UNIVERSIDADE ESTADUAL PAULISTA
"JÚLIO DE MESQUITA FILHO"
Campus de São José do Rio Preto

Guilherme Brandão Martins

PROCESSAMENTO E ANÁLISE DE VÍDEOS UTILIZANDO FLORESTA DE CAMINHOS ÓTIMOS

Bauru
2016

Guilherme Brandão Martins

PROCESSAMENTO E ANÁLISE DE VÍDEOS UTILIZANDO FLORESTA DE CAMINHOS ÓTIMOS

Dissertação apresentada como parte dos requisitos para obtenção do título de Mestre em Ciência da Computação, junto ao Programa de Pós-Graduação em Ciência da Computação, do Instituto de Biociências, Letras e Ciências Exatas da Universidade Estadual Paulista "Júlio de Mesquita Filho", Campus de São José do Rio Preto.

Orientador: Prof. Dr. João Paulo Papa
Coorientador: Prof. Dr. Jurandy Gomes de Almeida Junior

Bauru
2016

Martins, Guilherme Brandão
Processamento e análise de vídeos utilizando Floresta de Caminhos Ótimos / Guilherme Brandão Martins. – Bauru: [s.n.], 2016.
63 p. : il. ; 30 cm.

Orientador: João Paulo Papa
Dissertação (mestrado) - Universidade Estadual Paulista “Júlio de Mesquita Filho”, Instituto de Biociências, Letras e Ciências Exatas

1. Ciência da computação. 2. Reconhecimento de padrões. 3. Processamento de vídeos. I. Papa, João Paulo. III. Universidade Estadual Paulista “Júlio de Mesquita Filho”, Instituto de Biociências, Letras e Ciências Exatas. IV. Processamento e análise de vídeos utilizando Floresta de Caminhos Ótimos

CDU 00:000:000.0

Guilherme Brandão Martins

PROCESSAMENTO E ANÁLISE DE VÍDEOS UTILIZANDO FLORESTA DE CAMINHOS ÓTIMOS

Dissertação apresentada como parte dos requisitos para obtenção do título de Mestre em Ciência da Computação, junto ao Programa de Pós-Graduação em Ciência da Computação, do Instituto de Biociências, Letras e Ciências Exatas da Universidade Estadual Paulista "Júlio de Mesquita Filho", Campus de São José do Rio Preto.

Orientador: Prof. Dr. João Paulo Papa
Coorientador: Prof. Dr. Jurandy Gomes de Almeida Junior

Comissão Examinadora

Prof. Dr. João Paulo Papa
UNESP – Bauru
Orientador

Prof. Dr. Fábio Faria
UNIFESP – São José dos Campos

Prof. Dr. José Remo Ferreira Brega
UNESP – Bauru

Bauru
20 de Maio de 2016

RESUMO

Com os avanços relacionados às tecnologias de redes computacionais e armazenamento de dados observa-se que, atualmente, uma grande quantidade de conteúdo digital está sendo disponibilizada via internet, em especial por meio de redes sociais. A fim de explorar esse contexto, abordagens relacionadas ao processamento e apredizado de padrões em vídeos têm recebido crescente atenção nos últimos anos. Sistemas de recomendação de filmes, amplamente empregados em lojas virtuais, são umas das principais aplicações no que se refere aos avanços de pesquisa na área de processamento de vídeos. Com o objetivo de acelerar o processo de recomendação e redução de armazenamento, técnicas para classificação e sumarização de vídeos por meio de aprendizado de máquina têm sido utilizadas para explorar conteúdo informativo e também redundante. Por meio de técnicas de agrupamento e descrição de dados, é possível identificar quadros-chave de um conjunto de amostras a fim de que, posteriormente, estes sejam usados para sumarização do vídeo. Além disso, por meio de bases de vídeos rotulados, podemos classificar amostras de modo a organizá-las por gêneros de vídeo. O presente trabalho objetiva utilizar o classificador Floresta de Caminhos Ótimos para sumarização automática e classificação de vídeos por gênero, bem como o estudo de sua viabilidade nestes contextos. Os resultados obtidos mostram que o referido classificador obteve desempenho bastante promissor e próximo à algumas das técnicas de sumarização automática e classificação de vídeos que, atualmente, representam o estado-da-arte no atual contexto.

Palavras-chave: Sumarização de vídeos, Classificação de vídeo, Floresta de Caminhos Ótimos.

ABSTRACT

Currently, a number of improvements related to computational networks and data storage technologies have allowed a considerable amount of digital content to be provided on the internet, mainly through social networks. In order to exploit this context, video processing and pattern recognition approaches have received a considerable attention in the last years. Movie recommendation systems are widely employed in virtual stores, thus being one of the main applications regarding to research advances in the video processing field. Aiming to boost the content recommendation and storage cutback, different video categorization and video summarization techniques have been applied to handle with more informative and redundant content. By availing clustering and data description techniques, it is possible to identify keyframes from a given samples set in order to consider them as part of the video summarization process. Furthermore, through labeled video data collections it is possible to classify samples in order to arrange them by video genres. The main goal of this work is to employ the Optimum-Path Forest classifier in both video summarization and video genre classification processes as well as to conduct a viability study of such classifier in the aforementioned contexts. The results have shown this classifier can achieve promising performance, being very close in terms of summary quality and consistent recognition rates to some state-of-the-art video summarization and classification approaches.

Keywords: Video summarization, video genre classification, Optimum-Path Forest.

Agradecimentos

Ao Criador, por me conceder a vida e permitir o desenvolvimento e conclusão deste trabalho.

A Jesus, por sempre guiar meus passos nas trilhas iluminadas da bondade, do esforço e do trabalho digno.

Ao meu orientador, João Paulo Papa, pela sabedoria e experiência compartilhada, pela atenção sempre oferecida a mim, por ser um grande exemplo de ser humano e profissional e pela amizade sincera.

Aos meus pais, Haroldo Pereira Martins Júnior e Rosana Lúcia Brandão Martins, que sempre me ensinaram o valor do conhecimento, do trabalho e do esforço, além do constante apoio dado a mim durante toda a vida. Nunca terei palavras suficientes para dizer o quanto sou grato a vocês. Pai e mãe, muito obrigado, de coração. Amo vocês!

Aos meus irmãos, Gustavo e Fernando, pelo companheirismo e amizade. O amor que tenho por vocês torna meus dias sempre mais alegres.

A toda a minha família que, direta ou indiretamente, participou desta etapa junto a mim.

A minha companheira, Heloísa, que em tão pouco tempo, tem me ensinado o verdadeiro significado de respeito e amizade entre dois seres que se amam mutuamente. Seu apoio e compreensão facilitaram a conclusão deste trabalho. Amo você, meu bem. Obrigado por tudo!

A todos os meus amigos, àqueles que estiveram ou estão presente em minha vida. Em especial ao Dheny Fernandes, grande amigo que conheci na universidade, e que sempre me auxiliou durante os dois anos desta pós-graduação.

Aos amigos Kauê, Rhuan, Wagner e Pedro, pela amizade verdadeira e pelo apoio diário que recebi de vocês durante os últimos anos. Vocês sempre farão parte da minha família, nunca se esqueçam disso.

A todos os professores e professoras que estiveram presentes em minha vida. Sem vocês, nunca teria chegado tão longe. Todo o conhecimento adquirido ao longo dos anos, eu dedido a vocês, mestres. Muito obrigado!

A UNESP e a todos os funcionários desta instituição, por oferecerem as condições necessárias ao meu desenvolvimento pessoal e profissional ao longo dos últimos 6 anos.

«O teu trabalho é a oficina em que podes forjar a tua própria luz.»

Emmanuel

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Abreviaturas e Siglas

ACC	Auto Color Correlogram
ANN-MLP	Artificial Neural Network - Multilayer Perceptron
BIC	Border/Interior Pixel Classification
BoS	Bag-of-Scenes
BoVW	Bag-of-Visual-Words
CBVR	Content-base Video Retrieval
CCV	Color Coherence Vector
CUS	Comparison of User Summaries
DCT	Discrete Cosine Transform
DT	Delaunay Triangulation
FPS	Frames per second
GCH	Global Color Histogram
GFD	Generic Fourier Descriptor
GJD	Generalized Jaccard Distance
GOP	Group of Pictures
HMP	Histogram of Motion Patterns
HSV	Hue-Saturation-Value
HWD	Haar-Wavelet Decomposition
k-NN	k-Nearest-Neighbors
MAP	Mean Average Precision
MOS	Mean Opinion Score
OPF	Optimum-Path Forest
OPT	Optimum-Path Tree
PCA	Principal Component Analysis
PoP	Pooling over Pooling
SIFT	Scale Invariant Feature Transform
STIMO	Still and Moving Video Storyboard

SVM	Support Vector Machines
VISON	Video Summarization for Online Applications
ZNCC	Zero-mean Normalized Cross Correlation

Capítulo 1

Introdução

1.1 Motivação e Justificativa

Nos últimos anos, os recentes avanços ligados às tecnologias de armazenamento, compressão e transmissão de dados têm facilitado a propagação de conhecimento e informação via internet. Como consequência, surge então um novo contexto em que comunicação e interação virtual entre pessoas intensifica-se cada vez mais, principalmente por meio de redes sociais, tais como Facebook e Twitter, por exemplo. Além disso, percebe-se um crescente aumento na quantidade de conteúdo digital multimídia produzido não apenas por computadores *desktop* e *notebooks*, mas principalmente por dispositivos móveis como *tablets* e *smartphones*.

Grande parte do volume de dados gerado por dispositivos móveis diariamente na internet é composto por vídeos. Por conseguinte, a quantidade de vídeos disponíveis em ambiente virtual é tão grande que tornou-se inviável realizar uma busca por um determinado conteúdo de interesse simplesmente pela análise de todos os vídeos disponíveis [8]. Surge então a necessidade do estudo de novas possibilidades a fim de descobrir soluções viáveis capazes de superar as limitações computacionais existentes relacionadas ao atual contexto.

Dessa maneira, o gerenciamento eficiente de dados de vídeo passa a ser de grande importância, pois com sua implementação é possível otimizar aplicações como motores de busca e bibliotecas digitais [5]. Nesse contexto, a área de processamento de vídeos tem recebido grande atenção por parte dos pesquisadores. Dentre as tarefas de maior interesse nessa área, podemos citar classificação ou categorização [8, 11, 24, 49], recuperação de informações

baseada em conteúdo [22, 27, 43, 52], compressão/codificação por meio de técnicas de aprendizado de máquina [14, 48, 50] e sumarização automática [12, 17, 32, 44]

Para melhor lidar com arquivos multimídia e facilitar às pessoas o acesso a vídeos diversos por meio de busca e recuperação intuitivas, é necessário integrar metadados relacionados a detalhes técnicos, termos de uso, descrição e análise do conteúdo dos vídeos [11]. Além disso, a representação de seus conteúdos e classificação por categorias semanticamente significativas são questões igualmente importantes [24].

Um modo interessante de resolver essa questão é restringir as possíveis escolhas a serem feitas por um usuário agrupando vídeos segundo categorias ou gêneros específicos [8]. Por exemplo, uma pesquisa por um vídeo pertencente ao gênero negócios, seria realizada considerando-se apenas o conjunto de vídeos rotulados com essa mesma categoria. Essa técnica é chamada de classificação ou categorização de vídeo, e torna busca e recuperação de conteúdos em videos tanto mais intuitivas quanto eficientes. Dessa forma, pode ser de grande utilidade para construção de sistemas de recomendação em bibliotecas digitais, nos quais a pré-organização categórica do conteúdo multimídia é essencial ao bom desempenho computacional destes sistemas. Além disso, segundo Xu e Li [49], a classificação de vídeos também é útil para selecionar automaticamente um programa desejado pelo usuário, bem como para classificar um vídeo a partir de seu conteúdo.

Considerando as diversas metodologias para classificação de vídeos presentes na literatura [8], podemos agrupá-las em quatro categorias básicas: textuais, baseadas em áudio, visuais e combinadas, sendo a categoria visual a mais utilizada e custosa em termos de processamento computacional. Dentre as técnicas, podemos citar Modelos de Mistura Gaussiana – *Gaussian Mixture Model* –, Modelos Ocultos de Markov – *Hidden Markov Models*, Redes Neurais Artificiais, Máquinas de Vetores de Suporte (SVM), entre outras.

Já a recuperação de informações baseada em vídeos (*Content-based Video Retrieval* - CBVR) tem por objetivo encontrar vídeos de conteúdo semelhante a um dado vídeo de consulta. É válido ressaltar que, para este tipo de aplicação, a eficiência no processo de recuperação é uma prioridade, já que vídeos são compostos por uma sequência de centenas ou milhares de quadros, o que torna

o processamento computacional bastante dispendioso.

Quanto à compressão e codificação, diversas técnicas de aprendizado de máquina têm sido estudadas com o intuito de tornar o conteúdo digital de alta resolução mais acessível em relação a tempo e processamento. As chamadas "TVs interativas" são um bom de exemplo de aplicação, já que estas são responsáveis por disponibilizar o conteúdo em sinal digital e também por permitir a interação de usuários com programas de televisão.

Um canal esportivo, por exemplo, é capaz de disponibilizar aos seus usuários tomadas de câmeras em ângulos especiais durante uma partida de futebol, podendo estas serem acessadas pelo controle remoto da televisão. Além disso, o usuário pode rever, a seu critério, um momento favorito de uma partida de futebol, mesmo que isso ocorra durante a transmissão dessa partida.

Assim, é possível destacar a importância de estudos direcionados ao "entendimento de vídeo" – *video understanding* – que significa, em suma, explorar as características intrínsecas de cada vídeo, de forma a proporcionar uma descrição mais eficaz do mesmo.

Outra aplicação que vem sendo bastante estudada nos últimos anos é a sumarização automática. Esta pode ser entendida como o processo de aquisição de quadros ou de tomadas de um vídeo digital resultando na geração de uma sequência de imagens ou de segmentos capazes de representar o conteúdo mais relevante do vídeo.

Dado que o uso de vídeos tem aumentado, e por isso existe a necessidade de processá-los, um dos objetivos da sumarização, neste contexto, é fornecer aos usuários uma versão reduzida de um vídeo, de forma que esta seja sintética, visualmente útil e fácil de ser interpretada [10].

De modo geral, um sumário automático pode ser apresentado de forma estática (*storyboard*) ou dinâmica (*dynamic video skimming*) [47]. Um sumário estático ou *storyboard* é a representação sequencial dos quadros mais relevantes de um vídeo, enquanto que o sumário dinâmico é formado de pequenos segmentos animados, também de maior relevância no vídeo, e que são apresentados ao usuário na forma de um vídeo de curta duração [10]. Esses tipos de representação podem ser de grande valia para emissoras de televisão, por exemplo. Neste caso, um sumário poderia ser utilizado para resumir o conteúdo

de uma partida de futebol, de forma que possa ser utilizada por comentaristas em programas de esportes.

Outra aplicação interessante para a sumarização está na geração de bases de dados compostas apenas por sumários de vídeos, e estas poderiam ser posteriormente utilizadas para classificação e recuperação de informação baseada em conteúdo de vídeo. Dessa maneira, realizando o processamento de um sumário, ao invés do vídeo todo, é possível reduzir a carga computacional e aumentar a eficiência do processo como um todo.

1.2 Objetivos

Introduzir o classificador Floresta de Caminhos Ótimos (*Optimum-Path Forest* – OPF) na área de processamento de vídeos, especificamente em sumarização automática e classificação de vídeos utilizando, respectivamente, as versões não supervisionada e supervisionada do referido classificador.

Adicionalmente, este trabalho visa comparar a *performance* do OPF com outras abordagens presentes na literatura, bem como analisar os resultados alcançados a fim de otimizar esse classificador no desempenho de sumarização automática e classificação de vídeos.

1.3 Organização do Trabalho

O restante do documento é organizado como segue. O referencial teórico acerca dos conceitos fundamentais de vídeo digital, sumarização automática, classificação de vídeos e descritores visuais é apresentado no Capítulo 2. O Capítulo 3 descreve a teoria a respeito do classificador Floresta de Caminhos Ótimos para os aprendizados supervisionado e não supervisionado. As técnicas mais relevantes de sumarização estática de vídeos, que foram estudas para desenvolvimento do presente trabalho, são apresentadas e brevemente descritas no Capítulo 2.5. Os Capítulos 4 e 5 descrevem, respectivamente, as abordagens propostas utilizando OPF para sumarização estática de vídeos e classificação baseada em gêneros, bem como suas análises de comportamento e desempenho, e resultados alcançados. Finalmente, as considerações

finais, trabalhos futuros e publicações decorrentes deste trabalho estão presentes no Capítulo 6.

Capítulo 2

Referencial Teórico

Este capítulo apresenta a teoria utilizada no desenvolvimento deste trabalho. Sendo a área de processamento de vídeos muito vasta, são descritos nas seções seguintes apenas os conteúdos específicos e essenciais para a adequada compreensão das abordagens de sumarização estática e classificação de vídeos baseada em gêneros.

A Seção 2.1 descreve, brevemente, a estrutura de um vídeo digital e a terminologia dos principais conceitos utilizados no presente trabalho, enquanto que as Seções 2.2 e 2.3 introduzem os temas de sumarização automática e classificação de vídeos, respectivamente. Os descritores de características visuais são brevemente descritos na Seção 2.4, enquanto que as técnicas mais relevantes de sumarização estática são descritas na Seção 2.5.

2.1 Conceitos fundamentais

A base dos trabalhos apresentados neste documento é o vídeo digital, o qual pode ser definido como uma sequência limitada de inúmeras imagens digitais, codificadas de forma a criar a ilusão de movimento, sendo esta mais suave à medida em que o número de imagens exibidas por tempo é aumentada. A medida que cresce a quantidade de imagens a serem codificadas, maior será a suavidade de exibição do conteúdo do vídeo. Todavia, isso exigirá maior espaço de armazenamento em disco.

Um vídeo não necessariamente é um tipo restrito à representação visual, pois permite que a ele sejam incorporadas outras modalidades de informação

como, por exemplo, áudio e texto. Dessa forma, um vídeo também pode ser compreendido como um tipo de repositório complexo de dados [10].

Realizando uma análise hierárquica da estrutura de um vídeo, podemos organizá-lo a partir de certas unidades fundamentais. A Figura 2.1 apresenta a anatomia de um vídeo digital considerando essa estrutura de unidades.

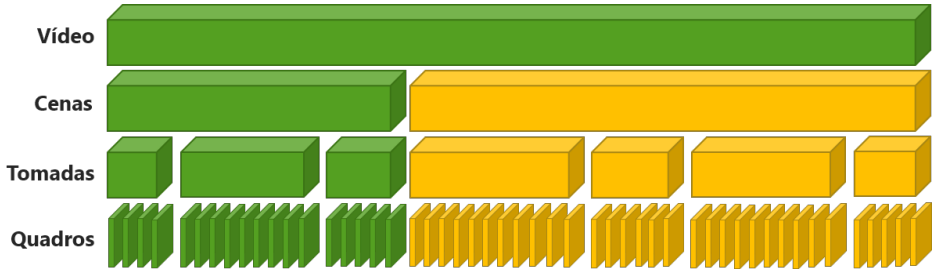


FIGURA 2.1: Unidades fundamentais do vídeo digital.

A primeira e mais básica dessas unidades é o **quadro** (imagem digital convencional). Em seguida, temos a **tomada** ou *shot*, a qual pode ser entendida como um conjunto de quadros obtidos em apenas uma passagem de câmera, ou seja, que não apresente cortes. Reunindo diversas tomadas, é possível obter uma unidade que apresente um sentido lógico e contextual, para a qual damos o nome de **cena**.

2.1.1 Terminologia

Formalmente, podemos definir as unidades fundamentais já citadas, e que estarão presentes nos demais capítulos deste trabalho.

Um vídeo V é definido como uma sequência de quadros q_t, dada por:

$$V = (q_1, q_2, ..., q_n), \tag{2.1}$$

onde q_t representa a posição do quadro no tempo t, e n o número total de quadros presentes no vídeo V.

A menor unidade é o quadro, isto é, uma imagem digital convencional que pode ser representada na forma $q_i(x, y) = (r, g, b)$, onde $x \in 1, ..., M$, $y \in 1, ..., N$, (x, y) representa a localização de um *pixel* dentro de uma imagem, $M \times N$ refere-se ao tamanho do quadro de vídeo, e (r, g, b) representa os

valores de intensidade para os canais vermelho (*red*), verde (*green*) e azul (*blue*) da imagem, respectivamente.

Uma tomada T de vídeo é um conjunto limitado de quadros obtidos em uma única movimentação de câmera, sendo assim, uma sequência espaço-temporal de quadros. Já uma cena C representa a unidade de vídeo mais complexa dentre as já descritas, e pode ser definida como um conjunto de tomadas que possuem algum tipo de correlação contextual e que, além disso, apresentam proximidade temporal.

2.2 Sumarização automática

A próxima etapa dessa conceituação é compreender algumas das novas possibilidades de representação do vídeo digital. A primeira delas é a sumarização automática, técnica que permite, por meio de processamento de imagens e reconhecimento de padrões, a criação de uma versão reduzida de um vídeo que, além disso, é capaz de preservar as informações mais relevantes do mesmo [4].

Um sumário pode ser construído a partir dos diversos tipos de informação presentes no vídeo digital. A criação de um sumário de vídeo depende da análise do conteúdo fundamental e da incorporação de elementos audiovisuais do mesmo a fim de apresentar ao usuário uma representação condensada e sucinta [32].

Quadro-chave, tomada de vídeo e texto compõem os elementos audiovisuais mais relevantes de um vídeo, e são brevemente descritos a seguir:

- Quadro-chave: imagens mais representativas do vídeo, organizadas e apresentadas em ordem temporal. Sumários gerados a partir de quadros-chave são estáticos, portanto contém apenas informações visuais (imagens e texto).

- Tomada de vídeo: extensão dinâmica de um quadro-chave, isto é, segmento capaz de preservar tanto informações visuais, quanto as de movimento e de áudio mais relevantes do vídeo.

- Texto: informação extraída do vídeo por meio de descritores textuais. Normalmente, é mais utilizado como suporte aos outros tipos de elementos na geração de sumários.

A partir desses elementos, é possível criar diferentes tipos de sumários. De modo geral, podemos classificá-los em estáticos e dinâmicos [47]. Um sumário **estático** ou *static storyboard* é composto por um sequência de quadros-chave extraídos do vídeo original, e que podem conter informação textual. Além disso, não estão sujeitos à sincronização e são mais flexíveis no que se refere a organização para propósitos de busca e navegação [5].

O sumário do tipo **dinâmico**, ou *dynamic video skimming*, é composto por tomadas de vídeo, e são capazes de incorporar elementos de movimento e áudio, o que melhora a expressividade e informações transmitidas ao usuário [4].

A construção do sumário, normalmente, é a última etapa do processo de sumarização automática. As etapas anteriores à geração do sumário podem ser diversas dependendo do embasamento teórico e das técnicas utilizadas, bem como do tipo ou gênero de vídeo a ser sumarizado.

No que diz respeito ao processo de sumarização estática de um vídeo, foi possível conceber, de modo geral, uma estrutura básica de etapas considerando as metodologias de sumarização das principais abordagens presentes na literatura. A referida estrutura de sumarização é apresentada na Figura 2.2.

FIGURA 2.2: Estrutura básica do processo de sumarização estática.

Inicialmente, o vídeo necessita ser decodificado, o que pode ser feito de modo parcial ou integral. Essa tarefa ocorre na etapa de pré-amostragem, a qual tem por finalidade transformar o vídeo de entrada em um conjunto de quadros que possam ser devidamente processados em etapas subsequentes.

Para obtenção de um determinado número de quadros, é necessário estabelecer uma taxa de amostragem para o vídeo, isto é, uma forma de controlar a quantidade de quadros decodificados em um determinado período. Basicamente, quanto maior a taxa de amostragem, maior será o conjunto de quadros pré-amostrados, e vice-versa.

Em seguida, utilizando o conjunto de quadros pré-amostrados, é feita a extração das características de cada imagem para obtenção de seus descritores visuais, representados na forma de vetores de características, e que juntos compõem o conjunto de amostras que de fato serão utilizadas para a construção do sumário automático. A criação dos vetores de características pode ser feita por meio de abordagens diversas presentes na área de processamento de imagens digitais. Todavia, as abordagens mais comuns – devido, principalmente, à eficiência na representação das características visuais da imagem – são os histogramas de cores [4, 5, 15, 18, 33].

Na etapa seguinte, geralmente é aplicada uma técnica de aprendizagem não supervisionada ao conjunto de amostras com o intuito de agrupá-las segundo algum tipo de semelhança visual. Consequentemente, abordagens de *clustering* são as mais utilizadas para execução desta etapa. Calculados os agrupamentos, é realizada a seleção de quadros-chave, isto é, segundo alguma diretriz, escolhemos os quadros que melhor representam, as informações de cada agrupamento de amostras. Finalmente, os quadros-chave são organizados temporalmente dando origem ao sumário estático do vídeo.

Vale ressaltar que, na estrutura de sumarização, podem ser introduzidas etapas adicionais de pré e pós-processamento, que têm por objetivo melhorar a representação do sumário a ser gerado por meio da remoção de quadros inexpressivos e redundantes. Algumas das abordagens mais relevantes da literatura incluíram essas etapas em suas metodologias de sumarização e obtiveram excelentes resultados, ou seja, sumários estáticos contendo um pequeno número de quadros-chave, sendo estes capazes de representar de maneira concisa o conteúdo original do vídeo [4, 5].

2.3 Classificação baseada em gêneros

Um dos problemas fundamentais na área de reconhecimento de padrões é a classificação, a qual visa predizer e associar uma dentre *c* classes possíveis à cada amostra do conjunto de classificação [34]. Além disso, tem por base a extração de características de baixo nível, objetivando determinar a similaridade entre vídeos por meio do cálculo da distância entre vetores de características [3]. Os objetos ou elementos a serem classificados são chamados de amostras e juntos formam um conjunto de dados que, normalmente, está dividido em dois subconjuntos: treinamento e teste. O primeiro é destinado a construção do classificador, enquanto que o segundo é utilizado para validar as amostras classificadas medindo-se os erros de classificação. Dessa forma, a classificação pode ser aplicada na resolução de inúmeros problemas, sendo um destes a categorização de vídeos digitais.

A quantidade de conteúdo digital disponível atualmente já não permite que a busca por um dado vídeo de interesse seja feita manualmente. A fim de otimizar esse processo em termos de agilidade de pesquisa, organização, anotação e recuperação de conteúdo em bases de vídeos, um dos métodos mais pesquisados é a busca por um vídeo tendo por base gêneros ou categorias específicas [8, 51].

Geralmente, no processo de classificação, são considerados os tipos possíveis de características que estão presentes em um vídeo digital como, por exemplo, texto, áudio e elementos visuais [8]. Essas três modalidades compõem a base das abordagens de classificação de vídeo, e podem ainda ser utilizadas em conjunto com o intuito de otimizar este processo.

Cada categoria ou gênero é formada por determinado conjunto de características que juntas são capazes de caracterizar um dado vídeo. Por exemplo, filmes de terror tendem a apresentar baixa iluminação, enquanto que em filmes de comédia ocorre justamente o contrário. Em filmes de ação, a movimentação e cortes de câmera são constantes, o que já não ocorre em filmes da categoria drama [8]. Esse cenário é útil, pois mostra que a escolha das características a serem extraídas de um vídeo pode ser determinante com relação à correta distinção entre gêneros.

A seguir, são descritas as modalidades de características que compõem um vídeo e que influenciam no processo de classificação baseada em gêneros:

- Textual: textos presentes no vídeo e que podem ser visualizados, além daqueles obtidos pela transcrição de diálogo. O primeiro está presente em objetos de cena, legendas e elementos gráficos (placar de jogos, por exemplo); já o segundo é obtido por meio de técnicas de reconhecimento de fala ou legenda.

- Áudio: extraído a partir de ruídos, sons emitidos por objetos e falas de um personagem, além de música ambiente, entre outros. De acordo com Liu et al. [28], os exemplos anteriores podem pertencer, respectivamente, a três camadas de entendimento de áudio: níveis baixo, médio e alto.

- Visual: toda e qualquer característica extraída a partir de cores, texturas ou movimento. Essa modalidade é a mais utilizada na literatura no contexto de classificação, podendo, ainda, ser dividida em características baseadas em cores, tomadas, objetos e movimento.

O uso dessas modalidades pode ser combinado, isto é, podemos incorporar elementos textuais, de áudio e visuais considerando o processo de visualização de vídeo feito por um usuário com a finalidade de aprimorar a qualidade da classificação, minimizando-se os pontos negativos de cada modalidade. Vale ressaltar que a combinação de características textuais e visuais pode facilitar o processo de classificação de vídeos com base em gêneros diversos [8].

2.4 Descritores visuais

Um descritor de imagens é responsável por caracterizar as propriedades de uma imagem e computar suas similaridades, o que torna possível o ranqueamento destas de acordo com suas propriedades ou características visuais [38].

Para que esse processo ocorra, o descritor é representado na forma de um vetor de características, que tem por intuito armazenar as informações de cor, textura, forma e relação espacial entre objetos presentes em uma imagem, por

exemplo. Tanto para sumarização, quanto para classificação de vídeos, foram utilizados descritores de imagem para representação do conteúdo dos vídeos.

Os itens 2.4.1 e 2.4.2 descrevem, de forma resumida, os descritores visuais aplicados em sumarização estática e classificação de vídeos baseada em gêneros, respectivamente.

2.4.1 Descritores para sumarização estática

A escolha dos descritores para sumarização estática de vídeo foi embasada nos estudos comparativos realizados por Pennati et al. [38], que analisaram diversos descritores de cor e textura com o objetivo de filtrar os melhores algoritmos para descrição de imagens digitais.

Auto Color Correlogram

Proposto por Huang et al. [23], o descritor de imagens nomeado *Auto Color Correlogram* (ACC) faz o refinamento e distribuição global da correlação espacial das cores, tendo este bastante utilidade para indexação e recuperação de imagens. A principal vantagem do descritor ACC é o fato deste ser tolerante à grandes mudanças com relação à aparência e forma causadas por mudanças na posição de visualização, aproximação de câmera e oclusão parcial, por exemplo. Além disso, tem alta efetividade na recuperação de imagens baseada em conteúdo, considerando grandes bases de imagens.

De forma geral, um correlograma de cores é uma tabela indexada por pares de cores, na qual a k-ésima entrada para $< i, j >$ especifica a probabilidade α de encontrar um pixel da cor j na distância k para um dado pixel i da imagem. A probabilidade α é calculada a partir do histograma de cores dessa imagem.

Border/Interior Pixel Classification

É composto por três componentes básicos: (1) simples mas poderoso algoritmo de análise de imagens; (2) nova abordagem de distância logarítmica (dLog) ; e (3) representação compacta para as características visuais extraídas das imagens. Essa abordagem por proposta por Stehling et al. [45] e visa domínios amplos de imagens digitais.

O descritor *Border/Interior Pixel Classification* (BIC) [45], dada uma imagem qualquer, é construído tendo como base o espaço de cores RGB quantificado em 64 cores. Os *pixels* dessa imagem são classificados como "de borda" ou "de interior", e baseia-se na propriedade visual binária inerente à imagem. A partir dessa classificação, são calculados dois histogramas de cores, um considerando apenas os *pixels* de borda, e outro que leva em conta apenas os *pixels* de interior, cada qual possuindo 64 posições. Em seguida, os histogramas são comparados como sendo apenas um de 128 posições por meio da distância dLog. Por fim, o resultado é armazenado em metade do espaço da representação original, que é composto por 128 características.

Color Coherence Vector

Color Coherence Vector (CCV), proposto por Pass, Zabih e Miller [36], é um método para comparação de imagens baseado em histograma de cores capaz de incorporar informação espacial das imagens. De modo geral, os *pixels* de uma dada imagem são classificados em coerentes ou incoerentes com base na região de similaridade a que fazem parte. Essa característica mostrou-se capaz de proporcionar distinções mais sutis em comparação a um histograma de cores.

Inicialmente, a imagem é ligeiramente borrada a fim de eliminar pequenas variações de cor entre *pixels* próximos. Para tal, calcula-se o valor médio de cor considerando um vizinhança de 8 *pixels*. Em seguida, todos os *pixels* da imagem são conectados entre si de modo a criar regiões de similaridade de cor. A partir dessas regiões, cada *pixel* é classificado como sendo coerente (quantidade de *pixels* de sua região de similaridade é maior que um dado limiar) ou incoerente (caso contrário).

Por fim, são formados pares contendo a quantidade de *pixels* coerentes e incoerentes para cada cor da imagem, chamados pares de coerência. O agrupamento de todos os pares de coerência origina o vetor de características CCV.

Global Color Histogram

Introduzido por Swain e Ballard [46], *Global Color Histogram* (GCH) é uma técnica já bastante conhecida e largamente utilizada em sistemas de recuperação de imagens baseados em conteúdo. Além disso, GCH destaca-se por servir como base de comparação em diversos trabalhos de análise de imagens presentes na literatura [37].

Um histograma de cores é uma representação da distribuição das cores existentes em uma dada imagem. A ocorrência de *pixels* para cada cor da imagem é armazenada em uma posição de um vetor de características. Vale ressaltar que, para que o histograma seja construído eficientemente, é necessário previamente discretizar o espaço de cores do qual a imagem digital pertence.

Como principais vantagens de utilização de descritores GCH, podem ser citadas invariância à translação e à rotação sobre um eixo perpendicular á imagem. Além disso, esses descritores são capazes de suportar pequenas mudanças com relação à rotação sobre outros tipos de eixo, oclusão e mudanças na distância para objetos na imagem. Desse modo, os histogramas de cores são excelentes representações na identificação de objetos [46].

Generic Fourier Descriptor

Comumente utilizado para descrição de formas. O descritor GFD, *Generic Fourier Descriptor*, tem por base a transformada de Fourier para obtenção de informações espectrais da imagem digital. Segundo Zhang e Lu [53], obter características no domínio da frequência é útil para evitar ruídos comuns em imagens digitais, além do que essas características são mais concisas em comparação àquelas obtidas no domínio espacial.

A construção do descritor GFD ocorre em quatro etapas: normalização de translação, aplicação da transformada de *Fourier* polar, normalização da rotação e normalização de escala. Basicamente, a Transformada de *Fourier* unidimensional é calculada para extração de informações de contorno da forma usando uma função de distância centróide. Em seguida, a Transformada de *Fourier* bi-dimensional é aplicada no espaço polar para garantir que o descritor seja invariante a rotação e escala.

Haar-Wavelet Descriptor

O descritor HWD – *Haar-Wavelet Descriptor* – é uma técnica de representação de imagens que tem como base a obtenção das informações espectrais de uma dada imagem. Essa abordagem, proposta por Jacobs et al. [25], considera o espaço de cores YIQ (Luminância, Matiz e Saturação) para representar imagens. A decomposição destas foi baseada na *wavelet* de Haar devido a seu baixo custo computacional, e foi utilizada a função de base padrão retangular para decomposição.

A fim de melhorar o poder discriminativo da técnica, optou-se por manter, após a decomposição *wavelet*, apenas os coeficientes de maior magnitude, os quais foram quantificados em dois níveis básicos. Finalmente, aplicou-se um fator de normalização na função de base padrão para tornar todas as *wavelets* obtidas ortonormais umas as outras. O objetivo principal desse descritor é tornar o processo de busca e recuperação de imagens mais ágil em grandes bases de imagens.

2.4.2 Descritores para classificação de vídeos

Em relação à classificação de vídeos, optamos por utilizar novas abordagens de descrição de imagens, as quais visam aprimorar a representação de vídeos genéricos [3].

Bag-of-Visual-Words

Esse descritor representa o conteúdo visual através de informações estatísticas obtidas dos padrões locais por meio da codificação das ocorrências das características locais quantificadas. Para isso, foi construído um dicionário visual a partir da quantificação do espaço de características ([7]).

As características *Bag-of-Visual-Words* (BoVW) são então extraídas considerando a técnica *Pooling over Pooling* (PoP) proposta por [3], e o número total de palavras visuais que irão compor o dicionário. Além disso, foi considerado um *pooling* médio na computação da BoVW de cada quadro de vídeo, e um *pooling* máximo para combinação destes mesmos quadros.

Bag-of-Scenes

Técnica desenvolvida por Penatti et al. [37], *Bag-of-Scenes* busca realizar a codificação das propriedades visuais de vídeos digitais. Baseada em um dicionário de cenas, é capaz de carregar mais informação semântica do que os dicionários tradicionais baseados em descrição local.

Bastante similar à abordagem BoVW, necessita que seja definido o total desejado de cenas para geração do dicionário. Para calcular a BoS de cada quadro e posteriormente combiná-los, foram utilizados *poolings* médio e máximo, respectivamente.

Histogram of Motion Patterns

Abordagem estrutural e estatística para representação visual. Introduzida por Almeida, Leite e Torres [2], foi desenvolvida com a finalidade de comparar vídeos utilizando decodificação parcial da sequência de vídeo, extração de características de movimento e geração de assinatura para sua representação.

De modo geral, para cada grupo de figuras (*group of pictures* - GOP[1]) de uma dada sequência de vídeo, apenas os *I-frames*[2] são selecionados devido à sua baixa perda de informação, em comparação aos demais quadros.

Em seguida, cada um desses quadros é dividido em 6 macroblocos de 8x8 *pixels* (4 de luminância e 2 de crominância), dos quais são calculados um valor que representa 8 vezes a intensidade média de cada bloco de *pixels* (termo DC). Aplica-se então, a transformada discreta cosseno (DCT) aos blocos de luminância para obtenção de uma valor médio (assinatura) que represente cada macrobloco.

Os valores de intensidade dos macroblocos são ranqueados de modo a formar uma matriz que representa os padrões espaço-temporais existentes na sequência de vídeo. Finalmente, esses padrões são acumulados para formar um histograma normalizado.

[1] Unidade básica que compõe um vídeo no formato MPEG.

[2] Um GOP é composto por uma sequência de quadros, os quais podem ser de três tipos: *I-frames*, *P-frames* e *B-frames*. Um *I-frame* é o primeiro quadro de um GOP e é capaz de representar a unidade básica como um todo (alta representatividade.)

Devido a capacidade do HMP em reconhecer padrões de movimento de um vídeo e contar suas ocorrências para construção de um histograma, esta técnica é bastante adequada para aplicação em grandes coleções de sequências de vídeos [2].

2.5 Estudos Correlatos

Nesta Seção são apresentadas as principais abordagens presentes na literatura referentes a sumarização estática de vídeos e que foram utilizadas como base para o desenvolvimento deste trabalho. O item 2.5.5 contém as conclusões acerca das metodologias de cada abordagem de sumarização estática.

2.5.1 Agrupamento de quadros por Delaunay Triangulation

Agrupamento de quadros por *Delaunay Triangulation* (DT) é uma abordagem de sumarização estática de vídeo proposta por Mundur, Rao e Yesha [33], e tem como finalidade facilitar o processamento em lote de vídeos sem a necessidade de interação humana, ou seja, livre de parâmetros de inicialização.

O processo de sumarização do DT ocorre ao longo de três etapas: (1) Extração de características, (2) Agrupamento de quadros do vídeo, e (3) Seleção de quadros-chave. Opcionalmente, pode-se executar uma etapa de pré-amostragem, ou seja, antes da etapa (1), onde o vídeo de entrada é decodificado a partir de uma taxa de amostragem visando obter um número reduzido de quadros. Essa decodificação pode ser feita utilizando tanto detecção de tomadas quanto a simples amostragem dos quadros do vídeo.

A extração de características é realizada no espaço de cores HSV para criação de histogramas de cores que darão origem a vetores de características com 256 dimensões. A partir desses vetores, uma matriz de dimensões $n \times 256$ é construída, onde n é o total de vetores (quadros amostrados do vídeo). A fim de reduzir as dimensões dessa matriz – e consequentemente o tempo de processamento –, é aplicado à essa estrutura o *Principal Component Analysis* (PCA) [40], gerando, dessa maneira, um espaço refinado de características.

Na segunda etapa, é realizada a computação dos agrupamentos utilizando o algoritmo *Delaunay Triangulation*. A ideia básica desse algoritmo é representar os quadros do vídeo como sendo pontos de dados na dimensão reduzida (obtida pela aplicação do PCA) para geração do diagrama de Delaunay, que utiliza como referência a identificação de arestas intra-agrupamentos e inter-agrupamentos na modelagem DT. Os agrupamentos são encontrados pela remoção das arestas inter-agrupamentos, situação esta que pode ser visualizada na Figura 2.3. Todo o processo de computação dos agrupamentos pode ser consultado em [33]. Identificados os agrupamentos, é feita a seleção de quadros-chave, isto é, a centróide de cada um dos agrupamentos que irão compor o sumário estático do vídeo.

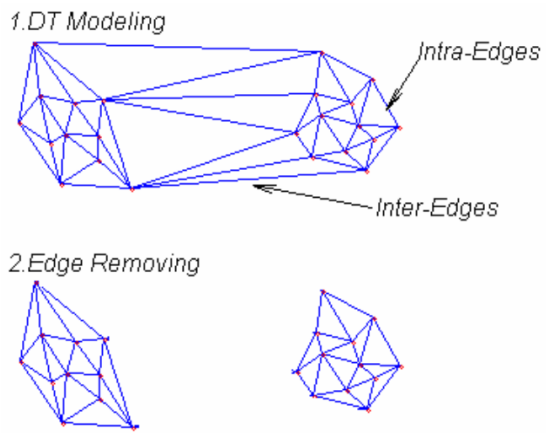


FIGURA 2.3: Arestas de Delaunay. Fonte: Mundur, Rao e Yesha, 2006, p. 9.

Devido a falta de uma metodologia robusta para avaliação dos sumários automáticos, os autores construíram um modelo para medir o desempenho do DT em comparação aos sumários da OV[3] e agrupamento de quadros por k-médias [29]. Dessa maneira, foram utilizadas métricas com o intuito de (1) determinar a significância do conteúdo representado em cada agrupamento (Fator de Significância), (2) realizar uma comparação significativa entre sumários de diferentes abordagens (Fator de Sobreposição), e (3) quantificar a redução do número de quadros em relação ao vídeo original (Fator de Compressão). A análise do

[3]OpenVideo Project, base de vídeo pública amplamente utilizada no contexto de sumarização automática de vídeos

desempenho do DT, bem como os resultados comparativos às outras abordagens, podem ser consultados em maiores detalhes em *"Keyframe-based Video Summarization using Delaunay Clustering"* [33].

2.5.2 Still and Moving Video Storyboard

Still and Moving Video Storyboard (STIMO) é a abordagem de sumarização estática e dinâmica proposta por Furini et al. [18]. Os principais objetivos dessa abordagem são: (1) gerenciamento de vídeos genéricos, (2) customização avançada por parte dos usuários, (3) fazer uso exclusivamente de informações visuais e de áudio na geração do sumário e, (4) construção deste em tempo razoável e de qualidade aceitável para garantir o uso do *storyboard* em tempo real.

Como o foco desta monografia é a sumarização estática, daremos enfoque apenas à descrição das etapas de funcionamento da versão estática do STIMO, bem como o modo como este foi avaliado. A estrutura dessa abordagem, composta por três etapas, pode ser visualizada através dos diagramas presentes na Figura 2.4. A Extração de Características é realizada no espaço de cores HSV para construção de vetores de características a partir de histogramas de cor genéricos de 256 dimensões [18]. Em seguida, os vetores de características são inseridos em uma matriz de dimensão $n \times 256$, onde n é o total de vetores.

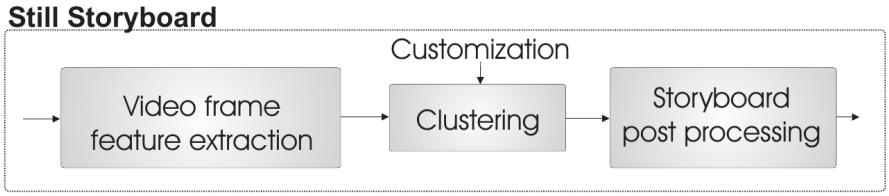


FIGURA 2.4: Estrutura do STIMO para sumarização estática.
Fonte: Furini et al., 2010, p. 7.

A matriz de características obtida passa a ser o conjunto de dados que será utilizado como entrada na etapa de agrupamento. Os agrupamentos de quadros semelhantes foram calculados por meio de uma versão modificada do algoritmo *Furthest Point First* [19, 21], de modo a evitar o cálculo desnecessário de algumas distâncias entre quadros (vetores de características) para acelerar

a computação dos agrupamentos [18]. De modo geral, são computados conjuntos de agrupamentos ao longo de um número limitado de iterações, sendo que a solução ideal ocorre na iteração k. Para cada vetor de características, é calculada sua máxima distância GFD (*Generalized Jaccard Distance*) até o agrupamento mais próximo, de modo a tornarem-se o centro de seus respectivos agrupamentos.

Em seguida, é necessário atualizar o mapeamento de agrupamentos mais próximos para cada vetor, o que é feito calculando-se a distância entre o centro de cada agrupamento e todos os vetores de características. Terminada a iteração k, são definidos os quadros-chave que irão compor o sumário a partir das medóides[4] dos agrupamentos. Vale ressaltar ainda que, nessa etapa, o usuário é capaz de definir parâmetros de customização, como a pré-amostragem dos quadros do vídeo, o número de quadros desejados e tempo máximo de processamento do sumário, por exemplo.

A etapa final diz respeito ao pós-processamento dos quadros-chave, ou seja, a remoção de quadros monocromáticos, considerados inexpressivos para compor o sumário do vídeo. Esta remoção é feita a partir da distribuição de cores HSV dos quadros-chave.

A metodologia de avaliação baseou-se no teste *Mean Opinion Score* (MOS) para inferir a qualidade dos sumários de vídeo gerados para o STIMO, comparando-o à outras duas abordagens de sumarização: DT [33] e k-médias [29]. Dessa forma, foi utilizado um grupo de vinte pessoas para avaliar a qualidade dos sumários a partir de notas que variam na escala de 1 (baixa qualidade) à 5 (alta qualidade). Os detalhes dessa metodologia, bem como a discussão dos resultados obtidos podem ser consultados em *"STIMO: STIll and MOving Video Storyboard for the Web Scenario"* [18].

2.5.3 VSUMM

Avila et al. [5] propuseram o VSUMM, uma nova abordagem para sumarização estática de vídeos, que baseia-se em extração de características de

[4]Elemento central em um agrupamento de amostras semelhantes, isto é, próxima à amostras consideradas ruídos ou *outliers* e que, ao mesmo tempo, reduz a média do erro quadrado no agrupamento do qual faz parte.

cor dos quadros do vídeo, classificação não supervisionada e uma metodologia subjetiva para avaliação dos sumários automáticos. A Figura 2.5 apresenta as etapas que compõe essa abordagem de sumarização.

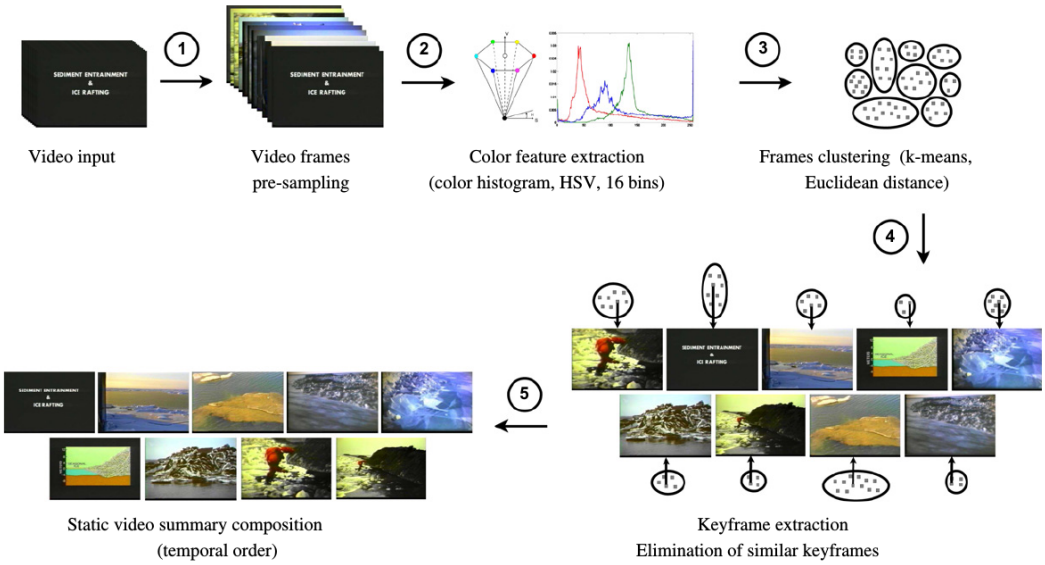


FIGURA 2.5: Funcionamento do VSUMM. Fonte: Ávila et al., 2011, p. 58.

Primeiro, é feita uma pré-amostragem dos quadros (Figura 2.5), isto é, o vídeo de entrada é segmentado com base em uma dada taxa de amostragem, que foi definida em um quadro por segundo (1 FPS). Sua finalidade é reduzir a quantidade de quadros que serão analisados nas próximas etapas [5].

Na segunda etapa, ocorre a extração de características de cor para cada um dos quadros pré-amostrados (Figura 2.5). Avila et al. [5] optaram por utilizar o espaço de cores HSV como base na geração dos histogramas de cores devido a este ser capaz de representar as cores de forma intuitiva, de modo a se aproximar da forma como o ser humano as percebe. Assim, são obtidos vetores de características para cada quadro do vídeo, os quais formam o conjunto de dados que será utilizado nas etapas seguintes.

Em seguida, os quadros inexpressivos presentes nesse conjunto são removidos segundo o desvio padrão das cores obtido de cada vetor de características (Figura 2.5). Se este valor for bastante próximo de zero, ou seja, o quadro é do tipo monocromático, então é removido do conjunto de dados.

Na quarta etapa são computados os agrupamentos de quadros semelhantes utilizando o algoritmo de aprendizado não supervisionado k-médias [29] em um versão modificada[5]. Como o número de agrupamentos k necessita ser informado *a priori*, foi calculada a distância Euclidiana entre quadros consecutivos a fim de estimar o valor de k com base na mudança brusca (pico) que pode ocorrer entre quadros não semelhantes. A descoberta de um pico representa uma mudança de tomada, dessa forma o valor de k é incrementado.

A quinta etapa do VSUMM diz respeito à seleção dos quadros mais representativos de cada agrupamento (Figura 2.5). A base dessa seleção é a identificação de agrupamentos-chave, isto é, aqueles cujo tamanho é maior do que a metade do tamanho médio dos clusters [5]. Em seguida, para cada agrupamento-chave, o quadro mais próximo de sua centróide (cálculo da distância Euclidiana entre ambos) é selecionado como quadro-chave.

Finalmente, é realizada a eliminação de quadros-chave semelhantes (Figura 2.5) pela comparação de seus histogramas de cores. Valores de comparação maiores que um dados limiar indicam que dois quadros não são semelhantes e, portanto, estes são inseridos no sumário final.

Como forma de avaliar o desempenho do VSUMM, os autores propuseram uma metodologia de avaliação chamada *CUS: Comparison Of User Summaries*, que compara sumários criados por diversos usuários àqueles gerados automaticamente por algoritmos de sumarização. Para maiores detalhes, consulte *"VSUMM: A mechanism designed to produce static video summaries and a novel evaluation method"* [5].

2.5.4 Video Summarization for Online Applications

Video Summarization for Online Applications (VISON) é a abordagem para sumarização estática de vídeos proposta por Almeida et al. [4]. Tem como objetivos principais o (1) gerenciamento de vídeos genéricos, (2) permitir ao usuários customização avançada do processo de sumarização, e (3) produção de sumários em tempo razoável e que sejam de qualidade aceitável para utilização em tempo real.

[5]Os quadros são inicialmente agrupados em ordem sequencial, diferentemente da versão original do k-médias, na qual os quadros são agrupados randomicamente.

Diferente das demais abordagens – que fazem uso de uma etapa de pré-amostragem no processo de sumarização –, o VISON atua no domínio comprimido, isto é, na forma em que geralmente um vídeo é disponibilizado para utilização. Segundo o autor, é desejável processar diretamente o vídeo sem decodificá-lo a fim de reduzir consideravelmente o custo computacional [4].

A Figura 2.6 apresenta um fluxograma para facilitar o entendimento dos processos desempenhados pelo VISON, que realiza a sumarização de um vídeo em três etapas. Primeiro, é feita a extração das características das imagens ou quadros do vídeo. Cada imagem é reduzida em blocos de 8×8 *pixels*, e que são posteriormente transformados em coeficientes DCT com o intuito de obter o termo DC de cada bloco [4]. Assim, juntando-se os termos DC dos blocos, obtemos uma imagem DC que será utilizada para geração de seu respectivo vetor de características. Nesse processo, foi utilizado o espaço de cores HSV para construção dos histogramas de cores de 256 dimensões.

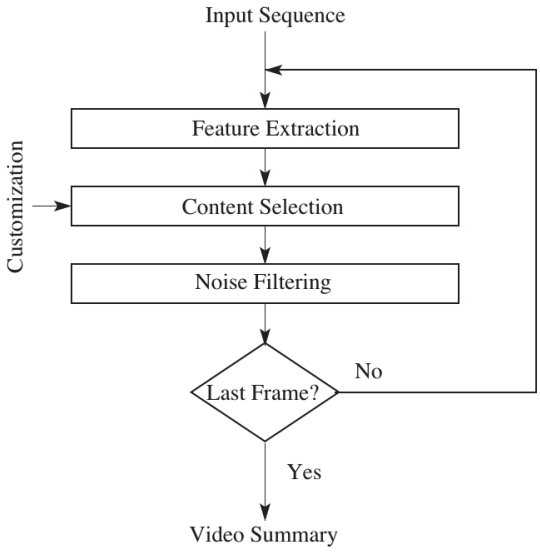


FIGURA 2.6: Representação do funcionamento do VISON na forma de fluxograma. Fonte: Almeida et al., 2012, p. 399.

A Seleção de Conteúdo refere-se à computação dos agrupamentos de quadros semelhantes. A ideia fundamental dessa etapa é segmentar o vídeo a partir da descoberta de tomadas, o que pode ser realizado por meio do cálculo da distância vetorial ZNCC entre quadros consecutivos. Se a diferença entre eles for maior do que um dado limiar, entende-se que uma tomada de vídeo

foi encontrada. Em seguida, os segmentos ou agrupamentos muito pequenos são removidos devido a sua irrelevância informativa. De cada agrupamento restante, é selecionado o quadro presente em sua posição mediana (quadro-chave).

Vale ressaltar que, durante a etapa de seleção de conteúdo, é permitido ao usuário ajustar dois parâmetros que controlam a qualidade do sumário e o tempo gasto para processá-los e criá-los. Finalmente, a Filtragem de Ruído tem como objetivo evitar a presença de quadros redundantes ou inexpressivos no sumário gerado. A remoção de um quadro redundante é feita pela comparação *pixel*-a-*pixel* entre o quadro analisado e todos os demais existentes no sumário. A similaridade entre *pixels* considera valores de intensidade para 4-vizinhos e é dada pela razão entre o número de pixels similares e o total de pixels do quadro analizado. Já um quadro inexpressivo é removido do sumário caso o valor de variância das cores de seu vetor de características for igual ou muito próximo de zero.

VISON foi avaliado utilizando-se a mesma metodologia empregada por Avila et al. [5], no entanto com a diferença de que a métrica de desempenho utilizada foi o *F-measure*, em decorrência de sua ampla aplicação no contexto de processamento de vídeos [4].

2.5.5 Conclusão

A partir das descrições das etapas de sumarização das abordagens apresentadas nas seções anteriores, foi possível perceber a similaridade estrutural apresentada por cada uma delas. Com o intuito de facilitar a compreensão, bem como a comparação entre as abordagens, apresentamos na Tabela 2.1 um pequeno resumo das etapas desempenhadas por cada abordagem de sumarização estática descrita no presente capítulo.

É possível notar a preferência na utilização de histogramas de cores de 256 dimensões no espaço de cores HSV. Isso se dá, possivelmente, pela capacidade do espaço HSV em representar as cores de forma intuitiva, isto é, próxima à percepção humana em relação às cores [5].

TABELA 2.1: Etapas de sumarização estática.

Etapas	**Abordagem de sumarização**			
	Mundur, Rao e Yesha [33]	**Furini et al. [18]**	**Ávila et al. [5]**	**Almeida et al. [4]**
Pré-amostragem	Opcional	Sim	Sim	Não
Pré-processamento	Não	Não	Sim	Não
Extração de características	Histograma de cores HSV (256-D)	Histograma de cores HSV (256-D)	Histograma de cores HSV (16-D)	Histograma de cores HSV (256-D)
Técnica de agrupamento	*Delaunay triangulation*	*Furthest Point First*	k-médias	Dissimilaridade entre pares de quadros
Função de distância	–	GFD	Euclidiana	ZNCC
Seleção de quadros-chave	Centróide	Medóide	Centróide	Centróide
Pós-processamento	Não	Sim	Sim	Sim

Outra característica comum à praticamente todas as abordagens, é a seleção de quadros-chave baseada na centróide dos agrupamentos. Naturalmente, em um agrupamento, as amostras ou pontos mais próximos ao centro do mesmo tendem a melhor representá-lo, portanto, a escolha de uma amostra muito próxima a centróide, significaria a seleção de um quadro de vídeo de alta representatividade.

Finalmente, pudemos perceber a necessidade de aplicação de um pós-processamento após desempenhada a seleção de quadros-chave. Isso ocorre para a maioria das abordagens apresentadas devido, principalmente, a possível existência de quadros de baixa expressividade no conjunto de quadros-chave (erroneamente selecionados na etapa de agrupamento). De modo geral, desempenhando essa etapa, é possível refinar de forma considerável o conjunto de quadros-chave a fim de apresentar ao usuário um sumário final mais consistente.

Capítulo 3

Floresta de Caminhos Ótimos

No presente Capítulo é descrita a teoria a respeito do classificador Floresta de Caminhos Ótimos, proposto por Rocha, Cappabianco e Falcão [39] em sua versão não supervisionada e por Papa et al. [34, 35] em sua versão supervisionada.

O OPF baseia-se na partição de um grafo para realizar o reconhecimento de padrões em bases de dados. Uma base de dados contém amostras que são representadas por vetores de características, e cada um destes, é interpretado como sendo um nó na estrutura de grafo. Além disso, os nós estão conectados entre si segundo uma relação de adjacência predefinida.

De modo geral, o algoritmo OPF é executado em duas fases. Na fase de treinamento, são encontrados os nós protótipos, isto é, nós de maior representatividade no grafo. Em seguida, esses protótipos competem entre si a fim de conquistar os nós restantes no grafo e atribuir suas respectivas classes aos nós conquistados. Como resultado, obtemos uma floresta de caminhos ótimos, ou seja, uma coleção de árvores de caminho ótimo (OPTs – *Optimum-Path Trees*) enraizadas por cada protótipo.

O reconhecimento de padrões utilizando OPF pode ser feito de modo não supervisionado ou supervisionado. Ambas as versões do classificador são descritas em maiores detalhes nas Seções 3.1 e 3.2, respectivamente.

3.1 Aprendizado não supervisionado

Seja $\mathcal{D}$ uma base de dados tal que, para toda amostra $s \in \mathcal{D}$, existe um vetor de características $\vec{v}(s)$. Seja $d(s,t)$ a distância entre s e t no espaço de

características. O problema fundamental na área de agrupamento de dados é identificar grupos de amostras em $\mathcal{D}$, sendo que amostras de um mesmo grupo deveriam apresentar algum nível de semelhança segundo algum significado semântico.

É dito que uma amostra t é adjacente a uma amostra s (isto é, $t \in A(s)$ ou $(s,t) \in A$) quando alguma relação de adjacência é satisfeita. Por exemplo,

$$t \in A_1(s) \text{ se } d(s,t) \leq d_f \text{ ou} \tag{3.1}$$

$$t \in A_2(s) \text{ se } t \text{ é } k\text{-vizinho mais próximo de } s \text{ no espaco de características,} \tag{3.2}$$

onde d_f e $k > 1$ são parâmetros do tipo real e inteiro, respectivamente. Assim sendo, o par $(\mathcal{D}, A_k)$ define então um grafo k-nn, onde A_k é uma relação de adjacência do tipo A_2 e, posteriormente, do tipo A_3 (Equação 3.4). Os arcos são ponderados por $d(s,t)$ e os nós $s \in \mathcal{D}$ são ponderados por um valor de densidade $\rho(s)$, dado por

$$\rho(s) \quad = \quad \frac{1}{\sqrt{2\pi\sigma^2}|\mathcal{A}(s)|} \sum_{\forall t \in \mathcal{A}(s)} \exp\left(\frac{-d^2(s,t)}{2\sigma^2}\right), \tag{3.3}$$

onde $\sigma = \frac{d_f}{3}$ e d_f é o comprimento do maior arco em $(\mathcal{D}, A_k)$. A escolha deste parâmetro considera todos os nós para o cálculo da densidade, assumindo que uma função gaussiana cobre a grande maioria das amostras [1] com $d(s,t) \in [0, 3\sigma]$. Note que relações de adjacência simétricas – Equação 3.1, por exemplo – resultam em relações de conectividade simétricas, entretanto A_2 na Equação 3.2 é uma relação de adjacência assimétrica.

Dado que um máximo da função densidade de probabilidade (fdp) pode ser um subconjunto de amostras adjacentes com um mesmo valor de densidade, existe a necessidade da garantia da conectividade entre qualquer par de

[1] "A regra dos três-sigmas" expressa uma convenção heurística de que três vezes a variância é capaz de cobrir aproximadamente 99.7% de uma curva Gaussiana [26].

amostras naquele máximo. Assim, qualquer amostra deste conjunto de máximos pode ser representativa e alcançar outras amostras desse máximo e suas respectivas zonas de influência por um caminho ótimo. Isto requer uma modificação na relação de adjacência A_2, para que a mesma seja simétrica nos platôs de ρ com o intuito de calcular os clusters:

$$\begin{aligned} \text{se } t &\in \mathcal{A}_2(s), \\ s &\notin \mathcal{A}_2(t) \text{ e} \\ \rho(s) &= \rho(t), \text{ então} \\ \mathcal{A}_3(t) &\leftarrow \mathcal{A}_2(t) \cup \{s\}. \end{aligned} \tag{3.4}$$

Se tivéssemos uma amostra por máximo, formando um conjunto S, então a maximização da função f_1 resolveria o problema, ou seja:

$$\begin{aligned} f_1(\langle t \rangle) &= \begin{cases} \rho(t) & \text{se } t \in S \\ -\infty & \text{caso contrário} \end{cases} \\ f_1(\pi_s \cdot \langle s,t \rangle) &= \min\{f_1(\pi_s), \rho(t)\}. \end{aligned} \tag{3.5}$$

A função f_1 possui um termo de inicialização e um termo de propagação, o qual associa a cada caminho π_t o menor valor de densidade ao longo do mesmo. Toda amostra $t \in S$ define um caminho trivial $\langle t \rangle$ devido ao fato de não ser possível alcançar t através de outro máximo da fdp sem passar através das amostras com valores de densidade menores que $\rho(t)$. As amostras restantes iniciam com caminhos triviais de valor $-\infty$, assim qualquer caminho oriundo de S possuirá valor maior. Considerando todos os caminhos possíveis de S a toda amostra $s \notin S$, o caminho ótimo $P^*(s)$ será aquele cujo menor valor de densidade seja máximo.

Visto que não temos os máximos da fdp, a função de conectividade precisa ser escolhida de tal forma que seus valores iniciais h definam os máximos relevantes da fdp. Para $f_1(\langle t \rangle) = h(t) < \rho(t), \forall t \in \mathcal{D}$, alguns máximos da fdp serão preservados e outros serão alcançados por caminhos oriundos de outros máximos, cujos valores serão maiores do que seus valores iniciais. Por exemplo,

se

$$\begin{aligned} h(t) &= \rho(t) - \delta, \\ \delta &= \min_{(s,t)\in\mathcal{A}|\rho(t)\neq\rho(s)} |\rho(t) - \rho(s)|, \end{aligned} \tag{3.6}$$

então todos os máximos de ρ serão preservados. Para altos valores de δ, os domos da fdp com altura menor que δ não definirão zonas de influência.

É desejado, também, evitar a divisão da zona de influência de um máximo em múltiplas zonas de influência, cada uma enraizada por uma amostra naquele máximo. Dado que o algoritmo do OPF primeiro identifica os máximos da fdp, antes de propagar suas zonas de influência, podemos modificá-lo de tal forma a detectar uma primeira amostra t para cada máximo, definindo o conjunto S em tempo real (*on-the-fly*). Então foi trocado $h(t)$ por $\rho(t)$ e esta amostra irá conquistar as amostras restantes do mesmo máximo. Assim, a função de conectividade f_2 final será dada por

$$\begin{aligned} f_2(\langle t\rangle) &= \begin{cases} \rho(t) & \text{se } t \in S \\ h(t) & \text{caso contrário} \end{cases} \\ f_2(\pi_s \cdot \langle s,t\rangle) &= \min\{f(\pi_s), \rho(t)\}. \end{aligned} \tag{3.7}$$

O problema agora direciona-se em encontrar o melhor valor de k para definir A_k. A solução proposta [39] para encontrar o melhor k^* considera o corte mínimo no grafo provida pelos resultados do processo de *clustering* para $k^* \in [1, k_{max}]$, de acordo com a medida $C(k)$ sugerida por [42]:

$$C(k) = \sum_{i=1}^{c} \frac{W_i'}{W_i + W_i'}, \tag{3.8}$$

$$W_i = \sum_{\forall(s,t)\in\mathcal{A}|L(s)=L(t)=i} \frac{1}{d(s,t)}, \tag{3.9}$$

$$W_i' = \sum_{\forall(s,t)\in\mathcal{A}|L(s)=i,L(t)\neq i} \frac{1}{d(s,t)}, \tag{3.10}$$

onde $L(t)$ é o rotulo da amostra t, W_i' utiliza todos os pesos dos arcos entre o *cluster* i e os demais, e W_i utiliza todos os pesos dos arcos que pertencem ao *cluster* $i = 1, 2, \ldots, c$.

3.2 Aprendizado supervisionado

Considere agora um conjunto de dados λ-rotulado $\mathcal{D} = \mathcal{D}_1 \cup \mathcal{D}_2$, em que $\mathcal{D}_1$ e $\mathcal{D}_2$ são conjuntos de treinamento e teste, respectivamente. $\mathcal{S} \subset \mathcal{D}_1$ é o conjunto contendo os protótipos de todas as classes, ou seja, amostras-chave que melhor representam as classes, e $(\mathcal{D}_1, A)$ é um grafo completo cujos nós são as amostras de $\mathcal{D}_1$ e os arcos $A = \mathcal{D}_1 \times \mathcal{D}_1$ são definidos por qualquer par de amostras e ponderados segundo o valor de distância desse par (nós correspondentes). Além disso, π_s é um caminho em $(\mathcal{D}_1, A)$ que possui seu término na amostra $s \in D_1$.

O algoritmo OPF proposto por Papa et al. [34, 35] faz uso da função de custo de caminho f_{max} devido as suas propriedades teóricas para estimar os protótipos (Seção 3.2.1 descreve esse procedimento):

$$\begin{aligned} f_{max}(\langle s \rangle) &= \begin{cases} 0 & \text{se } s \in S \\ +\infty & \text{caso contrário,} \end{cases} \\ f_{max}(\pi_s \cdot \langle s, t \rangle) &= \max\{f_{max}(\pi_s), d(s,t)\}, \end{aligned} \tag{3.11}$$

onde $d(s,t)$ representa a distância entre os nós s e t, sendo que $s, t \in \mathcal{D}_1$. Por conseguinte, $f_{max}(\pi_s)$ calcula a máxima distância entre amostras adjacentes em π_s quando este não é um caminho trivial. De modo geral, o OPF tenta minimizar a função $f_{max}(\pi_t)$, $\forall t \in \mathcal{D}_1$.

3.2.1 Treinamento

Considere $\mathcal{S}^*$ um conjunto ótimo de protótipos quando o algoritmo OPF é capaz de minimizar os erros de classificação para toda amostra $s \in \mathcal{D}_1$. S^* pode ser encontrado explorando a relação teórica entre a árvore de espalhamento mínimo (*Minimum Spanning Tree* – MST) e a árvore de caminho ótimo para f_{max} [1].

A fase de treinamento consiste em encontrar S^* e um classificador OPF enraizado em S^*. Pelo cálculo da MST no grafo completo $(\mathcal{D}_1, A)$, obtém-se um grafo conectado acíclico cujos nós são todas as amostras de $\mathcal{D}_1$ e os arcos são unidirecionais e ponderados pelas distâncias entre amostras adjacentes. Na MST, todo par de amostras está conectado por um único caminho ótimo segundo f_{max}. Consequentemente, a árvore de espalhamento mínimo é composta por uma árvore de caminho ótimo para qualquer nó raiz selecionado.

Os protótipos ótimos são os elementos mais próximos da MST com rótulos diferentes em $\mathcal{D}_1$, isto é, elementos que permeiam a fronteira das classes. Pela remoção dos arcos entre classes diferentes, seus nós adjacentes tornam-se protótipos em S^* e, portanto, o algoritmo OPF é capaz de definir uma floresta de caminhos ótimos com erros mínimos de classificação em $\mathcal{D}_1$.

3.2.2 Classificação

Para cada amostra $t \in \mathcal{D}_2$, consideram-se todos os arcos que possuem conexões t com amostras $s \in \mathcal{D}_1$, como se t fosse parte do grafo de treinamento. Considerando todos os possíveis caminhos de S^* à t, o caminho ótimo $P^*(t)$ de S^* e rótulo t com a classe $\lambda(R(t))$ é aquele que possui o protótipo $R(t) \in S^*$ mais fortemente conectado. Esse caminho pode ser identificado de forma incremental pela avaliação do custo ótimo $C(t)$, que é definido pela equação seguinte:

$$C(t) = \min\{\max\{C(s), d(s,t)\}\},\ \forall s \in D_1. \tag{3.12}$$

Seja o nó $s^* \in \mathcal{D}_1$ aquele que satisfaz a Equação 3.12, ou seja, o predecessor $P(t)$ no caminho ótimo $P^*(t)$. Dado que $L(s^*) = \lambda(R(t))$, a fase de classificação simplesmente tem por finalidade atribuir $L(s^*)$ como sendo a classe de t. Vale ressaltar que ocorre um erro quando $L(s^*) \neq \lambda(t)$.

Capítulo 4

Sumarização estática de vídeos utilizando Agrupamento de Dados por Floresta de Caminhos Ótimos

Neste Capítulo, descrevemos a abordagem para sumarização estática de vídeos utilizando agrupamento por Floresta de Caminhos Ótimos. A partir dos estudos iniciais já realizados por Martins et al. [30] e apresentados no *19th Iberoamerican Congress on Pattern Recognition* (CIARP 2014), incluímos novas etapas à metodologia de sumarização com o intuito de aprimorar o processo de geração dos *storyboards* levando em conta as informações espaciais e temporais existentes em um vídeo digital, sendo estas modificações apresentadas em um artigo recentemente submetido ao *IEEE International Conference on Image Processing (2016)*. Além disso, as discussões e resultados alcançados a partir dos experimentos realizados também são apresentados no presente capítulo.

4.1 Metodologia

Organizamos o funcionamento da abordagem de sumarização em seis etapas básicas: (1) pré-amostragem do vídeo, (2) extração de características, (3) pré-processamento, (4) agrupamento e seleção de quadros-chave, (5) pós-processamento e (6) geração do sumário. A Figura 4.1 apresenta a estrutura da abordagem proposta para sumarização, a qual é descrita em maiores detalhes nas seções seguintes.

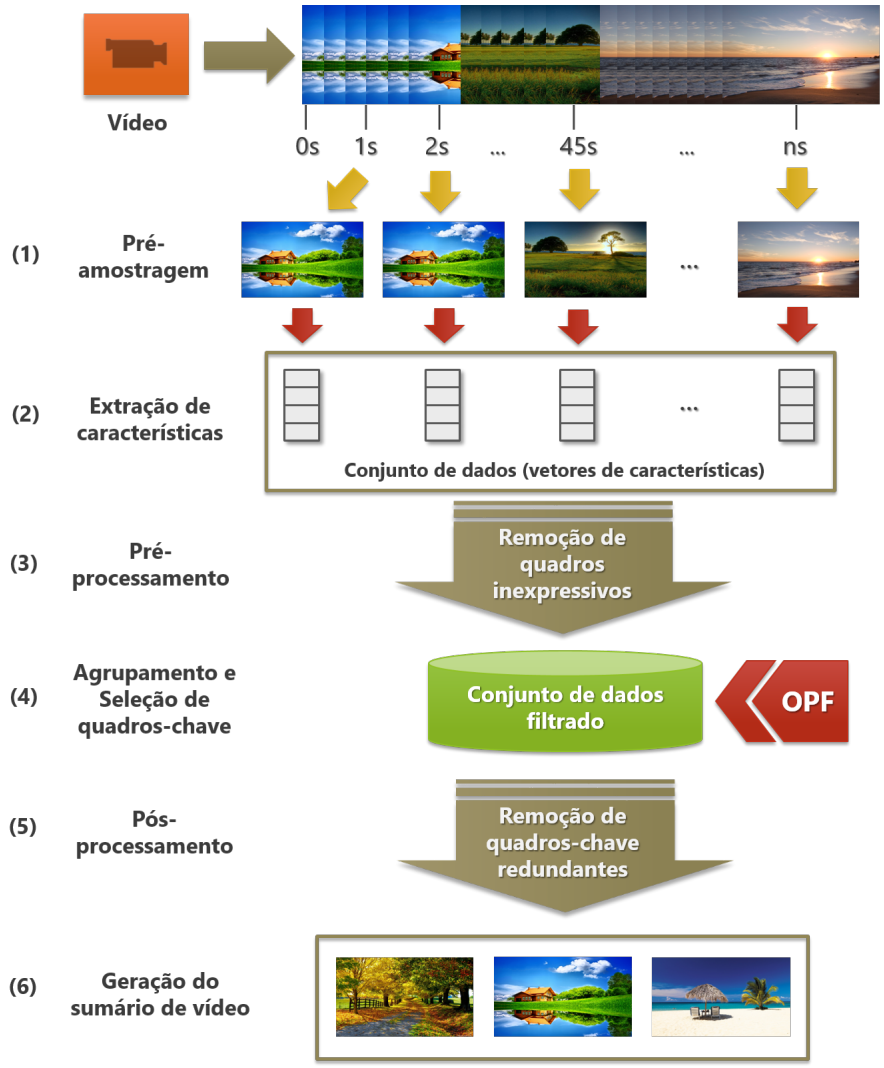


FIGURA 4.1: Abordagem proposta para sumarização estática de vídeos.

4.1.1 Pré-amostragem do vídeo

Inicialmente, precisamos decodificar o vídeo para obtenção de seus quadros,s os quais serão posteriormente analisados durante o processo de construção do sumário.

Para desempenho desta tarefa, utilizamos a ferramenta *ffmpeg*[1]. Configuramos a ferramenta de forma que a decodificação ocorresse a uma taxa de um

[1]http://www.ffmpeg.org

quadro por segundo (1 FPS) a fim de evitar a obtenção de um número exagerado de quadros ao final da pré-amostragem do vídeo, o que significaria um volume de informação redundante a ser processado. Dessa forma, foi possível reduzir a carga de processamento computacional nessa etapa.

4.1.2 Extração de características

Na segunda etapa, realizamos a extração das características de cada quadro do vídeo. Para essa tarefa, consideramos dois descritores para codificar as informações de cores dos quadros, GCH e CCV, os quais foram aplicados às bases Open Video e Youtube, respectivamente. Essa escolha tem por base os resultados obtidos a partir dos estudos feitos por [30], que constataram o comportamento distinto desses dois descritores para as bases de vídeos já citadas.

Obtemos como resultado desta etapa duas bases de dados baseadas em características, e que podem ser processadas nas etapas subsequentes.

4.1.3 Pré-processamento

Em seguida, removemos os quadros inexpressivos da base de dados obtida na etapa anterior com a finalidade de evitar o processamento desnecessário destes quadros durante o processo de agrupamento. Um quadro é dito inexpressivo quando este é formado por uma única cor – por exemplo, preto ou branco – devido a efeitos de *fade-in* ou *fade-out* presentes. Para realizar a remoção deste tipo de quadro, consideramos o valor de variância de suas cores. Caso este valor seja igual a zero, então o referido quadro é removido da base de dados baseada em características.

4.1.4 Agrupamento e seleção de quadros-chave

Na quarta etapa, aplicamos a versão não supervisionada do OPF à base de dados baseada em características com a finalidade de encontrar os quadros mais representativos quadros-chave em cada agrupamento computado.

Como o OPF seleciona os protótipos nas regiões de alta densidade, estes tendem a estar localizados nos centros de seus respectivos agrupamentos.

Dessa forma, os protótipos são ótimos candidatos a serem escolhidos como quadros-chave. Entretanto, o problema principal relacionado à abordagem proposta por Martins et al. [30] está no fato de que OPF foi aplicado em toda a base de dados, isto é, quadros espacialmente similares estão associados ao mesmo agrupamento. Todavia, estes não necessariamente possuem alguma inter-relação temporal. A fim de evitar a perda deste tipo de informação durante a computação dos agrupamentos, procuramos realizar o particionamento do conjunto de dados em n subconjuntos menores como forma de preservar, tanto a relação espacial, quanto temporal entre os quadros.

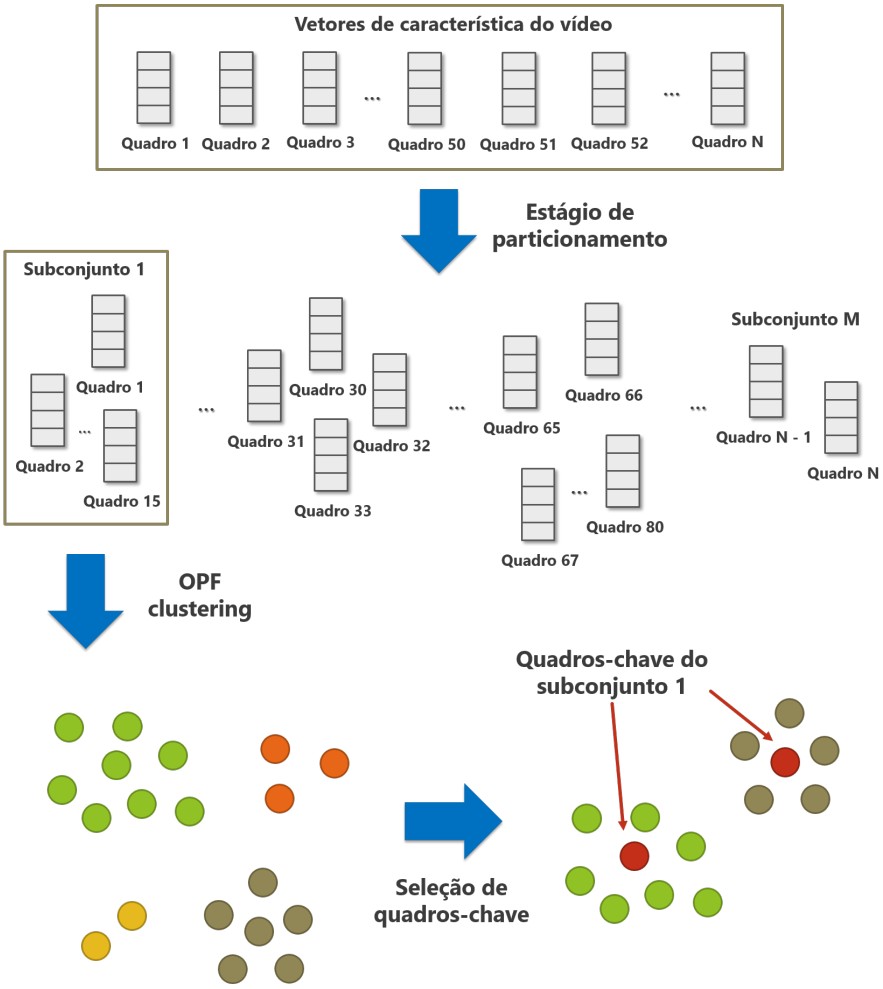


FIGURA 4.2: Particionamento e agrupamento por OPF.

Dessa maneira, sendo o OPF executado em cada subconjunto, podemos acelerar o processo de agrupamento do OPF por meio de múltiplas execuções

paralelas, dado que a fase de aprendizado do algoritmo ocorre de modo independente. O processo pode ser visualizado em maiores detalhes na Figura 4.2. Além disso, propomos uma melhoria relacionada ao mecanismo utilizado pelo OPF para cálculo da "distância"entre quadros. Desenvolvemos uma função capaz de considerar mais informação do que apenas o conteúdo espacial codificado a partir da distância Euclidiana entre os vetores de características de cada quadro, como apresentado por Martins et al.[30].

A função proposta é formada por dois termos: o primeiro relaciona-se à informação temporal T_{ij} entre os quadros i e j, enquanto o segundo está relacionado à informação espacial S_{ij} entre estes. O termo temporal é representado na Equação 4.1:

$$T_{ij} = |p_i - p_j|\,, \tag{4.1}$$

onde p_i e p_j denotam as posições normalizadas dos quadros i e j, respectivamente. Vale ressaltar que a posição de cada quadro denota sua localização cronológico no vídeo. Portanto, a posição normalizada é dada pela razão entre o número do quadro e o total quadros do vídeo. Assim, o termo espacial é dado por:

$$S_{ij} = \frac{d(i,j)}{d_{max}}, \tag{4.2}$$

onde $d(i,j)$ é a distância Euclidiana entre os quadros i e j, e d_{max} representa a máxima distância Euclidiana entre dois quadros distintos. Finalmente, a função de distância proposta é dado pela Equação 4.3:

$$D_{ij} = S_{ij} + \alpha T_{ij}, \tag{4.3}$$

onde α é o termo que pondera a quantidade de informação temporal considerada durante o cálculo final da distância.[2]

Uma consequência decorrente da execução da etapa de agrupamento considerando subconjuntos de dados é que poderemos obter agrupamentos com um número muito reduzido de quadros. Isto significa que agrupamentos pequenos podem não contribuir com informações relevantes para a geração do

[2]Note que esse procedimento é aplicado em cada subconjunto de dados.

sumário. A fim de removê-los, calculamos o tamanho médio dos agrupamentos em cada subconjunto de dados, mantendo apenas os agrupamentos-chave, ou seja, aqueles cujo tamanho é maior do que metade do tamanho médio dos agrupamentos [5].

Em seguida, para cada agrupamentos chave selecionamos apenas um quadro-chave, sendo este o protótipo de seu respectivo agrupamento.

4.1.5 Pós-processamento

A quinta etapa é responsável pela filtragem do conjunto de quadros-chave, selecionando apenas aqueles que não sejam redundantes. O processo de remoção de um quadro-chave redundante ocorre da seguinte forma: cada quadro-chave é comparado aos demais utilizando a distância Euclidiana. Se a distância resultante for menor do que 0.15[3], o quadro-chave é considerado redundante e, portanto, é removido do conjuntos de quadros-chave.

4.1.6 Geração do sumário

Realizado o pós-processamento, o conjunto resultante tem seus quadros-chave organizados em ordem cronológica para serem apresentados na forma de *storyboard*. Finalmente, o sumário obtido pode ser utilizado com próposito de ser comparado àqueles gerados por outras abordagens.

4.2 Bases de dados

A obtenção de análises mais sólidas quanto a comparação de resultados entre diferentes abordagens de sumarização pode depender de uma escolha adequada da base de dados, pois esta pode ter grande impacto quanto a validação dos desempenhos obtidos. Dessa forma, escolhemos duas bases públicas de vídeos[4]: *Open Video Project* (OV)[5] e *Youtube*.

[3] Valor definido a partir de testes empíricos.
[4] http://sites.google.com/site/vsummsite
[5] http://www.open-video.org/

A base de dados *Open Video* é composta por 50 vídeos distribuídos em 3 categorias: Documentário (44), Educacional (2) e Palestra (4). Todas as sequências de vídeo foram codificadas no formato MPEG-1 contendo áudio e quadros coloridos, e possuem duração que varia de 1 a 4 minutos. Já a base de dados Youtube foi construída por Ávila et al. [5], por meio da coleta de 40 vídeos de web sites diversos como, por exemplo, Youtube. Os vídeos possuem duração que varia de 1 a 10 minutos e estão distribuídos em 5 categorias: Esportes, Notícias, Comerciais, Vídeos Caseiros, e Programas de Auditório.

4.3 Avaliação

No contexto de sumarização de vídeos, ainda não existe uma plataforma ou metodologia consistente de avaliação de sumários. Desse modo, ao invés de construir um método de avaliação pouco consistente, optamos por utilizar a abordagem *CUS: Comparison of User Summaries* [5]. Os objetivos fundamentais dessa metodologia são: reduzir a subjetividade existente na tarefa de avaliação, quantificar a qualidade dos sumários de vídeo, e permitir a comparação entre diferentes abordagens de sumarização.

CUS funciona da seguinte forma: inicialmente, o usuário assiste ao vídeo e seleciona, a seu critério, um conjunto de quadros que melhor representam esse vídeo. Não são definidos limites mínimos e máximos de quadros para composição do sumários, portanto, o tamanho deste pode ser diferente para cada usuário. Um grupo de pessoas é escolhido para realizar essa tarefa, e os sumários de cada vídeo da base construídos por elas forma o *ground-truth* de avaliação. Em seguida, os sumários de usuários são comparados aos *storyboards* gerados pelas abordagens de sumarização automática.

Como a comparação entre abordagens ocorre com base nos quadros presentes nos sumários, estes podem ser avaliados por meio das métricas Precisão e Revocação. A razão entre o número de quadros correspondentes e o total de quadros no sumário automático representa Precisão, enquanto que Revocação é dada pela razão entre o número de quadros correspondentes e o total de quadros no sumários do usuário. Uma combinação dessas medidas dá origem ao *F-measure*, definido na Equação 4.4:

$$F = \frac{2 * (Precisão * Revocação)}{Precisão + Revocação}. \quad (4.4)$$

Um ponto negativo relacionado à utilização das métricas citadas, está no fato de que altos valores de precisão conduzem a baixos valores de revocação, e vice-versa [4]. A fim de evitar esse comportamento, avaliamos os sumários automáticos pela métrica *F-measure* (Equação 4.4), que combina precisão e revocação em uma única medida por meio de média harmônica [6].

4.4 Discussão e Resultados

Na presente seção descrevemos os resultados obtidos a partir dos diversos experimentos desempenhados para avaliação de *performance* da abordagem de sumarização proposta. Ainda, as discussões acerca dos resultados obtidos e comportamentos apresentados pelo OPF também são descritos na presente seção.

4.4.1 Definição do parâmetro k_{max}

Primeiramente, antes de iniciar as comparações de desempenho do OPF com as demais abordagens de sumarização, realizamos uma análise do comportamento de *F-measure* com relação ao parâmetro k_{max}, valor este responsável por determinar o limite máximo de k-vizinhos-mais-próximos para cálculo dos agrupamentos. A fim de descobrir o valor de k_{max} que máxima o resultado da medida *F-measure*, executamos OPF para cada descritor visual variando k_{max} de 5 até 50 em intervalos de 5. As Figuras 4.3 e 4.4 apresentam os valores de *F-measure* obtidos para diferentes configurações do parâmetro k_{max} nas bases Open Video e Youtube, respectivamente.

A partir dos resultados iniciais obtidos, é possível notar que, para todos os descritores, conforme incrementava-se k_{max}, o valor de *F-measure* decaía. Essa ocorrência apontou uma característica interessante decorrente do comportamento desse parâmetro: valores muito altos de k_{max} significam que serão computados poucos agrupamentos e, consequentemente, ao final do processo, serão gerados poucos quadros-chave. Em contrapartida, utilizando um baixo

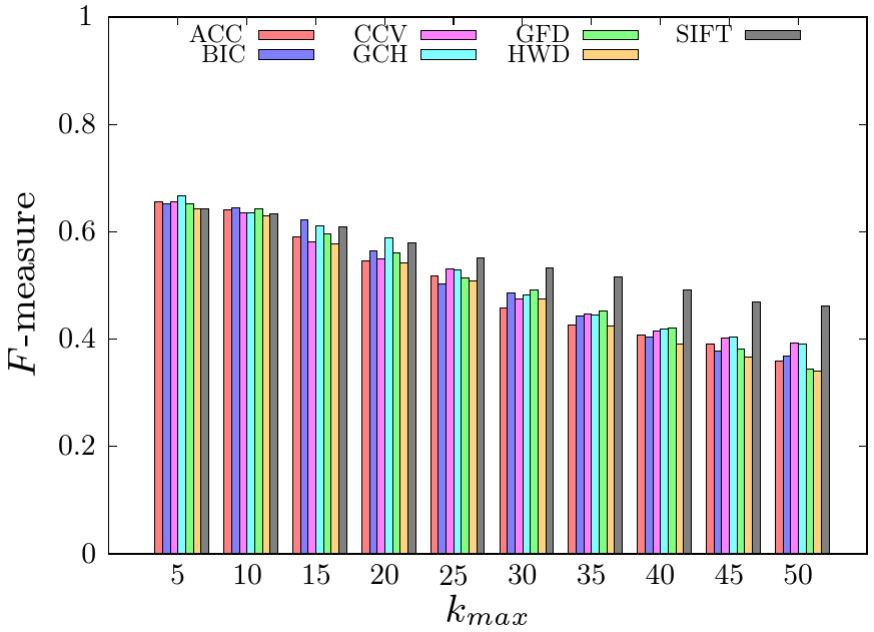


FIGURA 4.3: Valores de *F-measure* obtidos pela variação de k_{max} na base Open Video.

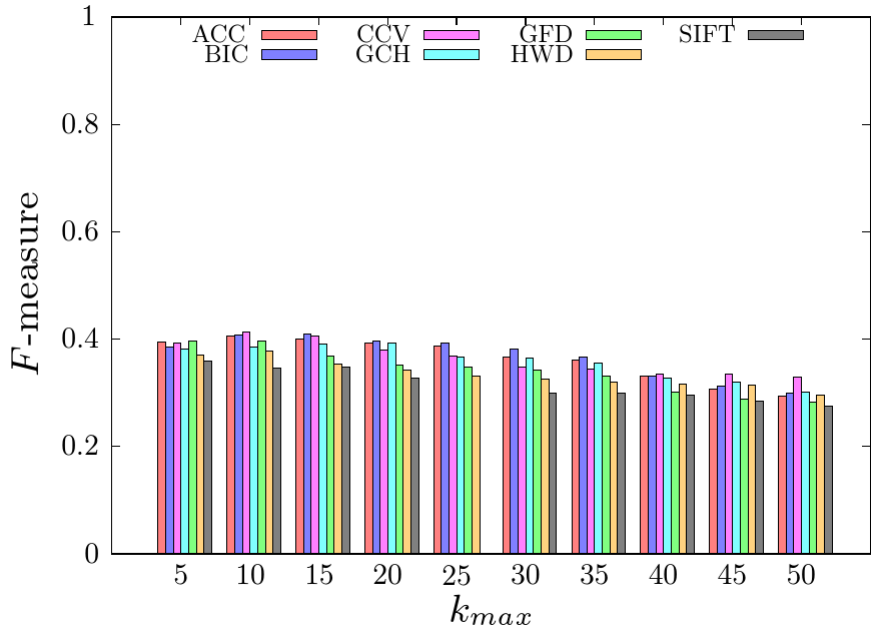


FIGURA 4.4: Valores de *F-measure* obtidos pela variação de k_{max} na base Youtube.

valor de k_{max}, obtemos tanto um número maior de agrupamentos quanto de quadros-chave. Dessa forma, notamos que valores baixos de k_{max} resultam em desempenhos melhores para o OPF.

Foi possível observar ainda que, para o mesmo valor de k_{max}, a variação de desempenho entre os sete descritores considerados foi muito pequena. A única exceção foi BoF utilizando SIFT, que mostrou ser o descritor mais estável com relação à mudança de k_{max}. Para ambas as bases de vídeos, os melhores resultados foram obtidos utilizando-se os descritores de cor GCH e CCV, respectivamente. Além disso, os valores ideais para o parâmetro do OPF foram $k_{max} = 5$ e $k_{max} = 10$.

4.4.2 Comparação do OPF à outras abordagens de sumarização

Utilizando as configurações ideais para o parâmetro k_{max}, reportadas no parágrafo anterior, comparamos os sumários gerados pelo OPF aos sumários obtidos das demais técnicas descritas na Seção 2.5. Nos testes feitos com a base de vídeos *Open Video Project*, OPF foi comparado a DT, STIMO, VISON, VSUMM e OV. Esta última abordagem está presente no site *Open Video Project* e os sumários foram construídos a partir do algoritmo proposto por DeMenthon et al. [16].

A partir da Figura 4.5, podemos observar que o desempenho do OPF, considerando a base de vídeos *Open Video Project* e os três gêneros de vídeo apresentados (Documentário, Educacional e Palestra), foi o terceiro melhor com *F-Measure* $= 0.667$, sendo inferior apenas às técnicas VISON e VSUMM. É importante destacar que os resultados obtidos pelo OPF são bastante promissores devido à simplicidade de sua metodologia[6], isto é, não foram consideradas etapas para refinamento do sumário final, como foi feito pelas abordagens VISON e VSUMM, por exemplo.

[6]É válido ser ressaltado que a metodologia apresentada na Seção 4.1 diz respeito às melhorias implementadas na abordagem de sumarização para refinamento dos *storyboards* gerados. Dessa maneira, em relação a estes experimentos, consideramos três etapas básicas: pré-amostragem do vídeo, extração de características e agrupamentos de quadros semelhantes. Para maiores detalhes quanto a metodologia de sumarização mencionada, consulte [30].

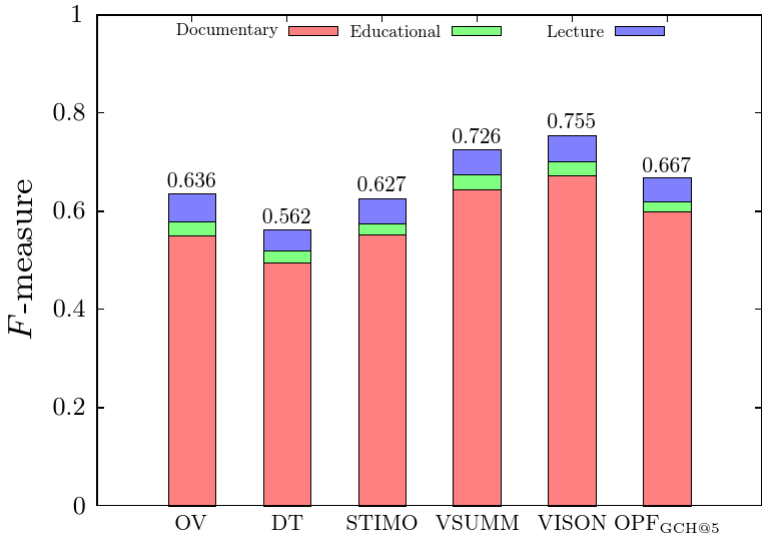


FIGURA 4.5: Desempenho por gêneros de vídeo na base Open Video.

Com relação à base de vídeos Youtube, Figura 4.6, o classificador OPF foi comparado apenas à VISON e VSUMM. As demais abordagens não foram consideradas nessa comparação devido a ausência de experimentos e resultados para essa base de vídeos. Mais uma vez, o desempenho do OPF foi ligeiramente inferior ao das duas outras abordagens, apresentando *F-Measure* $= 0.414$. É bastante perceptível que o desempenho geral de todas as abordagens é bastante inferior em comparação aos experimentos realizados com a base de vídeos *Open Video Project*. Isso deve-se, provavelmente, às características dos vídeos da base *Youtube* como, por exemplo, duração das sequências de vídeo (1 a 10 minutos) e maior variedade de gêneros.

É digno de nota que, uma das principais características do OPF na tarefa de sumarização de vídeos é o uso de apenas um parâmetro para controle do tamanho e qualidade dos sumários. Em aplicações que exijam processos de sumarização mais automatizados, ou seja, com menor interação humana possível, o uso do OPF destaca-se nesse contexto.

Diante destes resultados iniciais, apresentamos agora os novos experimentos desempenhados considerando a metodologia descrita na Seção 4.1 que, em poucas palavras, visa preservar tanto a informação espacial quanto temporal dos quadros de vídeo durante o processo de sumarização.

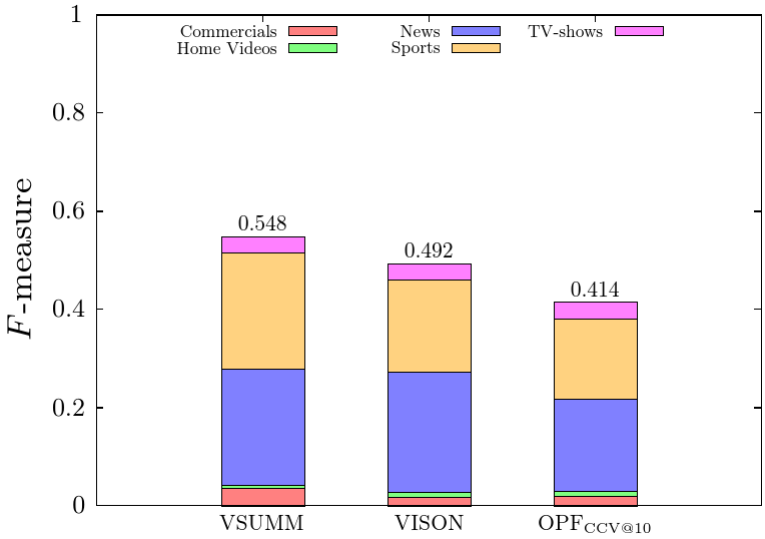


FIGURA 4.6: Desempenho por gêneros de vídeo na base Youtube.

4.4.3 Redefinição de k_{max}

A título de comparação, avaliamos os resultados obtidos pela abordagem proposta, a qual chamaremos de **OPF***, nas duas bases públicas de vídeos descritas neste trabalho: Open Video e Youtube. Na primeira, OPF* foi comparado aos resultados obtidos por OPF [30], OV, DT, STIMO, VSUMM e VISON. Já na segunda, OPF* foi comparado apenas a OPF, VSUMM e VSION, dado que as outras abordagens de sumarização não possuem resultados reportados com relação a essa base de vídeos.

Como já mencionado, OPF computa os agrupamentos em tempo real tendo como base a variável k_{max}[7], responsável por determinar o número máximo de vizinhos mais próximos a serem considerados durante a computação desses agrupamentos. É digno de nota que, mudanças no valor de k_{max} causam um impacto menor na computação dos agrupamentos em comparação à variação do parâmetro k do algoritmo k-médias.

Com o objetivo de determinar o melhor valor para o parâmetro k_{max} em cada subconjunto de dados, utilizamos a seguinte metodologia: realizamos testes experimentais com diferentes percentuais de tamanhos de subconjuntos e, para cada um destes, avaliamos $k_{max} \in [5, 50]$ em passos de 5. As Figuras 4.7 e

[7] Capítulo 3, Seção 3.1.

4.8 apresentam os valores de F-*measure* obtidos a partir desses experimentos para as duas bases de vídeos já citadas[8]. Finalmente, selecionamos o tamanho de subconjunto e o valor de k_{max} que juntos maximizam F-*measure*, ou seja, 25% e $k_{max} = 5$.

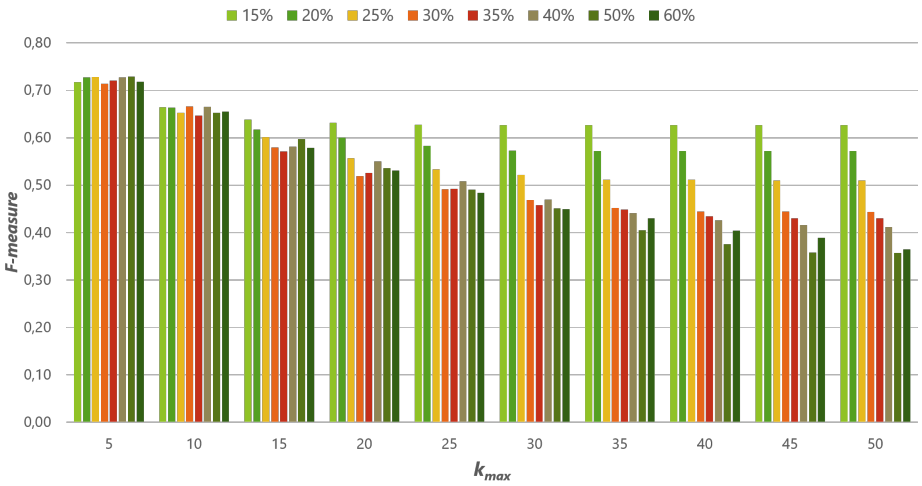


FIGURA 4.7: Avaliação de k_{max} por tamanho de subconjuntos na base Open Video.

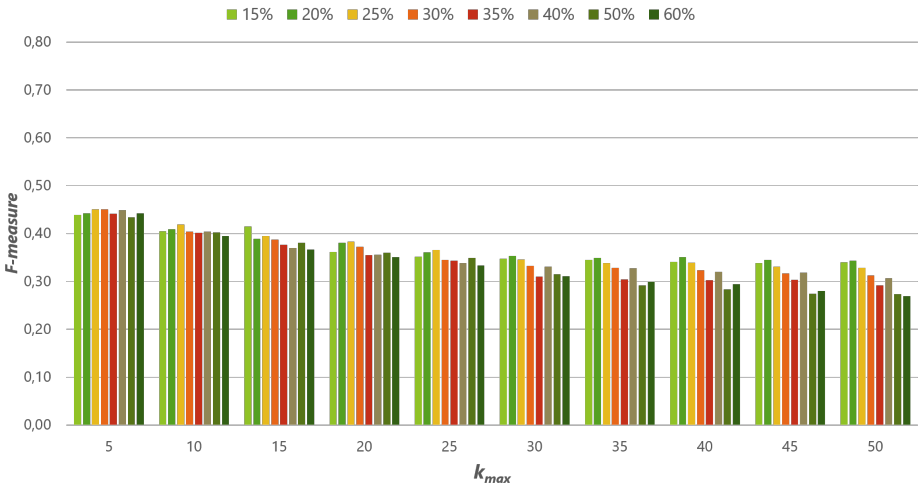


FIGURA 4.8: Avaliação de k_{max} por tamanho de subconjuntos na base Youtube.

[8]Vale relembrar que utilizamos os descritores GCH e CCV nas bases Open Video e Youtube, respectivamente

Notamos que o OPF* se comportou melhor utilizando subconjuntos menores, já que subconjuntos maiores não foram favoráveis à preservação da informação temporal dos vídeos. Além disso, a definição de $\alpha = 0.86$ funcionou bem em ambas as bases de vídeos. Vale ressaltar que, em nosso experimentos, observamos que valores de α muito pequenos não contribuíram para melhoria dos resultado finais.

4.4.4 Comparação do OPF* a outras abordagens de sumarização

Utilizamos as configurações avaliadas anteriormente para comparar o OPF* às abordagens de sumarização e analisar seu desempenho. Os resultados comparativos com relação aos valores de F-measure obtidos em cada categoria de vídeo nas bases Open Video e Youtube são apresentados nas Figuras 4.9 e 4.10, respectivamente. Claramente, o OPF* obteve resultados mais precisos em comparação ao OPF em ambas as bases. Além disso, atingiu o segundo melhor F-*measure* na base Open Video, e terceiro melhor desempenho considerando a base Youtube.

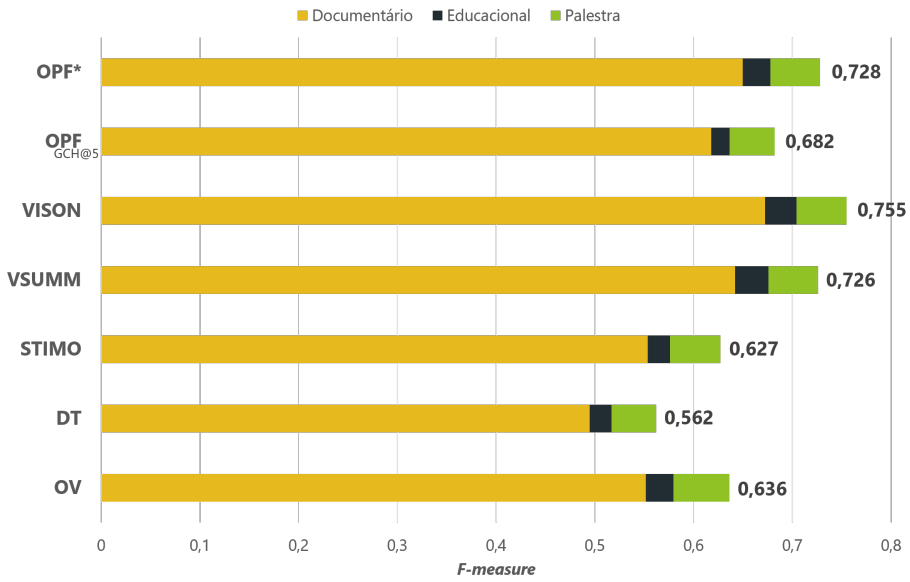


FIGURA 4.9: *F-measure* médio atingido pelas abordagens de sumarização em cada categoria de vídeo na base Open Video.

Todavia, a melhor abordagem na base Youtube utiliza o algoritmo k-médias para computação dos agrupamentos, portanto, exigindo que o número de agrupamentos seja pré-definido. No que diz respeito ao OPF, esse tipo de informação não é uma dificuldade.

Fazendo um comparativo com VISON, o OPF exige menos interação humana para geração de um sumário, o que não ocorre com o primeira abordagem, já que esta possui alguns parâmetros de inicialização devido a sua característica em permitir ao usuário customizar a geração do sumário de vídeo.

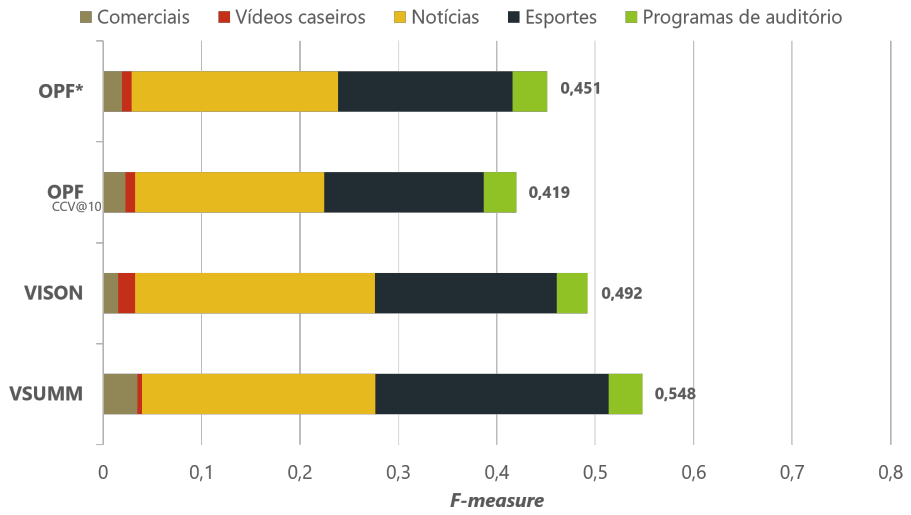


FIGURA 4.10: *F-measure* médio atingido pelas abordagens de sumarização em cada categoria de vídeo na base Youtube.

Com relação à base Open Video, OPF* obteve desempenho superior ao OPF considerando os vídeos das categorias Documentário e Educacional. A lógica por trás disso está no fato de que vídeos do tipo Educacional contém quadros similares localizados em posições temporais distintas no vídeo. Para melhor explicar essa situação, considere o seguinte exemplo: imagine algum palestrante ministrando uma aula sobre um assunto qualquer e, num dado momento, o interlocutor passa a ilustrar esse assunto, de forma que ele perde o foco da câmera. Em seguida, o palestra ganha o foco novamente, e continua a explanar sobre o assunto. Nesta situação, os quadros de vídeo são espacialmente similares, todavia, estão presentes em posições temporais distintas no vídeo.

Quanto à base de vídeos Youtube, a principal melhoria do OPF* está relacionada aos vídeos esportivos. Isso ocorreu devido ao mesmo comportamento relatado no exemplo anterior. Como utilizamos descritores de cor para geração do vetores de características dos quadros de vídeo, é bem provável que deste ponto de vista, diferentes partidas de futebol sejam consideradas similares, por exemplo. Mais um vez, ressaltamos o importante papel da informação temporal nessas situações.

Por fim, disponibilizamos alguns sumários obtidos a partir dos experimentos relatados neste capítulo, que estão presentes no Apêndice A deste trabalho.

Capítulo 5

Classificação supervisionada de vídeos por Floresta de Caminhos Ótimos

O presente Capítulo apresenta a descrição da abordagem de classificação de vídeos baseada em gênero apresentada no artigo intitulado *Supervised Video Genre Classification Using Optimum-Path Forest*, publicado recentemente no *20th Iberoamerican Congress on Pattern Recognition* [31].

O principal objetivo deste trabalho foi introduzir o algoritmo Floresta de Caminhos Ótimos no contexto de classificação de vídeos por gênero, assim como comparar seu desempenho a alguns dos algoritmos de reconhecimento de padrões comumente encontrados na literatura.

5.1 Metodologia

Estruturamos a abordagem proposta em duas etapas básicas. Primeiro, é feita a extração de características do vídeo de entrada utilizando três técnicas para codificação de suas propriedades visuais[1]: *Bag-of-Visual-Words*, *Bag-of-Scenes*, e *Histogram of Motion Patterns* (HMP).

Na segunda etapa, classificamos o vídeo amostrado utilizando o algoritmo Floresta de Caminhos Ótimos e, em seguida, o mesmo procedimento é realizado para os seguintes classificadores: Redes Neurais Artificiais com *Perceptron* Multicamada (ANN-MLP) [20], k-Vizinhos Mais Próximos (k-NN) [13], e

[1] Capítulo 2, Seção 2.4.

Máquinas de Vetores de Suporte (SVM) [9]. Vale ressaltar que foram utilizados *kernels* polinomal (SVM-POLY) e de base radial (SVM-RBF) para o classificador SVM, e os parâmetros de inicialização do algoritmo foram otimizados por meio de validação cruzada. Em relação ao classificador k-NN, o valor ideal de k foi obtido por meio da minimização da taxa de erro desse classificador utilizando diferentes valores[2] de k.

TABELA 5.1: Configurações adotadas para testes experimentais.

Teste	**Descritor**	**Configuração**		
		Assignment	***Pooling***	**Características**
1	BoVW	*Hard*	*Average*	1000 palavras visuais
2	BoS	*Hard*	*Average*	1000 cenas
3	BoS	*Hard*	*Max*	1000 cenas
4	BoS	*Soft*	*Average*	1000 cenas
5	BoS	*Soft*	*Max*	1000 cenas
6	BoS	*Hard*	*Average*	100 cenas
7	BoS	*Hard*	*Max*	100 cenas
8	BoS	*Soft* ($\sigma = 1$)	*Average*	100 cenas
9	BoS	*Soft* ($\sigma = 1$)	*Max*	100 cenas
10	BoS	*Soft* ($\sigma = 2$)	*Average*	100 cenas
11	BoS	*Soft* ($\sigma = 2$)	*Max*	100 cenas
12	HMP	–	–	6075 padrões de movimento

Definimos doze experimentos envolvendo diferentes configurações de codificação das características visuais. Todos os experimentos foram aplicados a cada um dos classificadores considerados nessa metodologia. A Tabela 5.1 apresenta mais detalhadamente as configurações escolhidas para os testes experimentais.

[2]Foram utilizados apenas valores ímpares de forma a evitar empates durante o processo de classificação das amostras.

5.2 Base de dados

Escolhemos uma base de dados de *benchmarking* fornecida pelos organizadores da *MediaEval 2012* para a Tarefa de Marcação de Gêneros (*Genre Tagging Task*) [41]. Esta base base é composta por 14.838 vídeos divididos em conjuntos de desenvolvimento (5.288 vídeos) e teste (9.550 vídeos).

As sequências de vídeo foram coletadas em maio de 2015 do site blip.tv[3], e estão distribuídas em 26 categorias de gênero, as quais foram atribuídas pela plataforma de mídia blip.tv. A seguir, são apresentadas as categorias de gênero e seus respectivos números totais de vídeos: Artes (530), Automóveis e Veículos (21), Negócios (281), Jornalismo Cidadão (401), Comédia (515), Conferência e Outros Eventos (247), Documentário (353), Educacional (957), Comida e Bebida (261), Jogos (401), Saúde (268), Literatura (222), Filmes e Televisão (868), Música e Entretenimento (1148), Pessoal ou Auto-biográfico (165), Política (1107), Religião (868), Escola e Educação (171), Esportes (672), Tecnologia (1343), Meio Ambiente (188), Grande Mídia (324), Viagem (175), *Videoblogging* (887), Desenvolvimento Web e Sites (116), e Categoria Padrão (2349)[4].

O principal desafio relacionado ao uso dessa base é a presença de uma alta diversidade de gêneros de vídeo, além da existência de uma grande variedade de conteúdos visuais dentro de cada categoria de gênero.

5.3 Discussão e Resultados

Seguindo o mesmo objetivo de avaliação definido para sumarização estática, optamos por utilizar Precisão e Revocação, combinando essas duas medidas em uma única métrica de desempenho. Para isso, selecionamos *Mean Average Precision* (MAP) como métrica de avaliação dos resultados de classificação de vídeos, a qual é representada pela equação 5.1:

$$MAP = \frac{\sum_{q=1}^{Q} AveP(q)}{Q}, \tag{5.1}$$

[3] http://blip.tv.

[4] Vídeos que não puderam ser atribuídos a nenhuma das demais categorias de gênero.

onde o total de classes é dado por Q, $AveP(q)$ representa a média de precisão para a classe q e MAP é a pontuação média obtida em todas as classes consideradas. Segundo Almeida et al. [3], MAP é um ótimo indicador de eficácia considerando todas as posições das listas de classificação obtidas.

Adicionalmente, fizemos uso de uma taxa de reconhecimento proposta por Papa et al. [34], capaz de considerar dados não balanceados, isto é, quando há classes com tamanhos distintos de amostras. A métrica *Accuracy* é dada pela equação 5.2, apresentada a seguir:

$$Acc = 1 - \frac{\sum_{i=1}^{c} E(i)}{2c}, \tag{5.2}$$

onde c é o total de classes presente no conjunto de dados e $E(i)$ representa a soma parcial[5] do erro de classificação para a classe i.

Ainda, calculamos a carga computacional (tempo de processamento) necessária para executar ambas as fases de treinamento e classificação dos algoritmos avaliados. As Figuras 5.1 e 5.2 apresentam os resultados considerando as métricas de avaliação MAP e *Accuracy*, respectivamente.

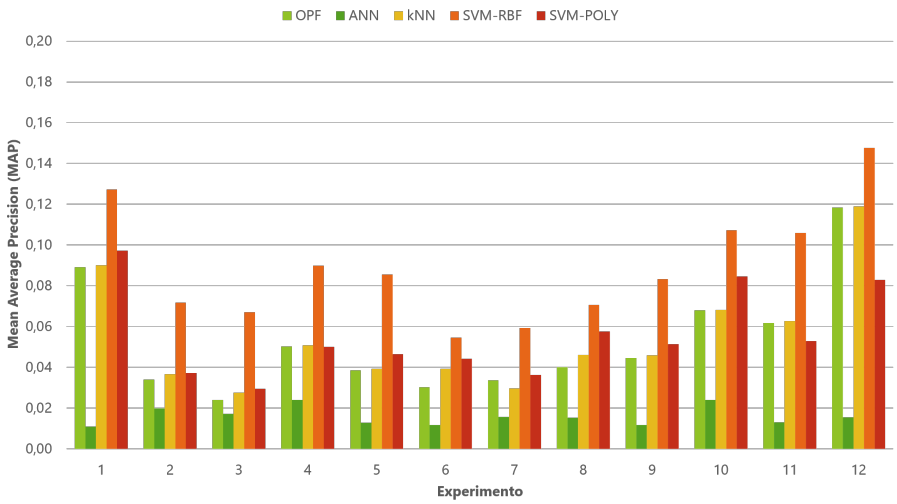


FIGURA 5.1: Resultados de reconhecimento para MAP.

Considerando ambas as medidas de performance, o experimento número 12 apresentou os melhores resultados para todos os classificadores, exceto

[5] $E(i)$ é formado pelos erros obtidos a partir dos falsos positivos e falsos negativos para a classe i.

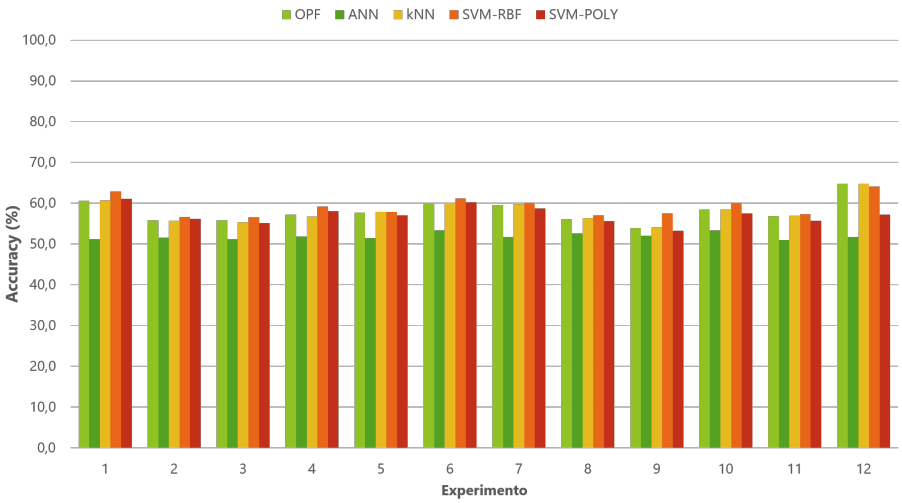


FIGURA 5.2: Resultados de reconhecimento para *Accuracy*.

para o classificador ANN-MLP. O descritor HMP obteve grande destaque nessa análise por ser robusto em relação a diversas transformações, além de ser bastante apropriado para uso em grandes coleções de vídeos [2], o que está de acordo com a base de vídeos *MediaEval 2012*, utilizada nesses experimentos. Já com relação à medida MAP, OPF se estabeleceu em segundo ou terceiro lugares na maioria dos casos, enquanto que para a medida *Accuracy* OPF obteve a primeira ou segunda posição em grande parte dos experimentos.

Avaliando a carga computacional nas etapas de treinamento e classificação, apresentadas, respectivamente, nas Figuras 5.3 e 5.4, é possível observar que OPF foi o classificador mais rápido na etapa de treinamento para quase todos os experimentos, assim como foi o segundo mais rápido considerando o tempo de classificação.

A partir desses resultados, é possível concluir que o OPF pode ser adequado para desempenho da classificação de vídeos baseada em informação visual, já que este classificador obteve boas taxas de reconhecimento em uma pequeno espaço de tempo de execução quando comparado às demais técnicas (com exceção da ANN-MLP). Isso demonstra que o uso do OPF em aplicações de classificação em tempo real e sistemas de recomendação pode ser de grande interesse, visto que uma alta relação custo/benefício entre efetividade e eficiência é algo extremamente desejado neste contexto.

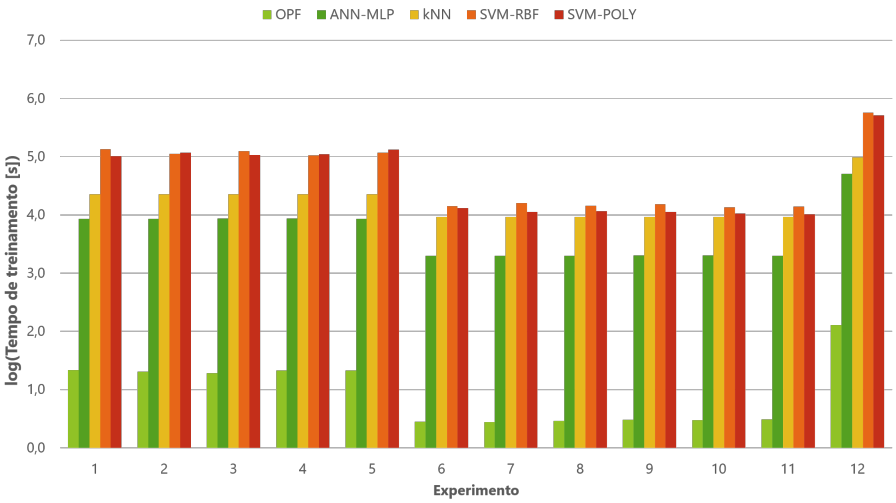


FIGURA 5.3: Carga computacional: etapa de treinamento.

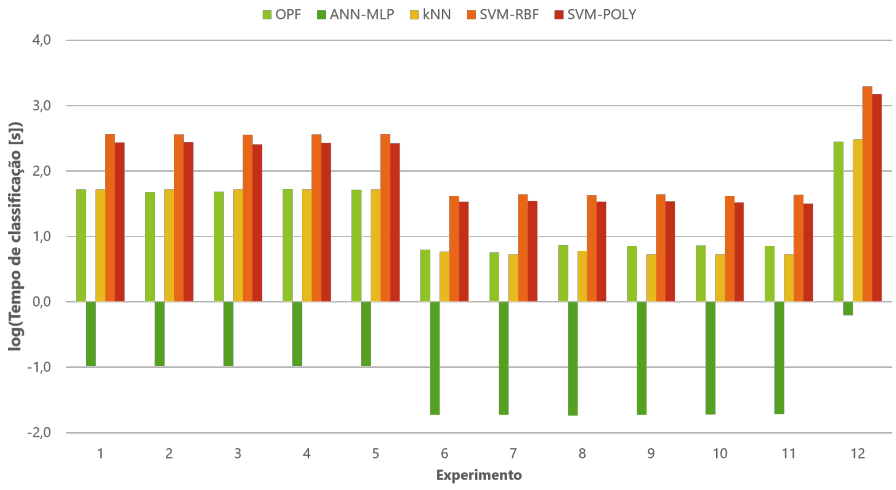


FIGURA 5.4: Carga computacional: etapa de classificação.

Capítulo 6

Considerações finais

No presente trabalho, foi apresentado o estudo do classificador Floresta de Caminhos Ótimos para utilização deste nas tarefas de sumarização estática e de classificação de vídeos baseada em gêneros. Técnicas de sumarização também foram estudadas a fim de melhor compreender o contexto e possibilitar um planejamento mais consistente na adequação do algoritmo OPF em versão não supervisionada para criação qualitativa e eficiente de sumários de vídeos.

A partir dos estudos e resultados apresentados, notamos que a tarefa crítica do processo de sumarização estática é a segmentação do vídeo durante a etapa de agrupamento dos quadros, que basicamente ocorre com a detecção de tomadas de vídeo. De modo geral, encontrando as tomadas que compõem o vídeo, podemos extrair de cada uma destas, um quadro de maior relevância, utilizado para composição do *storyboard*.

Assim, uma das análises realizadas demonstrou a relação direta entre a qualidade dos sumários gerados pelo OPF e a adequada escolha do parâmetro k_{max}, que controla o limite máximo de k-vizinhos mais próximos para cálculo dos agrupamentos e, consequentemente, influenciando o número final de quadros-chave considerados na composição do sumário do vídeo.

Além disso, outro elemento importante, também ligado à etapa de agrupamento, é a detecção de agrupamentos muito pequenos. Um agrupamento que possua um número muito reduzido de quadros denota informações de baixa relevância no vídeo e, portanto, não é necessário incluir no sumário um quadro-chave que o represente.

Levando em conta que a definição de um valor baixo de k-vizinhos-mais-próximos eleva a quantidade de agrupamentos computados e reduz o número

de quadros em cada um destes, e vice-versa, realizar a filtragem de agrupamentos muito pequenos foi uma escolha adequada considerando $k_{max} = 5$.

A partir dos resultados alcançados por Martins et al. [30], notamos que, para o OPF, as informações de cor melhor descrevem os quadros dos vídeos, quando comparadas às propriedades espectrais[1] obtidas dos mesmos.

O desempenho do OPF considerando informações espaço-temporais foi bastante satisfatório, já que, comparado aos resultados iniciais [30], a abordagem obteve a segunda melhor colocação de *performance* na base Open Video, e terceira na base Youtube. Assim, o uso do classificador OPF mostrou-se bastante adequado para desempenho de sumarização estática de vídeos, considerando os experimentos desempenhados e apresentados neste trabalho.

Quanto à classificação de vídeos baseada em gêneros, as análises realizadas tiveram por objetivo inicial avaliar a eficiência e eficácia do algoritmo OPF no contexto atual fazendo uso de descritores de vídeo, o que foi comprovado a partir dos resultados obtidos e apresentados em [31]. O classificador OPF apresentou taxas de reconhecimento consistentes, tanto para a análise da medida MAP, quanto para *Accuracy*, em todos os experimentos desempenhados.

Ainda, podemos destacar o desempenho superior obtido pela maioria das técnicas de classificação quando utilizado o descritor HMP (experimento 12). A capacidade de preservar informações espaço-temporais por meio de padrões de movimento mostrou-se evidente nos experimentos realizados para as métricas MAP e *Accuracy*. Diferentemente dos descritores BoVW e BoS, HMP é livre de parâmetros de inicialização, o que o torna, dessa maneira, uma escolha adequada para representação de vídeos considerando sistemas que atuam em tempo real e de forma automatizada.

Comparado às demais técnicas avaliadas, em especial o classificador SVM, OPF demanda uma baixa carga computacional, o que pôde ser evidenciado por meio dos tempos de processamento obtidos tanto na etapa de treinamento, quanto na de classificação.

Mesmo que os resultados obtidos sejam reflexo de experimentos iniciais em classificação de vídeos baseada em gênero, e que, portanto, introduzem nossa abordagem neste contexto, o desempenho alcançado pelo classificador

[1] Vetores de características obtidos a partir do descritor HWD (*Haar Wavelet Decomposition*) [25], por exemplo

OPF aponta-o como uma alternativa viável para sistemas de recomendação e recuperação de vídeos em tempo real, onde o equilíbrio entre eficiência e eficácia é altamente desejado.

6.1 Trabalhos Futuros

Naturalmente, o foco atual está em estudar novas formas para aprimoramento do OPF em sumarização automática e classificação de vídeos. Assim, nos atentaremos para as seguintes possibilidades de melhoria:

- Otimização do processo de sumarização desempenhado pelo OPF para cada gênero de vídeo;
- Utilização de diferentes taxas de pré-amostragem para sumarização automática;
- Implementação do OPF para sumarização dinâmica de vídeos;
- Combinação de descritores para classificação de vídeos;
- Possibilidade de criação de descritores visuais a partir de sumários automáticos, e posterior aplicação destes em classificação de vídeos e;
- Estudar a viabilidade de construção de uma plataforma completa para processamento de vídeos utilizando o classificador Floresta de Caminhos Ótimos.

6.2 Publicações

- **G. B. Martins et al**. ***Static Video Sumarization through Optimum-Path Forest***. Em: *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications: 19th Iberoamerican Congress, CIARP 2014, Puerto Vallarta, Mexico, November 2-5, 2014. Proceedings*. *Springer International Publishing, 2014*. *Springer International Publishing*, 2014, pp 893-900.

- **G. B. Martins, J. Almeida e J. P. Papa**. ***"Supervised Video Genre Classification Using Optimum-Path Forest"***. Em: *"Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications: 19th Iberoamerican Congress, CIARP 2015, Montevideo, Uruguay, November 9-12, 2015. Proceedings"*. Em: *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications. Springer,* 2015, pp. 735-742.

Apêndice A

Sumários de vídeo

Neste material suplementar, apresentamos alguns exemplos de *storyboards* gerados pelas abordagens de sumarização considerados no presente trabalho. Na Figura A.1 podemos visualizar sumários automáticos gerados pelas diversas abordagens apresentadas neste trabalho, e referem-se a um vídeo da base *Open Video*. Já as Figuras A.2 e A.3 apresentam sumários de usuários comparados visualmente aos obtidos por diferentes abordagens de sumarização automática, e estão presentes na base de vídeos *Youtube*.

(a) OV: F-measure = 0.417

(b) DT: F-measure = 0.268

(c) STIMO: F-measure = 0.526

(d) VSUMM: F-measure = 0.715

(e) VISION: F-measure = 0.873

(f) OPF$_{GCH@5}$: F-measure = 0.708

(g) OPF*: F-measure = 0.878

FIGURA A.1: Sumários obtidos por diferentes abordagens considerando o vídeo *"A New Horizon, segment 2"*.

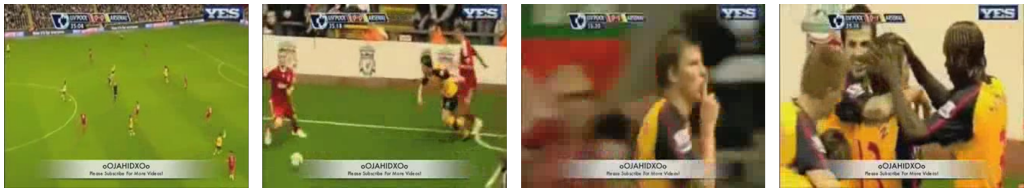
(a) Usuário #4

(b) VSUMM

(c) VISON

(d) OPF*

FIGURA A.2: Um sumário de usuário (a) e três sumários automáticos obtidos por VSUMM (b), VISON (c) e OPF* (d) de um vídeo da categoria “comercial”.



(a) Usuário #2

(b) OPF$_{CCV@10}$

(c) OPF*

FIGURA A.3: Um sumário de usuário (a) e sumários automáticos obtidos a partir de OPF$_{CCV@10}$ (b) e OPF* (c) de um vídeo da categoria “esportes”.

Referências bibliográficas

[1] C. Allène et al. «Some Links Between Extremum Spanning Forests, Watersheds and Min-cuts». Em: *Image Vision Computing* 28.10 (2010), pp. 1460–1471. ISSN: 0262-8856.

[2] J. Almeida, N. J. Leite e R. S. Torres. «Comparison of Video Sequences with Histograms of Motion Patterns». Em: *ICIP*. 2011, pp. 3673–3676.

[3] J. Almeida, D. C. G. Pedronette e O. A. B. Penatti. «Unsupervised Manifold Learning for Video Genre Retrieval». Em: *CIARP*. 2014, pp. 604–612.

[4] J. Almeida, R.S. Torres e N.J. Leite. «VISON: VIdeo Summarization for ONline applications». Em: *Pattern Recognition Letters* 33.4 (2012), pp. 397–409.

[5] S.E.F. Avila et al. «VSUMM: A mechanism designed to produce static video summaries and a novel evaluation method». Em: *Pattern Recognition Letters* 32.1 (2011), pp. 56–68.

[6] Henk M Blanken et al. *Multimedia retrieval*. Springer, 2007.

[7] Y.-L. Boureau et al. «Learning Mid-Level Features for Recognition». Em: *CVPR*. 2010, pp. 2559–2566.

[8] D. Brezeale e D.J. Cook. «Automatic video classification: A survey of the literature». Em: *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews* 38.3 (2008), pp. 416–430.

[9] Christopher J. C. Burges. «A Tutorial on Support Vector Machines for Pattern Recognition». Em: *Data Min. Knowl. Discov.* 2.2 (jun. de 1998), pp. 121–167. ISSN: 1384-5810. DOI: 10.1023/A:1009715923555. URL: http://dx.doi.org/10.1023/A:1009715923555.

[10] E. J. Y. C. Cahuina. «A New Method for Static Video Summarization Using Visual Words and Video Temporal Segmentation». Tese de mestrado. Federal University of Ouro Preto, 2013.

[11] L. Capodiferro et al. «SVM for historical sport video classification.» Em: *Proceedings of 5th International Symposium on the Communications Control and Signal Processing*. 2012, pp. 1–4.

[12] V. Chasanis, A. Likas e N. Galatsanos. «Video rushes summarization using spectral clustering and sequence alignment». Em: *Proceedings of the 2nd ACM TRECVid Video Summarization Workshop*. New York, NY, USA, 2008, pp. 75–79.

[13] T. Cover e P. Hart. «Nearest Neighbor Pattern Classification». Em: *IEEE Trans. Inf. Theor.* 13.1 (set. de 2006), pp. 21–27. ISSN: 0018-9448. DOI: 10.1109/TIT.1967.1053964. URL: http://dx.doi.org/10.1109/TIT.1967.1053964.

[14] C. Cramer, E. Gelenbe e I. Bakircioglu. «Video compression with random neural networks». Em: *Proceedings of the International Workshop on Neural Networks for Identification, Control, Robotics, and Signal/Image Processing*. 1996, pp. 476–484.

[15] T.O. Cunha et al. «Rushes video summarization based on spatio-temporal features». Em: *Proceedings of the ACM Symposium on Applied Computing*. ACM, 2012, pp. 45–50.

[16] D. DeMenthon, V. Kobla e D.S. Doermann. «Video Summarization by Curve Simplification». Em: *ACM Int. Conf. Multimedia (MM'08)*. 1998, pp. 211–218.

[17] E. Dumont e B. Mérialdo. «Rushes video summarization and evaluation.» Em: *Multimedia Tools Applications* 48.1 (2010), pp. 51–68.

[18] M. Furini et al. «STIMO: STIll and MOving video storyboard for the web scenario». Em: *Multimedia Tools Appl.* 46.1 (2010), pp. 47–69.

[19] Teofilo F. Gonzalez. «Clustering to minimize the maximum intercluster distance». Em: *Theoretical Computer Science* 38 (1985), pp. 293 –306. ISSN: 0304-3975. DOI: http://dx.doi.org/10.1016/0304-3975(85)90224-5. URL: http://www.sciencedirect.com/science/article/pii/0304397585902245.

[20] Mohamad H. Hassoun. *Fundamentals of Artificial Neural Networks*. 1st. Cambridge, MA, USA: MIT Press, 1995. ISBN: 026208239X.

[21] Dorit S. Hochbaum e David B. Shmoys. «A Best Possible Heuristic for the k-Center Problem». Em: *Mathematics of Operations Research* 10.2 (1985), pp. 180–184. DOI: 10.1287/moor.10.2.180. eprint: http://dx.doi.org/10.1287/moor.10.2.180. URL: http://dx.doi.org/10.1287/moor.10.2.180.

[22] W. Hu et al. «A survey on visual content-based video indexing and retrieval». Em: *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews* 41.6 (2011), pp. 797–819.

[23] J. Huang et al. «Image Indexing Using Color Correlograms». Em: *IEEE Int. Conf. Computer Vision and Pattern Recognition (CVPR'97)*. 1997, pp. 762–768.

[24] J. Y. Kim, D.C. Park e D.M. Woo. «Classification of video data using centroid neural network.» Em: *Proceedings of the IEEE International Symposium on Signal Processing and Information Technology*. 2009, pp. 408–411.

[25] C.E. Jacobs, A. Finkelstein e D. Salesin. «Fast Multiresolution Image Querying». Em: *Int. Conf. Computer Graphics and Interactive Techniques (SIGGRAPH'95)*. 1995, pp. 277–286.

[26] L. Kazmier. *Schaum's Outline of Business Statistics*. Schaum's Outline Series, p. 359. ISBN: 9780071430999.

[27] N.X. Lian e Y.P. Tan. «Probabilistic approach to k-nearest neighbour video retrieval». Em: *Proceedings of the International Symposium on Circuits and Systems*. Vol. 2. 2004, pp. 193–196.

[28] Z. Liu, Y. Wang e T. Chen. «Audio feature extraction and analysis for scene segmentation and classification». Em: *VLSI Signal Processing Systems* 20.1–2 (1998), pp. 61–79.

[29] J. MacQueen. «Some methods for classification and analysis of multivariate observations». Em: *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics*. Berkeley, Calif.: University of California Press, 1967, pp. 281–297. URL: http://projecteuclid.org/euclid.bsmsp/1200512992.

[30] G. B. Martins et al. «Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications: 19th Iberoamerican Congress, CIARP 2014, Puerto Vallarta, Mexico, November 2-5, 2014. Proceedings». Em: ed. por Eduardo Bayro-Corrochano e Edwin Hancock. Cham: Springer International Publishing, 2014. Cap. Static Video Summarization through Optimum-Path Forest Clustering, pp. 893–900. ISBN: 978-3-319-12568-8. DOI: 10.1007/978-3-319-12568-8_108. URL: http://dx.doi.org/10.1007/978-3-319-12568-8_108.

[31] Guilherme B Martins, Jurandy Almeida e Joao Paulo Papa. «Supervised Video Genre Classification Using Optimum-Path Forest». Em: *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*. Springer, 2015, pp. 735–742.

[32] A. G. Money e H. W. Agius. «Video summarization: A conceptual framework and survey of the state of the art». Em: *J. Visual Communication and Image Representation* 19.2 (2008), pp. 121–143.

[33] P. Mundur, Y. Rao e Y. Yesha. «Keyframe-based video summarization using Delaunay clustering». Em: *Int. J. on Digital Libraries* 6.2 (2006), pp. 219–232.

[34] J.P. Papa, A.X. Falcão e C. T. N. Suzuki. «Supervised pattern classification based on optimum-path forest». Em: *International Journal of Imaging Systems and Technology* 19.2 (2009), pp. 120–131.

[35] J.P. Papa et al. «Efficient supervised optimum-path forest classification for large datasets». Em: *Pattern Recognition* 45.1 (2012), pp. 512–520.

[36] G. Pass, R. Zabih e J. Miller. «Comparing Images Using Color Coherence Vectors». Em: *ACM Int. Conf. Multimedia (ACM-MM'96)*. 1996, pp. 65–73.

[37] O. A. B. Penatti et al. «A Visual Approach for Video Geocoding using Bag-of-Scenes». Em: *ICMR*. 2012, pp. 1–8.

[38] O.A.B. Penatti, E. Valle e R.S. Torres. «Comparative Study of Global Color and Texture Descriptors for Web Image Retrieval». Em: *Journal of Visual Communication and Image Representation* 23.2 (2012), pp. 359–380.

[39] L.M. Rocha, F.A.M. Cappabianco e A.X. Falcão. «Data clustering as an optimum-path forest problem with applications in image analysis». Em: *International Journal of Imaging Systems and Technology* 19.2 (2009), pp. 50–68.

[40] Emile Sahouria e Avideh Zakhor. «Content analysis of video using principal components». Em: *Circuits and Systems for Video Technology, IEEE Transactions on* 9.8 (1999), pp. 1290–1298.

[41] S. Schmiedeke, C. Kofler e I. Ferrané. «Overview of MediaEval 2012 Genre Tagging Task». Em: *MediaEval*. 2012.

[42] J. Shi e J. Malik. «Normalized cuts and image segmentation». Em: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22.8 (2000), pp. 888–905.

[43] A.F. Smeaton et al. «Content-based video retrieval: Three example systems from TRECVid». Em: *International Journal of Imaging Systems and Technology* 18.2–3 (2008), pp. 195–201.

[44] A. Sony et al. «Video summarization by clustering using euclidean distance». Em: *Proceedings of the International Conference on Signal Processing, Communication, Computing and Networking Technologies*. 2011, pp. 642–646.

[45] R.O. Stehling, M.A. Nascimento e A.X. Falcão. «A Compact and Efficient Image Retrieval Approach based on Border/Interior Pixel Classification». Em: *ACM Int. Conf. Information and Knowledge Management (CIKM'02)*. 2002, pp. 102–109.

[46] M.J. Swain e B.H. Ballard. «Color Indexing». Em: *Int. J. Computer Vision* 7.1 (1991), pp. 11–32.

[47] B. T. Truong e S. Venkatesh. «Video abstraction: A systematic review and classification». Em: *ACM Trans. Multimedia Comput. Commun. Appl.* 3.1 (2007), pp. 1–37.

[48] P.H. Tsai, J. Seun e H. Gunes. «Video object encoder using selective local-space support vector machines». Em: *Proceedings of the IEEE 6th Workshop on Multimedia Signal Processing*. 2004, pp. 427–429.

[49] L.Q. Xu e Y. Li. «Video classification using spatial-temporal features and PCA». Em: *Proceedings of the International Conference on Multimedia and Expo*. Vol. 3. 2003, pp. 485–488.

[50] G. Yazbek, C. Mokbel e G. Chollet. «Video segmentation and compression using hierarchies of gaussian mixture models». Em: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*. Vol. 1. 2007, pp. 1009–1012.

[51] J. You, G. Liu e A. Perkis. «A semantic framework for video genre classification and event analysis». Em: *Signal Processing: Image Communication* 25.4 (2010), pp. 287–302. ISSN: 0923-5965.

[52] M. Zampoglou, T. Papadimitriou e K.I. Diamantaras. «Support vector machines content-based video retrieval based solely on motion information». Em: *Proceedings of the IEEE Workshop on Machine Learning for Signal Processing*. 2007, pp. 176–180.

[53] D. Zhang e G. Lu. «Shape-based Image Retrieval using Generic Fourier Descriptor». Em: *Signal Processing: Image Communication* 17.10 (2002), pp. 825–848.