

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

JORDY JACKSON ANTUNES DA ROCHA,  
JULIA ULSON TRETTEL,  
RODRIGO MACHADO

**ANÁLISE COMPARATIVA DE ALGORITMOS PARA  
ASSINATURA DIGITAL DE VÍDEOS**

TRABALHO DE CONCLUSÃO DE CURSO

CURITIBA  
2018

JORDY JACKSON ANTUNES DA ROCHA,  
JULIA ULSON TRETTEL,  
RODRIGO MACHADO

## **ANÁLISE COMPARATIVA DE ALGORITMOS PARA ASSINATURA DIGITAL DE 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. Rodrigo Minetto  
Universidade Tecnológica Federal do Paraná

Coorientador: Prof. Dr. Ricardo Dutra da Silva  
Universidade Tecnológica Federal do Paraná

CURITIBA  
2018

## **Agradecimientos**

TODO AGRADECIMENTOS

## RESUMO

Rocha, J. J. A, Tretel, J. U, Machado, R.. Análise Comparativa de Algoritmos para Assinatura Digital de Vídeos. 2018. 17 f. Trabalho de Conclusão de Curso – Bacharelado em Sistemas de Informação, Universidade Tecnológica Federal do Paraná. Curitiba, 2018.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Curabitur luctus ante nec sem pretium, vel tincidunt arcu imperdiet. Interdum et malesuada fames ac ante ipsum primis in faucibus. Donec auctor, nunc sed elementum mattis, urna ex commodo metus, nec mattis metus felis at turpis. Pellentesque tincidunt metus eros, in dapibus libero imperdiet in. Sed sit amet ipsum venenatis leo bibendum mollis eget non erat. Ut ut mauris a nisl euismod semper. Sed pharetra, dui eu tempus vulputate, neque nulla varius quam, eget consequat diam ante euismod dolor. Interdum et malesuada fames ac ante ipsum primis in faucibus. Quisque vitae tincidunt nisi, vel pulvinar nunc. Vestibulum quam neque, bibendum quis iaculis at, finibus ut neque. Maecenas tincidunt eget arcu vel aliquet. Proin non iaculis ante. Etiam blandit quam at arcu consectetur, vitae volutpat elit blandit. Nam in ornare nisi, quis egestas diam. In vestibulum mauris neque, vitae iaculis mauris tincidunt vitae.

**Palavras-chave:** Assinatura Digital de Vídeo, Descritor

## LISTA DE FIGURAS

Figura 1 – Estrutura de um vídeo. Referência: ??(??)	3
Figura 2 – Exemplo de marca d'água em uma imagem.	6
Figura 3 – Exemplo com a divisão em blocos, cálculo das intensidades médias e ordem atribuída a cada valor. Referência: (??)	13
Figura 4 – Sequência de transformações feitas pelo algoritmo utilizando um quadro do filme Mulher Maravilha	16
Figura 5 – Divisão da imagem para cálculo dos elementos diferenciais. Referência: ??(??)	17

## LISTA DE TABELAS

Tabela 1 – Lista de repositórios utilizados . . . . .	10
Tabela 2 – Palavras-chave utilizadas na revisão sistemática. . . . .	10
Tabela 3 – Categorização dos artigos. . . . .	10

## Sumário

<b>1 – Definições</b>	<b>2</b>
1.1 Definição de Vídeo	2
1.1.1 Quadro	2
1.2 Definição de Assinatura	3
1.3 Definição de Quadro de Cena	3
1.4 Tipos de descritores de vídeo	4
1.4.1 Descritores Locais	4
<b>2 – Estado da Arte</b>	<b>6</b>
2.1 Detecção de Cópias Baseada em Conteúdo	6
2.2 Recuperação de Vídeo Baseada em Conteúdo	7
2.3 Recuperação de Imagens Baseada em Conteúdo	7
<b>3 – Trabalhos relacionados</b>	<b>8</b>
<b>4 – Metodologia</b>	<b>9</b>
4.1 Revisão da Literatura	9
4.1.1 Definições	9
4.2 Definição e Obtenção da Base de Vídeos	10
4.3 Definição e implementação dos algoritmos	11
4.4 Definição e Implementação do método de comparação	11
4.5 Validação das implementações	11
4.6 Assinatura de vídeo baseada na medida ordinal	13
4.7 Assinatura de vídeo baseada em gradientes	13
4.8 Assinatura de vídeo baseada na diferença entre quadros	14
4.9 Assinatura de vídeo baseada em wavelets	14
4.9.1 Transformada de Haar	15
4.9.2 Assinatura baseada em wavelets	15
4.9.3 Assinatura baseada em distribuição espacial de gradientes	15
4.10 Assinatura baseada em quadros de cena	16
4.10.1 A extração de assinatura	17
4.10.2 Diminuição do espaço de memória utilizado	17

A Detecção de Cópias Baseada em Conteúdo é um método utilizado para a proteção de propriedade intelectual de mídias digitais. Ele consiste em extrair uma assinatura da mídia original e de uma mídia de teste, e então compará-las para identificar que se trata do mesmo conteúdo ou não. Esta abordagem assume que a mídia contenha alguma informação única que possa ser utilizada para identificar uma cópia ??(??).

Suponha que uma plataforma online de publicação de vídeos (como YouTube ou Vimeo) receba vários pedidos vindos de criadores de conteúdo digital (como estúdios de cinema, cineastas independentes ou “vloggers”) solicitando a retirada de vídeos da plataforma alegando o infringingimento de direitos autorais. Segundo estes criadores, seus filmes estão sendo reproduzidos parcial ou integralmente na plataforma.

Enquanto o número de pedidos é baixo, não há grandes problemas em definir se realmente se trata de uma cópia: Basta assistir ao vídeo original e à suposta cópia, e então comparar os conteúdos. A medida que o número de pedidos cresce ou que os a extensão dos vídeos seja cada vez mais longa (filmes tendem a ter pelo menos duas horas de duração), torna-se claramente inviável a realização desse processo de forma manual.

Para realizar esta tarefa de forma automática, podem ser utilizados métodos de Detecção de Cópias Baseada em Conteúdo para a criação uma base de dados de assinaturas, para então ser realizada a busca e verificação de cópias.

Um dos serviços comerciais mais utilizados para este fim é o *Content ID* da Audible Magic. O método utilizado cria uma assinatura baseando-se no áudio de uma mídia digital (podendo então ser utilizado tanto para vídeos, quanto para músicas), e disponibiliza um banco de dados global de assinaturas de conteúdos protegidos por propriedade intelectual. Este serviço é usado por grandes corporações como Facebook, SoundCloud e Vimeo ??(??).

No entanto, o uso do áudio para a criação de assinaturas pode não ser o melhor atributo utilizado na criação de uma assinatura, pois vídeos (principalmente longas-metragens) tendem a ter versões dubladas na língua de cada país onde são distribuídos.

Para combater este problema, foram propostos inúmeros métodos que utilizam informações contidas nos quadros dos vídeos para a geração das assinaturas. Este trabalho propõe-se a fazer uma revisão da literatura de métodos para Detecção de Cópias Baseada em Conteúdo, selecionar seis candidatos que possuam abordagens distintas, compará-los e avaliar quais atributos do conteúdo digital devem ser escolhidos a fim de otimizar a acurácia na detecção de cópias.



## 1 Definições

Nas próximas seções estão definidos conceitos utilizados neste trabalho, como quadro (*frame*), vídeo e tomada, além da definição de assinatura de vídeo e algumas características consideradas importantes na geração destas.

### 1.1 Definição de Vídeo

#### 1.1.1 Quadro

Um quadro (ou *frame*)  $I$  é uma matriz de altura  $h$  e largura  $w$ . Cada ponto  $I[x,y]$  representa uma intensidade de pixel [??(??)]. Além disso, um quadro possui um determinado tempo, que representa o instante em que aparece em um vídeo. Portanto, seguindo a definição de ??(??), um vídeo é representado por uma sequência de quadros, todos dispostos através da amostra temporal.

Um vídeo pode ser segmentado em cenas, que por sua vez possuem diferentes tomadas, compostas por diferentes quadros em sequência, conforme Figura 1. As tomadas de um vídeo, como pode ser observado na Figura 1, são definidas por um ou mais quadros capturados continuamente, representando uma ação ininterrupta no tempo e no espaço, de acordo com ??(??). Um conjunto de tomadas semelhantes, por sua vez, é intitulado de *cena*.

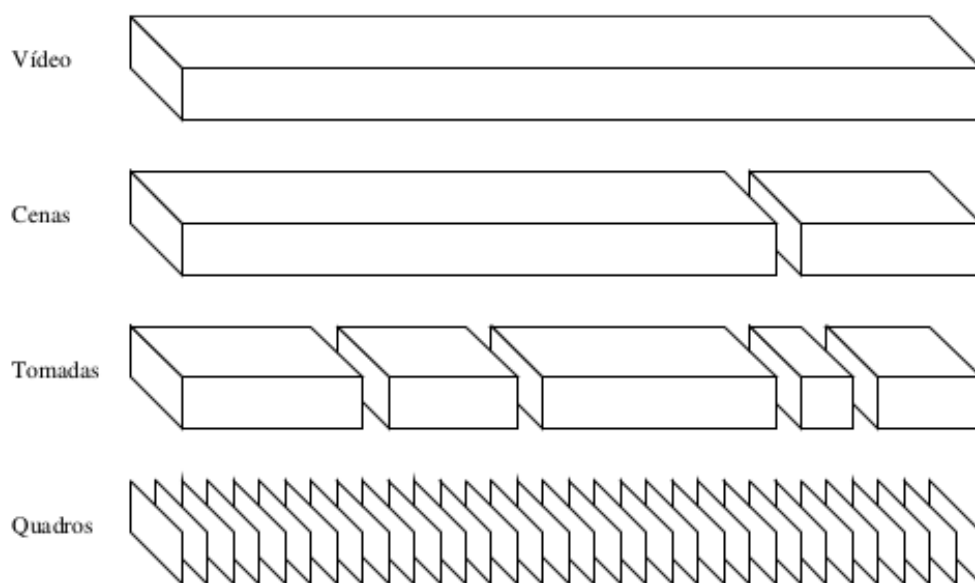


Figura 1 – Estrutura de um vídeo. Referência: ??(??)

## 1.2 Definição de Assinatura

Uma assinatura de vídeo é definida como um vetor de características que representa um vídeo e o diferencia de outros ??(??). Outro termo utilizado para fazer essa referência é o de descritor, uma vez que a assinatura irá descrever um determinado vídeo. As características do vídeo podem ser obtidas de diferentes formas, as quais serão apresentadas na seção 4.5.

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 ??(??), uma assinatura é considerada robusta caso o descritor gerado para um vídeo modificado seja similar ao descritor do vídeo original. A singularidade é a capacidade do algoritmo de 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 avaliados apenas a robustez e a singularidade das assinaturas.

## 1.3 Definição de Quadro de Cena

De acordo com ??(??), um quadro de cena pode ser representado por quadros que sigam duas características: deve ser uma imagem, como definido em 1.1.1; e os

mesmos objetos, juntamente com um *background* altamente similar, devem pertencer a uma mesma cena.

[ARRUMAR: Procurar outras definições de quadros de cena]

#### 1.4 Tipos de descritores de vídeo

[fonte livro] Em visão computacional pode-se descrever imagens utilizando algoritmos de identificação de pontos de interesse, sendo conhecidos como descritores locais. Há também descritores globais, que consideram características da imagem como um todo, e não apenas pontos de interesse.

Cada abordagem tem vantagens e desvantagens distintas frente a cada tipo de variação ou ataque às imagens, porém, têm o mesmo objetivo, gerar uma descrição da imagem, normalmente tendo por saída um vetor, que pode ser considerado a assinatura da imagem.

Idealmente, um descritor de imagens, seja global ou local, deve gerar assinaturas com robustez mediante os principais tipos de variação fotométricas, como borrados, iluminação, ruído e compressão JPEG e também em relação a variações geométricas, como rotação, escala (zoom), translação e ponto de referência (viewpoint).

##### 1.4.1 Descritores Locais

Descritores locais utilizam características locais de cada imagem, comumente denominadas pontos e regiões de interesse, para gerar sua saída. Uma imagem pode ter vários pontos ou regiões de interesse. Os pontos de interesse podem ser determinados a partir de regiões que possuem uma acentuada variação na orientação do gradiente dos objetos presentes em uma imagem, 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 de cada ponto de interesse. Aproveitando o exemplo anterior, uma região de interesse seria a região ao redor da intersecção do enquadramento de uma porta ou passagem.

[inserir imagem demonstrando ponto e região de interesse]

Existem algoritmos de descrição local que são baseados em pontos de interesse, algoritmos que são baseados em regiões de interesse e algoritmos que são baseados tanto em pontos quanto em regiões de interesse. O que determina a escolha do algoritmo é o tipo de ataque ou distorção que se deseja combater, detectar ou evitar com o descritor ( Krystian Mikolajczyk and Cordelia Schmid ).

Descritores locais normalmente são mais robustos contra variações fotométricas (borrados, iluminação, cores, ruído e compressão JPEG) e na maioria dos casos mais custosos computacionalmente que os descritores globais [por que?]. Um descritor local consiste normalmente de três etapas: detecção das características, descrição das características e combinação das características, de acordo com (Zen Chen and Shu-Kuo Sun 2010).

Na primeira etapa do processo são determinados os pontos e regiões de interesse, como por exemplo o detector de pontos de interesse SIFT, que utiliza uma janela circular na imagem. Esta abordagem com o SIFT, entretanto, não é robusta contra variações geométricas. Na sequência, é definida uma janela de tamanho fixo e é utilizado um descritor das características de região para essa janela. Com todas as regiões de interesse descritas, a última etapa é definir uma função para combinar as diferentes regiões de interesse e assim gerar o vetor de descrição da imagem. (Zen Chen and Shu-Kuo Sun 2010)

## 2 Estado da Arte

### 2.1 Detecção de Cópias Baseada em Conteúdo

Segundo ??(?), o método de detecção de cópias baseada em conteúdo, do inglês *Content Based Copy Detection* (CBCD), tem se tornado uma alternativa à marca d'água, conforme ilustra a Figura 2, para identificar e proteger vídeos e seus direitos autorais através da criação de uma assinatura digital para o conteúdo. Apesar de o método CBCD vir sendo largamente utilizado em diversas aplicações, a detecção de cópias é um grande desafio e o trabalho de ??(?) expõe determinadas limitações do método.

O conteúdo a ser procurado pode sofrer uma intensa redução em sua qualidade entre o vídeo original e a cópia, podendo até mesmo sofrer alterações, como por exemplo, distorções e transformações. É uma tarefa difícil buscar cópias através de mecanismos baseados em pesquisa por quadros sem uma ferramenta que consiga unir corretamente a duração de cada trecho a ser pesquisado com o vídeo original. Por último, é necessária uma representação compacta e eficiente das assinaturas para que se construa uma ferramenta de busca e detecção de cópias para grandes sistemas.



Figura 2 – Exemplo de marca d'água em uma imagem.

## 2.2 Recuperação de Vídeo Baseada em Conteúdo

De acordo com ??(?), recuperação de vídeo baseado em conteúdo, do inglês *Content Based Video Retrieval* (CBVR) é o procedimento para gerar, pesquisar e analisar assinaturas digitais. Essa área de estudo produz os algoritmos que geram os identificadores digitais, estuda a similaridade entre assinaturas e também busca trechos em vídeos. Ainda segundo ??(?), CBVR tem foco em procurar e encontrar vídeos em uma mesma categoria, como por exemplo, jogos de futebol ou vídeos sobre balões.

## 2.3 Recuperação de Imagens Baseada em Conteúdo

Conforme ??(?), recuperação de imagens baseada em conteúdo, em inglês *Content Based Image Retrieval* (CBIR), é um sistema que auxilia na recuperação e extração de imagens de acordo com o conteúdo da imagem. Para representar uma imagem, o sistema CBIR se baseia em elementos visuais como cores, texturas e formas ??(?). Há diversas aplicações que se beneficiam desta tecnologia, como: previsão do tempo, serviços de informações geográficas, design de interiores, galerias de arte, etc ??(?).

### 3 Trabalhos relacionados

Existem diversos trabalhos correlatos que discutem assuntos como assinatura digital de vídeos, formas de obter as descrições, analisar e comparar assinaturas, métodos para realizar buscas em vídeos, descritores globais e locais, etc.

Em 1999, quando a visualização de vídeos na internet ainda estava em sua infância, ??(??) já buscava métodos para encontrar vídeos pirateados na web. Este propôs um algoritmo para geração de assinatura temporal baseada nos limites das cenas de um vídeo. Embora seja boa para encontrar filmes inteiros, esta técnica não é apropriada para vídeos curtos, que dominam as redes sociais de vídeos (aproximadamente 4 minutos) ??(??).

Desde então, várias técnicas têm sido usadas para a geração de assinaturas. ??(??) propôs o conceito de funções de *hash* como uma ferramenta para identificação de vídeo, criando um algoritmo espaço-temporal baseado no diferencial da luminância entre regiões de quadros. Outras abordagens incluem o uso de descritores globais que utilizam a distribuição da intensidade de movimento e cor ??(??), além de medidas ordinais ??(??), que se provaram robustas para variadas resoluções, mudanças de iluminação e formatos de vídeos.

Abordagens que utilizam características locais também foram extensamente pesquisadas, como em ??(??), cujo algoritmo apresentado busca ser eficiente para buscas em grandes bases de dados, tanto em velocidade quanto a qualidade. Há também a pesquisa de ??(??), que usa o algoritmo de Harris para encontrar pontos de interesse no vídeo e criar uma assinatura compacta.

??(??) fez um estudo comparativo entre descritores globais e locais e mostrou como unir os dois tipos de descritores através de algoritmos genéticos. ??(??) também fez vários 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.

??(??) discorre 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.

## 4 Metodologia

A metodologia do projeto consiste das seguintes etapas:

1. Revisão da literatura;
2. Definição e obtenção da base de vídeos
3. Definição dos algoritmos;
4. Implementação dos algoritmos;
5. Definição do método de comparação;
6. Implementação do método de comparação;
7. Validação das implementações;
8. Definição dos experimentos;
9. Experimentos;
  - a) Compilação de uma nova base de vídeos;
  - b) Geração dos vídeos com distorções;
  - c) Geração da assinatura para todos os vídeos;
  - d) Aplicação do método de comparação;
10. Análise dos resultados;

### 4.1 Revisão da Literatura

Foi realizada uma revisão da literatura pertinente à pesquisa. Este método de revisão da literatura foi escolhido pois define formas de identificar, interpretar e avaliar pesquisas disponíveis coerentes com o tema.

Além de artigos recentes relacionados à área, as referências de ??(??) também foram utilizadas na revisão e passaram pelo processo de filtragem, uma vez que este trabalho baseia-se no de ??(??).

#### 4.1.1 Definições

O primeiro passo da revisão é a definição das perguntas que devem ser respondidas, das palavras-chave a serem utilizadas e dos critérios de exclusão de artigos. O intuito da revisão é definido a seguir:

- encontrar algoritmos para tratar diferentes problemas relacionados à descrição de vídeos e compreender seu funcionamento;
- encontrar métodos de *matching* de descritores;
- encontrar e compilar uma base de vídeos para realização dos testes;



Além das referências de ??(??), foram escolhidos repositórios em português e em inglês para a busca das palavras-chave. Os repositórios estão definidos na tabela 1 enquanto as palavras-chave na tabela 2.

Língua	Repositórios	Endereço
Inglês	IEEE	< <a href="http://ieeexplore.ieee.org/Xplore/home.jsp">http://ieeexplore.ieee.org/Xplore/home.jsp</a> >
	ScienceDirect	< <a href="http://www.sciencedirect.com/">http://www.sciencedirect.com/</a> >
	ACM	< <a href="http://dl.acm.org/">http://dl.acm.org/</a> >
	Google Scholar	< <a href="https://scholar.google.com.br/">https://scholar.google.com.br/</a> >
Português	Periódico CAPES	< <a href="http://www.periodicos-capes.gov.br.ez48.periodicos.capes.gov.br">http://www.periodicos-capes.gov.br.ez48.periodicos.capes.gov.br</a> >
	SciELO	< <a href="http://www.scielo.org">http://www.scielo.org</a> >

Tabela 1 – Lista de repositórios utilizados.

Português	Inglês
Detecção de cópia baseada em conteúdo	Content-Based Video Copy Detection
Descritor de vídeo	Video fingerprint
Detecção de cópia de vídeo	video copy detection

Tabela 2 – Palavras-chave utilizadas na revisão sistemática.

Para auxiliar na indexação dos artigos achados e na sua utilização como base para este trabalho, os artigos foram divididos em categorias, como mostra a tabela 3.

	Categoria	Descrição
1	Descritores Globais	Algoritmos que produzem descritores baseados em características globais de um vídeo
2	Descritores Locais	Algoritmos que produzem descritores baseados em características locais de um vídeo
3	Comparação de Descritores	Métodos para comparação de descritores

Tabela 3 – Categorização dos artigos.

## 4.2 Definição e Obtenção da Base de Vídeos

Foi escolhida a base UCF50 - Action Recognition Data Set ??(??) e embora seja utilizada para reconhecimento de movimento humano baseado em ações, como por exemplo ciclismo, natação, caminhada com o cachorro, TaiChi, etc., os motivos para sua escolha foram a quantidade de vídeos e sua licença de livre utilização.

Do total de 6.681 itens, foram selecionados 1.264, pois os vídeos estavam fragmentados em diversas cenas, ou seja, apenas uma cena de cada vídeo foi utilizada. Cada cena tem uma duração média de 3 a 9 segundos.

Para cada vídeo selecionado foram aplicadas 14 distorções, totalizando 18.960 itens, sendo destes, 17.696 vídeos resultantes das distorções.

[escrever das 14 distorções]

### 4.3 Definição e implementação dos algoritmos

Como este trabalho visa continuar a pesquisa em descritores de vídeo baseando-se em *??(??)*, os primeiros algoritmos estudados e implementados são os três algoritmos já implementados por ele, descritos nas Seções 4.6, 4.7 e 4.8.

Embora *??(??)* tenha optado por comparar apenas algoritmos de descrição global, devido a sua simplicidade de implementação, neste trabalho serão utilizados tanto descritores locais quanto globais.

POR QUE DOS OUTROS TRÊS ALGORITMOS? [sao recentes e locais]

Foi escolhida a linguagem Python (versão 2.7.12) para a implementação dos algoritmos por ser multiplataforma e por sua simplicidade na integração com bibliotecas de alta-performance implementadas em linguagens de baixo nível (C). Isto foi importante pois serão utilizadas as bibliotecas *OpenCV* (versão 3.0.0) e *NumPy* (versão 1.12.1) para manipulação e operação de imagens.

Para a compilação da base de vídeos foi utilizada a biblioteca *MagickImage* (versão 7.0.7-8-Q16-x64 - Windows), que inclui os programas *convert* e *ffmpeg*, utilizados para conversão de formatos, aplicação de distorções e transformações de vídeo e imagem respectivamente.

### 4.4 Definição e Implementação do método de comparação

pegar base de assinaturas distorcidas, e para cada distorcao, calcular a distancia para todas as assinaturas originais

pegar as duas menores distancias para cada distorcao de cada vídeo

se a dif do primeiro colocado (primera distancia) for diferente o suficiente para o segundo colocado , entao podemos assumir (podemos?) que encontramos a assinatura do vídeo original

EMBASAR ESSAS PARADAS LOUCAS

### 4.5 Validação das implementações

Antes de realizar-se qualquer experimento, é necessário que os algoritmos implementados passem por uma etapa de validação que consiste em comparar seus resultados com os dos artigos de base. Para isso serão revisados os artigos dos quais os algoritmos foram retirados, além de códigos disponibilizados através destes trabalhos.

A validação ocorrerá em duas etapas:

1. teste de software;

2. comparação das saídas dos algoritmos implementados para o trabalho e as implementações originais, dada a mesma entrada.

Estudaremos em detalhes os algoritmos das seguintes seções.

#### 4.6 Assinatura de vídeo baseada na medida ordinal

O algoritmo proposto por (??) baseia-se na intensidade dos *pixels* de cada quadro para compor a assinatura. O que se propõe é que, primeiramente, a taxa de amostragem, ou seja, a taxa de quadros por segundo (FPS, do inglês *frames per second*) do vídeo de entrada seja padronizada, para que a assinatura gerada fique mais compacta e seja tolerante a diferentes formatos de compressão, por exemplo. Além disso, o vídeo é convertido para escala de cinza.

Após esse pré-processamento, para cada quadro é realizada a divisão em  $M \times N$  blocos, como pode ser observado na Figura 3, bem como o cálculo da intensidade média para cada um dos blocos. Esses valores são então colocados em ordem crescente e representam cada elemento que compõe a assinatura.



Figura 3 – Exemplo com a divisão em blocos, cálculo das intensidades médias e ordem atribuída a cada valor. Referência: (??)

#### 4.7 Assinatura de vídeo baseada em gradientes

Este algoritmo, proposto por (??), 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. Outro procedimento utilizado é o redimensionamento dos quadros, tornando o método robusto independente da mudança de resolução do vídeo. Em seguida, o gradiente dos *pixels* de cada quadro é calculado da seguinte forma. Encontram-se os gradientes  $\mathbb{G}x$  e  $\mathbb{G}y$  através da equação 1.

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

O quadro é então dividido em  $M \times N$  blocos, para quais é determinado o valor do centroide dos gradientes, criando assim um vetor com  $(M \times N)$  elementos.

Para isso, é necessário encontrar a 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 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:

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

#### 4.8 Assinatura de vídeo baseada na diferença entre quadros

Dados dois quadros  $\mathbb{I}$  e  $\mathbb{J}$ , a assinatura temporal desenvolvida por (??) utiliza dois componentes principais para sua construção. O primeiro é a luminância total  $Y$ , ou seja, a soma dos *pixels* do *frame*, representando o brilho total da imagem. O outro parâmetro é  $dY$ , ou diferencial de luminância, sendo a diferença do quadro atual com o quadro anterior.  $dY$  é uma métrica para saber o quanto a luminância varia com a mudança dos quadros no vídeo. Essa variação pode acontecer, segundo o autor, a partir do movimento da câmera, corte entre cenas e objetos e pessoas que entram ou saem de cena.

O valor de  $dY$  pode ser obtido através da Equação 4 (??):

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

Portanto cada *frame* será representando por uma tupla  $(Y, dY)$ .

#### 4.9 Assinatura de vídeo baseada em wavelets

Esta abordagem foi escolhida por ter sido projetada especialmente para ser robusta a uma variedade de ataques fotométricos e de pós-produção, como modificações em contraste, brilho, contaminação por ruído e desfoque, inserção de logos, bordas e mudança de formato do quadro. Para se tornar ainda mais robustas a estes ataques, ??(??) também descreve um fluxo de pré-processamento, abordado na Seção ??.

Após a etapa de pré-processamento e para ser usado 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 ??(??) é composta de

uma parte baseada na transformada de Haar e em outra baseada na distribuição espacial de gradientes.

#### 4.9.1 Transformada de Haar

A transformada de Haar consiste em separar um sinal (ou quadro, neste caso) em 4 partes:

- $LL$ , que contém  $1/4$  dos dados originais, removendo os detalhes.
- $LH$ , que contém a derivada na horizontal do quadro
- $HL$ , que contém a derivada na vertical do quadro
- $HH$ , que contém a derivada na diagonal do quadro

Ela pode ser aplicada de forma recursiva  $n$  vezes, usando o quadro  $I$  como entrada inicial do algoritmo e o  $LL$  como entrada das chamadas subsequentes. Um exemplo do resultado da transformada de Haar pode ser visto na Figura 4.

#### 4.9.2 Assinatura baseada em wavelets

1. Seja  $I$  a imagem obtida após conversão para escala de cinza e normalização
2. Para  $i$  de 1 até  $n$ , onde  $n$  é o número de iterações da transformada de Haar
  - a) Aplicar a transformada de Haar sobre  $I$  para obter um vetor com  $(LL, LH, HL, HH)$
  - b) Computar energia de  $LH, HL, HH$  <sup>1</sup>
3. Computar a energia da subimagem  $I$ , do último valor de  $LL$  e concatenar estes valores em um vetor,
4. Concatenar os valores de energia obtidos nos passos anteriores para obter o descritor

#### 4.9.3 Assinatura baseada em distribuição espacial de gradientes

1. Dado um quadro  $I$  já pré-processado, aplicar a transformada de Haar com um nível e obter um vetor com  $LL, LH, HL, HH$
2. Computar o gradiente de  $LL$  <sup>2</sup>
3. Dividir a imagem de gradiente em  $N_p$  partições com o mesmo tamanho
4. Para cada partição, computar um histograma de gradiente
5. Concatenar os histogramas para obter o descritor

---

<sup>1</sup>  $\frac{1}{MN} \sum_{x=1}^M \sum_{y=1}^N |\mathbb{I}(x,y)|$

<sup>2</sup>A imagem  $LL$  é usada pois contém menos ruído que a imagem original graças à transformada de Haar



Figura 4 – Sequência de transformações feitas pelo algoritmo utilizando um quadro do filme Mulher Maravilha

#### 4.10 Assinatura baseada em quadros de cena

Outra abordagem, proposta por ??(?), é baseada na assinatura de quadros de cena. 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 em 1.3.

O algoritmo fundamenta-se na ideia de que as chances de existirem cinco quadros de cena seguidos é extremamente baixa, por isso são selecionados apenas os cinco primeiros quadros de cena de um vídeo. A forma como estes são determinados é descrita na Seção 4.10.1.

## 4.10.1 A extração de assinatura

A obtenção da assinatura, para  $??(??)$ , é realizada para todos os quadros, para então serem comparadas e selecionadas. Como pode ser observado no Diagrama  $??$ , os quadros passam por um pre-processamento, onde o componente de luminância é obtido. O quadro então é recortando, mantendo-se apenas sua região central e, por fim, redimensionado para o tamanho definido de 3/4QCIF, ou seja,  $(108 \times 132)$ .

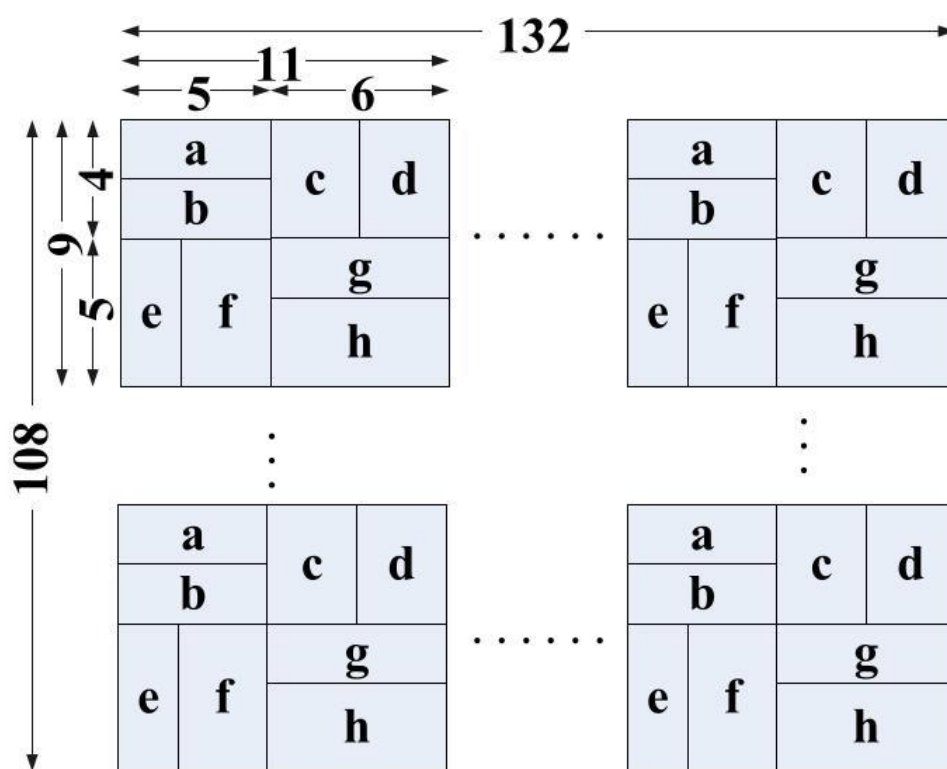


Figura 5 – Divisão da imagem para cálculo dos elementos diferenciais. Referência:  $??(??)$

## 4.10.2 Diminuição do espaço de memória utilizado

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 qualificação quaternária, na qual os valores são classificados de acordo com um *threshold*.