Logo acima na hierarquia existem as tomadas, termo que se refere a um ou mais quadros capturados em sequência, representando uma ação ininterrupta no tempo e no espaço. Tomadas contínuas podem ser agrupadas em cenas para gerar coerência na história do vídeo. Um vídeo pode ser composto por uma ou mais cenas, como pode ser observado na Figura 1.

Figura 1 – Composição hierárquica de um vídeo.



Fonte: Autoria própria.

Um conceito importante também relacionado ao vídeo é o quadro chave (do inglês *key-frame*), ou quadro de cena. [Rachmadi, Uchimura e Koutaki \(2016\)](#) descrevem um quadro de cena como a imagem que define os pontos de início e fim de qualquer transição suave de imagens (o quadro que melhor representa a cena no geral). Quadros

de cena são amplamente utilizados na produção de animações, em que normalmente o objeto ou sujeito da imagem se move em relação ao fundo (*background*). De acordo com [Mao et al. \(2016\)](#), o quadro de cena deve possuir todos os elementos (pessoas, objetos, animais, etc.) exibidos na cena e o *background* deve ser altamente similar ao restante dos quadros. Na Figura 2, qualquer um dos cinco quadros pode ser eleito como o quadro de cena, pois todos têm os mesmos elementos (a mesma pessoa e o mesmo cachorro), além do *background* ser praticamente o mesmo em todos os quadros.

Figura 2 – Sequência com cinco quadros representando uma cena, retirado de um dos vídeos disponibilizados por ([REDDY; SHAH, 2013](#)).



Fonte: Autoria própria.

Em um vídeo existe também a dimensão espacial e a dimensão temporal. A dimensão espacial é classificada como a distribuição e a maneira com que os elementos estão organizados em um quadro. A dimensão temporal é a relação na qual os elementos e os quadros mudam ao longo de um vídeo ([HAMPAPUR; HYUN; BOLLE, 2001](#)).

2.2 Técnicas de detecção de cópias de vídeo

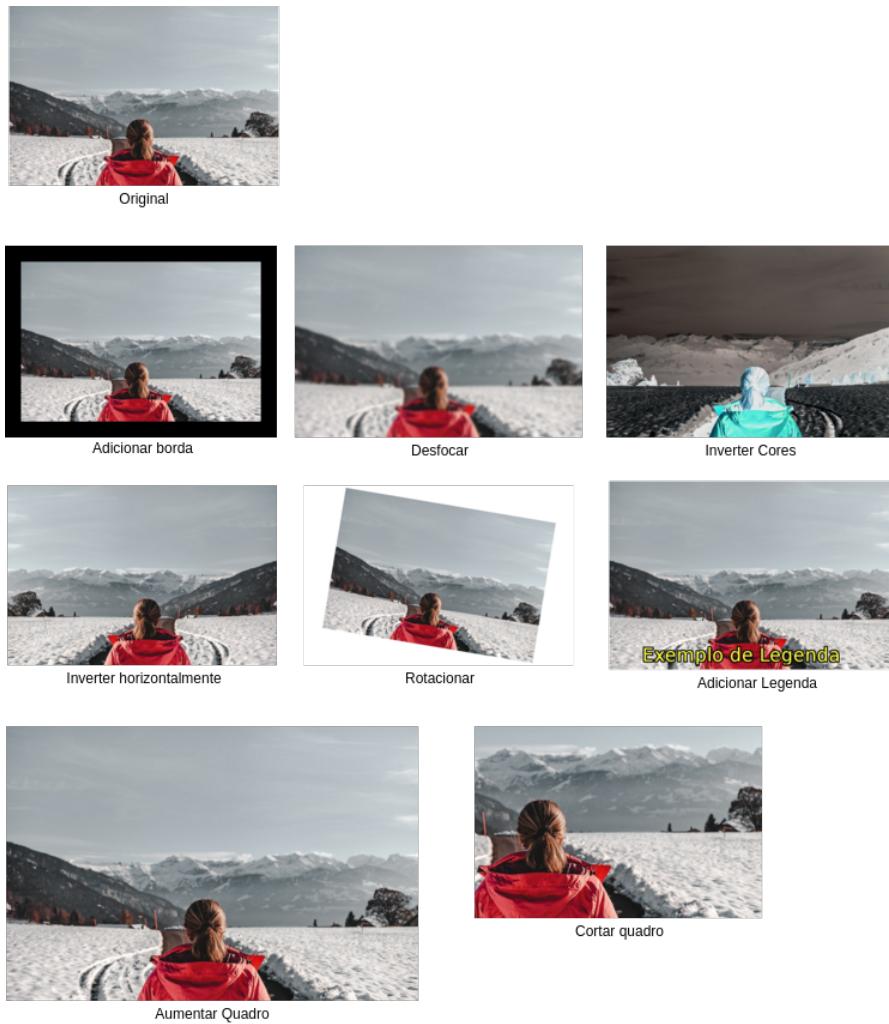
A detecção de cópias baseada em conteúdo (CBCD, do inglês *Content Based Copy Detection*) é uma técnica para identificar vídeos através da criação de uma assinatura digital baseada em seu conteúdo ([JIANG et al., 2011](#)). Apesar de o método ser utilizado em diversas aplicações, a detecção de cópias torna-se um desafio, considerando que a cópia pode sofrer ataques, distorções e transformações que dificultem a identificação do vídeo por um sistema automatizado.

2.3 Tipos de ataques em vídeos

Para evitar a detecção de duplicatas é comum a realização de ataques, ou distorções, nos vídeos. Existem várias maneiras de modificar um vídeo, como redimensionamento do tamanho, inserção ou remoção de alguns quadros, e alteração de cada quadro individualmente através da adição ou remoção de elementos, tais como bordas e legendas.

No escopo deste trabalho são estudados 14 tipos de ataques para simular as táticas mais comuns utilizadas pelos apropriadores de conteúdo que tentam dificultar o reconhecimento de duplicatas. Os ataques são: adição de texto e legendas no vídeo, adição de marca d'água, adição de quadro ou bordas, redimensionamento da altura e largura dos quadros, eliminação de uma faixa ou região dos quadros, inversão/espelelhamento, rotação, borramento, inversão de cores, alteração do formato de compressão dos quadros do vídeo, aceleração do vídeo e, finalmente, remoção de quadros. A Figura 3 ilustra o resultado da aplicação de 8 dos ataques descritos acima em uma imagem.

Figura 3 – Exemplos de ataques em uma imagem.



Fonte: Autoria própria.

Os ataques dos tipos remoção de quadros, alteração do formato de compressão e aceleração do vídeo só são perceptíveis quando visualizados em um dispositivo

de exibição, como um computador, e por isso não estão representados na Figura 3. O ataque do tipo remoção de quadros consiste em remover alguns quadros ao longo do vídeo, e muitas vezes passa despercebido ao olho humano. O maior indicativo de que um vídeo sofreu esse ataque é observar a diferença temporal entre uma duplicata e o vídeo original, sendo que a duplicata terá uma duração menor em relação ao original. O ataque do tipo alteração da taxa de quadros também é realizado através da remoção de alguns quadros do vídeo, entretanto, o vídeo permanece com a mesma duração, pois os quadros restantes são na verdade exibidos por mais tempo.

Os ataques podem ainda ser classificados como geométricos, fotométricos ou temporais. Ataques geométricos afetam as características globais dos vídeos, ataques fotométricos afetam as características locais e, os ataques temporais, a dimensão temporal dos vídeos. A classificação dos ataques que serão utilizados nos vídeos deste trabalho é exibida na Tabela 1.

Tabela 1 – Classificação de ataques em vídeo.

#	Ataque	Tipo
1	Aceleramento	Temporal
2	Adição de bordas	Fotométrico
3	Adição de texto (legendas)	Fotométrico
4	Alteração do formato de compressão	Fotométrico
5	Borramento	Fotométrico
6	Eliminação de faixa ou região	Geométrico
7	Espelhamento	Geométrico
8	Inversão de cores	Fotométrico
9	Marca d'água	Fotométrico
10	Redimensionamento	Geométrico
11	Remoção de quadros	Temporal
12	Rotação	Geométrico

2.4 Assinaturas de vídeo

Uma assinatura de vídeo é definida como um vetor de características que representa um vídeo e o diferencia de outros (LEE; YOO, 2008). Em outras palavras, a assinatura é uma representação de um vídeo em uma estrutura de dados.

Para um algoritmo de geração de assinatura ser considerado eficiente, é importante que três características sejam consideradas: robustez, singularidade e eficiência de busca. De acordo com Lee e Yoo (2008), uma assinatura é considerada robusta caso o descritor gerado para um vídeo modificado seja similar ao descritor do vídeo.

original. A singularidade, ou unicidade, é a capacidade do algoritmo gerar assinaturas diferentes para vídeos perceptivelmente diferentes. Por fim, eficiência de busca é a capacidade da assinatura ser utilizada por uma aplicação para buscas em banco de dados de larga escala. Nesta monografia, são avaliadas apenas a robustez e a singularidade das assinaturas.

Existem duas classes principais de algoritmos para descrever vídeos e então gerar assinaturas, são elas: locais e globais. Cada classe tem características que a faz mais robusta para combater diferentes tipos de ataques, sendo então cada uma indicada para situações diferentes.

Descritores locais geram assinaturas baseadas em pontos ou regiões de interesse de cada quadro do vídeo. Os pontos de interesse são determinados a partir de regiões que possuem uma acentuada variação na orientação do gradiente dos elementos presentes em um quadro, como, por exemplo, o enquadramento de uma porta ou o pico de uma montanha. As regiões de interesse são determinadas pelos *pixels* ao redor de um ponto de interesse e ajudam a determinar os limites dessas regiões ([RADHAKRISHNAN; BAUER, 2007](#)). Esses descritores normalmente são robustos contra variações fotométricas (borrados, variação de luminância e cor, ruído e compressões) e podem ser custosos computacionalmente devido aos cálculos necessários para determinar os pontos de interesse ([NAINI; RANE; RAMALINGAM, 2014](#)). Um descritor local consiste normalmente de três etapas: detecção das características, descrição das características e combinação das características ([CHEN; SUN, 2010](#)).

A classe de descritores globais, ao contrário dos descritores locais, geram assinaturas utilizando informações pertinentes ao quadro como um todo, como por exemplo a luminância total do quadro. Esses podem apresentar vantagens em relação aos descritores locais, pois normalmente é menos custoso em termos computacionais trabalhar com informações gerais do quadro do que realizar uma análise para determinar pontos de interesse. Além disso, os descritores globais podem gerar assinaturas mais robustas quando considerados os ataques geométricos às imagens ([LAW-TO et al., 2007](#)).

2.5 Algoritmos para geração de assinaturas

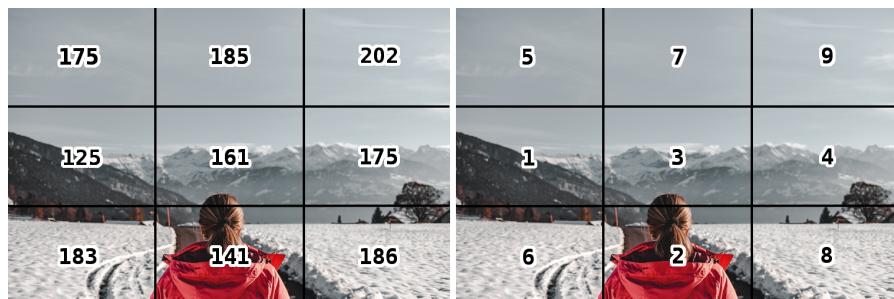
A seguir é apresentado o estudo dos sete algoritmos que foram selecionados e implementados para a realização dos experimentos.

2.5.1 Assinatura baseada na medida ordinal

O algoritmo global proposto por [Hua, Chen e Zhang \(2004\)](#) baseia-se na intensidade dos *pixels* de cada quadro para compor a assinatura. Primeiramente, a taxa de amostragem, ou seja, a taxa de quadros por segundo (FPS) do vídeo de entrada é padronizada, para que a assinatura gerada fique mais compacta e seja tolerante a diferentes formatos de compressão. Além disso, o vídeo é convertido para escala de cinza.

Após esse pré-processamento, cada quadro é particionado em $M \times N$ blocos, como pode ser observado na Figura 4, e a intensidade média para cada um dos blocos é computada. O descritor é então formado pela concatenação dos vetores em cada bloco. O método é resumido na Figura 5. O tamanho final de uma assinatura baseada em medida ordinal é $N \times M \times k$ onde k é o número de quadros de um vídeo. Para esta implementação, foi utilizado $M = 5$ e $N = 5$.

Figura 4 – Exemplo de divisão em blocos, cálculo das intensidades médias e ordem atribuída a cada valor.



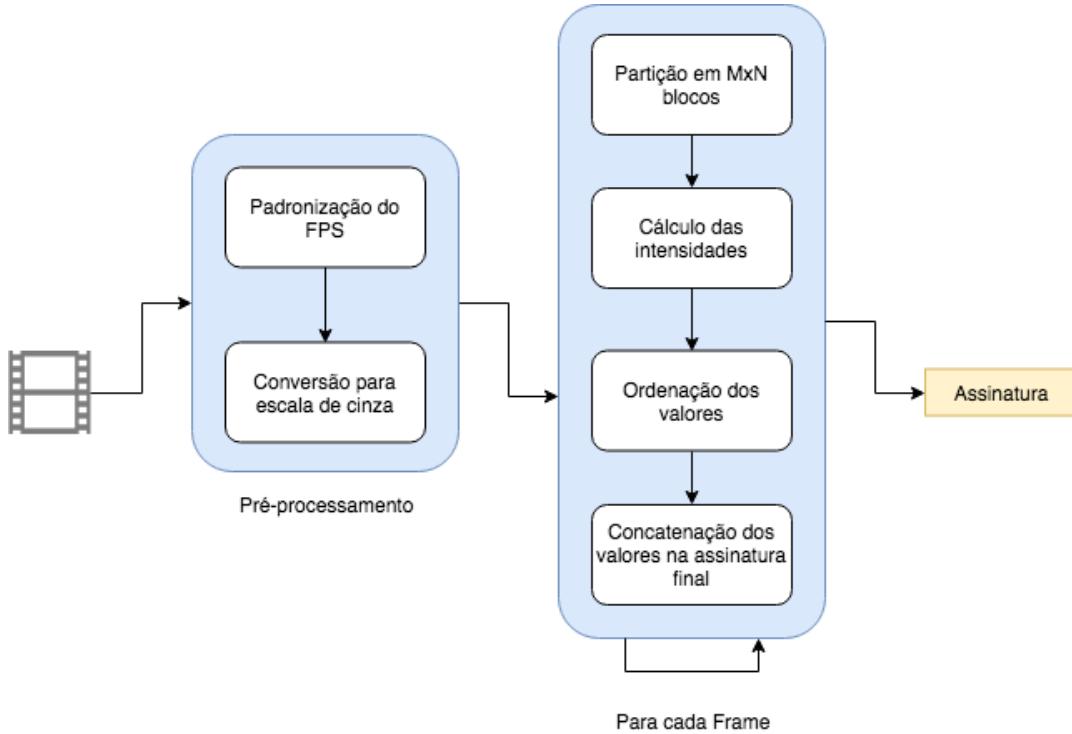
Fonte: Autoria própria.

2.5.2 Assinatura baseada em gradientes

O algoritmo global proposto por [Lee e Yoo \(2008\)](#) utiliza a distribuição dos gradientes para geração de assinaturas. O primeiro passo é definir uma taxa de quadros por segundo (FPS) fixa, além da conversão para escala de cinza. Também é realizado o redimensionamento dos quadros, tornando o método robusto independente da mudança de resolução do vídeo. Em seguida, os gradientes G_x e G_y dos pixels de cada quadro são calculados como mostra a Equação 1.

$$\begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \partial I / \partial x \\ \partial I / \partial y \end{bmatrix} = \begin{bmatrix} I(x+1, y) - I(x-1, y) \\ I(x, y+1) - I(x, y-1) \end{bmatrix} \quad (1)$$

Figura 5 – Diagrama do algoritmo baseado em medida ordinal.



Fonte: Autoria própria.

O quadro é então dividido em $M \times N$ blocos, para os quais é determinado o valor do centroide dos gradientes, criando assim um vetor com $M \times N$ elementos. Para isso, são computadas as imagens de magnitude $w(x,y)$ e a orientação $\Theta(x,y)$, conforme mostra a Equação 2.

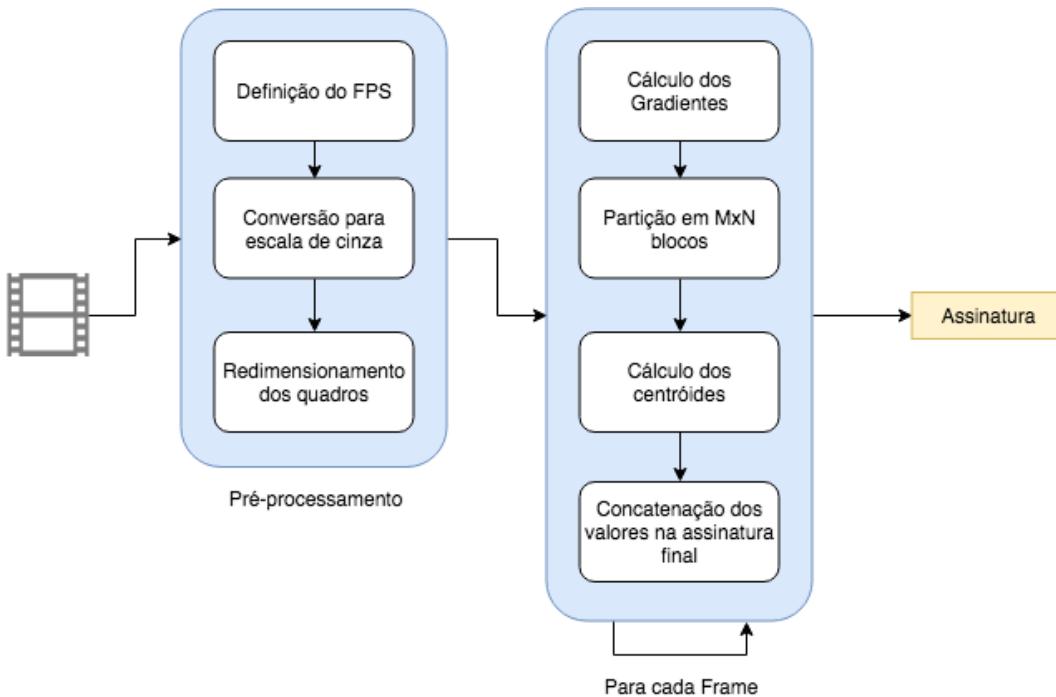
$$w(x,y) = \sqrt{\mathbb{G}x^2 + \mathbb{G}y^2} \quad \Theta(x,y) = \tan^{-1} \left(\frac{\mathbb{G}y}{\mathbb{G}x} \right) \quad (2)$$

Em seguida o centroide para cada bloco $b[i]$ é obtido a partir do somatório do produto da magnitude e orientação, dividido pela somatória de todas as magnitudes daquele bloco, como pode ser observado na Equação 3.

$$b[i] = \frac{\sum_{x,y \in b[i]} w(x,y)\Theta(x,y)}{\sum_{x,y \in b[i]} w(x,y)} \quad (3)$$

A assinatura final é obtida pela concatenação desses vetores (Figura 6). O tamanho final de uma assinatura baseada em gradiente é $N \times M \times k$ onde k é o número de quadros de um vídeo. Para esta implementação, foi utilizado $M = 3$ e $N = 4$.

Figura 6 – Diagrama do algoritmo baseado em gradientes.



Fonte: Autoria própria.

2.5.3 Assinatura baseada na diferença entre quadros

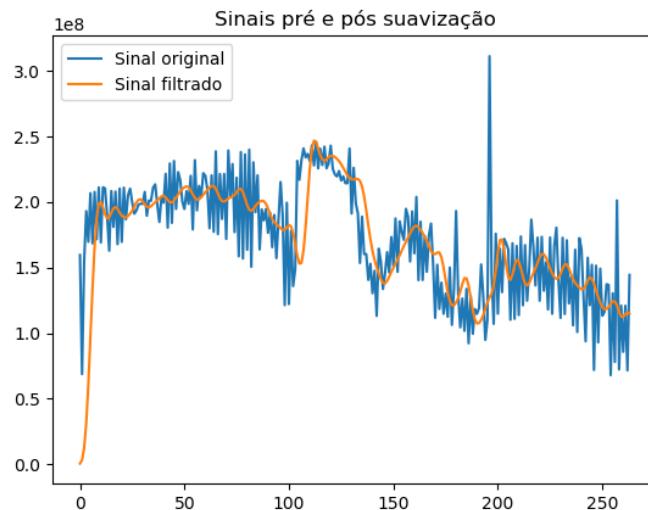
O algoritmo proposto por Cook (2011) utiliza características globais de luminância e de diferença de luminância intra-quadros. Para cada quadro do vídeo são coletadas características primárias, como a luminância total (Y), obtida através da soma da luminância de todos os pixels do quadro; a luminância máxima (Y_{max}), que é o valor do pixel mais brilhante do quadro; e a luminância diferencial (dY), que é a diferença absoluta de luminância pixel a pixel do quadro atual com o quadro que estava visível a 100 milissegundos, a diferença resultante é somada conforme mostra a Equação 4. Os valores obtidos são então normalizados, levando em conta a dimensão dos quadros do vídeo.

$$dY = \sum_{x,y \in I,J} |I(x,y) - J(x,y)| \quad (4)$$

Após a obtenção das características primárias, um filtro passa-baixa Gaussiano é utilizado para suavizar a assinatura, como pode ser observado na Figura 7. A assinatura é a concatenação dos somatórios das diferenças entre quadros e a luminância total de cada quadro. O tamanho final de uma assinatura baseada em diferença entre quadros é

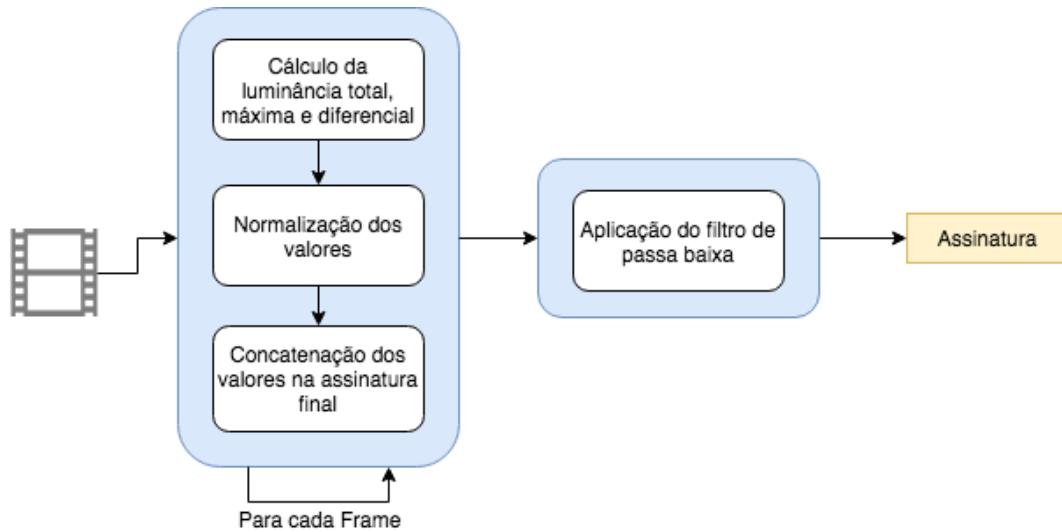
$2 \times k$ onde k é o número de quadros de um vídeo.

Figura 7 – Linha azul mostra os valores originais do vetor dY e a linha alaranjada mostra os valores pós filtro passa-baixa.



Fonte: Autoria própria.

Figura 8 – Diagrama do algoritmo baseado na diferença entre quadros.



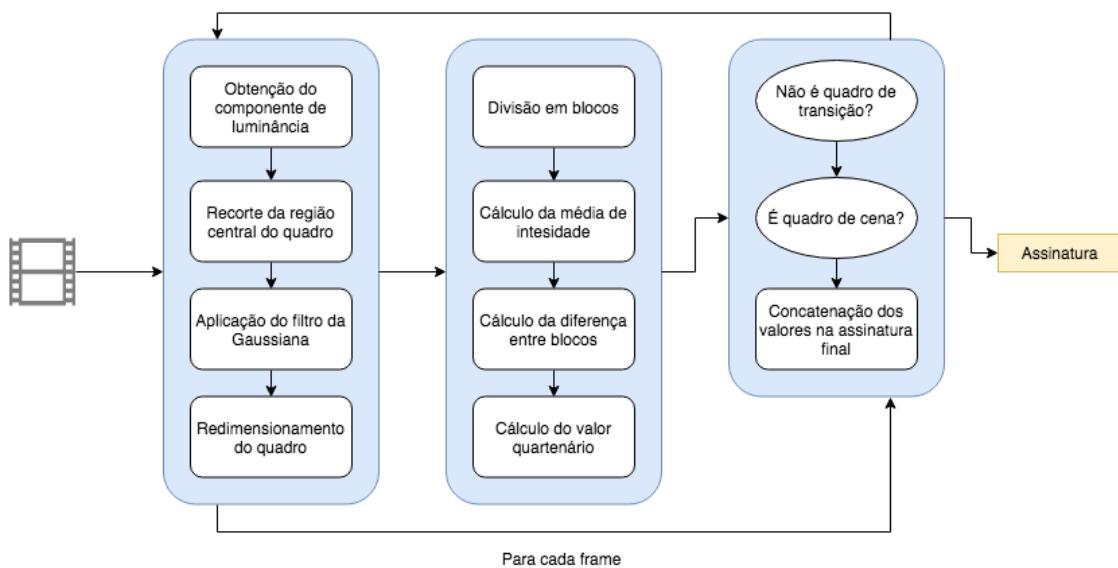
Fonte: Autoria própria.

2.5.4 Assinatura baseada em quadros de cena

A abordagem proposta por [Mao et al. \(2016\)](#) é baseada na assinatura de quadros de cena (Seção 2.1). De acordo com os autores, os quadros de cena podem ser *intraframes*, ou seja, quadros que iniciam tomadas, quanto *interframes*, contanto que sigam as características descritas na Seção 2.1, ou seja, possuir todos os mesmos elementos dos quadros restantes e o mesmo *background*. Esse é um algoritmo que gera uma assinatura do tipo global.

A assinatura individual de cada quadro é calculada conforme os passos descritos na Figura 9. Os quadros passam por um pré-processamento, onde o componente de luminância pertencente ao espaço HSL (*hue*, *saturation*, *lightness*) da imagem é extraído, uma vez que apenas este valor é considerado pelo algoritmo. O quadro é recortado mantendo-se apenas sua região central e redimensionado para o tamanho definido de 108×132 pixels. Além disso, é aplicado o filtro passa-baixa Gaussiano.

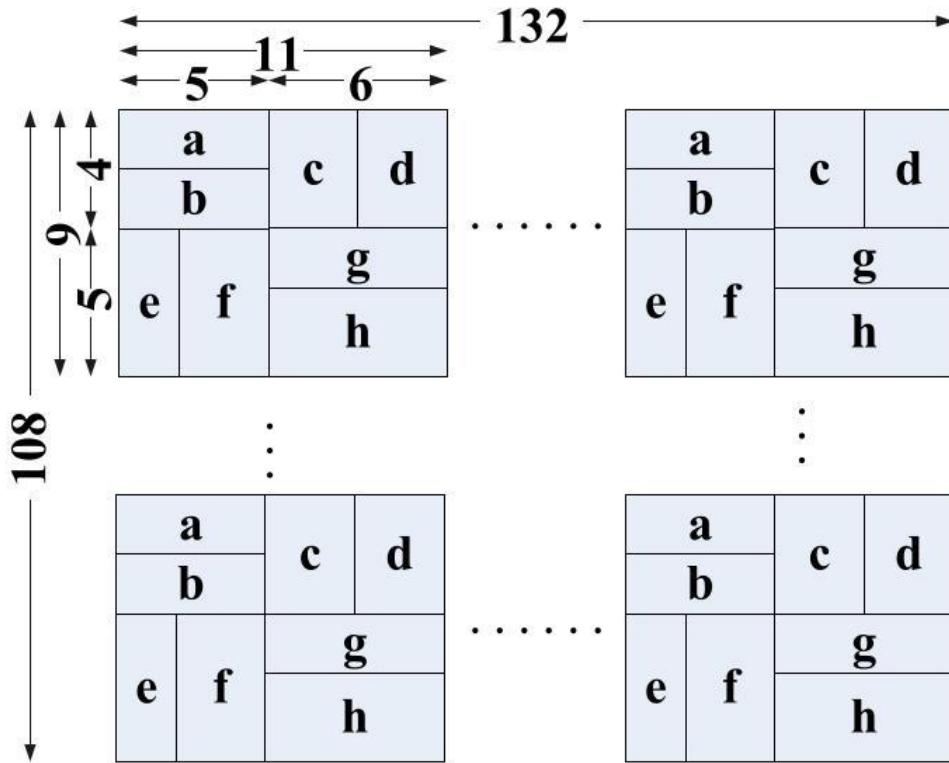
Figura 9 – Diagrama do algoritmo baseado em quadros de cena.



Fonte: Autoria própria.

Após o processamento inicial, o quadro é então dividido em 144 pedaços menores, de tamanho 9×11 pixels, cuja intensidade média irá compor parte da assinatura deste quadro. Além dos 144 valores, o descritor é composto também por 576 elementos diferenciais, totalizando 720 valores. Para obter esses elementos diferenciais, cada fragmento é dividido em oito elementos menores, conforme a Figura 10, e então é realizada a subtração de $a - b$, $c - d$, $e - f$ e $g - h$.

Figura 10 – Divisão da imagem para cálculo dos elementos diferenciais.



Fonte: ([MAO et al., 2016](#)).

O artigo também propõe uma alternativa para diminuir o espaço de memória utilizado para armazenar as assinaturas, visto que o banco de dados dos vídeos pode ser grande. Para isso, é proposta uma técnica chamada quantificação quaternária, na qual os valores são classificados de acordo com um limiar, calculado dinamicamente para cada quadro. O primeiro limiar, utilizado para as 144 médias de intensidade, é obtido da seguinte maneira: considere que m_i representa cada valor dentro do conjunto M de médias de intensidade. Para todo m_i em M , é calculado o valor absoluto $a_i = |m_i - 128|$, obtendo novos valores A . Os valores de A são, então, organizados em ordem crescente. Para separar os valores em 4 faixas, é utilizado o triségimo-sexto valor do conjunto ordenado A , definindo o limiar ThM . O limiar ThM utilizado será a_i onde

$i = \lfloor 0.25 * 144 \rfloor$. Finalmente, cada valor m_i é então definido através da Equação 5.

$$m_i = \begin{cases} 3, & \text{Se } m_i - 128 > \text{ThM} \\ 2, & \text{Se } 0 < m_i - 128 \leq \text{ThM} \\ 1, & \text{Se } -\text{ThM} < m_i - 128 \leq 0 \\ 0, & \text{Se } m_i - 128 \leq -\text{ThM} \end{cases} \quad (5)$$

A mesma lógica é aplicada para os 576 valores diferenciais, porém, para estes não é necessário realizar as subtrações por 128.

Com os valores quaternizados, a assinatura será adicionada ao vetor final de resultado com duas condições: o quadro não pode ser um quadro de transição, chamado de *black frame* pelos autores, e deve ser considerado quadro de cena. Portanto, a assinatura final é composta de valores no intervalo de [0,3]. O tamanho final de uma assinatura baseada em quadro de cena é $720 \times f$ onde f é o número de quadros de cena de um vídeo.

2.5.5 Assinatura baseada em padrões binários por região

O descriptor local apresentado por [Kim, Lee e Ro \(2014\)](#) propõe criar uma assinatura que seja robusta para ataques de rotação e espelhamento. Trata-se de um descriptor local que utiliza a dimensão espacial dos quadros do vídeo para a extração da assinatura.

O método divide o quadro em N regiões circulares (aneis) (Figura 12), e então divide os círculos em P sub-regiões, ou seja, sub-círculos ou fatias, como pode ser observado na Figura 11. O algoritmo então calcula a soma da luminância de cada sub-região ($l_{n,p}$) e a média da luminância de todas as regiões combinadas (R_μ). Dessa sub-região, o algoritmo gera dois padrões binários com a justificativa de preservar as informações da dimensão espacial do quadro e dessa forma manter a robustez da assinatura em relação aos ataques de rotação e espelhamento. O primeiro padrão binário (*RBP*, *Region Binary Pattern*) representa uma única região circular é calculado com a Fórmula 6, enquanto o segundo padrão binário representa a relação entre a primeira região e as regiões adjacentes, dado pela Fórmula 7. Após, esses dois padrões são agrupados e concatenados em um vetor, que é a assinatura resultante do algoritmo. A Figura 13 ilustra o fluxo geral do algoritmo. O tamanho final de uma assinatura baseada em padrões binários por região é $2 \times N \times M \times k$ onde k é o número de quadros de

um vídeo. Para esta implementação, foi utilizado $N = 3$ e $P = 6$.

$$b_{n,p} = \begin{cases} 1 & \text{se } l_{n,p} \geq R_\mu \\ 0 & \text{se } l_{n,p} < R_\mu \end{cases} \quad (6)$$

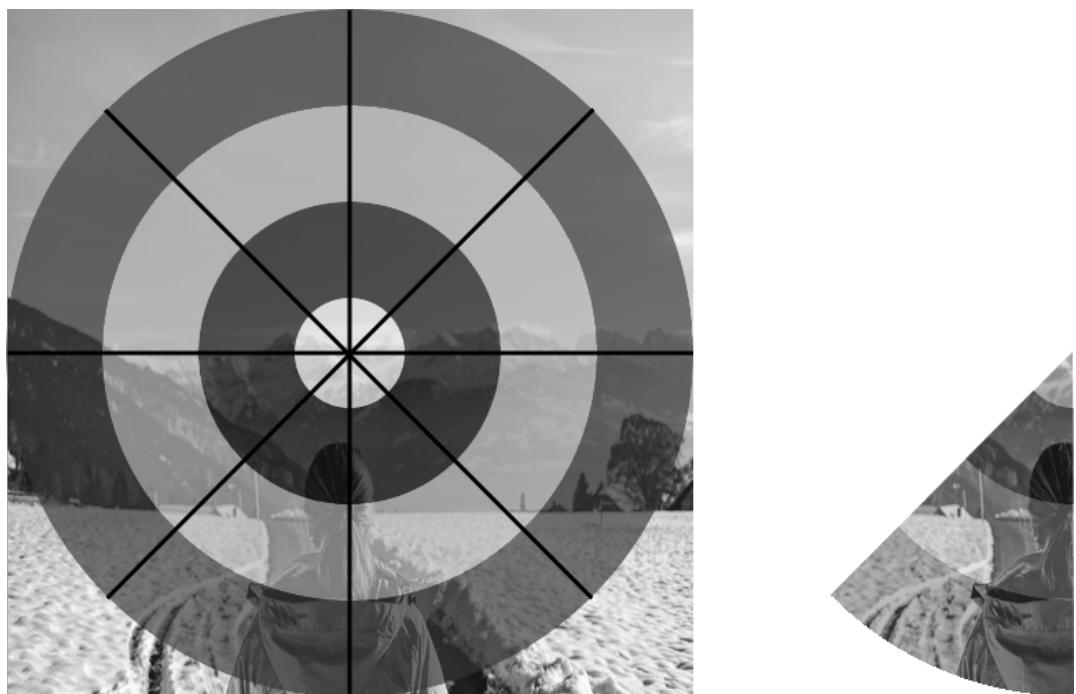
$$b_{n+N,p} = \begin{cases} 1 & \text{se } l_{n,p} \geq l_{n+1,p} \\ 0 & \text{se } l_{n,p} < l_{n+1,p} \end{cases} \quad (7)$$

Figura 11 – Divisão de um quadro em regiões circulares.



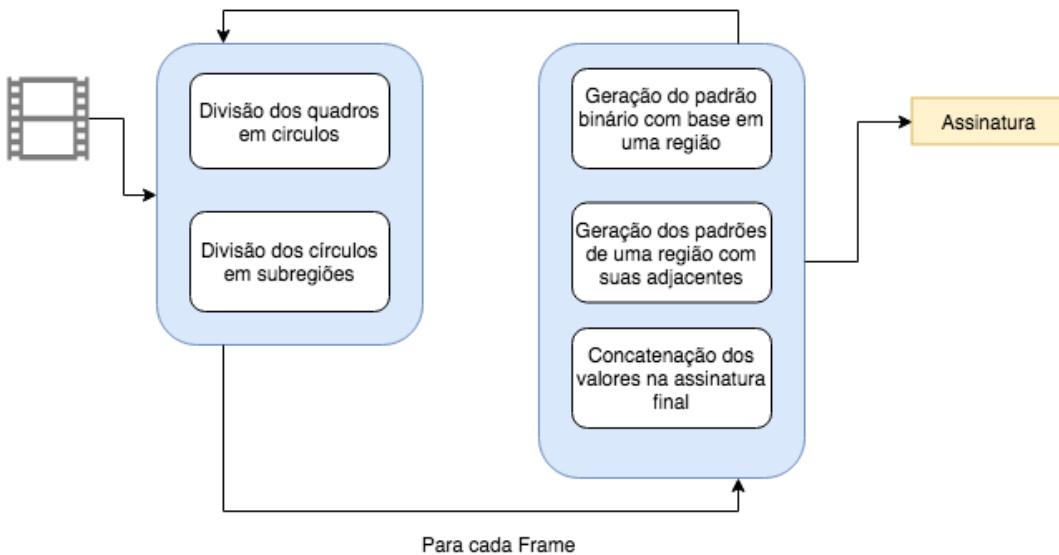
Fonte: Autoria própria.

Figura 12 – Divisão de um quadro em regiões fatias.



Fonte: Autoria própria.

Figura 13 – Diagrama do algoritmo baseado padrões binários por região.



Fonte: Autoria própria.

2.5.6 Assinatura baseada em wavelets

Esta abordagem foi escolhida por ter sido projetada especialmente para ser robusta a uma variedade de ataques fotométricos, como modificações em contraste, brilho, contaminação por ruído e desfoque, inserção de logos, bordas e mudança de formato do quadro, ou seja, utiliza características globais e locais para a composição da assinatura. Para a abordagem se tornar ainda mais robusta a esses ataques, Dutta, Saha e Chanda (2013) também realizam uma etapa de pré-processamento em que possíveis bordas dos quadros são removidas, então o ruído das imagens é removido através de um filtro gaussiano e finalmente o histograma de cada quadro é equalizado.

Após a etapa de pré-processamento, para serem utilizados como entrada para este algoritmo, os vídeos devem ser transformados para escala de cinza e ter suas intensidades normalizadas para o intervalo [0,1]. A assinatura proposta por Dutta, Saha e Chanda (2013) é baseada na transformada de Haar bidimensional, ela faz parte da família de Transformadas de Wavelets, que são ferramentas matemáticas para decompor funções de forma hierárquica (STOLLNITZ; DEROSE; SALESIN, 1995). No caso desta monografia essas funções são imagens.

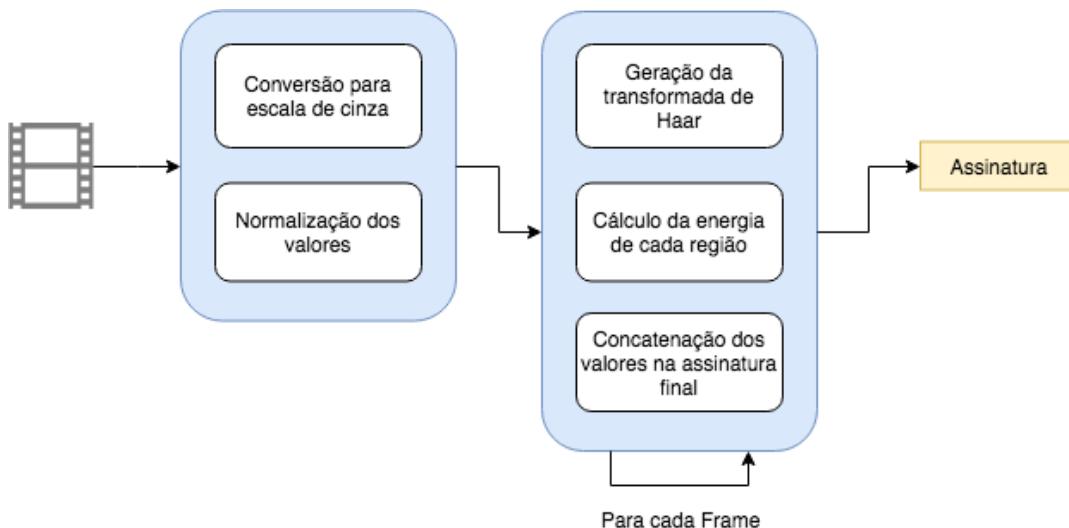
Para gerar a assinatura são aplicadas n iterações da transformada de Haar sobre a imagem de entrada I , para computar as bandas HH, LH, HL e LL . Então

são computadas as energias das bandas LH , HL , HH , conforme a Equação 8.

$$\frac{1}{MN} \sum_{x=1}^M \sum_{y=1}^N |I(x,y)| \quad (8)$$

Em seguida, computa-se somente a energia do último valor de LL da subimagem I . Finalmente, os valores de energia obtidos nos passos anteriores são concatenados em um vetor, resultando na assinatura do vídeo. O tamanho final de uma assinatura baseada em wavelet é $(3 \times n) + 1$ onde k é o número de quadros de um vídeo e n o número de iterações da transformada de Haar.

Figura 14 – Diagrama do algoritmo baseado em wavelets.

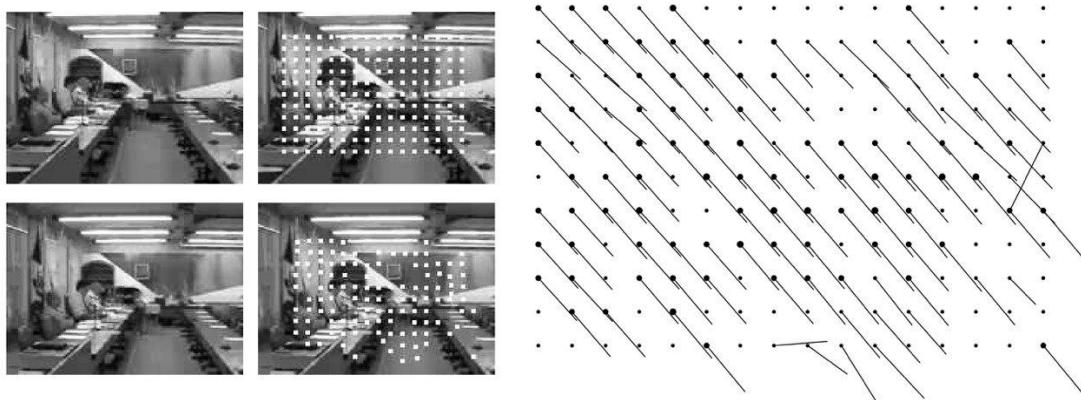


Fonte: Autoria própria.

2.5.7 Assinatura baseada no movimento da câmera

O algoritmo proposto por [Minetto, Leite e Stolfi \(2007\)](#) detecta o movimento da câmera em um vídeo em quatro níveis: *pan* (movimentos para esquerda ou direita, sem mover a base da câmera), *tilt* (movimentos para cima ou para baixo, sem mover a base da câmera), *zoom* (movimentos de aproximação ou de afastamento) e *roll* (movimento de rotação com câmera voltada para o mesmo ponto). É um algoritmo de assinatura local que é capaz de determinar o movimento da câmera em 95% das tomadas nos casos de testes realizados pelos autores, em um ambiente onde 98% das tomadas haviam algum tipo de movimento de câmera. A Figura 15 apresenta um exemplo de movimentação de câmera entre dois quadros.

Figura 15 – Movimento da câmera entre dois quadros consecutivos. Os pontos as direita representam os vetores de movimento. Neste caso há uma combinação de *pan* para a esquerda e *tilt* para cima.



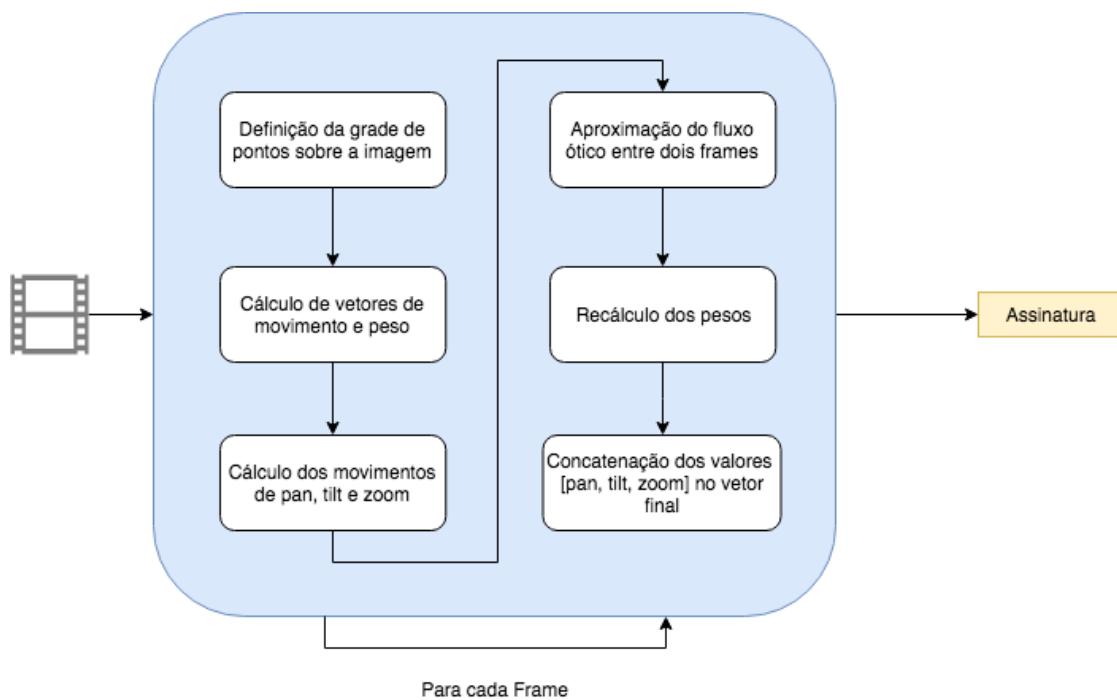
Fonte: ([MINETTO; LEITE; STOLFI, 2007](#)).

Seu funcionamento, segundo [Minetto, Leite e Stolfi \(2007\)](#) é dividido em duas etapas. A primeira é determinar uma região de fluxo óptico, ou seja, uma região no vídeo onde os movimentos de câmera serão monitorados. A segunda etapa é estimar os movimentos da câmera dentro dessa região de fluxo óptico.

O fluxo óptico de uma imagem I para uma imagem subsequente J é uma função f , que, para cada ponto u do domínio D da imagem, associa um vetor de velocidade $f(u)$, tal que $u + f(u)$ pertencem ao domínio D . O algoritmo assume que o fluxo está fixado em um conjunto de pontos na região de fluxo óptico e formam uma lista de vetores na qual cada vetor tem um peso correspondente, que representa a sua confiabilidade.

A assinatura para este algoritmo será composta pelo vetor de movimentos calculados pelo algoritmo. O diagrama completo do funcionamento do algoritmo pode ser visto na Figura 16.

Figura 16 – Diagrama do algoritmo de Camera Motion.

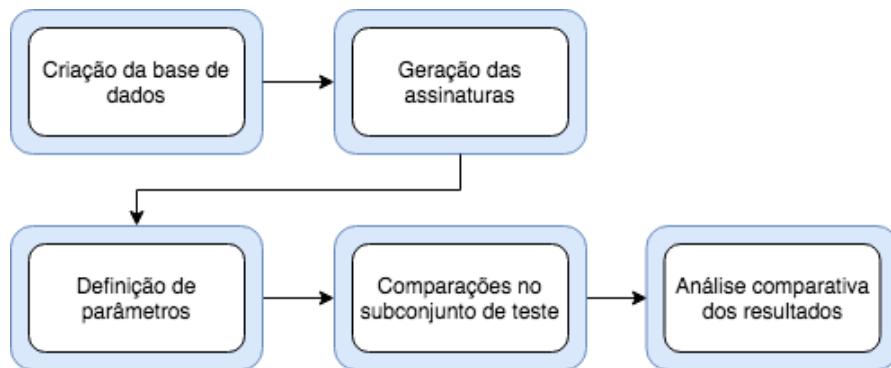


Fonte: Autoria própria.

3 Metodologia

A metodologia foi dividida nos passos mostrados na Figura 17. O primeiro passo realizado no trabalho foi a criação da base de vídeos, usando a base UCF50 - Action Recognition Data Set (REDDY; SHAH, 2013), sobre a qual foram aplicadas as 14 distorções descritas na Seção 2.3. Em seguida, foram geradas as assinaturas utilizando os algoritmos de diferença de quadro, gradientes, medida ordinal, quadros de cena, padrão binário por região, wavelets e movimento de câmeras, descritos no Capítulo 2. Foram então realizados procedimentos para definição de parâmetros de classificação para os métodos usando um subconjunto dos vídeos. Por fim, foi elaborada a análise comparativa dos resultados obtidos em um segundo subconjunto (conjunto de teste). Estes resultados serão discutidos em detalhes no Capítulo 4. As seções a seguir detalham cada um dos passos.

Figura 17 – Diagrama das etapas de desenvolvimento.

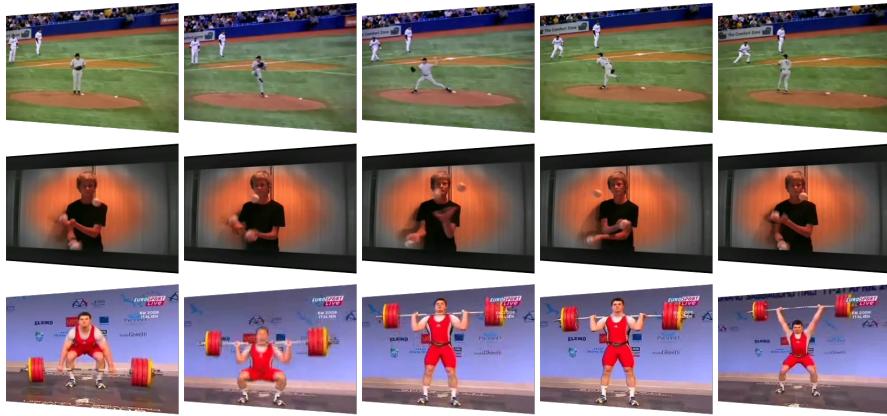


Fonte: Autoria própria.

3.1 Criação da Base de Vídeos

Para a realização dos experimentos desta monografia, foi escolhida a base de vídeos UCF50 - Action Recognition Data Set (REDDY; SHAH, 2013), que é comumente utilizada em projetos de reconhecimento de movimento humano. Ela foi escolhida pela quantidade de vídeos que contém, sua licença de livre utilização e a possibilidade de cortar uma única cena (que permite uma comparação simplificada baseada na capacidade dos descritores, já que não é preciso procurar uma cena em um vídeo longo).

Figura 18 – Exemplos de vídeos retirados da base. O primeiro mostra um jogo de beisebol, o segundo mostra uma criança fazendo malabarismo, o terceiro mostra uma competição de levantamento de pesos.



Fonte: Autoria própria.

A base contém 50 categorias diferentes que representam ações do quotidiano, como, por exemplo, ciclismo, natação, caminhada com o cachorro, TaiChi, etc. A base original é composta de 6.681 vídeos com duração média entre 3 e 9 segundos. A Figura 18 apresenta alguns exemplos de vídeos da base.

Dentro de cada categoria há pelo menos 4 vídeos pertencentes a uma mesma gravação, apresentando assim os mesmos personagens, fundo e ponto de vista. A fim de reduzir o número de vídeos que apresentam o mesmo conteúdo, foi selecionado apenas o primeiro vídeo de cada grupo. Após a seleção descrita no parágrafo anterior restaram 1.264 vídeos. Para cada vídeo selecionado foram aplicadas as distorções descritas na Seção 2.3, totalizando 18.960 vídeos, sendo 17.696 vídeos resultantes das distorções.

A base foi criada para propiciar a avaliação do desempenho das diferentes assinaturas revisadas neste trabalho quanto à robustez e à unicidade, e para garantir a reprodutibilidade do experimento, os parâmetros utilizados para a aplicação de cada uma das distorções são explicitados na Tabela 2. Para facilitar a discussão dos resultados, serão utilizadas as siglas definidas na primeira coluna da tabela quando houver referência às distorções.

3.2 Geração das Assinaturas

O passo final de preparação para os experimentos deste trabalho foi a criação das assinaturas utilizando os sete algoritmos descritos no Capítulo 2. As assinaturas

Tabela 2 – Parâmetros usados na aplicação das distorções.

Sigla	Distorção	Parâmetros
TEXT	Adição de texto e legendas	Texto: "Testing descriptor for scene"
WATERMARK	Adição de marca d'água	Texto: "Copyright" Tamanho da fonte: 20 Opacidade: 65%
BORDER	Adição de bordas	Tamanho da borda: 25 pixels
BIG	Redimensionamento da altura e largura dos vídeos	Fator de redimensionamento: 2 Mantém relação de aspecto: sim
SMALL	Redimensionamento da altura e largura dos vídeos	Fator de redimensionamento: 0.5 Mantém relação de aspecto: sim
CROP	Recorte de uma faixa ou região dos quadros	Largura máxima: 620 Altura Máxima: 338 Ponto de recorte X: 107 Ponto de recorte Y: 107
FLOP	Inversão/espelhamento	Sentido: horizontal
ROTATE	Rotação	10 graus
BLUR	Desfoque	Tipo: gaussiano Sigma: 3
COLOR	Inversão das cores	
JPEG	Alteração das compressão dos quadros do vídeo	Apenas 20% da qualidade
FAST1	Aceleração do vídeo 1	Aumento da velocidade em 50%
FAST2	Aceleração do vídeo 2	Aumento da velocidade em 25%
FRAME	Remoção de Frames	Remoção de 20%

foram geradas para todos os 18.960 vídeos, originais e distorcidos, totalizando 132.720¹ assinaturas. Para facilitar a realização dos experimentos, todas as assinaturas foram armazenadas em um banco de dados juntamente com: os dados do vídeo usados para computar a assinatura, se o vídeo é um vídeo original ou distorcido, o tipo da distorção, uma referência ao vídeo original, além do tipo de assinatura utilizada. A Tabela 3 apresenta todas as assinaturas utilizadas, na qual a primeira coluna apresenta a sigla de cada assinatura que será utilizada no Capítulo 4. Todas as assinaturas foram normalizadas para o intervalo [0,1].

¹18960 vídeos × 7 assinaturas

Tabela 3 – Assinaturas utilizadas para comparação.

Sigla	Assinatura
gradiente	Gradiente
framediff	FrameDiff
medidaordinal	Medida Ordinal
wavelets	Wavelets
rpb	RBP
sceneframe	Scene Frame

Fonte: Autoria própria.

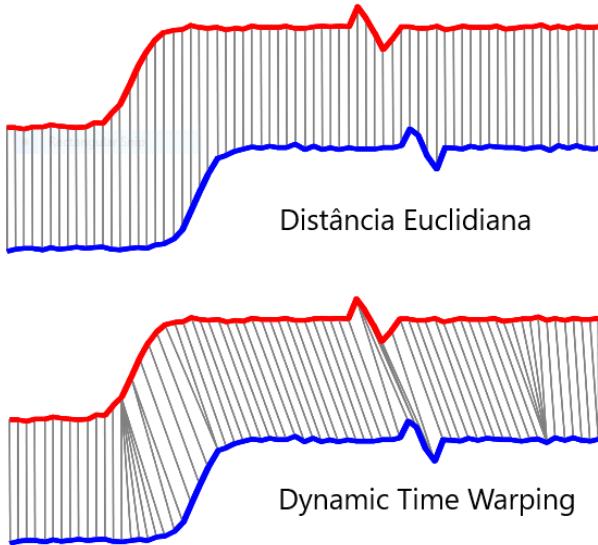
3.3 Comparação de Assinaturas

As assinaturas geradas na etapa anterior são compostas de vetores de características e cada um dos trabalhos descreve uma forma de compará-los, geralmente utilizando a distância Euclidiana ou a distância de Manhattan. Para simplificar a implementação e análise dos resultados, foi necessária a escolha de uma medida única capaz de lidar com as características de cada assinatura. Um dos fatores a ser considerado é que o tamanho das assinaturas é proporcional ao tamanho dos vídeos dos quais elas provém, além disso, este trabalho usa algumas distorções do tipo temporal que alteram a velocidade e o *framerate* de vídeos, alterando seu tamanho. Sendo assim, a medida utilizada precisa lidar com assinaturas de tamanhos e frequências diferentes.

Enquanto a distância Euclidiana pode ser útil para comparar as assinaturas de um único quadro de cada vídeo, o fato dela comparar um a um cada valor de duas assinaturas a torna suscetível a distorções temporais. Para resolver os problemas mencionados, foi escolhido o DTW (*Dynamic Time Warping*) como forma de comparação, uma técnica conhecida para alinhamento entre duas sequências temporais (MÜLLER, 2007). Originalmente, esta técnica foi utilizada para a comparação de diferentes padrões vocais em aplicações de reconhecimento de voz, mas o DTW já foi utilizado com sucesso para lidar com deformações temporais e velocidades diferentes em dados dependentes de tempo, explica Müller (2007). A Figura 19 mostra um comparativo entre o funcionamento da distância Euclidiana e do DTW, nota-se o pareamento um para um da distância Euclidiana (imagem de cima), e o pareamento n para um do DTW.

Para achar esse pareamento, o DTW compara todos os valores das duas sequências temporais utilizando uma medida de distância (normalmente a distância Manhattan ou Euclidiana) como parâmetro, como mostra a Figura 20. Em seguida, percorre-se a matriz da última célula até a primeira, sempre escolhendo a célula com o

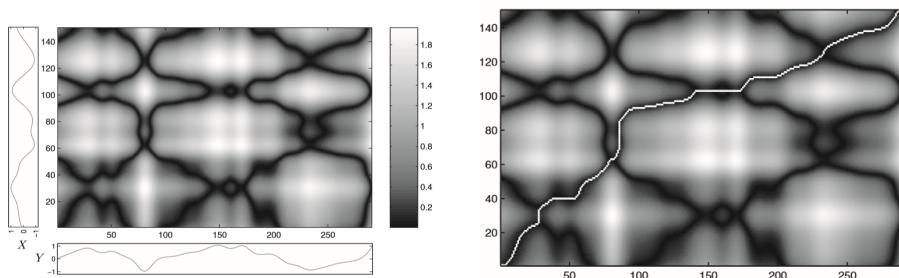
Figura 19 – Distância entre duas sequências temporais medida usando distância Euclidiana (na imagem de cima) e o DTW (na imagem de baixo).



Fonte: ([MÜLLER, 2007](#)).

menor custo como próximo passo. O caminho formado é o pareamento entre valores das duas sequências temporais. Sua saída é um valor numérico que é relativo à medida de distância escolhida como parâmetro. É necessário calcular um limiar para utilizar o resultado da comparação do algoritmo, sendo que os valores menores que o limiar representam que há similaridade entre as duas sequências comparadas ([MÜLLER, 2007](#)).

Figura 20 – a) Matriz de custo formada comparando duas sequências temporais. b) Caminho com menor custo.



a) Matriz de custo

b) Caminho mais barato

Fonte: ([MÜLLER, 2007](#)).

Em sua forma original, o DTW tem uma complexidade de $O(N^2)$, por esse motivo, foi utilizada uma variante do algoritmo chamada FastDTW, que computa

alinhamentos ótimos ou quase ótimos com complexidade de $O(N)$ ([SALVADOR; CHAN, 2007](#)).

3.4 Experimentos

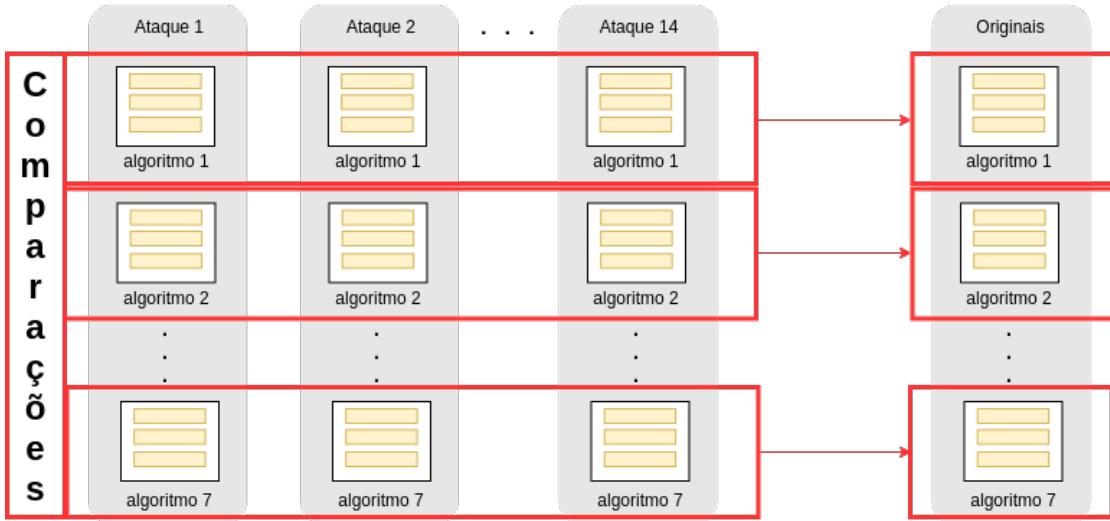
Os experimentos realizados neste trabalho estão divididos em duas etapas: a obtenção dos parâmetros de cada assinatura para a classificação de cópias (treinamento), e testes das assinaturas. Para a criação dos casos de treinamento/teste, são necessárias algumas definições. O conjunto $A_o = \{a_1, a_2, a_3, \dots, a_n\}$ contém todas as n assinaturas dos vídeos originais, o conjunto $A_d = \{a_1, a_2, a_3, \dots, a_m\}$ contém as m assinaturas dos vídeos distorcidos. Como há 1.264 vídeos originais na base de vídeos e 7 tipos de assinaturas sendo examinadas neste trabalho, o conjunto A_o tem tamanho $n = 8.848$. Como há 14 distorções sendo utilizadas neste trabalho, o conjunto A_d tem tamanho $m = 123.872$.

Para cada assinatura distorcida $a \in A_d$, existe uma assinatura correspondente $b \in A_o$ tal que a é uma cópia distorcida de b . Um caso de treinamento/teste c é composto de uma tupla (a, b) tal que $a \in A_o$ e $b \in A_d$. Um caso de treinamento c é marcado como “cópia” se b for uma cópia distorcida de a , senão é marcado como “não cópia”. Sendo assim, há um total de 1264×17696 (ou seja, 22.367.744) possíveis casos de treinamento/teste para cada tipo de assinatura, sendo apenas 17.696 marcados como “cópia”. Visando a criação de um conjunto de casos de treinamento balanceado, foram escolhidos aleatoriamente 50% dos 17.696 casos de “cópia” (8.792), e o mesmo número de casos “não cópia” para cada algoritmo, como pode ser observado na Figura [22](#). O mesmo critério foi utilizado para a criação dos conjuntos de teste: o resto dos 50% dos casos “cópia” foram selecionados, além de 8.792 casos “não cópia”. Para a criação de um caso de teste, é escolhida uma assinatura do grupo de assinaturas originais e uma assinatura do grupo de teste, sempre geradas pelo mesmo algoritmo como pode ser visto da Figura [21](#).

Para realizar uma classificação de um vídeo de teste como sendo cópia de um dos vídeos originais, as assinaturas dos dois são comparadas utilizando o DTW (definido da Seção [3.3](#)), que retorna como resultado uma medida de distância. Essa classificação utilizando o valor de distância só é possível após definido um limiar de corte para cada algoritmo, sendo que um vídeo de teste é considerado cópia de um vídeo original se a distância obtida entre os dois através do DTW estiver abaixo deste limiar. Logo, a primeira etapa dos experimentos é a definição destes limiares de forma empírica.

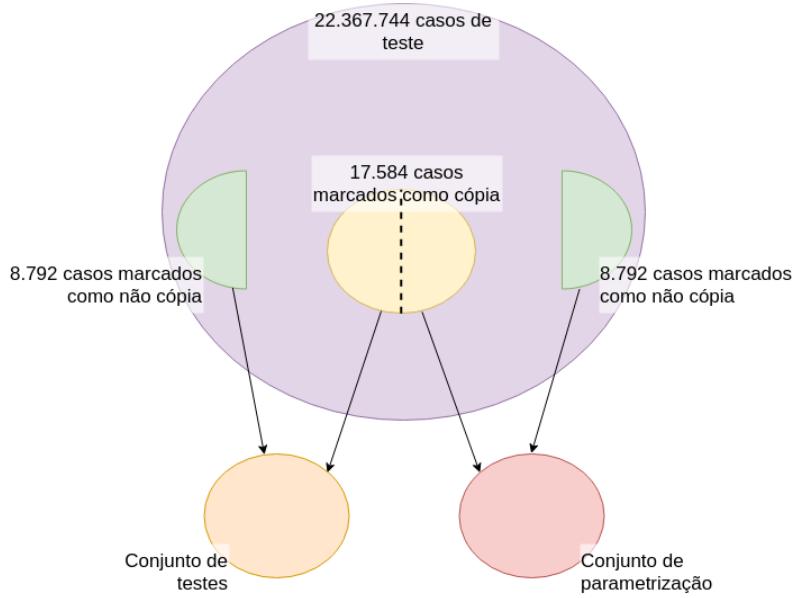
O conjunto de parametrização definido anteriormente é dividido em 5 subcon-

Figura 21 – Diagrama das comparações dos casos de teste.



Fonte: Autoria própria.

Figura 22 – Diagrama da divisão dos casos de teste para parametrização e testes.



Fonte: Autoria própria.

juntos organizados de forma aleatória (para evitar o enviesamento da parametrização e manter um número significativo de casos em cada subconjunto). Para cada subconjunto, é simulada a classificação de “cópia” e “não-cópia” com um intervalo de valores, a fim de encontrar um limiar ideal. Com o intuito de maximizar o número de verdadeiros-positivos e diminuir o número de falsos-negativos, é utilizada a medida

F-measure, escolhendo o limiar que maximiza a medida. Ao final da simulação para os 5 subconjuntos, o limiar final de cada assinatura é definido como a média dos limiares obtidos com a simulação para cada subconjunto.

Na fase de testes, são utilizados os limiares encontrados na fase de parametrização para classificar o conjunto de teste e avaliar cada tipo de assinatura quanto à sua robustez e unicidade quando usadas individualmente. Por fim, é realizada a combinação das assinaturas “gradiente”, “wavelets”, “medida ordinal”, “scene frame” e “rbp”, com a assinatura “camera motion” para avaliar se a combinação de assinaturas temporais e espaciais é mais eficaz na detecção de cópias.

4 Análise e Discussão dos Resultados

Este capítulo discute os resultados obtidos nos experimentos de detecção de cópias de vídeos através da aplicação das assinaturas digitais revisadas neste trabalho sobre os conjuntos de parametrização e teste detalhados na Seção 3.4.

Conforme detalhado no capítulo anterior, para a determinação do limiar de classificação de cada assinatura, são realizadas simulações de classificação utilizando um intervalo de limiares. Os casos de parametrização de cada assinatura são divididos empiricamente em 5 subconjuntos e, ao final, é utilizada a média dos limiares resultantes da simulação com cada subconjunto como limiar para a assinatura. Na Figura 23, foram plotados os valores de precisão, revocação e *F-measure* para cada assinatura ao longo das simulações com diferentes limiares. O ponto vermelho, indicando o maior valor de *F-measure*, define o limiar para cada assinatura.

Os subconjuntos de parametrização foram nomeados de T.1 a T.5 e o melhor limiar resultante da simulação com cada subconjunto está disposto na Tabela 4, além do limiar final de cada algoritmo.

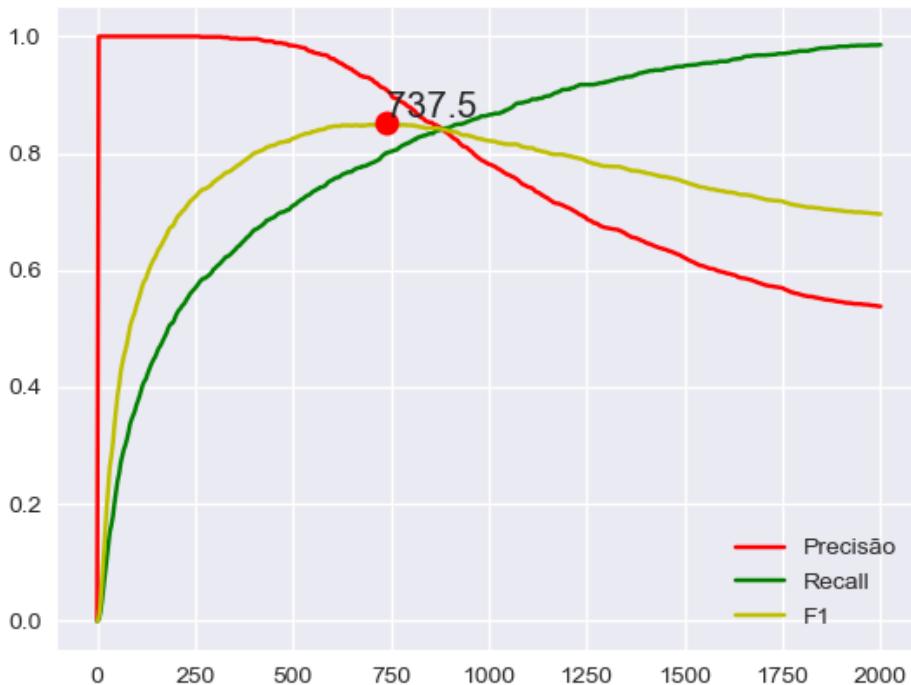
Tabela 4 – Limiares obtidos nas simulações com cada subconjunto de parametrização.

Assinatura	T.1	T.2	T.3	T.4	T.5	Limiar Final
Gradiente	761.52	637.27	681.36	705.41	629.25	682.96
FrameDiff	6.63	6.48	6.33	6.78	6.93	6.63
Medida Ordinal	657.31	729.45	661.32	725.45	749.49	704.60
Wavelets	597.19	593.18	625.25	601.20	625.25	608.41
RBP	5501.00	6793.58	6553.10	6147.29	4569.13	5912.82
Scene Frame	399.49	391.95	399.49	391.95	399.49	396.48
Camera Motion	111.82	123.84	107.01	139.47	128.65	122.16

Fonte: Autoria própria.

Na sequência, serão discutidos os resultados obtidos nos testes utilizando os limiares definidos na etapa anterior para cada tipo de assinatura. As assinaturas serão analisadas quanto a sua robustez e unicidade por tipo de distorção aplicada (fotométrica, geométrica ou temporal). Por fim, será avaliado se a junção de um algoritmo temporal (camera motion) e um algoritmo espacial é mais eficaz em detectar cópias de vídeo que a utilização de um tipo de assinatura de forma isolada.

Figura 23 – Exemplo de simulação de classificação para um tipo de assinatura. O eixo x é composto dos limiares testados para a assinatura. O ponto vermelho indica o valor máximo de *F-measure*, ponto em que o limiar apresenta o melhor resultado. F1 representa o *F-measure*.



Fonte: Autoria própria.

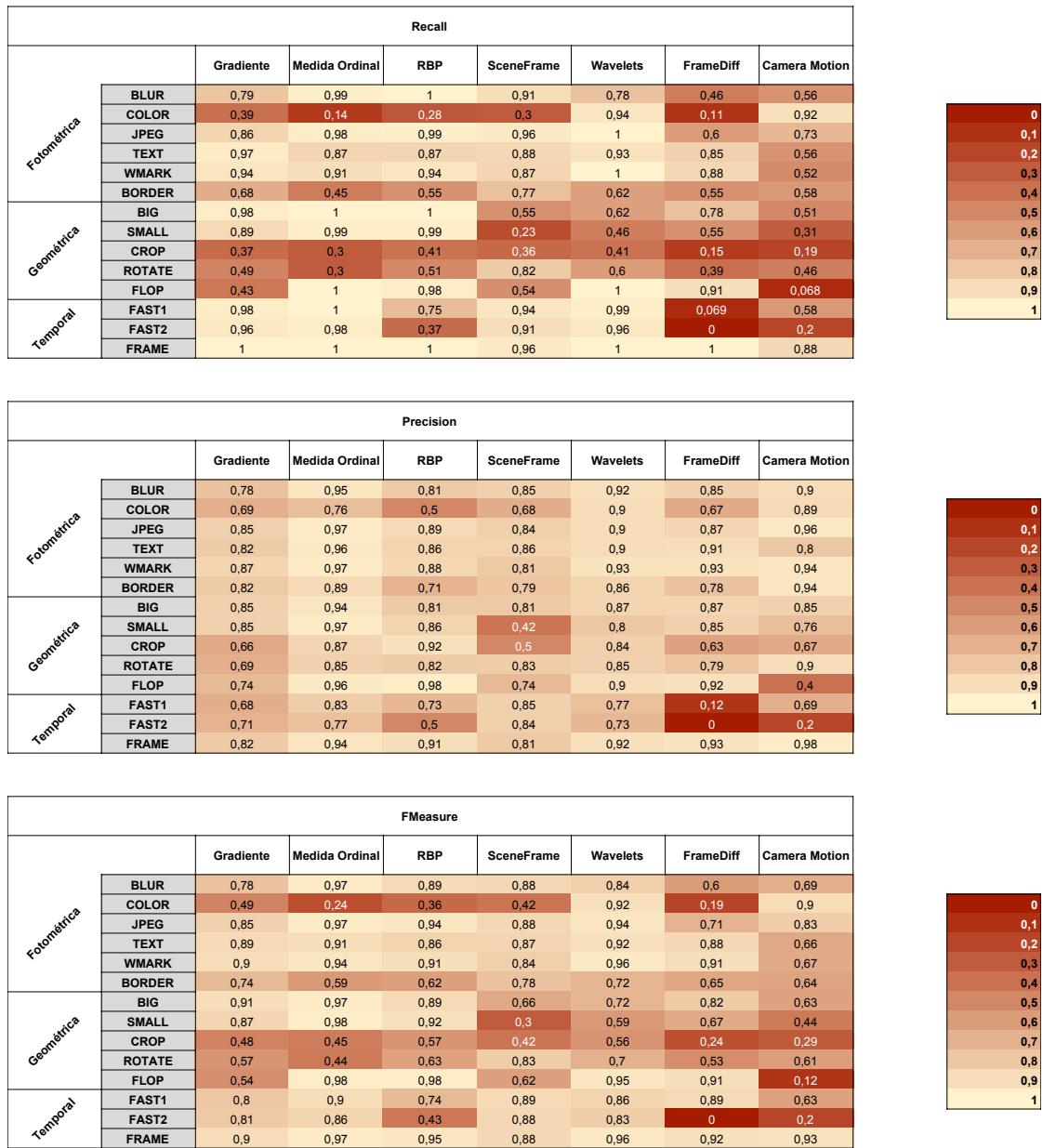
4.1 Robustez

Segundo [Hua, Chen e Zhang \(2004\)](#), robustez é a capacidade de uma assinatura ser tolerante a ruído, o que quer dizer que dois vídeos com o mesmo conteúdo devem ter assinaturas idênticas ou muito similares, mesmo que eles tenham passado por algum tipo de distorção.

Ao tentar determinar a robustez de um algoritmo, leva-se em consideração apenas os casos de teste em que de fato um par de vídeos tem o mesmo conteúdo, ou seja, o vídeo de teste é uma cópia do vídeo original. A partir daí, a pergunta levantada é “quantos destes casos de cópia a assinatura consegue classificar corretamente?”, ou em termos de recuperação de informações “quantos elementos relevantes foram selecionados?”. Esta é exatamente a definição da medida de revocação ([TING, 2011](#)),

que foi utilizada para este teste. As medidas de revocação obtidas para cada assinatura e para diferentes tipos de distorção podem ser vistas na Figura 24.

Figura 24 – Mapa de calor de revocação, precisão e fmeasure de cada tipo de assinatura com cada tipo de distorção. Revocação é definida na Seção 4.1, precisão é definida na Seção 4.2 e F-measure é definido na Seção 4.3.



Fonte: Autoria própria.

As assinaturas temporais (Camera Motion e FrameDiff) obtiveram os piores resultados na análise da robustez em geral, sendo que elas são especialmente afetadas

pelas distorções temporais de aceleração fast1 e fast2. É possível notar também que a medida que os vídeos foram mais acelerados, as assinaturas temporais tiveram resultados piores. A assinatura temporal FrameDiff classificou corretamente aproximadamente 6% das cópias no teste com a distorção fast1 e 0% com a distorção fast2, mostrando que esse tipo de assinatura é extremamente sensível à perda de informação causada pela diminuição do número de quadros de um vídeo. A Camera Motion foi mais sensível a distorção flop, que inverte o vídeo horizontalmente. Isso faz sentido pois esta distorção inverte a translação e rotação da câmera. A última distorção temporal, remoções de frames (frame), foi a que causou menos problemas para as assinaturas em geral (3 delas conseguiram classificar corretamente em 100% dos casos e todas as outras têm revocação maior que 80%), pois a quantidade de quadros retirados de um vídeo é pequena se comparado às outras distorções temporais, além disso este tipo de distorção é pouco perceptível por humanos.

Outras distorções que tiveram pouco efeito sobre a classificação são as que adicionam textos de forma uniforme nos vídeos: text e wmark. O único caso em que estas distorções afetaram significativamente a classificação é com a utilização da assinatura Camera Motion. Como ela se baseia no rastreamento do movimento de regiões do vídeo entre os quadros, o posicionamento estático do texto realizado pela distorção o impede de acompanhar os movimentos precisamente.

Entre as distorções fotométricas, a inversão de cores (color) foi a mais efetiva em enganar a detecção de cópias, apenas as assinaturas Camera Motion e Wavelets conseguiram bons resultados, com revocação acima dos 90%. Todas as outras assinaturas obteram valores de revocação abaixo de 40%. Os resultados favoráveis a essas duas assinaturas se devem ao fato destas não utilizarem valores de luminância para descrever um vídeo, enquanto as outras o fazem. Já a distorção blur afetou negativamente todas as assinaturas que usam bordas ou pequenas regiões de interesse para descrever um vídeo, pois enfraquece os detalhes de um quadro. A distorção jpeg teve resultados semelhantes por também causar uma perda de informação dentro de cada quadro do vídeo.

A revocação sozinha não pode ser utilizada para a avaliação de uma assinatura, pois bastaria marcar todos os pares de vídeos como sendo cópias para atingir um valor de 100% de revocação. Vê-se necessária uma análise complementar à da robustez, sobre a capacidade discriminante de uma assinatura: a unicidade. A seção a seguir discute sua definição, apresenta uma forma de medi-la e avalia cada assinatura quanto a essa métrica.

4.2 Unicidade

Segundo [Hua, Chen e Zhang \(2004\)](#), unicidade é a capacidade discriminativa de uma assinatura de vídeo, o que significa que vídeos com conteúdos diferentes devem ter assinaturas diferentes. Esta característica é essencial para a utilidade de uma assinatura, pois uma técnica que gera muitos casos de falsos positivos perde sua utilidade como ferramenta de automação de detecção de cópias. Para avaliar esta característica, é utilizada a medida de precisão, que mede a quantidade de casos que são relevantes (verdadeiros positivos), dentre todos os pares de vídeos que são classificados como cópias (verdadeiros positivos e falsos positivos) ([TING, 2011](#)).

A Figura [24](#) apresenta os valores de precisão resultantes dos testes com todas as assinaturas. Em termos gerais, a assinatura Medida Ordinal tem os melhores resultados neste teste, não apresentando nenhum valor de precisão abaixo de 75% e tendo valores acima de 90% para 8 das 14 distorções. Assim como no análise da robustez, as assinaturas temporais obtiveram os piores resultados e o fizeram exatamente nos mesmos casos: fast1, fast2 e flop.

Para entender o motivo destes resultados para as assinaturas temporais e verificar se estes não se devem a uma escolha ruim de limiares, foram plotados histogramas dos resultados das comparações de cada tipo de assinatura. A Figura [25](#) apresenta quatro histogramas de valores resultantes da comparação entre pares de vídeos de teste e vídeos originais para a assinatura Camera Motion. Os casos que são cópias estão marcados de verde, enquanto que os não-cópias estão marcados de azul. Quanto mais afastados os dois grupos, maior o poder de discriminação da assinatura para aquela distorção. Nota-se que para a distorção color, a assinatura é capaz de separar melhor os grupos de cópia e não cópia, enquanto que para as outras distorções a assinatura sobrepõe os dois grupos de forma que não é possível separá-los apenas utilizando um limiar de corte. O mesmo ocorre com a assinatura FrameDiff, que não é capaz de classificar assinaturas que sofreram as distorções fast1 e fast2, mas consegue classificar quase perfeitamente casos com as distorções flop e frame (Figura [26](#)).

4.3 Peso da escolha do limiar sobre a detecção de cópias

Para chegar a uma conclusão sobre o tipo de assinatura mais adequado para a detecção de cópias em relação a sua robustez e unicidade, é preciso analisar ambas a precisão e a revocação de forma conjunta. Ao comparar as duas medidas (Figura [24](#)), nota-se uma relação entre estas: os pontos escuros (piores resultados) estão localizados nas mesmas regiões, mas no geral, todas as assinaturas tiveram resultados de precisão

Figura 25 – Histogramas de resultados de comparações para a assinatura Camera Motion.

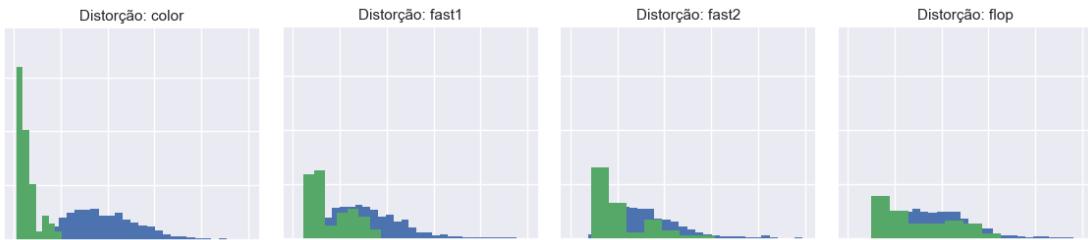


Figura 26 – Histogramas de resultados de comparações para a assinatura FrameDiff.



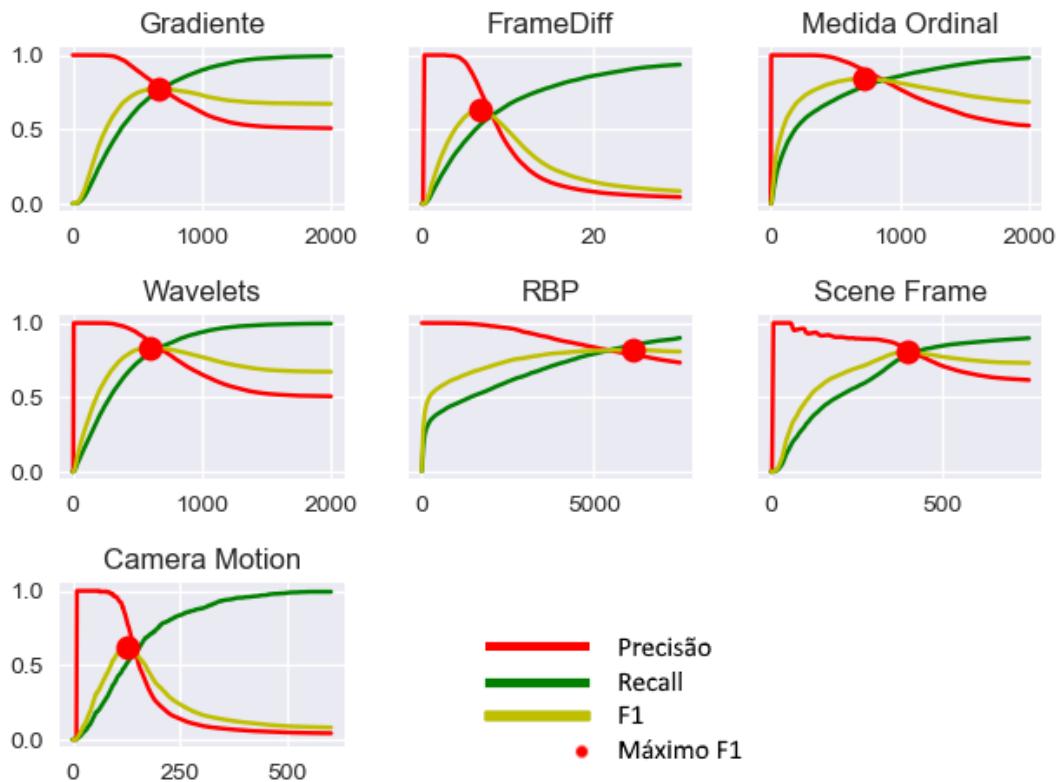
Fonte: Autoria própria.

melhores que os de revocação. Além disso nota-se também que a relação entre revocação e precisão não é linear. A Figura 24 usa a medida *F-measure* para combinar precisão e revocação, ela é definida por $F\text{measure} = 2 \times \frac{p \times r}{p + r}$, onde p é o valor de precisão e r é o valor de revocação.

A Figura 27 mostra a variação das medidas de precisão e revocação à medida que o limiar de corte cresce. Há claramente uma diminuição da capacidade discriminativa de cada assinatura à medida com que o limiar de corte se afasta do zero, além disso, cada assinatura apresenta uma proporção diferente para essa diminuição. As assinaturas Medida Ordinal, Wavelets, Scene Frame e RBP são as que mantêm a precisão em níveis mais altos.

A seguir, será discutido se a aplicação de dois tipos de assinatura de forma conjunta é mais eficiente na detecção de cópias do que a aplicação de apenas uma de forma isolada. Será definido o método de combinação destas assinaturas, além da escolha de um limiar de classificação para cada uma destas combinações.

Figura 27 – Simulação de classificação para cada um dos tipos de assinatura. A medida que o limiar se afasta do zero, o valor de precisão diminui e o de revocação aumenta. Cada tipo de assinatura tem uma proporção diferente para essa variação.



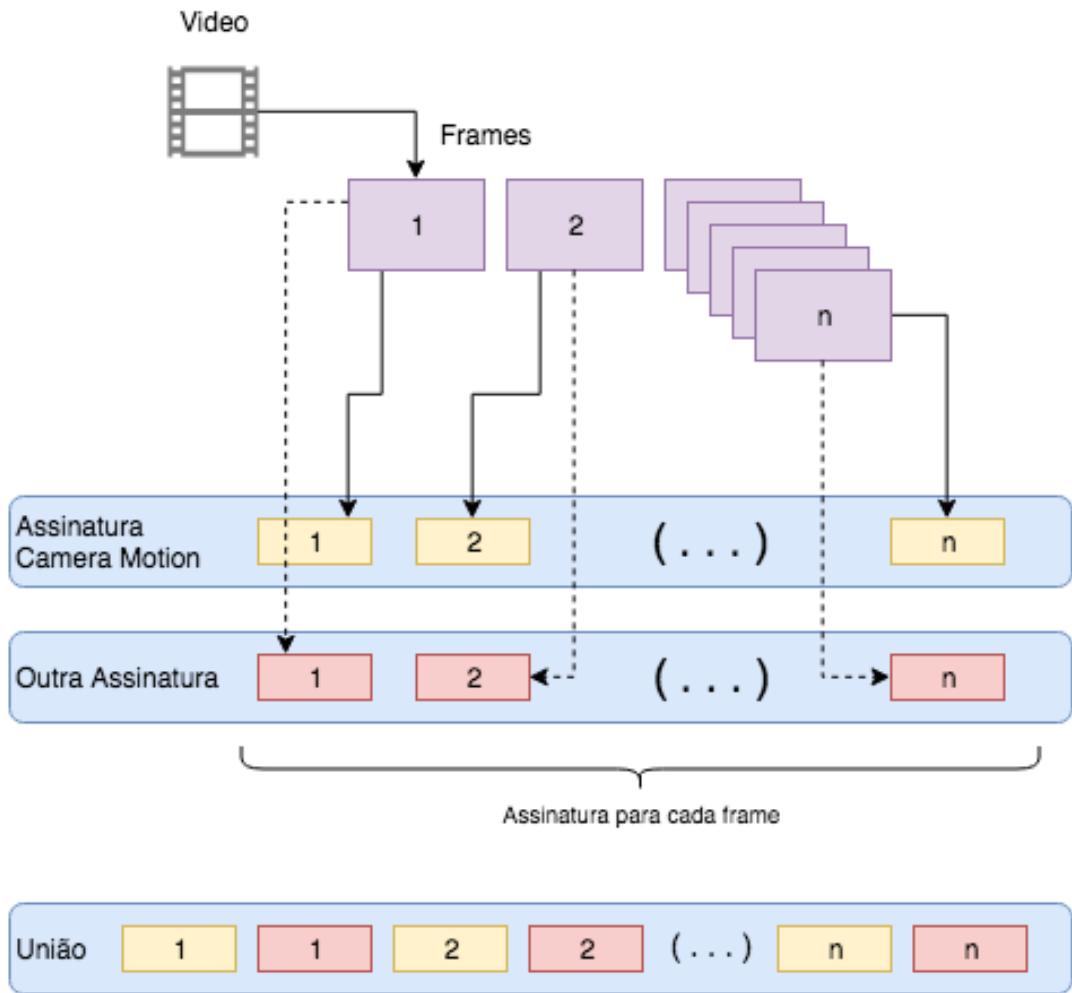
Fonte: Autoria própria.

4.4 Combinação de Assinaturas

Assim como as distorções, as assinaturas podem ser classificadas quanto às características de um vídeo que elas utilizam em sua formação. Das assinaturas usadas neste trabalho, duas são temporais (Camera Motion e FrameDiff), enquanto todas as outras são espaciais. Nesta etapa dos experimentos, estes dois tipos de assinatura foram combinados a fim de definir se as métricas de robustez e unicidade são melhoradas em relação às assinaturas utilizadas separadamente. Embora várias combinações de assinaturas sejam possíveis, para este trabalho, a assinatura Camera Motion foi combinada com todas as outras.

O primeiro passo é a definição do método de combinação destas assinaturas. Como visto no Capítulo 3, uma assinatura de vídeo é um vetor de características que

Figura 28 – Exemplo de combinação de assinaturas.



Fonte: Autoria própria.

é formado pela concatenação das características de cada quadro de um vídeo. Sendo assim, podemos separar a assinatura em sub-vetores onde cada um destes representa um quadro no vídeo. Para combinar dois tipos diferentes de assinaturas que descrevem o mesmo vídeo, os sub-vetores dos quadros correspondentes são concatenados em pares e então adicionados em sequência, como mostra a Figura 28.

Como cada tipo de assinatura gera valores muito diferentes ao ser usado como entrada para o DTW (veja Tabela 4 para exemplos), esses valores passam por uma etapa de normalização utilizando o método Min-Max ([SHALABI; SHAABAN; KASASBEH, 2006](#)), que realiza uma transformação linear para um intervalo de valores pré-definido

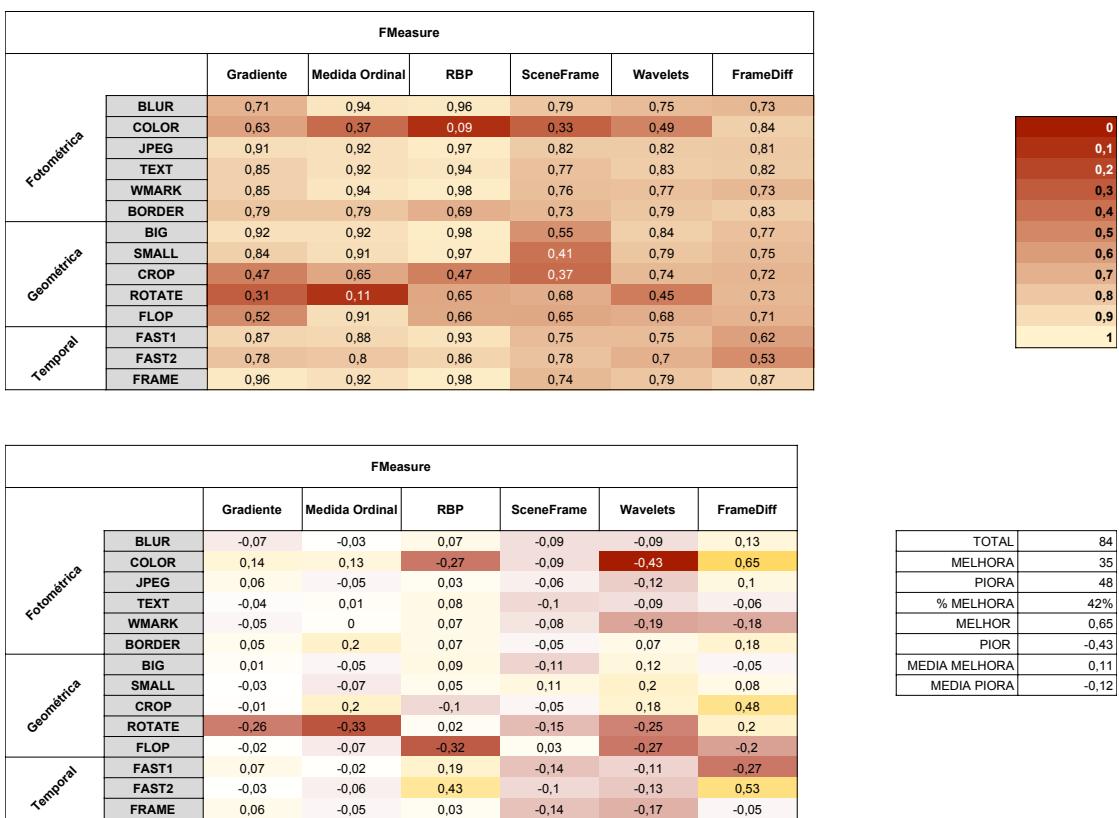
(que é $[0, 1]$), como mostra a Equação 9.

$$A' = \left(\frac{A - \min(A)}{\max(A) - \min(A)} \right) * (D - C) + C \quad (9)$$

Onde A' é o vetor A normalizado, e C e D são os valores mínimos e máximos definidos previamente. No caso desta monografia, foram utilizados $C = 0$ e $D = 1$ para normalizar os vetores no intervalo $[0, 1]$.

Para comparar os resultados com aqueles dos algoritmos separados, os valores da medida *F-measure* para cada teste de classificação estão dispostos na Figura 29. No geral, a combinação de duas assinaturas não teve efeitos positivos na classificação. Com ela, todas as combinações passaram a ser sensíveis a distorções temporais e poucos casos apresentaram melhorias. Nos casos em que a mudança foi positiva, o ganho foi de menos de 5%.

Figura 29 – Histograma com valores de fmeasure para cada tipo de assinatura. Em cima, resultados da combinação da assinatura Camera Motion com as demais, embaixo, a diferença entre os resultados com e sem a combinação das assinaturas.



Fonte: Autoria própria.

5 Conclusão

Este trabalho apresenta uma revisão da literatura moderna sobre detecção de cópias de vídeo baseada em conteúdo. Este tema é cada vez mais importante devido a quantidade de conteúdo audio-visual sendo enviado para a internet a cada minuto e a facilidade de propagação ilegal de conteúdo protegido por direitos autorais. Para isso, foram realizados experimentos com um conjunto importante de algoritmos para assinatura a fim de descobrir seus pontos fracos e verificar se a combinação de abordagens de tipos diferentes pode melhorar a detecção de cópia baseada em conteúdo.

Nos resultados, a utilização de assinaturas temporais obteve o pior resultado em todos os testes, sendo especialmente sensíveis a distorções temporais e fotométricas. Enquanto isso, assinaturas que utilizam características globais de um vídeo como luminância obtiveram desempenho razoável, mas foram altamente sensíveis a distorções fotométricas.

De modo geral, a assinatura Wavelets obteve os melhores resultados na classificação, conseguindo classificar corretamente a maioria dos casos de distorção fotométrica e temporal, mas sendo mais afetada por distorções de tipo geométrico. Mesmo com esta sensibilidade a ataques geométricos, no caso em que obteve os piores resultados (distorção crop), seu desempenho foi acima da média geral e ficou em segundo lugar na medida F-measure (que combina as métricas de robustez e unicidade).

No teste de combinação de assinaturas, notou-se uma clara diminuição na robustez e unicidade de modo geral em comparação à aplicação de assinaturas de modo isolado. Ao invés de uma combinação das características que tornam cada tipo de assinatura robusto, a combinação somou a sensibilidade da assinatura temporal Camera Motion a todas as outras, piorando as métricas de robustez e unicidade em praticamente todos os casos. No entanto, isso não comprova que a combinação de tipos assinatura não pode ser benéfica à detecção de cópias.

5.1 Trabalhos Futuros

Os resultados deste trabalho mostram como diferentes tipos assinatura são afetados por diferentes tipos de ataque, revelando os pontos mais críticos de cada um. Em uma continuação deste trabalho, poderia ser estudada a utilização de métodos de detecção de padrões para identificar tipos de distorção em vídeos e utilizar o melhor tipo de assinatura para a comparação de similaridade.

Isso implicaria no uso de múltiplas assinaturas para cada vídeo, tornando ainda mais importante a velocidade de busca e o tamanho de cada assinatura, dois itens que não foram considerados durante a realização deste trabalho.

Referências

- ANDRADE, F. d. S. P. de et al. Combinação de descritores locais e globais para recuperação de imagens e vídeos por conteúdo. Campinas, SP, 2012.
- AUDIBLE MAGIC CORPORATION. **Compliance Automation for Media Sharing Platforms**: How it works. 2018. Acessado em 14/03/2018. Disponível em: <<http://www.audiblemagic.com/compliance-service/#how-it-works>>.
- CHEN, Z.; SUN, S.-K. A zernike moment phase-based descriptor for local image representation and matching. **IEEE Transactions on Image Processing**, IEEE, v. 19, n. 1, p. 205–219, 2010.
- COOK, R. An efficient, robust video fingerprinting system. In: IEEE. **Multimedia and Expo (ICME), 2011 IEEE International Conference on**. [S.I.], 2011. p. 1–6.
- COSKUN, B.; SANKUR, B.; MEMON, N. Spatio-temporal transform based video hashing. **IEEE Transactions on Multimedia**, IEEE, v. 8, n. 6, p. 1190–1208, 2006.
- DUTTA, D.; SAHA, S. K.; CHANDA, B. An attack invariant scheme for content-based video copy detection. **Signal, Image and Video Processing**, v. 7, n. 4, p. 665–677, 2013. ISSN 1863-1711. Disponível em: <<http://dx.doi.org/10.1007/s11760-013-0482-x>>.
- HAMPAPUR, A.; HYUN, K.; BOLLE, R. M. Comparison of sequence matching techniques for video copy detection. In: INTERNATIONAL SOCIETY FOR OPTICS AND PHOTONICS. **Electronic Imaging 2002**. [S.I.], 2001. p. 194–201.
- HU, W. et al. A survey on visual content-based video indexing and retrieval. **IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)**, IEEE, v. 41, n. 6, p. 797–819, 2011.
- HUA, X.-S.; CHEN, X.; ZHANG, H.-J. Robust video signature based on ordinal measure. In: IEEE. **Image Processing, 2004. ICIP'04. 2004 International Conference on**. [S.I.], 2004. v. 1, p. 685–688.
- INDYK, P.; IYENGAR, G.; SHIVAKUMAR, N. **Finding pirated video sequences on the internet**. [S.I.]: Stanford University, 1999.
- JIANG, M. et al. Pku-idm@ trecvid 2011 cbcd: content-based copy detection with cascade of multimodal features and temporal pyramid matching. In: CITESEER. **TRECVID Workshop: NIST**. [S.I.], 2011.
- JOLY, A.; BUISSON, O.; FRELICOT, C. Content-based copy retrieval using distortion-based probabilistic similarity search. **IEEE Transactions on Multimedia**, IEEE, v. 9, n. 2, p. 293–306, 2007.

- KIM, S.; LEE, S. H.; RO, Y. M. Rotation and flipping robust region binary patterns for video copy detection. **Journal of Visual Communication and Image Representation**, Elsevier, v. 25, n. 2, p. 373–383, 2014.
- KING, D. **Content ID turns three**. 2010. Acessado em 28/04/2018. Disponível em: <<https://youtube.googleblog.com/2010/12/content-id-turns-three.html>>.
- LAW-TO, J. et al. Robust voting algorithm based on labels of behavior for video copy detection. In: ACM. **Proceedings of the 14th ACM international conference on Multimedia**. [S.I.], 2006. p. 835–844.
- LAW-TO, J. et al. Video copy detection: a comparative study. In: ACM. **Proceedings of the 6th ACM international conference on Image and video retrieval**. [S.I.], 2007. p. 371–378.
- LEE, S.; YOO, C. D. Robust video fingerprinting based on affine covariant regions. In: IEEE. **Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on**. [S.I.], 2008. p. 1237–1240.
- LELLA, A. **comScore Releases January 2014 U.S. Online Video Rankings**. [S.I.], 2018. Acessado em 14/03/2018. Disponível em: <<http://www.comscore.com/Insights/Press-Releases/2014/2/comScore-Releases-January-2014-US-Online-Video-Rankings>>.
- LIENHART, R.; PFEIFFER, S.; EFFELSBERG, W. Video abstracting. **Communications of the ACM**, ACM, v. 40, n. 12, p. 54–62, 1997.
- MAO, J. et al. A method for video authenticity based on the fingerprint of scene frame. **Neurocomputing**, Elsevier, v. 173, p. 2022–2032, 2016.
- MINETTO, R.; LEITE, N. J.; STOLFI, J. Reliable detection of camera motion based on weighted optical flow fitting. In: **VISAPP (2)**. [S.I.: s.n.], 2007. p. 435–440.
- MÜLLER, M. Dynamic time warping. **Information retrieval for music and motion**, Springer, p. 69–84, 2007.
- NAINI, R.; RANE, S.; RAMALINGAM, S. A vanishing point-based global descriptor for manhattan scenes. In: IEEE. **Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on**. [S.I.], 2014. p. 4349–4353.
- RACHMADI, R. F.; UCHIMURA, K.; KOUTAKI, G. Video classification using compacted dataset based on selected keyframe. In: **2016 IEEE Region 10 Conference (TENCON)**. [S.I.: s.n.], 2016. p. 873–878.
- RADHAKRISHNAN, R.; BAUER, C. Content-based video signatures based on projections of difference images. In: IEEE. **Multimedia Signal Processing, 2007. MMSP 2007. IEEE 9th Workshop on**. [S.I.], 2007. p. 341–344.
- REDDY, K. K.; SHAH, M. Recognizing 50 human action categories of web videos. **Machine Vision and Applications**, Springer, v. 24, n. 5, p. 971–981, 2013.

- SALVADOR, S.; CHAN, P. Toward accurate dynamic time warping in linear time and space. **Intelligent Data Analysis**, IOS Press, v. 11, n. 5, p. 561–580, 2007.
- SHALABI, L. A.; SHAABAN, Z.; KASASBEH, B. Data mining: A preprocessing engine. **Journal of Computer Science**, v. 2, n. 9, p. 735–739, 2006.
- STOLLNITZ, E. J.; DEROSE, A. D.; SALESIN, D. H. Wavelets for computer graphics: a primer. 1. **IEEE Computer Graphics and Applications**, IEEE, v. 15, n. 3, p. 76–84, 1995.
- TING, K. M. Precision and recall. In: **Encyclopedia of machine learning**. [S.I.]: Springer, 2011. p. 781–781.