**Fusão de informação multimodal por detecção de correlação
para tarefas de análise de vídeo**

Rodrigo Mitsuo Kishi

Tese de Doutorado do Programa de Pós-Graduação em Ciências de
Computação e Matemática Computacional (PPG-CCMC)

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: _____

Rodrigo Mitsuo Kishi

Fusão de informação multimodal por detecção de correlação para tarefas de análise de vídeo

Tese apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP, como parte dos requisitos para obtenção do título de Doutor em Ciências – Ciências de Computação e Matemática Computacional. *VERSÃO REVISADA*

Área de Concentração: Ciências de Computação e Matemática Computacional

Orientador: Prof. Dr. Rudinei Goularte

USP – São Carlos
Maio de 2020

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados inseridos pelo(a) autor(a)

K46f Kishi, Rodrigo Mitsuo
Fusão de informação multimodal por detecção de
correlação para tarefas de análise de vídeo / Rodrigo
Mitsuo Kishi; orientador Rudinei Goularte. -- São
Carlos, 2020.
165 p.

Tese (Doutorado - Programa de Pós-Graduação em
Ciências de Computação e Matemática Computacional) --
Instituto de Ciências Matemáticas e de Computação,
Universidade de São Paulo, 2020.

1. multimodalidade. 2. fusão prévia. 3. análise
de co-ocorrência. 4. multimídia. 5. vídeo digital. I.
Goularte, Rudinei, orient. II. Título.

Rodrigo Mitsuo Kishi

**Multimodal information fusion by correlation detection for
video analysis tasks**

Thesis submitted to the Institute of Mathematics and
Computer Sciences – ICMC-USP – in accordance with
the requirements of the Computer and Mathematical
Sciences Graduate Program, for the degree of
Doctor in Science. *FINAL VERSION*

Concentration Area: Computer Science and
Computational Mathematics

Advisor: Prof. Dr. Rudinei Goularte

USP – São Carlos
May 2020

Este trabalho é dedicado aos meus pais, Ester e Orlando.

AGRADECIMENTOS

Devo agradecer, principalmente, aos meus pais, Ester e Orlando, pelo apoio e incentivo durante toda a minha trajetória pessoal, profissional e acadêmica. Sem o seu amor e apoio eu não teria suportado alguns infortúnios que marcaram meu caminho. Mesmo estando fisicamente só, eu nunca me senti sozinho. Agradeço também à minha esposa Aline pelo apoio, compreensão e parceria durante o período de intersecção entre o início da nossa relação e os anos finais do curso de doutorado (em especial nos momentos finais). Você tornou tudo menos difícil e mais feliz.

O percurso do doutorado é longo e nesse tempo muitos colaboraram nos mais variados aspectos para que eu pudesse alcançar o fim. Agradeço, inicialmente, ao meu amigo Ricardo Marcacini por ter me ajudado a escolher e ingressar no Instituto de Ciências Matemáticas e de Computação (ICMC) da Universidade de São Paulo (USP). Agradeço também aos meus companheiros de trabalho na Universidade Federal de Mato Grosso do Sul (UFMS) que cuidaram com muita competência nosso tão querido curso durante o período no qual estive afastado. Um agradecimento especial aos professores Ivone Yugoshi, Juliano Yugoshi, Ronaldo Fiorilo e Maxwell Santos, com os quais pude contar em muitas ocasiões nos últimos anos.

Agradeço à Ivone e ao Juliano pela amizade, apoio e companheirismo durante a minha estadia na cidade de São Carlos, tendo me apoiado em momentos difíceis. É alentador saber que posso contar com vocês em qualquer ocasião.

Eu agradeço a todos os integrantes do Laboratório Intermídia com os quais compartilhei, diariamente, esses longos anos de duração do doutorado. Em especial, agradeço aos amigos Tiago Trojahn, Humberto Antonelli, Arthur Fortes, Felipe Maciel, Sandra Rodrigues, Marcio Funes, Felipe Giuntini, Alef Vinicius, Erikson Aguiar, Rafael D'Addio, Omar Mozo, Kifayat Ullah, André Salgado, Heitor Freitas, Flávia Santos, Raoni Ferreira, Thiago Costa e Kleberson Serique pelos bons momentos dentro e fora do laboratório.

Agradeço individualmente ao meu amigo e companheiro de pesquisas Tiago Trojahn, por ter compartilhado seu valioso conhecimento em análise de vídeo digital no início do doutorado e pelas incontáveis contribuições e auxílios em experimentos, escritas, dentre outros. Você foi fundamental para o desenvolvimento do meu trabalho.

Agradeço aos professores e funcionários do ICMC/USP pelo excelente serviço prestado. Destaco um agradecimento ao Professor Gustavo Batista e ao Denis Reis, ambos

do Laboratório de Inteligência Computacional, pela ajuda com a tarefa de classificação de vídeos.

Agradeço à UFMS por viabilizar meu afastamento durante o doutorado.

Agradeço ao Centro de Ciências Matemáticas Aplicadas à Indústria - USP pela disponibilização de recursos computacionais que facilitaram as atividades experimentais do meu projeto.

Por fim, agradeço especialmente ao meu orientador, que considero também um amigo, Rudinei Goularte, por todo o cuidado dedicado à orientação do meu projeto e pelas conversas sobre os mais variados assuntos. Agradeço fortemente seu apoio, paciência, compreensão e confiança no meu trabalho. Admiro sua conduta como professor e também como ser humano.

RESUMO

KISHI, R. M. **Fusão de informação multimodal por detecção de correlação para tarefas de análise de vídeo.** 2020. 165 p. Tese (Doutorado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2020.

O emprego de fusão prévia multimodal tem se mostrado eficaz em grande parte das tarefas de análise de vídeo existentes. Os métodos de fusão prévia encontrados na literatura foram desenvolvidos para melhorar a eficácia em tarefas específicas e, por esse motivo, são essencialmente vinculados a particularidades de suas respectivas tarefas fim. Com isso, alguns aspectos importantes para a produção de uma representação expressiva por meio de fusão de informação, bem como o potencial de generalização quanto ao domínio de aplicação foram negligenciados em pesquisas até o presente momento. Esta tese de doutorado propõe um método, M4InFus, destinado a realizar fusão de informação multimodal sem utilizar especificidades de domínio de aplicação. O método M4InFus é baseado em identificação de co-ocorrência de padrões unimodais em segmentos de vídeo e cobre lacunas existentes na área de fusão de informação multimodal. O método proposto foi aplicado em dois experimentos na tarefa de Segmentação Temporal de Vídeo em Cenas e em um experimento na tarefa de Classificação de Vídeo, promovendo ganhos em eficácia em ambas as tarefas. Considerando que a eficácia em tais tarefas é limitada pela Lacuna Semântica, há um indício de que representações geradas pelo método M4InFus são menos distantes da semântica contida nos segmentos de vídeo de origem. Este projeto de doutorado também gerou, como contribuição, a implementação do M4InFus e a formação de recursos humanos em níveis de doutorado e de iniciação científica.

Palavras-chave: Multimodalidade, fusão prévia, análise de co-ocorrência, multimídia, vídeo digital.

ABSTRACT

KISHI, R. M. **Multimodal information fusion by correlation detection for video analysis tasks**. 2020. 165 p. Tese (Doutorado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2020.

Multimodal early fusion has been shown to be effective on many of existing video analysis tasks. Available early fusion methods found in the literature had been developed to improve efficacy at specific tasks and, therefore, are essentially tied to particularities of their respective tasks. In this context, research on both, important aspects to compute meaningful representations by information fusion and generalization potential regarding application domain, have been neglected up to this date. This PhD thesis proposes M4InFus, a method intended to perform multimodal information fusion without using application domain specificities. M4InFus method is based on co-occurrence detection of unimodal patterns on video segments and covers existing gaps on multimodal information fusion area. The proposed method have been applied in two experiments on the Temporal Video Scene Segmentation task and one experiment on the Video Classification task, promoting efficacy gains in both tasks. Considering the efficacy in those tasks limited by the Semantic Gap, this information is a clue about the representations generated by the M4InFus method to be less distant from the semantics contained in the original video segments. This doctoral project also produced, as a contribution, an implementation of the M4InFus method and human resources formation on doctoral and undergraduate research levels.

Keywords: Multimodality, early fusion, co-occurrence analysis, multimedia, digital video.

LISTA DE ILUSTRAÇÕES

Figura 1 – Conceito de Lacuna Semântica.	25
Figura 2 – Interação entre modalidades visual e aural.	26
Figura 3 – Exemplo de situação que demanda análise multimodal.	26
Figura 4 – Sequência de quadros compondo uma imagem em movimento.	35
Figura 5 – Exemplo de quadro composto por de linhas e de colunas de pixels.	36
Figura 6 – Conceito de onda sonora e seus atributos.	36
Figura 7 – Onda sonora proveniente do som de um latido de cachorro.	37
Figura 8 – Processo de amostragem de áudio.	38
Figura 9 – Par de tomadas adjacentes.	39
Figura 10 – Relação entre tomadas e cenas.	40
Figura 11 – Aplicação de filtro DoG.	43
Figura 12 – Cálculo de oitavas no SIFT.	44
Figura 13 – Extração de pontos chave SIFT usando a API OpenImaj.	45
Figura 14 – Correspondencia de características SIFT.	46
Figura 15 – Arcabouço da fusão tardia.	50
Figura 16 – Arcabouço da fusão prévia.	51
Figura 17 – Conjunto de elementos de duas diferentes classes divididas por um hiperplano.	57
Figura 18 – Conceito de margem em SVMs.	57
Figura 19 – Dados de duas classes não separáveis linearmente.	58
Figura 20 – Dados transformados passam a ser linearmente separáveis no novo es- paço.	58
Figura 21 – Trecho de arquivo de anotação de tomadas da base BBC.	61
Figura 22 – Trecho de arquivo de anotação de cenas da base BBC.	61
Figura 23 – Trecho de arquivo de anotação de tomadas da base OVSD <i>Dataset</i>	62
Figura 24 – Trecho de arquivo de anotação de cenas da base OVSD <i>Dataset</i>	63
Figura 25 – Avaliação de anotação de posições de fronteiras entre cenas pelo con- junto de medidas PR.	65
Figura 26 – Avaliação de anotação de posições de fronteiras entre cenas pelo con- junto de medidas CO.	67
Figura 27 – Trecho de arquivo de anotação de classes de alguns vídeos da base CCV <i>Database</i>	71
Figura 28 – Visão geral da técnica fusão tardia de Sidiropoulos <i>et al.</i> (2011).	78

Figura 29 – Visão geral da técnica fusão prévia de Gao e Ma (2012).	80
Figura 30 – Visão geral da técnica fusão prévia de Baraldi, Grana e Cucchiara (2015a).	82
Figura 31 – Visão geral da técnica de Rotman, Porat e Ashour (2017).	84
Figura 32 – Visão geral da técnica de Jhuo <i>et al.</i> (2014).	88
Figura 33 – Ilustração do grafo bipartido de palavras aurais e visuais. A largura das arestas é proporcional ao peso daquela aresta. Quanto mais larga é uma aresta, maior o valor de correlação s relacionado a ela.	89
Figura 34 – Visão geral do método de fusão proposto.	93
Figura 35 – Visão geral da extração de características no método M4InFus.	95
Figura 36 – Início do processo de produção de características BoF.	97
Figura 37 – Etapa final da produção de características BoF.	98
Figura 38 – Computação dos histogramas de ocorrência de cada palavra em todos os segmentos de vídeo.	99
Figura 39 – Agrupamento de palavras unimodais com base em seus padrões de ocorrência em diferentes segmentos de vídeo.	100
Figura 40 – Exemplo de co-ocorrência de padrões aurais/visuais em diferentes segmentos de vídeo.	101
Figura 41 – Abstração do processo de <i>pooling</i>	102
Figura 42 – Exemplo de aplicação do <i>average pooling</i> em um grupo de palavras unimodais.	103
Figura 43 – Exemplo de aplicação do <i>max pooling</i> em um grupo de palavras unimodais.	104
Figura 44 – Obtenção dos histogramas de ocorrência de palavras fundidas por segmento de vídeo.	105
Figura 45 – <i>Pipeline</i> de segmentação temporal de vídeo em cenas.	111
Figura 46 – Variação da eficácia de acordo com o valor de δ	115
Figura 47 – Variação da eficácia de acordo com o valor de w	115
Figura 48 – Variação da eficácia de acordo com os tamanhos dos dicionários.	116
Figura 49 – <i>Pipeline</i> de classificação de vídeos proposto por Jhuo <i>et al.</i>	129
Figura 50 – <i>Pipeline</i> de classificação de vídeos modificado para emprego do M4InFus.	130
Figura 51 – Exemplo de semântica expressa em dois diferentes vídeos das classes “festa de casamento” e “dança de casamento” da base confiável CCV <i>Database</i>	133
Figura 52 – Exemplo de semântica expressa em dois diferentes vídeos da classe “festa de casamento” da base confiável CCV <i>Database</i>	133

LISTA DE ALGORITMOS

Algoritmo 1 – Arcabouço da técnica de comparação de documentos por BoF . . .	49
Algoritmo 2 – Grafo de Transição de Cenas	55
Algoritmo 3 – Seleção de quadros chave	112

LISTA DE TABELAS

Tabela 1 – Aspectos importantes dos vídeos da <i>BBC Dataset</i>	61
Tabela 2 – Aspectos importantes dos vídeos da <i>OVSD Dataset</i>	63
Tabela 2 – Aspectos importantes dos vídeos da <i>OVSD Dataset</i>	64
Tabela 3 – Distribuição dos vídeos nas diferentes classes da <i>CCV Database</i>	70
Tabela 4 – Resultados médios de F_{CO} reportados em Sidiropoulos <i>et al.</i> (2011).	79
Tabela 5 – Resultados médios de F_{PR} reportados em Gao e Ma (2012)	81
Tabela 6 – Resultados médios de F_{CO} reportados em Baraldi, Grana e Cucchiara (2015a)	83
Tabela 7 – Resultados médios de F_{CO} reportados em Rotman, Porat e Ashour (2017)	85
Tabela 8 – Valores obtidos por medidas de eficácia na base confiável <i>BBC Dataset</i>	116
Tabela 9 – Comparação entre valores de medidas de eficácia obtidos pelo <i>pipeline</i> proposto com e sem o emprego do método M4InFus.	117
Tabela 10 – Comparação entre valores de F_{CO} obtidos pelo <i>pipeline</i> proposto com os de dois outros métodos de segmentação na base confiável <i>BBC Dataset</i>	118
Tabela 11 – Parametrização do algoritmo STG, por vídeo, que gerou os melhores resultados na base confiável <i>OVSD Dataset</i>	123
Tabela 12 – Valores de Precisão, Abrangência, F_{PR} , <i>Coverage</i> , <i>Overflow</i> e F_{CO} obtidos na base confiável <i>BBC Dataset</i> , usando características extraídas por redes neurais profundas.	123
Tabela 13 – Comparação dos valores de F_{PR} obtidos na base confiável <i>BBC Dataset</i> pelas duas diferentes versões do <i>pipeline</i> de segmentação temporal de vídeo em cenas.	124
Tabela 14 – Valores de F_{CO} obtidos na base confiável <i>BBC Dataset</i> , usando características extraídas por redes neurais profundas.	125
Tabela 15 – Valores de medidas de eficávis obtidos na base confiável <i>OVSD Dataset</i> , usando características extraídas por redes neurais profundas.	126
Tabela 16 – Comparação entre valores de F_{CO} obtidos pelo <i>pipeline</i> proposto ao empregar características extraídas por redes neurais profundas e dois métodos de segmentação em estado da arte, na base confiável <i>OVSD Dataset</i>	128
Tabela 17 – Comparação entre valores de <i>Mean Average Precision</i> obtidos três <i>pipelines</i> de classificação de vídeo.	132
Tabela 18 – Aspectos técnicos dos arquivos de vídeo da base confiável <i>OVSD Dataset</i>	165

LISTA DE ABREVIATURAS E SIGLAS

F_n	<i>false negatives</i>
F_p	<i>false positives</i>
P	Precisão
R	Abrangência
T_p	<i>true positives</i>
ACCK	Análise de Correlação Canônica com <i>Kernel</i>
AP	<i>Average Precision</i>
BoAW	<i>Bag-of-Aural-Words</i>
BoF	<i>Bag-Of-Features</i>
BoVW	<i>Bag-of-Visual-Words</i>
BoW	<i>Bag-of-Words</i>
CCV Database	<i>Columbia Consumer Video Database</i>
CNN	<i>Convolutional Neural Network</i>
CO	<i>Coverage e Overflow</i>
CSIFT	<i>Colored SIFT</i>
dB	decibéis
DoG	<i>Difference-of-Gaussians</i>
DoH	<i>Determinant of Hessian</i>
FPS	<i>Frames per Second</i>
HMM	<i>Hidden Markov Models</i>
Hz	hertz
kHz	kilohertz
LoG	<i>Laplacian-of-Gaussians</i>
M4InFus	<i>Multimedia Multimodal Information Fusion</i>
MAP	<i>Mean Average Precision</i>
MFCC	<i>Mel-Frequency Cepstral Coefficients</i>
NO	<i>New Overflow</i>
OVSD	<i>Open Video Scene Detection Dataset</i>
PR	<i>Precision and Recall</i>
RGB	<i>Red, Green and Blue</i>
RNN	<i>Recurrent Neural Network</i>

SIFT	<i>Scale-Invariant Feature Transform</i>
STG	<i>Scene Transition Graph</i>
SVM	<i>Support Vector Machines</i>
YCbCr	<i>Luminance, Chroma Blue and Chroma Red</i>

SUMÁRIO

1	INTRODUÇÃO	23
1.1	Fusão prévia em análise de vídeo	28
1.2	Enunciado do problema	29
1.3	Objetivo desta tese	30
1.4	Organização	30
2	CONCEITOS RELACIONADOS	33
2.1	Vídeo digital	33
2.1.1	<i>Informação visual</i>	34
2.1.2	<i>Informação aural</i>	36
2.1.3	<i>Estrutura hierárquica</i>	38
2.2	Características de vídeo	41
2.2.1	<i>Extração de características de baixo nível semântico</i>	42
2.2.2	<i>Extração de características de médio nível semântico</i>	47
2.3	Fusão de modalidades	50
2.4	Métodos para tarefas de análise de vídeo	52
2.4.1	<i>Detecção de fronteiras entre cenas</i>	52
2.4.2	<i>Classificação de vídeos</i>	56
2.5	Avaliação de tarefas de análise de vídeo	59
2.5.1	<i>Avaliação de segmentação temporal de vídeo em cenas</i>	60
2.5.2	<i>Avaliação de classificação de vídeos</i>	69
2.6	Considerações finais	72
3	TRABALHOS RELACIONADOS	75
3.1	Trabalhos em segmentação temporal de vídeo em cenas	76
3.2	Trabalhos em classificação de vídeos	87
3.3	Considerações finais	90
4	DESCRIÇÃO DO MÉTODO	93
4.1	Extração de características	94
4.2	Representação por <i>bag-of-features</i>	96
4.3	Detecção de co-ocorrência	99
4.4	Síntese em representação única	101

4.4.1	<i>Average Pooling</i>	102
4.4.2	<i>Max Pooling</i>	103
4.4.3	<i>Pooling ponderado</i>	104
4.4.4	<i>Conversão em histogramas por segmento de vídeo</i>	105
4.5	Considerações finais	106
5	AVALIAÇÃO EXPERIMENTAL	109
5.1	Experimento 1: segmentação temporal usando algoritmos <i>baseline</i>	110
5.1.1	<i>Metodologia</i>	110
5.1.2	<i>Resultados</i>	116
5.2	Experimento 2: segmentação temporal usando <i>deep features</i>	119
5.2.1	<i>Metodologia</i>	120
5.2.2	<i>Resultados</i>	123
5.3	Experimento 3: classificação de vídeos	128
5.3.1	<i>Metodologia</i>	129
5.3.2	<i>Resultados</i>	131
5.4	Considerações finais	134
6	CONCLUSÃO	139
6.1	Contribuições	139
6.2	Publicações relacionadas	140
6.3	Limitações e trabalhos futuros	142
REFERÊNCIAS		145
GLOSSÁRIO		163
APÊNDICE A	INFORMAÇÕES TÉCNICAS SOBRE OS VÍDEOS DA BASE CONFIÁVEL OVSD	165

INTRODUÇÃO

Uma das principais fontes de informação adotadas pela sociedade é o vídeo digital. Pessoas assistem a vídeos a todo momento em seus televisores, *smartphones* e até mesmo em telas acopladas nos mais diversos dispositivos como, por exemplo, as centrais multimídia automotivas. É por meio dos vídeos digitais que muitas pessoas recebem notícias, avaliam produtos, se entretêm e até mesmo estudam. Além do público comum, empresas utilizam vídeo digital para produzir propaganda e para comunicarem-se com os consumidores finais. Dia após dia são expostas novas situações onde o vídeo digital tem se revelado útil. Considerando o crescimento no consumo de vídeo digital pela sociedade, é natural que haja aumento no volume de conteúdo de vídeo digital disponível, bem como na produção de novos conteúdos. Essa afirmação encontra respaldo em dados do mercado: de acordo com um relatório apresentado pela [Cisco \(2018\)](#), vídeo digital correspondeu a 75% do tráfego mundial em redes IP e deve subir a 82% até o ano 2022. O mesmo relatório informa que o tráfego de vídeo sob demanda vai praticamente dobrar até 2022, sendo equivalente à capacidade de dez bilhões de DVDs por mês.

O grande volume de dados de vídeo disponível leva os consumidores de vídeo a uma situação chamada **Sobrecarga de Informação** (*Information Overload*). A Sobrecarga de informação é uma situação onde um indivíduo é incapaz de lidar com um grande conjunto de dados ao qual é exposto, para obter uma pequena parte que lhe é de interesse. O termo sobrecarga de informação, que inicialmente não dizia respeito ao cenário de computação, foi cunhado por [Gross \(1965\)](#), popularizado por [Toffler \(1970\)](#) e descreve perfeitamente a situação atual dos consumidores de vídeo. As dificuldades decorrentes da sobrecarga de informação motivam o estudo e desenvolvimento de técnicas computacionais automáticas eficazes para a análise de grandes volumes de dados de vídeo. A análise de vídeo, neste contexto, corresponde à extração de informações que serão utilizadas em aplicações que compõem o processo de consumo do vídeo pelo espectador ([HAUPTMANN, 2009](#)). Aplicações de vídeo como, por exemplo, a busca ([CHEN et al., 2018](#); [LOKOč et al., 2018](#)), a

navegação (MÜNZER; SCHOEFFMANN, 2018) e a adaptação (ROTHFUSS; MÜNSTER; ZIMMERMANN, 2019) ainda demandam desenvolvimento para atingir melhor eficácia. A eficácia destas aplicações depende, diretamente, de sua capacidade de análise dos vídeos, por meio de tarefas como a segmentação temporal (ROTMAN; PORAT; ASHOUR, 2017), classificação (JIANG *et al.*, 2018), dentre outras.

A eficácia das tarefas de análise de vídeo, por sua vez, depende da capacidade de análise da semântica contida nos vídeos. Um exemplo desta dependência pode ser visto na tarefa de segmentação temporal de vídeo em cenas, realizada de modo automatizado. Dado um vídeo, a segmentação temporal deste vídeo em cenas consiste em definir onde, temporalmente, estão as fronteiras entre as cenas do vídeo. Uma cena é definida como um conjunto de tomadas adjacentes semanticamente relacionadas (SARACENO; LEONARDI, 1997) onde, tomadas, por sua vez, são sequências de quadros capturados ininterruptamente por uma única câmera (BRUNELLI; MICH; MODENA, 1999; KOPRINSKA; CARRATO, 2001). Considerando que a maioria das técnicas de segmentação de vídeo em cenas é baseada no agrupamento de tomadas adjacentes por sua semântica (FABRO; BÖSZÖRMENYI, 2013; ROTMAN; PORAT; ASHOUR, 2017), a extração de informações que representem algum aspecto ou indício de existência de algum tipo de semântica das tomadas de um vídeo caracteriza-se como etapa fundamental nesse processo. Outro exemplo de situação onde há necessidade de extração de informações que possibilitem explorar algum aspecto da semântica é a da tarefa de classificação de vídeos, onde, dados um vídeo e um conjunto de categorias, é necessário decidir se o vídeo pertence a uma ou mais categorias do conjunto dado como entrada (BREZEALE; COOK, 2008). A maioria das abordagens para classificação de vídeos consiste em extrair uma representação compacta do vídeo que é fornecida como entrada para um ou mais classificadores, previamente treinados, que decidirão a qual(is) classe(s) o vídeo original pertence. Neste caso, quanto melhor for a representação compacta em relação à semântica contida no vídeo original, melhor será a eficácia do(s) classificador(es).

A extração de semântica de vídeo é considerada uma tarefa difícil, já que enfrenta a **Lacuna Semântica** (*Semantic Gap*) (SMEULDERS *et al.*, 2000), termo que descreve a falta de ligação entre a representação computacional de um vídeo e de seu significado ao ser assistido por um espectador. A Figura 1 ilustra o conceito de Lacuna Semântica em um vídeo.

Na Figura 1, o quadro à esquerda representa a codificação binária do conteúdo de um vídeo, armazenada no arquivo `cachorro.mp4`. Este arquivo codifica cores de pixels, suas posições e também amplitudes quantizadas de sinais de áudio. O quadro à direita, ilustra os estímulos visuais gerados por uma tela ao reproduzir o conteúdo de `cachorro.mp4`. Considere que o sinal `aural` codificado em `cachorro.mp4` é reproduzido juntamente com o fluxo visual. Um ser humano que assista a tal vídeo, compreenderá

Figura 1 – Conceito de Lacuna Semântica.

que o vídeo `cachorro.mp4` ilustra um cão, dados a imagem formada pela exibição de seus pixels e os sons do cão reproduzidos do sinal de áudio, mesmo que esta informação sobre o cão não esteja representada explicitamente no arquivo de vídeo. A Lacuna Semântica, ilustrada entre os dois quadros, é a falta de ligação entre o significado da reprodução do conteúdo, um cachorro, e de sua codificação binária, que é um conjunto de valores e posições de cores de uma imagem e de amplitudes que compõem ondas sonoras.

São necessários, então, esforços rumo à redução da Lacuna Semântica, para que seja possível produzir melhores representações computacionais, que preservem e exponham a semântica presente no conteúdo de vídeo. Estes esforços costumam se dar pelo desenvolvimento de métodos de análise dos dados brutos de vídeo destinados a expor algum aspecto da semântica presente e representá-la de forma que seja mais facilmente utilizável. A semântica contida em um vídeo é expressa usando os canais de informação visual e aural (SNOEK; WORRING, 2002). Estes canais, chamados **modalidades** (SNOEK; WORRING, 2002), normalmente se complementam para expressar conceitos semânticos como, por exemplo, na Figura 2.

Nesta figura, extraída do filme *open content* chamado *Tears of Steel*¹, é possível observar uma situação onde a modalidade visual exibe um objeto metálico em formato de mão e a modalidade aural emite sons de mecanismos robóticos. Nesta figura, há ainda a expressão, pela modalidade visual através de texto, das palavras em língua inglesa “*robot hand*” (mão robótica, em português). As duas modalidades, em conjunto, expressam o conceito mão robótica.

Apesar de existirem técnicas de análise de vídeo **unimodais** (empregam apenas uma modalidade), o emprego de **multimodalidade** (múltiplas modalidades) é necessária em grande parte das tarefas de análise de vídeo, correspondendo à natureza intrinsecamente multimodal dos vídeos (SNOEK; WORRING, 2002; JIANG *et al.*, 2018). Um exemplo de situação onde uma técnica de análise unimodal baseada apenas no conteúdo

¹ Disponível em <<https://mango.blender.org/>>.

Figura 2 – Interação entre modalidades visual e aural.

visual falharia pode ser vista na Figura 3, que contém quadros extraídos do episódio “From Pole to Pole” da série de documentários “BBC Planet Earth”².

BBC Planet Earth - Episódio *From Pole to Pole*
Quadro 7060 / Tomada 37 / Cena 06
Semântica: filhotes saindo da toca pela primeira vez

BBC Planet Earth - Episódio *From Pole to Pole*
Quadro 7370 / Tomada 39 / Cena 07
Semântica: filhotes famintos tentando alcançar sua mãe

Figura 3 – Exemplo de situação que demanda análise multimodal.

Pensando na tarefa de segmentação temporal de vídeo em cenas, a Figura 3 contém, à esquerda, um quadro extraído de uma cena que apresenta ao espectador filhotes de urso andando na neve pela primeira vez. A cena seguinte apresenta conteúdo visual semelhante, porém, com significado diferente, de filhotes de urso tentando alcançar sua mãe. Neste caso,

² Disponível em <<https://www.bbc.co.uk/programmes/b006mywy>>.

a fronteira entre as cenas é definida pela modalidade aural, com o narrador explicando o conteúdo da cena. Uma técnica de segmentação em cenas baseada apenas no conteúdo visual seria incapaz de capturar a mudança de semântica entre os dois conjuntos adjacentes de tomadas.

Apesar da possibilidade de aumentar a eficácia pelo acréscimo de informação complementar, o emprego de multimodalidade demanda a implementação de uma estratégia para combinar a informação proveniente das diferentes modalidades (ATREY *et al.*, 2010). Um dos meios mais comuns de se realizar a combinação da informação proveniente de diferentes modalidades é a **fusão tardia** (ATREY *et al.*, 2010). Na fusão tardia é realizada uma etapa de processamento para cada modalidade disponível e, em seguida, uma fusão dos resultados individuais, dando origem a um único resultado consensual. Há muitos métodos de análise de vídeo que adotam a fusão tardia como, por exemplo, os propostos por Sundaram e Chang (2000), Sidiropoulos *et al.* (2011) e Lopes, Trojahn e Goularte (2014).

A popularidade da fusão tardia se deve, principalmente, ao fato de que produz resultados razoavelmente bons e ao mesmo tempo é relativamente simples de ser implementada. Apesar da facilidade de aplicação, os métodos fusão tardia ignoram qualquer correlação existente entre diferentes modalidades na etapa principal de processamento (ATREY *et al.*, 2010), como é o caso da situação ilustrada na Figura 2. Essa característica da fusão tardia pode limitar os métodos à obtenção de resultados sub ótimos (ROTMAN; PORAT; ASHOUR, 2017). Os métodos de fusão tardia também são considerados ineficientes já que é necessária uma etapa distinta de processamento para cada modalidade ou para cada conjunto diferente de dados de entrada.

Há outra abordagem para combinar informações de diferentes modalidades, a **fusão prévia** (ATREY *et al.*, 2010), onde as informações extraídas dos dados de cada modalidade, individualmente, são combinadas dando origem a um conjunto de informações único que será utilizado para realizar o processamento principal em uma única etapa. Na fusão prévia é possível efetuar um refinamento da informação disponível com o objetivo de aproveitar a correlação entre informações de diferentes modalidades bem como diminuir o custo computacional de processamento ao descartar porções de informação redundante.

Técnicas de análise de vídeo multimodais baseadas em fusão prévia costumam ser mais eficientes que aquelas baseadas em fusão tardia (SNOEK; WORRING; SMEULDERS, 2005; ATREY *et al.*, 2010). Apesar de haver um custo computacional extra para a etapa de fusão, este custo, acrescido do custo de uma única etapa de processamento principal na fusão prévia, costuma ser menor do que o custo total de múltiplas etapas de processamento principal da fusão tardia. Além disso, a fusão prévia permite que a representação multimodal seja computada uma única vez, podendo ser reutilizada em outras

tarefas de análise de vídeo ou mesmo em outras instâncias de execução de uma mesma tarefa de análise de vídeo. Desse modo, o custo computacional da fusão prévia em cenários de aplicação onde há diversas análises de um mesmo vídeo (ou de um conjunto de vídeos) pode ser ainda mais atrativo do que o custo computacional da fusão tardia.

Devido aos potenciais ganhos de eficiência, pela redução de custo computacional, e de eficácia, pelo aproveitamento de correlação entre informação intermodal, justificam-se atividades de estudo e aprimoramento da fusão prévia em análise de vídeo. Computar representações mais expressivas pode permitir a melhora em métodos de tarefas de vídeo como busca e recomendação e, conseqüentemente, beneficiar serviços como o Netflix³ e o Youtube⁴ que empregam métodos para a realização de tais tarefas.

1.1 Fusão prévia em análise de vídeo

Os primeiros esforços orientados à aplicação de fusão prévia em análise de vídeo, propuseram a realização da fusão de maneira ingênua, simplesmente concatenando as informações das diferentes modalidades antes do processo de decisão principal. Essa abordagem é suscetível à maldição da dimensionalidade (FONCUBIERTA-RODRÍGUEZ; MÜLLER; DEPEURSINGE, 2014) e não lida com a correlação intermodalidades (entre duas ou mais modalidades) fazendo com que a identificação e uso de qualquer correlação existente sejam delegados às etapas posteriores de análise. Neste tipo de abordagem ingênua, a análise costuma se dar pela aplicação de alguma técnica clássica de aprendizado de máquina como, por exemplo, algoritmos de agrupamento de dados (ELLOUZE; BOUJEMAA; ALIM, 2010) e de classificação de dados (XU; FENG; XU, 2013), sem qualquer modificação para identificação de informação redundante e/ou complementar entre as modalidades. Além da possível indiferença à correlação intermodal na etapa de decisão, abordagens ingênuas de fusão prévia podem a ser ineficientes. Isso acontece pois uma única etapa de decisão pode ser computacionalmente mais cara do que múltiplas etapas de decisão com entradas de tamanho menor. A etapa única de decisão pode ser ineficiente devido ao tamanho da entrada obtida por concatenação.

O emprego de fusão prévia de modo não ingênuo esbarra em alguns dilemas de projeto (discutido na Seção 2.3): identificação de correlação entre informações heterogêneas provenientes de diferentes modalidades; falta de sincronismo temporal entre informações intermodais correlacionadas; ausência de um modelo bem estabelecido de representação de informação multimodal utilizando uma única estrutura de dados. Apesar dessas dificuldades, a fusão prévia tem se mostrado uma abordagem promissora, considerando resultados obtidos em trabalhos de análise de vídeo como Gao e Ma (2012), Jhuo *et al.*

³ Acessível em <www.netflix.com>.

⁴ Acessível em <www.youtube.com>.

(2014), Baraldi, Grana e Cucchiara (2015a), Liu, Yuan e Wang (2019a) que contam com métodos mais sofisticados de fusão de informação.

Dentre as abordagens não triviais, há um subconjunto onde a fusão de informação é delegada às redes de aprendizado profundo. Apesar da possibilidade de obtenção de bons resultados, a informação fundida produzida pelo aprendizado profundo é desprovida de referência para as informações unimodais originais. Isso impede que atividades de pós processamento possam ser aplicadas, as quais comumente visam refinar a representação em busca de aumento de eficácia. Além disso, como indicado nos trabalhos de Jin *et al.* (2014) e Nanni, Ghidoni e Brahnam (2017), as representações geradas por redes neurais contém informações diferentes das representações geradas por métodos de outras naturezas. Por fim, as técnicas baseadas em aprendizado profundo, tem suas redes projetadas para lidar com uma quantidade fixa de modalidades. Uma mudança nas modalidades envolvidas no processo de fusão, além de demandar a modificação do projeto da rede, também faz com que toda a etapa de treinamento e de parametrização deva ser repetida.

Em relação às técnicas de fusão prévia não baseadas em aprendizado profundo existentes, não se tem notícia, para o melhor do conhecimento obtido, da existência de alguma apta a produzir uma representação única que contenha padrões unimodais e multimodais ao mesmo tempo. Por existirem padrões unimodais essenciais para uma representação adequada de algum aspecto da semântica de um vídeo, técnicas limitadas ao tratamento de padrões multimodais produzirão uma representação incompleta daquele aspecto, que fará com que o vídeo seja analisado sem que seja considerada a porção expressada por qualquer padrão unimodal. Outra limitação dessas técnicas é não-escalabilidade em relação a quantidade de diferentes modalidades ou características que podem ser empregadas no processo de fusão. Há diversas situações onde três ou mais modalidades são necessárias para representar um padrão, ou até mesmo três ou mais perspectivas diferentes de uma ou mais modalidades. As técnicas disponíveis foram projetadas para a fusão de informações de apenas duas naturezas diferentes.

1.2 Enunciado do problema

Os métodos de fusão prévia existentes, por terem sido projetados sem maiores pretensões de generalização quanto à aplicação, acabam por serem intrinsecamente ligados às tarefas que motivaram seu desenvolvimento. Por esse motivo, lacunas como a não-rastreabilidade de valores pertencentes a representações, a inflexibilidade em relação à representação simultânea de padrões unimodais e multimodais e a não-escalabilidade quanto a quantidade de diferentes características/modalidades empregadas no processo de fusão não foram alvos específicos de pesquisas. Tais lacunas são barreiras rumo a ganhos em eficácia nas tarefas de análise de vídeo que dependem de fusão de informação

multimodal e fazem com que seja difícil reaproveitar os métodos existentes em outras tarefas.

Conclui-se então que não há, para o melhor do conhecimento obtido, métodos de fusão de informação multimodal desacoplados de tarefas fim e tão pouco capazes de lidar simultaneamente com as lacunas supracitadas. Esse contexto limita tanto a eficácia em tarefas de análise de vídeo quanto os possíveis domínios de aplicação.

1.3 Objetivo desta tese

O objetivo desta tese é propor um método de fusão prévia de informação multimodal capaz de propiciar ganhos de eficácia em diferentes tarefas de análise de vídeo.

Para tanto, o método proposto, chamado *Multimedia Multimodal Information Fusion* (M4InFus), produz representações multimodais únicas de [segmentos de vídeo](#) de um conjunto dado como entrada. O método M4InFus produz tais representações sem utilizar informações específicas de domínio, possibilitando tratar, ao mesmo tempo, os dilemas clássicos da fusão prévia e as lacunas levantadas. Com isso, o resultado da fusão multimodal se torna mais representativo que os resultados de abordagens similares, além de possuir potencial para generalização quanto ao domínio de aplicação.

1.4 Organização

Além deste capítulo de introdução, contextualizando a área e o problema tratado, esta tese está organizada em mais cinco capítulos. O Capítulo 2 apresenta os conceitos necessários à compreensão desta tese. O primeiro dos conceitos abordados é o de vídeo digital, suas fontes de informação e sua estrutura temporal hierárquica. Em seguida, são explorados conceitos básicos de representação de conteúdo de vídeo, por meio de vetores de características. A representação por vetores de características permite a comparação entre diferentes vídeos por algum aspecto de seu conteúdo. Logo após, introduz-se fusão de modalidades, que possibilita o emprego de informação de diferentes origens na solução de um problema. O método de fusão de informação multimodal proposto nesta tese será avaliado através de sua aplicação nos problemas de segmentação temporal de vídeo em cenas e de classificação de vídeos. Por este motivo, o final deste capítulo trata de métodos para lidar com essas duas tarefas, descrevendo-os e discutindo formas de avaliá-los.

O Capítulo 3 tem como objetivo apontar as lacunas de pesquisa em fusão de informação nos métodos existentes, que são acoplados a técnicas de análise de vídeo. Essa ação foi realizada para as tarefas de segmentação temporal de vídeo em cenas e de classificação de vídeo, por meio de análises das etapas de fusão de informação em métodos destinados a ambas as tarefas.

O Capítulo 4 descreve, detalhadamente, o método de fusão de informação multimodal proposto nesta tese. Este método é dividido em quatro etapas: extração de características, representação por *bag-of-features*, detecção de co-ocorrência e síntese em representação única. Juntamente com a explicação do funcionamento de cada etapa, apresenta-se uma discussão do raciocínio por trás de cada etapa e de seus efeitos esperados.

O Capítulo 5 relata três experimentos que foram realizados para avaliar a eficácia do método de fusão de informação multimodal proposto. Em dois destes experimentos, o método proposto foi combinado a técnicas de segmentação temporal de vídeo em cenas. No terceiro, e último, o método proposto foi acoplado a uma técnica de classificação de vídeos. O objetivo, nos três experimentos, era analisar a influência do método de fusão de informação proposto na eficácia destas técnicas.

O Capítulo 6 apresenta inicialmente as conclusões resultantes do trabalho realizado. Em seguida, são listadas as contribuições científicas e as publicações originadas da pesquisa realizada. Por fim, são indicadas as limitações do trabalho realizado acompanhadas de sugestões para trabalhos futuros.

CONCEITOS RELACIONADOS

Este capítulo apresenta conceitos necessários para a compreensão desta tese. A Seção 2.1 apresenta a definição de vídeo digital e uma explicação sobre seus componentes: os canais de informação visual e aural. Essa seção também apresenta o conceito de estrutura hierárquica de vídeo. Já a Seção 2.2 contém uma explicação sobre extração e representação de características de vídeo. A Seção 2.3 tem como tema a fusão de informação, apontando seus benefícios e discutindo as dificuldades de aplicação. Na Seção 2.4 são discutidas abordagens computacionais para as tarefas de análise de vídeo de segmentação temporal e de classificação de vídeo. Por fim, a Seção 2.5 trata das formas de avaliação de tarefas de segmentação e de classificação de vídeo, discutindo as principais bases de dados e medidas de avaliação existentes.

2.1 Vídeo digital

Vídeo é uma **mídia** utilizada para transmitir informação a espectadores, por meio de um ou mais canais sensoriais como, por exemplo, os canais visual e aural (SNOEK; WORRING, 2002). Até o fim do século XX, a maioria do conteúdo de vídeo era produzida, armazenada e reproduzida utilizando tecnologia analógica (SAVAGE; VOGEL, 2008). Graças à evolução e a popularização dos computadores, o conteúdo de vídeo tem migrado, em todas as suas etapas (produção, armazenamento e reprodução), para a tecnologia digital, caracterizando-o como **vídeo digital** (SAVAGE; VOGEL, 2008). Apesar de existirem vídeos compostos apenas por um canal de informação visual, a maioria dos vídeos digitais disponíveis é composta por pelo menos um canal de informação visual e um ou mais canais de informação aurais.

2.1.1 Informação visual

O olho humano recebe, a todo momento, informação visual em formato bidimensional que muda de maneira contínua com o tempo (WATKINSON, 2007) e permite uma sensação de movimento. Dada essa característica do sistema visual humano, o conteúdo visual de um vídeo, chamado **imagem em movimento** (do inglês, *moving picture* ou *motion picture*) (SAVAGE; VOGEL, 2008; CHAPMAN; CHAPMAN, 2009) tem, como objetivo, ser capaz de causar a sensação de movimento aos espectadores (SAVAGE; VOGEL, 2008; CHAPMAN; CHAPMAN, 2009). A sensação de movimento contínuo natural que é vivenciada pelas pessoas, pode ser simulada com o emprego de informação visual em formato discreto, graças a um fenômeno conhecido como persistência da visão. Este fenômeno ocorre quando uma sucessão de imagens estáticas que ilustram instantes sequenciais de elementos em movimento e diferem pouco entre si é apresentada a um espectador, a uma taxa mínima¹ (WATKINSON, 2007; SAVAGE; VOGEL, 2008; CHAPMAN; CHAPMAN, 2009). Nesta situação, o espectador experimentará uma ilusão de movimento contínuo, ao invés de perceber uma sequência de imagens estáticas.

Com base no fenômeno da persistência da visão, a imagem em movimento de um vídeo digital é composta por uma sequência de imagens estáticas tais que, quando apresentadas sucessivamente a uma determinada taxa, podem causar a sensação de movimento em um espectador (SAVAGE; VOGEL, 2008; CHAPMAN; CHAPMAN, 2009). Cada uma das imagens estáticas de uma imagem em movimento é chamada **quadro**. A taxa específica em que os quadros devem ser apresentados para causar a sensação de movimento é chamada **taxa de reprodução**, normalmente medida em **quadros por segundo** ou, em inglês, *Frames per Second* (FPS). A imagem em movimento é exemplificada na Figura 4, extraída do conteúdo visual do filme *open content Meridian*².

Na Figura 4, é ilustrada uma situação onde há dois automóveis: um parado, em primeiro plano, e um em segundo plano, movimentando-se. O veículo que está em movimento, a cada quadro da sequência, está posicionado mais próximo da borda esquerda da imagem. Ao serem reproduzidos sucessivamente em taxa adequada, estes quadros causam a sensação de que um dos veículos está se movimentando de maneira contínua.

Cada quadro é uma **imagem digital**, que por sua vez é uma matriz bidimensional de valores, onde cada valor representa um ponto da imagem estática original que foi digitalizada. Denominamos cada elemento desta matriz como **pixel**, cujo valor corresponde a uma codificação das intensidades de luminância e componentes de cores que compõem a cor da imagem original naquela posição (POYNTON, 1996). Há dois principais sistemas de representação de cor de pixels empregados em vídeo digital: *Red, Green and Blue* (RGB) e *Luminance, Chroma Blue and Chroma Red* (YCbCr). O sistema RGB descreve

¹ A taxa mínima para que a persistência da visão ocorra varia em função da luminosidade.

² Disponível em <https://www.youtube.com/watch?v=u1LTx4h7D_0>.

Figura 4 – Sequência de quadros compondo uma imagem em movimento.

um pixel como uma combinação de componentes de intensidade vermelho (R), verde (G) e azul (B). Já o sistema YCbCr descreve um pixel como uma combinação de um componente luminância (Y), e dois componentes crominância que são, a diferença do azul (Cb) e a diferença do vermelho (Cr).

Ainda em relação aos atributos de um quadro, a **resolução espacial** determina o quão acuradamente ele representa imagens originalmente contínuas, por meio de uma quantidade finita de pixels (POYNTON, 1996; CHAPMAN; CHAPMAN, 2009). A resolução espacial de um vídeo digital é dada pela resolução espacial de um de seus quadros (CHAPMAN; CHAPMAN, 2009) já que normalmente todos os quadros possuem a mesma resolução espacial. A resolução espacial de um vídeo é especificada por um par de valores correspondentes às quantidades de linhas e colunas de pixels que compõem um quadro daquele vídeo, com notação no formato $x \times y$, onde x é o número de colunas e y é o número de linhas. A Figura 5 ilustra o conceito de resolução espacial de um quadro.

Nesta figura há um quadro proveniente do filme *open content* chamado *Big Buck Bunny*³, dividido pixel a pixel, exemplificando a matriz de cores a que este quadro corresponde e com a indicação das quantidades de linhas e colunas que compõem sua resolução espacial, de 100×56 ⁴. Cada pequeno quadrado colorido na grade corresponde a um pixel. A matriz de pixels exibida com os pixels dispostos em suas posições bi-dimensionais pré-determinadas compõem a reprodução discreta da imagem contínua original.

³ Disponível em <<https://peach.blender.org/>>.

⁴ A resolução espacial do vídeo original é de 854×480 e foi reduzida para 100×56 neste exemplo para que as linhas que representam as fronteiras entre pixels permanecessem visíveis.

Figura 5 – Exemplo de quadro composto por de linhas e de colunas de pixels.

2.1.2 Informação aural

Além da informação visual, um vídeo costuma possuir um ou mais canais de áudio, destinados a reprodução sincronizada com o conteúdo visual. A informação aural de um vídeo é um conjunto de estímulos codificados tais que, ao serem reproduzidos, produzam sons. O som é uma forma de energia transmitida por meio de vibrações em um meio elástico como, por exemplo, o ar (SAVAGE; VOGEL, 2008; CHAPMAN; CHAPMAN, 2009). Estas vibrações no meio podem ser abstraídas como uma onda e, ao alcançarem o tímpano humano, fazem com que ele vibre na mesma frequência da origem do som (CHAPMAN; CHAPMAN, 2009). A vibração do tímpano é convertida pelo sistema auricular humano em impulsos nervosos que são interpretados pelo cérebro como informação (CHAPMAN; CHAPMAN, 2009). A onda sonora possui três características essenciais (SAVAGE; VOGEL, 2008), ilustradas na Figura 6, inspirada em Savage e Vogel (2008).

Figura 6 – Conceito de onda sonora e seus atributos.

A primeira destas características é a **amplitude**, que representa a pressão do som e tem efeito na percepção de **volume** pelos ouvintes. Um som com alta amplitude

causa a sensação de volume alto. A amplitude costuma ser medida em decibéis (dB). A amplitude é representada pelo eixo vertical da Figura 6. A segunda característica, chamada **frequência**, corresponde ao número de repetições do padrão da onda em um determinado intervalo de tempo. A frequência é indicada na Figura 6, como sendo a distância entre dois picos de onda e é medida em hertz (Hz). Cada **hertz** corresponde a uma repetição do padrão de onda por segundo. A frequência é compreendida pelo ouvinte como a altura do som onde, frequências altas equivalem a sons agudos e frequências baixas equivalem a sons graves. A terceira característica de uma onda sonora é a **duração**, que determina quanto tempo o som dura. A duração é representada pelo eixo horizontal na Figura 6.

A Figura 6 ilustra o mais simples padrão de onda sonora, classificada como **som puro**. Os sons, em geral, apresentam padrões de onda mais complexos do que os sons puros. Estes padrões de onda, no entanto, nada mais são do que combinações de diversas componentes de sons puros. A Figura 7, produzida utilizando o editor de áudio online Twisted Wave ⁵, apresenta um exemplo de formato de onda baseada em um som real, que contém um latido de cachorro.

Figura 7 – Onda sonora proveniente do som de um latido de cachorro.

De maneira análoga ao processo de digitalização de conteúdo visual, os sistemas de áudio digital representam as ondas sonoras, de natureza contínua, como elementos discretos de informação (SAVAGE; VOGEL, 2008). Nos sistemas atuais, o processo de digitalização de som se dá em duas etapas. A primeira etapa é a conversão das vibrações que compõem as ondas (energia mecânica) em sinal elétrico por um **transdutor** como, por exemplo, um microfone ou um captador de guitarra. Este sinal elétrico é uma variação de tensão elétrica ao longo do tempo. A segunda etapa é a conversão do sinal elétrico contínuo gerado pela primeira etapa em um sinal digital discreto, no processo de **amostragem**, ilustrado na Figura 8.

Nesta figura, a onda correspondente ao sinal elétrico original é ilustrada com a cor azul. O sinal é dividido em intervalos de tempo e cada intervalo é representado por um valor discreto, ilustrado em vermelho. Este valor discreto é o que melhor representa o

⁵ Disponível em <<https://twistedwave.com/online>>.

Figura 8 – Processo de amostragem de áudio.

valor do sinal elétrico naquele intervalo. No exemplo da Figura 8, o critério adotado para determinar o valor correspondente a um intervalo é utilizar a primeira tensão do sinal elétrico contida naquele intervalo.

O sinal digital é uma sequência de valores numéricos, onde cada valor corresponde a amplitude da onda sonora em um determinado instante (SAVAGE; VOGEL, 2008), ou seja, o sinal digital é uma sequência de amostras do som. Considerando que há informação perdida na discretização da onda, já que o sinal analógico é infinitamente divisível, a qualidade do som amostrado depende, principalmente, de dois fatores: **resolução da amostra** e **taxa de amostragem**. A resolução de uma amostra diz respeito à precisão do valor que pode ser representado em uma amostra e é determinada pela quantidade de bits utilizados para codificar uma amostra de amplitude (SAVAGE; VOGEL, 2008). Os valores mais comuns de resolução de áudio são 8, 16 (qualidade de CD) e 24 bits (SAVAGE; VOGEL, 2008). A taxa de amostragem diz respeito ao intervalo de frequências que podem ser representadas naquela codificação. Ela é definida pela quantidade de amostras computadas em um intervalo de tempo fixo (SAVAGE; VOGEL, 2008). Dado que, em geral, são extraídas milhares de amostras por segundo, as taxas de amostragem costumam ser medidas em kilohertz (kHz). Taxa de amostragem de um áudio em qualidade de CD, por exemplo, é de 44,1kHz.

2.1.3 Estrutura hierárquica

A informação contida em um vídeo é dotada de uma organização temporal hierárquica intrínseca, útil a diversos serviços de análise e processamento de vídeo (SIDIROPOULOS *et al.*, 2011; BARALDI; GRANA; CUCCHIARA, 2015a; ROTMAN; PORAT; ASHOUR, 2017). A menor das unidades de interesse dos sistemas de análise de vídeo é o quadro, definido anteriormente na Seção 2.1.1, como uma das imagens de uma sequência de imagens que compõem a imagem em movimento de um vídeo. Os quadros se organizam em unidades maiores cuja origem da definição vem da própria sistemática usual de

criação de um vídeo, onde são concatenados múltiplos trechos de vídeo capturados individualmente para compor o vídeo completo. Cada uma destas unidades é chamada **tomada** e é definida como uma sequência de quadros capturados ininterruptamente por uma única câmera (BRUNELLI; MICH; MODENA, 1999; KOPRINSKA; CARRATO, 2001). A Figura 9 apresenta um exemplo de trecho de vídeo contendo duas tomadas, extraídas do filme *open content* chamado *Tears of Steel*⁶.

Figura 9 – Par de tomadas adjacentes.

Nesta figura há duas tomadas, uma de índice 65, que é uma captura do rosto de uma atriz falando algo a um outro ator, oculto na captura, e uma de índice 66, que é uma captura lateral de ambos os atores interagindo. A decomposição de um vídeo em tomadas, apesar de servir a uma ampla gama de serviços de análise de vídeo, ainda é insuficiente em ocasiões, como por exemplo, onde se faz necessário processar individualmente trechos que contém conceitos semânticos completos (HANJALIC; LAGENDIJK; BIEMOND, 1999; SIDIROPOULOS *et al.*, 2011). No exemplo da Figura 9, caso fosse necessário processar o trecho do filme que ilustra toda a interação entre o par de atores, seria insuficiente conhecer as fronteiras entre tomadas. Todas as tomadas relacionadas àquela interação deveriam ser processadas. Esta é a principal motivação para que exista uma decomposição do vídeo em unidades determinadas pela semântica.

Em um filme, por exemplo, podem haver informações que codificam diálogos entre pessoas, perseguições, atividades de ação, dentre outras unidades semânticas. Estas diferentes unidades semânticas, em geral, se encontram dispostas separadamente, ao longo do eixo temporal do vídeo. Considerando que dificilmente há mudanças de semântica dentro de uma tomada, a decomposição do vídeo em unidades que diferem entre si pela semântica, costuma ser realizada agrupando tomadas em unidades maiores (FABRO; BÖSZÖRMENYI, 2013; ROTMAN; PORAT; ASHOUR, 2017), denominadas **cenas**. A definição do conceito de cena, entretanto, não é consensual entre os autores de trabalhos da área de segmentação temporal de vídeo. Há, na literatura, diversas definições do que é uma cena (FABRO; BÖSZÖRMENYI, 2013) e, apesar da maioria das diferentes definições disponíveis caracterizarem cena como uma sequência de tomadas adjacentes relacionadas, cada uma delas adota um critério distinto para a relação entre tomadas. Alguns autores

⁶ Disponível em <<https://mango.blender.org/>>.

propuseram definições de cena baseadas em conceitos objetivos e de fácil análise como consistência temporal de cromaticidade/iluminação/som ambiente (SUNDARAM; CHANG, 2002), conteúdo visual semelhante (HANJALIC; LAGENDIJK; BIEMOND, 1999) e semelhança entre plano de fundo (cenário) (CHEN; LAI; LIAO, 2008). Há também casos de autores que propuseram definições de cena atreladas a domínios muito específicos como, por exemplo, documentários artísticos produzidos na União Soviética nas décadas de 20 e 30 (MITROVIĆ *et al.*, 2010), vídeos de casamento (SAWAI *et al.*, 2011) e vídeos de performances circenses (IWAN; THOM, 2017). Estas definições de cena, apesar de facilitarem a tarefa de segmentação em cenas em seus respectivos domínios, são inadequadas para aplicações voltadas ao processamento de vídeos independentemente de seus domínios.

Uma definição bastante ampla do conceito de cena foi proposta por Saraceno e Leonardi (1997). Para eles, cena é uma sequência de uma ou mais tomadas consecutivas e semanticamente correlacionadas. Esta definição, por ser altamente subjetiva, atende à concepção de cena dos usuários em geral de sistemas de vídeo, em que, uma mudança de assunto corresponde a uma mudança de cena. Para que seja possível segmentar vídeo em cenas por meio do agrupamento de tomadas por semântica, é necessário extraí-la dos dados brutos que codificam tomadas, que são valores, por exemplo, de componentes de cores e amplitudes de ondas sonoras. Esta necessidade submete qualquer tentativa computacional e automática de segmentação de vídeos em cenas à **Lacuna Semântica** (SMEULDERS *et al.*, 2000), caracterizada como a falta de coincidência entre a informação codificada de um vídeo e a interpretação desta informação por um espectador deste vídeo. A lacuna semântica é uma das maiores dificuldades existentes na área de segmentação automática de vídeo em cenas.

A definição de Saraceno e Leonardi (1997) é amplamente adotada em trabalhos de segmentação temporal de vídeo em cenas e é independente de domínio, justificando sua adoção nesta tese. A Figura 10 ilustra os conceitos de tomada e de cena definidos, bem como sua relação.

Figura 10 – Relação entre tomadas e cenas.

A Figura 10 apresenta um vídeo k dotado de m tomadas e n cenas. As tomadas são representadas pelos retângulos coloridos e os conjuntos de retângulos de mesma cor representam as cenas. A coloração das tomadas serve para representar a semântica: tomadas de mesma cor são semanticamente relacionadas.

2.2 Características de vídeo

Um sistema de recuperação de vídeo deve ser capaz de indexar os vídeos de sua coleção pelos seus respectivos conteúdos (MARQUES; FURHT, 2002). Graças ao grande volume de dados e à complexidade das informações que compõem os vídeos, a indexação é realizada por meio de alguma propriedade discriminatória, chamada **característica** (ATREY *et al.*, 2010). As características de vídeo se dividem em três categorias, de acordo com seu nível de abstração (MARQUES; FURHT, 2002; MARTINET; SAYAD, 2012): baixo, médio e alto nível semântico. Características de baixo nível semântico são leituras numéricas computadas diretamente dos dados crus, sem que haja o emprego de conhecimento externo, aprendizado de máquina ou análise estatística de outros documentos (MARTINET; SAYAD, 2012). Um exemplo de característica de baixo nível semântico são as informações de cor de uma imagem no espaço de cores RGB. Tais informações são computadas como uma série de números inteiros no intervalo $[0 - 255]$, para os componentes vermelho, verde e azul.

As características de médio nível semântico são definidas como leituras numéricas ou simbólicas produzidas por uma etapa de refinamento semântico em características de baixo nível semântico (MARQUES; FURHT, 2002; BOUREAU *et al.*, 2010; MARTINET; SAYAD, 2012). Exemplos de características de médio nível semântico são as representações produzidas pelos métodos *Bag-Of-Features* (BoF) (SIVIC; ZISSERMAN, 2003), *Spatial Pyramids* (LAZEBNIK; SCHMID; PONCE, 2006), bem como aquelas extraídas das camadas superiores de redes neurais convolucionais (LECUN; BENGIO, 1998). Já as características de alto nível semântico representam o significado das informações contidas em um vídeo, diretamente interpretáveis por humanos (MARQUES; FURHT, 2002; MARTINET; SAYAD, 2012). Elas se apresentam, costumeiramente, por meio de texto descrevendo o conteúdo. Por exemplo, um vídeo pode ter a ele associados os rótulos “carro” e “moto”, indicando que tal vídeo ilustra um carro e uma moto. Estes rótulos podem também representar conceitos não materiais do vídeo como, por exemplo, “raiva” ou “tranquilidade”. As características de alto nível semântico podem ser geradas por anotação manual ou automaticamente por análise computacional do conteúdo (a partir de características de baixo/médio nível semântico), sujeita à lacuna semântica.

As características de baixo e de médio nível semântico extraídas de um vídeo são codificadas em vetores numéricos n -dimensionais, chamados **vetores de características** (MARQUES; FURHT, 2002; MARTINET; SAYAD, 2012). Os vetores de características, representações obtidas por meio de processamento e análise computacional, servem para realizar comparação entre vídeos e se caracterizam como representações compactas e relevantes do conteúdo. O processo de indexar automaticamente uma base de vídeos por suas características é chamado **extração de características** (MARQUES; FURHT, 2002). A uma ferramenta que computa um vetor de características de um canal de in-

formação é dado o nome de detector ou extrator de características (TUYTELAARS; MIKOLAJCZYK, 2008).

2.2.1 Extração de características de baixo nível semântico

As características de baixo nível semântico costumam ser extraídas da imagem em movimento e do áudio dos vídeos. Esta seção descreve dois extratores de características de baixo nível semântico que foram importantes no desenvolvimento do trabalho.

Extração de características visuais

As características extraídas da imagem em movimento são, muitas vezes, as mesmas utilizadas em imagens estáticas, já que os quadros são imagens estáticas. Características de imagens estáticas se dividem em globais e locais, de acordo com seu escopo de representação. As características globais se referem à imagem inteira e normalmente são compactas. Uma das mais populares características globais de comparação de conteúdo visual é o histograma de cor (ZHANG; KANKANHALLI; SMOLIAR, 1993; STOCKMAN; SHAPIRO, 2001). Um histograma de cor de uma imagem é uma contagem dos pixels, por cor, de todos os pixels dessa imagem. Ele pode ser implementado como um vetor unidimensional onde cada índice corresponde a um intervalo de cores e o conteúdo de cada célula corresponde ao total de pixels da imagem que se encaixam no intervalo relacionado àquele índice. A obtenção de histogramas de cor tem baixo custo computacional aliado ao fato de que eles são relativamente invariantes a diferenças de translação, rotação centralizada no eixo de captura, pequenas rotações não centralizadas no eixo de captura, mudanças de escala e oclusão parcial (STOCKMAN; SHAPIRO, 2001). Como os histogramas de cor desconsideram informação espacial dos pixels, eles falham quando precisam diferenciar imagens muito diferentes, com cores semelhantes. Além da indiferença à informação espacial, histogramas de cor falham ao comparar duas imagens com conteúdo parecido, mas que foram capturadas em condições de iluminação muito diferentes (LU; DREW; AU, 2001). Histogramas de cor estão presentes em muitos trabalhos de segmentação temporal de vídeo em cenas como Kender e Yeo (1998), Chen e Li (2010), Sidiropoulos *et al.* (2011) e Rotman, Porat e Ashour (2017) e de classificação de vídeos como Iyengar e Lippman (1997), Lu, Drew e Au (2001) e Lin e Hauptmann (2002).

Uma característica local, diferentemente de uma global, se refere a um subconjunto de uma imagem. Este subconjunto pode ser um ponto, região ou mesmo um segmento de aresta contendo um padrão que o diferencia de sua vizinhança (TUYTELAARS; MIKOLAJCZYK, 2008). Espera-se, com a extração de características locais, representar estruturas distintivas de uma imagem. De acordo com Grauman e Leibe (2011), um bom extrator de características locais deve garantir repetibilidade e precisão, extraíndo sem-

pre as mesmas características de duas imagens distintas que exibem o mesmo objeto e, ao mesmo tempo, oferecer distintividade, a habilidade de distinguir dois objetos diferentes.

O método de extração de características locais *Scale-Invariant Feature Transform* (SIFT), proposto por Lowe (2004), é um dos mais conhecidos atualmente (DU; IKENAGA, 2019; IONESCU; POPESCU, 2016). O artigo que o apresenta é apontado, pelo sistema CiteSeerX (2015), como o quarto mais citado de toda a Ciência da Computação. O método SIFT foi apresentado como uma combinação do detector de regiões de interesse por diferenças de gaussianas, do inglês *Difference-of-Gaussians* (DoG), e de uma técnica para produzir um vetor de características. O vetor de características SIFT se mostrou útil quando combinado com outros detectores de regiões de interesse (MIKOLAJCZYK; SCHMID, 2005). As características extraídas pelo método SIFT são invariantes a escala e a rotação. O método SIFT foi descrito originalmente para aplicação em imagens em escala de cinza.

O detector de regiões de interesse baseia-se no fato de que pontos de máximo de laplacianas de gaussianas, do inglês *Laplacian-of-Gaussians* (LoG), são equivalentes a arestas e/ou cantos de uma imagem. O cálculo de LoG em uma imagem é computacionalmente custoso e, por isso, o SIFT usa a DoG como uma aproximação para a LoG. A DoG é basicamente uma subtração envolvendo uma imagem e uma versão suavizada dela mesma. Essa operação realça bordas e alguns detalhes presentes na imagem e, ao mesmo tempo, descarta outros detalhes e ruído. A Figura 11 exemplifica uma operação de DoG em uma imagem.

Figura 11 – Aplicação de filtro DoG.

Na Figura 11, a imagem original (i) e sua versão desfocada (ii) são submetidas a uma operação de subtração, originando a imagem (iii), que contém apenas as bordas e

detalhes identificados pelo filtro DoG. A Figura 12 apresenta os passos iniciais do método SIFT.

Figura 12 – Cálculo de oitavas no SIFT.

Para cada imagem, o SIFT computa três cópias redimensionadas sucessivamente, onde cada cópia tem dimensões correspondentes a metade da cópia anterior. As diferentes escalas são chamadas oitavas e sua obtenção compõe a estratégia responsável pela invariância à escala. Para cada oitava são computadas quatro cópias suas com suavização gaussiana, aumentada sequencialmente em cada cópia. A DoG é, então, computada entre as imagens de cada oitava e a imagem seguinte com maior grau de suavização da mesma oitava. Pontos aproximados de máximo e mínimo são encontrados, realizando uma busca em vizinhanças $3 \times 3 \times 3$ na matriz tridimensional resultante da combinação das matrizes bidimensionais DoG vizinhas. Como estes pontos são aproximações e os máximos/mínimos reais podem estar em regiões subpixel, seus valores são obtidos por meio da expansão de Taylor (ARFKEN; WEBER, 2012). Os pontos computados são chamados pontos chave.

A quantidade de pontos chave produzida é grande e muitos deles não correspondem a regiões interessantes. Por este motivo, são realizadas duas etapas de filtragem, rejeitando pontos com baixo contraste e depois aqueles com alta resposta de aresta em uma única direção usando determinante do hessiano, do inglês *Determinant of Hessian* (DoH) (BEAUDET, 1978). Depois das etapas de filtragem, inicia-se a construção do vetor de características, ao se determinar a orientação mais proeminente dos gradientes de direção em torno de cada ponto-chave. Essa orientação é atribuída ao ponto chave e todo cálculo posterior será relativo a ele, permitindo a invariância à rotação. A orientação é obtida por meio de um histograma de 36 posições, onde cada posição cobre 10 graus e tem como valor a soma das magnitudes dos gradientes. O maior pico, assim como os outros

picos com valor de pelo menos 80% dele, são transformados em pontos chave individuais onde a orientação é dada pelo valor da posição daquele pico no histograma.

O último passo do método SIFT consiste em gerar uma assinatura para cada ponto chave, que é o próprio vetor de características. O vetor de características é obtido pela inspeção de uma janela 16×16 em volta do ponto chave. Essa janela é dividida em dezesseis janelas 4×4 e cada janela produz um histograma de 8 posições de magnitudes de gradientes, onde cada posição do histograma cobre 45 graus. O conteúdo adicionado a cada posição do histograma também é proporcional à distancia do ponto chave. Fazendo isso para cada uma das dezesseis regiões, um vetor de características de $16 \times 8 = 128$ bytes é produzido. Esse vetor de características é normalizado, dele é subtraída a orientação do ponto chave e por fim é limiarizado para 0,2. A subtração da orientação do ponto chave serve para prover independência à rotação. Já a limiarização provê independência à diferenças de iluminação. Um exemplo de extração de pontos chave por um programa Java/OpenImaj⁷ pode ser visto na Figura 13.

Figura 13 – Extração de pontos chave SIFT usando a API OpenImaj.

A Subfigura 13a, contém uma foto de um casaco com um logotipo impresso. A Subfigura 13b contém uma segunda foto do mesmo casaco da Subfigura 13a, de um ponto de vista diferente, onde o casaco aparece rotacionado. As Subfiguras 13c e 13d indicam a posição dos pontos chave SIFT extraídos das Subfiguras 13a e 13b.

⁷ OpenImaj é uma API (*Application Programming Interface*) de *The University of Southampton*, disponível em <www.openimaj.org>

Uma demonstração de como o SIFT é repetível e robusto pode ser visto na Figura 14, que apresenta uma correspondência entre os pontos chave das Subfiguras 13c e 13d, calculada com distância euclidiana.

Figura 14 – Correspondência de características SIFT.

O método SIFT tem sido empregado em diversos trabalhos de segmentação temporal de vídeo em cenas como [Tapu e Zaharia \(2011\)](#), [Lopes, Trojahn e Goularte \(2014\)](#) e [Baraldi, Grana e Cucchiara \(2015c\)](#), bem como em trabalhos de classificação de vídeos, como [Zhou *et al.* \(2008\)](#), [Jhuo *et al.* \(2014\)](#) e [Sikka e Sharma \(2018\)](#).

Extração de características aurais

As características aurais, de maneira análoga às características visuais, representam propriedades específicas do sinal de áudio em um formato compacto, possibilitando a comparação com outros sinais de áudio. Um dos mais famosos extratores de características aurais, o *Mel-Frequency Cepstral Coefficients* (MFCC), foi introduzido por [Davis e Mermelstein \(1980\)](#) e era destinado à tarefa de reconhecimento de palavras monossilábicas. Mais tarde, o MFCC se mostrou aplicável em tarefas como o reconhecimento de locutores, integrando o estado da arte em análise de áudio atual ([YANG *et al.*, 2019](#)).

O MFCC foi desenvolvido para mimetizar o comportamento da percepção de fala humana. A frequência percebida por um humano é diferente do que é emitido pela fonte. A mimetização é realizada pelo uso da Escala Mel ([STEVENS; VOLKMANN; NEWMAN, 1937](#)), uma função da frequência para valores mel, que são unidades obtidas pela percepção por ouvintes de frequências consideradas como iguais, em distância, uma da outra. A extração de características MFCC é iniciada pela divisão do sinal de áudio em quadros. O sinal nestes quadros é convertido para o domínio de frequências por meio de uma transformada discreta de Fourier ([OPPENHEIM; SCHAFER, 2009](#)) e um espectro de frequências mel é computado sobre estas frequências. O espectro de frequências mel é

submetido a uma função logarítmica, para imitar a percepção humana de volume, também logarítmica. O próximo passo consiste na remoção de características dependentes do locutor por meio do cálculo de coeficientes cepstrais. Este passo se baseia no fato de que a fala humana pode ser modelada por um modelo fonte-filtro (FANT, 1970) que descreve a produção de fala como um sinal e um filtro. O cepstro, que é o espectro de um espectro, tem como objetivo suprimir o sinal fonte, mantendo apenas o sinal de filtro. Ele é obtido pela aplicação de uma transformada discreta de cosseno no sinal, mantendo os coeficientes de ordem baixa, que apresentam a resposta de frequência do trato vocal. Os coeficientes de ordem alta que apresentam a resposta de frequência do sinal de origem são descartados. Os vetores MFCC típicos mantêm 13 coeficientes cepstrais.

Apesar de amplamente utilizado, o MFCC possui limitações e é passível de melhora (SHARMA; ALI, 2015). Uma das mais problemáticas é a falta de significado de seus valores, já que apenas os dois primeiros coeficientes possuem significado determinado. O primeiro é a média da energia de todas as bandas de frequência e o segundo é o balanço de componentes de baixa e alta frequência do quadro de sinal. Os outros 11 coeficientes contêm detalhes do espectro. O MFCC também é bastante sensível à presença de ruído no sinal, enfrentando uma degradação de performance severa nesse tipo de situação. Outro problema é a presunção de que os coeficientes cepstrais da frequência fundamental são mais baixos do que dos componentes de frequência da mensagem linguística, que para alguns locutores do sexo feminino é falsa. O MFCC está presente em métodos de segmentação temporal de vídeo em cenas como, por exemplo, Lopes, Trojahn e Goularte (2014), Baraldi, Grana e Cucchiara (2017) e Rotman, Porat e Ashour (2017), e também em métodos de classificação de vídeo como Jhuo *et al.* (2014), Kurcius e Breckon (2014) e Dammak e Benayed (2017).

2.2.2 Extração de características de médio nível semântico

Características de médio nível semântico são produzidas a partir da análise de informações de baixo nível semântico que descrevem trechos do vídeo que se pretende representar, identificando padrões que permitam estruturá-las em representações de algum aspecto da semântica contida no vídeo. Estes aspectos são codificados em vetores de características por valores numéricos, sem que haja uma tentativa de classificá-los em conceitos de alto nível semântico (rótulos, descrições textuais, etc). Apesar de não serem classificados, padrões presentes em vetores de características de médio nível semântico permitem a comparação com outros vetores de características do mesmo tipo quanto a algum aspecto da semântica. A semântica contida em vetores de médio nível é chamada **semântica latente**, pelo fato de ser uma representação de semântica que, mesmo sem ser descrita em alto nível, permite a comparação com outros vetores de características. Nos próximos parágrafos, é apresentado um dos mais conhecidos métodos de produção de

vetores de características de médio nível semântico, o BoF.

O método BoF é uma adaptação do *Bag-of-Words* (BoW), um método oriundo da área de Processamento de Linguagem Natural, cuja função é gerar representações simplificadas de textos (AGGARWAL; ZHAI, 2012). No método BoW, um texto é representado por um vetor que contém contagens de ocorrências de suas palavras, de acordo com um dicionário. Com isso, dois documentos podem ser comparados por seus vetores BoW gerados com base em um dicionário comum, usando uma medida de distância entre vetores. Os primeiros trabalhos do método BoF, análogo do BoW aplicado em outros tipos de informação, são da área de Visão Computacional e datam do início deste milênio. Neles, os textos foram substituídos por imagens e as palavras por *textons*, vetores que representam texturas (CULA; DANA, 2001; LEUNG; MALIK, 2001). Mais tarde, Csurka *et al.* (2004) propuseram o *Bag-of-Keypoints*, também destinado à comparação de imagens, que utilizava centroides de grupos de pontos chave SIFT no lugar dos *textons*. Muitos autores utilizam o termo *Bag-of-Visual-Words* (BoVW), quando referenciam um método BoF orientado a conteúdo visual já que no método os padrões distintivos representados pelos centroides são análogos às palavras de um texto. O método BoF também foi estendido para outras naturezas de dados como, por exemplo, dados aurais, no *Bag-of-Aural-Words* (BoAW) proposto por Carletti *et al.* (2013).

Apesar de existirem variações em determinadas aplicações, é possível enquadrar os métodos BoF num arcabouço genérico para comparar dois ou mais documentos de qualquer natureza. O método BoF se aplica em cenários de comparação com dois ou mais documentos onde há múltiplos vetores de características por documento, onde cada um destes vetores de características descreve um trecho do documento. Exemplos destes vetores de características são os produzidos pelo SIFT, que representam determinados pontos em uma imagem, ou os produzidos pelo MFCC, que representam a percepção por um humano de trechos do áudio original.

A primeira etapa do método BoF se dá pela reunião, em um único conjunto, de todos os vetores de características de todos os documentos envolvidos na produção da representação. Este conjunto passa por um agrupamento, cujo objetivo é identificar padrões, por meio dos grupos produzidos. Espera-se que os vetores de características que compõem um padrão sejam incluídos em um mesmo grupo neste processo. Estes padrões podem ter qualquer documento como origem. Para representar toda a informação de cada grupo em um único vetor, é escolhido, para cada um dos grupos, o centróide ou o medóide, que recebe o nome de **palavra**. O conjunto de palavras produzido é chamado **dicionário**, e corresponde a um conjunto com representações compactas de todos os padrões identificados no documento.

Dado o dicionário, é realizado um mapeamento de todos os vetores de características de cada documento nas palavras do dicionário, por similaridade. Para cada documento

são então contabilizados quantos vetores de características correspondem a cada palavra do dicionário. Isso equivale a contabilizar as ocorrências dos padrões que aquelas palavras representam nos documentos. O conjunto das ocorrências de palavras nos documentos é organizado em um vetor chamado histograma de ocorrências de palavras. Dois documentos podem ser comparados, por meio de alguma medida de distância/similaridade entre seus histogramas de ocorrências de palavras. Esta comparação produzirá um valor que indica o quão próximos/distantes dois documentos são, de acordo com os padrões que contém. Uma descrição em alto nível do arcabouço BoF para comparação de dois documentos é apresentada no Algoritmo 1.

Algoritmo 1 – Arcabouço da técnica de comparação de documentos por BoF

COMPARA_BOF(a, b, k, δ): recebe um par de documentos a e b , um tamanho de dicionário k e uma distância de similaridade entre vetores como entrada. Devolve um valor d_{bof} que corresponde à similaridade entre a e b .

- 1: Extraia múltiplos vetores de características de a e de b , e armazene-os em conjuntos de vetores de características v_a e v_b , respectivamente;
 - 2: $D \leftarrow v_a \cup v_b$;
 - 3: Agrupe os elementos do conjunto D em k grupos, obtendo um conjunto de grupos W ;
 - 4: Calcule, para cada grupo $w \in W$, um vetor de características que melhor represente os elementos de w . O conjunto resultante destes vetores é o dicionário *Dict*;
 - 5: Calcule os histogramas h_a e h_b , classificando cada vetor de v_a e de v_b , respectivamente, em uma palavra de *Dict* e contabilizando quantas ocorrências de cada uma destas palavras houve em cada documento;
 - 6: **retorna** $\delta(h_a, h_b)$
-

As Linhas 1 e 2 do Algoritmo 1, descrevem a extração de características dos dois documentos e sua fusão em um conjunto único. O passo descrito na Linha 3 do algoritmo serve para encontrar padrões entre todos os vetores de características de a e de b . A instrução presente na Linha 4 do algoritmo é responsável por definir uma representação única para cada um dos padrões identificados no passo anterior. A determinação do vetor de características que melhor representa um elemento de um grupo normalmente se dá por meio do cálculo de um centróide ou de um medóide. O conjunto de representações únicas corresponde a um dicionário de padrões encontrados nos documentos. Na etapa descrita na Linha 5 é realizada uma contagem de quantos vetores de características, do total de um documento, são mais próximos a cada um dos elementos de *Dict*. O resultado dessa contagem é o histograma de ocorrência de padrões. Por fim, a Linha 6 contém o cálculo de distância entre os vetores de ocorrência de padrões. É esperado que, nesta etapa, documentos com padrões semelhantes sejam considerados semelhantes.

2.3 Fusão de modalidades

O conceito de multimodalidade tem suas origens na teoria da comunicação. A definição de multimodalidade em vídeo, de acordo com [Snoek e Worring \(2002\)](#) é⁸:

A capacidade do autor de um documento de vídeo de expressar uma idéia semântica predefinida, pela combinação de um layout com um conteúdo específico, usando ao menos dois canais de informação.

Conforme discutido no Capítulo 1, com situações exemplificadas pelas Figuras 2 e 3, os sistemas de análise e processamento de conteúdo de vídeo devem ser multimodais, para atender à natureza multimodal dos vídeos. O emprego de multimodalidade em um sistema exige a definição de algum tipo de estratégia para combinar a informação proveniente de diferentes fontes. A combinação da informação de diferentes modalidades é chamada fusão multimodal.

De acordo com [Atrey et al. \(2010\)](#), a fusão multimodal é realizada, geralmente, em dois níveis. Um destes níveis é a **fusão tardia**, também conhecida como fusão no nível de decisão. O arcabouço de fusão tardia é apresentado na Figura 15, inspirada em [Atrey et al. \(2010\)](#).

Figura 15 – Arcabouço da fusão tardia.

Na fusão tardia, há uma etapa de decisão para cada uma das n modalidades ou mesmo diferentes tipos de características de mesmas modalidades, representados pelos retângulos cinza na Figura 15. Estas n decisões individuais são realizadas por n módulos de decisão, representados por paralelogramos de cor rosa na Figura 15. As n etapas de decisão produzem n respostas parciais à tarefa, ilustradas na Figura 15 pelas elipses azuis. As respostas parciais são então combinadas em uma resposta consensual por um módulo de fusão de decisões, representado na Figura 15 pelo quadrado amarelo. A resposta final é representada na Figura 15 por um círculo verde.

Uma das vantagens da fusão tardia é a facilidade em combinar informações heterogêneas já que cada diferente representação proveniente de uma diferente modalidade dará origem a uma resposta parcial. Combinar as respostas parciais é relativamente simples

⁸ Citação traduzida do inglês.

considerando que todas as respostas têm a mesma representação. A fusão tardia também oferece relativa facilidade de inclusão/remoção de modalidades no processo pelo mesmo motivo. Outra vantagem da fusão tardia é a possibilidade de escolha de diferentes métodos de decisão para as diferentes modalidades. É possível, não somente escolher o método mais adequado para cada modalidade, como também parametrizá-lo para melhor aproveitar a informação unimodal. A fusão tardia, no entanto, é incapaz de utilizar qualquer correlação existente entre as diferentes modalidades antes de uma etapa de decisão, podendo levar à perda de informação importante para esta etapa. Além disso, por requerer múltiplas etapas de decisão individuais, torna-se computacionalmente cara.

O segundo nível de fusão multimodal existente é a **fusão prévia**, também conhecida como fusão no nível de características. A fusão prévia é ilustrada na Figura 16, inspirada em [Atrey *et al.* \(2010\)](#).

Figura 16 – Arcabouço da fusão prévia.

Em uma abordagem fusão prévia, há uma combinação dos vetores de características de n modalidades, representados na Figura 16 por retângulos de cor cinza, produzindo novos vetores de características que contêm combinações das informações dos vetores de características de origem. A ferramenta que efetua esta combinação é chamada módulo de fusão e é representada pelo quadrado de cor amarela da Figura 16. Os vetores com informação combinada, chamados vetores de características fundidas, são ilustrados na Figura 16 por um paralelogramo azul. Os vetores de características fundidas são então fornecidos ao módulo de decisão, representado na Figura 16 pelo hexágono de cor rosa, que produz a resposta esperada, representada na Figura 16 pelo círculo de cor verde, de acordo com a tarefa de vídeo. Em um sistema de segmentação temporal de vídeo, por exemplo, o módulo de decisão utiliza os vetores de características fundidas para encontrar as fronteiras entre os segmentos de vídeo.

A fusão prévia possibilita a identificação e o uso de correlação entre as diferentes características em uma etapa inicial, viabilizando a extração de melhores recursos para a execução da tarefa ([ATREY *et al.*, 2010](#)). Outra vantagem da fusão prévia é a necessidade de uma única etapa de decisão que utiliza o vetor de características fundidas como entrada, dispensando o processamento individual dos múltiplos vetores unimodais. Isso se traduz em um ganho de eficiência, caso o vetor de características fundidas seja pequeno em relação ao tamanho de todos os vetores de características unimodais juntos. Por fim, a

complexidade em parametrizar e ajustar um único método de decisão é consideravelmente menor do que fazer o mesmo para n modalidades.

Apesar de suas vantagens, a fusão prévia é difícil de ser realizada. Os motivos pelos quais sua aplicação é um problema são chamados nesta tese de **dilemas de projeto** ou simplesmente **dilemas**. Um dos dilemas da fusão prévia é a dificuldade de identificação de correlação entre características. Isso acontece porque as características de diferentes modalidades tem diferentes representações. Como exemplo, um determinado vetor de características pode carregar magnitudes de gradientes em suas células enquanto um segundo vetor de características de origem diferente carrega coeficientes que caracterizam um espectro sonoro. Outro dilema existente é a dificuldade de identificação de correlação entre características intermodais não sincronizadas temporalmente. Um padrão visual, por exemplo, pode ser complementar a um padrão aural que ocorre em um instante de tempo diferente. O terceiro, e último, dilema diz respeito a ausência de um modelo bem estabelecido de representação única de informação multimodal. Não há um protocolo, modelo ou padrão adequado para representar a informação combinada proveniente de múltiplas modalidades em uma única estrutura de dados compacta.

2.4 Métodos para tarefas de análise de vídeo

As tarefas de análise de vídeo, em geral, são executadas em duas etapas: a de extração/representação do conteúdo a ser analisado e a de execução de um método específico de computação de resposta à tarefa em questão. Esta seção aborda os métodos de computação de resposta para duas das mais comuns tarefas de análise de vídeo: segmentação temporal de vídeo em cenas, na Seção 2.4.1, e classificação de vídeo, na Seção 2.4.2.

2.4.1 Detecção de fronteiras entre cenas

A maioria dos métodos de segmentação temporal de vídeo em cenas baseia-se em agrupamento de tomadas (FABRO; BÖSZÖRMENYI, 2013; ROTMAN; PORAT; ASHOUR, 2017) fazendo com que, em geral, eles possuam uma etapa bem definida que consiste em determinar as sequências de tomadas que são relacionadas. Há diferentes abordagens para o agrupamento de tomadas, como Modelos Ocultos de Markov (em inglês *Hidden Markov Models* (HMM)) (YAŞAROĞLU; ALATAN, 2003; HUANG; LIU; WANG, 2005), alinhamento de sequências (CHASANIS; LIKAS; GALATSANOS, 2009), gramática cinematográfica (CHEN; OZSU, 2002; TAVANAPONG; ZHOU, 2004) e técnicas baseadas em Grafos (YEUNG; YEO; LIU, 1998; SIDIROPOULOS *et al.*, 2011).

HMMs são modelos estocásticos e os métodos de detecção baseados em HMM funcionam, em geral, pela construção de um modelo com estados que correspondem a rótulos de tomadas. Esses rótulos modelam algum aspecto útil para o agrupamento das

tomadas como, por exemplo, classes de conteúdo semântico: diálogo, não diálogo, ação, *etc.* Os estados são ligados por transições associadas a valores de probabilidades que são computados por uma etapa de treinamento realizada com vídeos previamente anotados. Dados um modelo treinado e um vídeo não anotado, utiliza-se o algoritmo de Viterbi (VITERBI, 1967) para determinar a sequência de estados que rotulam cada tomada daquele vídeo com base no modelo, tal que a probabilidade daquela sequência de rótulos ter sido gerada pelo modelo é máxima. No final, a sequência de rótulos é traduzida em uma lista de posições de fronteiras entre cenas, agrupando tomadas adjacentes rotuladas com a mesma classe ou com classes diferentes que podem compor um determinado tipo de cena. Os métodos baseados em HMM são bastante precisos (FABRO; BÖSZÖRMENYI, 2013). Em contrapartida, esses métodos são limitados a um domínio de vídeos restrito pelo conjunto de treinamento.

Nas abordagens baseadas em alinhamento de sequências, as tomadas de um vídeo para o qual se deseja determinar as fronteiras entre cenas são agrupadas por algum critério de semelhança como, por exemplo, conteúdo visual. Cada grupo computado é associado a um rótulo distinto. Os rótulos dos grupos são atribuídos individualmente a cada tomada na sequência original de tomadas do vídeo usando o mesmo critério de semelhança empregado no agrupamento, produzindo uma sequência de rótulos. Essa sequência é analisada em busca de subsequências que representam padrões de ocorrência de rótulos. Essa busca é motivada pelos fatos de que um padrão de ocorrências de rótulos deve corresponder a uma cena e de que duas cenas distintas apresentam diferentes padrões de ocorrência de rótulos. A análise é realizada por meio do método de alinhamento de sequências global (NEEDLEMAN; WUNSCH, 1970), que é aplicado nessa situação para computar um valor de similaridade entre um par de sequências dadas como entrada. A análise é realizada por meio de um par de janelas deslizantes vizinhas de mesmo comprimento. A cada iteração as subsequências delimitadas pelo par de janelas são alinhadas e a similaridade deste alinhamento é analisada. Alta similaridade indica que ambas as subsequências integram uma mesma cena. Baixa similaridade indica que cada uma das subsequências contém padrões de uma cena diferente e que, conseqüentemente, há uma fronteira entre cenas ali. Os métodos baseados em alinhamento de sequências são independentes de domínio e não demandam dados/processamento para treinamento. Por outro lado, sua eficácia depende da parametrização do tamanho da janela. Além disso, falham em situações onde a mudança de conteúdo entre cenas é gradual.

Os métodos de detecção baseados em gramática cinematográfica se baseiam em regras de composição de cenas adotadas por diretores de vídeo. Estas regras são conhecidas como gramática cinematográfica ou regras de edição de filmes (FABRO; BÖSZÖRMENYI, 2013). Um exemplo de regra dessa natureza é a regra **tomada/tomada reversa**, que determina a composição de uma cena por dois tipos de tomadas que são exibidos alternadamente alternadas. A regra tomada/tomada reversa costuma ser usada em cenas de

diálogo entre dois personagens, com a câmera focalizada no personagem que está falando em determinado momento. Nesses métodos é necessária a escolha de um conjunto de tipos de tomadas e de tipos de padrões (regras de edição) que se deseja identificar no vídeo. Definidos os tipos de tomadas e padrões, classifica-se cada tomada do vídeo para o qual se deseja determinar as fronteiras entre cenas em uma das classes de tomadas pré-determinadas. Dada a sequência resultante desse processo de rotulamento, é realizada uma varredura nesta sequência, detectando as fronteiras onde a regra de edição muda. Essas fronteiras correspondem à predição de posicionamento das fronteiras entre cenas. Os métodos baseados em gramática cinematográfica são de simples implementação, eficientes e eficazes em domínios onde as regras de edição de filmes foram seguidas com rigor. Por outro lado, eles falham sempre que duas cenas adjacentes seguem o mesmo subconjunto de regras de edição. Além disso, há vídeos onde os diretores ignoram intencionalmente as regras de edição para causar alguma sensação específica no espectador.

Há vários métodos de detecção de fronteiras entre cenas baseados em grafos. Nos métodos dessa classe, as tomadas são agrupadas por algum critério de similaridade ou distância temporal e organizadas em uma representação por grafos, onde os nós representam tomadas ou grupos de tomadas e as arestas indicam alguma relação entre os nós conectados. As posições de fronteiras entre tomadas são preditas pela aplicação de algum algoritmo de segmentação em grafos, dividindo o grafo original em subgrafos que representam cenas. Um dos mais conhecidos é o método Grafo de Transição de Cenas (do inglês *Scene Transition Graph* (STG)), proposto em um dos trabalhos seminais de segmentação temporal de vídeo em cenas (YEUNG; YEO; LIU, 1998). Este método é explicado em detalhes por figurar como parte do ferramental experimental deste trabalho.

No método STG, as tomadas são agrupadas pelo algoritmo de agrupamento hierárquico *complete linkage*, modificado para evitar a inclusão em um mesmo grupo, de tomadas temporalmente distantes no vídeo, mesmo que sejam similares visualmente. Essa peculiaridade do agrupamento modificado é propiciada pelo cálculo de distância $\hat{d}_{max}$ entre dois grupos C_i e C_j dada por

$$\hat{d}_{max}(C_i, C_j) = \max_{x \in C_i, y \in C_j} \hat{d}(x, y), \quad (2.1)$$

onde

$$\hat{d}(S_i, S_j) = \begin{cases} d(S_i, S_j), & \text{se } d_t(S_i, S_j) \leq T; \\ \infty, & \text{caso contrário.} \end{cases} \quad (2.2)$$

Na Equação 2.2, as variáveis S_i e S_j representam tomadas i e j , d é medida de distância adotada entre histogramas, d_t é a distancia temporal entre grupos de tomadas e T é uma janela temporal. O critério de parada do processo de agrupamento hierárquico

é definido por um parâmetro δ . O agrupamento é encerrado quando $\hat{d}_{max}(A, B) > \delta, \forall$ par de grupos (A, B) com $A \neq B$. Os grupos obtidos correspondem aos vértices do grafo de transição de cenas. Neste mesmo grafo, são criadas arestas entre vértices cujos grupos correspondentes contém ao menos um par de tomadas, uma em cada grupo, que são temporalmente adjacentes no vídeo original. A etapa final do método STG se dá pela detecção das arestas de corte no grafo de transição de cenas. Uma aresta de corte é uma aresta que conecta dois subgrafos conexos disjuntos. Estas arestas correspondem às transições entre cenas no vídeo original e cada subgrafo corresponde a uma cena, definindo a divisão do vídeo em cenas.

O raciocínio por trás da busca por arestas de corte é o de que de tomadas tem interações confinadas a grupos da mesma cena, com exceção de uma transição entre cenas. Dessa forma, é muito provável que as arestas de corte correspondam a transições entre cenas. O Algoritmo 2 contém o pseudocódigo de uma versão genérica em relação às características de representação das tomadas.

Algoritmo 2 – Grafo de Transição de Cenas

STG(S, H, d, w, δ): recebe um conjunto de tomadas $S = \{s_1, \dots, s_n\}$, um conjunto de vetores de características $H = \{h_1, \dots, h_n\}$ das S tomadas, uma distância d entre vetores de características, um parâmetro w de janela temporal e um parâmetro δ de limite de agrupamento como entrada. Devolve um conjunto $B = \{b_1, \dots, b_k\}$ onde cada $b \in B$ é um par de índices de tomadas que definem as tomadas em que começa e termina cada cena.

- 1: Efetue um agrupamento *complete linkage* sobre as tomadas em S por meio de suas representações em H usando a distância d . Considere, neste agrupamento, $d(h_i, h_j) = \infty$ caso s_i e s_j estejam separados por mais do que w tomadas no vídeo original (ver Equações 2.1 e 2.2). Pare quando todas as distâncias entre grupos forem maiores do que δ e atribua os grupos resultantes ao conjunto C ;
 - 2: Crie um grafo $G = \{V, E\}$;
 - 3: $V \leftarrow$ índices de C ;
 - 4: Insira arestas $e = (a, b)$ em E , com $a, b \in V$ para todos os pares de tomadas $s_i \in c_a$ e $s_{i+1} \in c_b$, com $\{s_i, s_{i+1}\} \in S$ e $\{c_a, c_b\} \in C$;
 - 5: Encontre e remova todas as arestas de corte em G usando busca em profundidade. Atribua os subgrafos resultantes ao conjunto R ;
 - 6: Crie uma lista vazia B ;
 - 7: **para** cada grafo $r \in R$ **faça**
 - 8: $FS(r) \leftarrow$ índice da primeira tomada representada por r ;
 - 9: $LS(r) \leftarrow$ índice da última tomada representada por r ;
 - 10: Insira $FS(r)$ e $LS(r)$ em B .
 - 11: **fim para**
 - 12: Ordene B ;
 - 13: **retorna** B .
-

O método STG tem sido amplamente reutilizado desde sua proposição (RASHEED; SHAH, 2003; SIDIROPOULOS *et al.*, 2011; XU *et al.*, 2012; COLLYDA *et al.*, 2017), e tem alcançado resultados competitivos, mesmo para os padrões atuais, por meio da

versão multimodal com fusão tardia de Sidiropoulos *et al.* (2011). As variações do STG são, em geral, tentativas de acoplar ao método, representações semanticamente mais ricas do conteúdo das tomadas, aliadas a medidas de similaridade com maior capacidade de discriminação de aspectos da semântica.

2.4.2 Classificação de vídeos

Dados um conjunto contendo vídeos com conteúdo semântico desconhecido e um conjunto de categorias, um algoritmo de classificação de vídeos deve mapear cada um dos vídeos a uma das categorias, onde cada categoria é associada a um rótulo (BREZEALE; COOK, 2008). Há uma variação desta tarefa de classificação de vídeos onde cada vídeo deve ser mapeado a uma ou mais categorias do conjunto (JIANG *et al.*, 2011; JHUO *et al.*, 2014). Os métodos de classificação de vídeos utilizam, em sua maioria, classificadores estatísticos de aprendizado supervisionado em estado da arte, treinados com quantidades suficientes de dados para capacitá-los a discriminar um dado inédito (BALLAN *et al.*, 2011). Dentre as abordagens conhecidas, pode-se citar as redes neurais, redes Bayesianas, modelos de mistura de gaussianas, modelos ocultos de Markov e *Support Vector Machines* (SVM). O método SVM será descrito por ser o mais utilizado em tarefas de classificação de vídeos.

O SVM (BOSER; GUYON; VAPNIK, 1992; CORTES; VAPNIK, 1995) foi proposto, inicialmente, como um método destinado a classificação binária, tendo sido aplicado em diversos domínios desde então (LIN; HAUPTMANN, 2002). Um SVM é um classificador de aprendizado supervisionado onde, a partir de um conjunto de dados de treinamento que representam um conjunto de elementos, onde cada elemento é representado por um vetor de valores numéricos, é construído um modelo capaz de atribuir novos vetores para uma, dentre duas classes. O algoritmo SVM mapeia cada vetor do conjunto de treinamento em um ponto num espaço n -dimensional e determina um hiperplano que separa os pontos por suas classes. O valor n corresponde à quantidade de atributos de cada vetor. Elementos cujas classes são desconhecidas podem ser mapeados neste espaço, tendo suas classes definidas pelo lado do hiperplano em que se posicionam. A Figura 17 ilustra um conjunto de dados dividido por um hiperplano.

No exemplo da Figura 17, os elementos possuem dois atributos e se dividem em duas classes, onde uma das classes é representada por círculos verdes e a outra por quadrados vermelhos. O hiperplano ilustrado por uma reta azul separa os pontos das duas classes. O algoritmo do SVM tem como objetivo, não apenas encontrar um hiperplano que separe elementos de diferentes classes, mas também encontrar um hiperplano que possua a maior distância de vetores de ambas as classes, chamada margem. A Figura 18 ilustra o conceito de margem.

Apenas alguns pontos são utilizados para definir a posição do hiperplano. Estes

Figura 17 – Conjunto de elementos de duas diferentes classes divididas por um hiperplano.

Figura 18 – Conceito de margem em SVMs.

pontos são chamados vetores de suporte e são os pontos mais próximos do hiperplano, de acordo com um parâmetro γ . À medida que se aumenta o valor de γ , apenas pontos mais próximos do hiperplano passam a ser considerados vetores de suporte e vice-versa. Além do γ , há ainda o parâmetro de regularização C , que determina o quanto se deseja evitar um erro de predição de classe nos vetores de treinamento. Valores altos de C farão com que o algoritmo SVM tenda a determinar um hiperplano com menor margem que classifique mais vetores de treinamento corretamente. Valores baixos de C farão com que o algoritmo determine um hiperplano com maior margem, mesmo que este hiperplano classifique erroneamente alguns vetores do conjunto de treinamento.

Na maioria das aplicações reais, os dados disponíveis não são linearmente separáveis, ou seja, é impossível determinar um hiperplano que divida os vetores de acordo com suas classes. Esta situação é exemplificada na Figura 19.

Figura 19 – Dados de duas classes não separáveis linearmente.

Nesta figura, os elementos possuem dois atributos e se dividem em duas classes, indicadas por círculos verdes e quadrados vermelhos. Não existe, neste espaço bidimensional, um hiperplano (neste caso uma reta) que separe corretamente as duas classes. Uma maneira de lidar com esta situação, bem como com outras semelhantes, é aplicar uma transformação nos dados, adicionando dimensões ao espaço de atributos, conforme ilustrado na Figura 20.

Figura 20 – Dados transformados passam a ser linearmente separáveis no novo espaço.

Na Figura 20, os dados originalmente bidimensionais, foram transformados em dados tridimensionais, onde o novo atributo, chamado Atributo 3, tem seus valores determinados por $(\text{Atributo 1})^2 + (\text{Atributo 2})^2$. No exemplo da Figura 20 torna-se possível separar os dados com um hiperplano, representado pelo retângulo azul (visto em ângulo) de bordas pontilhadas.

O algoritmo SVM utiliza o produto interno entre pontos como medida de simi-

laridade e, baseando-se neste fato, é utilizada a estratégia chamada *kernel trick* para fazer com seja possível determinar hiperplanos que separam os dados em espaços com dimensões adicionais, como nas situações como a exemplificada nas Figuras 19 e 20. Essa estratégia emprega **funções de kernel**, capazes de computar o produto interno de dois vetores em um espaço com dimensões adicionais à partir dos valores destes vetores em seu espaço original (TAN; STEINBACH; KUMAR, 2005). O nome *kernel trick*, ou truque do *kernel* em português, se deve ao fato de que o cálculo do produto interno em espaços com dimensões adicionais usando funções de *kernel* dispensa o conhecimento da própria função de mapeamento dos pontos originais nestes espaços.

Tanto o problema original de classificação de vídeos quanto a sua variação que admite a atribuição de cada vídeo a múltiplas classes, não podem ser resolvidos diretamente por um único SVM, já que este é um classificador binário. A versão original da classificação de vídeos corresponde à classificação multiclases e a sua variação corresponde à classificação multirrótulos, ambas tarefas clássicas da área de aprendizado de máquina. Na classificação multiclases, cada instância deve ser atribuída a uma classe de um conjunto de n classes enquanto na classificação multirrótulos cada instância deve ser atribuída a uma ou mais classes de um conjunto de n classes. A aplicação do método SVM na classificação de vídeo costuma se dar pela estratégia *One-Vs-All* que determina o treinamento de um classificador SVM para cada uma das n classes, capaz de decidir se um vídeo pertence ou não àquela classe. O conjunto das saídas destes classificadores para um determinado vídeo é um vetor que corresponde a uma resposta para a variação (multirrótulos) da tarefa original. A obtenção de uma resposta para a tarefa original (multiclases) de classificação de vídeo se dá por um processamento deste vetor, determinando qual das predições positivas possui maior probabilidade, sendo escolhida, conseqüentemente, como resposta da tarefa.

2.5 Avaliação de tarefas de análise de vídeo

A avaliação da qualidade de técnicas de análise de vídeo costuma ser realizada com o emprego de uma base de dados anotada, chamada base confiável (em inglês utiliza-se o termo *groundtruth*) e de uma ou mais medidas de eficácia (HIEMSTRA; KRAAIJ, 2007). A base confiável é um conjunto de vídeos apropriados para a tarefa que se deseja avaliar, acompanhado da resposta correta para aquela tarefa. A medida de eficácia é uma função que toma como entrada uma resposta predita e uma resposta correta para a tarefa e devolve um valor que representa o quão próxima, ou distante, a resposta predita está da resposta correta. Nas seções seguintes discutem-se o ferramental e os protocolos de avaliação das tarefas de segmentação temporal de vídeo em cenas e de classificação de vídeos.

2.5.1 Avaliação de segmentação temporal de vídeo em cenas

Uma base confiável para a segmentação temporal de vídeo em cenas é composta por uma ou mais instâncias de vídeos, onde cada vídeo é acompanhado por uma anotação das posições onde se encontram as fronteiras reais entre todas as suas tomadas e cenas. Uma das deficiências na pesquisa em segmentação temporal de vídeo em cenas é a falta de bases confiáveis disponíveis para uso (FABRO; BÖSZÖRMENYI, 2013; ROTMAN; PORAT; ASHOUR, 2017). Apesar da pesquisa em segmentação de vídeo em cenas ter suas origens na década de 90, muitos autores utilizaram bases confiáveis privadas, impedindo a replicabilidade e a comparação com os resultados publicados (FABRO; BÖSZÖRMENYI, 2013). Há trabalhos, como o de Sidiropoulos *et al.* (2011) e de Kyperountas, Kotropoulos e Pitas (2007), que utilizaram uma base que foi fornecida apenas a participantes da competição TRECVID⁹, sem disponibilizar os vídeos ou mesmo as anotações. Há, ainda, trabalhos como o de Chasanis, Likas e Galatsanos (2009) e Lopes, Trojahn e Goularte (2014), que utilizaram conteúdo proprietário pago, filmes longa metragem, e não disponibilizaram as anotações de tomada/cena empregadas.

Para o melhor do conhecimento obtido, há apenas duas bases confiáveis adequadas à segmentação de vídeo em cenas disponíveis. Estas bases confiáveis são a BBC Dataset¹⁰, apresentada por Baraldi, Grana e Cucchiara (2015a) e a Open Video Scene Detection Dataset (OVSD)¹¹, apresentada por Rotman, Porat e Ashour (2017).

BBC Dataset

A base confiável BBC Dataset foi criada sobre os episódios da primeira temporada da série de documentários BBC Planet Earth¹². A BBC Planet Earth estreou em 2006 e é composta por onze episódios, onde cada episódio tem como tema uma visão geral de um diferente habitat do Planeta Terra. Todos os episódios contam com narração e tem cerca de 48 minutos de duração. Os arquivos de vídeo fornecidos com a base confiável são em formato MP4, com o fluxo visual codificado pelo codec H.264/AVC, com resolução espacial de 360×288 pixels e uma taxa de reprodução de 25 FPS. O fluxo aural foi codificado pelo codec MPEG-4 AAC com canais estéreo, resolução de amostra variável, e taxa de amostragem de 48 kHz. Um exemplo contendo um trecho de um arquivo de anotação de tomadas da base confiável BBC, com anotações do episódio “From Pole to Pole”, pode ser visto na Figura 21.

Os arquivos com anotações de tomadas contém uma tomada por linha, onde cada

⁹ Mais informações em <<https://trecvid.nist.gov/>>

¹⁰ A base confiável BBC Dataset pode ser obtida em <<http://imagelab.ing.unimore.it/imagelab/page.asp?IdPage=5>>.

¹¹ A base confiável OVSD pode ser obtida em <https://www.research.ibm.com/haifa/projects/imt/video/Video_DataSet.shtml>

¹² Mais informações em <<https://www.bbc.co.uk/programmes/b006mywy>>

```

0      632
650    770
771    891
892    1301
1302   1568
1569   1818
1819   2110
2111   2589
2648   2682

```

Figura 21 – Trecho de arquivo de anotação de tomadas da base BBC.

tomada é representada por dois números naturais: os índices dos quadros de início e fim de cada tomada, separados por uma tabulação. Os índices são inclusivos, ou seja, a primeira tomada representada no exemplo da Figura 21 começa no quadro 0 e termina no 632. A Figura 22 contém um trecho de um arquivo de anotação de cenas da base confiável BBC, também com anotações do episódio “*From Pole to Pole*”.

```
0,1,9,15,23,27,38,44,49,52,53,63,64,72,92,99,100,107,109,121,129,
```

Figura 22 – Trecho de arquivo de anotação de cenas da base BBC.

Os arquivos com anotações de cenas contém índices de tomadas que iniciam cada cena, de maneira inclusiva e separados por vírgulas. No exemplo da Figura 22, a primeira cena inicia-se na tomada 0 e termina na tomada 0, já que a segunda cena inicia-se na tomada 1. Já a segunda cena inicia-se na tomada 1 e termina na tomada 8, inclusive, já que a terceira cena inicia-se na tomada 9.

A anotação fornecida com a base confiável lista 4916 tomadas e 672 cenas. As anotações de tomadas/cenas de referência foram realizadas por um grupo de especialistas humanos. Maiores detalhes sobre a estrutura temporal hierárquica contida nos arquivos de anotação fornecidos podem ser vistos na Tabela 1

Tabela 1 – Aspectos importantes dos vídeos da BBC *Dataset*

Episódio	Duração	Tomadas	Cenas
From Pole to Pole	00:49:15	450	66
Mountains	00:48:04	395	53
Fresh Water	00:41:17	535	63
Caves	00:48:55	393	57
Deserts	00:48:59	469	55
Ice Worlds	00:49:17	425	62
Great Plains	00:49:03	473	71
Jungles	00:49:14	461	65
Shallow Seas	00:49:14	368	62
Seasonal Forests	00:49:19	529	65
Ocean Deep	00:49:14	418	53
Total	08:51:51	4916	672

A coluna Episódio na Tabela 1 contém os títulos de cada episódio do conjunto de vídeos da base confiável. A coluna Duração contém a duração de cada vídeo, no formato horas:minutos:segundos. As colunas Tomadas e Cenas contém, respectivamente, as quantidades de tomadas e de cenas de cada vídeo. Os vídeos da base confiável *BBC Dataset*, apresentam quantidades de cenas e de tomadas consideradas aceitáveis para o propósito de avaliação de métodos de segmentação de vídeo em cenas quando comparadas com experimentos de outros autores (SIDIROPOULOS *et al.*, 2011; GAO; MA, 2012; WU; JIN, 2015; BARALDI; GRANA; CUCCHIARA, 2015a; ROTMAN; PORAT; ASHOUR, 2017). A *BBC Dataset* é uma base variada em relação ao conteúdo visual. Há, por exemplo, um vídeo sobre os pólos norte e sul do planeta (Episódio “*From Pole to Pole*”), e outro sobre selvas (Episódio “*Jungles*”). Outra característica interessante sobre a *BBC Dataset* é o fato de que todos os vídeos possuem um narrador falando durante todo o vídeo.

OVSD Dataset

A *OVSD Dataset* é uma base confiável que foi desenvolvida usando vídeos de natureza *open content* para permitir acesso irrestrito a qualquer indivíduo interessado em realizar uma avaliação de técnica de segmentação temporal de vídeo em cenas. Ela foi apresentada por Rotman, Porat e Ashour (2016) e continha, inicialmente, 6 vídeos. Em um trabalho seguinte, Rotman, Porat e Ashour (2017) integraram 15 vídeos à base confiável original. Atualmente, o conjunto de vídeos da *OVSD Dataset* é composto por 21 filmes, majoritariamente de autoria independente. Apesar de contar com vídeos de uma única classe, a de filmes, o conteúdo da *OVSD Dataset* pode ser considerado bastante variado pois, com exceção de 5 vídeos da *Blender Foundation*¹³, todos os outros vídeos são de diferentes produtoras/autores. Há vídeos filmados com equipamento amador, outros com produção e equipamento profissional, animações 2D e animações 3D. Considerando que os formatos dos arquivos de vídeo são diversos, maiores informações sobre os detalhes técnicos deles podem ser vistos no Apêndice A. Em relação à anotação, a Figura 23, contém um exemplo com um trecho do arquivo de anotação de tomadas do vídeo “*Big Buck Bunny*”.

```
0.75
6.65
13.77
19.35
35.33
51.85
58.33
```

Figura 23 – Trecho de arquivo de anotação de tomadas da base *OVSD Dataset*.

Os arquivos com anotações de tomadas contém valores em segundos, que indicam o final de cada tomada, separados por quebras de linha. No exemplo da Figura 23, a primeira

¹³ Mais informações em <<https://www.blender.org/>>

tomada termina aos 0,75 segundos, a segunda tomada termina aos 6,65 segundos, a terceira tomada termina aos 13,77 segundos, e assim por diante.

Um exemplo contendo um trecho de um arquivo de anotação de cenas do vídeo “*Big Buck Bunny*” da base confiável OVSD *Dataset* pode ser visto na Figura 24.

```

50      749
811     1812
1813    2900
2901    3512
3513    4462

```

Figura 24 – Trecho de arquivo de anotação de cenas da base OVSD *Dataset*.

Os arquivos com anotações de cenas contêm uma cena por linha, onde cada cena é representada por dois números naturais: os índices de início e fim de cada cena, separados por uma tabulação. Os índices são inclusivos, ou seja, a primeira cena representada no exemplo da Figura 24 começa no quadro 50 e termina no quadro 749. Os detalhes sobre a estrutura temporal hierárquica contida nos arquivos de anotação podem ser vistos na Tabela 2.

Tabela 2 – Aspectos importantes dos vídeos da OVSD *Dataset*

Vídeo	Duração	Tomadas	Cenas
Big Buck Bunny	00:09:56	138	15
Cosmos Laundromat	00:12:10	107	7
Elephants Dream	00:10:53	137	9
Sintel	00:14:48	151	8
Tears of Steel	00:12:14	146	11
Valkaama	01:33:05	635	51
1000 Days	00:43:40	423	22
Boy Who Never Slept	01:09:46	353	36
CH7	01:26:28	1279	44
Fires Beneath Water	01:16:06	301	62
Honey	01:26:49	386	20
Jathia's Wager	00:21:01	159	15
La Chute D'une Plume	00:10:23	81	10
Lord Meia	00:37:05	274	27
Meridian	00:11:58	58	9
Oceania	00:54:19	271	31
Pentagon	00:50:33	364	31
Route 66	01:43:25	1086	55
Seven Dead Men	00:57:03	162	34

(continua na próxima página)

Tabela 2 – Aspectos importantes dos vídeos da OVSD *Dataset*

Vídeo	Duração	Tomadas	Cenas
Sita Sings The Blues	01:21:31	965	52
Star Wreck	01:43:31	1291	55
Total	17:26:44	8767	604

A coluna Vídeo na Tabela 2 contém os títulos de cada vídeo do conjunto que compõe a base confiável. A coluna Duração contém a duração de cada vídeo, no formato horas:minutos:segundos. As colunas Tomadas e Cenas contém, respectivamente, as quantidades de tomadas e de cenas de cada vídeo. A anotação de cada um dos primeiros seis vídeos da Tabela 2, *Big Buck Bunny*, *Cosmos Laundromat*, *Elephant Dreams*, *Sintel*, *Tears of Steel*, *Valkaama*, foi extraída diretamente do *script* do respectivo filme, com um ajuste para que as fronteiras das cenas coincidissem com as das tomadas. Não havia *script* disponível para os outros quinze vídeos restantes e, por este motivo, a anotação foi realizada por um grupo de especialistas humanos. A base confiável OVSD *Dataset* é maior em duração e em quantidade de tomadas que a BBC *Dataset*, sendo semelhante em quantidade de cenas. Além disso, ela é mais variada em relação ao seu conteúdo semântico e de padrões de conteúdo aural/visual, fazendo com que seja mais desafiadora para os métodos de segmentação de vídeo em cenas.

Medidas de eficácia

Uma medida de eficácia para a segmentação temporal de vídeo em cenas, é uma função em que, dadas duas diferentes anotações de posições de fronteiras entre cenas de um mesmo vídeo, sendo uma sabidamente correta e a outra predita, computa o quão próxima, ou distante, a anotação predita está da correta. Apesar de uma quantidade razoável de medidas de avaliação ter sido proposta ao longo dos anos, ainda não há um consenso em relação a qual medida é a ideal, já que todas as medidas propostas falham em avaliar determinadas situações.

O conjunto de medidas de eficácia Precisão e Abrangência (do inglês *Precision and Recall* (PR)), eleito como o mais usado em avaliações de segmentação temporal de vídeo em cenas no *survey* de Fabro e Böszörményi (2013), provém da área de Recuperação de Informação (RIJSBERGEN, 1979). Dadas duas anotações das posições onde se encontram as fronteiras entre cenas de um vídeo, sendo uma de fronteiras reais e uma de fronteiras preditas, a aplicação do conjunto de medidas PR inicia-se por meio de contagens de fronteiras, de acordo com alguns critérios (HUA *et al.*, 2002; BESACIER *et al.*, 2004). São contabilizadas como verdadeiros positivos (do inglês *true positives* (T_p)), as fronteiras preditas corretamente, ou seja, as fronteiras da anotação de fronteiras preditas que tem um correspondente na anotação de fronteiras reais. A segunda contagem é de falsos po-

sitivos (do inglês *false positives* (F_p)), que são fronteiras preditas erroneamente, ou seja, as fronteiras da anotação de fronteiras preditas que não possuem um correspondente na anotação de fronteiras reais. Por fim, realiza-se a contagem dos falsos negativos (do inglês *false negatives* (F_n)), que são fronteiras erroneamente não preditas, ou seja, são fronteiras presentes na anotação de fronteiras reais que não possuem correspondente na anotação de fronteiras preditas.

Computados os valores de T_p , F_p e F_n , a Precisão (P), definida como

$$P = \frac{T_p}{T_p + F_p}, \quad (2.3)$$

mensura a proporção de fronteiras corretamente identificadas dentre todas as fronteiras preditas no vídeo. Já a Abrangência (R), definida como

$$R = \frac{T_p}{T_p + F_n}, \quad (2.4)$$

mensura a proporção de fronteiras corretamente identificadas dentre todas as fronteiras reais do vídeo. Uma boa anotação de posições de fronteiras produzida por um método de segmentação deve obter valores altos tanto em P quanto em R . Para simplificar a comparação entre diferentes métodos por meio do conjunto PR, as duas medidas costumam ser combinadas usando uma média harmônica F_{PR} , chamada na literatura de *F-measure* (BESACIER *et al.*, 2004; FABRO; BÖSZÖRMENYI, 2013), onde tanto P quanto R têm a mesma importância no cálculo:

$$F_{PR} = 2 \times \frac{P \times R}{P + R} \quad (2.5)$$

O conjunto de medidas PR, por não ter sido projetado originalmente para a segmentação temporal de vídeo em cenas, avalia de maneira injusta algumas situações de anotação. Ele não lida com situações onde uma cena predita pode ser parcialmente correta. Um exemplo desse tipo de situação é ilustrada na Figura 25.

Figura 25 – Avaliação de anotação de posições de fronteiras entre cenas pelo conjunto de medidas PR.

O exemplo retrata duas anotações de posições de fronteiras entre cenas de um vídeo com 10 tomadas e 3 cenas. As linhas pontilhadas representam fronteiras entre cenas.

A anotação “Real” indica as posições corretas das fronteiras entre cenas, posicionadas ao final das tomadas 3 e 8. A anotação “Predição” indica que as fronteiras entre cenas estão no final das tomadas 2 e 7. Apesar das fronteiras preditas estarem deslocadas apenas uma tomada em relação às suas posições corretas no vídeo, os valores de P , R e F_{PR} dessa anotação serão todos zero. Essa avaliação é indesejavelmente rigorosa, já que 8 dentre 10 tomadas da predição foram corretamente atribuídas às suas respectivas cenas.

Na tentativa de minimizar o efeito das situações onde o conjunto de medidas PR é indesejavelmente rigoroso, alguns autores incluem um critério de tolerância à posição de uma fronteira predita, considerando como igualmente posicionadas fronteiras reais/preditas distantes até uma determinada quantidade de tomadas uma da outra (RASHEED; SHAH, 2003) ou então um determinado intervalo de tempo uma da outra (HANJALIC; LAGENDIJK; BIEMOND, 1999). A inclusão de um critério de tolerância, apesar de tornar o conjunto de medidas PR menos rigoroso, dificulta a comparação entre diferentes trabalhos pois os autores utilizam critérios diferentes.

Além do rigor excessivo da medida original ao avaliar algumas situações, Sidiropoulos *et al.* (2012) demonstraram que o F_{PR} não é simétrico em erros de identificação de fronteiras, ou seja, uma fronteira erroneamente predita a n tomadas além da correta, pode ser avaliada de maneira diferente do que se tivesse sido predita a n tomadas antes da correta. Este comportamento é indesejável para a avaliação de segmentação temporal de vídeo em cenas.

Motivados pela imprecisão decorrente do emprego de medidas de avaliação adaptadas para a segmentação temporal de vídeo em cenas, Vendrig e Worring (2002) propuseram o conjunto de medidas de eficácia *Coverage* e *Overflow* (CO) (Cobertura e Excedência em português), amplamente empregado em avaliações de segmentação temporal de vídeo em cenas.

Dados $S = \{s_1, \dots, s_{|S|}\}$ como o conjunto de cenas preditas, $\tilde{S} = \{\tilde{s}_1, \dots, \tilde{s}_{|\tilde{S}|}\}$ como o conjunto de cenas reais e $|s_i|$ como o número de tomadas em uma cena s_i , a *coverage* C_t de uma cena $\tilde{s}_t$ é definida como

$$C_t = \frac{\max_{i=1, \dots, |S|} |s_i \cap \tilde{s}_t|}{|\tilde{s}_t|} \quad (2.6)$$

e mensura a quantidade de tomadas de $\tilde{s}_t$ agrupadas corretamente na predição, o que corresponde à mais longa sobreposição entre s_i e $\tilde{s}_t$. A *overflow* O_t de uma cena $\tilde{s}_t$ é definida como

$$O_t = \frac{\sum_{i=1}^{|S|} |s_i \setminus \tilde{s}_t| \times \min(1, |s_i \cap \tilde{s}_t|)}{|\tilde{s}_{t-1}| + |\tilde{s}_{t+1}|} \quad (2.7)$$

e mensura quantas tomadas foram preditas além das fronteiras de $\tilde{s}_t$ na cena correspon-

dente predita em S . Ao contrário das outras medidas em que, quanto maior o valor, melhor a predição, na *overflow*, quanto menor o valor, melhor é a predição.

A *coverage* e a *overflow* para as predições de um vídeo inteiro são definidas como

$$C = \sum_{t=1}^{|\tilde{S}|} C_t \times \frac{|\tilde{S}_t|}{\eta} \quad \text{e} \quad O = \sum_{t=1}^{|\tilde{S}|} O'_t \times \frac{|\tilde{S}_t|}{\eta}, \quad (2.8)$$

onde η é a quantidade total de tomadas do vídeo. Assim como na Precisão/A-brangência, uma média harmônica também é empregada para sumarizar os resultados de *coverage* e *overflow* de todo o vídeo. Considerando que uma boa anotação de posição das fronteiras entre cenas deve obter valores altos para *coverage* e baixos para *overflow*, a *F-measure* para CO é dada pela média harmônica entre C e $1 - O$, com pesos iguais para ambas as medidas.

O conjunto de medidas CO, apesar de ter sido projetado especificamente para avaliar predições de segmentação de vídeo em cenas, apresenta algumas deficiências. Conforme apontado por Sidiropoulos *et al.* (2012) e por Baraldi, Grana e Cucchiara (2015a), o F_{CO} , assim como o F_{PR} , não é simétrico em erros de identificação de fronteiras, podendo produzir resultados bastante diferentes para erros com o mesmo comprimento. O fato do valor da *coverage* ser computado exclusivamente sobre a maior cena para a qual há sobreposição também pode levar a avaliações imprecisas, já que ele não penaliza qualquer outra cena sendo coberta simultaneamente. Uma grave deficiência da *overflow* é a sua incapacidade de avaliar adequadamente os casos extremos de subsegmentação, onde a quantidade de cenas preditas é muito menor do que a quantidade de cenas reais, fazendo com que hajam cenas muito grandes na predição. Nestes casos, devido ao fato da *overflow* possuir apenas as tomadas imediatamente vizinhas em seu denominador, os valores de *overflow* computados podem facilmente ultrapassar o valor máximo que deveria ser 1, fazendo com que o F_{CO} assumam valores negativos ou mesmo acima de 1. Um exemplo desse tipo de situação é ilustrada na Figura 26.

Figura 26 – Avaliação de anotação de posições de fronteiras entre cenas pelo conjunto de medidas CO.

O exemplo retrata duas anotações de posições de fronteiras entre cenas de um vídeo com 10 tomadas e 3 cenas. As linhas pontilhadas representam fronteiras entre cenas. A anotação “Real” corresponde às posições corretas das fronteiras entre cenas, posicionadas ao final das tomadas 3 e 8. A anotação “Predição” indica que há apenas uma cena em

todo o vídeo e, dessa forma, não há fronteiras entre cenas. O valor de C neste exemplo é 1, indicando que a predição não deixou de cobrir qualquer trecho de qualquer cena, o que está correto. Por outro lado, o valor de O será de 1,24, ultrapassando o valor máximo definido para a medida que é 1. Com isso, o valor de F_{CO} será também inválido: $-0,63$.

A fim de corrigir essa deficiência da *overflow* original, Han e Wu (2011) propuseram uma modificação dessa medida, chamada nesta tese de *New Overflow* (NO) e tal que, para uma cena real $\tilde{s}_t$, é dada por:

$$NO_t = 1 - \frac{|\tilde{s}_t|}{\sum_{i=1}^{|S|} |s_i| \times \min(1, |s_i \cap \tilde{s}_t|)}, \quad (2.9)$$

onde todos os comprimentos de tomadas reais cobertas por uma predição excedente errônea participam do cálculo, tornando-a uma medida mais precisa. A *new overflow* também pode ser calculada para todo o vídeo, de maneira análoga ao cálculo da *overflow* para todo o vídeo, conforme a Equação 2.8. Além disso, também é possível combinar *coverage* e *new overflow* por meio de uma média harmônica F_{CNO} , calculando-a entre C e $1 - NO$. Retomando o exemplo da Figura 26, a avaliação por emprego de *coverage* e *new overflow* produziria os valores $C = 1$, $O = 0,62$ e $F_{CNO} = 0,765$, valores válidos e muito mais precisos em relação à quão correta é a predição.

Ha ainda, algumas outras medidas dignas de menção como, por exemplo, a distância de edição diferencial (do inglês *differential edit distance*), apresentada por Sidiropoulos *et al.* (2012), que propõe uma modelagem das anotações de segmentação de vídeo em cenas como sequências de caracteres binários. Com isso, é possível calcular a distância entre as anotações real e predita usando a distância de edição diferencial. A distância de edição diferencial é uma extensão da bem conhecida distância de edição (HAMMING, 1950), que pode ser resumida como uma medida de quão diferentes são duas sequências de caracteres, com valor dado pela contagem de operações sobre um caractere necessárias para transformar uma das sequências na outra. A distância de edição diferencial se distingue da distancia de edição clássica pelo critério adicional de particionamento idêntico. Este critério promove uma flexibilização da correspondência entre caracteres, permitindo que a medida seja orientada à equivalência de particionamento das sequências ao invés de uma estrita igualdade de rótulos.

Outra medida de avaliação de segmentação temporal de vídeo em cenas é a intersecção sobre união, também conhecida como Índice de Jaccard. A intersecção sobre união foi originalmente proposta por Jaccard (1912) como uma proporção de espécies de plantas compartilhadas entre duas amostras, em relação ao total de espécies de plantas. Esta medida é bastante empregada em avaliações de segmentação espacial de imagens (LATEEF; RUICHEK, 2019), tendo sido estendida à avaliação de segmentação temporal de vídeo em cenas por Baraldi, Grana e Cucchiara (2015a). Na segmentação temporal

de vídeo em cenas, é calculada a intersecção sobre união máxima de quadros entre cada cena da anotação das cenas reais, e as cenas detectadas, calculando uma média sobre todo o vídeo. O mesmo é feito para as cenas detectadas em relação às cenas reais. A média aritmética destas duas quantidades corresponde ao valor da intersecção sobre união entre as duas anotações de decomposição de vídeo em cenas.

Apesar de serem tentativas de resolver deficiências de outras medidas de avaliação disponíveis, a distância de edição diferencial e a intersecção sobre união, para o melhor do conhecimento obtido, não foram adotadas em trabalhos posteriores aos de suas respectivas apresentações no domínio de segmentação temporal de vídeo em cenas.

2.5.2 Avaliação de classificação de vídeos

Uma base confiável para a classificação de vídeos é composta por um conjunto de vídeos, acompanhados de anotações indicando a quais classes cada um dos vídeos pertence. A área de classificação de vídeos conta com uma variedade de bases confiáveis destinadas a experimentação e avaliação de novos métodos e técnicas de classificação de vídeo. Exemplos destas bases são a Hollywood2 (MARSZALEK; LAPTEV; SCHMID, 2009), TRECVID MED (EVALUATION, 2010), HMDB-51 (KUEHNE *et al.*, 2011), UCF-101 (SOOMRO; ZAMIR; SHAH, 2012), CCV Database (JIANG *et al.*, 2011), ActivityNet (HEILBRON *et al.*, 2015) e Youtube-8M (ABU-EL-HAJJA *et al.*, 2016). Dentre elas, a base confiável *Columbia Consumer Video Database* (CCV Database) merece atenção especial, pois foi empregada na avaliação da técnica de classificação de Jhuo *et al.* (2014), baseada em um método de fusão de características semelhante ao método M4InFus. A CCV Database é uma das poucas bases confiáveis que disponibiliza vetores de características de baixo nível semântico previamente extraídos, permitindo que dois diferentes métodos de classificação de vídeos sejam avaliados de maneira justa, sem qualquer possibilidade de influência de diferentes implementações da etapa de extração de características. Além disso, a base confiável CCV Database é considerada complexa e desafiadora mesmo em trabalhos recentes como o de Zhang *et al.* (2018).

CCV Database

A base confiável CCV Database é composta por vídeos produzidos por cinegrafistas amadores que foram obtidos do sistema de compartilhamento Youtube¹⁴. O interesse em vídeos de origem amadora se justifica pelo crescimento na produção e compartilhamento deste tipo de conteúdo, aliado ao fato de que estes vídeos possuem conteúdo diverso, complexo e com baixa disponibilidade de descrição textual de conteúdo. De acordo com Jiang *et al.* (2011), os vídeos da base CCV Database possuem, em média, 2,5 rótulos enquanto um vídeo qualquer do Youtube possui, em média, 9 rótulos. A baixa disponibilidade de

¹⁴ Disponível em <<http://www.youtube.com>>.

descrição textual do conteúdo dificulta seu tratamento por mecanismos de recuperação de informação, reforçando sua escolha como componente de avaliação.

O conjunto de vídeos da *CCV Database* é composto por 9317 vídeos, com duração total de 210 horas (aproximadamente 80 segundos por vídeo). Cada vídeo deste conjunto pertence a zero ou mais classes, de um conjunto de 20 diferentes classes, que incluem objetos, cenas, eventos esportivos e atividades descritas em alto nível. A Tabela 3 apresenta a distribuição dos vídeos nas diferentes classes.

Tabela 3 – Distribuição dos vídeos nas diferentes classes da *CCV Database*.

Classe	Quantidade de Vídeos
Basquete	364
Baseball	304
Futebol	328
Patinando no gelo	386
Esquiando	394
Nadando	404
Andando de bicicleta	274
Gato	388
Cão	654
Pássaro	340
Graduação	290
Aniversário	319
Festa de casamento	259
Cerimônia de casamento	224
Dança de casamento	351
Performance musical	806
Performance não musical	692
Desfile	389
Praia	377
Área de recreação	336

Os autores da *CCV Database* fornecem um conjunto de arquivos contendo as seguintes informações: lista de identificadores de todos os vídeos na plataforma Youtube, características de médio nível semântico BoF de cada vídeo, construídas sobre características de baixo nível semântico SIFT, MFCC e STIP e uma matriz de anotação indicando a quais classes cada vídeo pertence. A Figura 27 contém um exemplo com um trecho do arquivo de anotação das classes dos vídeos do conjunto de treino da base *CCV Database*:

Os arquivos com anotações de classes dos vídeos de treino e de teste possuem o mesmo formato. Cada linha indica a quais classes o vídeo pertence. Cada coluna é relacionada a uma classe, seguindo a mesma ordem da lista de classes apresentadas na Tabela 3. O valor um em uma coluna indica que o vídeo daquela linha pertence à classe correspondente àquela coluna, e o valor zero indica o contrário. O valor contido na terceira coluna, por exemplo, indica se o vídeo pertence (valor 1) ou não (valor 0) à classe Futebol.

```

1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

```

Figura 27 – Trecho de arquivo de anotação de classes de alguns vídeos da base CCV *Database*.

No exemplo da Figura 27, todos os vídeos pertencem à primeira classe, que é Basquete. O vídeo correspondente à quinta linha, pertence, também à classe Performance não musical.

Medidas de eficácia

Uma medida de eficácia para a classificação de vídeos é uma função tal que, dados um ou mais vídeos, um conjunto de classes e uma predição indicando a quais classes do conjunto cada vídeo pertence, calcula um valor que indica quão correta está a predição em relação às reais classes de cada vídeo. A medição da eficácia da classificação de um conjunto de vídeos pode ser realizada empregando a maioria das medidas clássicas de classificação de dados em geral como, por exemplo, o *F-measure* ou Índice de Jaccard (HOS-SIN; SULAIMAN, 2015). A *Mean Average Precision* (MAP) (BUCKLEY; VOORHEES, 2000) merece atenção especial, já que a maioria dos trabalhos de classificação de vídeo encontrados nos últimos anos a empregam em suas respectivas avaliações.

A MAP é uma média aritmética simples calculada de um conjunto de valores de *Average Precision* (AP) (SALTON; MCGILL, 1986). A AP é uma medida calculada, individualmente, para cada classe presente em uma predição de classificação. Ela avalia a qualidade de uma predição binária para uma determinada classe (se cada vídeo pertence ou não àquela classe) e corresponde a uma aproximação da área sob a curva de Precisão/Abrangência daquela predição. Desse modo, a AP corresponde a integral da função com os valores de Precisão P , dados os valores de Abrangência r :

$$AP = \int_0^1 P(r) dr, \quad (2.10)$$

Na tarefa de classificação de vídeos, de maneira análoga ao que acontece na segmentação temporal de vídeos, a Precisão mensura a proporção de vídeos corretamente preditos positivos dentre o total de vídeos preditos positivos. Já a Abrangência, mensura a proporção de vídeos preditos positivos do total de vídeos verdadeiramente positivos. Ambas as medidas tem valores que variam entre 0 (predições erradas) e 1 (predições corretas). Na tarefa de classificação de vídeos, os valores de Precisão costumam ser inversamente proporcionais aos de Abrangência.

Considerando que uma predição de classes de vídeos é composta por probabilidades, os valores de Precisão e Abrangência dependem de um limiar que define qual a probabilidade mínima para que um vídeo seja considerado predito positivamente àquela classe. O cálculo da AP elimina a necessidade de determinar um valor para este limiar, resumizando o comportamento da predição sob diferentes limiares em apenas um valor. Uma aproximação ao valor de AP costuma ser obtida pelo cálculo da equação (EVERINGHAM *et al.*, 2010):

$$AP = \frac{1}{11} \sum_{r \in \{0, 0.1, \dots, 1\}} P(r), \quad (2.11)$$

onde a Precisão P é calculada em onze diferentes níveis de Abrangência r , igualmente espaçados, entre 0 e 1, inclusive. O valor de P em um determinado nível de Abrangência r é definido como a maior Precisão medida naquela predição tal que a Abrangência correspondente exceda o valor de r .

2.6 Considerações finais

Este capítulo apresentou os conceitos considerados pelo autor como necessários para a compreensão desta tese. Iniciou-se esta apresentação na Seção 2.1 pelas definições de vídeo e de vídeo digital, já que o tema do trabalho é a fusão de informação multimodal de vídeo. Após a definição de vídeo digital, foram descritos os dois principais canais de informação que compõem um vídeo: os canais visual e aural. Essa descrição é essencial para que, posteriormente, se pudesse discutir maneiras de se representar vídeos ou mesmo subunidades de vídeo. Foi apresentada, em seguida, a estrutura temporal hierárquica de vídeo, necessária para a compreensão da tarefa de segmentação temporal de vídeo em cenas, uma das duas tarefas em que o método M4InFus foi avaliado.

Terminada a discussão dos aspectos de vídeo interessantes ao desenvolvimento do texto desta tese, foram apresentadas e discutidas, na Seção 2.2, as propriedades discriminatórias de vídeo chamadas características. É por meio de características de vídeo que se realiza a comparação entre o conteúdo de diferentes vídeos/trechos de vídeos. Essas características são classificadas em baixo, médio e alto nível semântico, de acordo com a sua complexidade estrutural em relação à semântica. O método M4InFus foi projetado para processar características de baixo e de médio nível semântico e, dessa forma, foram discutidos métodos de extração dessas duas classes de características.

O método M4InFus é destinado à fusão de informação multimodal. Por este motivo, o conceito de multimodalidade foi definido na Seção 2.3. Além desse conceito, foram apresentadas as principais estratégias de fusão de informação: fusão tardia e fusão prévia. O M4InFus é um método de fusão de informação multimodal e, assim, é caracterizado como

um método fusão prévia. Ainda nessa seção foram discutidas qualidades e deficiências de ambas as estratégias apresentadas.

A Seção 2.4 aborda duas tarefas de análise de vídeo, cujas soluções são necessárias em muitos dos sistemas de recuperação de vídeo. Essas tarefas foram escolhidas para compor a metodologia de avaliação experimental do método de fusão de informação multimodal proposto. A intenção dos experimentos foi de aplicar o método de fusão de informação proposto em tarefas clássicas de análise de vídeo, colher e avaliar os resultados de modo que fosse possível detectar possíveis ganhos em eficácia propiciados por ele. As tarefas escolhidas são a segmentação temporal de vídeo em cenas e a classificação de vídeos, para as quais foram discutidos métodos existentes.

Para completar a metodologia de avaliação do método M4InFus, a Seção 2.5 trata da avaliação das duas tarefas de análise de vídeo escolhidas para este propósito, já que a avaliação do método foi realizada de modo indireto. A avaliação de ambas as tarefas, bem como na maioria das tarefas de análise de vídeo, depende da disponibilidade de uma base confiável e de uma medida de eficácia por tarefa. Para a tarefa de segmentação temporal de vídeo em cenas, foram descritas as bases confiáveis *BBC Dataset* e *OVSD Dataset*, por serem adotadas por trabalhos recentes e em estado da arte, além de serem as únicas disponíveis publicamente no momento da escrita desta tese. Além das bases confiáveis, foram discutidos os conjuntos de medidas de eficácia Precisão/Abrangência e *Coverage/Overflow*, presentes na maioria dos trabalhos recentes e também em estado da arte. Já para a tarefa de classificação de vídeos, onde há farta disponibilidade de bases confiáveis, foi escolhida e descrita a base confiável *CCV Database*, por ter sido empregada em um trabalho sobre classificação de vídeos com etapa de fusão de informação semelhante ao método M4InFus. Por fim, foi descrita a medida de eficácia *Mean Average Precision*, empregada na maioria dos trabalhos recentes em classificação de vídeos.

TRABALHOS RELACIONADOS

Este capítulo aborda os principais trabalhos relacionados ao tema central desta tese, a fusão de informação multimodal. Não foram encontrados relatos, na literatura, sobre métodos de fusão de informação multimodal de propósito geral. Dada a ausência de trabalhos dedicados à fusão de informação, este capítulo trata de técnicas destinadas a tarefas de análise de vídeo que empregam multimodalidade com fusão de informação relacionadas ao método proposto.

A área de análise de vídeo conta com diversas tarefas como a segmentação temporal em tomadas e em cenas, a classificação, a sumarização, a detecção de objetos, dentre outras (BREZEALE; COOK, 2008; HU *et al.*, 2011; FABRO; BőSZőRMENYI, 2013). No rol de tarefas conhecidas, observou-se que tanto a segmentação temporal de vídeo em cenas quanto a classificação de vídeos são problemas considerados em aberto pela literatura, demandando o emprego de multimodalidade para a obtenção de bons resultados (FABRO; BőSZőRMENYI, 2013; JHUO *et al.*, 2014). Ambas as tarefas apresentam potencial de ganho em eficácia pela computação de melhores representações da entrada, tornando-as boas escolhas para aplicação do método M4InFus. Além disso, há uma ampla disponibilidade de trabalhos que exploram multimodalidade com fusão de informação para ambas as tarefas, o que não necessariamente acontece em outras tarefas como, por exemplo, a sumarização de vídeo. A alta disponibilidade de trabalhos que lidam com fusão de informação multimodal facilita comparações com características dos métodos existentes com as do método M4InFus e viabiliza discussões comparativas de resultados. As Seções 3.1 e 3.2 apresentam, respectivamente, discussões sobre o estado da arte em segmentação temporal de vídeo em cenas e classificação de vídeos.

3.1 Trabalhos em segmentação temporal de vídeo em cenas

A segmentação temporal de vídeo em cenas de modo automático é considerada um problema em aberto pela literatura ([FABRO; BÖSZÖRMENYI, 2013](#); [BARALDI; GRANA; CUCCHIARA, 2017](#); [ROTMAN; PORAT; ASHOUR, 2017](#)). A dificuldade no desenvolvimento de soluções para tal problema é decorrente da definição de cena, dada em função da relação semântica entre tomadas, submetendo qualquer método automático de identificação de fronteiras entre cenas à Lacuna Semântica. Há uma grande dificuldade em produzir representações das tomadas que sejam suficientemente distintivas em relação à semântica, tal que possibilitem seu correto agrupamento em cenas.

O início da pesquisa em segmentação temporal de vídeo em cenas data da segunda metade da década de 90, com os trabalhos seminais de [Yeung, Yeo e Liu \(1998\)](#) e [Kender e Yeo \(1998\)](#). [Yeung, Yeo e Liu \(1998\)](#) introduziram o Grafo de Transição de Cenas. A abordagem proposta pelos autores é unimodal e baseada em conteúdo visual. Nesta técnica cada tomada é representada por um subconjunto de seus respectivos quadros (quadros chave ¹) e o agrupamento de tomadas em cenas se dá pelo algoritmo STG descrito na Seção 2.4.1. O trabalho de [Kender e Yeo \(1998\)](#) apresentou uma técnica também unimodal e baseada em conteúdo visual. Nela, possíveis fronteiras entre cenas são calculadas por uma medida de coerência com memória de curto prazo tomada a tomada. Após o cálculo, os pontos de mínimo local são identificados como fronteiras entre cenas.

No final da década de 90, observou-se a possibilidade de melhorar os resultados na tarefa de segmentação temporal de vídeo em cenas pelo emprego de multimodalidade. Surgiram trabalhos como o de [Sundaram e Chang \(2000\)](#) e [Jiang, Lin e Zhang \(2000\)](#), que empregavam a modalidade aural em combinação com a modalidade visual por meio de fusão tardia. O trabalho de [Sundaram e Chang \(2000\)](#) apresentou uma técnica baseada em coerência de cor com memória de curto prazo para realizar a segmentação usando a modalidade visual. O vídeo é segmentado paralelamente usando a modalidade aural, por meio da identificação de mínimos locais dos valores de uma função de correlação, calculada no intervalo de uma janela deslizante que percorre todo o vídeo. O resultado dos processos de segmentação unimodais são combinados usando um algoritmo de vizinho mais próximo para identificar fronteiras aurais e visuais que estão temporalmente próximas uma da outra, ocorrendo juntas dentro de uma janela temporal. As fronteiras visuais que possuem fronteiras aurais correspondentes dentro da janela temporal são entregues como resposta da técnica.

O trabalho de [Jiang, Lin e Zhang \(2000\)](#) apresentou uma técnica que determina,

¹ Um quadro chave é um dos quadros do segmento de vídeo que melhor representa o conteúdo visual daquele segmento de vídeo.

inicialmente, posições de prováveis fronteiras entre cenas pela modalidade aural. Isso é feito detectando os instantes de tempo nos quais ocorre uma mudança de classe (música, som ambiente, fala, etc) ou de locutor no áudio. Dentre esses instantes de tempo, todos os que coincidem temporalmente com uma fronteira entre tomadas são marcados como possíveis fronteiras entre cenas. Numa segunda etapa de processamento, é realizado um agrupamento de tomadas usando um método chamado janela crescente (do inglês “*expanding window*”), que determina uma possível cena com três tomadas e tenta incluir a tomada adjacente nessa possível cena usando uma medida de correlação de cor como critério. Caso a tomada adjacente apresente um grau aceitável de correlação de cor, ela é incorporada àquela possível cena. Caso contrário, uma nova possível cena de três tomadas é criada naquele ponto e o processo se repete. Ao final, as possíveis fronteiras entre cenas determinadas pela modalidade aural que coincidirem com possíveis fronteiras entre cenas determinadas pela modalidade visual são definidas como as fronteiras finais entre cenas preditas pela técnica.

Nos anos seguintes surgiram diversas abordagens para a segmentação temporal de vídeo em cenas, cujos avanços se davam principalmente por: uso de multimodalidade com fusão prévia ² (CHAISORN; CHUA; LEE, 2003; FENG *et al.*, 2005), emprego de características mais representativas em relação à semântica (CHASANIS; KALOGERATOS; LIKAS, 2009; MITROVIĆ *et al.*, 2010), apresentação de novos algoritmos de detecção de fronteiras entre cenas (LIN; ZHANG; SHI, 2001; WANG *et al.*, 2007; WENG; CHU; WU, 2009) e aperfeiçoamento/extensão de algoritmos de detecção de fronteiras entre cenas existentes (WENGANG; DE, 2003; NGO; MA; ZHANG, 2005; PARSHIN; PARADZINETS; CHEN, 2006).

Na última década consolidou-se a necessidade do emprego de multimodalidade para técnicas de segmentação de vídeo em cenas independentes de domínio e, consequentemente, de desenvolvimento de estratégias de fusão de informação. Dentre os trabalhos multimodais com fusão de informação disponíveis, os de Sidiropoulos *et al.* (2011), Gao e Ma (2012), Baraldi, Grana e Cucchiara (2015a), Rotman, Porat e Ashour (2017) e Rotman *et al.* (2018) serão cuidadosamente discutidos. Os trabalhos de Sidiropoulos *et al.* (2011) e Rotman, Porat e Ashour (2017) são estado da arte em relação ao emprego de multimodalidade com fusão tardia e servem para comparação entre abordagens baseadas em fusão prévia e fusão tardia. Já o trabalho de Gao e Ma (2012), é estado da arte em fusão prévia sem emprego de aprendizado profundo, abordagem pertencente a mesma classe do método de fusão proposto, servindo para que se compare a eficácia entre abordagens semelhantes. Já os trabalhos de Baraldi, Grana e Cucchiara (2015a) e Rotman *et al.* (2018), são estado da arte em multimodalidade com fusão de informação e emprego de aprendizado profundo, apresentando os mais altos resultados em eficácia disponíveis e

² A fusão prévia nessas técnicas era realizada de modo ingênuo, por uma simples concatenação dos vetores de características.

reprodutíveis/comparáveis. Todos esses trabalhos, com exceção do trabalho de [Gao e Ma \(2012\)](#) cujos resultados não são reprodutíveis/comparáveis, foram empregados nos experimentos descritos no Capítulo 5, servindo de referência para comparação com técnicas de segmentação temporal de vídeo em cenas baseadas no método de fusão de informação proposto.

[Sidiropoulos et al. \(2011\)](#) descrevem uma abordagem bimodal para a segmentação temporal de vídeo em cenas, que emprega uma combinação de características aurais e visuais, tanto de baixo, quanto de alto nível semântico. Eles propuseram uma aproximação de menor custo computacional ao algoritmo STG (abordado na Seção 2.4.1), descrevendo também como estendê-lo para realizar a segmentação temporal de vídeo em cenas empregando características não-visuais. A Figura 28 ilustra a abordagem proposta pelos autores, onde a cor azul indica a modalidade visual e a cor rosa indica a modalidade aural.

Figura 28 – Visão geral da técnica fusão tardia de [Sidiropoulos et al. \(2011\)](#).

Para a extração de informação visual são, inicialmente, selecionados quadros chave de cada tomada. Os quadros chave escolhidos são processados por um extrator de histogramas de cor HSV e também por um classificador SVM que computa vetores de características baseados em conceitos de alto nível semântico. O fluxo aural é dividido em segmentos de áudio, análogos às tomadas, por classificação de som de fundo e *speaker diarization*³. Os segmentos produzidos são rotulados como ruído, silêncio, música, e código de identificação do locutor, caso qualquer fala seja detectada. Estes segmentos de áudio são submetidos a detectores de eventos de alto nível semântico baseados em SVMs, *multilayer perceptrons* e modelos de mistura gaussiana. Para cada uma das quatro combinações entre baixo/alto nível semântico e origem aural/visual de características, são gerados múltiplos STGs com parâmetros escolhidos de maneira randômica. Faz-se isso para reduzir a dependência da abordagem aos parâmetros do algoritmo STG. Para cada fronteira i entre tomadas existente, a quantidade de STGs de um tipo y identificando i adicionalmente como uma fronteira entre cenas é dividido pela quantidade total de STGs

³ *Speaker diarization* é o ato de dividir o áudio em segmentos temporais de acordo com o locutor em cada segmento.

do tipo y . O valor p_i^y resultante de tal divisão é interpretado como uma medida de confiança da classificação adicional de i como sendo também uma fronteira entre cenas. Os valores de confiança de cada tipo de característica são combinados linearmente em um único valor de confiança para cada fronteira, efetuando a fusão tardia, de acordo com a Equação 3.1:

$$p_i = \sum_y w_y \times p_i^y \quad (3.1)$$

onde w_y são parâmetros de ponderação para ajustar a influência de cada tipo de característica. As fronteiras entre tomadas cujo valor de confiança associado excede um determinado limiar compõem o conjunto de fronteiras entre cenas da anotação final produzida pela técnica.

A abordagem de Sidiropoulos *et al.* (2011) é puramente baseada em fusão tardia e, assim, não aproveita qualquer possível correlação intermodalidades antes das etapas de segmentação. Essa técnica também despense esforço computacional repetido nos múltiplos processos de segmentação unimodais, característica indesejável do ponto de vista prático. Os autores executaram sua técnica, acompanhada de outras três, em uma base confiável composta por quinze vídeos de documentários e em outra composta por seis filmes, ambas indisponíveis atualmente. Anotações de posições confiáveis de fronteiras entre cenas foram produzidas por espectadores humanos para ambas as bases. A eficácia foi medida usando o conjunto de medidas *Coverage/Overflow* e os resultados reportados são apresentados na Tabela 4.

Tabela 4 – Resultados médios de F_{CO} reportados em Sidiropoulos *et al.* (2011).

F_{CO}		
Técnica	Base confiável	
	Documentários	Filmes
Sidiropoulos <i>et al.</i> (2011)	0,88	0,85
Nitanda, Haseyama e Kitajima (2005)	0,80	0,75
Chasanis, Likas e Galatsanos (2009)	0,73	0,80
Wilson e Divakaran (2009)	0,81	0,79

A técnica de Sidiropoulos *et al.* apresentou eficácia consideravelmente superior às das outras técnicas empregadas na comparação. Apesar de ambas as bases empregadas nesses experimentos não serem publicamente disponíveis durante o desenvolvimento desta tese, é possível efetuar comparações com a técnica de Sidiropoulos *et al.* por meio dos resultados reportados nos trabalhos de Baraldi, Grana e Cucchiara (2015a) e Rotman, Porat e Ashour (2017). Os autores desses trabalhos avaliaram a técnica de Sidiropoulos *et al.* nas bases confiáveis *BBC Dataset* e *OVSD Dataset*, disponíveis para acesso público.

Gao e Ma (2012) propuseram uma técnica bimodal baseada em fusão prévia para a tarefa de segmentação temporal de vídeo em cenas que emprega apenas características de baixo nível semântico. A Figura 29 ilustra a abordagem proposta pelos autores, onde a cor azul indica a modalidade visual e a cor rosa indica a modalidade aural.

Figura 29 – Visão geral da técnica fusão prévia de Gao e Ma (2012).

O conteúdo visual de cada tomada é resumido em um conjunto de quadros chave extraído por uma variação da técnica de Rasheed e Shah (2003). São extraídos, dos quadros chave de cada tomada, histogramas HSV. Como os histogramas HSV não representam informação espacial, são então complementados por vetores de características de Gradiente de Entropia e de Conteúdo de Movimento. O conteúdo aural das tomadas é resumido em segmentos de áudio que circundam temporalmente cada quadro chave do conteúdo visual. São extraídos, de cada um destes segmentos, vetores de características MFCC, Energia de Termo Curto, Taxa de Cruzamento por Zero, Tom, Fluxo Espectral e Grau Harmônico. Não há, por parte dos autores, explicações sobre a escolha destes extratores de características aurais. Os vetores de características visuais de todos os tipos de extratores de características visuais de um quadro são concatenados em um único vetor de características visuais para representar aquele quadro. O mesmo é feito, de maneira análoga, com os segmentos de áudio e seus respectivos vetores de características aurais.

A fusão de características é realizada aplicando a Análise de Correlação Canônica com *Kernel* (ACCK) (FYFE; LAI, 2000; BACH; JORDAN, 2003), uma extensão da Análise de Correlação Canônica (HOTELLING, 1936), que por sua vez é uma técnica para encontrar vetores base de dois conjuntos de variáveis que maximizem a correlação entre as projeções das variáveis nestes vetores base (HARDOON; SZEDMAK; SHAW-TAYLOR, 2004). A ACCK realiza a mesma tarefa da Análise de Correlação Canônica, projetando antes os dados em um espaço de características expandido. A ACCK é empregada em situações onde a Análise de Correlação Canônica não é capaz de identificar correlação por operar em um espaço linear. Na técnica de Gao e Ma (2012), a ACCK é realizada com o emprego de um *kernel* gaussiano, tomando como entrada os vetores de características visuais/aurais concatenados. Sejam x e y os vetores de características visuais e aurais de uma tomada, respectivamente, o vetor de características bimodal Z que representará a

mesma tomada é obtido pela Equação 3.2:

$$Z = \begin{pmatrix} w_x \\ w_y \end{pmatrix}^T \begin{pmatrix} x \\ y \end{pmatrix} \quad (3.2)$$

onde w_x e w_y são vetores base canônicos que maximizam um valor de correlação ρ .

Os vetores de características audiovisuais resultantes da fusão são utilizados na construção de um grafo. Neste grafo, cada nó corresponde a uma tomada e as arestas entre dois nós possuem pesos dados pela distância euclidiana entre os vetores de características fundidas que representam as tomadas correspondentes àqueles nós. Um algoritmo de corte em grafos é aplicado para encontrar uma partição deste grafo que maximize a similaridade entre os nós de cada subgrafo e minimize a similaridade entre subgrafos. Cada subgrafo possuirá um conjunto de tomadas associadas que compõem uma cena.

Efetuar a ACCK, base do método de fusão de características, tem complexidade de tempo cúbica para ser resolvida (LOPEZ-PAZ *et al.*, 2014), considerada alta para cenários de *Big Data* (UURTIO; BHADRA; ROUSU, 2019), como é o caso da análise de vídeo. Os autores não especificam o tipo de correlação que seu método de fusão de características tem como objetivo capturar, dificultando o pós processamento. Outra limitação do método é o fato de que ele só identifica correlação entre vetores de características que representam a mesma tomada, ignorando qualquer correlação em informação proveniente de tomadas distintas. Além disso, o algoritmo de fusão baseado em ACCK descrito é limitado a realizar a fusão de apenas dois conjuntos de variáveis, ou seja, duas modalidades. Por fim, como a representação produzida pela ACCK é um vetor que contém apenas valores de correlação, padrões não correlacionados que forem importantes para a segmentação serão ignorados (SARGIN *et al.*, 2007).

A técnica proposta por Gao e Ma foi avaliada em uma base confiável composta por quatro filmes e, infelizmente, não há menção no texto de como as fronteiras reais entre cenas foram determinadas. Esta base confiável não foi disponibilizada para acesso público. Gao e Ma relatam em seu trabalho os resultados alcançados por duas outras técnicas de segmentação temporal de vídeo em cenas na mesma base confiável. A eficácia foi medida usando o conjunto de medidas Precisão/Abrangência e os resultados de eficácia reportados são apresentados na Tabela 5.

Tabela 5 – Resultados médios de F_{PR} reportados em Gao e Ma (2012)

Técnica	F _{PR}
Gao e Ma (2012)	0,83
Rasheed e Shah (2005)	0,64
Kyperountas, Kotropoulos e Pitas (2007)	0,74

A técnica de Gao e Ma apresentou eficácia consideravelmente superior às das outras duas técnicas experimentadas na mesma base confiável. A comparação com a técnica de Gao e Ma foi impedida pela indisponibilidade de acesso à base confiável. Além disso, não foram relatadas informações necessárias à reprodução adequada do experimento como, por exemplo, o parâmetro de tolerância do conjunto de medidas Precisão/Abrangência.

Baraldi, Grana e Cucchiara (2015a) propuseram uma abordagem bimodal fusão prévia baseada em aprendizado profundo para o problema da segmentação temporal de vídeo em cenas. Sua técnica emprega características visuais extraídas por meio de aprendizado profundo e também características textuais de médio nível semântico. A Figura 30 ilustra a abordagem proposta por Baraldi, Grana e Cucchiara (2015a), onde a cor azul indica a modalidade visual e a cor rosa indica a modalidade textual. A cor laranja indica a informação temporal.

Figura 30 – Visão geral da técnica fusão prévia de Baraldi, Grana e Cucchiara (2015a).

O conteúdo visual de cada tomada é representado pela saída de uma Rede Neural Convolucional (LECUN; BENGIO, 1998), cuja entrada é o quadro central daquela tomada. O conteúdo textual de uma tomada é representado por vetores de características de médio nível semântico *Bag-of-Words* computados por extratores de características Word2vec (MIKOLOV *et al.*, 2013). Estes dois vetores de características, juntamente com o índice da tomada em questão, são concatenados, e então o vetor resultante desta concatenação é fornecido como entrada a uma das ramificações de uma Rede Siamesa Profunda (BROMLEY *et al.*, 1993). A outra ramificação da Rede Siamesa Profunda recebe entrada produzida de modo similar, representando outra tomada. A Rede Siamesa Profunda produz representações fundidas em cada uma de suas ramificações, entregues às camadas superiores da rede, que desempenham o papel de função de similaridade. Mais especificamente, a rede proposta por Baraldi, Grana e Cucchiara (2015a) é treinada usando uma função de perda contrastiva (*contrastive loss*) com regularização de norma l_2 quadrática

presente na Equação 3.3:

$$L(w) = \frac{\lambda}{2} \|w\|_2^2 + \frac{1}{2N} \sum_{(i,j) \in D} y_{ij} d_{ij}^2 + (1 - y_{ij}) \max(1 - d_{ij}^2, 0) \quad (3.3)$$

onde λ é o parâmetro de regularização do termo de norma l_2 quadrática, w corresponde aos pesos da rede neural, d_{ij}^2 é a distância l_2 quadrática entre as tomadas i e j , D é o conjunto de pares de treinamento, N é o tamanho de D e $y_{ij} \in \{0, 1\}$ é o rótulo, onde 0 denota um par diferente e 1 denota um par igual. Uma matriz de similaridade entre tomadas é computada, convertendo as distâncias d_{ij} em valores de similaridade pelo emprego de um *kernel* gaussiano. Esta matriz serve de entrada para a execução de um algoritmo de Agrupamento Espectral (YU; SHI, 2001). As fronteiras entre cenas são posicionadas entre tomadas que são temporalmente adjacentes no vídeo e que forem atribuídas a grupos diferentes pelo algoritmo de agrupamento espectral.

Por ser baseada em aprendizado profundo, essa técnica efetua etapas de treinamento para extrair as características visuais e para o cálculo de similaridade entre tomadas. Após treinado, é limitada a segmentar vídeos contendo padrões semelhantes aos dos vídeos dos dados de treinamento. Essa técnica foi testada na base confiável BBC *Dataset*, descrita na Seção 2.5.1, que foi disponibilizada pelos autores para acesso público. A eficácia foi medida usando o conjunto de medidas *Coverage/Overflow* e os autores reportaram os resultados de outras duas técnicas na mesma base confiável. Esses resultados são apresentados na Tabela 6.

Tabela 6 – Resultados médios de F_{CO} reportados em Baraldi, Grana e Cucchiara (2015a)

Técnica	F_{CO}
Baraldi, Grana e Cucchiara (2015a)	0,62
Chasanis, Likas e Galatsanos (2009)	0,45
Sidiropoulos <i>et al.</i> (2011)	0,54

A técnica de Baraldi, Grana e Cucchiara alcançou eficácia superior às das outras duas técnicas experimentadas na base confiável BBC *Dataset*. Os resultados relatados para as técnicas de Chasanis, Likas e Galatsanos (2009) e Sidiropoulos *et al.* (2011) são inferiores aos reportados por seus respectivos autores em seus trabalhos originais. A menor eficácia observada na base confiável BBC *Dataset* por essas técnicas se deve, principalmente, à complexidade da base confiável.

O trabalho de Baraldi, Grana e Cucchiara foi utilizado como referência de eficácia em uma comparação experimental nesta tese, já que tanto a base confiável empregada nos experimentos quanto o *script* de avaliação foram disponibilizados pelos autores. Essa comparação é relatada no Capítulo 5.

Rotman, Porat e Ashour (2017) apresentaram uma técnica bimodal de segmentação temporal de vídeo que combina informações aurais e visuais por meio de uma abordagem intitulada fusão intermediária (*intermediate-fusion*), que se propõe a efetuar uma única etapa de agrupamento de tomadas em cenas, sem realizar fusão de características. A Figura 31 ilustra a abordagem proposta pelos autores, onde a cor azul indica a modalidade visual e a cor rosa indica a modalidade aural.

Figura 31 – Visão geral da técnica de Rotman, Porat e Ashour (2017).

Na técnica descrita pelos autores, o conteúdo visual de cada tomada é representado por histogramas RGB e o conteúdo aural de cada tomada é representado por vetores de características MFCC. Eles não descrevem o emprego de quadros chave ou qualquer outra estratégia semelhante para reduzir o volume de dados a ser processado por tomada. Os vetores de características extraídos de uma modalidade x são ordenados pelos índices de suas respectivas tomadas de origem, compondo uma série X . Do ponto de vista teórico, o objetivo da técnica é encontrar uma outra série monotonicamente crescente $\bar{t}$, composta por K índices de elementos de X (K é estimado por meio da Estatística da Lacuna (TIBSHIRANI; WALTHER; HASTIE, 2000)), que define um particionamento de X . Cada elemento de $\bar{t}$ indica o último vetor de características de um segmento de X , segmento este que corresponde a uma cena no vídeo em análise. A escolha da série $\bar{t}$ é reduzida à solução de um problema de otimização combinatória, onde se deseja encontrar $\bar{t}$ que minimize uma função de custo $H_x(\bar{t})$, uma soma dos custos de todos os segmentos de um particionamento $\bar{t}$ de X . Os autores propuseram um algoritmo de programação dinâmica, chamado Agrupamento Sequencial Ótimo (*Optimal Sequential Grouping*), cuja recorrência principal determina a melhor dentre todas as possibilidades de seccionamento de um segmento, considerando um seccionamento ótimo para os segmentos restantes. A técnica proposta determina a execução do Agrupamento Sequencial Ótimo individualmente nos conjuntos de vetores de características das modalidades visual usando distância de Bhattacharyya, e aural, usando distância euclidiana. As matrizes produzidas pelo algoritmo de programação dinâmica no cálculo dos agrupamentos são utilizadas para computar uma outra tabela, preenchida de acordo com a Equação 3.4, com $1 \leq n \leq N$ e $1 \leq k \leq K$:

$$\mathcal{J}(n, k) = \underset{i}{\operatorname{argmin}} \{ \min(\hat{G}_v^{n, k}(i), \hat{G}_a^{n, k}(i)) \} \quad (3.4)$$

onde N é a quantidade de vetores de características da série X , K é a quantidade de segmentos em que se pretende particionar X , $\hat{G}_v^{n,k}(i)$ e $\hat{G}_a^{n,k}(i)$ são os custos para particionar n vetores de X em k segmentos, para as modalidades visual e aural, respectivamente. Os valores de $\hat{G}_v^{n,k}(i)$ e $\hat{G}_a^{n,k}(i)$ são previamente padronizados por z -score para permitir a comparação direta. A série $\bar{t}$ de menor custo computada com base nas informações bimodais é finalmente convertida em índices de tomadas onde se encontram as fronteiras entre cenas para compor a anotação produzida pela técnica.

A técnica de [Rotman, Porat e Ashour \(2017\)](#) não realiza qualquer análise para identificar correlação entre as informações intermodais e, com isso, possui as mesmas limitações em eficácia dos métodos fusão tardia. Apesar dos autores alegarem que ela efetua uma única etapa de segmentação em cenas, são realizados dois processamentos equivalentes a segmentações unimodais individuais, já que o Agrupamento Sequencial Ótimo é executado uma vez para cada modalidade, produzindo um particionamento de índices de vetores de características que corresponde a uma segmentação do vídeo original em cenas. Essa técnica foi testada na base confiável OVSD *Dataset*, descrita na Seção 2.5.1, disponibilizada para acesso público pelos autores. A eficácia foi medida usando o conjunto de medidas *Coverage/Overflow* e os autores reportaram os resultados de outras duas técnicas quando executadas na mesma base confiável. Esses resultados são apresentados na Tabela 7.

Tabela 7 – Resultados médios de F_{CO} reportados em [Rotman, Porat e Ashour \(2017\)](#)

Técnica	F_{CO}
Rotman, Porat e Ashour (2017)	0,60
Sidiropoulos et al. (2011)	0,52
Baraldi, Grana e Cucchiara (2016)	0,33

A técnica de Rotman, Porat e Ashour obteve resultados superiores aos das outras duas técnicas em relação à medida F_{CO} na base confiável OVSD *Dataset*. Mais uma vez os resultados dessas outras técnicas se revelaram inferiores aos reportados por seus respectivos autores em seus trabalhos originais. É interessante observar, entretanto, que a F_{CO} média obtida pela técnica de [Sidiropoulos et al. \(2011\)](#) neste trabalho é próxima da obtida na avaliação de [Baraldi, Grana e Cucchiara \(2015a\)](#), o que pode ser visto como um indício de que as bases empregadas na avaliação original favoreceram tal técnica. A explicação para a obtenção de menores valores de F_{CO} nessa avaliação é a mesma comentada anteriormente: complexidade da base confiável OVSD *Dataset*.

A técnica proposta por [Rotman, Porat e Ashour \(2017\)](#) foi modificada em [Rotman et al. \(2018\)](#), pela substituição dos vetores de características unimodais de baixo nível semântico por vetores de características extraídos por redes neurais profundas. Mais especificamente, a técnica modificada de Rotman emprega a arquitetura Inception-v3 ([SZEGEDY et al., 2016](#)), previamente treinada na base confiável ImageNet ([DENG et](#)

al., 2009), para extrair vetores de características dos quadros centrais de cada tomada e representar conteúdo visual. Realiza-se tal operação pela captura dos valores de saída dos neurônios da camada imediatamente anterior à camada final de classificação da rede neural convolucional que compõe a Inception-v3. Cada vetor de características resultante deste processamento tem 2048 dimensões e representa uma tomada. A técnica modificada de Rotman também emprega dados aurais, extraídos por uma versão modificada (HERSHEY *et al.*, 2017) do modelo de rede neural profunda VGG (SIMONYAN; ZISSERMAN, 2015), que extrai um vetor de características de 128 dimensões a cada 0,96 segundos de áudio. Os autores não especificam como foi realizado o treinamento desta rede neural profunda para extração de informação aural.

Após a extração de vetores de características aurais e visuais, a técnica modificada de Rotman efetua um cálculo de normas L_2 par a par entre os vetores de características de cada modalidade, dando origem a uma matriz de distâncias por modalidade. Estas duas matrizes de distâncias são normalizadas individualmente e, em seguida, é calculada uma matriz única de distâncias onde cada posição possui a média dos valores de posições correspondentes na matriz de modalidade aural e na matriz de modalidade visual. A combinação de matrizes é, essencialmente, uma média das similaridades entre cada par de tomadas, permitindo que ambas as modalidades contribuam na decisão de particionamento. A matriz de distâncias médias serve de entrada para uma execução do algoritmo de Agrupamento Sequencial Ótimo, produzindo as melhores possibilidades de seccionamento do conjunto. A série $\bar{t}$ de menor custo é convertida em índices de tomadas e se obtém a lista de posições de fronteiras entre cenas.

A técnica modificada de Rotman assim como a original, não realiza qualquer análise de correlação entre informações intermodais. Nessa versão estendida há apenas uma execução do Agrupamento Sequencial Ótimo, aproximando a técnica de uma abordagem fusão prévia, em relação à versão original. O cálculo de distâncias par a par entre os vetores de características para cada modalidade, no entanto, continua sendo computacionalmente custoso. Essa técnica também foi avaliada sobre a base confiável OVSD *Dataset*, descrita na Seção 2.5.1, de acesso público, usando o conjunto de medidas de avaliação *Coverage/Overflow*. A versão modificada da técnica de Rotman foi comparada com os resultados reportados em Rotman, Porat e Ashour (2017), apresentados na Tabela 7. Seu resultado médio de F_{CO} foi de 0,64, o que representa um aumento de 0,04 em relação à técnica original. Tal aumento se deve, provavelmente, pelo emprego de características extraídas por redes neurais profundas, conhecidas pelo seu alto poder representativo.

3.2 Trabalhos em classificação de vídeos

A classificação de vídeos também é considerado um problema em aberto (MA *et al.*, 2018; ZHANG *et al.*, 2018; XU *et al.*, 2018; SOLTANIAN; GHAEMMAGHAMI, 2019) devido à Lacuna Semântica, já que, para atribuir um rótulo a um vídeo cujo conteúdo é desconhecido, é necessário que o sistema computacional seja capaz de reconhecer algum aspecto de seu conteúdo semântico. Assim como na segmentação temporal de vídeo em cenas e em outras tarefas de análise de vídeo, o emprego de multimodalidade é fundamental para que se obtenha boa eficácia na maioria dos domínios de dados existentes. O arcabouço geral de classificação de vídeos é composto por uma etapa de extração de características de uma ou mais modalidades, uma etapa de fusão prévia ou tardia e uma etapa de classificação, onde cada classe é aprendida por um ou mais classificadores (TZELEPIS *et al.*, 2016).

Há uma grande quantidade de trabalhos em classificação de vídeos disponíveis na literatura como, por exemplo, os de Jiang *et al.* (2011), Liu *et al.* (2013), Jhuo *et al.* (2014), Wu *et al.* (2014), Zhao, Song e Su (2016), Ma *et al.* (2018), Zhang *et al.* (2018), Soltanian e Ghaemmaghami (2019). Um dos atributos pelos quais se pode classificar os trabalhos multimodais existentes é o do tipo de fusão efetuada: prévia ou tardia. Considerando o tema desta tese, o foco da discussão sobre métodos existentes de classificação de vídeo será sobre os multimodais que efetuam fusão prévia.

Uma importante abordagem multimodal de classificação de vídeos é a de Jiang *et al.* (2011), que propõe a realização da fusão de características de maneira ingênua, por uma concatenação dos vetores de entrada. Os vetores concatenados resultantes são fornecidos a um arcabouço de classificação multirrótulos *One-Vs-All* que conta com um SVM por classe a ser aprendida. Outra abordagem interessante é a apresentada por Wu *et al.* (2014), que propõe realizar a classificação usando uma rede neural profunda destinada à classificação multiclases de cada vetor de características dado como entrada. Nesse método, cuja entrada é uma simples concatenação dos vetores de características disponíveis (semelhante ao que se faz em fusão prévia ingênua), a fusão de características é delegada a camadas intermediárias da rede neural profunda, por meio da incorporação de uma equação de regularização que explora correlação existente na entrada. O resultado da classificação é computado nas camadas finais da rede neural profunda. Recentemente, (LIU; YUAN; WANG, 2019b) propuseram a aplicação de uma camada *multimodal factorized bilinear pooling* para efetuar a fusão de representações aurais e visuais em redes de aprendizado profundo. O resultado da fusão é entregue a uma camada de classificação composta por um classificador *Mixture-of-Experts* que produz a classificação final.

Dentre todos os disponíveis, o trabalho de Jhuo *et al.* (2014) foi escolhido para que se realize uma discussão mais cuidadosa. Isso se deve ao fato de que esse trabalho descreve um método de classificação de vídeo considerado estado da arte na base confiável

CCV Database (SOLTANIAN; GHAEMMAGHAMI, 2019) e de que efetua fusão prévia de informação de maneira próxima à do método M4InFus. Tais fatos, aliados ao de que ele foi experimentado em uma base confiável disponível para acesso público, fazem com que ele seja boa escolha para uma avaliação comparativa experimental.

Jhuo *et al.* (2014) propuseram um método de classificação de vídeos bimodal com fusão prévia, que emprega características aurais e visuais de baixo nível semântico. A Figura 32 ilustra a abordagem proposta pelos autores, onde a cor azul indica a modalidade visual e a cor rosa indica a modalidade aural.

Figura 32 – Visão geral da técnica de Jhuo *et al.* (2014).

O método proposto representa o conteúdo visual de cada vídeo, inicialmente, por meio de vetores de características produzidos pelos extratores de características SIFT (LOWE, 2004) e STIP (LAPTEV *et al.*, 2008). Os vetores de características SIFT são extraídos de um quadro a cada dois segundos de vídeo e os vetores de características STIP são extraídos dos chamados volumes de espaço-tempo locais. Estas unidades são escolhidas e representadas em vetores de características pelo algoritmo descrito em Laptev (2005). Ambos os conjuntos de vetores de características SIFT e STIP são processados, individualmente, por um algoritmo BoF, dando origem, por tomada, a um vetor de características BoF baseado em SIFT e a um vetor de características BoF baseado em STIP. Os vetores de características BoF SIFT/STIP de cada tomada são concatenados e cada vetor resultante de concatenação passa a ser a representação única do conteúdo visual do vídeo correspondente. O conteúdo aural de cada vídeo é representado por vetores de características MFCC, extraídos de janelas temporais de 32 ms, com sobreposição de 50%. Os vetores de características MFCC extraídos dos vídeos também são processados por um algoritmo BoF, que produz um vetor de características BoF baseado em MFCC para representar o conteúdo aural de cada vídeo.

A fusão de informação aural e visual em uma representação bimodal é iniciada pela construção de um grafo bipartido $G = \{V, E\}$, onde $V = V^a \cup V^v$ é o conjunto de vértices e E é o conjunto de arestas. Cada vértice em V^a corresponde a uma palavra aural do dicionário BoF aural e cada vértice em V^v corresponde a uma palavra visual do dicionário

BoF visual. Cada aresta $e_{kl} \in E$ conecta um vértice em $k \in V^a$ a um vértice em $l \in V^v$, com um peso associado s_{kl} calculado de acordo com a Equação 3.5:

$$s_{kl} = \frac{\sum_{i=1}^n h_i^a(k) h_i^v(l)}{\sum_{i=1}^n h_i^a(k) \sum_{i=1}^n h_i^v(l)} \quad (3.5)$$

onde $h_i^a(k)$ corresponde à célula k do histograma BoF da modalidade aural do vídeo i e $h_i^v(l)$ corresponde à célula l do histograma BoF da modalidade visual do vídeo i . Esta equação computa o valor do somatório das probabilidades conjuntas da presença das palavras k e l , dividido por um fator de normalização que penaliza palavras frequentemente presentes no conjunto de vídeos. A próxima etapa do método de fusão proposto determina que se execute o algoritmo de particionamento de grafos bipartidos descrito por [Dhillon \(2001\)](#). Um exemplo de divisão produzida pelo algoritmo de particionamento é ilustrado na Figura 33, onde a cor azul indica a modalidade visual e a cor rosa indica a modalidade aural.

Figura 33 – Ilustração do grafo bipartido de palavras aurais e visuais. A largura das arestas é proporcional ao peso daquela aresta. Quanto mais larga é uma aresta, maior o valor de correlação s relacionado a ela.

Os grupos produzidos na etapa de particionamento são as palavras bimodais que podem ocorrer nos vídeos. O método de [Jhuo et al. \(2014\)](#) determina três estratégias de *pooling* para produzir a representação única baseada nestes grupos: *max*, *average* e *hybrid pooling*, que é uma versão híbrida entre *max/average*. Após a execução de um dos três métodos de *pooling*, as representações bimodais únicas dos vídeos são utilizadas para treinar/testar um arcabouço *one-vs-all* de SVMs com o objetivo de realizar classificação multirrótulos de vídeos.

O método de fusão de características de [Jhuo et al. \(2014\)](#), por ter sido modelado como um grafo bipartido, é limitado à análise de apenas duas modalidades. Além disso, a estratégia *average pooling* descrita pelos autores, que alcançou a melhor eficácia com dicionário de tamanho fixo, só computa histogramas de ocorrência para grupos que possuam ao menos uma palavra de cada modalidade, ignorando grupos compostos por palavras de apenas uma das duas modalidades. Este método foi experimentado nas bases confiáveis

TRECVID MED e CCV *Database* (descrita na Seção 2.5.2). A base TRECVID MED foi disponibilizada apenas aos participantes do evento TRECVID ⁴ e a base CCV *Database* é disponível para acesso público.

3.3 Considerações finais

Este capítulo apresentou uma discussão sobre trabalhos em estado da arte que efetuam fusão de informação multimodal e que são relacionados ao método M4InFus. A discussão abordou duas tarefas clássicas de análise de vídeo em que a multimodalidade é essencial para a obtenção de bons resultados e cujos métodos disponíveis tem sua eficácia limitada por lacunas existentes na pesquisa em fusão de informação multimídia. As duas tarefas de análise de vídeo são a segmentação temporal de vídeo em cenas e a classificação de vídeos. Elas tem disponibilidade pública de bases confiáveis, medidas de avaliação e resultados que permitem avaliação comparativa experimental de eficácia, tornando-as adequadas para o propósito de verificação de ganhos em eficácia pretendido por esta tese.

A Seção 3.1 apresentou uma visão geral sobre a área de segmentação temporal de vídeo em cenas e também uma discussão mais específica sobre os trabalhos em estado da arte que descrevem abordagens multimodais com fusão de informação. Dentre esses trabalhos, figuram os de Sidiropoulos *et al.* (2011), Rotman, Porat e Ashour (2017) e Rotman *et al.* (2018) que, apesar de não realizarem fusão de características (realizam fusão de resultados), são considerados estado da arte em segmentação temporal de vídeo em cenas. Esses três trabalhos são baseados em fusão tardia e, com isso, são incapazes de aproveitar qualquer correlação existentes nos dados intermodais antes de alguma etapa de decisão.

Sidiropoulos *et al.* (2011) reportaram a obtenção de valores de eficácia considerados altos, medidos em F_{CO} , que podem ter sido viabilizados pelo emprego de características de alto nível semântico e pelo mecanismo de parametrização automática dos STGs, computacionalmente custoso. A obtenção de alta eficácia em F_{CO} pela técnica, entretanto, não se repetiu nos experimentos reportados por Baraldi, Grana e Cucchiara (2015a), usando a base confiável BBC *Dataset* e a medida F_{CO} e por Rotman, Porat e Ashour (2017), usando a base confiável OVSD *Dataset* e a medida F_{CO} . A técnica de Sidiropoulos *et al.* (2011) obteve resultados consideravelmente mais baixos em tais experimentos, tendo sido superada por outras técnicas nessas bases confiáveis. Os trabalhos de Rotman, Porat e Ashour (2017) e Rotman *et al.* (2018) são os mais recentes encontrados sobre segmentação temporal de vídeo em cena com emprego de multimodalidade. Ambos os trabalhos reportaram resultados superiores aos de Sidiropoulos *et al.* (2011) e de Baraldi, Grana e Cucchiara (2016), tornando-os interessantes referências para comparação de eficácia com

⁴ Maiores informações em <<https://trecvid.nist.gov/>>.

outros trabalho nessa tarefa.

O trabalho de [Gao e Ma \(2012\)](#) apresenta uma técnica que realiza fusão prévia de vetores de características. Essa técnica é limitada em relação alguns a aspectos necessários para que se produza uma representação expressiva. As limitações são a incapacidade de representar padrões unimodais, a restrição de identificação de correlação dentro de cada tomada e a limitação a duas fontes de informação para fusão. Todas essas lacunas são cobertas pelo método M4InFus mas, infelizmente, não é possível compará-los dada a indisponibilidade tanto do código, quanto da base confiável empregada nos experimentos.

O trabalho de [Baraldi, Grana e Cucchiara \(2015a\)](#) descreve uma abordagem fusão prévia para a segmentação temporal de vídeo em cenas baseada em aprendizado profundo. A rede de aprendizado profundo é composta por uma combinação de duas redes neurais convolucionais e de uma rede siamesa. Tal rede é limitada a fusão das duas fontes de informação para as quais foi projetada. Além disso, a representação gerada para representação de cada tomada é desprovida de referência para os valores originais dos vetores de características de entrada, impossibilitando pós processamento. Além disso, por ser baseada em aprendizado, muito provavelmente apresentará resultados ruins em instâncias com padrões diferentes dos existentes nos dados de treino. Todas essas lacunas são cobertas pelo M4InFus, fazendo com que seja interessante comparar o trabalho de [Baraldi, Grana e Cucchiara \(2015a\)](#) com uma técnica de segmentação temporal de vídeo em cenas que empregue o método M4InFus.

O próximo capítulo descreve o método M4InFus, destinado a fusão de informação multimodal. Esse método é escalável em relação à quantidade de modalidades ou mesmo de diferentes tipos de características a serem fundidos e que, além disso, permite a representação simultânea de palavras unimodais e multimodais em um mesmo vetor de características, de modo que a informação produzida se mantém rastreável a origem, permitindo processamento específico para refinar ainda mais a semântica presente.

DESCRIÇÃO DO MÉTODO

Este capítulo apresenta o método M4InFus de fusão multimodal de características de vídeo, destinado a produzir representações únicas com base em múltiplas representações de segmentos de vídeo de uma coleção. Isso é feito de modo que tais representações únicas por segmento conttenham informações relevantes provenientes das múltiplas representações originais. Essas representações únicas são ditas **fundidas**. O método proposto é independente de domínio de aplicação e foi projetado para ser escalável em relação à quantidade de modalidades a serem fundidas, oferecendo também flexibilidade ao ser capaz de produzir representações que conttenham, simultaneamente, padrões unimodais e multimodais. A Figura 34 apresenta uma visão em alto nível dos passos do método.

Figura 34 – Visão geral do método de fusão proposto.

Conforme pode ser visto na Figura 34, o M4InFus é composto por quatro etapas. A primeira delas (Extração) determina múltiplos processos de produção de vetores de características, um por modalidade, extraídos dos segmentos de vídeo de entrada. A entrada do método M4InFus é flexível em relação à quantidade de segmentos de vídeo, ao tipo e duração dos segmentos de vídeo, permitindo o acoplamento do método a técnicas de diversas aplicações. Em um problema como o da segmentação temporal de vídeo em cenas,

por exemplo, cada tomada corresponde a um segmento do conjunto de entrada para que, ao final, obtenha-se uma representação fundida por tomada. Já em um problema como o da classificação de vídeos, cada vídeo de treinamento/teste corresponde a um segmento de vídeo do conjunto de entrada para que, ao final, obtenha-se uma representação fundida por vídeo.

A segunda etapa do método (Refinamento) se dá pela produção, a partir das características unimodais brutas, de representações em formato comum por meio de características de médio nível semântico. Essa etapa é efetuada individualmente por modalidade. Na terceira etapa (Identificação de co-ocorrência) é realizada uma análise das características unimodais de médio nível semântico em busca de correlação inter/intra modalidades. Por fim, na última etapa (Síntese da representação fundida), uma representação única por segmento de vídeo é produzida, a partir das representações de médio nível semântico e das correlações descobertas entre elas. As Seções 4.1, 4.2, 4.3 e 4.4 descrevem, respectivamente e detalhadamente, as Etapas 1, 2, 3 e 4.

4.1 Extração de características

O M4InFus é destinado a realizar fusão de informação proveniente de m modalidades para produção de representações únicas de segmentos de vídeo. A entrada do método é composta por vários conjuntos de vetores de características, um por modalidade/extrator de características. Cada segmento de vídeo do conjunto de entrada é representado por m subconjuntos dos conjuntos de entrada (um subconjunto por modalidade/extrator).

Os vetores de características de entrada do M4InFus podem ser originados de qualquer modalidade/descritor, desde que sejam compostos por elementos que permitam a comparação desses vetores com outros vetores de mesma origem. Isso se faz necessário para que se possa aplicar o método BoF, que efetua comparações entre vetores de características em várias de suas etapas. Os vetores de características de entrada não possuem restrição quanto à dimensionalidade.

A representação fundida de cada segmento de vídeo é construída de modo iterativo onde, inicialmente, os segmentos de vídeo são representados por múltiplos vetores de características, extraídos dos dados pertencentes a cada uma das modalidades que se deseja adotar no processo de fusão. Devem ser extraídos vetores de características para cada uma das modalidades escolhidas. As características extraídas representam trechos de informação de cada segmento de vídeo, possibilitando seu refinamento por meio do modelo BoF, cuja intenção é produzir uma representação única e resumida de múltiplas representações correlacionadas de padrões. A Figura 35 contém uma abstração do processo de extração de características.

Os dados que compõem um segmento de vídeo são massivos e costumam conter

Figura 35 – Visão geral da extração de características no método M4InFus.

uma grande fração de informação redundante. Por esse motivo, conforme ilustrado na Figura 35, realiza-se inicialmente uma redução dos dados originais do segmento de vídeo, indicada na Figura 35 por (i), por meio da seleção de trechos que sejam representativos em relação ao conteúdo semântico. Em relação à extração de características da modalidade visual, é realizada a seleção de um ou mais quadros chave para representar o conteúdo de cada segmento de um determinado vídeo ou até mesmo de todo o vídeo, dependendo da tarefa em questão. Um **quadro chave** é um dos quadros que compõem a imagem em movimento do segmento de vídeo, tal que seja o quadro que melhor represente todos os quadros da imagem em movimento do segmento de vídeo. O uso de quadros chave para representar o conteúdo visual é prática comum nas tarefas de análise de vídeo (SIDIROPOULOS *et al.*, 2011; JHUO *et al.*, 2014; BARALDI; GRANA; CUCCHIARA, 2015a). A seleção de quadros chave é um problema individual. A escolha de poucos quadros chave pode ser insuficiente para representar a diversidade do conteúdo de um determinado segmento de vídeo. Por outro lado, a escolha de muitos quadros chave pode fazer com que informação redundante seja representada e processada desnecessariamente. A quantidade de quadros chave costuma ser determinada empiricamente em cada aplicação.

O mesmo raciocínio da extração de quadros chave na modalidade visual se aplica, de maneira análoga, à modalidade aural de vídeos. Pode ser escolhido um subconjunto de

amostras que melhor representam o conteúdo aural de todo o segmento de vídeo a ser analisado tal que seja, ao mesmo tempo, compacto o suficiente para possibilitar e/ou agilizar o processamento. A informação necessária para representar o conteúdo aural, entretanto, é consideravelmente menor do que a necessária para representar o conteúdo aural de um segmento de vídeo. Por esse motivo, o conjunto de amostras de áudio escolhidas para representar o conteúdo aural costuma abranger um intervalo de tempo maior do que os intervalos de tempo representados por quadros chave.

Outras modalidades também podem ser representadas por subconjuntos dos segmentos de vídeo para reduzir o custo computacional e, adicionalmente, descartar informação redundante. Um detalhe que merece atenção diz respeito à sincronia temporal das amostras de diferentes modalidades. Em alguns dos métodos fusão prévia existentes como, por exemplo, o de [Baraldi, Grana e Cucchiara \(2015a\)](#), é necessário que haja sincronia temporal entre as amostras representativas de diferentes modalidades. Isso acontece pois a informação complementar entre diferentes modalidades costuma ocorrer de modo sincronizado no segmento de vídeo e tais métodos se aproveitam desse fato para identificar correlação. No método M4InFus a correlação é identificada por co-ocorrência em todo o segmento de vídeo, fazendo com que não seja necessário que haja sincronia temporal entre amostras de diferentes modalidades.

Ao fim da redução dos dados vem a extração de características, indicada na Figura 35 por (ii). Para tal, são executados um ou mais extratores de características, discutidos anteriormente na Seção 2.2, sobre as amostras representativas produzidas em (i). O objetivo da extração de características é a produção de vetores de características que representem aspectos visuais, aurais, dentre outros, que sejam úteis a tarefa fim de análise de vídeo. Diferentes extratores de características costumam produzir vetores de características de significados completamente diferentes e que, por esse motivo, não são diretamente comparáveis. Assim, cada conjunto de vetores de características produzido por modalidade deve ser processado individualmente para a computação de representações que contenham informações de mesmo significado (apesar de ainda não serem diretamente comparáveis). Tais representações, no método M4InFus, são vetores de características BoF. Cada conjunto de vetores de características de uma determinada modalidade extraídos em (ii) é provido como entrada para uma execução individual da etapa seguinte do método M4InFus, descrita na Seção 4.2.

4.2 Representação por *bag-of-features*

A etapa seguinte do método de fusão proposto tem duas funções principais: extrair a semântica latente contida nos vetores de características codificando-a em uma representação concisa e representativa e, também, dar início ao processo de transferên-

cia das informações unimodais representadas de maneira heterogênea para um espaço de representação comum. Estes dois objetivos são alcançados computando vetores de características BoF para cada modalidade, a partir dos vetores de características extraídos diretamente dos dados brutos. Converter as diferentes representações unimodais para um espaço de representação comum lida parcialmente com o dilema da heterogeneidade de representação de características de diferentes extratores de características. Isso acontece pois o significado dos elementos dos vetores de características BoF são iguais em qualquer vetor de características BoF. A heterogeneidade de representação de características ainda é um problema em aberto na literatura (GIALAMPOUKIDIS *et al.*, 2019). A Figura 36 ilustra a primeira parte do processo de computação de vetores de características BoF unimodais.

Figura 36 – Início do processo de produção de características BoF.

O conjunto com todos os f vetores de características provenientes de um extrator de características passa, inicialmente, por um agrupamento representado na figura por (i), produzindo k grupos, representados na Figura 36 por retângulos de cor vermelha. Os grupos indicam a existência de padrões naquela modalidade. Como exemplo, espera-se que vetores de características MFCC que representam o som da voz de um locutor específico sejam atribuídos a um mesmo grupo, por serem mais próximos um do outro do que dos demais vetores de características MFCC.

Cada grupo/padrão é então representado por um centróide, processo representado na Figura 36 por (ii), que é um vetor de características sintetizado com as médias dos vetores de características que compõem aquele grupo. O conjunto com os k centróides produzidos neste processo é chamado dicionário de características e a cada um destes centróides é dado o nome de **palavra**, em analogia ao modelo BoW que emprega um dicionário de palavras textuais. A produção do dicionário é representada na Figura 36 por (iii). Além disso, costuma-se explicitar na terminologia a modalidade de origem da palavra (*e.g.* palavra visual/aural). A etapa seguinte, e final, da computação de vetores de características BoF tem como objetivo a produção, por modalidade, de um vetor de características que represente cada segmento de vídeo da entrada, tal que o significado das

células desses vetores seja o mesmo: contagem de ocorrências de um padrão. Essa etapa é ilustrada na Figura 37, por meio da contagem de ocorrências das palavras do dicionário de características em cada um dos segmentos de vídeo.

Figura 37 – Etapa final da produção de características BoF.

O processo ilustrado na Figura 37 é efetuado comparando cada vetor de características proveniente de um segmento de vídeo com todas as palavras do dicionário de características. É contabilizada, para cada um destes vetores, uma ocorrência da palavra que mais se assemelha àquele vetor de características em questão. O conjunto de contagens de ocorrências das palavras em um segmento de vídeo é armazenado em um vetor chamado histograma de ocorrência de palavras (ou vetor de características BoF) que é considerado uma representação de médio nível semântico desse segmento de vídeo. Há um mapeamento entre o índice da célula de um vetor de características BoF de uma modalidade e cada palavra daquela modalidade (por exemplo, a terceira célula de qualquer vetor de características BoF é relacionado à palavra w_3 , s.p.g), permitindo a comparação entre vetores por meio de uma medida de similaridade/distância entre vetores unidimensionais.

O raciocínio por trás da representação pelo modelo BoF é o de que segmentos de vídeo com histogramas de ocorrência semelhantes devem apresentar padrões semelhantes naquela modalidade e, assim, serem relacionados a conceitos semânticos semelhantes. Por ser possível identificar a presença de padrões e até mesmo comparar diferentes segmentos de vídeo por seus padrões sem que haja qualquer tipo de conhecimento sobre o significado ou interpretação destes padrões, essa informação classificada como **semântica latente**.

Apesar dos vetores de características BoF de diferentes modalidades conterem contagens de padrões em suas células e, possivelmente, possuírem mesmos comprimentos, os padrões representados por eles em células de mesmo índice são diferentes e provavelmente sequer relacionados. Uma comparação direta entre vetores de características BoF de diferentes modalidades não tem utilidade aparente. É necessário um processamento adicional para concluir a transferência das informações heterogêneas unimodais para um espaço de representação comum. Este processamento tem seu início descrito na Seção 4.3.

4.3 Detecção de co-ocorrência

A detecção de co-ocorrência completa o processo que permite a comparação entre informações unimodais representadas de maneira heterogênea e ainda lida com outro dilema da fusão prévia: a falta de sincronismo temporal entre informações de diferentes modalidades. Esta etapa tem como entrada os vetores de características BoF produzidos para as diferentes modalidades na etapa descrita na Seção 4.2 e produz como saída um mapeamento dos padrões correlacionados por co-ocorrência indicando quais deles, em conjunto, participam da expressão de um padrão específico presente no conjunto de segmentos de vídeo. No método proposto, o início desse processo se dá por meio de uma conversão dos histogramas de ocorrência de todas as palavras de cada modalidade em cada segmento de vídeo (histograma de ocorrências por segmento de vídeo) em histogramas de ocorrência de cada palavra em todos os segmentos de vídeo (histograma de ocorrências por palavra). A Figura 38 ilustra esta conversão para uma das modalidades, onde cores iguais indicam que se tratam de contagens de uma mesma palavra.

Figura 38 – Computação dos histogramas de ocorrência de cada palavra em todos os segmentos de vídeo.

A conversão é realizada por meio de uma operação de transposição da matriz formada pela sobreposição dos histogramas de ocorrências de palavras em cada segmento de vídeo. Os histogramas de ocorrência por palavra contêm o padrão de ocorrência daquela palavra em diferentes segmentos de vídeo. Dessa forma, o cálculo de similaridade/dissimilaridade entre histogramas de ocorrências por palavra corresponde à comparação de palavras em relação às suas ocorrências nos segmentos de vídeo do conjunto. A computação dos histogramas de ocorrência por palavra de todas as modalidades permite a comparação, relativa ao padrão de ocorrência, entre palavras de diferentes modalidades. Como palavras correspondem a padrões nos segmentos de vídeo, ao comparar palavras

de diferentes modalidades estarão se comparando, essencialmente, os padrões que as originaram. Baseando-se neste fato, o processo de detecção de co-ocorrência se completa pelo agrupamento de padrões unimodais que ocorrem em proporções semelhantes nos mesmos segmentos de vídeo do conjunto de entrada. Com isso, espera-se que sejam identificados padrões unimodais complementares que compõem padrões multimodais no segmento de vídeo. Essa análise é realizada pela etapa ilustrada na Figura 39, onde cores iguais indicam que se trata de uma mesma modalidade.

Figura 39 – Agrupamento de palavras unimodais com base em seus padrões de ocorrência em diferentes segmentos de vídeo.

A etapa ilustrada na Figura 39 é efetuada pela aplicação de um algoritmo de agrupamento, agrupando as palavras w_{ij} em k grupos pela distância entre seus histogramas de ocorrência por palavra, onde i é o índice de uma modalidade e j é o índice de cada palavra da modalidade i , com $1 \leq i \leq m$ e $1 \leq j \leq k$. Os grupos f_o , com $1 \leq o \leq s$ produzidos neste processo contém palavras (padrões unimodais) que ocorrem em quantidades semelhantes em determinados segmentos de vídeo. Ao determinar esses grupos, espera-se identificar padrões unimodais que se complementam durante a expressão, no segmento de vídeo, de um conceito semântico. É importante ressaltar que esse processo é flexível em relação à capacidade de detectar padrões uni/multimodais, ou seja, ele também identifica padrões unimodais compostos por apenas uma ou mais palavras de uma mesma modalidade. Um exemplo de padrões unimodais que complementam um ao outro pode ser visto na Figura 40.

Os dois quadros presentes na Figura 40, são provenientes de segmentos de vídeo diferentes e contém padrões visuais semelhantes. Além disso, há padrões aurais – sons de pinguins – que também ocorrem nos mesmos dois segmentos de vídeo. Em um processo de agrupamento como o recém descrito, tais padrões visuais e aurais provavelmente seriam atribuídos a um mesmo grupo, já que os padrões visuais presentes na representação

Figura 40 – Exemplo de co-ocorrência de padrões aurais/visuais em diferentes segmentos de vídeo.

gráfica de um pinguim são, na *BBC Dataset*, sempre acompanhados de sons de pinguins. Terminado o processo de computação de grupos de palavras unimodais correlacionadas, é necessário gerar uma representação única para a ocorrência de cada grupo em cada segmento de vídeo que combine informações de ocorrência de seus respectivos padrões unimodais componentes. Esse processo é descrito na Seção 4.4.

4.4 Síntese em representação única

A etapa final do método de fusão de características proposto lida com o dilema restante da fusão prévia: produz uma representação única para a expressão de múltiplos padrões correlacionados, mantendo o significado das representações originais, ao mesmo passo em que descarta informações menos significativas. Tal representação única permite pós processamento com o objetivo de ampliar a utilidade da semântica na tarefa fim. A representação única da presença de padrões correlacionados é produzida por meio da técnica de *pooling* (HUBEL; WIESEL, 1962).

A Figura 41 ilustra a aplicação do *pooling* no método de fusão de características proposto, com uma visão detalhada dos histogramas que representam as palavras do grupo f_1 .

Considerando os grupos $f_1, \dots, f_s$, compostos por palavras unimodais, produzidos pela etapa descrita na Seção 4.3, o *pooling* ocorre por meio da aplicação, para cada grupo $f \in \{f_1, \dots, f_s\}$, de uma função que toma como entrada o conjunto dos histogramas de ocorrência em segmentos de vídeo h_w de cada palavra unimodal w que compõe f nos segmentos de vídeo $v_1, \dots, v_n$ e produz como saída um histograma único h^{avg} que representa f .

Figura 41 – Abstração do processo de *pooling*.

Há duas estratégias de *pooling* adequadas à aplicação no método proposto, que diferem em como as ocorrências do padrão correspondente a um determinado grupo serão contabilizadas: *average pooling*, descrito na Seção 4.4.1 e *max pooling*, descrito na Seção 4.4.2. Ambas as estratégias podem, ainda, ser modificadas utilizando uma heurística de ponderação para aprimorar os histogramas de ocorrências de padrões produzidos pela fusão, estratégia está descrita na Seção 4.4.3. Após a realização do *pooling*, é necessário converter os histogramas de ocorrências de cada palavra em todos os segmentos de vídeo em histogramas de ocorrências de todas as palavras por segmento de vídeo, conforme será descrito na Seção 4.4.4.

4.4.1 Average Pooling

O *average pooling*, conforme sugere o nome, determina que seja computada uma média das ocorrências de cada palavra unimodal em cada segmento de vídeo. Os valores das células do histograma de ocorrências h_f^{avg} , do *average pooling* do grupo f nos segmentos de vídeo $v_1, \dots, v_n$, são calculados usando a equação

$$h_f^{avg}[v_i] = \frac{\sum_{w_j \in f} h_{w_j}[v_i]}{|f|}, \quad (4.1)$$

onde h_{w_j} é o histograma da palavra w_j que compõe o grupo f . Um exemplo de aplicação do *average pooling* pode ser visto na Figura 42.

Neste exemplo, o grupo de palavras f_s é composto pelas três palavras unimodais w_{19} , w_{26} e w_{m5} , onde *average pooling* determina que seja realizada inicialmente a soma dos valores individuais de cada coluna dos histogramas destas palavras obtendo os totais. Estes valores totais são então divididos por 3, a quantidade de palavras unimodais que compõem f_s .

A estratégia *average pooling* preserva informação de todas as palavras unimodais

Figura 42 – Exemplo de aplicação do *average pooling* em um grupo de palavras unimodais.

envolvidas, sendo adequada à representação de padrões que costumam ocorrer de modo homogêneo em relação à sua composição, seja ela unimodal ou multimodal. Como exemplo, considerando um padrão multimodal que corresponde a uma explosão e sabendo que, costumeiramente, explosões são expressas de maneira complementar por padrões aurais e visuais, histogramas *average pooling* de dois diferentes segmentos de vídeo contendo explosões serão, muito provavelmente, semelhantes. Já um histograma *average pooling* proveniente de um trovão, cujo som é parecido com o de uma explosão, mas que carece de um padrão visual complementar, vai ser diferente de ambos. Qualquer expressão incompleta de uma explosão, como por exemplo um trecho de segmento de vídeo onde há o som de explosão enquanto a imagem em movimento expressa o rosto de um ator, também produzirá um histograma diferente do esperado para uma explosão, já que a contagem final dos padrões que a representam cairá drasticamente devido à fração faltante dos padrões que a compõem. A estabilidade do *average pooling* é proporcional à quantidade de palavras que compõem cada grupo. Quanto mais palavras compõem um determinado grupo, menor será a influência de trechos faltantes daquele grupo em diferentes expressões do padrão correspondente àquele grupo.

4.4.2 Max Pooling

O *max pooling*, também como sugere o nome, determina que sejam utilizados os máximos dentre todas as ocorrências de palavras unimodais que compõem um determinado grupo em cada segmento de vídeo. Os valores das células do histograma de ocorrências *max pooling* h_f^{max} , do grupo f nos segmentos de vídeo $v_1, \dots, v_n$ são calculados usando a equação

$$h_f^{max}[v_i] = \max_{w_j \in f} h_{w_j}[v_i], \quad (4.2)$$

onde h_{w_j} é o histograma da palavra w_j que compõe o grupo f . Um exemplo de

aplicação do *max pooling* pode ser visto na Figura 43.

Figura 43 – Exemplo de aplicação do *max pooling* em um grupo de palavras unimodais.

Este exemplo utiliza o mesmo grupo de palavras f_s do exemplo apresentado para o *average pooling*, composto pelas três palavras unimodais w_{19} , w_{26} e w_{m5} . O *max pooling* determina que seja escolhido o maior valor de cada coluna dos histogramas das palavras contidas em f_s para compor a coluna correspondente de $h_{f_s}^{max}$.

O *max pooling*, em contraste com o *average pooling*, produz um histograma de ocorrências de padrões baseado na parte mais representativa das ocorrências de palavras unimodais que compõem cada padrão. Dessa forma, o *max pooling* é capaz de representar ocorrências heterogêneas com contagens semelhantes de um mesmo padrão, desde que haja uma palavra unimodal componente deste padrão sendo expressa de maneira semelhante em todas estas ocorrências. Considerando o mesmo exemplo apresentado para o *average pooling*, diferentes expressões de explosões podem possuir conteúdo visual distinto e mesmo assim apresentarem histogramas de ocorrência *max pooling* semelhantes, desde que seu conteúdo aural seja semelhante e mais representativo do que o visual. Apesar de mais flexível, o *max pooling* é mais suscetível a erros de representação já que o valor final de contagem de ocorrências pode ser originado de qualquer uma das palavras unimodais que compõem um determinado padrão.

4.4.3 Pooling ponderado

Dentre as virtudes do método de fusão proposto estão a preservação do significado original nos valores componentes dos vetores de características fundidos, bem como a rastreabilidade das origens destes valores. Graças a estas virtudes, é possível aprimorar os vetores de características fundidas identificando situações onde duas ou mais fontes de informação apresentam padrões de ocorrências semelhantes de palavras unimodais. Duas ou mais fontes de informação indicando a existência de um padrão em um segmento de vídeo aumentam a confiança sobre a existência daquele padrão. Pode-se então expressar a menor probabilidade de erro na identificação de um padrão ponderando sua contagem no

histograma de ocorrências. O destaque destes padrões mais confiáveis aumenta o poder discriminatório das medidas de distância entre histogramas que serão empregadas nas ferramentas de análise de vídeo. A abordagem proposta para realizar a ponderação se dá pela aplicação da Equação 4.3 nos histogramas de ocorrências fundidos h_f^{fused} dos grupos $f \in \{f_1, \dots, f_s\}$, produzindo histogramas de ocorrências ponderados $h_f^{weighted}$:

$$h_f^{weighted}[v] = \begin{cases} h_f^{fused}[v] \times \delta, & \text{se } |f| > 1 \\ h_f^{fused}[v], & \text{caso contrário,} \end{cases} \quad (4.3)$$

onde v é um dos segmentos de vídeo em $\{v_1, \dots, v_n\}$ e δ é o fator de ponderação. Os histogramas h_f^{fused} podem ser histogramas produzidos por *average pooling* ou *max pooling*.

4.4.4 Conversão em histogramas por segmento de vídeo

A informação produzida pelo método M4InFus ao final da execução de uma das estratégias de *pooling* é estruturada para representar os padrões de ocorrências de cada palavra em cada um dos segmentos de vídeo do conjunto. Essa organização não representa cada vídeo individualmente. Por esse motivo é necessário efetuar uma conversão dos histogramas de ocorrências de cada palavra nos segmentos de vídeo em histogramas de ocorrência de todas as palavras em cada segmento de vídeo. Essa conversão é realizada por meio de uma transposição da matriz composta pelos histogramas de ocorrências de cada palavra nos segmentos de vídeo. Tal processo é ilustrado na Figura 44.

Figura 44 – Obtenção dos histogramas de ocorrência de palavras fundidas por segmento de vídeo.

Na Figura 44 é ilustrada, por meio das setas pontilhadas, a produção dos histogramas de ocorrência h_v^{fused} por segmento de vídeo $v \in \{v_1, \dots, v_n\}$, a partir de histogramas

de ocorrência h_f^{fused} , por palavra $f \in \{f_1, \dots, f_s\}$. Os histogramas h^{fused} podem ser histogramas produzidos por *average pooling*, *max pooling* ou ponderação. Esses histogramas são o produto final do M4InFus e servirão de entrada para uma técnica-fim de análise de vídeo.

4.5 Considerações finais

Este capítulo descreveu o método M4InFus, proposto para realizar fusão prévia de informação multimodal. O método M4InFus tem como etapa inicial a extração de características. Nessa etapa, são escolhidas amostras representativas das informações de diferentes modalidades que compõem os segmentos de vídeo, com o objetivo de reduzir a quantidade de informação redundante e o custo computacional. As amostras representativas são processadas, usando um ou mais extratores de características por modalidade, para produzir vetores de características unimodais que capturem e representem algum aspecto daquela modalidade que seja interessante à solução da tarefa fim. Essa etapa foi projetada de modo que não houvesse restrição quanto à origem dos vetores de características, desde que sejam comparáveis entre si por alguma função de distância ou similaridade.

A segunda etapa é o refinamento dos vetores de características unimodais pelo método BoF e se dá pela extração de representações unimodais de mesmo significado (contagem de ocorrências de um padrão) a partir de representações unimodais de significados variados. Essa etapa é o passo inicial do processo que lida com o dilema da heterogeneidade no método M4InFus. O refinamento de características por emprego do método BoF detecta e representa ocorrências de padrões, descartando detalhes específicos de cada tipo de vetor de características. Tal processamento é interessante por permitir a extração de uma representação comum a qualquer natureza de informação. Por outro lado, é descartada uma grande quantidade de informação que poderia ser útil em algum aspecto da tarefa fim. A representação de todo um segmento de vídeo passa a ser feita por um único vetor de características. Por esse motivo, o sincronismo temporal de diferentes padrões dentro de um segmento de vídeo é irrelevante desse ponto em diante.

Em seguida, inicia-se a principal novidade proposta nesta tese. É efetuada uma etapa de identificação de co-ocorrência baseada em agrupamento dos padrões BoF. São identificados grupos de padrões unimodais que compõem padrões possivelmente multimodais. A identificação de relações existentes entre informações intermodais é uma grande vantagem de métodos fusão prévia como o M4InFus em relação a métodos fusão tardia como os propostos em [Sidiropoulos et al. \(2011\)](#), em [Rotman, Porat e Ashour \(2017\)](#) e em [Rotman et al. \(2018\)](#). Nem todo grupo identificado será multimodal, já que há a possibilidade de computação de grupos com apenas um padrão ou mesmo com vários padrões de uma única modalidade. É interessante observar que, com isso, tanto representações unimo-

dais quanto multimodais são detectadas, ao contrário de outras técnicas como a de [Gao e Ma \(2012\)](#) que é limitada à detecção de padrões multimodais. Como esse agrupamento é feito pela similaridade de ocorrência de padrões BoF em todo um segmento de vídeo, permite uma maior versatilidade em relação à sincronização da informação intermodal. Esta etapa provê escalabilidade em relação à quantidade de modalidades a serem fundidas, pois não impõe um limite no número de modalidades ou mesmo a diferentes tipos de características empregadas como entrada. Abordagens como as propostas por [Gao e Ma \(2012\)](#), [Jhuo et al. \(2014\)](#) e [Baraldi, Grana e Cucchiara \(2015a\)](#) são projetadas para efetuar a fusão de uma determinada quantidade de tipos de características.

A etapa seguinte complementa a etapa imediatamente anterior e corresponde ao processo de síntese em representação única. Essa etapa gera representações únicas obtidas por meio do processamento da informação intermodal por técnicas de *pooling*. Tal representação é capaz de preservar o significado original de contagens de ocorrências de padrões unimodais, acompanhadas das contagens de ocorrências de padrões multimodais, caracterizando o método de fusão como sendo flexível em relação à representação de padrões, cobrindo outra lacuna no estado da arte. Técnicas como a de [Gao e Ma \(2012\)](#) e a de [Jhuo et al. \(2014\)](#) não são flexíveis quanto à possibilidade de representar padrões unimodais e multimodais simultaneamente. O trabalho de [Jhuo et al. \(2014\)](#), apesar de empregar *average pooling*, utiliza uma versão que só representa padrões bimodais.

O *pooling* é capaz de resumir trechos de informação redundante em uma menor quantidade de informação, tornando a representação produzida compacta enquanto se mantém discriminativa. Como o *pooling* é realizado por meio de uma função que toma como entrada vetores de características unimodais e calcula, de modo controlado, um vetor de características fundido cujas células mantém o significado da origem, a informação fundida será plenamente rastreável e interpretável, avançando, em mais um aspecto, o estado da arte. Em redes de aprendizado profundo como a que figura no trabalho de [Baraldi, Grana e Cucchiara \(2015a\)](#), não é possível rastrear os valores que compõem uma representação produzida. Além disso, não há interpretação sobre o significado dos valores contidos na representação produzida, dificultando o seu pós processamento em busca de aumento de eficácia.

AVALIAÇÃO EXPERIMENTAL

Este capítulo descreve a avaliação experimental realizada para analisar a eficácia do método M4InFus descrito no Capítulo 4. Os demais métodos de fusão de informação multimídia disponíveis até então foram desenvolvidos para atender a tarefas específicas de análise de vídeo, sem qualquer pretensão aparente de generalização quanto ao domínio de aplicação. Por este motivo, tais métodos de fusão de informação multimídia não foram avaliados isoladamente por seus autores. Não há, para o melhor do conhecimento obtido, um arcabouço ou sequer um protocolo de avaliação destinado especificamente a métodos de fusão de informação multimídia. Assim, uma avaliação do método M4InFus pode ser realizada de maneira indireta, combinando-o a alguma técnica de análise de vídeo e então comparando a eficácia desta técnica resultante com a eficácia de outras técnicas de análise de vídeo com fusão de informação. A avaliação comparativa foi arquitetada de modo a permitir inferir separadamente os ganhos em eficácia propiciados pela etapa de fusão de informação no resultado. Foram realizados três experimentos, sendo dois deles em segmentação temporal de vídeo em cenas, descritos nas Seções 5.1, 5.2 e um em classificação de vídeos, descrito na Seção 5.3.

A maioria dos computadores pessoais atuais é capaz de executar os experimentos supracitados em tempo aceitável. Por questões de conveniência, entretanto, tais experimentos foram executados no *cluster* computacional Euler ¹, da Universidade de São Paulo. A única exceção foi a etapa de computação de características por redes neurais profundas descrita na Seção 5.2. O *cluster* computacional Euler possui 104 lâminas computacionais IP113, cada uma com 2 processadores Intel Xeon E5-2680v2 de 2,8 GHz com dez núcleos cada e 128 GB DDR3 1866MHz de memória principal e mais 40 lâminas computacionais IP113, cada uma com 2 Processadores Intel Xeon E5-2680v4 de 2,4 GHz com quatorze núcleos e 128 GB DDR3 1866MHz de memória principal. O *cluster* computacional Euler emprega o sistema operacional Linux e o gerenciador de tarefas Altair PBS Pro. A etapa

¹ Maiores informações em <<http://www.cemeai.icmc.usp.br/Euler/index.html>>.

de computação de características por redes neurais profundas foi realizada em um computador pessoal com Processador Pentium Intel Core i7 6700K, com 64 GB DDR4 2400 MHz de memória principal, com placa de vídeo nVidia Geforce Titan V, que empregava o sistema operacional Linux.

5.1 Experimento 1: segmentação temporal usando algoritmos *baseline*

No Experimento 1, o método M4InFus foi acoplado a uma técnica de segmentação de vídeo em cenas que emprega apenas algoritmos *baseline*. Nesta tese, *baseline* se refere a um algoritmo, técnica ou método implementado em sua versão original sem incorporar qualquer tipo de otimização. A técnica resultante foi avaliada e seus resultados foram comparados aos de duas técnicas de segmentação de vídeo em cenas multimodais em estado da arte. Uma eventual obtenção de resultados superiores aos de outras técnicas em estado da arte só pode ser consequência do emprego do método M4InFus.

A técnica de segmentação temporal de vídeo em cenas composta por algoritmos *baseline* foi definida no formato de *pipeline*, onde cada etapa produz um resultado que será utilizado na etapa seguinte, sucessivamente, até a produção de uma predição descrevendo posições das fronteiras entre cenas de um vídeo. Para a análise comparativa desta técnica, foi necessária a definição de uma metodologia de avaliação, composta pela implementação do *pipeline* de segmentação, de uma base confiável e de dois conjuntos de medidas de avaliação. Os detalhes da metodologia de avaliação são descritos na Seção 5.1.1. Os resultados do experimento são apresentados na Seção 5.1.2.

5.1.1 Metodologia

A base confiável escolhida para este experimento foi a BBC *Dataset*, descrita na Seção 2.5.1, por ter sido empregada na avaliação de trabalhos relacionados. Além disso, para o melhor do conhecimento obtido, é uma das duas únicas bases públicas e acessíveis atualmente. Os conjuntos de medidas de avaliação são os de Precisão/Abrangência e *Coverage/Overflow/New Overflow*, também descritos na seção 2.5.1. Os *scripts* de avaliação de Precisão/Abrangência e *New Overflow* foram implementados em linguagem Python ² enquanto os *scripts* de avaliação de *Coverage/Overflow* foram providos por Baraldi, Grana e Cucchiara (2015a) e estão codificados em Matlab ³.

Os programas para execução do *pipeline* de segmentação temporal de vídeo em cenas utilizados neste experimento, disponíveis em <<https://github.com/rodrigokishi>>,

² De *Python Software Foundation*, disponível em <<https://www.python.org>>

³ De *The MathWorks, Inc*, disponível em <<https://ch.mathworks.com/products/matlab.html>>

foram escritos em linguagem de programação Java ⁴, empregando a API (*Application Programming Interface*) OpenIMAJ ⁵ (HARE; SAMANGOOEI; DUPPLAW, 2011). A OpenIMAJ é uma biblioteca de ferramentas multimídia que contém implementações dos principais extratores de características como os histogramas de cor, SIFT, *Colored SIFT* (CSIFT) e MFCC, contando também com algoritmos de aprendizado de máquina como o *k-means*. Os extratores de histogramas de cor, CSIFT, MFCC e o método de agrupamento *k-means* empregados seguem as definições sugeridas pelos trabalhos em que foram propostos originalmente (DAVIS; MERMELSTEIN, 1980; LLOYD, 1982; SWAIN; BALLARD, 1991; BURGHOUTS; GEUSEBROEK, 2009).

Pipeline de segmentação temporal de vídeo em cenas

A Figura 45 apresenta uma visão geral do *pipeline* de segmentação em cenas utilizado neste experimento. A linha pontilhada indica o fluxo do pipeline sem o emprego do M4InFus.

Figura 45 – *Pipeline* de segmentação temporal de vídeo em cenas.

A Etapa 1 do *pipeline* experimental, segmentação em tomadas, tem como objetivo determinar quais são as posições dos quadros em que começam e terminam cada tomada do vídeo. Esta etapa não demandou processamento neste experimento, já que a base confiável BBC *Dataset* contém arquivos de anotação das posições de fronteiras entre tomadas. Os autores de Baraldi, Grana e Cucchiara (2015a) utilizaram esta anotação em seus experimentos, permitindo uma comparação justa com outra técnica que também o faça. Esta abordagem é interessante, pois isola a avaliação da segmentação em cenas de qualquer possível erro de uma segmentação em tomadas diferente da real.

Na Etapa 2, de extração de características, foram geradas representações das informações provenientes das modalidades visual e aural de cada tomada. Para a extração de características visuais, dado o fato de que o conteúdo visual de uma tomada é composto por uma quantidade massiva de dados, em grande parte redundantes, obteve-se uma representação reduzida por meio de um ou mais quadros chave, pelos motivos discutidos na

⁴ De Oracle Corporation, disponível em <<https://www.java.com>>

⁵ De The University of Southampton, disponível em <www.openimaj.org>

Seção 4.1. Nesse método, o conjunto de quadros chave de cada tomada foi composto, inicialmente, pelo quadro mais similar a todos os outros. A medida de similaridade adotada é a de intersecção de histogramas de cor (SWAIN; BALLARD, 1991), por ser pouco custosa computacionalmente e por sua comprovada eficácia em tarefas de vídeo relacionadas à extração de semântica (HAN; WU, 2011; BARALDI; GRANA; CUCCHIARA, 2015b). Em seguida, foram acrescentados ao conjunto, sucessivamente, outros quadros da tomada (excluindo-se os já eleitos como quadros chave), tais que fossem os menos similares a todos os quadros do conjunto de quadros chave. Esse processo foi repetido até que não houvesse qualquer quadro na tomada que fosse menos similar do que um determinado limiar, ao quadros já eleitos como chave. O Algoritmo 3 contém um pseudocódigo do processo de seleção de quadros chave descrito.

Algoritmo 3 – Seleção de quadros chave

SELECIONA_KEYFRAMES(F, l, σ): recebe um conjunto não vazio de quadros $F = \{f_1, \dots, f_n\}$, um parâmetro l de limite do processo de seleção e uma função σ que computa um valor de similaridade entre dois quadros. Devolve um conjunto $K = \{k_1, \dots, k_m\}$ de índices de quadros chave.

```

1:  $Q \leftarrow F$ ;
2:  $K \leftarrow \emptyset$ ;
3:  $K \leftarrow K \cup \{c\}$ , tal que  $c \in Q$  é um quadro cuja média dos valores de similaridade  $\sigma$  par
   a par a todos os outros elementos de  $Q$  é máxima;
4:  $Q \leftarrow Q \setminus \{c\}$ ;
5:  $c \leftarrow d$ , tal que  $d \in Q$  é um quadro cuja média dos valores de similaridade  $\sigma$  par a par
   a todos os elementos de  $K$  é mínima;
6:  $\text{sim\_media} \leftarrow (\sum_{k \in K} \sigma(c, k)) / |K|$ ;
7: enquanto  $Q \neq \emptyset$  E  $\text{sim\_media} \leq l$  faça
8:    $K \leftarrow K \cup \{c\}$ ;
9:    $Q \leftarrow Q \setminus \{c\}$ ;
10:   $c \leftarrow d$ , tal que  $d \in Q$  é um quadro cuja média dos valores de similaridade  $\sigma$  par a
    par a todos os elementos de  $K$  é mínima;
11:   $\text{sim\_media} \leftarrow (\sum_{k \in K} \sigma(c, k)) / |K|$ ;
12: fim enquanto
13: retorna  $K$ .
```

O limiar de para o encerramento da seleção de quadros chave foi definido por meio de uma análise empírica e subjetiva. Ele foi variado entre 0 e 1, em passos incrementais de 0,01, revelando que valores superiores a 0,4 produzem uma quantidade excessiva de quadros chave visualmente semelhantes, de acordo com um espectador humano. Valores inferiores a 0,4 produzem apenas um quadro chave por tomada, mesmo quando há quadros consideravelmente distintos visualmente. Considerando que os quadros chave têm como objetivo representar a maior quantidade possível de informação distinta da tomada original, evitando a representação de informação redundante, o valor 0,4 foi adotado como o limiar para a similaridade entre os quadros das tomadas, produzindo uma média de 1,16 quadros chave por tomada.

Obtidos os quadros-chave, a representação do conteúdo visual de cada tomada foi produzida a partir deles, utilizando o extrator de características locais CSIFT (BURGHOUTS; GEUSEBROEK, 2009), uma versão do descritor SIFT apresentado na Seção 2.2, robusto também a variações de cor. O uso de cores em segmentação temporal de vídeo é bastante comum na literatura (YEUNG; YEO; LIU, 1998; FABRO; BÖSZÖRMENYI, 2013; ROTMAN; PORAT; ASHOUR, 2017) pois a composição de cores do conteúdo visual de tomadas com o mesmo conteúdo semântico costuma ser semelhante. O CSIFT produz vetores de características com 384 dimensões que, assim como no SIFT, representam pontos chave. Dado o conjunto de quadros chave de cada tomada do vídeo, a representação do conteúdo visual de cada tomada foi obtida pelo conjunto composto por todos os vetores de características CSIFT extraídos de todos os quadros chave daquela tomada.

Terminada a representação do conteúdo visual, inicia-se da representação do conteúdo aural. O tamanho dos dados aurais de uma tomada é significativamente menor do que o de seus dados visuais, permitindo o processamento de todo o áudio da tomada para extrair sua representação, ao invés de escolher trechos representativos de maneira heurística, como foi feito com o conteúdo visual. Essa abordagem é interessante à aplicação do método proposto já que permite a representação de uma informação proveniente do fluxo aural que não é sincronizada com outra informação proveniente do fluxo visual. Informações assíncronas e correlacionadas podem incluir padrões multimodais passíveis de reconhecimento e representação pelo método de fusão de características proposto. A representação aural de uma tomada foi obtida pela extração de vetores de características MFCC (apresentado na Seção 2.2) de todo o fluxo aural, dividido em quadros de 30ms com uma sobreposição de 10ms. Essa estratégia demonstrou ser adequada para sistemas de reconhecimento de fala (RAO; KOOLAGUDI, 2012). O MFCC produz vetores de 13 dimensões, extraídos de cada quadro de áudio de uma tomada, prática comum na literatura e que produz boas representações (RAO; KOOLAGUDI, 2012; JHUO *et al.*, 2014).

Após a extração das características unimodais visuais e aurais de baixo nível semântico, foi realizado, para cada modalidade, um processamento para síntese de características de médio nível semântico BoF, conforme os passos descritos na Seção 4.2. O agrupamento dos vetores de características locais foi efetuado pelo método *k-means* com o parâmetro *k* (tamanho de dicionário unimodal) tendo sido experimentado com os valores do conjunto {15, 50, 100, 250, 500, 1000, 1500, 2000, 2500, 3000}, para cobrir diferentes granularidades de tamanho de dicionário menores ou iguais a 3000, já que dicionários com mais de 3000 palavras geravam uma alta quantidade de palavras/grupos vazios.

A Etapa 3 do *pipeline* empregado nos experimentos é a fusão das características BoF unimodais aurais e visuais. Esta etapa foi realizada pelo método descrito nas Seções 4.3 e 4.4. O dicionário de palavras fundidas foi obtido também empregando o método *k-means* com o parâmetro *k* (tamanho de dicionário multimodal) tendo sido experimentado

com um conjunto de valores que sera descrito adiante, tomando como entrada características unimodais de mesmo tamanho (*e.g.* dicionário de palavras fundidas de tamanho 500 foi gerado à partir de características visuais e aurais também de tamanho 500). As quatro estratégias de *pooling* descritas na Seção 4.4 (*max* e *average pooling* e suas duas versões ponderadas) foram experimentadas. Dentre elas, o *max pooling* ponderado foi a estratégia que apresentou melhores resultados, tendo sido eleita para produção da representação única do *pipeline* definitivo. O fator de ponderação da etapa do *max pooling* ponderado é igual a 2, ou seja, o valor que representa ocorrências de uma palavra multimodal é dobrado. O raciocínio que motivou a escolha do valor 2 é o de que o acordo entre informações provenientes de duas origens distintas sobre a ocorrência de um padrão deve corresponder a uma maior probabilidade daquele padrão ter sido corretamente identificado. Não foram realizados experimentos para determinar valores ideais de ponderação.

A tarefa da Etapa 4 do *pipeline* de segmentação temporal de vídeo em cenas é o agrupamento de tomadas, efetuado com base nas representações únicas produzidas na etapa de fusão de informação. Mais especificamente, dados um conjunto de tomadas $S = \{s_1, \dots, s_n\}$ e um conjunto de vetores de características fundidas $H = \{h_1, \dots, h_n\}$, onde h_i é a representação de s_i , com $1 \leq i \leq n$, a tarefa consiste em obter um conjunto $B = \{b_1, \dots, b_k\}$ onde cada $b \in B$ é um par de índices de tomadas que definem as tomadas em que começa e termina a sequência de tomadas de cada cena. O algoritmo escolhido para executar esta tarefa foi o STG, descrito na Seção 2.4.1, por ser um método *baseline*, mas com potencial para alcançar resultados comparáveis aos em estado da arte, como visto em Sidiropoulos *et al.* (2011). Além disso, o STG permite o uso de diferentes tipos de entrada como, por exemplo, os vetores de características fundidos produzidos pelo método M4InFus. A distância cosseno foi escolhida para as comparações entre os vetores de características fundidas, dada a sua conhecida aplicabilidade em métodos que empregam características BoF (JIANG; NGO, 2009; BARALDI; GRANA; CUCCHIARA, 2015a), obtendo boa eficácia. O parâmetro de janela temporal w e limiar de agrupamento δ foram determinados em uma análise empírica.

A análise empírica para parametrização do STG foi realizada avaliando a eficácia do *pipeline* proposto na base confiável BBC *Dataset*. Nesta análise também foram escolhidos os tamanhos dos dicionários unimodais/fundido. A eficácia foi medida empregando o conjunto de medidas Precisão/Abrangência, considerando o fato de que ele é o mais rígido dentre os disponíveis, tornando-o ideal para a análise de parametrização. Os valores de *Coverage/Overflow* podem ser indesejavelmente tolerantes para a parametrização conforme os motivos apontados na Seção 2.5. O parâmetro w foi testado com valores do conjunto $\{3, 5, 7, 10, 12, 15, 20, 25, 30, 35\}$ e o parâmetro δ foi testado com os valores entre 0,1 e 0,95, inclusive, em intervalos de 0,05. Valores de w maiores que 35 levam a resultados de subsegmentação extrema. Já o parâmetro δ só admite valores entre 0 e 1, sendo razoável experimentar valores distribuídos de modo uniforme em todo esse intervalo.

Os dicionários de palavras unimodais/fundidas foram testados com os valores do conjunto $\{15, 50, 100, 250, 500, 1000, 1500, 2000, 2500, 3000\}$. A justificativa para a escolha destes valores para os dicionários de palavras fundidas é a mesma da escolha dos dicionários de palavras unimodais. O gráfico ilustrado na Figura 47 apresenta os valores médios de F_{PR} obtidos de acordo com a variação do parâmetro δ do STG.

Figura 46 – Variação da eficácia de acordo com o valor de δ

Os valores mais altos de F_{PR} foram alcançados no intervalo entre 0,35-0,5. O valor 0,35 foi escolhido em detrimento dos outros valores considerando o fato de que o STG tende a subsegmentar (encontrar menos fronteiras do que deveria) os vídeos a medida que δ sobe, já que mais tomadas podem ser agrupadas em um mesmo grupo (Algoritmo 2). O gráfico ilustrado na Figura 47 apresenta os valores médios de F_{PR} obtidos de acordo com a variação do parâmetro w do STG.

Figura 47 – Variação da eficácia de acordo com o valor de w

As médias mais altas de F_{PR} foram obtidas com tamanhos de janela 5 e 7. O valor 7 foi escolhido por permitir uma janela temporal mais flexível. Um maior tamanho de janela temporal também capacita o algoritmo STG a produzir cenas mais longas contendo

tomadas similares que estão temporalmente distantes entre si. O último parâmetro a ser definido foi o de tamanho dos dicionários unimodais e fundido. A Figura 48 apresenta as médias dos valores de FPR de acordo com a variação dos tamanhos dos dicionários.

Figura 48 – Variação da eficácia de acordo com os tamanhos dos dicionários.

Os melhores resultados foram obtidos com dicionários aural, visual e fundido de tamanho 100. Este valor foi adotado para a configuração final do *pipeline* de segmentação temporal de vídeo em cenas empregado neste experimento.

5.1.2 Resultados

A Tabela 8 apresenta os valores de Precisão (P), Abrangência (R), FPR , $Coverage$ (C), $Overflow$ (O), F_{CO} , $New Overflow$ (NO) e F_{CNO} obtidos para cada episódio da base confiável BBC *Dataset* pelo *pipeline* de segmentação temporal de vídeo em cenas proposto.

Tabela 8 – Valores obtidos por medidas de eficácia na base confiável BBC *Dataset*.

Episódio	P	R	FPR	C	O	F _{CO}	NO	F _{CNO}
From Pole to Pole	0,57	0,65	0,61	0,83	0,53	0,60	0,44	0,67
Mountains	0,43	0,87	0,57	0,62	0,33	0,64	0,25	0,68
Fresh Water	0,35	0,78	0,48	0,58	0,27	0,65	0,27	0,65
Caves	0,49	0,86	0,62	0,67	0,32	0,68	0,28	0,70
Deserts	0,40	0,85	0,54	0,66	0,26	0,70	0,26	0,70
Ice Worlds	0,49	0,74	0,59	0,74	0,40	0,66	0,38	0,67
Great Plains	0,63	0,77	0,69	0,76	0,42	0,66	0,38	0,69
Jungles	0,63	0,55	0,59	0,75	0,57	0,55	0,48	0,61
Shallow Seas	0,58	0,69	0,63	0,75	0,49	0,61	0,40	0,67
Seasonal Forests	0,49	0,66	0,56	0,72	0,47	0,61	0,38	0,66
Ocean Deep	0,48	0,83	0,61	0,67	0,22	0,72	0,22	0,72
Média	0,50	0,75	0,59	0,70	0,39	0,64	0,34	0,67

Os valores de Abrangência obtidos indicam que houve sobresegmentação, ou seja, a ferramenta de segmentação tende a encontrar mais fronteiras entre cenas do que deveria. Este é um problema comum em segmentação temporal de vídeo em cenas e algumas técnicas existentes efetuam pós processamento para reduzir seu impacto (ZHAI; SHAH,

2006; CHENG; LU, 2008; FABRO; BÖSZÖRMENYI, 2013). Nenhum tipo de pós processamento foi efetuado neste experimento. A sobresegmentação que ocorreu é aceitável já que, em contrapartida, o *pipeline* proposto obteve valores altos de *coverage*. Isto significa que apesar de ter havido sobresegmentação, as frações das cenas reais cobertas pela predição foram altas. Considerando que a base confiável BBC *Dataset* apresenta uma média de 61 cenas por vídeo, o *pipeline* de segmentação proposto identificou 54% mais cenas (em média 94 cenas). Ocorreu um único caso de subsegmentação, na instância Jungles, evidenciado pelo baixo valor de Abrangência obtido. Essa instância pode ser considerada uma das mais difíceis da base confiável, para técnicas de segmentação temporal de vídeo em cenas que não utilizem informação textual. Isso acontece pois suas tomadas apresentam informações aurais e visuais bastante repetitivas ao longo das tomadas, induzindo as técnicas que não utilizam informação textual à subsegmentação.

A Tabela 9 apresenta uma comparação entre os valores das medidas F_{PR} , F_{CO} e F_{CNO} obtidos pelo *pipeline* proposto empregando o método M4InFus e os valores das mesmas medidas obtidos pelo *pipeline* proposto sem empregá-lo (coluna **Sem fusão**). Essa comparação tem como objetivo analisar os possíveis ganhos em eficácia propiciados pelo método M4InFus. O *pipeline* de segmentação temporal de vídeo em cenas foi executado sem a etapa de fusão, usando apenas características BoF unimodais visuais e aurais, separadamente. Os valores de medidas de eficácia das colunas **Sem fusão** contém o valor máximo obtido para cada instância entre as respectivas execuções individuais do *pipeline* com características CSIFT e MFCC.

Tabela 9 – Comparação entre valores de medidas de eficácia obtidos pelo *pipeline* proposto com e sem o emprego do método M4InFus.

Episódio	F _{PR}		F _{CO}		F _{CNO}	
	Sem fusão	M4InFus	Sem fusão	M4InFus	Sem fusão	M4InFus
From Pole to Pole	0,56	0,61	0,62	0,60	0,64	0,67
Mountains	0,38	0,57	1,00*	0,64	0,53	0,68
Fresh Water	0,38	0,48	1,00*	0,65	0,54	0,65
Caves	0,58	0,62	0,67	0,68	0,68	0,70
Deserts	0,54	0,54	0,65	0,70	0,66	0,70
Ice Worlds	0,53	0,59	0,70	0,66	0,70	0,67
Great Plains	0,54	0,69	0,64	0,66	0,65	0,69
Jungles	0,54	0,59	0,60	0,55	0,61	0,61
Shallow Seas	0,60	0,63	0,69	0,61	0,69	0,67
Seasonal Forests	0,48	0,56	0,49	0,61	0,54	0,66
Ocean Deep	0,50	0,61	0,68	0,72	0,67	0,72
Média	0,51	0,59	—	0,64	0,63	0,67

Os resultados da Tabela 9 mostram que o método M4InFus melhora consideravelmente os resultados obtidos. Todos os resultados obtidos em F_{PR} pelo *pipeline* com o

emprego do método M4InFus foram, com exceção da instância *Deserts* onde houve empate, superiores aos obtidos sem o emprego do método. O *pipeline* sem emprego de fusão produziu predições de fronteiras entre tomadas com subsegmentação extrema nas instâncias *Mountains* e *Fresh Water*. Tal fato pode ser observado pelos valores máximos produzidos para F_{CO} , ao mesmo tempo em que as duas instâncias obtiveram valores baixíssimos de F_{PR} (os dois menores). Os valores de F_{CO} devem ser considerados inválidos devido ao motivo descrito na Seção 2.5.1 e não devem ser considerados em qualquer análise. Os valores inválidos foram destacados com o símbolo asterisco na Tabela 9. Os valores de F_{CNO} obtidos pela versão do *pipeline* combinada com o método M4InFus foram melhores do que os obtidos pela versão sem fusão em 0,04 na média e em 8 de 11 instâncias, tendo empatado em uma das três restantes.

A Tabela 10 apresenta uma comparação dos resultados obtidos pelo *pipeline* proposto (STG_{EF}) com o método fusão tardia baseado em características aurais e visuais de Sidiropoulos *et al.* (2011) e o método baseado em aprendizado profundo com características visuais e textuais de Baraldi, Grana e Cucchiara (2015a).

Tabela 10 – Comparação entre valores de F_{CO} obtidos pelo *pipeline* proposto com os de dois outros métodos de segmentação na base confiável BBC *Dataset*.

F_{co}			
Episódio	Sidiropoulos	Baraldi	Pipeline Proposto
From Pole to Pole	0,47	0,56	0,60
Mountains	0,55	0,63	0,64
Fresh Water	0,56	0,66	0,65
Caves	0,46	0,61	0,68
Deserts	0,42	0,55	0,70
Ice Worlds	0,44	0,64	0,66
Great Plains	0,65	0,59	0,66
Jungles	0,63	0,64	0,55
Shallow Seas	0,61	0,64	0,61
Seasonal Forests	0,58	0,64	0,61
Ocean Deep	0,52	0,64	0,72
Média	0,54	0,62	0,64

O *pipeline* de segmentação temporal de vídeo em cenas superou, em média, as outras duas técnicas na medida F_{CO} . A técnica fusão tardia de Sidiropoulos *et al.*, apesar de empregar características de alto nível semântico, semanticamente mais ricas e prontas para uso, e de empregar uma estratégia força bruta para determinação de parâmetros em tempo de execução, foi superada, em todas as instâncias, pelas outras duas abordagens fusão prévia. A baixa eficácia desta técnica, em relação às outras duas, se deve, principalmente ao fato da fusão tardia ignorar a correlação entre diferentes modalidades antes das etapas de segmentação. A técnica de rede siamesa profunda de Baraldi *et al.* baseia-se no alto poder de extração de informação representativa dessas redes neurais siamesas. Esta

técnica também emprega características textuais, sabidamente mais ricas, e de fácil processamento, quando comparadas às características de baixo nível semântico. Os resultados inferiores obtidos pela técnica de rede siamesa profunda são um indício de que ela sofre com a perda de referência para o significado original das características unimodais, que ocorre no processo convolucional presente em redes neurais.

Considerando que o *pipeline* de segmentação temporal de vídeo em cenas empregou apenas características de baixo nível semântico CSIFT e MFCC, em conjunto com o algoritmo *baseline* seminal de agrupamento de tomadas STG, os resultados superiores em relação aos das técnicas plenamente desenvolvidas, se devem exclusivamente à capacidade de computação de representações mais expressivas da etapa de fusão de características.

5.2 Experimento 2: segmentação temporal usando *deep features*

No Experimento 2, o *pipeline* de segmentação de vídeo em cenas do Experimento 1 foi modificado, substituindo-se os vetores de características gerados por extratores de características *baseline* por vetores de características gerados por extratores de características baseados em redes neurais profundas. Tais características são conhecidas como *deep features*. A técnica resultante foi avaliada e seus resultados foram comparados aos resultados obtidos por sua versão anterior, assim como aos de uma técnica multimodal baseada em redes neurais profundas. Com isso, analisa-se a aplicabilidade do método M4InFus quando combinado a extratores de características em estado da arte e, ainda, seu potencial atual nesta tarefa.

A metodologia deste experimento é semelhante à descrita na Seção 5.1. A principal diferença consiste no fato de que foram empregados vetores de características produzidos por redes neurais profundas no processo de fusão de informação. A produção destes vetores de características por redes neurais profundas foi efetuada por uma versão do método de segmentação temporal de vídeo em cenas de Trojahn (2019), adaptado para esse novo propósito. A técnica de Trojahn determina a extração de vetores de características de baixo nível semântico diretamente dos dados brutos, que são, em seguida, fornecidos como entrada a uma rede neural convolucional (*Convolutional Neural Network* (CNN)) (WIATOWSKI; BÖLCSKEI, 2018) por modalidade. As redes neurais convolucionais são conhecidas por serem capazes de gerar representações compactas e ao mesmo tempo semanticamente ricas de um conteúdo. Na técnica de Trojahn, a saída de cada uma destas redes serve de entrada para uma respectiva rede neural recorrente (*Recurrent Neural Network* (RNN)) (RUMELHART; HINTON; WILLIAMS, 1988; ELMAN, 1990), destinada, no cenário daquele trabalho, à análise das relações entre sequências de tomadas com o objetivo de determinar a probabilidade de cada tomada ser um fim de cena. Neste

trabalho, modificou-se a técnica de Trojahn para possibilitar a extração das representações das tomadas diretamente das saídas das redes neurais convolucionais, ignorando as saídas das redes neurais recorrentes.

Outras diferenças do experimento anterior são, a adoção de parametrização individual do *pipeline* por vídeo, o acréscimo de uma segunda base confiável aos experimentos e a comparação com duas outras técnicas de segmentação temporal de vídeo em cenas. As modificações no *pipeline* descrito na Seção 5.1 tem o intuito de alcançar a máxima eficácia do método de fusão de informação na tarefa de segmentação temporal de vídeo em cenas, por meio de sua combinação com os mais avançados métodos/técnicas disponíveis. Essas modificações são descritas na Seção 5.2.1. A inclusão de uma segunda base confiável aos experimentos permitiu a comparação com técnicas de segmentação temporal de vídeo em cenas ainda mais recentes. Os resultados desse experimento são discutidos na Seção 5.2.2.

5.2.1 Metodologia

Nesse experimento, além da *BBC Dataset*, também foi empregada a base confiável *OVSD Dataset*. A *OVSD Dataset* não contém arquivos com as posições de fronteiras entre tomadas. Os autores de Rotman *et al.* (2018), entretanto, gentilmente forneceram os arquivos de posições de fronteiras entre tomadas que foram utilizados em seus experimentos, permitindo uma comparação justa. Os arquivos utilizados e providos por Rotman *et al.* foram computados automaticamente pela técnica de Baraldi, Grana e Cucchiara (2015c). Além destas bases confiáveis, foram empregadas as mesmas medidas de avaliação empregadas na Seção 5.1, e ainda a medida F_{CNO} , descrita na Seção 2.5.1, que foi necessária à análise de um aspecto específico desta avaliação. O *pipeline* modificado de segmentação temporal de vídeo em cenas é descrito a seguir.

Versão modificada do pipeline de segmentação temporal de vídeo em cenas

O método de Trojahn determina a representação do conteúdo visual de cada tomada por cinco quadros chave, eleitos pela ferramenta VKFrameS⁶ (TROJAHN; GOU-LARTE, 2013), com parâmetros definidos conforme a recomendação dos autores. Os quadros chave de cada tomada foram fornecidos como entradas, um a um, à rede neural convolucional VGGNet⁷ (SIMONYAN; ZISSERMAN, 2015) de 16 camadas, previamente treinada sobre a base de dados ImageNet (DENG *et al.*, 2009). O treinamento de todas as amostras da base (excluindo o vídeo a ser avaliado) é chamado época. A VGGNet, originalmente destinada à classificação de imagens, teve, neste trabalho, a função de extrator de características. O uso de redes neurais convolucionais para extração de vetores de características de imagens, prática comum na literatura (SIMO-SERRA *et al.*, 2015),

⁶ Disponível em <<https://github.com/Trojahn/VKFrameS>>.

⁷ Disponível em <http://www.robots.ox.ac.uk/~vgg/research/very_deep/>.

se dá pela captura dos valores produzidos pelos neurônios de uma das camadas internas da rede. Estes valores, concatenados, compõem o vetor de características que representa uma imagem fornecida originalmente como entrada à rede. No *pipeline* modificado, os cinco quadros chave de cada tomada foram redimensionados para a resolução espacial 244×244 , atendendo ao tamanho de entrada definido pelos autores, tendo sido entregues, em seguida, à VGGNet. Cada quadro chave produz um vetor de características de 4096 dimensões, gerando um total de 5×4096 valores que representam cada tomada. Esses vetores de características são chamados ConvFeats ([TROJAHN, 2019](#)).

Além das características provenientes do processamento por redes neurais convolucionais, os cinco quadros chave de cada tomada também serviram de fonte para a extração de vetores de características CSIFT. A extração padrão de características CSIFT gera um número variável e, em geral, grande de vetores de características. Para padronizar a representação de cada tomada tal que ela seja compatível com a entrada de uma rede neural de tamanho fixo, foram escolhidos 100 vetores de características, usando uma estratégia semelhante à da construção de dicionários de palavras do método BoF. A estratégia consiste em executar, para cada conjunto de vetores de características CSIFT de uma tomada, um algoritmo de agrupamento *k-means*, com $k = 100$. Após a definição dos 100 grupos, são eleitos, para cada um deles, um medóide como seu representante e consequentemente como componente do conjunto de 100 vetores de características CSIFT que representarão cada tomada na etapa de refinamento semântico por meio de redes neurais profundas.

O conteúdo aural de cada tomada foi representado, inicialmente, por vetores de características MFCC, usando exatamente a mesma configuração do experimento descrito na Seção 5.1. A extração de vetores de características MFCC de uma tomada, assim como a extração de vetores de características CSIFT, também produz quantidades variáveis de vetores de características e, por isso, adotou-se o mesmo processo de escolha baseado em medóides de grupos utilizado na seleção de vetores de características CSIFT. Foram escolhidos 500 vetores de características MFCC de cada tomada, gerando um total de 500×13 valores que representam cada tomada.

A etapa que segue a extração inicial de vetores de características, consiste em um refinamento semântico viabilizado pela arquitetura de redes neurais profundas também proposta em [Trojahn \(2019\)](#). Nesse método fusão tardia, há uma dupla de redes neurais, uma CNN e uma RNN ligadas em série, dedicadas a receber uma sequência de representações de tomadas por modalidade, e produzir uma sequência de valores, um por representação de tomada. Os valores da sequência produzida, que variam entre zero e um, inclusive, indicam a probabilidade daquela tomada ser uma tomada final de uma cena. O valor zero denota que a probabilidade da tomada correspondente ser final de cena é nula e o valor um denota que a probabilidade da tomada correspondente ser final de cena é de cem por cento. Há ainda uma RNN destinada a realizar a fusão das previsões produzidas

por cada dupla de redes neurais unimodais. Esta terceira RNN não foi necessária a este experimento e foi desconsiderada.

Seja denominado vídeo alvo, um vídeo pertencente ao conjunto de vídeos da base confiável para o qual se deseja produzir representações de suas tomadas em um determinado momento. As redes unimodais individuais foram treinadas usando a estratégia *leave one out*, onde as representações das tomadas de todos os vídeos, com exceção daquelas do vídeo alvo, foram fornecidas às redes. O treinamento de todos os vídeos da base, excluindo-se o vídeo alvo, é chamada época. A cada época, usando a medida entropia binária cruzada, foi calculado o quão distante da resposta correta, era a predição produzida pela rede. Este treinamento foi realizado até que se atingisse um valor médio de entropia binária cruzada menor que o limiar de 0,01 ou então que fosse alcançado um número máximo de 850 épocas. Foram extraídos vetores de características a cada 50 épocas, compostos pelas saídas dos neurônios da camada imediatamente anterior à primeira camada da RNN de cada rede unimodal. Os vetores de características provenientes de ConvFeats, CSIFT e MFCC produzidos por esta arquitetura, tem 256, 128 e 128 dimensões, respectivamente. Cada um destes vetores de características representa uma tomada e, com sua obtenção, é encerrada a etapa de geração de representações por métodos em estado da arte.

As representações unimodais produzidas pelas redes neurais da arquitetura de Trojahn são concisas e ricas em semântica latente. Além disso, apesar de não haver referência às informações originais e também de não haver interpretação dos valores contidos nos vetores de características devido ao processamento por rede neural profunda, espera-se que, por terem sido computados por redes semelhantes, os valores sejam comparáveis entre si, permitindo a aplicação direta dos processos de detecção de co-ocorrência e posteriormente da síntese em representação única descritos nas Seções 4.3 e 4.4, respectivamente. O tamanho do dicionário de padrões fundidos foi definido em 256 palavras, reduzindo pela metade, a dimensionalidade do conjunto da combinação ingênua dos três tipos de vetores de características (256 dos ConvFeats, 128 dos CSIFT e 128 dos MFCC).

A etapa seguinte consiste na aplicação do algoritmo STG, assim como no experimento anteriormente descrito na Seção 5.1, para obter as predições de posições de tomadas posicionadas em fronteiras entre cenas. A escolha dos parâmetros de δ e janela temporal $\mathcal{W}$ do STG para a base confiável BBC *Dataset* foi realizada de maneira semelhante à do experimento da Seção 5.1, para permitir comparação direta com os resultados daquele experimento. Os valores que alcançaram melhor eficácia média em toda a base confiável foram $\delta = 0,10$ e $\mathcal{W} = 5$. Em relação à base confiável OVSD *Dataset*, a parametrização do algoritmo STG foi um pouco mais flexível, tendo sido determinada individualmente por instância, para que se alcançasse a maior eficácia possível. Os conjuntos de valores δ e $\mathcal{W}$ experimentados para cada instância foi o mesmo do experimento descrito na Se-

ção 5.1. A Tabela 11 apresenta os parâmetros para o algoritmo STG, por instância, que apresentaram a melhor eficácia nos experimentos efetuados.

Tabela 11 – Parametrização do algoritmo STG, por vídeo, que gerou os melhores resultados na base confiável OVSD *Dataset*.

Vídeo	δ	$\mathcal{W}$	Vídeo	δ	$\mathcal{W}$
1000 Days	0,20	12	Meridian	0,10	7
Big Buck Bunny	0,35	5	Oceania	0,10	3
Boy Who Never Slept	0,10	3	Pentagon	0,15	7
CH7	0,35	10	Route 66	0,10	12
Cosmos Laundromat	0,90	7	Seven Dead Men	0,20	5
Elephants Dream	0,30	3	Sintel	0,90	12
Fires Beneath Water	0,10	3	Sita Sings The Blues	0,10	3
Honey	0,10	3	Star Wreck	0,25	7
Jathias Wager	0,15	7	Tears of Steel	0,15	7
La Chute D Une Plume	0,10	10	Valkaama	0,15	10
Lord Meia	0,10	7			

5.2.2 Resultados

A Tabela 12 apresenta os valores de Precisão (P), Abrangência (R), F_{PR} , $Coverage$ (C), $Overflow$ (O) e F_{CO} obtidos para cada episódio da base confiável BBC *Dataset* pelo *pipeline* modificado de segmentação temporal de vídeo em cenas.

Tabela 12 – Valores de Precisão, Abrangência, F_{PR} , $Coverage$, $Overflow$ e F_{CO} obtidos na base confiável BBC *Dataset*, usando características extraídas por redes neurais profundas.

Episódio	P	R	F _{PR}	C	O	F _{CO}
From Pole to Pole	0,66	0,76	0,70	0,74	0,51	0,59
Mountains	0,49	0,85	0,62	0,63	0,34	0,64
Fresh Water	0,60	0,89	0,71	0,66	0,34	0,66
Caves	0,64	0,86	0,74	0,69	0,39	0,64
Deserts	0,62	0,78	0,69	0,66	0,35	0,66
Ice Worlds	0,69	0,82	0,75	0,68	0,37	0,66
Great Plains	0,65	0,80	0,72	0,70	0,36	0,67
Jungles	0,53	0,78	0,63	0,62	0,29	0,66
Shallow Seas	0,72	0,72	0,72	0,75	0,45	0,64
Seasonal Forests	0,56	0,85	0,67	0,62	0,37	0,63
Ocean Deep	0,67	0,79	0,72	0,67	0,36	0,65
Média	0,62	0,81	0,70	0,67	0,38	0,65

Assim como no experimento descrito na Seção 5.1, os altos valores de Abrangência obtidos, em combinação com valores mais baixos de Precisão, indicam que houve sobre-segmentação. O *pipeline* de segmentação temporal de vídeo em cenas detectou 27% mais cenas (em média 78 cenas) do que o esperado. Este valor, entretanto, foi 27% menor do

que o do experimento descrito na Seção 5.1, onde foram detectadas 54% mais cenas. Ou seja, reduziu-se o problema da sobresegmentação pela fusão de vetores de características provenientes de redes neurais profundas. Os valores de Precisão foram consideravelmente melhores neste experimento, indicando que o emprego de vetores de características fundidos reduziram ainda mais a quantidade de erros do algoritmo STG ao realizar o agrupamento de tomadas. O *pipeline* modificado obteve melhores resultados de abrangência na instância *Jungles*, indicando que não houve problema de subsegmentação, diferentemente do que ocorreu no experimento descrito na Seção 5.1.

A Tabela 13 apresenta uma comparação dos valores de F_{PR} obtidos pelas duas diferentes versões do *pipeline* de segmentação temporal de vídeo em cenas. Os valores de F_{PR} da técnica de Baraldi, Grana e Cucchiara (2015a) não foram divulgadas pelos autores.

Tabela 13 – Comparação dos valores de F_{PR} obtidos na base confiável BBC *Dataset* pelas duas diferentes versões do *pipeline* de segmentação temporal de vídeo em cenas.

F_{PR}		
Episódio	Baseline	Profundo
From Pole to Pole	0,61	0,70
Mountains	0,57	0,62
Fresh Water	0,48	0,71
Caves	0,62	0,74
Deserts	0,54	0,69
Ice Worlds	0,59	0,75
Great Plains	0,69	0,72
Jungles	0,59	0,63
Shallow Seas	0,63	0,72
Seasonal Forests	0,56	0,67
Ocean Deep	0,61	0,72
Média	0,59	0,70

As colunas *Baseline* e *Profundo* contém os valores de F_{PR} obtidos pelas versões do *pipeline* de segmentação temporal de vídeo em cenas que empregam características geradas por algoritmos *baseline* e características geradas por redes neurais profundas, respectivamente. A versão modificada do *pipeline* de segmentação temporal de vídeo em cenas obteve valores de F_{PR} superiores aos da versão original em todas as instâncias. Este comportamento indica que o método M4InFus é flexível quanto ao tipo de característica de entrada. A Tabela 14 apresenta uma comparação dos valores de F_{CO} obtidos pelas duas versões (colunas *Baseline* e *Profundo*) do *pipeline* de segmentação temporal de vídeo em cenas proposto e também pela técnica de Baraldi, Grana e Cucchiara (2015a) (coluna Baraldi).

A versão modificada do *pipeline* de segmentação temporal de vídeo em cenas proposto obteve valores de F_{CO} maiores do que os obtidos pela técnica de Baraldi, Grana e Cucchiara (2015a) em todas as instâncias, com exceção das instâncias dos vídeos *Fresh*

Tabela 14 – Valores de F_{CO} obtidos na base confiável BBC *Dataset*, usando características extraídas por redes neurais profundas.

Episódio	Fco		
	Baraldi	Baseline	Profundo
From Pole to Pole	0,56	0,60	0,59
Mountains	0,63	0,64	0,64
Fresh Water	0,66	0,65	0,66
Caves	0,61	0,68	0,64
Deserts	0,55	0,70	0,66
Ice Worlds	0,64	0,66	0,66
Great Plains	0,59	0,66	0,67
Jungles	0,64	0,55	0,66
Shallow Seas	0,64	0,61	0,64
Seasonal Forests	0,64	0,61	0,63
Ocean Deep	0,64	0,72	0,65
Média	0,62	0,64	0,65

Water, onde obteve o mesmo valor e *Seasonal Forests*, onde obteve um valor 0,01 menor. Considerando que a técnica de Baraldi, Grana e Cucchiara (2015a) também emprega vetores de características produzidos por redes neurais profundas, deduz-se que a melhor eficácia alcançada pelo *pipeline* modificado é devida, principalmente, ao método de fusão de informação proposto. O método M4InFus expôs a semântica latente presente nos vetores de características produzidos pelas redes neurais profundas da técnica de Trojahn (2019), permitindo seu aproveitamento pelo algoritmo STG no processo de agrupamento de tomadas. Tal afirmação é reforçada pelo fato de que a única diferença entre as versões original e modificada do *pipeline* de segmentação temporal de vídeo em cenas se dá pelas características de entrada. Dessa forma, o ganho em eficácia observado da versão modificada do *pipeline* em relação à versão original, só pode se justificar pela computação de uma melhor representação dos dados de entrada.

O *pipeline* de segmentação temporal de vídeo em cenas modificado foi experimentado, adicionalmente, na base confiável OVSD *Dataset*. Os resultados obtidos são apresentados na Tabela 15.

O *pipeline* modificado apresentou boa eficácia média nesta base confiável, com exceção de algumas instâncias. Uma delas é o vídeo *Meridian*, onde o valor de F_{CO} obtido é baixo. Tal vídeo apresentou alto valor de *overflow*, que se deve ao fato de possuir curta duração (com poucas tomadas/cenas) o que faz com que a medida penalize excessivamente a anotação obtida já que, proporcionalmente, mesmo uma única tomada que ultrapasse a fronteira original da cena, corresponde a uma grande fração desta cena. Além do vídeo *Meridian*, houve baixa eficácia nos vídeos *Boy Who Never Slept*, *Elephants Dream*, *Honey*, *Oceania*, *Route 66* e *Sita Sings The Blues*, onde os valores de Precisão e Abrangência obtidos são baixos. Todos os baixos valores de Precisão e Abrangência obtidos em

Tabela 15 – Valores de medidas de eficávis obtidos na base confiável OVSD *Dataset*, usando características extraídas por redes neurais profundas.

Vídeo	P	R	F _{PR}	C	O	F _{CO}	NO	F _{CNO}
1000 Days	0,57	0,73	0,64	0,82	0,40	0,69	0,28	0,76
Big Buck Bunny	0,57	0,87	0,68	0,59	0,27	0,66	0,29	0,65
Boy Who Never Slept	0,43	0,17	0,24	0,92	19,79*	1,92*	0,89	0,20
CH7	0,45	0,64	0,53	0,74	0,42	0,66	0,36	0,69
Cosmos Laundromat	0,60	0,86	0,71	0,60	0,38	0,60	0,24	0,67
Elephants Dream	1,00	0,22	0,36	1,00	6,01*	2,49*	0,82	0,31
Fires Beneath Water	0,64	0,90	0,75	0,57	0,40	0,58	0,29	0,63
Honey	0,67	0,10	0,17	1,00	16,85*	2,12*	0,88	0,21
Jathias Wager	0,56	0,60	0,58	0,88	0,05	0,89	0,66	0,49
La Chute D Une Plume	0,71	1,00	0,83	0,89	0,07	0,94	0,06	0,91
Lord Meia	0,50	0,70	0,58	0,62	0,47	0,57	0,33	0,64
Meridian	0,75	0,67	0,71	0,69	0,73	0,39	0,34	0,68
Oceania	0,40	0,13	0,20	0,90	20,76*	1,88*	0,90	0,18
Pentagon	0,52	0,74	0,61	0,58	0,39	0,60	0,32	0,63
Route 66	0,67	0,04	0,07	1,00	60,14*	2,03*	0,95	0,10
Seven Dead Men	0,87	0,76	0,81	0,75	0,57	0,55	0,42	0,66
Sintel	0,67	0,75	0,71	0,86	0,26	0,79	0,28	0,78
Sita Sings The Blues	0,71	0,10	0,17	0,99	37,19*	2,03*	0,95	0,09
Star Wreck	0,26	0,65	0,37	0,48	0,26	0,59	0,25	0,58
Tears of Steel	0,73	0,73	0,73	0,72	0,44	0,64	0,30	0,71
Valkaama	0,53	0,61	0,57	0,71	0,49	0,60	0,38	0,66
Média	0,61	0,57	0,52	0,78	—	—	0,49	0,53

tais instâncias indicadas correspondem a situações extremas de subsegmentação como, por exemplo, no vídeo *Route 66*, onde o *pipeline* modificado segmentou o vídeo de 1084 tomadas em apenas três cenas quando, de acordo com a anotação de posições de fronteiras entre tomadas da base confiável, haviam 55 diferentes cenas. A ocorrência de subsegmentação extrema faz com que a medida de avaliação *overflow* e, conseqüentemente o F_{CO} , apresentem valores inválidos, conforme discutido na Seção 2.5.1, impossibilitando a inclusão destes resultados na avaliação. Os valores inválidos foram marcados com o símbolo asterisco na Tabela 15.

O principal motivo do mal funcionamento do *pipeline* modificado nestas instâncias é a unicidade do conteúdo visual destes vídeos em relação ao restante da base confiável. Como exemplo, pode-se destacar a instância “*Sita Sings The Blues*”, único vídeo de animação 2D de toda a base confiável. A arquitetura de Trojahn (2019) determina uma etapa de treinamento por tipo de vetor de características unimodal, o que faz com que seja necessária uma quantidade mínima de exemplos semelhantes para que a rede seja capaz de aprender qualquer padrão que ocorra em determinado vídeo. Considerando que o próprio vídeo não foi empregado na etapa de treinamento, é esperado que as redes neurais profundas unimodais da arquitetura de Trojahn nem sempre sejam capazes de

gerar representações adequadas nesta base confiável. A técnica de Rotman *et al.* (2018), que será base de comparação com o *pipeline* modificado, emprega uma rede pré-treinada em uma base confiável extensa e outra rede cujos autores não descreveram a etapa de treinamento, que pode ter incluído os próprios vídeos sendo segmentados. Há mais detalhes sobre o método de Rotman *et al.* no Capítulo 3.

Como não foi possível avaliar a *overflow* e, conseqüentemente, a F_{CO} obtida pelo *pipeline* modificado nesta base, foram calculadas, adicionalmente, as medidas de avaliação *new overflow* e F_{CNO} para possibilitar uma melhor avaliação. Estas medidas avaliam de maneira correta as instâncias onde o *pipeline* modificado funcionou mal. As instâncias onde houve subsegmentação extrema produziram valores altos de NO (média de 0,89), e baixos de F_{CNO} (média de 0,18) como esperado, retratando a má performance. Os valores obtidos nestas duas medidas, entretanto, não são adotadas por autores dos trabalhos recentes de segmentação temporal de vídeo em cenas. Estes autores divulgam apenas os valores de F_{CO} de suas respectivas técnicas. Por este motivo, foram desconsideradas as instâncias onde o *pipeline* modificado apresentou valores inválidos de F_{CO} na comparação com técnicas de segmentação de vídeo em cenas em estado da arte. A Tabela 16 apresenta uma comparação entre os valores de F_{CO} obtidos pelo *pipeline* de segmentação temporal de vídeo em cenas modificado (coluna **Pipeline Proposto_{Deep}**) e os obtidos por duas técnicas de segmentação temporal de vídeo em cenas multimodais. Estas técnicas são a abordagem multimodal de Rotman, Porat e Ashour (2017) (coluna **Rotman_{MM}**) e sua extensão que emprega vetores de características gerados por redes neurais profundas (coluna **Rotman_{DeepMM}**) descrita em Rotman *et al.* (2018).

O *pipeline* modificado obteve um valor médio de F_{CO} igual ao da versão modificada da técnica de Rotman *et al.* e obteve o maior valor de F_{CO} em 6 dos 15 vídeos avaliados, contra 5 vídeos onde a técnica modificada de Rotman *et al.* obteve maiores valores. A versão original da técnica de Rotman *et al.* obteve o menor valor médio de F_{CO} e apenas 4 maiores valores de F_{CO} , apenas 4. A versão modificada da técnica de Rotman *et al.* possui um algoritmo de agrupamento de tomadas plenamente desenvolvido, superior ao empregado no *pipeline* modificado. Além disso, os vetores de características aurais empregados na técnica de Rotman *et al.* são ricos em semântica latente, dado que os próprios autores relatam uma diminuição de 8% no valor de F_{CO} médio quando o experimento foi realizado sem o emprego da modalidade aural. Os autores relatam em Rotman *et al.* (2018) apenas a etapa de treinamento da computação de vetores de características baseados em conteúdo visual, não descrevendo como foi realizado o treinamento da rede neural que produziu os vetores de características baseados na modalidade aural. É possível que os próprios vídeos da base confiável, incluindo o próprio vídeo que se desejava representar, tenham sido empregados neste treinamento.

Estes resultados mostram que o método de fusão de informação multimodal pro-

Tabela 16 – Comparação entre valores de F_{CO} obtidos pelo *pipeline* proposto ao empregar características extraídas por redes neurais profundas e dois métodos de segmentação em estado da arte, na base confiável OVSD *Dataset*.

Vídeo	F_{CO}		
	Rotman _{MM}	Rotman _{DeepMM}	Pipeline Proposto _{Deep}
1000 Days	0,38	0,60	0,69
Big Buck Bunny	0,83	0,63	0,66
CH7	0,63	0,60	0,66
Cosmos Laundromat	0,53	0,51	0,60
Fires Beneath Water	0,57	0,59	0,58
Jathia's Wager	0,75	0,63	0,89
La Chute d'une Plume	0,53	0,72	0,94
Lord Meia	0,69	0,64	0,57
Meridian	0,63	0,79	0,39
Pentagon	0,73	0,65	0,60
Seven Dead Men	0,68	0,70	0,55
Sintel	0,46	0,58	0,79
Star Wreck	0,55	0,61	0,59
Tears of Steel	0,50	0,73	0,64
Valkaama	0,63	0,70	0,60
Média	0,61	0,65	0,65

posto é capaz de capturar e representar adequadamente a semântica latente contida em vetores de características extraídos por redes neurais profundas, já que o *pipeline* modificado apresentou resultados equiparáveis aos da técnica em estado da arte de [Rotman et al. \(2018\)](#).

5.3 Experimento 3: classificação de vídeos

O Experimento 3 tem como objetivo a verificação da aplicabilidade do método M4InFus em uma segunda tarefa de análise de vídeo, a classificação de vídeos. Tal verificação foi planejada para se dar pela substituição da etapa de fusão de características de uma técnica de classificação de vídeos multimodal atual pelo método M4InFus. Uma melhora nos resultados obtidos pela técnica modificada em relação aos originais, pode indicar a aplicabilidade do método M4InFus em um segundo domínio de aplicação.

O problema da classificação de vídeos, assim como o da segmentação temporal de vídeo em cenas, também é considerado em aberto pela literatura ([MA et al., 2018](#)). A tarefa de classificar um vídeo depende, em grande parte, da capacidade do sistema de produzir uma representação expressiva daquele vídeo. Neste experimento, foi replicado um *pipeline* de classificação de vídeo com etapa de fusão de informação multimodal por grafo bipartido disponível na literatura. O *pipeline* replicado foi modificado, substituindo-se a etapa de fusão existente, pelo método de fusão de informação proposto, descrito

no Capítulo 4. O *pipeline* escolhido foi descrito em Jhuo *et al.* (2014). Sua escolha para este experimento é justificada pelo fato de que o trabalho de Jhuo *et al.* é considerado estado da arte em classificação de vídeos na base confiável CCV *Dataset* (SOLTANIAN; GHAEMMAGHAMI, 2019). Mais ainda, dentre os trabalhos em estado da arte nessa base confiável, é o único que efetua fusão prévia sem o emprego de aprendizado profundo, assim como o método M4InFus. A disponibilidade, tanto da base confiável, quanto dos vetores de características empregados no experimento de Jhuo *et al.* também motivou sua escolha. A Seção 5.3.1 descreve os detalhes do *pipeline* replicado e a Seção 5.3.2 apresenta e discute os resultados obtidos por ele.

5.3.1 Metodologia

O *pipeline* de Jhuo *et al.*, ilustrado na Figura 49, foi detalhado no Capítulo 3, onde há uma explicação sobre como foi realizada a extração de características MFCC, SIFT e STIP, seu posterior refinamento em histogramas BoF, bem como a fusão de informação baseada em grafo bi-partido e a classificação multirrótulos de vídeo por SVM.

Figura 49 – *Pipeline* de classificação de vídeos proposto por Jhuo *et al.*

A etapa representada na Figura 49 por um quadrado rosa, corresponde à fusão de informação proposta por Jhuo *et al.*. Tal etapa foi substituída pelo método M4InFus, resultando no *pipeline* ilustrado na Figura 50, onde o método M4InFus é representado pelo quadrado amarelo.

O *pipeline* de classificação resultante do acoplamento do método M4InFus dispensa a etapa de concatenação de vetores de características SIFT e STIP do *pipeline* de classificação original. Isso é possível graças ao fato do M4InFus ser escalável em relação à quantidade de modalidades envolvidas na fusão de informação. O *pipeline* de classificação acoplado ao M4InFus foi empregado neste experimento.

Figura 50 – Pipeline de classificação de vídeos modificado para emprego do M4InFus.

A base confiável empregada neste experimento, assim como no experimento de [Jhuo et al. \(2014\)](#), foi a *CCV Database*, descrita na Seção 2.5.2. Os vetores de características BoF baseados em MFCC, SIFT e STIP estão disponíveis para *download* por meio de arquivos de texto individuais que contêm os vetores relacionados aos vídeos do conjunto de treino e de teste. Há um arquivo com vetores de características BoF baseadas em MFCC representando os vídeos do conjunto de treino e outro semelhante representando os vídeos do conjunto de teste. Há arquivos de texto organizados de maneira semelhante contendo os vetores de características BoF baseados em SIFT e em STIP. Nestes arquivos, há uma linha contendo valores da representação BoF de cada vídeo, tal que cada coluna de uma linha corresponde às ocorrências de um padrão (palavra) do dicionário BoF naquele vídeo.

Os vetores de características BoF de cada tipo foram fundidos por um módulo de fusão de informação que implementa o método descrito no Capítulo 4. Este módulo foi implementado em Java/OpenIMAJ. Ele difere do módulo empregado na segmentação temporal de vídeo em cenas por ter de lidar com dois conjuntos de vídeos (treino e teste), ao invés de apenas um (tomadas de um vídeo). A diferença consiste no fato de que o módulo destinado à representação de conjuntos deve gerar o dicionário de padrões multimodais usando apenas as informações do conjunto de treino. Já a etapa onde se contabilizam as ocorrências de padrões multimodais em cada vídeo, resultando em um histograma por vídeo, é realizada para cada um dos vídeos do conjunto com todos os vídeos da base confiável. A geração de histogramas multimodais é feita com base no dicionário de padrões gerado sobre o conjunto de testes.

O tamanho de dicionário multimodal escolhido para o módulo de fusão proposto foi de 6000 padrões. Este tamanho foi escolhido para equivaler ao empregado em [Jhuo](#)

et al. (2014), garantindo uma comparação justa. O melhor resultado relatado por Jhuo *et al.* foi obtido pelo emprego do Multiple Kernel Learning (VEDALDI *et al.*, 2009), que cria um dicionário composto pela combinação de palavras bi-modais identificadas em diferentes granularidades de tamanho de dicionário. Como os autores variaram essa granularidade entre 2000 e 12000 palavras bi-modais e não apresentaram a configuração do dicionário final gerado, não há como replicá-lo no método de fusão proposto. Além disso, há a possibilidade de que o tamanho do dicionário final tenha sido muito próximo de uma simples concatenação dos vetores BoF STIP, SIFT e MFCC de cada tomada (14000 palavras). O dicionário bi-modal de tamanho fixo empregado nos experimentos de Jhuo *et al.* que permitiu a obtenção do melhor resultado reportado, tinha 6000 padrões (foram experimentados entre 2000 e 12000, com incrementos de 1000). Todas as estratégias de *pooling* descritas na Seção 4.4 foram experimentadas no *pipeline* modificado e o *average pooling* foi o que alcançou melhores resultados. O *max pooling* e as variações ponderadas tanto de *average* quanto de *max pooling* apresentaram resultados de eficácia ligeiramente mais baixos.

O módulo de classificação multirrótulos *one-vs-all* foi codificado em linguagem Python, fazendo uso das bibliotecas Scikit-Learn⁸ e Numpy⁹, que contém, respectivamente, ferramentas de aprendizado de máquina e de computação científica. A eficácia da técnica de classificação de vídeo resultante da substituição do método de fusão de informação por grafos bipartidos pelo método de fusão de informação proposto foi medida usando a medida MAP, descrita na Seção 2.5.2.

5.3.2 Resultados

A Tabela 17 apresenta uma comparação entre os valores de MAP obtidos pelo *pipeline* classificação de vídeo resultante do emprego do método de fusão de informação proposto no Capítulo 4 (coluna **SVM_{EF}**) e os valores MAP obtidos pelo *pipeline* original de Jhuo *et al.* (2014) (coluna **SVM_{Jhuo}**). Além dos dois *pipelines* alvo da comparação principal desse experimento, a coluna **SVM_{Naive}** contém os resultados obtidos por uma versão do *pipeline* de Jhuo *et al.* com fusão efetuada de modo ingênuo (concatenação dos vetores de características de diferentes modalidades).

Os valores de MAP obtidos pelo **SVM_{EF}** foram melhores do que os obtidos pelo **SVM_{Jhuo}** em 12 de 20 classes, tendo empatado em outras três. Além disso, o valor médio de MAP de todas as classes obtido pelo **SVM_{EF}** é 2% maior do que o valor de MAP obtido pelo **SVM_{Jhuo}**. Os valores de MAP obtidos pelo **SVM_{Naive}** são consideravelmente inferiores aos das duas outras abordagens.

O principal motivo para a melhor eficácia obtida pelo *pipeline* modificado está

⁸ Maiores informações em <<https://scikit-learn.org/stable/>>.

⁹ Maiores informações em <<https://numpy.org/>>.

Tabela 17 – Comparação entre valores de *Mean Average Precision* obtidos três *pipelines* de classificação de vídeo.

<i>Mean Average Precision</i>			
Classe	SVM _{Naive}	SVM _{Jhuo}	SVM _{EF}
Basquete	0,73	0,79	0,81
Baseball	0,53	0,57	0,62
Futebol	0,57	0,63	0,69
Patinando no gelo	0,81	0,87	0,90
Esquiando	0,72	0,78	0,81
Nadando	0,73	0,78	0,82
Andando de bicicleta	0,49	0,50	0,55
Gato	0,43	0,52	0,53
Cão	0,45	0,46	0,47
Pássaro	0,36	0,39	0,36
Graduação	0,48	0,56	0,56
Aniversário	0,58	0,57	0,56
Festa de casamento	0,31	0,33	0,39
Cerimônia de casamento	0,63	0,70	0,65
Dança de casamento	0,63	0,66	0,66
Performance musical	0,70	0,77	0,78
Performance não musical	0,69	0,69	0,67
Desfile	0,65	0,72	0,69
Praia	0,69	0,76	0,77
Área de recreação	0,55	0,62	0,62
Média	0,59	0,63	0,65

na capacidade do método de fusão de informação multimodal proposto de representar simultaneamente padrões unimodais e multimodais em um único vetor de características. Os vídeos da base confiável em questão possuem instâncias onde há ambos os tipos de padrão como, por exemplo os vídeos da classe “dança de casamento”. A maioria dos vídeos dessa classe exibem um ou mais homens vestindo gravatas e dançando ao som de música. Dois quadros extraídos de vídeos desta classe podem ser vistos na Figura 51.

O padrão “homem vestindo gravata” costuma ocorrer junto com a voz destes homens na modalidade aural. Já o padrão aural “música” não possui um padrão visual complementar. Ambos os elementos semânticos são úteis para a discriminação de vídeos da categoria “dança de casamento” em relação a outros vídeos que contenham apenas um dos dois padrões. Como exemplo, temos os vídeos da categoria “festa de casamento” ilustrados na Figura 52.

Estes vídeos, apesar de conterem o padrão “homem vestindo gravata”, não contém o padrão aural “música”. Ou seja, a presença do padrão aural unimodal “música” permite ao classificador *one-vs-all* classificar um vídeo como “festa de casamento” se identificar “homem vestindo gravata” e ainda classificá-lo adicionalmente como “dança de casamento”, caso detecte “música”.

Figura 51 – Exemplo de semântica expressa em dois diferentes vídeos das classes “festa de casamento” e “dança de casamento” da base confiável *CCV Database*.

Figura 52 – Exemplo de semântica expressa em dois diferentes vídeos da classe “festa de casamento” da base confiável *CCV Database*.

Todos os *pipelines* apresentaram baixos valores da medida MAP na classe “pássaro”. Isso se deve principalmente ao módulo de classificação SVM, que depende de treinamento. Os vídeos da CCV apresentam uma variedade muito grande de tipos pássaros, com aparência e sons distintos entre si, fazendo com que o módulo de classificação não fosse capaz de se especializar em todos.

O método de Jhuo *et al.* também peca por ser incapaz de analisar padrões compostos por informações provenientes de três ou mais tipos diferentes. Por ser baseado em um grafo bipartido, ele é limitado a fundir informações de duas origens distintas. O fato dos dicionários BoF SIFT e STIP terem sido concatenados para compor a partição visual

do grafo bipartido, faz com que não sejam inseridas arestas ligando padrões destes dois diferentes tipos de vetores de características. Como as arestas no grafo bipartido medem correlação entre os padrões correspondentes aos seus vértices, que posteriormente serão agrupados, não haverá identificação de correlação entre os padrões BoF SIFT e STIP separadamente. Retomando o exemplo ilustrado pelas Figuras 51 e 52 e lembrando que o extrator de características STIP detecta e representa movimento, o método de Jhuo *et al.* pode ter falhado em discriminar alguns vídeos pertencentes unicamente à classe “festa de casamento” e vídeos pertencentes às classes “festa de casamento” e “dança de casamento” simultaneamente, já que o padrão de movimentos é bastante diferente, mesmo que o conteúdo visual estático representado pelos padrões SIFT indique semelhança. O método de fusão de informação proposto, em contrapartida, permite a combinação separada de quaisquer dos padrões unimodais, independentemente de sua origem, permitindo a produção de representações mais expressivas.

5.4 Considerações finais

Este capítulo descreveu uma avaliação experimental comparativa realizada para verificar se o método M4InFus era capaz de propiciar ganhos em eficácia em tarefas de análise de vídeo. Como não há disponibilidade de procedimentos bem estabelecidos para efetuar a avaliação específica de métodos de fusão de informação multimodal, ela foi realizada de modo indireto. O método M4InFus foi combinado a técnicas de análise de vídeo estruturadas como *pipelines*, e avaliado de acordo com os procedimentos usuais das respectivas tarefas de análise de vídeo. As tarefas escolhidas foram a segmentação temporal de vídeo em cenas e a classificação de vídeo.

A avaliação foi dividida em três experimentos e no primeiro deles, Experimento 1, o método M4InFus foi acoplado a um *pipeline* bimodal (modalidades visual e aural) de segmentação temporal de vídeos composto por algoritmos e técnicas *baseline*. O objetivo do Experimento 1 era verificar se o método M4InFus ocasionaria aumento de eficácia em uma técnica *baseline* por ser capaz de produzir representações mais expressivas das tomadas. A base confiável utilizada nesse experimento foi a BBC *Dataset*, com a eficácia medida em Precisão/Abrangência e *Coverage/Overflow*.

No *pipeline* proposto para o experimento, a seleção de amostras da imagem em movimento foi realizada pelo método de Trojahn e Goularte (2013), que escolhe quadros chave. O áudio de cada tomada foi processado integralmente, não tendo sido amostrado, já que seu tamanho não dificulta o processamento. Ao usar todo o áudio de uma tomada, aumentou-se a disponibilidade para o método M4InFus correlacionar padrões aurais e visuais não sincronizados temporalmente. Foram extraídas características CSIFT dos quadros chave e características MFCC do áudio de cada tomada. Em seguida, cada tipo de carac-

terística foi processada individualmente pelo método BoF, cujo tamanho do dicionário foi determinado empiricamente analisando as eficácias obtidas na base confiável BBC *Dataset* para cada parâmetro variando em um determinado intervalo.

Em seguida, foram computadas as representações fundidas, por meio da identificação de co-ocorrência por agrupamento de padrões unimodais seguida da representação única por meio de *pooling*. O parâmetro de tamanho de dicionário fundido foi determinado como sendo igual aos dos dicionários unimodais computados anteriormente. As quatro estratégias de *pooling* apresentadas no Capítulo 4 foram experimentadas e o *max pooling* ponderado foi o que apresentou melhor eficácia. As representações fundidas foram fornecidas a uma implementação ingênua do algoritmo seminal de detecção de fronteiras entre cenas STG, descrito na Seção 2.4.1, que também foi parametrizado empiricamente usando a base confiável BBC *Dataset*.

Os resultados obtidos pelo *pipeline* proposto indicaram que houve sobresegmentação nas predições de posições de fronteiras entre cenas. Apesar disso, os altos valores de *coverage* indicaram que grandes frações das cenas reais foram corretamente identificadas. A comparação com outras técnicas de segmentação temporal de vídeo em cenas foi efetuada por meio da medida F_{CO} , escolhida por Baraldi, Grana e Cucchiara (2015a) para realização de seus experimentos. O *pipeline* proposto superou, em média, as técnicas de Sidiropoulos *et al.* (2011) e de Baraldi, Grana e Cucchiara (2015a). O *pipeline* proposto superou a técnica de Sidiropoulos *et al.* em todas as instâncias individualmente. Das 11 instâncias da base confiável BBC *Dataset*, o *pipeline* proposto superou a técnica de Baraldi, Grana e Cucchiara em 7 delas.

Os resultados obtidos no Experimento 1 mostram que o *pipeline* proposto foi mais eficaz do que as técnicas de Sidiropoulos *et al.* e de Baraldi, Grana e Cucchiara. Com isso foi possível confirmar a indagação inicial do experimento, já que uma técnica de segmentação temporal de vídeo em cenas composta exclusivamente por métodos e algoritmos *baseline* só pode ter sido capaz de superar técnicas plenamente desenvolvidas pelo emprego do método M4InFus.

O Experimento 2 também foi realizado na tarefa de segmentação temporal de vídeo em cenas. O objetivo desse experimento era verificar se o M4InFus seria flexível em relação ao tipo de informação a ser fundida e também testar seu potencial de refinamento de informação na tarefa em questão. Para tal, empregou-se o mesmo pipeline do Experimento 1, modificando-o para o uso de aprendizado profundo nas características de entrada.

A informação das tomadas foi representada por vetores de características produzidos por redes de aprendizado profundo ao invés de ter sido representada diretamente por vetores de características BoF produzidos a partir de características CSIFT e MFCC. Os vetores de características unimodais CSIFT e MFCC foram fornecidos como entrada, individualmente, a uma rede de aprendizado profundo composta por subredes neurais

convolucionais e recorrentes proposta por Trojahn (2019) para segmentação temporal de vídeo em cenas. Tal rede foi modificada para que fosse possível extrair representações dos dados antes de suas camadas finais de decisão. Além das *deep features* oriundas de CSIFT e MFCC, também foram utilizadas características produzidas pela rede neural convolucional VGGNet (SIMONYAN; ZISSERMAN, 2015), que extrai vetores de características diretamente de quadros chave de cada tomada.

O *pipeline* modificado foi experimentado nas bases confiáveis BBC *Dataset* e OVSD *Dataset* e avaliado usando as medidas de eficácia Precisão/Abrangência e *Coverage/Overflow*. Em relação aos resultados do *pipeline* original na base confiável BBC *Dataset*, o *pipeline* modificado apresentou uma redução de 27% na quantidade de cenas detectadas, diminuindo o problema da sobresegmentação. A menor sobresegmentação foi traduzida em um aumento de 0,12 em precisão. É interessante observar que mesmo com a menor quantidade de fronteiras entre cenas detectadas, houve um aumento de 0,06 em abrangência. Os valores de *Coverage/Overflow* também melhoraram, com aumento de 0,03 em *Coverage* e uma redução de 0,01 em *Overflow*. A medida F_{PR} aumentou de 0,59 obtida pelo *pipeline* original para 0,70 obtida pelo *pipeline* modificado. Na medida F_{CO} , o aumento foi de 0,01 em relação ao *pipeline* original.

Em uma comparação com a técnica de Baraldi, Grana e Cucchiara (2015a) na base confiável BBC *Dataset*, pôde-se observar que o *pipeline* modificado apresentou valores maiores de F_{CO} em 9 de 11 instâncias, tendo empatado em uma das duas instâncias restantes. A F_{CO} média foi 0,03 maior do que a da técnica de Baraldi, Grana e Cucchiara (2015a). como a técnica de Baraldi, Grana e Cucchiara também emprega *deep features*, os melhores resultados obtidos pelo *pipeline* modificado são uma evidência de que o M4InFus foi capaz de refinar a representação oriunda de aprendizado profundo, mostrando que é flexível quanto ao tipo de informação a ser fundida. Além disso, o *pipeline* modificado apresentou resultados nitidamente superiores ao de uma técnica plenamente desenvolvida, mesmo que ainda efetue a detecção de fronteiras entre cenas por um método *baseline* (STG). Deve estar claro que ao utilizar o M4InFus com algoritmos não *baseline*, os quais tratam, por exemplo, questões de sub e sobresegmentação, os resultados serão ainda melhores. Isso indica que a representação produzida pelo M4InFus é altamente expressiva.

Após a execução do *pipeline* modificado na base confiável BBC *Dataset*, ele também foi executado na base confiável OVSD *Dataset*. Dos resultados obtidos nas 21 instâncias da base confiável OVSD *Dataset*, 6 não puderam ser considerados na avaliação por F_{CO} . Isso ocorreu por uma combinação entre a incapacidade das redes neurais profundas empregadas no *pipeline* modificado de gerar representações adequadas de vídeos para os quais não há exemplos de treino suficientes e a inadequação da medida *Overflow* em avaliar situações de subsegmentação extrema. A medida *NO* foi computada por ser capaz de lidar com situações de subsegmentação extrema. Tal medida, entretanto, não é adotada por outros

pesquisadores, impedindo uma comparação completa dos resultados.

Os 15 valores de F_{CO} válidos obtidos foram comparados aos das técnicas multimodais em estado da arte apresentadas em [Rotman, Porat e Ashour \(2017\)](#) e [Rotman et al. \(2018\)](#). Nessa comparação, o resultado médio de F_{CO} foi superior ao do reportado em [Rotman, Porat e Ashour \(2017\)](#) por 0,04 e igual ao reportado em [Rotman et al. \(2018\)](#). Apesar dos resultados não se mostrarem superiores aos da técnica de [Rotman et al. \(2018\)](#), pode-se concluir que o método proposto foi, mais uma vez, capaz de capturar e representar informação relevante a uma tarefa de análise de vídeo. A obtenção de resultados equiparáveis aos de uma técnica que emprega um algoritmo de detecção de fronteiras entre cenas plenamente desenvolvido e mais complexo do que o STG usado no *pipeline* modificado é decorrente das representações mais expressivas computadas pelo M4InFus. Assim, reforça-se a afirmação de que o M4InFus é flexível quanto ao tipo de informação a ser fundida.

Ao combinar e analisar os resultados obtidos nos Experimentos 1 e 2 foi possível observar, ainda, que a fusão pelo método M4InFus levou a aumento de expressividade na representação fundida resultante independentemente do tipo das informações de origem. O aumento de expressividade levou a um aumento na eficácia nas tarefas de análise de vídeo que empregaram tais representações. Pelos resultados obtidos, pode-se afirmar que, quanto mais expressiva for a informação a ser fundida, mais expressiva será a representação resultante do processo de fusão, já que a detecção e uso de correlação tende a melhorar representações existentes.

O terceiro e último experimento, Experimento 3, tinha como objetivo a verificação da aplicabilidade do M4InFus em uma segunda tarefa de análise de vídeo, a classificação de vídeos. Nesse experimento, uma técnica de classificação de vídeos com etapa de fusão de características foi minuciosamente replicada, substituindo-se o método de fusão originalmente adotado pelo método M4InFus. Com isso, a obtenção de um aumento na eficácia indicaria que o método M4InFus, foi capaz de gerar representações mais expressivas do que as produzidas pelo método de fusão de características original e ainda que é útil em mais de um domínio de aplicação.

O Experimento 3 replicou toda a metodologia dos experimentos descritos em [Jhuo et al. \(2014\)](#). Assim como no trabalho [Jhuo et al.](#), esse experimento foi realizado com a base confiável *CCV Database* e os resultados obtidos foram avaliados com a medida MAP. Os experimentos de [Jhuo et al.](#) empregaram características SIFT, STIP e MFCC que foram disponibilizadas publicamente pelos autores. Tais arquivos de vetores de características pré-extraídas foram usadas no Experimento 3. Os vetores de características unimodais foram fornecidos ao método M4InFus e os vetores de características fundidos resultantes foram usados para realizar a classificação. A classificação foi efetuada por um arcabouço de classificadores SVM *One-vs-All*, configurado e parametrizado exatamente conforme

descrito em [Jhuo et al. \(2014\)](#).

Os valores da medida MAP obtidos pelo *pipeline* de classificação modificado SVM_{EF} foram melhores do que os obtidos pelo *pipeline* de classificação original SVM_{Jhuo} de [Jhuo et al. \(2014\)](#) em 12, tendo empatado em outras 3, de um total de 20 classes. A média da medida MAP obtida pelo SVM_{EF} considerando as 20 classes foi 0,02 superior à obtida pelo SVM_{Jhuo}. Dado que a única diferença entre ambos os experimentos consiste no método de fusão de características, conclui-se que a melhor eficácia obtida foi promovida pelo M4InFus. Os motivos pelos quais o método M4InFus levou a ganhos em eficácia em relação ao método do grafo bipartido de Jhuo et al. são a capacidade do método M4InFus em representar simultaneamente padrões unimodais e multimodais em vetores de características e a capacidade de analisar padrões compostos por mais de duas fontes de informação. O fato de que o método M4InFus viabilizou o aumento em eficácia na tarefa de classificação de vídeos, é um indício de que ele é generalizável quanto ao domínio de aplicação.

CONCLUSÃO

Este capítulo contém as considerações finais à respeito da pesquisa desenvolvida durante este doutorado. A Seção 6.1 destaca as contribuições do trabalho realizado neste doutorado. A Seção 6.2 lista as publicações geradas durante o doutorado que decorreram de partes do trabalho realizado, acompanhadas de uma explicação sobre sua relação com o trabalho realizado. Por fim, na Seção 6.3 são discutidas as limitações existentes no método de fusão de informação proposto, bem como os aspectos que ainda podem ser explorados em trabalhos futuros.

6.1 Contribuições

O objetivo desta tese era propor um método de fusão prévia de informação multimodal capaz de propiciar ganhos de eficácia em diferentes tarefas de análise de vídeo. Esse objetivo foi alcançado. Um novo método de fusão multimodal foi apresentado e em seguida, demonstrou-se que ele é aplicável a mais de uma tarefa de análise de vídeo, combinando-o a duas diferentes técnicas de segmentação temporal de vídeo em cenas e a uma técnica de classificação de vídeo. O método proposto propiciou ganhos de eficácia em todas as tarefas nas quais foi aplicado e avaliado, conforme os resultados apresentados no Capítulo 5. Dessa forma, este trabalho tem como principal contribuição um método de fusão de informação multimodal eficaz e com bons indícios de ser generalizável.

A principal lacuna no estado da arte coberta é a inflexibilidade dos métodos existentes até então em representar, simultaneamente, padrões unimodais e multimodais. O método de fusão de informação proposto foi capaz de identificar e representar adequadamente tais padrões em conjunto, fazendo com que a representação final fosse mais expressiva. Outra importante lacuna coberta pelo método proposto é a da falta de escalabilidade em relação à quantidade de modalidades/diferentes tipos de características que podem ser combinadas. O método proposto foi capaz de combinar diretamente tantos ti-

pos de características quanto necessário em cada um dos experimentos, extraindo de cada tipo de características as informações mais relevantes para a representação adequada da semântica original do conteúdo. Por fim, o método proposto também garantiu a rastreabilidade do significado dos valores originais unimodais na representação fundida, o que permitiu seu processamento por meio da ponderação. Esse pós processamento permitiu destacar a semântica latente existente nas representações finais.

A avaliação experimental mostrou que representações geradas pelo M4InFus viabilizaram ganhos em eficácia em tarefas de análise de vídeo. Tal informação, aliada ao fato de que eficácia em tais tarefas é limitada pela Lacuna Semântica, é um indício de que representações geradas pelo M4InFus são menos distantes da semântica contida nos vídeos, o que pode significar uma redução na Lacuna Semântica.

Além do método M4InFus, o trabalho desenvolvido durante este doutorado também gerou como contribuição a implementação do método M4InFus e a formação de recursos humanos qualificados em nível de doutorado e de iniciação científica.

6.2 Publicações relacionadas

O trabalho realizado durante este doutorado resultou na publicação dos seguintes trabalhos em anais de eventos científicos:

- KISHI, R. M.; TROJAHN, T. H.; GOULARTE, R. An Evaluation of Readily Usable Automatic Video Shot Segmentation Techniques. Em Simpósio Brasileiro de Sistemas Multimídia e Web, 2016, Teresina. Webmedia '16: Proceedings of the 22Nd Brazilian Symposium on Multimedia and the Web. New York: ACM, 2016. v. 1. p. 199-202.
- TROJAHN, T. H.; KISHI, R. M.; GOULARTE, R. A fast video shot segmentation tool. Em Simpósio Brasileiro de Sistemas Multimídia e Web, 2016, Teresina. Anais do XXII Simpósio Brasileiro de Sistemas Multimídia e Web. Porto Alegre: Sociedade Brasileira de Computação, 2016. v. 2. p. 128-129.
- KISHI, R. M.; GOULARTE, R. . Video scene segmentation through an early fusion multimodal approach. Em Simpósio Brasileiro de Sistemas Multimídia e Web, 2016, Teresina. Anais do XXII Simpósio Brasileiro de Sistemas Multimídia e Web. Porto Alegre: Sociedade Brasileira de Computação, 2016. v. 2. p. 41-46.
- KISHI, R. M.; TROJAHN, T. H.; GOULARTE, R. . Temporal Video Scene Segmentation By Fused Bags-of-Features. Em Simpósio Brasileiro de Sistemas Multimídia e Web, 2018, Salvador. Proceedings of the 24th Brazilian Symposium on Multimedia and the Web - WebMedia '18. New York: ACM, 2018. p. 173-180.

- TROJAHN, T. H.; KISHI, R. M.; GOULARTE, R. A new multimodal deep-learning model to video scene segmentation. Em Simpósio Brasileiro de Sistemas Multimídia e Web, 2018, Salvador. Proceedings of the 24th Brazilian Symposium on Multimedia and the Web - WebMedia '18. New York: ACM, 2018. p. 205-212.

O artigo curto “*An Evaluation of Readily Usable Automatic Video Shot Segmentation Techniques*”, descreve uma avaliação realizada com técnicas de segmentação temporal de vídeo em tomadas disponíveis naquela ocasião. Este trabalho foi realizado em uma etapa preliminar dos experimentos em segmentação temporal de vídeo em cenas, que dependiam da obtenção de anotações de posições de fronteiras entre tomadas.

O pôster “*A fast video shot segmentation tool*”, apresenta um ambiente web desenvolvido para facilitar e organizar a segmentação temporal de vídeo em tomadas. A motivação para desenvolver esse ambiente surgiu nos experimentos preliminares em segmentação temporal de vídeo em cenas, quando observou-se que todas as ferramentas de segmentação temporal de vídeo em tomadas disponíveis eram acessíveis somente via linha de comando e contando com sintaxe complexa. O ambiente foi projetado tal que, no futuro, seja possível estendê-lo para que se torne um ambiente completo de análise automática de vídeo, onde serão integradas funcionalidades como segmentação temporal de vídeo em tomadas e classificação de vídeos/cenas.

O artigo curto “*Video scene segmentation through an early fusion multimodal approach*”, foi publicado nos anais do evento Webmedia e descrevia os objetivos deste projeto de doutorado, bem como a metodologia prevista para alcançá-los.

O artigo completo “*Temporal Video Scene Segmentation By Fused Bags-of-Features*”, descreve uma extensão do trabalho apresentado no artigo intitulado “*Correlation based feature fusion for the temporal video scene segmentation task*”. Neste trabalho, foram realizados novos experimentos com extratores de características mais adequados, o que levou a melhora na eficácia do método de segmentação temporal de vídeo em cenas. Os resultados deste artigo estão presentes na Seção 5.1. Este trabalho foi agraciado com uma menção honrosa no evento WebMedia 2018.

O artigo completo “*A new multimodal deep-learning model to video scene segmentation*” descreve uma técnica de segmentação temporal de vídeo em cenas multimodal, baseada em aprendizado profundo. Este artigo apresenta uma versão preliminar do método de extração de características por aprendizado profundo empregado no experimento descrito na Seção 5.2.

Além das publicações em anais de eventos, o trabalho realizado durante o doutorado também resultou na escrita do seguinte artigo em periódico:

KISHI, R. M.; TROJAHN, T. H.; GOULARTE, R. Correlation based fea-

ture fusion for the temporal video scene segmentation task. MULTIMEDIA TOOLS AND APPLICATIONS, v. 1, p. 1-24, 2018.

Esse artigo descreve um método de segmentação temporal de vídeo em cenas baseado no método de fusão de informação descrito no Capítulo 4, combinado com extratores de características *baseline* (SIFT e MFCC) e um algoritmo de agrupamento de tomadas *baseline* (STG).

Há um segundo artigo que foi submetido ao Periódico Multimedia Tools and Applications e se encontra em fase de avaliação, no qual o autor desta tese atuou como co-autor.

Por fim, este doutorado também resultou na publicação do seguinte capítulo de livro:

GOULARTE, R; TROJAHN, T. H.; KISHI, R. M. Recuperação de Informação Multimídia em Big Data Utilizando OPENCV Pyhton. Em XXV Simpósio Brasileiro de Sistemas Multimídia e Web: Minicursos. Rio de Janeiro, Brasil: Sociedade Brasileira de Computação, 2019. p. 165-199. ISBN 978-85-7669-481-6.

Esse capítulo de livro descreve um minicurso destinado a introduzir seus respectivos alunos às ferramentas e técnicas atuais de indexação, extração e processamento de conteúdo multimídia multimodal, por meio da biblioteca OpenCV Python¹.

6.3 Limitações e trabalhos futuros

A primeira das limitações observadas no M4InFus diz respeito à impossibilidade de representar separadamente determinados padrões distintos entre si que ocorram nos mesmos segmentos de vídeo. Essa limitação é decorrente da granularidade das unidades de vídeo empregadas pelo método. Como o método detecta co-ocorrência de padrões unimodais por histogramas de ocorrência em segmentos de vídeo inteiros, um determinado padrão *A* que não é relacionado e sequer ocorre de modo temporalmente sincronizado com um segundo padrão *B*, pode ser erroneamente considerado correlacionado a ele quando ambos apresentarem ocorrências semelhantes nos mesmos segmentos de vídeo. Como exemplo, suponha que *A* e *B* representam dois padrões que participam, respectivamente, da composição da expressão dos conceitos “carro” e “pistola”. Caso *A* e *B* sejam expressos em proporções parecidas nos mesmos segmentos de vídeo, será detectada co-ocorrência entre eles, o que fará com que componham um mesmo padrão multimodal na representação.

¹ Maiores informações em <<https://pypi.org/project/opencv-python/>>.

Uma segunda limitação observada no método M4InFus é a dependência de parametrização dos algoritmos de agrupamento empregados na geração dos dicionários unimodais e multimodal. Observou-se, nos experimentos, que a eficácia do método M4InFus é altamente dependente da parametrização adequada dos algoritmos de agrupamento que o integram. Como a parametrização costuma influir na quantidade de grupos e, consequentemente, no tamanho dos dicionários, parâmetros que levem à produção de poucos grupos podem fazer com que padrões que não deveriam ser considerados como correlacionados sejam agrupados, tornando a representação menos precisa.

Trabalhos futuros

Seria interessante, em trabalhos futuros, verificar a aplicabilidade do método M4InFus em outras tarefas de análise de vídeo além das abordadas neste trabalho, ou mesmo em outras tarefas de análise de informação multimídia. Um exemplo de tarefa na qual o método M4InFus pode ser aplicado é a *cross-modal retrieval* (KARPATHY; FEI-FEI, 2017; HUANG *et al.*, 2019), ou, em português, recuperação cruzada entre modalidades. Esta tarefa consiste na recuperação de informação de uma determinada modalidade, descrevendo tal informação por meio de informação de outra modalidade. Considerando que o método de fusão de informação proposto possui uma etapa de identificação de correlação entre informação de origens distintas, pode ser possível utilizá-lo para, em um passo seguinte, computar distâncias entre instâncias de diferentes modalidades, e, parafraseando Huang *et al.* (2019): desemaranhar as relações compartilhadas por informações multimodais.

No trabalho atual, o método M4InFus foi experimentado com até três diferentes tipos de características. Espera-se que a quantidade e o tipo das fontes de informação adotadas como entrada influenciem diretamente na expressividade da representação fundida. A adoção de mais tipos de características e/ou modalidades disponibiliza mais informações para a síntese da representação fundida e assim permite o aumento na expressividade. É esperado, também, que haja um limiar para a quantidade de fontes de informação que, quando ultrapassado, passe a gerar ruído e até mesmo dificulte o processamento. Além da quantidade, pôde-se observar que a fusão de características mais expressivas como as provenientes de aprendizado profundo, levou a aumento de expressividade na também na representação fundida e consequente aumento na eficácia da tarefa de análise de vídeo que a empregou. Ou seja, quanto mais expressiva for a informação a ser fundida, mais expressiva será a representação resultante do processo de fusão, já que a detecção e uso de correlação tende apenas a melhorar representações existentes. Tais indagações à respeito das informações a serem fundidas merecem ser melhor estudadas para que seja possível aproveitar o potencial do método M4InFus.

O método de fusão de informação multimídia proposto foi avaliado de maneira indireta, tendo sido acoplado a métodos existentes destinados a tarefas de análise de ví-

deo. Foi necessário inferir os ganhos em eficácia proporcionados pelo método de fusão de informação multimídia de resultados produzidos em experimentos que avaliaram primordialmente estes outros métodos, incrementados pelo método de fusão de informação proposto. A criação de um protocolo específico para realizar avaliação, possivelmente acompanhado de uma base confiável e de uma medida de avaliação voltados à avaliação isolada de métodos de fusão de informação multimídia será de grande valia.

No método proposto, a detecção de co-ocorrência é realizada agrupando as palavras unimodais por histogramas de ocorrência calculados diretamente com as unidades de vídeo de uma determinada tarefa de análise de vídeo. São calculados, por exemplo, histogramas de ocorrência em tomadas na segmentação temporal de vídeo em cenas, e histogramas de ocorrência em vídeos na classificação de vídeo. É possível que a identificação de co-ocorrência em unidades menores ou maiores de vídeo permita que se identifique a co-ocorrência de maneira mais precisa, ou mesmo que se possibilite a identificação de alguma categoria de co-ocorrência impossível de ser identificada pela abordagem atual.

Há ainda a possibilidade de realizar um estudo sobre outras estratégias de *pooling*, já que o trabalho realizado se limitou à aplicação e experimentação com o *max* e o *average pooling*. Além do emprego de outras estratégias de *pooling*, o *pooling* ponderado apresentado neste trabalho é bastante rudimentar. Há diversas possibilidades de torná-lo dinâmico em relação aos dados que representa, por meio de ponderação relativa à modalidade, ponderação proporcional ao tamanho do padrão fundido, dentre outras heurísticas.

REFERÊNCIAS

ABU-EL-HAIJA, S.; KOTHARI, N.; LEE, J.; NATSEV, P.; TODERICI, G.; VARADARAJAN, B.; VIJAYANARASIMHAN, S. Youtube-8m: A large-scale video classification benchmark. **CoRR**, abs/1609.08675, 2016. Disponível em: <http://arxiv.org/abs/1609.08675>. Citado na página 69.

AGGARWAL, C. C.; ZHAI, C. A survey of text classification algorithms. In: _____. **Mining Text Data**. Boston, MA: Springer US, 2012. p. 163–222. ISBN 978-1-4614-3223-4. Disponível em: https://doi.org/10.1007/978-1-4614-3223-4_6. Citado na página 48.

ARFKEN, G. B.; WEBER, H. J. **Mathematical Methods for Physicists, Seventh Edition: A Comprehensive Guide**. 7. ed. Waltham, MA, USA: Academic Press, 2012. Hardcover. ISBN 0123846544. Citado na página 44.

ATREY, P. K.; HOSSAIN, M. A.; SADDIK, A. E.; KANKANHALLI, M. S. Multi-modal fusion for multimedia analysis: a survey. **Multimedia Systems**, v. 16, n. 6, p. 345–379, Nov 2010. ISSN 1432-1882. Disponível em: <https://doi.org/10.1007/s00530-010-0182-0>. Citado nas páginas 27, 41, 50 e 51.

BACH, F. R.; JORDAN, M. I. Kernel independent component analysis. **J. Mach. Learn. Res.**, JMLR.org, v. 3, p. 1–48, mar. 2003. ISSN 1532-4435. Disponível em: <https://doi.org/10.1162/153244303768966085>. Citado na página 80.

BALLAN, L.; BERTINI, M.; BIMBO, A. D.; SEIDENARI, L.; SERRA, G. Event detection and recognition for semantic annotation of video. **Multimedia Tools and Applications**, v. 51, n. 1, p. 279–302, Jan 2011. ISSN 1573-7721. Disponível em: <https://doi.org/10.1007/s11042-010-0643-7>. Citado na página 56.

BARALDI, L.; GRANA, C.; CUCCHIARA, R. A deep siamese network for scene detection in broadcast videos. In: **Proceedings of the 23rd ACM International Conference on Multimedia**. New York, NY, USA: ACM, 2015. (MM '15), p. 1199–1202. ISBN 978-1-4503-3459-4. Disponível em: <http://doi.acm.org/10.1145/2733373.2806316>. Citado nas páginas 14, 17, 28, 29, 38, 60, 62, 67, 68, 77, 79, 82, 83, 85, 90, 91, 95, 96, 107, 110, 111, 114, 118, 124, 125, 135 e 136.

_____. Measuring scene detection performance. In: PAREDES, R.; CARDOSO, J. S.; PARDO, X. M. (Ed.). **Pattern Recognition and Image Analysis**. Cham: Springer International Publishing, 2015. p. 395–403. ISBN 978-3-319-19390-8. Citado na página 112.

_____. Shot and scene detection via hierarchical clustering for re-using broadcast video. In: _____. **Computer Analysis of Images and Patterns: 16th International Conference, CAIP 2015, Valletta, Malta, September 2-4, 2015 Proceedings, Part I**.

Cham: Springer International Publishing, 2015. p. 801–811. ISBN 978-3-319-23192-1. Disponível em: <http://dx.doi.org/10.1007/978-3-319-23192-1_67>. Citado nas páginas 46 e 120.

_____. Analysis and re-use of videos in educational digital libraries with automatic scene detection. In: CALVANESE, D.; NART, D. D.; TASSO, C. (Ed.). **Digital Libraries on the Move**. Cham: Springer International Publishing, 2016. p. 155–164. ISBN 978-3-319-41938-1. Citado nas páginas 85 e 90.

_____. Recognizing and presenting the storytelling video structure with deep multimodal networks. **IEEE Transactions on Multimedia**, v. 19, n. 5, p. 955–968, May 2017. ISSN 1520-9210. Citado nas páginas 47 e 76.

BEAUDET, P. R. Rotationally invariant image operators. In: **Proceedings of the 4th International Joint Conference on Pattern Recognition**. Kyoto, Japan: IEEE, 1978. p. 579–583. Citado na página 44.

BESACIER, L.; QUÉNOT, G.; AYACHE, S.; MORARU, D. Video story segmentation with multi-modal features: Experiments on trecvid 2003. In: **Proceedings of the 6th ACM SIGMM International Workshop on Multimedia Information Retrieval**. New York, NY, USA: ACM, 2004. (MIR '04), p. 221–227. ISBN 1-58113-940-3. Disponível em: <<http://doi.acm.org/10.1145/1026711.1026748>>. Citado nas páginas 64 e 65.

BOSER, B. E.; GUYON, I. M.; VAPNIK, V. N. A training algorithm for optimal margin classifiers. In: **Proceedings of the Fifth Annual Workshop on Computational Learning Theory**. New York, NY, USA: ACM, 1992. (COLT '92), p. 144–152. ISBN 0-89791-497-X. Disponível em: <<http://doi.acm.org/10.1145/130385.130401>>. Citado na página 56.

BOUREAU, Y.; BACH, F.; LECUN, Y.; PONCE, J. Learning mid-level features for recognition. In: **2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition**. San Francisco, CA, USA: IEEE, 2010. p. 2559–2566. ISSN 1063-6919. Citado na página 41.

BREZEALE, D.; COOK, D. J. Automatic video classification: A survey of the literature. **Trans. Sys. Man Cyber Part C**, IEEE Press, Piscataway, NJ, USA, v. 38, n. 3, p. 416–430, maio 2008. ISSN 1094-6977. Disponível em: <<https://doi.org/10.1109/TSMCC.2008.919173>>. Citado nas páginas 24, 56 e 75.

BROMLEY, J.; GUYON, I.; LECUN, Y.; SÄCKINGER, E.; SHAH, R. Signature verification using a “siamese” time delay neural network. In: **Proceedings of the 6th International Conference on Neural Information Processing Systems**. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993. (NIPS'93), p. 737–744. Citado na página 82.

BRUNELLI, R.; MICH, O.; MODENA, C. A survey on the automatic indexing of video data,. **J. Vis. Comun. Image Represent.**, Academic Press, Inc., Orlando, FL, USA, v. 10, n. 2, p. 78–112, jun. 1999. ISSN 1047-3203. Disponível em: <<http://dx.doi.org/10.1006/jvci.1997.0404>>. Citado nas páginas 24 e 39.

BUCKLEY, C.; VOORHEES, E. M. Evaluating evaluation measure stability. In: **Proceedings of the 23rd Annual International ACM SIGIR Conference on Research**

and Development in Information Retrieval. New York, NY, USA: ACM, 2000. (SIGIR '00), p. 33–40. ISBN 1-58113-226-3. Disponível em: <<http://doi.acm.org/10.1145/345508.345543>>. Citado na página 71.

BURGHOUTS, G. J.; GEUSEBROEK, J.-M. Performance evaluation of local colour invariants. **Comput. Vis. Image Underst.**, Elsevier Science Inc., New York, NY, USA, v. 113, n. 1, p. 48–62, jan. 2009. ISSN 1077-3142. Disponível em: <<http://dx.doi.org/10.1016/j.cviu.2008.07.003>>. Citado nas páginas 111 e 113.

CARLETTI, V.; FOGGIA, P.; PERCANNELLA, G.; SAGGESE, A.; STRISCIUGLIO, N.; VENTO, M. Audio surveillance using a bag of aural words classifier. In: **2013 10th IEEE International Conference on Advanced Video and Signal Based Surveillance**. Krakow, Poland: IEEE, 2013. p. 81–86. Citado na página 48.

CHAISORN, L.; CHUA, T.-S.; LEE, C.-H. A multi-modal approach to story segmentation for news video. **World Wide Web**, v. 6, n. 2, p. 187–208, Jun 2003. ISSN 1573-1413. Disponível em: <<https://doi.org/10.1023/A:1023622605600>>. Citado na página 77.

CHAPMAN, N.; CHAPMAN, J. **Digital Multimedia**. 3rd. ed. Chichester, United Kingdom: Wiley Publishing, 2009. ISBN 0470512164. Citado nas páginas 34, 35 e 36.

CHASANIS, V.; KALOGERATOS, A.; LIKAS, A. Movie segmentation into scenes and chapters using locally weighted bag of visual words. In: **Proceedings of the ACM International Conference on Image and Video Retrieval**. New York, NY, USA: ACM, 2009. (CIVR '09), p. 35:1–35:7. ISBN 978-1-60558-480-5. Citado na página 77.

CHASANIS, V. T.; LIKAS, A. C.; GALATSANOS, N. P. Scene detection in videos using shot clustering and sequence alignment. **IEEE Transactions on Multimedia**, v. 11, n. 1, p. 89–100, Jan 2009. ISSN 1520-9210. Citado nas páginas 52, 60, 79 e 83.

CHEN, H.; LI, C. A practical method for video scene segmentation. In: **2010 3rd International Conference on Computer Science and Information Technology**. Chengdu, China: IEEE, 2010. v. 9, p. 153–156. Citado na página 42.

CHEN, L.; OZSU, M. T. Rule-based scene extraction from video. In: **Proceedings. International Conference on Image Processing**. New York, USA: IEEE, 2002. v. 2, p. II–II. ISSN 1522-4880. Citado na página 52.

CHEN, L.-H.; LAI, Y.-C.; LIAO, H.-Y. M. Movie scene segmentation using background information. **Pattern Recogn.**, Elsevier Science Inc., New York, NY, USA, v. 41, n. 3, p. 1056–1065, mar. 2008. ISSN 0031-3203. Disponível em: <<http://dx.doi.org/10.1016/j.patcog.2007.07.024>>. Citado na página 40.

CHEN, Z.; LU, J.; FENG, J.; ZHOU, J. Nonlinear structural hashing for scalable video search. **IEEE Transactions on Circuits and Systems for Video Technology**, v. 28, n. 6, p. 1421–1433, June 2018. ISSN 1051-8215. Citado na página 23.

CHENG, W.; LU, J. Video scene oversegmentation reduction by tempo analysis. In: **2008 Fourth International Conference on Natural Computation**. Jinan, China: IEEE, 2008. v. 4, p. 296–300. ISSN 2157-9563. Citado na página 117.

CISCO. **Cisco Visual Networking Index: Forecast and Trends, 2017–2022**. 2018. White Paper. Disponível em: <<https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white-paper-c11-741490.html>>. Citado na página 23.

CITeseerX. **Most Cited Computer Science Articles**. 2015. Disponível em: <<http://citeseerx.ist.psu.edu/stats/articles>>. Citado na página 43.

COLLYDA, C.; APOSTOLIDIS, E.; POURNARAS, A.; MARKATOPOULOU, F.; MEZARIS, V.; PATRAS, I. Videoanalysis4all: An on-line tool for the automatic fragmentation and concept-based annotation, and the interactive exploration of videos. In: **Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval**. New York, NY, USA: ACM, 2017. (ICMR '17), p. 470–474. ISBN 978-1-4503-4701-3. Disponível em: <<http://doi.acm.org/10.1145/3078971.3079015>>. Citado na página 55.

CORTES, C.; VAPNIK, V. Support-vector networks. **Machine Learning**, v. 20, n. 3, p. 273–297, Sep 1995. ISSN 1573-0565. Disponível em: <<https://doi.org/10.1007/BF00994018>>. Citado na página 56.

CSURKA, G.; DANCE, C. R.; FAN, L.; WILLAMOWSKI, J.; BRAY, C. Visual categorization with bags of keypoints. In: **In Workshop on Statistical Learning in Computer Vision, ECCV**. Czech Republic: Springer, 2004. p. 1–22. Citado na página 48.

CULA, O. G.; DANA, K. J. Compact representation of bidirectional texture functions. In: **Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on**. Kauai, HI, USA, USA: IEEE, 2001. v. 1, p. I–1041–I–1047 vol.1. ISSN 1063-6919. Citado na página 48.

DAMMAK, N.; BENAYED, Y. Hierarchical vs non-hierarchical audio indexation and classification for video genres. In: **Tenth International Conference on Machine Vision**. Vienna, Austria: SPIE, 2017. v. 10696. Disponível em: <<https://doi.org/10.1117/12.2309852>>. Citado na página 47.

DAVIS, S.; MERMELSTEIN, P. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. **IEEE Transactions on Acoustics, Speech, and Signal Processing**, v. 28, n. 4, p. 357–366, Aug 1980. ISSN 0096-3518. Citado nas páginas 46 e 111.

DENG, J.; DONG, W.; SOCHER, R.; LI, L.; LI, K.; FEI-FEI, L. Imagenet: A large-scale hierarchical image database. In: **2009 IEEE Conference on Computer Vision and Pattern Recognition**. Miami, FL, USA: IEEE, 2009. p. 248–255. Citado nas páginas 86 e 120.

DHILLON, I. S. Co-clustering documents and words using bipartite spectral graph partitioning. In: **Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining**. New York, NY, USA: ACM, 2001. (KDD '01), p. 269–274. ISBN 1-58113-391-X. Disponível em: <<http://doi.acm.org/10.1145/502512.502550>>. Citado na página 89.

DU, S.; IKENAGA, T. Low-dimensional superpixel descriptor and its application in visual correspondence estimation. **Multimedia Tools and Applications**, Feb 2019. ISSN 1573-7721. Disponível em: <<https://doi.org/10.1007/s11042-019-7248-6>>. Citado na página 43.

- ELLOUZE, M.; BOUJEMAA, N.; ALIMI, A. M. Scene pathfinder: unsupervised clustering techniques for movie scenes extraction. **Multimedia Tools and Applications**, v. 47, n. 2, p. 325–346, Apr 2010. ISSN 1573-7721. Disponível em: <<https://doi.org/10.1007/s11042-009-0325-5>>. Citado na página 28.
- ELMAN, J. L. Finding structure in time. **Cognitive Science**, v. 14, n. 2, p. 179–211, 1990. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1207/s15516709cog1402_1>. Citado na página 119.
- EVALUATION, T. V. R. **TRECVID Multimedia Event Detection Evaluation Track**. 2010. Disponível em: <<https://www.nist.gov/itl/iad/mig/trecvid-multimedia-event-detection-evaluation-track>>. Citado na página 69.
- EVERINGHAM, M.; GOOL, L. V.; WILLIAMS, C. K. I.; WINN, J.; ZISSERMAN, A. The pascal visual object classes (voc) challenge. **International Journal of Computer Vision**, v. 88, n. 2, p. 303–338, Jun 2010. ISSN 1573-1405. Disponível em: <<https://doi.org/10.1007/s11263-009-0275-4>>. Citado na página 72.
- FABRO, M. D.; BÖSZÖRMENYI, L. State-of-the-art and future challenges in video scene detection: A survey. **Multimedia Systems**, Springer-Verlag New York, Inc., Secaucus, NJ, USA, v. 19, n. 5, p. 427–454, out. 2013. ISSN 0942-4962. Disponível em: <<http://dx.doi.org/10.1007/s00530-013-0306-4>>. Citado nas páginas 24, 39, 52, 53, 60, 64, 65, 75, 76, 113 e 117.
- FANT, G. **Acoustic Theory of Speech Production**. The Hague, France: Mouton De Gruyter, 1970. (D A C S R Series). ISBN 9789027916006. Citado na página 47.
- FENG, H.; ZHAI, X.; FAN, J.; FANG, Y. Story segmentation in news video. In: **2005 International Conference on Neural Networks and Brain**. Beijing, China: IEEE, 2005. v. 2, p. 831–835. ISSN null. Citado na página 77.
- FONCUBIERTA-RODRÍGUEZ, A.; MÜLLER, H.; DEPEURSINGE, A. Retrieval of high-dimensional visual data: Current state, trends and challenges ahead. **Multimedia Tools Appl.**, Kluwer Academic Publishers, Hingham, MA, USA, v. 69, n. 2, p. 539–567, mar. 2014. ISSN 1380-7501. Disponível em: <<http://dx.doi.org/10.1007/s11042-012-1327-2>>. Citado na página 28.
- FYFE, C.; LAI, P. L. Ica using kernel canonical correlation analysis. In: **In Proc. Int. Workshop on Independent Component Analysis and Blind Signal Separation (ICA2000)**. Helsinki, Finland: Helsinki University of Technology, 2000. p. 279–284. Citado na página 80.
- GAO, G.; MA, H. Multi-modality movie scene detection using kernel canonical correlation analysis. In: **Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)**. Tsukuba, Japan: IEEE, 2012. p. 3074–3077. ISSN 1051-4651. Citado nas páginas 14, 17, 28, 29, 62, 77, 78, 80, 81, 91 e 107.
- GIALAMPOUKIDIS, I.; CHATZILARI, E.; NIKOLOPOULOS, S.; VROCHIDIS, S.; KOMPATSIARIS, I. Multimodal fusion of big multimedia data. In: _____. **Big Data Analytics for Large-Scale Multimedia Search**. Hoboken, NJ, USA: John Wiley & Sons, Ltd, 2019. cap. 5, p. 121–156. ISBN 9781119376996. Citado na página 97.

GRAUMAN, K.; LEIBE, B. **Visual Object Recognition**. 1st. ed. United States: Morgan & Claypool Publishers, 2011. ISBN 1598299689, 9781598299687. Citado na página 42.

GROSS, B. M. The managing of organizations: The administrative struggle, vols. i and ii. **The ANNALS of the American Academy of Political and Social Science**, Sage Publications, v. 360, n. 1, p. 197–198, 1965. Citado na página 23.

HAMMING, R. W. Error detecting and error correcting codes. **The Bell System Technical Journal**, v. 29, n. 2, p. 147–160, April 1950. ISSN 0005-8580. Citado na página 68.

HAN, B.; WU, W. Video scene segmentation using a novel boundary evaluation criterion and dynamic programming. In: **2011 IEEE International Conference on Multimedia and Expo**. Barcelona, Spain: IEEE, 2011. p. 1–6. ISSN 1945-7871. Citado nas páginas 68 e 112.

HANJALIC, A.; LAGENDIJK, R. L.; BIEMOND, J. Automated high-level movie segmentation for advanced video-retrieval systems. **IEEE Transactions on Circuits and Systems for Video Technology**, v. 9, n. 4, p. 580–588, June 1999. ISSN 1051-8215. Citado nas páginas 39, 40 e 66.

HARDOON, D. R.; SZEDMAK, S.; SHAW-ETAYLOR, J. Canonical correlation analysis: An overview with application to learning methods. **Neural Computation**, v. 16, n. 12, p. 2639–2664, Dec 2004. ISSN 0899-7667. Citado na página 80.

HARE, J. S.; SAMANGOOEI, S.; DUPPLAW, D. P. Openimaj and imagerterrier: Java libraries and tools for scalable multimedia analysis and indexing of images. In: **Proceedings of the 19th ACM international conference on Multimedia**. New York, NY, USA: ACM, 2011. (MM '11), p. 691–694. ISBN 978-1-4503-0616-4. Disponível em: <<http://doi.acm.org/10.1145/2072298.2072421>>. Citado na página 111.

HAUPTMANN, A. Video content analysis. In: _____. **Encyclopedia of Database Systems**. Boston, MA: Springer US, 2009. p. 3271–3276. ISBN 978-0-387-39940-9. Citado na página 23.

HEILBRON, F. C.; ESCORCIA, V.; GHANEM, B.; NIEBLES, J. C. Activitynet: A large-scale video benchmark for human activity understanding. In: **2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**. Boston, MA, USA: IEEE, 2015. p. 961–970. ISSN 1063-6919. Citado na página 69.

HERSHEY, S.; CHAUDHURI, S.; ELLIS, D. P. W.; GEMMEKE, J. F.; JANSEN, A.; MOORE, R. C.; PLAKAL, M.; PLATT, D.; SAUROUS, R. A.; SEYBOLD, B.; SLANEY, M.; WEISS, R. J.; WILSON, K. Cnn architectures for large-scale audio classification. In: **2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)**. New Orleans, LA, USA: IEEE, 2017. p. 131–135. Citado na página 86.

HIEMSTRA, D.; KRAAIJ, W. Evaluation of multimedia retrieval systems. In: _____. **Multimedia Retrieval**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007. p. 347–366. ISBN 978-3-540-72895-5. Disponível em: <https://doi.org/10.1007/978-3-540-72895-5_13>. Citado na página 59.

- HOSSIN, M.; SULAIMAN, M. A review on evaluation metrics for data classification evaluations. **International Journal of Data Mining & Knowledge Management Process**, Academy & Industry Research Collaboration Center (AIRCC), v. 5, n. 2, p. 1, 2015. Citado na página 71.
- HOTELLING, H. Relations between two sets of variates. **Biometrika**, [Oxford University Press, Biometrika Trust], v. 28, n. 3/4, p. 321–377, 1936. ISSN 00063444. Disponível em: <<http://www.jstor.org/stable/2333955>>. Citado na página 80.
- HU, W.; XIE, N.; LI, L.; ZENG, X.; MAYBANK, S. A survey on visual content-based video indexing and retrieval. **IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)**, v. 41, n. 6, p. 797–819, Nov 2011. ISSN 1558-2442. Citado na página 75.
- HUA, X.-S.; ZHANG, D.; LI, M.; ZHANG, H.-J. Performance evaluation protocol for video scene detection algorithms. In: **Workshop on Multimedia Information Retrieval, in conjunction with 10th ACM Multimedia**. Juan Les Pins, France: ACM, 2002. Citado na página 64.
- HUANG, J.; LIU, Z.; WANG, Y. Joint scene classification and segmentation based on hidden markov model. **IEEE Transactions on Multimedia**, v. 7, n. 3, p. 538–550, June 2005. ISSN 1941-0077. Citado na página 52.
- HUANG, P.-Y.; VAIBHAV; CHANG, X.; HAUPTMANN, A. G. Improving what cross-modal retrieval models learn through object-oriented inter- and intra-modal attention networks. In: **Proceedings of the 2019 on International Conference on Multimedia Retrieval**. New York, NY, USA: ACM, 2019. (ICMR '19), p. 244–252. ISBN 978-1-4503-6765-3. Disponível em: <<http://doi.acm.org/10.1145/3323873.3325043>>. Citado na página 143.
- HUBEL, D. H.; WIESEL, T. N. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. **The Journal of Physiology**, v. 160, n. 1, p. 106–154, 1962. Disponível em: <<https://physoc.onlinelibrary.wiley.com/doi/abs/10.1113/jphysiol.1962.sp006837>>. Citado na página 101.
- IONESCU, R. T.; POPESCU, M. **Knowledge Transfer Between Computer Vision and Text Mining: Similarity-based Learning Approaches**. 1st. ed. Switzerland: Springer Publishing Company, Incorporated, 2016. ISBN 3319303651, 9783319303659. Citado na página 43.
- IWAN, L. H.; THOM, J. A. Temporal video segmentation: detecting the end-of-act in circus performance videos. **Multimedia Tools and Applications**, v. 76, n. 1, p. 1379–1401, Jan 2017. ISSN 1573-7721. Disponível em: <<https://doi.org/10.1007/s11042-015-3130-3>>. Citado na página 40.
- IYENGAR, G.; LIPPMAN, A. B. Models for automatic classification of video sequences. In: **SPIE Proc. Storage and Retrieval for Image and Video Databases**. San Jose, CA, United States: SPIE, 1997. v. 3312, p. 216–227. Disponível em: <<https://doi.org/10.1117/12.298454>>. Citado na página 42.
- JACCARD, P. The distribution of the flora in the alpine zone. **New Phytologist**, v. 11, n. 2, p. 37–50, fev. 1912. Citado na página 68.

JHUO, I.-H.; YE, G.; GAO, S.; LIU, D.; JIANG, Y.-G.; LEE, D. T.; CHANG, S.-F. Discovering joint audio—visual codewords for video event detection. **Mach. Vision Appl.**, Springer-Verlag New York, Inc., Secaucus, NJ, USA, v. 25, n. 1, p. 33–47, jan. 2014. ISSN 0932-8092. Disponível em: <<http://dx.doi.org/10.1007/s00138-013-0567-0>>. Citado nas páginas 14, 28, 29, 46, 47, 56, 69, 75, 87, 88, 89, 95, 107, 113, 129, 130, 131, 137 e 138.

JIANG, H.; LIN, T.; ZHANG, H.-J. Video segmentation with the assistance of audio content analysis. In: **2000 IEEE International Conference on Multimedia and Expo. ICME2000. Proceedings. Latest Advances in the Fast Changing World of Multimedia (Cat. No.00TH8532)**. New York, USA: IEEE, 2000. v. 3, p. 1507–1510 vol.3. ISSN null. Citado na página 76.

JIANG, Y.; WU, Z.; TANG, J.; LI, Z.; XUE, X.; CHANG, S. Modeling multimodal clues in a hybrid deep learning framework for video classification. **IEEE Transactions on Multimedia**, v. 20, n. 11, p. 3137–3147, Nov 2018. ISSN 1520-9210. Citado nas páginas 24 e 25.

JIANG, Y.-G.; NGO, C.-W. Visual word proximity and linguistics for semantic video indexing and near-duplicate retrieval. **Computer Vision and Image Understanding**, v. 113, n. 3, p. 405 – 414, 2009. ISSN 1077-3142. Special Issue on Video Analysis. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1077314208001604>>. Citado na página 114.

JIANG, Y.-G.; YE, G.; CHANG, S.-F.; ELLIS, D.; LOUI, A. C. Consumer video understanding: A benchmark database and an evaluation of human and machine performance. In: **Proceedings of the 1st ACM International Conference on Multimedia Retrieval**. New York, NY, USA: ACM, 2011. (ICMR '11), p. 29:1–29:8. ISBN 978-1-4503-0336-1. Disponível em: <<http://doi.acm.org/10.1145/1991996.1992025>>. Citado nas páginas 56, 69 e 87.

JIN, L.; GAO, S.; LI, Z.; TANG, J. Hand-crafted features or machine learnt features? together they improve rgb-d object recognition. In: **2014 IEEE International Symposium on Multimedia**. Taichung, Taiwan: IEEE, 2014. p. 311–319. Citado na página 29.

KARPATHY, A.; FEI-FEI, L. Deep visual-semantic alignments for generating image descriptions. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 39, n. 4, p. 664–676, April 2017. Citado na página 143.

KENDER, J. R.; YEO, B. L. Video scene segmentation via continuous video coherence. In: **Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition**. Washington, DC, USA: IEEE Computer Society, 1998. (CVPR '98), p. 367–. ISBN 0-8186-8497-6. Disponível em: <<http://dl.acm.org/citation.cfm?id=794191.794801>>. Citado nas páginas 42 e 76.

KOPRINSKA, I.; CARRATO, S. Temporal video segmentation: A survey. **Signal Processing: Image Communication**, v. 16, n. 5, p. 477 – 500, 2001. ISSN 0923-5965. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0923596500000114>>. Citado nas páginas 24 e 39.

KUEHNE, H.; JHUANG, H.; GARROTE, E.; POGGIO, T.; SERRE, T. Hmdb: A large video database for human motion recognition. In: **2011 International Conference on**

Computer Vision. Barcelona, Spain: IEEE, 2011. p. 2556–2563. ISSN 2380-7504. Citado na página 69.

KURCIUS, J.; BRECKON, T. Using compressed audio-visual words for multi-modal scene classification. In: **Computational Intelligence for Multimedia Understanding (IWCIM), 2014 International Workshop on, 1-2 November 2014, Paris, France ; proceedings**. Paris: IEEE, 2014. p. 99–103. Disponível em: <<http://dro.dur.ac.uk/14361/>>. Citado na página 47.

KYPEROUNTAS, M.; KOTROPOULOS, C.; PITAS, I. Enhanced eigen-audioframes for audiovisual scene change detection. **IEEE Transactions on Multimedia**, v. 9, n. 4, p. 785–797, June 2007. ISSN 1520-9210. Citado nas páginas 60 e 81.

LAPTEV, I. On space-time interest points. **International Journal of Computer Vision**, v. 64, n. 2, p. 107–123, Sep 2005. ISSN 1573-1405. Disponível em: <<https://doi.org/10.1007/s11263-005-1838-7>>. Citado na página 88.

LAPTEV, I.; MARSZALEK, M.; SCHMID, C.; ROZENFELD, B. Learning realistic human actions from movies. In: **2008 IEEE Conference on Computer Vision and Pattern Recognition**. Anchorage, AK, USA: IEEE, 2008. p. 1–8. ISSN 1063-6919. Citado na página 88.

LATEEF, F.; RUICHEK, Y. Survey on semantic segmentation using deep learning techniques. **Neurocomputing**, v. 338, p. 321 – 348, 2019. ISSN 0925-2312. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S092523121930181X>>. Citado na página 68.

LAZEBNIK, S.; SCHMID, C.; PONCE, J. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: **2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)**. New York, NY, USA, USA: IEEE, 2006. v. 2, p. 2169–2178. ISSN 1063-6919. Citado na página 41.

LECUN, Y.; BENGIO, Y. The handbook of brain theory and neural networks. In: ARBIB, M. A. (Ed.). Cambridge, MA, USA: MIT Press, 1998. cap. Convolutional Networks for Images, Speech, and Time Series, p. 255–258. ISBN 0-262-51102-9. Citado nas páginas 41 e 82.

LEUNG, T.; MALIK, J. Representing and recognizing the visual appearance of materials using three-dimensional textons. **International Journal of Computer Vision**, v. 43, n. 1, p. 29–44, 2001. ISSN 1573-1405. Disponível em: <<http://dx.doi.org/10.1023/A:1011126920638>>. Citado na página 48.

LIN, T.; ZHANG, H.-J.; SHI, Q.-Y. Video scene extraction by force competition. In: **IEEE International Conference on Multimedia and Expo, 2001. ICME 2001**. Tokyo, Japan, Japan: IEEE, 2001. p. 753–756. ISSN null. Citado na página 77.

LIN, W.-H.; HAUPTMANN, A. News video classification using svm-based multimodal classifiers and combination strategies. In: **Proceedings of the Tenth ACM International Conference on Multimedia**. New York, NY, USA: ACM, 2002. (MULTIMEDIA '02), p. 323–326. ISBN 1-58113-620-X. Disponível em: <<http://doi.acm.org/10.1145/641007.641075>>. Citado nas páginas 42 e 56.

- LIU, D.; LAI, K.; YE, G.; CHEN, M.; CHANG, S. Sample-specific late fusion for visual category recognition. In: **2013 IEEE Conference on Computer Vision and Pattern Recognition**. Portland, OR, USA: IEEE, 2013. p. 803–810. ISSN 1063-6919. Citado na página 87.
- LIU, J.; YUAN, Z.; WANG, C. Towards good practices for multi-modal fusion in large-scale video classification. In: LEAL-TAIXÉ, L.; ROTH, S. (Ed.). **Computer Vision – ECCV 2018 Workshops**. Cham: Springer International Publishing, 2019. p. 287–296. ISBN 978-3-030-11018-5. Citado nas páginas 28 e 29.
- _____. Towards good practices for multi-modal fusion in large-scale video classification. In: LEAL-TAIXÉ, L.; ROTH, S. (Ed.). **Computer Vision – ECCV 2018 Workshops**. Cham: Springer International Publishing, 2019. p. 287–296. ISBN 978-3-030-11018-5. Citado na página 87.
- LLOYD, S. P. Least squares quantization in pcm. **IEEE Transactions on Information Theory**, v. 28, p. 129–137, 1982. Citado na página 111.
- LOKOČ, J.; BAILER, W.; SCHOEFFMANN, K.; MUENZER, B.; AWAD, G. On influential trends in interactive video retrieval: Video browser showdown 2015–2017. **IEEE Transactions on Multimedia**, v. 20, n. 12, p. 3361–3376, Dec 2018. ISSN 1520-9210. Citado na página 23.
- LOPES, B. L.; TROJAHN, T. H.; GOULARTE, R. Video scene detection by multimodal bag of features. **JIDM**, v. 5, n. 2, p. 194–205, 2014. Disponível em: <<https://seer.lcc.ufmg.br/index.php/jidm/article/view/632>>. Citado nas páginas 27, 46, 47 e 60.
- LOPEZ-PAZ, D.; SRA, S.; SMOLA, A. J.; GHAHRAMANI, Z.; SCHÖLKOPF, B. Randomized nonlinear component analysis. In: **Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32**. JMLR.org, 2014. (ICML'14), p. II–1359–II–1367. Disponível em: <<http://dl.acm.org/citation.cfm?id=3044805.3045044>>. Citado na página 81.
- LOWE, D. G. Distinctive image features from scale-invariant keypoints. **International Journal of Computer Vision**, v. 60, n. 2, p. 91–110, Nov 2004. ISSN 1573-1405. Disponível em: <<https://doi.org/10.1023/B:VISI.0000029664.99615.94>>. Citado nas páginas 43 e 88.
- LU, C.; DREW, M. S.; AU, J. Classification of summarized videos using hidden markov models on compressed chromaticity signatures. In: **Proceedings of the Ninth ACM International Conference on Multimedia**. New York, NY, USA: ACM, 2001. (MULTIMEDIA '01), p. 479–482. ISBN 1-58113-394-4. Disponível em: <<http://doi.acm.org/10.1145/500141.500217>>. Citado na página 42.
- MA, Z.; CHANG, X.; XU, Z.; SEBE, N.; G., A. Joint attributes and event analysis for multimedia event detection. **IEEE Transactions on Neural Networks and Learning Systems**, v. 29, n. 7, p. 2921–2930, July 2018. ISSN 2162-237X. Citado nas páginas 87 e 128.
- MARQUES, O.; FURHT, B. **Content-Based Image and Video Retrieval**. Springer US, 2002. (Multimedia Systems and Applications). ISBN 9781402070044. Disponível em: <<https://books.google.com.br/books?id=ds6xRLdlmJgC>>. Citado na página 41.

- MARSZALEK, M.; LAPTEV, I.; SCHMID, C. Actions in context. In: **2009 IEEE Conference on Computer Vision and Pattern Recognition**. Miami, FL, USA: IEEE, 2009. p. 2929–2936. ISSN 1063-6919. Citado na página 69.
- MARTINET, J.; SAYAD, I. E. Mid-level image descriptors. In: MA, Z. (Ed.). **Intelligent Multimedia Databases and Information Retrieval: Advancing Applications and Technologies**. USA: IGI Global, 2012. p. 46–60. Citado na página 41.
- MIKOLAJCZYK, K.; SCHMID, C. A performance evaluation of local descriptors. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 27, n. 10, p. 1615–1630, Oct 2005. ISSN 0162-8828. Citado na página 43.
- MIKOLOV, T.; SUTSKEVER, I.; CHEN, K.; CORRADO, G.; DEAN, J. Distributed representations of words and phrases and their compositionality. In: **Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2**. USA: Curran Associates Inc., 2013. (NIPS'13), p. 3111–3119. Disponível em: <http://dl.acm.org/citation.cfm?id=2999792.2999959>. Citado na página 82.
- MITROVIĆ, D.; HARTLIEB, S.; ZEPPELZAUER, M.; ZAHARIEVA, M. Scene segmentation in artistic archive documentaries. In: LEITNER, G.; HITZ, M.; HOLZINGER, A. (Ed.). **HCI in Work and Learning, Life and Leisure**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010. p. 400–410. ISBN 978-3-642-16607-5. Citado nas páginas 40 e 77.
- MÜNZER, B.; SCHOEFFMANN, K. Video browsing on a circular timeline. In: SCHOEFFMANN, K.; CHALIDABHONGSE, T. H.; NGO, C. W.; ARAMVITH, S.; O'CONNOR, N. E.; HO, Y.-S.; GABBOUJ, M.; ELGAMMAL, A. (Ed.). **MultiMedia Modeling**. Cham: Springer International Publishing, 2018. p. 395–399. ISBN 978-3-319-73600-6. Citado na página 24.
- NANNI, L.; GHIDONI, S.; BRAHNAM, S. Handcrafted vs. non-handcrafted features for computer vision classification. **Pattern Recognition**, v. 71, p. 158 – 172, 2017. ISSN 0031-3203. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0031320317302224>. Citado na página 29.
- NEEDLEMAN, S. B.; WUNSCH, C. D. A general method applicable to the search for similarities in the amino acid sequence of two proteins. **Journal of Molecular Biology**, v. 48, n. 3, p. 443 – 453, 1970. ISSN 0022-2836. Disponível em: <http://www.sciencedirect.com/science/article/pii/0022283670900574>. Citado na página 53.
- NGO, C.-W.; MA, Y.-F.; ZHANG, H.-J. Video summarization and scene detection by graph modeling. **IEEE Transactions on Circuits and Systems for Video Technology**, v. 15, n. 2, p. 296–305, Feb 2005. ISSN 1558-2205. Citado na página 77.
- NITANDA, N.; HASEYAMA, M.; KITAJIMA, H. Audio signal segmentation and classification for scene-cut detection. In: **2005 IEEE International Symposium on Circuits and Systems**. Kobe, Japan: IEEE, 2005. p. 4030–4033 Vol. 4. ISSN 2158-1525. Citado na página 79.
- OPPENHEIM, A. V.; SCHAFER, R. W. **Discrete-Time Signal Processing**. 3rd. ed. Upper Saddle River, NJ, USA: Prentice Hall Press, 2009. ISBN 0131988425, 9780131988422. Citado na página 46.

PARSHIN, V.; PARADZINETS, A.; CHEN, L. Multimodal data fusion for video scene segmentation. In: BRES, S.; LAURINI, R. (Ed.). **Visual Information and Information Systems**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006. p. 279–289. ISBN 978-3-540-32339-6. Citado na página [77](#).

POYNTON, C. A. **A Technical Introduction to Digital Video**. New York, NY, USA: John Wiley & Sons, Inc., 1996. ISBN 0-471-12253-X. Citado nas páginas [34](#) e [35](#).

RAO, K. S.; KOOLAGUDI, S. G. **Emotion Recognition Using Speech Features**. New York, USA: Springer Publishing Company, Incorporated, 2012. ISBN 1461451426, 9781461451426. Citado na página [113](#).

RASHEED, Z.; SHAH, M. Scene detection in hollywood movies and tv shows. In: **2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings**. Madison, WI, USA: IEEE, 2003. v. 2, p. II–343. ISSN 1063-6919. Citado nas páginas [55](#), [66](#) e [80](#).

_____. Detection and representation of scenes in videos. **IEEE Transactions on Multimedia**, v. 7, n. 6, p. 1097–1105, Dec 2005. ISSN 1941-0077. Citado na página [81](#).

RIJSBERGEN, C. J. V. **Information Retrieval**. 2nd. ed. Newton, MA, USA: Butterworth-Heinemann, 1979. ISBN 0408709294. Citado na página [64](#).

ROTHFUSS, D.; MÜNSTER, P.; ZIMMERMANN, G. Design guidelines for adaptable videos and video players on the web. In: BUCCHIANICO, G. D. (Ed.). **Advances in Design for Inclusion**. Cham: Springer International Publishing, 2019. p. 229–240. Citado na página [24](#).

ROTMAN, D.; PORAT, D.; ASHOUR, G. Robust and efficient video scene detection using optimal sequential grouping. In: **2016 IEEE International Symposium on Multimedia (ISM)**. San Jose, CA, USA: IEEE, 2016. p. 275–280. Citado na página [62](#).

_____. Robust video scene detection using multimodal fusion of optimally grouped features. In: **2017 IEEE 19th International Workshop on Multimedia Signal Processing (MMSP)**. Luton, UK: IEEE, 2017. p. 1–6. ISSN 2473-3628. Citado nas páginas [14](#), [17](#), [24](#), [27](#), [38](#), [39](#), [42](#), [47](#), [52](#), [60](#), [62](#), [76](#), [77](#), [79](#), [84](#), [85](#), [86](#), [90](#), [106](#), [113](#), [127](#) e [137](#).

ROTMAN, D.; PORAT, D.; ASHOUR, G.; BARZELAY, U. Optimally grouped deep features using normalized cost for video scene detection. In: **Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval**. New York, NY, USA: ACM, 2018. (ICMR '18), p. 187–195. ISBN 978-1-4503-5046-4. Citado nas páginas [77](#), [85](#), [90](#), [106](#), [120](#), [127](#), [128](#) e [137](#).

RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. Neurocomputing: Foundations of research. In: ANDERSON, J. A.; ROSENFELD, E. (Ed.). Cambridge, MA, USA: MIT Press, 1988. cap. Learning Representations by Back-propagating Errors, p. 696–699. ISBN 0-262-01097-6. Citado na página [119](#).

SALTON, G.; MCGILL, M. J. **Introduction to Modern Information Retrieval**. New York, NY, USA: McGraw-Hill, Inc., 1986. ISBN 0070544840. Citado na página [71](#).

SARACENO, C.; LEONARDI, R. Audio as a support to scene change detection and characterization of video sequences. In: **1997 IEEE International Conference on Acoustics, Speech, and Signal Processing**. Munich, Germany: IEEE, 1997. v. 4, p. 2597–2600 vol.4. ISSN 1520-6149. Citado nas páginas 24 e 40.

SARGIN, M. E.; YEMEZ, Y.; ERZIN, E.; TEKALP, A. M. Audiovisual synchronization and fusion using canonical correlation analysis. **IEEE Transactions on Multimedia**, v. 9, n. 7, p. 1396–1403, Nov 2007. ISSN 1941-0077. Citado na página 81.

SAVAGE, T. M.; VOGEL, K. **An Introduction to Digital Multimedia**. 1. ed. USA: Jones and Bartlett Publishers, Inc., 2008. ISBN 0763750522, 9780763750527. Citado nas páginas 33, 34, 36, 37 e 38.

SAWAI, K.; TAKAHASHI, T.; DEGUCHI, D.; IDE, I.; MURASE, H. Scene segmentation of wedding party videos by scenario-based matching with example videos. In: **Proceedings of the 19th ACM International Conference on Multimedia**. New York, NY, USA: ACM, 2011. (MM '11), p. 1545–1548. ISBN 978-1-4503-0616-4. Disponível em: <<http://doi.acm.org/10.1145/2072298.2072061>>. Citado na página 40.

SHARMA, D.; ALI, I. A modified mfcc feature extraction technique for robust speaker recognition. In: **2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI)**. Kochi, India: IEEE, 2015. p. 1052–1057. Citado na página 47.

SIDIROPOULOS, P.; MEZARIS, V.; KOMPATSIARIS, I.; MEINEDO, H.; BUGALHO, M.; TRANCOSO, I. Temporal video segmentation to scenes using high-level audiovisual features. **IEEE Trans. Cir. and Sys. for Video Technol.**, IEEE Press, Piscataway, NJ, USA, v. 21, n. 8, p. 1163–1177, ago. 2011. ISSN 1051-8215. Disponível em: <<http://dx.doi.org/10.1109/TCSVT.2011.2138830>>. Citado nas páginas 13, 17, 27, 38, 39, 42, 52, 55, 56, 60, 62, 77, 78, 79, 83, 85, 90, 95, 106, 114, 118 e 135.

SIDIROPOULOS, P.; MEZARIS, V.; KOMPATSIARIS, I.; KITTLER, J. Differential edit distance: A metric for scene segmentation evaluation. **IEEE Transactions on Circuits and Systems for Video Technology**, v. 22, n. 6, p. 904–914, June 2012. ISSN 1051-8215. Citado nas páginas 66, 67 e 68.

SIKKA, K.; SHARMA, G. Discriminatively trained latent ordinal model for video classification. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 40, n. 8, p. 1829–1844, Aug 2018. ISSN 0162-8828. Citado na página 46.

SIMO-SERRA, E.; TRULLS, E.; FERRAZ, L.; KOKKINOS, I.; FUA, P.; MORENO-NOGUER, F. Discriminative learning of deep convolutional feature point descriptors. In: **2015 IEEE International Conference on Computer Vision (ICCV)**. Santiago, Chile: IEEE, 2015. p. 118–126. Citado na página 120.

SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. In: BENGIO, Y.; LECUN, Y. (Ed.). **3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings**. ArXiv e-prints, 2015. Disponível em: <<http://arxiv.org/abs/1409.1556>>. Citado nas páginas 86, 120 e 136.

- SIVIC, J.; ZISSERMAN, A. Video google: a text retrieval approach to object matching in videos. In: **Proceedings Ninth IEEE International Conference on Computer Vision**. Nice, France, France: IEEE, 2003. p. 1470–1477 vol.2. Citado na página 41.
- SMEULDERS, A. W. M.; WORRING, M.; SANTINI, S.; GUPTA, A.; JAIN, R. Content-based image retrieval at the end of the early years. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 22, n. 12, p. 1349–1380, Dec 2000. ISSN 0162-8828. Citado nas páginas 24 e 40.
- SNOEK, C. G. M.; WORRING, M. A review on multimodal video indexing. In: **Proceedings. IEEE International Conference on Multimedia and Expo**. Lausanne, Switzerland: IEEE, 2002. v. 2, p. 21–24 vol.2. Citado nas páginas 25, 33 e 50.
- SNOEK, C. G. M.; WORRING, M.; SMEULDERS, A. W. M. Early versus late fusion in semantic video analysis. In: **Proceedings of the 13th Annual ACM International Conference on Multimedia**. New York, NY, USA: ACM, 2005. (MULTIMEDIA '05), p. 399–402. ISBN 1-59593-044-2. Citado na página 27.
- SOLTANIAN, M.; GHAEMMAGHAMI, S. Hierarchical concept score postprocessing and concept-wise normalization in cnn-based video event recognition. **IEEE Transactions on Multimedia**, v. 21, n. 1, p. 157–172, Jan 2019. ISSN 1941-0077. Citado nas páginas 87, 88 e 129.
- SOOMRO, K.; ZAMIR, A. R.; SHAH, M. **UCF101: A Dataset of 101 Human Actions Classes From Videos in The Wild**. 2012. Citado na página 69.
- STEVENS, S. S.; VOLKMANN, J.; NEWMAN, E. B. A scale for the measurement of the psychological magnitude pitch. **The Journal of the Acoustical Society of America**, v. 8, n. 3, p. 185–190, 1937. Disponível em: <<http://scitation.aip.org/content/asa/journal/jasa/8/3/10.1121/1.1915893>>. Citado na página 46.
- STOCKMAN, G.; SHAPIRO, L. G. **Computer Vision**. 1st. ed. Upper Saddle River, NJ, USA: Prentice Hall PTR, 2001. ISBN 0130307963. Citado na página 42.
- SUNDARAM, H.; CHANG, S.-F. Video scene segmentation using video and audio features. In: **2000 IEEE International Conference on Multimedia and Expo. ICME2000. Proceedings. Latest Advances in the Fast Changing World of Multimedia (Cat. No.00TH8532)**. New York, NY, USA: IEEE, 2000. v. 2, p. 1145–1148. Citado nas páginas 27 e 76.
- _____. Computable scenes and structures in films. **IEEE Transactions on Multimedia**, v. 4, n. 4, p. 482–491, Dec 2002. ISSN 1520-9210. Citado na página 40.
- SWAIN, M. J.; BALLARD, D. H. Color indexing. **International Journal of Computer Vision**, v. 7, n. 1, p. 11–32, Nov 1991. ISSN 1573-1405. Disponível em: <<https://doi.org/10.1007/BF00130487>>. Citado nas páginas 111 e 112.
- SZEGEDY, C.; VANHOUCKE, V.; IOFFE, S.; SHLENS, J.; WOJNA, Z. Rethinking the inception architecture for computer vision. In: **2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**. Las Vegas, NV, USA: IEEE, 2016. p. 2818–2826. Citado na página 85.

TAN, P.-N.; STEINBACH, M.; KUMAR, V. **Introduction to Data Mining, (First Edition)**. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2005. ISBN 0321321367. Citado na página 59.

TAPU, R.; ZAHARIA, T. A complete framework for temporal video segmentation. In: **2011 IEEE International Conference on Consumer Electronics -Berlin (ICCE-Berlin)**. Berlin, Germany: IEEE, 2011. p. 156–160. ISSN 2166-6814. Citado na página 46.

TAVANAPONG, W.; ZHOU, J. Shot clustering techniques for story browsing. **IEEE Transactions on Multimedia**, v. 6, n. 4, p. 517–527, Aug 2004. ISSN 1520-9210. Citado na página 52.

TIBSHIRANI, R.; WALTHER, G.; HASTIE, T. Estimating the number of clusters in a dataset via the gap statistic. v. 63, p. 411–423, 2000. Citado na página 84.

TOFFLER, A. **Future Shock**. Random House, 1970. Disponível em: <<https://books.google.com.br/books?id=-BhHAAAAMAAJ>>. Citado na página 23.

TROJAHN, T. H. **Um método de segmentação de vídeo em cenas baseado em aprendizagem profunda**. Tese (Doutorado) — Universidade de São Paulo, São Carlos, Agosto 2019. Citado nas páginas 119, 121, 125, 126 e 136.

TROJAHN, T. H.; GOULARTE, R. Video scene segmentation by improved visual shot coherence. In: **Proceedings of the 19th Brazilian Symposium on Multimedia and the Web**. New York, NY, USA: ACM, 2013. (WebMedia '13), p. 23–30. ISBN 978-1-4503-2559-2. Disponível em: <<http://doi.acm.org/10.1145/2526188.2526206>>. Citado nas páginas 120 e 134.

TUYTELAARS, T.; MIKOLAJCZYK, K. Local invariant feature detectors: A survey. **Found. Trends. Comput. Graph. Vis.**, Now Publishers Inc., Hanover, MA, USA, v. 3, n. 3, p. 177–280, jul. 2008. ISSN 1572-2740. Disponível em: <<http://dx.doi.org/10.1561/06000000017>>. Citado na página 42.

TZELEPIS, C.; MA, Z.; MEZARIS, V.; IONESCU, B.; KOMPATSIARIS, I.; BOATO, G.; SEBE, N.; YAN, S. Event-based media processing and analysis: A survey of the literature. **Image and Vision Computing**, v. 53, p. 3 – 19, 2016. ISSN 0262-8856. Event-based Media Processing and Analysis. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S026288561630083X>>. Citado na página 87.

UURTIO, V.; BHADRA, S.; ROUSU, J. Large-scale sparse kernel canonical correlation analysis. In: CHAUDHURI, K.; SALAKHUTDINOV, R. (Ed.). **Proceedings of the 36th International Conference on Machine Learning**. Long Beach, California, USA: PMLR, 2019. (Proceedings of Machine Learning Research, v. 97), p. 6383–6391. Disponível em: <<http://proceedings.mlr.press/v97/uurtio19a.html>>. Citado na página 81.

VEDALDI, A.; GULSHAN, V.; VARMA, M.; ZISSERMAN, A. Multiple kernels for object detection. In: **2009 IEEE 12th International Conference on Computer Vision**. Kyoto, Japan: IEEE, 2009. p. 606–613. Citado na página 131.

VENDRIG, J.; WORRING, M. Systematic evaluation of logical story unit segmentation. **IEEE Transactions on Multimedia**, v. 4, n. 4, p. 492–499, Dec 2002. ISSN 1520-9210. Citado na página 66.

VITERBI, A. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. **IEEE Transactions on Information Theory**, v. 13, n. 2, p. 260–269, April 1967. ISSN 1557-9654. Citado na página 53.

WANG, X.; WANG, S.; CHEN, H.; GABBOUJ, M. A shot clustering based algorithm for scene segmentation. In: **2007 International Conference on Computational Intelligence and Security Workshops (CISW 2007)**. Heilongjiang, China: IEEE, 2007. p. 259–252. ISSN null. Citado na página 77.

WATKINSON, J. **The Art of Digital Video, Fourth Edition**. USA: Focal Press, 2007. ISBN 024052005X. Citado na página 34.

WENG, C.; CHU, W.; WU, J. Rolenet: Movie analysis from the perspective of social networks. **IEEE Transactions on Multimedia**, v. 11, n. 2, p. 256–271, Feb 2009. ISSN 1941-0077. Citado na página 77.

WENGANG, C.; DE, X. A novel approach of generating video scene structure. In: **Conference on Convergent Technologies for Asia-Pacific Region**. Bangalore, India, India: IEEE, 2003. v. 1, p. 350–353 Vol.1. ISSN null. Citado na página 77.

WIATOWSKI, T.; BÖLCSKEI, H. A mathematical theory of deep convolutional neural networks for feature extraction. **IEEE Transactions on Information Theory**, v. 64, n. 3, p. 1845–1866, March 2018. Citado na página 119.

WILSON, K. W.; DIVAKARAN, A. Discriminative genre-independent audio-visual scene change detection. In: SCHETTINI, R.; JAIN, R. C.; SANTINI, S. (Ed.). **Multimedia Content Access: Algorithms and Systems III**. SPIE, 2009. v. 7255, p. 1 – 8. Disponível em: <<https://doi.org/10.1117/12.805624>>. Citado na página 79.

WU, S.; JIN, M. Study on a new video scene segmentation algorithm. **Applied Mathematics and Information Sciences**, v. 9, n. 1, p. 361–368, 2015. Cited By 0. Disponível em: <<https://www.scopus.com/inward/record.uri?eid=2-s2.0-84907246427&partnerID=40&md5=dd07505c1071cd1603e5206c25e41311>>. Citado na página 62.

WU, Z.; JIANG, Y.-G.; WANG, J.; PU, J.; XUE, X. Exploring inter-feature and inter-class relationships with deep neural networks for video classification. In: **Proceedings of the 22Nd ACM International Conference on Multimedia**. New York, NY, USA: ACM, 2014. (MM '14), p. 167–176. ISBN 978-1-4503-3063-3. Disponível em: <<http://doi.acm.org/10.1145/2647868.2654931>>. Citado na página 87.

XU, S.; FENG, B.; DING, P.; XU, B. Graph-based multi-modal scene detection for movie and teleplay. In: **2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)**. Kyoto, Japan: IEEE, 2012. p. 1413–1416. ISSN 2379-190X. Citado na página 55.

XU, S.; FENG, B.; XU, B. Temporal video segmentation to scene based on conditional random fields. In: LI, S.; SADDIK, A. E.; WANG, M.; MEI, T.; SEBE, N.; YAN, S.; HONG, R.; GURRIN, C. (Ed.). **Advances in Multimedia Modeling**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013. p. 374–384. ISBN 978-3-642-35728-2. Citado na página 28.

- XU, Z.; SU, L.; WANG, S.; HUANG, Q.; ZHANG, Y. S2l: Single-streamline for complex video event detection. In: **2018 IEEE International Conference on Multimedia Expo Workshops (ICMEW)**. San Diego, CA, USA: IEEE, 2018. p. 1–6. Citado na página 87.
- YANG, L.; WANG, Y.; DUNNE, D.; SOBOLEV, M.; NAAMAN, M.; ESTRIN, D. More than just words: Modeling non-textual characteristics of podcasts. In: **Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining**. New York, NY, USA: ACM, 2019. (WSDM '19), p. 276–284. ISBN 978-1-4503-5940-5. Disponível em: <<http://doi.acm.org/10.1145/3289600.3290993>>. Citado na página 46.
- YAŞAROĞLU, Y.; ALATAN, A. A. Summarizing video: Content, features, and hmm topologies. In: GARCÍA, N.; SALGADO, L.; MARTÍNEZ, J. M. (Ed.). **Visual Content Processing and Representation**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003. p. 101–110. ISBN 978-3-540-39798-4. Citado na página 52.
- YEUNG, M.; YEO, B.-L.; LIU, B. Segmentation of video by clustering and graph analysis. **Comput. Vis. Image Underst.**, Elsevier Science Inc., New York, NY, USA, v. 71, n. 1, p. 94–109, jul. 1998. ISSN 1077-3142. Disponível em: <<http://dx.doi.org/10.1006/cviu.1997.0628>>. Citado nas páginas 52, 54, 76 e 113.
- YU, S. X.; SHI, J. Grouping with bias. In: **Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic**. Cambridge, MA, USA: MIT Press, 2001. (NIPS'01), p. 1327–1334. Disponível em: <<http://dl.acm.org/citation.cfm?id=2980539.2980711>>. Citado na página 83.
- ZHAI, Y.; SHAH, M. Video scene segmentation using markov chain monte carlo. **IEEE Transactions on Multimedia**, v. 8, n. 4, p. 686–697, Aug 2006. ISSN 1520-9210. Citado na página 117.
- ZHANG, H.; KANKANHALLI, A.; SMOLIAR, S. W. Automatic partitioning of full-motion video. **Multimedia Systems**, v. 1, n. 1, p. 10–28, Jan 1993. ISSN 1432-1882. Disponível em: <<https://doi.org/10.1007/BF01210504>>. Citado na página 42.
- ZHANG, J.; MEI, K.; WANG, X.; ZHENG, Y.; FAN, J. From text to video: Exploiting mid-level semantics for large-scale video classification. In: **2018 24th International Conference on Pattern Recognition (ICPR)**. Beijing, China: IEEE, 2018. p. 1695–1700. ISSN 1051-4651. Citado nas páginas 69 e 87.
- ZHAO, Z.; SONG, Y.; SU, F. Specific video identification via joint learning of latent semantic concept, scene and temporal structure. **Neurocomput.**, Elsevier Science Publishers B. V., Amsterdam, The Netherlands, The Netherlands, v. 208, n. C, p. 378–386, out. 2016. ISSN 0925-2312. Disponível em: <<http://dx.doi.org/10.1016/j.neucom.2016.06.002>>. Citado na página 87.
- ZHOU, X.; ZHUANG, X.; YAN, S.; CHANG, S.-F.; HASEGAWA-JOHNSON, M.; HUANG, T. S. Sift-bag kernel for video event analysis. In: **Proceedings of the 16th ACM International Conference on Multimedia**. New York, NY, USA: ACM, 2008. (MM '08), p. 229–238. ISBN 978-1-60558-303-7. Citado na página 46.

GLOSSÁRIO

Application Programming Interface conjunto de funções, protocolos e ferramentas que funcionam como peças para a composição de um programa.

aural algo que é relativo à audição.

Big Data conjunto de dados que não pode ser tratado pelos aplicativos de processamento de dados tradicionais em tempo tolerável.

cluster computacional sistema computacional composto por múltiplos nós que trabalham em conjunto.

mídia meio usado para representar, armazenar, transmitir ou apresentar informação.

open content conteúdo de trabalho criativo que pode ser copiado ou modificado sem que haja necessidade de solicitação de permissão.

pipeline cadeia de elementos de processamento organizados sequencialmente tais que a saída de um elemento é fornecida como entrada ao próximo elemento.

script sequência de instruções definidas para a execução de uma tarefa.

segmento de vídeo subunidade temporal de vídeo. Nesta tese, um segmento de vídeo é uma sequência de quadros provenientes de um vídeo.

smartphone aparelho de telefonia móvel integrado a um sistema computacional móvel.

transdutor dispositivo que converte energia de uma determinada natureza, em energia de natureza distinta.

INFORMAÇÕES TÉCNICAS SOBRE OS VÍDEOS DA BASE CONFIÁVEL OVSD

Este apêndice contém os aspectos técnicos dos arquivos de vídeos da base confiável OVSD *Dataset*. Tais aspectos são apresentados na Tabela 18.

Tabela 18 – Aspectos técnicos dos arquivos de vídeo da base confiável OVSD *Dataset*

Vídeo	Visual			Aural		
	Codec	Resolução Espacial	Taxa de Reprodução	Codec	Canais de Áudio	Taxa de Amostragem
1000 Days	H.264	1920 × 1080	25	MPEG-4 AAC	2	48000
Big Buck Bunny	MPEG-4 Video	854 × 480	24	AC-3	5	48000
Boy Who Never Slept	H.264	524 × 360	30	MPEG-4 AAC	2	44100
CH7	H.264	640 × 360	25	MPEG-4 AAC	2	44100
Cosmos Laundromat	H.264	1920 × 804	24	MPEG-4 AAC	2	44100
Elephants Dream	DivX MPEG-4 V4	1024 × 576	24	AC-3	5	48000
Fires Beneath Water	DivX MPEG-4 V4	720 × 528	30	MPEG-1 Layer 3	2	48000
Honey	H.264	480 × 216	30	MPEG-4 AAC	2	44100
Jathias Wager	H.264	1280 × 720	30	MPEG-4 AAC	2	44100
La Chute D Une Plume	H.264	1264 × 720	25	MPEG-4 AAC	2	48000
Lord Meia	MPEG-4 Video	720 × 540	30	MPEG-1 Layer 3	2	48000
Meridian	H.264	1280 × 720	30	MPEG-4 AAC	2	44100
Oceania	MPEG-4 Video	720 × 416	24	MPEG-1 Layer 3	2	48000
Pentagon	ITU H.264	1024 × 576	25	MPEG-1 Layer 3	2	48000
Route 66	MPEG-4 Video	640 × 432	24	MPEG-1 Layer 3	2	44100
Seven Dead Men	H.264	640 × 360	24	MPEG-4 AAC	2	44100
Sintel	H.264	1024 × 436	24	MPEG-4 AAC	5	48000
Sita Sings	H.264	640 × 360	30	MPEG-4 AAC	2	44100
Star Wreck	H.264	640 × 304	25	MPEG-4 AAC	2	44100
Tears of Steel	H.264	1920 × 800	24	MPEG-1 Layer 3	2	44100
Valkaama	H.264	1920 × 1080	25	MPEG-4 AAC	2	48000