



Francisco Miguel Moreira Miranda Melo Tavares

Identificação de Conteúdos de Vídeo baseada na Análise de Textura e Movimento

Dissertação apresentada para cumprimento dos requisitos necessários à obtenção do
grau de Mestre em Engenharia Electrotécnica e de Computadores

Julho, 2017



UNIVERSIDADE DE COIMBRA



FCTUC FACULDADE DE CIÊNCIAS
E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

Identificação de Conteúdos de Vídeo baseada na Análise de Textura e Movimento

Francisco Miguel Moreira Miranda Melo Tavares

Coimbra, Julho de 2017



Identificação de Conteúdos de Vídeo baseada na Análise de Textura e Movimento

Orientador:

Prof. Doutor Paulo José Monteiro Peixoto

Co-Orientador:

Prof. Doutor Luís Alberto da Silva Cruz

Júri:

Prof. Doutor Jorge Manuel Moreira de Campos Pereira Batista

Prof. Doutor João Pedro de Almeida Barreto

Prof. Doutor Paulo José Monteiro Peixoto

Dissertação apresentada à Universidade de Coimbra, para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia Electrotécnica e de Computadores.

Coimbra, Julho de 2017

Agradecimentos

Primeiramente gostaria de agradecer aos meus orientadores Professor Paulo Peixoto e Professor Luis Cruz pela confiança que depositaram em mim. Esta dissertação seria impossível sem o apoio, experiência e dedicação deles.

Um agradecimento especial à empresa WeTek que proporcionou todo este projecto, garantindo o material necessário para a sua elaboração. Ao Engenheiro Pablo Silva pelo acompanhamento e motivação.

Ao Instituto de Telecomunicações que me acolheu e forneceu as melhores condições para trabalhar na investigação desta dissertação e, aos meus colegas de laboratório pela boa disposição e ambiente de cooperação.

A todos os meus amigos que partilharam comigo estes anos académicos, pelos bons momentos vividos, pelo companheirismo e inspiração que levo comigo para a vida.

Por fim, expresso o meu maior agradecimento à minha família, principalmente aos meus pais e ao meu irmão, por todo o apoio que me proporcionaram, possibilitando a presença e conclusão deste curso.

Resumo

Os conteúdos de vídeo são imprescindíveis nos dias de hoje. Apresentam-se como uma ferramenta poderosa de marketing e são uma fonte de informação rápida, de entretenimento e cultura.

No contexto da televisão, tem-se assistido a grandes desenvolvimentos, no que diz respeito à digitalização de conteúdos através de set-top box (STB) e novas funcionalidades têm-se juntado como a gravação de conteúdos (DVR) e outras aplicações.

A presente dissertação propõe identificar conteúdos de vídeo, mais concretamente, perceber se é programa ou publicidade, através de técnicas de visão por computador, processamento de imagem e descritores MPEG-7.

Do ponto de vista de deteção de publicidade, foram identificados e filtrados por classes os separadores de vídeo de cada canal e armazenados numa base de dados. O conteúdo guardado não se encontra sob a forma de imagem, porém através de atributos de cada método considerado, por exemplo, descritores ou *bins*, o que permite uma eficiente indexação e correspondência dos dados.

Da mesma forma, é necessário ter conhecimento sobre o programa. Há características que se destacam como a presença de um logo, com diferentes formas, cores e texturas, os logos identificam a estação e, por vezes, o tipo de conteúdo.

Este trabalho, propõe detectar e reconhecer de forma automática a presença de logos e outros objectos que permaneçam ao longo da transmissão, apresentando vários métodos para cada tipo de logo.

O sistema distingue-se pela versatilidade do algoritmo ao regular os parâmetros, consoante as características do ambiente e finalidade desejada, produzindo resultados mais satisfatórios.

Em todos os casos de teste, os resultados obtidos demonstram que estes métodos propostos são capazes de identificar correctamente os conteúdos de vídeo presentes em televisão. Serão apresentadas as características mais benéficas bem como a performance computacional

de cada método.

Todos os algoritmos propostos foram desenvolvidos e testados em MATLAB e os *frames* extraídos através do FFMPEG.

Palavras-chave: Identificação de vídeo, Descritores MPEG-7, Detecção automática de Programas de Televisão, Similaridade de imagens

Abstract

Nowadays, video contents are indispensable in our lives. This presents itself as a powerful marketing tool and is a source of quick information, entertainment and culture.

In the scope of television a great development has been achieved, in terms of digital contents through the set-top box and new functionalities are included as video recording and other video applications.

The present thesis proposes identify video contents on television, more precisely, understand if the contents are commercial or tv shows, across computer vision techniques, image processing and MPEG-7 descriptors.

From the point of view of advertising contents, were identifying and filtered by classes the video signatures of each tv channel and then stored in a database. The data stored in the database, it's not under image type but through bins or descriptors, providing an efficient analysis.

Furthermore, it's required to have knowledge about tv shows. There are evident features like a logo presence, with different colors, texture or shapes. These properties identify the current channel and content type.

This work proposes automatically detect and recognize logos and other objects throughout the broadcast, presenting several methods for each type of logos.

The system stands out for the versatility to manage input arguments, according to the features and purpose desired, producing better results for the end user.

In all cases of test, the results obtained prove that all methods proposed are able to correctly identify video contents on television. In detail, the main features and power computational from each method will be presented.

All proposed algorithms were developed and tested under MATLAB and their frames extracted across FFMPEG.

Keyword: Digital Video Content Identification, Background Subtraction, Similarity retrieval

eval in image database, Image segmentation, Automatic detection of TV show

"No great discovery was ever made without a bold guess"

— Isaac Newton

Conteúdo

Agradecimentos	iii
Resumo	v
Abstract	viii
Lista de Acrónimos	xv
Lista de Figuras	xvii
Lista de Tabelas	xix
1 Introdução	1
1.1 Motivação e contexto	1
1.2 Objectivos	2
1.3 Contribuições e Implementações	3
1.4 Descrição da estrutura da tese	3
2 Estado da arte	5
3 Métodos propostos	7
3.1 Deteção e Reconhecimento de Logos	8
3.1.1 Logos Opacos	8
3.1.2 Logos Semi-transparentes	10
3.2 Reconhecimento de Separadores de Vídeo	12
3.2.1 Descritor de Cor Dominante	12
3.2.2 Descritor de Layout de Cor	14
3.2.3 Histograma de cor	15

4	Procedimento experimental	17
4.1	Detecção e Reconhecimento de logos	17
4.1.1	Logos Opacos	18
4.1.2	Logos Semi-transparentes	19
4.2	Reconhecimento de Separadores de Vídeo	23
5	Testes e Resultados	25
5.1	Detecção e Reconhecimento de Logos	26
5.1.1	Logos Opacos	27
5.1.2	Logos Semi-transparentes	29
5.2	Reconhecimento de Separadores	32
5.2.1	Descritor de Layout de Cor	32
5.2.2	Histograma de cor	33
5.2.3	Descritor de Cor Dominante	34
5.2.4	Análise Crítica	36
6	Conclusão e Trabalho Futuro	37
6.1	Conclusão	37
6.2	Trabalho Futuro	37
7	Bibliografia	39

Lista de Acrónimos

BS	Background Subtraction
CLD	Color Layout Descriptor
CH	Color Histogram
DB	Database
DTV	Digital Television
DVR	Digital Video Recorder
DCT	Discrete Cousine Transform
DCD	Dominant Color Descriptor
EPG	Electronic Programming Guide
FPS	Frames per second
HD	High Definition
HSV	Hue, Saturation and Value
IS	Image Segmentation
IPTV	Internet Protocol Television
JPEG	Joint Photographic Experts Group
MPEG	Moving Picture Experts Group
MPEG-TS	MPEG Transport Stream
MPEG-7	Multimedia Content Description Interface
MPEG7d	MPEG-7 Descriptors

OpenCV	Open Source Computer Vision Library
PNG	Portable Network Graphics
RGB	Red, Green and Blue
SD	Standard Definition
STB	Set-top Box
SURF	Speeded up robust features
TV	Television
VoD	Video on demand
YCbCr	Luminance, Blue chrominance, Red chrominance

Lista de Figuras

1.1	Visão Geral do Sistema	3
3.1	Estrutura simplificada do conteúdo transmitido por TV	8
3.2	Logo opaco da RTP1	9
3.3	Logo opaco da SIC	9
3.4	Letterbox	10
3.5	Conteúdo 4:3 em 16:9	10
3.6	Logo semi-transparente da Eurosport 1	11
3.7	Logo semi-transparente da Odisseia	11
3.8	Estrutura da base de dados de separadores	13
3.9	Imagem representativa de um separador do canal AXN	14
3.10	Separador dividido em blocos de $n = 16$ e respectiva média	14
3.11	Processo de extração do CLD	15
4.1	Background do logo AXN Black HD	19
4.2	Variância com aplicação do filtro no logo AXN Black HD	19
4.3	Histograma de linhas do Background	19
4.4	Background do logo FOX Crime	19
4.5	Variância com aplicação do filtro no logo FOX Crime HD	19
4.6	Histograma de colunas do Background	19
4.7	Background do logo opaco da Sport TV 1	19
4.8	Variância do logo opaco da Sport TV 1	19
4.9	Máscara do logo opaco da Sport TV 1	19
4.10	Apresentação de um logo semi-transparente visível	21
4.11	Apresentação de um logo semi-transparente pouco visível	21
4.12	Representação de um degrade de cinza	21
4.13	Representação do Mínimo Brilho	22

4.14	Máscara do logo semi-transparente	22
4.15	<i>Prewitt</i> aplicado ao logo semi-transparente	22
4.16	Logo semi-transparente detectado	22
4.17	Logo semi-transparente reconhecido	22
4.18	Interseção de logo detectado e reconhecido	22
4.19	Representação de um separador da FOX Life por classes	24

Lista de Tabelas

5.1	Valores de configuração do algoritmo de logoss opacos	27
5.2	Resultados para o Teste 1	27
5.3	Resultados para Teste 2	28
5.4	Valores de configuração do método de logoss semi-transparentes	29
5.5	Resultados para o Teste 1	30
5.6	Resultados para o Teste 2	31
5.7	Resultados para Threshold = 10	33
5.8	Resultados para Threshold = 20	33
5.9	Valores de configuração do método Histograma de cor	33
5.10	Resultados para o Teste 1	34
5.11	Resultados para o Teste 2	34
5.12	Paramêtross de configuração do método Descritoss de Cor Dominante	35
5.13	Resultados do Descritoss de Cor Dominante para o Teste 1	35
5.14	Resultados do Descritoss de Cor Dominante para o Teste 2	36

1 Introdução

Neste capítulo, será feita uma introdução à dissertação. Começando por explicar os motivos que levaram à execução deste trabalho, bem como o avanços que têm sido feitos na área da multimédia.

Para além disso, serão apresentados os principais objectivos deste trabalho e quais as contribuições realizadas para a identificação de conteúdos de vídeo.

Por fim, será apresentada a estrutura global desta dissertação, resumindo o conteúdo de cada capítulo.

Alguns termos em inglês serão indicados em *itálico*.

1.1 Motivação e contexto

A iniciativa deste projecto vem no seguimento de uma colaboração com a WeTek, uma empresa dedicada à investigação e criação de soluções de multimédia. Reconhecendo na WeTek uma empresa inovadora, este projecto distingue-se por uma ambição conjunta de contribuir para a criação de novas funcionalidades com potencial para o actual serviço de TV.

O avanço das tecnologias digitais na área da multimédia têm revolucionado não só a produção e distribuição de conteúdos como também a necessidade de interpretar eficientemente os programas de televisão, permitindo que os gestores de conteúdos criem aplicações para os utilizadores.

No contexto atual de TV, os conteúdos de vídeo são transmitidos através de set-top box (STB). A capacidade de processamento que este tipo de equipamentos possuem, permite que sejam incorporadas funcionalidades extra que enriquecem o sistema, proporcionando ao utilizador melhor experiência de utilização e uma melhor satisfação com o serviço.

No que diz respeito à preferência dos clientes de TV, observa-se um desinteresse por parte dos conteúdos de publicidade. Desse ponto de vista, este reconhecimento, evita que

visualizem conteúdos que não são do seu interesse. Por outro lado, as agências encontram na publicidade uma estratégia poderosa de marketing para difundir os seus produtos e segmentar as visualizações com base no público alvo de cada programa. Dessa forma, este reconhecimento, permite controlar se os seus anúncios estão a ser transmitidos conforme contratados.

Nesse sentido, é fundamental identificar correctamente quando ocorrem as transições e os momentos de publicidade e/ou programa.

1.2 Objectivos

Com esta dissertação proponho desenvolver métodos baseados em segmentação de imagem (IS) e descritores MPEG-7 (MPEG7d) para a detecção automática de publicidade e programas televisivos, recorrendo a análise de vídeo.

Evidencio a importância deste reconhecimento, para os sistemas de gravação DVR, evitando um desperdício no espaço de armazenamento de conteúdos, na sincronização do guia de TV (EPG) bem como na identificação de um anúncio em particular o que possibilita aos anunciantes de publicidade verificarem se os seus anúncios estão a ser publicados conforme contratado. Para além disso, um sistema que identifique automaticamente a ocorrência de publicidade poderá ser utilizado para substituí-la por versões customizadas dessa mesma publicidade, de acordo com as preferências do utilizador.

Os métodos desenvolvidos foram testados offline, mais concretamente, realizada a extração das *frames* do vídeo, a etiquetagem manual com base no tipo de conteúdo ser programa, publicidade ou separador e, por fim, foram aplicados os algoritmos desenvolvidos.

O presente diagrama 1.1 ilustra uma perspectiva da estrutura do trabalho.

Realço o interesse da amostragem conter um número significativo de canais, para poder ter um sistema robusto que possa ser aplicado de forma automática, em qualquer canal.

Os métodos propostos nesta tese têm como objectivo:

- A detecção e reconhecimento de logos opacos e semi-transparentes.
- O reconhecimento de separadores de entrada e saída de um programa.

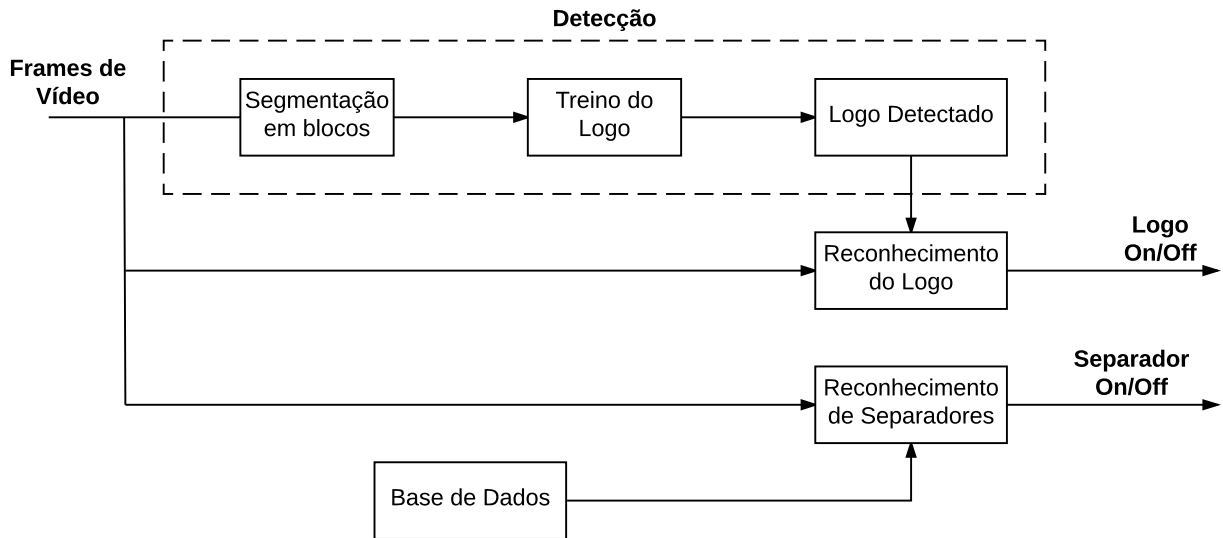


Figura 1.1: Visão Geral do Sistema

1.3 Contribuições e Implementações

Nesta tese são apresentadas duas contribuições principais, relativas à identificação de vídeo, a detecção e reconhecimento da presença de logos e o reconhecimento de separadores de entrada e saída de um programa.

No âmbito da detecção da presença de logos, foram desenvolvidos métodos que aprendem a reconhecer características que se destacam ao longo da transmissão. Estas técnicas carecem de um tempo mínimo de treino, sendo que o algoritmo pode, de forma automática, incrementar esse tempo caso entenda que não é suficiente para ter um bom *template*. O reconhecimento é iniciado automaticamente logo após o treino do logo.

No domínio do reconhecimento de separadores de vídeo, exploraram-se abordagens que, armazenam, caracterizam e compararam, de forma eficiente, a informação. Como é o caso de descritores visuais de MPEG-7 e histogramas de cor.

1.4 Descrição da estrutura da tese

A presente dissertação encontra-se estruturada da seguinte forma: o Capítulo 2 apresenta o trabalho relacionado com identificação de vídeo, nomeadamente as técnicas usadas, até ao momento, para deteção de publicidade e programas, seguido da demonstração dos métodos

propostos na literatura no Capítulo 3. O Capítulo 4 explica detalhadamente os procedimentos experimentais de cada método bem como destaca algumas soluções para alguns problemas encontrados. Os resultados obtidos dos testes laboratoriais serão apresentados no Capítulo 5, explicando as condições em que foram realizados. Por fim, o Capítulo 6 encerrará o documento, com as conclusões sobre o trabalho realizado e apresentação de soluções e melhorias para um trabalho futuro.

2 Estado da arte

Este capítulo introduz os fundamentos sobre identificação de conteúdo de vídeo e as técnicas usadas para a detecção de anúncios e programas de televisão. Serão apresentados artigos e trabalhos académicos que servem de base à dissertação.

Numa primeira análise, foi importante perceber as características dos anúncios para televisão de forma a poder perceber como poderão ser reconhecidos. Através de [1, 2, 3, 4] identificou-se que os intervalos temporais dos anúncios são curtos e para captar a atenção dos utilizadores possuem uma elevada frequência de corte e número de ações. Tipicamente, são também delimitados por tramas pretas e silenciosas. Para além disso, existem pedaços de anúncios e separadores que se repetem inúmeras vezes, o que potencia o interesse em ter uma base de dados [5, 6] que armazena essa informação. Na realidade, algumas destas características como a delimitação de tramas pretas entre anúncios, não ocorre com regularidade.

No que diz respeito à detecção de anúncios de publicidade, o detector Comskip em [7] é uma aplicação que permite que seja analisado o conteúdo de um ficheiro MPEG ou H264, podendo ser configurado um elevado número de parâmetros, devolvendo um ficheiro com a localização dos anúncios. Esses parâmetros, podem ser, por exemplo, o tempo mínimo e máximo de um anúncio, o máximo brilho para uma trama ser considerada preta e o volume máximo permitido nesses casos, o intervalo de tempo inferior e superior a manter após detecção ou o tempo máximo de treino do logo permitido.

A tecnologia MPEG-7 [8] inclui os modernos desenvolvimentos em pesquisa e recuperação de multimédia. Uma panorâmica mais completa sobre as suas funcionalidades é apresentada em [9]. Esta tecnologia de descrição de conteúdo abrange um quantidade de descritores visuais que são aplicados nesta dissertação. Algumas abordagens de detecção e extração destes descritores, foram identificadas em [10, 11, 12] com o objectivo de encontrar similaridade entre imagens. Para além destes descritores, um estudo baseado em histogramas de cor proposto em [13, 14], demonstra ser uma solução eficaz, para comparar duas imagens.

Considerando este problema como sendo binário, ou seja, o conteúdo de vídeo apenas pode ser classificado como programa ou publicidade é importante perceber também as características dos programas. Praticamente todas as abordagens baseiam-se na detecção do logo, como objecto que se destaca em toda a transmissão. Sabe-se também que os programas correm durante um período maior, que podem conter legendas e outros objectos que se sobressaem ao longo do tempo como, por exemplo, a *bolinha vermelha*, uma legenda indicativa de resultados, normalmente presente em eventos desportivos ou também um logo identificativo do programa.

Segmentar uma imagem em regiões de interesse é bastante conveniente porque a zona de extração já se encontra filtrada e dessa forma podem ser comparadas as características apenas nas superfícies determinadas.

A identificação de logos foi explorada por [15, 16]. Neste projecto, para a detecção de logos opacos, foi usada uma abordagem probabilística em [17] conhecida por encontrar objectos em movimento em sequências de vídeo. O problema seguinte foi tentar encontrar a melhor maneira de comparar o que foi detectado com o que foi extraído da imagem. A solução encontrada visa efectuar uma comparação puramente baseada na cor. No contexto dos logos semi-transparentes, foi explorada uma técnica baseada em redes neuronais pela France Telecom [18]. Algumas abordagens sugerem também o uso de detector de cantos para a detecção e reconhecimento deste tipo de objectos.

Em alguns trabalhos foram considerados descritores SURF [19, 20] como algoritmo de extração de recursos locais numa imagem, embora sabendo que é mais eficiente que o descritor SIFT, não deixa de ser uma solução mais pesada computacionalmente, em comparação com as indicadas acima.

Como suporte mais completo sobre processamento de imagem e vídeo foi consultado em [21]. Para além disso, a obra [22] apresenta uma análise detalhada ao conteúdo de TV, nomeadamente programas e anúncios de publicidade, explorando técnicas de extração e estruturação do conteúdo e as suas aplicações.

3 Métodos propostos

Neste capítulo serão abordados os métodos utilizados na identificação de conteúdo de vídeo. As técnicas apresentadas são baseadas em segmentação de imagem e descritores de cor MPEG-7, ignorando o áudio. Visam automatizar o reconhecimento de programas televisivos bem como os momentos de publicidade.

A segmentação de imagem é um processo de particionamento de uma imagem digital em múltiplos segmentos. Mais precisamente, é um processo que simplifica a representação de uma imagem de modo a que seja mais fácil e intuitivo analisá-la. Em termos práticos, para localizar objectos e os seus limites numa imagem.

Os descritores de cor MPEG-7 apresentam-se como uma solução eficiente de correspondência de segmentos de imagem ou vídeo baseado nas características da cor. A sua baixa complexidade de extração e correspondência permitem que sejam usados em tarefas de pesquisa e recuperação de imagens a partir de bases de dados em dispositivos com reduzido poder computacional.

A figura 3.1 apresenta uma visão geral dos conteúdos transmitidos através do serviço de TV e a sua respectiva sequência. Sabe-se que, logo após terminar o programa, irá aparecer um separador característico do canal por um período de tempo curto. Os instantes seguintes são dedicados a anúncios de publicidade e, momentos antes de iniciar o programa, será apresentado outro separador.

Apenas no método apresentado em 3.1.2 se aplica o vector de cor a uma dimensão, em escala de cinza, nos restantes são considerados o vector tridimensional de cor, usando o espaço de cor RGB, HSV ou YCbCr.

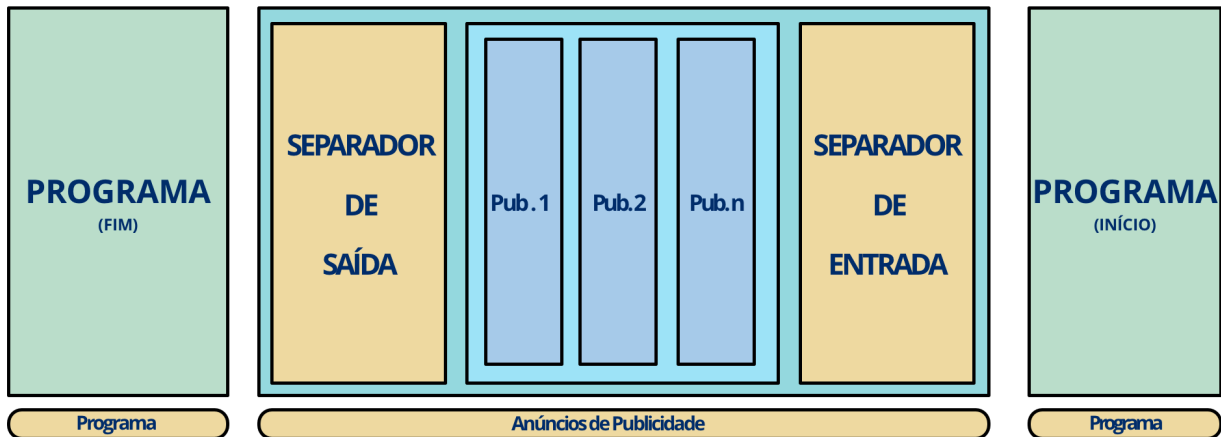


Figura 3.1: Estrutura simplificada do conteúdo transmitido por TV

3.1 Detecção e Reconhecimento de Logos

A presença de um logo durante a apresentação de um conteúdo é algo comum em qualquer canal de televisão. É uma forma de proteger o conteúdo, identificando o canal.

Para cada estação existe uma posição predefinida do logo, tipicamente, nos cantos da imagem. Podem ocorrer situações em que de programa para programa a posição do logo é alterada. No entanto, essa posição é constante ao longo da transmissão do mesmo programa.

Normalmente os logos podem ser de dois tipos: opacos ou semi-transparentes. Entende-se por logo opaco aqueles logos em que os píxeis que o definem não são afectados pelo programa que surge como fundo, ou seja, o valor de cor dos píxeis sobrepõe-se na totalidade ao plano de fundo. Considera-se um logo semi-transparente aquele que, contrariamente ao opaco, sofre uma perturbação resultante da alteração da imagem de fundo, modificando constantemente a sua cor ao longo da transmissão.

Nas sub-seções 3.1.1, 3.1.2 são apresentados os métodos propostos para lidar com cada tipo. Note-se que, para cada método, há uma fase de treino e outra de reconhecimento.

3.1.1 Logos Opacos

Como foi referido na secção anterior, na existência de um logo opaco, há uma sobreposição na íntegra ao plano da imagem. Esse efeito pode ser observado através das figuras (3.2, 3.3).

A subtracção de fundo é uma técnica usada para extrair o *foreground* de uma imagem através da subtracção da frame actual com o modelo de *background*. Este método é utilizado

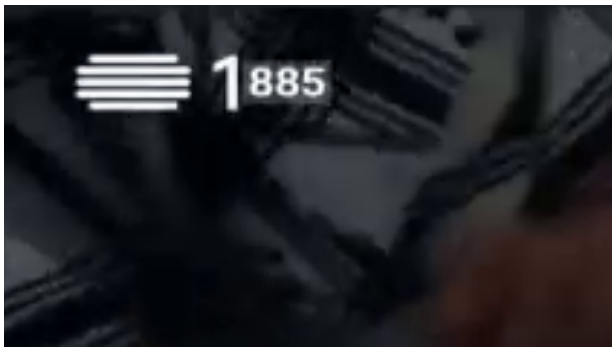


Figura 3.2: Logo opaco da RTP1



Figura 3.3: Logo opaco da SIC

para encontrar objectos em movimento numa sequência de vídeo.

O fundo de uma imagem pode ser considerado como uma média temporal em cada pixel da imagem, representado pela seguinte equação:

$$\mathbf{B}(\mathbf{x}, \mathbf{y}, t) = \frac{1-t}{t}B(x, y, t-1) + \frac{1}{t}I(x, y, t) \quad (3.1)$$

Em que $I(x, y, t)$ é o frame na posição (x, y) no instante temporal t .

Esta equação, ao longo do tempo, destaca os objectos estacionários dado que os píxeis não sofreram alterações, desse modo as zonas da imagem com movimento são identificadas como ruído.

Paralelamente à construção do modelo de fundo, o cálculo da variância em 3.2 irá indicar o *quão longe* o valor de cada píxel se encontra do valor esperado pelo modelo.

$$\vartheta(\mathbf{x}, \mathbf{y}, t) = \frac{1-t}{t}\vartheta(x, y, t-1) + \frac{1}{t}(I(x, y, t) - B(x, y, t))^2 \quad (3.2)$$

Na região em que o objecto é estacionário é expectável que a variância tome valores perto de zero. O cálculo da máscara binária que indica a posição (x, y) de cada pixel do logo toma por base os valores da variância para $t = t_{detecao}$ e o limiar entre $0 \leq T_{var} \leq 5$, definidos empiricamente.

$$mask(x, y) = \begin{cases} 1 & \text{se } \vartheta(x, y) \leq T_{var} \\ 0 & \text{se } \vartheta(x, y) > T_{var} \end{cases} \quad (3.3)$$

Podem ocorrer duas situações, a primeira exemplificada na Figura 3.4 denominada de *Letterbox* em que aparece barras pretas na parte superior e inferior da imagem. A segunda situação retratada na Figura 3.5 é quando o conteúdo é gravado, por exemplo, em 4 : 3 e transmitido num *widescreen*, neste caso, as barras pretas são verticais do lado esquerdo e

direito. Nesses casos, é necessário a aplicação de um filtro antes da obtenção da máscara binária, para remover essas barras pretas.

O algoritmo 1 é apresentado no próximo capítulo e é indicado para as situações em que o conteúdo transmitido foi gravado numa proporção diferente do ecrã a transmitir.



Figura 3.4: Letterbox



Figura 3.5: Conteúdo 4:3 em 16:9

Na fase do reconhecimento, a comparação entre duas frames I_1, I_2 é verificada pixel a pixel, conferindo se a cor de cada pixel se encontra dentro de uma gama, definida empiricamente por $I_1(x, y) - T_{cor} \leq I_2(x, y) \leq I_1(x, y) + T_{cor}$ com $T_{cor} \in [0, 30]$.

Sabendo que a comparação entre duas tramas I_1, I_2 apenas se realiza nos pontos pertencentes à máscara, ou seja, na região do logo, o resultado da comparação normalizado, deverá respeitar os valores definidos empiricamente entre $T_{eq} \in [0.8, 1]$, para que seja considerada a presença do logo.

3.1.2 Logos Semi-transparentes

O logo semi-transparente apresenta características que não permitem uma abordagem baseada em cor, apesar de manter a sua forma inalterável ao longo do tempo, o seu conteúdo em termos de luminância e cor varia. As figuras 3.6 e 3.7 ilustram um exemplo de um logo semi-transparente.

Apesar das variações de conteúdo referidas, o logo semi-transparente apresenta sempre uma luminância elevada. Por essa razão, o método encontrado para detectar a presença do logo é calcular o mínimo brilho em cada píxel da *frame*. Essa operação, permite encontrar a sua localização pois, na presença do logo, o valor mínimo será sempre elevado. O algoritmo 2 que calcula o mínimo brilho é apresentado no próximo capítulo.

Num passo seguinte, é necessário filtrar as componentes escuras, ou seja, as partes que não abrangem o objecto a detectar. Com a aplicação de uma máscara binária como definida na equação 3.4, o limiar T_{black} definido empiricamente, estabelece quais os píxeis da *frame*



Figura 3.6: Logo semi-transparente da Eurosport 1

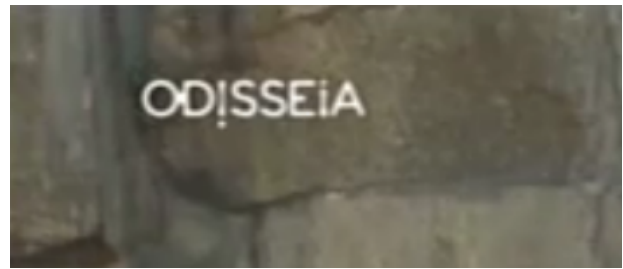


Figura 3.7: Logo semi-transparente da Odisseia

que não pertencem ao logo.

$$mask(x, y) = \begin{cases} 1 & \text{se } I(x, y) > T_{black} \\ 0 & \text{se } I(x, y) \leq T_{black} \end{cases} \quad (3.4)$$

No entanto, apesar da máscara nos indicar quais os píxeis que pertencem ao logo, em termos práticos apenas serão usados os píxeis que se encontram nas bordas.

A deteção de arestas é uma técnica de segmentação de imagem conhecida por encontrar os limites dos objectos numa imagem, normalmente originados por diferenças no brilho.

Neste método proposto para encontrar logos semi-transparentes, foram usados os operadores *Prewitt* e *Canny*. O *Canny* conhecido por ser mais sensível a detectar bordas, foi aplicado na fase de reconhecimento, visto que nesse passo é extremamente importante identificar todas as mudanças de brilho. O *Prewitt*, menos sensível, é usado na fase de deteção do logo pois com a sua aplicação resulta menos ruído.

A correspondência entre *edges* não pode ser vista como uma comparação de píxeis na mesma coordenada, ou seja, por vezes o pixel correspondente está num raio próximo. Desse modo, é preciso pesquisar num raio envolvente a menor distância d a que corresponde a borda. No caso de não existir nenhum ponto nesse raio é assumida uma distância elevada previamente definida para penalizar a procura. O algoritmo 3 que calcula a menor distância é apresentado no próximo capítulo e é aplicado a cada ponto do contorno detectado.

De seguida, é efectuado, um somatório de cada distância mínima representado por $D = \sum_{i=1}^n d(i)$. Em função do número total de n pontos existentes, o multiplicador m indica o quão próximos os pontos reconhecidos podem estar. Na equação 3.5, o R indica se foi reconhecido com sucesso a presença de um logo.

$$R = \begin{cases} 1 & \text{se } D \leq n \times m \\ 0 & \text{se } D > n \times m \end{cases} \quad (3.5)$$

3.2 Reconhecimento de Separadores de Vídeo

Analisando o que normalmente se passa nas transmissões televisivas, particularmente nos canais por cabo, podemos observar que quando um programa é interrompido para dar início a um bloco publicitário, é utilizado um separador que normalmente é único e identificativo do canal. Esse separador caracteriza-se por ser curto, utilizando normalmente cores fortes que dificilmente ocorrem num programa.

No final do bloco publicitário é também utilizado outro separador que é diferente do primeiro. Este facto sugere que uma técnica que poderá ser utilizada como meio de identificar a ocorrência de publicidade será o reconhecimento desses separadores. Utilizando um conjunto de descritores que permitam identificar de forma única esses separadores, criando como que uma assinatura de cada separador, será possível em tempo real, utilizando técnicas de aprendizagem máquina verificar se o segmento de vídeo actual é ou não identificado como um dos separadores. Esses descritores poderão ser armazenados numa base de dados. O seu armazenamento implica uma divisão por classes que é ilustrada através da figura 3.8.

Os descritores utilizados para criar a assinatura de cada separador foram os histogramas de cor e os descritores de cor.

A norma MPEG-7 define um conjunto de descritores que permitem representar conteúdos multimédia. Estes descritores ajudam os utilizadores e as aplicações a identificar, catalogar e a filtrar imagens ou conteúdo de vídeo.

O conteúdo multimédia pode ser descrito pelas suas características visuais, isto é, pela sua cor, textura, forma, tamanho e movimento dos objectos e da câmara.

Nesta dissertação, apenas são considerados os descritores de cor, nomeadamente o DCD e o CLD, nas seções 3.2.1 e 3.2.2 respectivamente. Estes descritores apresentam uma baixa complexidade computacional o que permite o seu cálculo em tempo real.

3.2.1 Descritor de Cor Dominante

O descritor de cor dominante fornece uma descrição compacta das cores representadas numa imagem ou região da imagem. Este descritor é usado tipicamente no reconhecimento de similaridade na pesquisa de imagens em base de dados com cores semelhantes.

O DCD é definido por:

$$F = \{(C_i, p_i, v_i), s\}, (i = 1, 2, \dots, N) \quad (3.6)$$

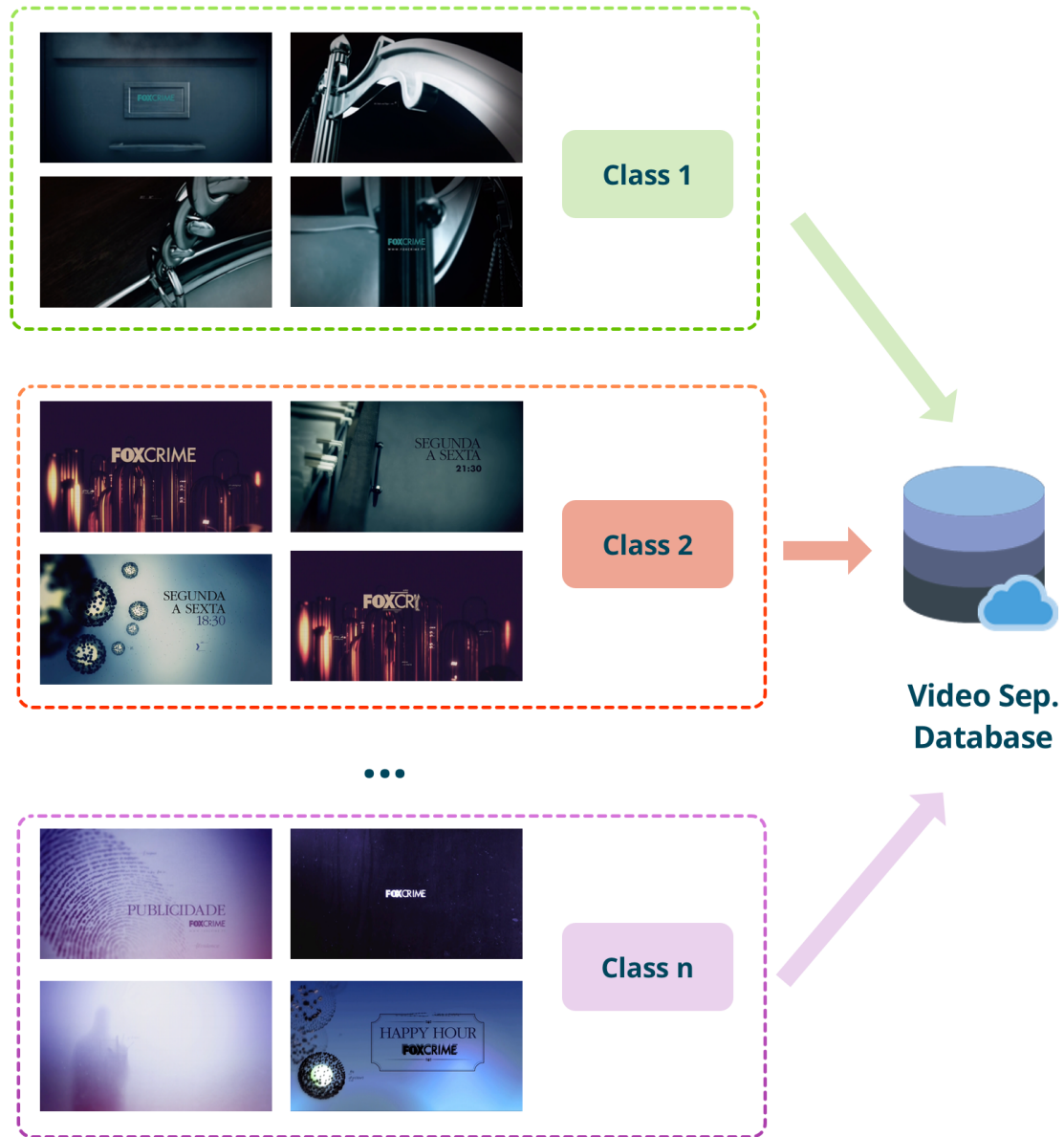


Figura 3.8: Estrutura da base de dados de separadores

em que N é o número de cores dominantes. Para cada cor dominante, C_i representa o vector tridimensional de cor representado num espaço de cor. Um máximo de 8 cores dominantes é suficiente para representar uma imagem ou região da imagem. A percentagem p_i , normalizada para valores entre 0 e 1, é a fracção de píxeis numa imagem ou região correspondente à cor C_i , sendo $\sum_i p_i = 1$.

Neste cenário, a imagem é dividida em blocos ($n \times n$) com $n = 8, 16, 32, 64$ e calculada a média de cores de cada bloco. Para obter as N cores dominantes, são percorridos todos os n^2 blocos e calculada a percentagem de cada cor dominante.

Considerando dois DCDs,



Figura 3.9: Imagem representativa de um separador do canal AXN



Figura 3.10: Separador dividido em blocos de $n = 16$ e respectiva média

$$F_1 = \{(C_{1i}, p_{1i}, v_{1i}), s_1\}, (i = 1, 2, \dots, N_1) \text{ e}$$

$$F_2 = \{(C_{2i}, p_{2i}, v_{2i}), s_2\}, (i = 1, 2, \dots, N_2)$$

Ignorando os parâmetros opcionais da variância v e a coerência espacial s , a dissimilaridade $D(F_1, F_2)$ em 3.7 mede o quão diferentes são os descritores.

$$D^2(F_1, F_2) = \sum_{i=1}^{N_1} p_{1i}^2 + \sum_{i=1}^{N_2} p_{2i}^2 - \sum_{i=1}^{N_1} \sum_{i=1}^{N_2} 2a_{1i,2j} p_{1i} p_{2j} \quad (3.7)$$

em que os índices 1 e 2 da equação 3.7 representam, respectivamente, as variáveis de F_1 e F_2 e $a_{k,l}$ o coeficiente de similaridade entre duas cores c_k e c_l ,

$$a_{k,l} = \begin{cases} 1 - \frac{d_{k,l}}{d_{max}} & \text{se } d_{k,l} \leq T_d \\ 0 & \text{se } d_{k,l} > T_d \end{cases} \quad (3.8)$$

onde $d_{k,l} = \|c_k - c_l\|$ é a distância Euclidiana entre duas cores c_k e c_l , T_d a máxima distância entre duas cores consideradas semelhantes e d_{max} a distância máxima admissível.

3.2.2 Descritor de Layout de Cor

O CLD captura a distribuição espacial de cor numa imagem ou região da imagem. A representação é baseada nos coeficientes da Transformada Discreta do Coseno (DCT). Este descritor é dos mais compactos do MPEG-7, adequado para aplicações limitadas em termos de desempenho, armazenamento e largura de banda.

O processo de extração deste descritor, ilustrado na figura 3.11 consiste em quatro etapas: particionamento da imagem, representação da cor seleccionada, transformada DCT e, por fim, quantitização não-linear dos coeficientes de zigzag.

No particionamento, a imagem é dividida em 64 blocos (8 x 8) para garantir a invariância à resolução e escala. Seguidamente, na representação de cores é calculada a média de cada bloco e convertido para o espaço de cor recomendado YCbCr, no entanto, outro espaço de cor poderá ser usado.

No estágio da DCT, para cada componente do espaço de cor YCbCr, são obtidos 64 coeficientes da DCT. Paralelamente é realizado uma amostragem em zigzag com o objectivo de agrupar os coeficientes de baixa frequência.

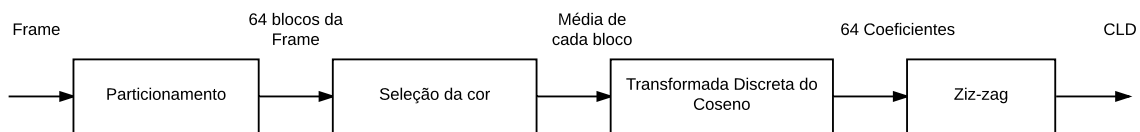


Figura 3.11: Processo de extração do CLD

Cada imagem pode assim ser representada por 3 componentes de 64 coeficientes. O armazenamento dos coeficientes dos CLD's na base de dados está ilustrado na Figura 3.8, separado por classes.

Na correspondência entre dois CLDs, (D_Y, D_{Cr}, D_{Cb}) e (D'_Y, D'_{Cr}, D'_{Cb}) é aplicada a equação 3.9 que nos dá uma medida de semelhança entre dois blocos. No nosso caso concreto e depois de testes realizados foi definido empiricamente que existe uma correspondência quando o valor desta medida é inferior a T_{cld} , com $T_{cld} \leq 20$.

$$D = \sqrt{\sum_i (DY_i - DY'_i)^2} + \sqrt{\sum_i (DCb_i - DCb'_i)^2} + \sqrt{\sum_i (DCr_i - DCr'_i)^2} \quad (3.9)$$

3.2.3 Histograma de cor

Em termos matemáticos, o histograma ou como também é conhecido a distribuição de frequências é uma forma gráfica de representação de um conjunto de dados dividido por classes uniformes ou não uniformes.

Aplicado ao contexto prático, o histograma de cor, permite representar todas as cores contidas numa imagem e a sua frequência de ocorrência. Podem ser representados por diferentes espaços de cor como, por exemplo, o RGB ou HSV. Cada componente de cor pode variar entre 0 e 255, no caso de uma imagem com uma profundidade de 8 bits. No entanto,

o histograma pode ser representado por n número de bins, permitindo que cores próximas sejam agrupadas.

A análise de histogramas de cor permite reconhecer imagens com diferentes características. Por outras palavras, perceber se a imagem apresenta um contraste alto ou baixo, se é mais escura ou mais clara, quais as cores que se destacam mais, bem como representar os níveis de saturação.

Esta abordagem é invariante à rotação e translação de uma imagem, contudo apresenta algumas desvantagens, nomeadamente:

- insensibilidade para pequenas mudanças de posição da câmara
- não fornece informação espacial
- insensibilidade a alterações de brilho e compressão numa imagem

Na comparação de dois histogramas normalizados foi aplicado o coeficiente de *Bhattacharyya* com base na sua definição:

$$C_B(p, q) = \sum_{x \in X} \sqrt{p(x)q(x)} \quad (3.10)$$

Considerando que p é o histograma da imagem da base de dados e q o histograma da frame a analisar.

Com a aplicação deste coeficiente, a distribuição de valores de 3.10 situa-se entre $[0, 1]$, deste modo, foi definido empiricamente que para valores de C_B acima de 0.9 é considerado uma boa semelhança.

4 Procedimento experimental

Este capítulo descreve os diferentes procedimentos experimentais que foram desenvolvidos para testar os métodos propostos. Na secção 4.1, são apresentados os passos para o treino e reconhecimento de logos opacos e semi-transparentes. Seguidamente, a secção 4.2 esclarece as metodologias usadas desde a importação até à identificação de separadores de vídeo. De destacar que estas técnicas foram implementadas em MATLAB e testadas em ambiente offline.

O vídeo capturado directamente dos canais de TV é gravado em MPEG-TS e descodificado através da framework multimédia FFMPEG, exportando cada frame num imagem no formato PNG com uma *frame rate* variável. Os vídeos analisados usam o codec de vídeo H 264.

A classificação é feita e avaliada tendo por base a etiqueta que é atribuída a cada *frame*, ou seja, após o processo de extração, cada *frame* é rotulada manualmente como sendo programa ou publicidade e, dentro da publicidade se é um separador ou não.

4.1 Deteção e Reconhecimento de logos

O algoritmo divide-se em duas fases: a da deteção em que é necessário um tempo mínimo para detectar o logo e a sua localização e do reconhecimento, aplicando o que foi detectado à sequência seguinte de *frames*. Por vezes, o tempo previamente definido pode não ser suficiente e nesse caso o algoritmo automaticamente incrementa-o até ser possível detectar a presença do logo.

Na primeira etapa, começa-se por segmentar cada *frame*, recolhendo os quatro cantos da imagem onde é conhecida a presença de um logo identificativo do canal, dando prioridade ao canto onde à priori se espera encontrar o logo.

Nos sub-capítulos seguintes serão abordados, de forma mais pormenorizada, os procedimentos propostos para cada tipo de logo, após a segmentação.

4.1.1 Logos Opacos

Como vimos anteriormente, o logo opaco sobrepõem-se na totalidade ao plano de fundo. Esta particularidade implica a necessidade de se trabalhar com um vector tridimensional de cor, visto que aquilo que procuramos se mantém inalterável ao longo do tempo.

Durante a fase de deteção, $B(x, y, t)$ em cada ponto (x, y) modela as partes da imagem que são constantes. Na presença do logo, o *background* vai apresentar os píxeis correspondentes ao logo e, fora do seu limite, outros valores que se considera serem lixo. Essa observação pode ser verificada na imagem 4.7.

Antes da obtenção da máscara, é aplicado um filtro responsável por eliminar as linhas e colunas totalmente pretas do *Background*, bem como as linhas que não são completamente pretas mas que contém o logo. A remoção dessas linhas e colunas consiste em aumentar significativamente a variância de modo a que não sejam detectadas como pertencentes ao logo. As figuras 4.1 a 4.6 ilustram que, na zona onde o background é preto a variância apresenta valores elevados (representado a vermelho). Por sua vez, o histograma mostra a percentagem de preto para cada linha em 4.3 ou coluna em 4.6.

Este fenómeno acontece quando os conteúdos de vídeo são gravados numa resolução diferente do ecrã a apresentar. Nesse caso, para manter o mesmo *aspect ratio* são adicionadas barras pretas. De modo a resolver este problema encontrado, o filtro é descrito no algoritmo 1. De salientar que este passo é opcional, uma vez que esta característica não acontece sempre.

Na caracterização da máscara binária, considera-se que os valores cuja variância se encontra no intervalo $[0, T_{var}]$, isto é, próximos de zero, estamos perante a existência de um píxel válido. Desta forma, é conhecida a localização de todos os píxeis que fazem parte do logo. As figuras 4.8, 4.9 ilustram essa observação.

Após a deteção, a primeira verificação é perceber se a máscara é diferente de zero, indicando a possibilidade da ocorrência do logo. Caso não se comprove, é porque o conteúdo indica ser publicidade e, neste caso, é feita uma nova deteção. Sendo a máscara válida, não é garantida a existência do logo, nesse sentido é aplicado o reconhecimento ao último frame de treino, indicando a necessidade de explorar ou não mais t frames. Por outras palavras, confirmar se realmente o objecto treinado corresponde ao logo.

O reconhecimento é uma comparação pixel a pixel, nos pontos onde a máscara é diferente de zero, verificando-se se a cor da imagem corresponde à cor do logo dentro do limiar admissível T_{cor} .



Figura 4.1: Background do logo AXN Black HD



Figura 4.2: Variância com aplicação do filtro no logo AXN Black HD

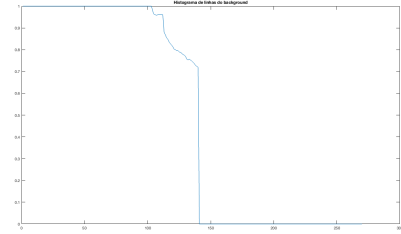


Figura 4.3: Histograma de linhas do Background



Figura 4.4: Background do logo FOX Crime



Figura 4.5: Variância com aplicação do filtro no logo FOX Crime HD

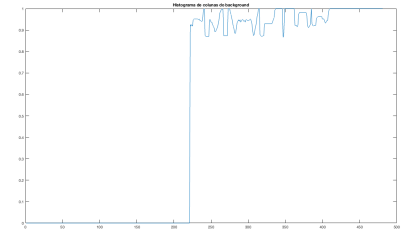


Figura 4.6: Histograma de colunas do Background



Figura 4.7: Background do logo opaco da Sport TV 1



Figura 4.8: Variância do logo opaco da Sport TV 1



Figura 4.9: Máscara do logo opaco da Sport TV 1

Em suma, um resultado cuja percentagem de igualdade T_{eq} é superior a 80% é indicativo da presença de um logo, ou seja, de um programa.

4.1.2 Logos Semi-transparentes

Uma característica importante deste tipo de logos é que o seu brilho apresenta sempre valores elevados de tal forma que, por vezes, quando a imagem é muito clara ele parece não estar visível, conforme pode ser observado na Figura 4.11.

Como a análise baseada na cor não traz benefícios para a deteção e reconhecimento, este método atua apenas na escala de cinza com uma resolução de 8 bits.

Retomando a etapa de deteção, ao longo do tempo t mínimo são percorridos todos os píxeis da imagem, por forma a encontrar em cada ponto o mínimo brilho, ou seja, o ponto mais escuro nesse intervalo t . Sabe-se que na posição do logo, esse brilho vai apresentar

Algorithm 1 Filtro de remoção de blocos pretos

```

background(h,w,color)                                ▷ Background obtido
var(h,w,color)                                       ▷ Variância
                                                    ▷ color ← Vector tridimensional de cor
height, width ← size(background)                    ▷ Tamanho do background (largura, altura)
blackpixel                                           ▷ Pixel considerado preto
for h = 1 : height do                               ▷ Percorre todas as linhas
  for w = 1 : width do                               ▷ Percorre todas as colunas
    if background(h,w,color) ≤ blackpixel then       ▷ Se o pixel for preto
      histy(h) ← histy(h) + 1                       ▷ Incrementa a coluna h
      histx(w) ← histx(w) + 1                       ▷ Incrementa a linha w
    end if
  end for
end for
histy ← histy ÷ width                               ▷ Normaliza as linhas
histx ← histx ÷ height                              ▷ Normaliza as colunas
                                                    ▷ Elimina as linhas/colunas totalmente pretas
posy ← find(histy == blackpixel)                   ▷ Linhas totalmente pretas
posx ← find(histx == blackpixel)                   ▷ Colunas totalmente pretas
var(posy,w,1) ← 255                                 ▷ Aumenta a variância das linhas pretas
var(h,posx,1) ← 255                                 ▷ Aumenta a variância das colunas pretas
                                                    ▷ Elimina as linhas parcialmente pretas
posy ← find(histy ≤ blackpixel && histy > pblack)  ▷
for h = min(posy) : max(posy) do                 ▷ Percorre as linhas parcialmente pretas
  for w = 1 : width do                             ▷ Percorre todas as colunas
    if background(h,w,color) ≤ blackpixel then
      var(h,w,1) ← 255                               ▷ Aumenta a variância dos pixels pretos
    end if
  end for
end for
                                                    ▷ Elimina as colunas parcialmente pretas
posx ← find(histx ≤ blackpixel && histx > pblack)  ▷
for h = 1 : height do                             ▷ Percorre todas as linhas
  for w = min(posx) : max(posx) do                 ▷ Percorre todas as colunas
    if background(h,w,color) ≤ blackpixel then
      var(h,w,1) ← 255                               ▷ Aumenta a variância dos pixels pretos
    end if
  end for
end for

```

sempre valores elevados em comparação com os demais.

O cálculo do mínimo brilho é descrito pelo algoritmo 2 e representado através da Figura 4.13.

De seguida, é calculada uma máscara em 4.14 com base num limiar de preto T_p que permite obter a localização de todos os píxeis pertencentes ao logo. Por fim, é aplicado um *edge detector* em 4.15 para obter as bordas do *template* do logo. A partir dos pontos do *edge*



Figura 4.10: Apresentação de um logo semi-transparente visível



Figura 4.11: Apresentação de um logo semi-transparente pouco visível

Algorithm 2 Cálculo do mínimo brilho

```

1:  $img(t,h,w)$  ▷ Frame no instante  $t$ 
2:  $height, width \leftarrow size(img)$  ▷ Tamanho da imagem
3:  $t\_det$  ▷ Tempo de detecção
4:  $minBright(h,w)$  ▷ Inicialização da matriz no máximo de brilho (e.g 255)
5: for  $t = 1 : t\_det$  do ▷ Percorre o tempo de detecção
6:   for  $h = 1 : height$  do ▷ Percorre todas as linhas
7:     for  $w = 1 : width$  do ▷ Percorre todas as colunas
8:       if  $img(t,h,w) < minBright(h,w)$  then ▷ Se encontrar um pixel com menor
         brilho
9:          $minBright(h,w) \leftarrow img(t,h,w)$  ▷ Actualiza o valor em  $(h, w)$ 
10:      end if
11:    end for
12:  end for
13: end for

```

são guardadas as posições que servirão para a fase posterior de reconhecimento.

Ao observar a figura 4.12 que representa um *degrade* de cinza em 8 bits, verificamos que os valores à esquerda da coluna azul são os mais escuros. Na verdade, esses valores a olho humano são indistinguíveis.

Nesse sentido, o limiar T_p atribuído tem como finalidade filtrar as componentes que não pertencem ao logo, neste caso, os píxeis mais escuros.



Figura 4.12: Representação de um degrade de cinza

Para validar a etapa da detecção e antes de avançar para o passo seguinte, é fundamental perceber se o intervalo de tempo analisado contém o objecto identificativo ou se o conteúdo era publicidade. Nesse sentido, a primeira verificação é analisar se a máscara binária está totalmente a zero, nesse caso não existe logo e a imagem é totalmente preta. Caso contrário, o último *frame* serve de chave para ser aplicado o reconhecimento. Se coincidir, existem indícios de que o logo pode estar presente. Caso contrário, será incrementado o tempo até ser obtido um *template* do logo em melhores condições.

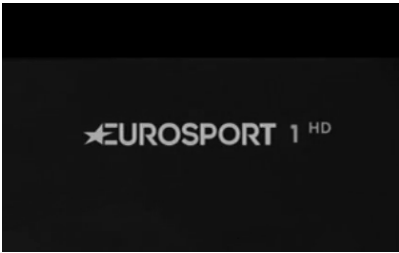


Figura 4.13: Representação do Mínimo Brilho



Figura 4.14: Máscara do logo semi-transparente



Figura 4.15: *Prewitt* aplicado ao logo semi-transparente

O passo seguinte é reconhecer, para isso em cada um dos pontos do *edge* previamente guardado na detecção será aplicado o algoritmo *refalg:mindistance* que calcula o somatório da menor distância em cada ponto. Este algoritmo tem como objectivo a correspondência de bordas sabendo que, por vezes, os pontos podem não coincidir na mesma coordenada, nesse caso é necessário procurar numa vizinhança em torno do ponto considerado. Esse comportamento pode ser verificado através da figura 4.18 que ilustra algumas falhas de interseção entre a imagem detectada em 4.16 e o *frame* reconhecido em 4.17 num determinado instante.

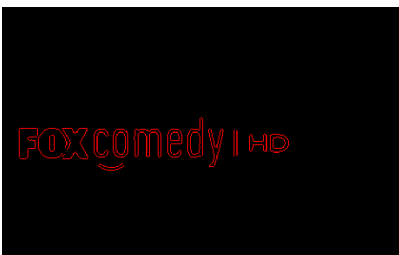


Figura 4.16: Logo semi-transparente detectado



Figura 4.17: Logo semi-transparente reconhecido

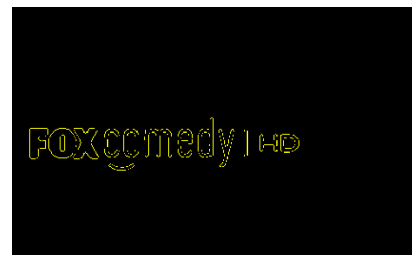


Figura 4.18: Interseção de logo detectado e reconhecido

A correspondência é realizada com base na equação 3.5 demonstrada no capítulo anterior, que indica o sucesso do reconhecimento. O somatório das distância mínimas terá de ser inferior à multiplicação entre o número total de pontos da borda detectada e o valor m aceitável em torno do ponto.

Algorithm 3 Cálculo da mínima distância

```

1:                                     ▷ xpoints, ypoints ← Pontos (x,y) do edge do logo detectado
2:                                     ▷ current_edge ← Pontos do edge da frame actual
3: nlevel                               ▷ Raio a procurar
4: max_dist                             ▷ Máxima distância considerada, quando não encontrado
5: for  $q = 1 : length(ypoints)$  do                                     ▷ Para cada ponto do edge do logo
6:   current_edge_dist(q) ← max_dist ▷ Inicia o vector em q com a máxima distância
7:   for  $yy = ypoints(q) - nlevel : ypoints(q) + nlevel$  do
8:     for  $xx = xpoints(q) - nlevel : xpoints(q) + nlevel$  do
9:       dx ←  $abs(xx - xpoints(q))$                                      ▷ Distancia x em relação ao ponto do edge
10:      dy ←  $abs(yy - ypoints(q))$                                      ▷ Distancia y em relação ao ponto do edge
11:      df ←  $sqrt(dx^2 + dy^2)$                                        ▷ Distância total
12:      if current_edge(yy,xx) == true OR current_edge_dist(q) > df then
13:        current_edge_dist(q) ← df                                     ▷ Actualiza a distância
14:      end if
15:    end for
16:  end for
17: end for
18: min_distance ← sum(current_edge_dist)

```

4.2 Reconhecimento de Separadores de Vídeo

Como foi descrito anteriormente, os separadores surgem na entrada e na saída de um bloco publicitário. Por vezes, o próprio separador pode ter um segmento que seja diferente de vez para vez (por exemplo pode ser utilizado para anunciar um próximo programa). Por esta razão vamos apenas analisar a parte inicial e a parte final do separador que é normalmente constante ao longo das várias ocorrências do mesmo.

O procedimento aplica-se aos três métodos propostos na secção 3.2 para identificação de separadores, os histogramas e descritores de cor (CLD, DCD).

Numa primeira fase, após a etiquetagem, as *frames* consideradas foram importadas para a base de dados e divididas em classes, como indicado na figura 4.19.

Note-se que a abordagem por classes permite verificar se determinada sequência ainda se encontra activa, possibilitando que os conteúdos da DB sejam atualizados periodicamente.

A informação contida na DB, difere em função do método utilizado. Para cada imagem, a abordagem por:

- **Histograma de cor:** armazena a frequência de cada bin;
- **DCD:** armazena as N cores dominantes num vector tridimensional e respectivas percentagens;

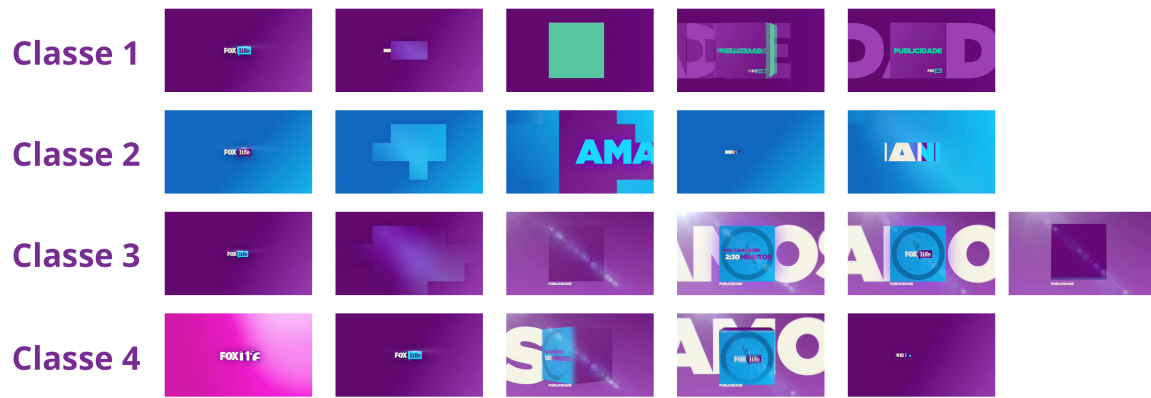


Figura 4.19: Representação de um separador da FOX Life por classes

- **CLD:** armazena os 64 coeficientes de cada componente do espaço de cor YCbCr

Esta representação simplificada dos dados proporciona uma rápida indexação e correspondência.

A comparação entre frames é realizada a cada elemento da classe na DB, através da equação característica de correspondência de cada método. Sabendo que à medida que vão sendo reconhecidos os separadores, os primeiros elementos de cada classe vão sendo omitidos, poupando assim tempo computacional.

5 Testes e Resultados

Este capítulo apresenta todos os testes efectuados para avaliar o desempenho dos métodos propostos. Está dividido em duas secções, a primeira apresenta os resultados relativos à deteção e reconhecimento de programas e a segunda, para o reconhecimento de separadores.

Cada *frame* dos vídeos de teste foi previamente etiquetado como sendo programa, publicidade ou separador, por forma a avaliar o desempenho dos métodos propostos. De modo a quantificar os resultados obtidos através desses métodos foram consideradas medidas estatísticas de desempenho de um teste de classificação binário, nomeadamente a Sensibilidade e Especificidade.

A sensibilidade (Eq. 5.1) mede a proporção de positivos que são correctamente identificados. A especificidade (Eq. 5.2) mede a proporção de negativos que são devidamente identificados. Idealmente, pretende-se que os falsos positivos e falsos negativos apresentem valores perto de zero.

$$\text{Sensibilidade} = \frac{\text{Verdadeiro Positivo}}{\text{Verdadeiro Positivo} + \text{Falso Negativo}} \quad (5.1)$$

$$\text{Especificidade} = \frac{\text{Verdadeiro Negativo}}{\text{Verdadeiro Negativo} + \text{Falso Positivo}} \quad (5.2)$$

$$R_{SE} = \text{Sensibilidade} \times \text{Especificidade} \quad (5.3)$$

De modo a ter um sistema robusto que possa ser aplicado de forma automática, é necessário ter uma amostragem de teste representativa de diversos canais.

A amostragem foi recolhida através da gravação de streams de canais de diferentes categorias:

- **Generalista:** Antena3, La Sexta1, RTP1, Sic, TVI
- **Informação:** RTP3, Sic Noticias, TVI 24,

- **Desporto:** Sport TV, Eurosport
- **Entretenimento:** Sic Radical, Sic Caras
- **Infantil:** Disney Channel
- **Lifestyle:** Sic Mulher
- **Filmes e Séries:** AMC, AXN, AXN Black, AXN White, Fox Crime, Fox Comedy, Fox Life, Fox Movies, TVCine
- **Documentários:** Discovery, Odisseia, National Geographic, Nat GEO Wild2

Após a gravação dos streams, foram extraídas os frames de cada vídeo. O formato de imagem escolhido foi o PNG pois em comparação com o JPEG apresenta uma melhor qualidade de imagem. Apesar do PNG também ser um formato de imagem comprimido não se verifica tantas perdas de compressão. Por outro lado, o PNG é mais pesado em termos de ocupação de espaço em disco.

Conforme referido anteriormente, os testes foram efectuados em ambiente *offline*. Em todos os cenários testados, foi utilizado um computador portátil baseado num processador Intel i7-2620M com uma frequência de relógio de 2.7GHz, 4 GB de memória DDR3 e um disco SSD com velocidades de leitura e escrita máximas de 400 MB/s. O *software* usado para correr os algoritmos foi o MATLAB R2016b.

De modo a simular um contexto real e prático, os streams não têm um conteúdo inicial predefinido, ou seja, tanto pode acontecer os streams começarem diretamente com um programa ou com publicidade. A amostra utilizada de cada canal representa na grande maioria dos casos pelo menos duas horas de vídeo a 1 frame/s.

Os resultados são sumariados nas tabelas (5.2, 5.1.2, 5.5, 5.6), onde a variável Tempo é expressa em segundos. O tempo mencionado é o resultado para o número Total de *Frames* de cada amostra. Por sua vez, a Resolução é expressa em píxeis e representa as medidas de comprimento e altura dos frames de cada canal.

5.1 Detecção e Reconhecimento de Logos

Nesta secção, serão apresentados as condições de teste para os métodos de logos opacos e semi-transparentes.

Inicialmente, foram identificados os canais cujos logos são opacos e quais é que são semi-transparentes, por forma a determinar qual o algoritmo a usar. Importa salientar que o algoritmo apresentado em 4.1.2 para a deteção e reconhecimento de logos semi-transparentes pode ser aplicado também ao logo opaco. Nesse sentido, serão apresentados os resultados deste algoritmo nas amostras cujo logo é opaco.

5.1.1 Logos Opacos

Neste algoritmo, os parâmetros de entrada do algoritmo definidos anteriormente na secção 4.1.1 encontram-se apresentados na tabela 5.1. Para cada TesteID foram realizados testes com o objectivo de comparar os seus efeitos no resultado.

TesteID	T_cor	T_var	T_eq
Teste 1	15	5	0.93
Teste 2	30	5	0.8

Tabela 5.1: Valores de configuração do algoritmo de logos opacos

Em todos os casos de teste, foi aplicado o filtro descrito no algoritmo 1 com $T_p = 20$ e o tempo mínimo t_{min_opaco} considerado para a deteção foi de 60 segundos.

Informação do Vídeo			Classificação		Resultados	
Canal	Resolução	Total Frames	Sens.	Especif.	R_{SE}	Tempo
Antena 3	1280 x 720	8940	0.92959	1	0.92959	407.1797
AXN Black HD	1920 x 1080	3042	0.99601	0.97561	0.97172	125.8388
AXN White HD	1920 x 1080	5094	0.98577	0.99681	0.98262	322.1958
AXN HD	1440 x 1080	3332	0.99761	0.97311	0.97078	146.0495
Fox Life HD	1920 x 1080	9236	0.97758	1	0.97758	574.7349
Fox Movies	720 x 576	13986	0.85904	0.99508	0.85481	168.3752
La Sexta 1	1280 x 720	8673	0.89151	0.9994	0.89098	238.7625
National Geographic	1280 x 720	10195	0.68815	1	0.68815	250.6247
National Geographic Wild	1280 x 720	6821	0.93141	1	0.93141	245.4347
RTP1	720 x 576	6601	0.39009	0.87745	0.34228	89.1392
RTP3	720 x 576	8591	0.51466	0.88676	0.45639	104.52
SIC	720 x 576	10778	0.71202	1	0.71202	171.9894
SIC MULHER	720 x 576	23468	0.76839	1	0.76839	306.6462
SIC NOTICIAS	720 x 576	7079	0.55087	1	0.55087	104.5922
SPORT TV 1 HD	1440 x 1080	11954	0.71961	1	0.71961	685.2229
SPORT TV 1	720 x 576	7942	0.41398	1	0.41398	113.7165
TVCine 3	1280 x 720	6287	0.99214	1	0.99214	141.3654

Tabela 5.2: Resultados para o Teste 1

Na tabela 5.2 estão representados os resultados obtidos para o Teste1 em 17 canais com características de logo opaco. Esta combinação de argumentos, mostrou resultados bastante

satisfatórios a reconhecer a ausência do logo, com valores de especificidade superiores a 87%, garantindo que este tipo de conteúdos não é erradamente identificado como programa. No que diz respeito à sensibilidade, na grande maioria dos canais o reconhecimento foi positivo, embora como se pode comprovar pela RTP1, RTP3, Sport TV 1 os valores são inferiores a 50%. Esse resultado deve-se ao facto dos logos da RTP1, RTP3 e Sport TV 1 apresentarem uma forma demasiado fina, por exemplo, sob a forma de linhas ou texto num tamanho de letra pequeno. Por esse motivo, os píxeis das bordas do logo podem sofrer pequenas variações de cor. Nesses casos, o T_{cor} teria de ser maior e o T_{eq} mais pequeno, para conseguir apanhar esses casos críticos que estão na fronteira.

Como se pode constatar os melhores resultados foram verificados nos canais HD, com uma resolução mais elevada, sinónimo de melhor qualidade de imagem.

Informação do Vídeo			Classificação		Resultados	
Canal	Resolução	Total Frames	Sens.	Especif.	R_{SE}	Tempo
Antena 3	1280 x 720	8940	0.99322	0.98606	0.97937	390.0788
AXN Black HD	1920 x 1080	4765	0.985	1	0.985	288.4
AXN White HD	1920 x 1080	5094	0.999	1	0.999	330.5
AXN HD	1440 x 1080	3332	0.9996	0.95599	0.95561	145.3972
Fox Life HD	1920 x 1080	9236	0.998	1	0.998	1083
Fox Movies	720 x 576	14006	0.99	0.995	0.986	167.6
La Sexta 1	1280 x 720	8673	0.99643	1	0.99643	345.9678
National Geographic	1280 x 720	10195	0.98401	0.98387	0.96814	250.6247
National Geographic Wild	1280 x 720	6821	0.95665	0.96238	0.92066	224.1409
RTP1	720 x 576	6601	0.777	1	0.777	99.6648
RTP3	720 x 576	8591	0.84399	0.88088	0.74346	102.2366
SIC	720 x 576	10778	0.99973	1	0.99973	145.9414
SIC MULHER	720 x 576	23468	0.997	1	0.997	517.8
SIC NOTICIAS	720 x 576	7079	1	1	1	100.4313
SPORT TV 1 HD	1440 x 1080	11954	0.984	1	0.984	625.9
SPORT TV 1	720 x 576	7962	0.988	1	0.988	115.7
TVCine 3	1280 x 720	6287	0.99935	1	0.99935	168.0534

Tabela 5.3: Resultados para Teste 2

Com vista a melhorar os valores de sensibilidade sem descartar a especificidade, foi aumentado o parâmetro T_{Cor} e reduzido a percentagem de igualdade T_{eq} conforme a tabela de configurações do Teste 2. O comportamento desta alternativa pode ser observado na tabela 5.1.2.

Verifica-se, conforme esperado, um aumento da sensibilidade, ou seja, uma melhoria na taxa de sucesso no reconhecimento de programas. Com excepção da RTP1 e RTP3, tem-se valores acima de 95%.

No que diz respeito à especificidade, não sofreu grandes alterações, confirmando em alguns

canais um ligeiro aumento, em detrimento de um decréscimo ligeiro noutros.

Esta melhoria, no âmbito geral, é justificada pelo facto de que, pelo menos, 80% de igualdade do logo é suficiente para fazer a correspondência. Para além disso, sabendo que os *frames* possuem uma resolução de 24 bits de cor, o equivalente a 16 milhões de cores diferentes, pequenas variações derivadas da descodificação e extração do vídeo serão detectadas utilizando um T_Cor maior.

Em suma, este método, como demonstrado pela tabela 5.1.2, é capaz de responder eficazmente ao reconhecimento de programas cujo logo é opaco.

5.1.2 Logos Semi-transparentes

No cenário do logo semi-transparente, o método apresentado envolve um raio de procura r , uma máxima distância max_dist , um multiplicador m e um limiar de preto T_preto como parâmetros de entrada do algoritmo. Para além destes valores, foi definido empiricamente que o tempo mínimo de deteção para encontrar o logo semi-transparente é de 50 segundos.

A tabela 5.4 mostra, para cada TesteID, os valores destes parâmetros nos testes realizados.

TesteID	m	r	max_dist	T_preto
Teste 1	2	6	200	50
Teste 2	1	4	150	50

Tabela 5.4: Valores de configuração do método de logos semi-transparentes

Conforme referido anteriormente, os resultados apresentados para este método abrangem não só os logos semi-transparentes como também os logos opacos. Dessa forma, é possível compreender se é mais vantajoso ter um método para cada tipo de logo ou se, este método pode ser aplicado de modo generalizado.

Informação do Vídeo			Classificação		Resultados	
Canal	Resolução	Total Frames	Sens.	Especif.	R_{SE}	Tempo
AMC	720 x 576	8953	0.98174	0.44444	0.43633	230.6595
Antena 3	1280 x 720	8950	0.99525	0.79034	0.78658	483.8744
AXN Black HD	1920 x 1080	4755	0.96719	1	0.96719	495.3367
AXN White HD	1920 x 1080	5084	0.99644	0.64856	0.64625	481.6504
AXN HD	1440 x 1080	3332	0.94986	0.7335	0.69672	215.0649
Disney Channel	720 x 576	15201	0.99373	0.64728	0.64322	393.3381
Eurosport 1 HD	1920 x 1080	6392	0.94946	1	0.94946	708.055
FOX Comedy	720 x 576	7139	0.98885	0.55227	0.54612	179.1699
FOX Crime HD	1920 x 1080	32323	0.94047	0.69187	0.65068	3192.4959
FOX Life HD	1920 x 1080	16396	0.91636	0.81255	0.74459	1512.6743
Fox Movies	720 x 576	13976	0.98657	0.89381	0.8818	333.2372
Gazeta TV	768 x 432	8036	0.94623	0.84211	0.79682	338.5555
La Sexta 1	1280 x 720	8673	0.95326	0.78461	0.74793	508.3742
National Geographic	1280 x 720	10195	0.98651	0.83333	0.82209	419.1388
National Geographic Wild	1280 x 720	6821	0.95244	0.73981	0.70463	334.7691
Odisseia	720 x 576	7143	0.99924	0.5717	0.57127	152
RTP 1	720 x 576	6641	0.9687	0.74863	0.72519	172.2372
RTP 3	720 x 576	8631	0.96893	0.72353	0.70105	205.978
Sic	720 x 576	10778	1	0.52492	0.52492	258.9372
Sic Caras	720 x 576	5352	0.99291	0.6963	0.69136	145.6662
Sic Mulher	720 x 576	23458	0.97178	0.65409	0.63563	536.7179
Sic Noticias	720 x 576	7079	0.89872	0.96839	0.87031	155.6735
Sic Radical	960 x 540	7623	0.9826	0.81734	0.80312	193.7141
SportTV HD	1440 x 1080	11944	0.99899	0.84681	0.84596	977.5255
SportTV SD1	720 x 576	7932	0.99711	0.5964	0.59468	218.405
Syfy	720 x 576	7138	0.98629	0.45561	0.44936	165.1028
TVCine 3	1280 x 720	6287	0.99967	0.67614	0.67592	214.1653
TVI	720 x 576	15537	0.80543	0.58104	0.46798	333.2937
TVI 24	720 x 576	7156	0.90789	0.58333	0.5296	172.1724

Tabela 5.5: Resultados para o Teste 1

Uma análise aos resultados apresentados na tabela 5.5 para Teste 1, permite-nos concluir que nos 29 canais testados os valores da sensibilidade são próximos de 1 com todos os valores acima de 80% de reconhecimento de programas. No entanto, a especificidade apresenta valores relativamente baixos em alguns casos, como por exemplo no canal Syfy.

Conforme referido anteriormente, este método foi aplicado também a logos opacos. No entanto, em comparação com os melhores resultados em logos opacos apresentados na tabela , o algoritmo dos logos opacos mostrou ser mais eficiente e vantajoso.

Informação do Vídeo			Classificação		Resultados	
Canal	Resolução	Total Frames	Sens.	Especif.	R_{SE}	Tempo
AMC	720 x 576	8953	0.95732	0.92063	0.88134	189.3727
Antena 3	1280 x 720	8950	0.98214	0.96016	0.94301	525.836
AXN Black HD	1920 x 1080	4755	0.94005	1	0.94005	399.2725
AXN White HD	1920 x 1080	5084	0.98973	0.99361	0.9834	456.8838
AXN HD	1440 x 1080	3332	0.84918	0.989	0.83984	212.0731
Disney Channel	720 x 576	15201	0.9391	0.98867	0.92846	441.6298
Eurosport 1 HD	1920 x 1080	6392	0.92474	1	0.92474	670.7585
FOX Comedy	720 x 576	7139	0.94767	0.99894	0.94667	156.1053
FOX Crime HD	1920 x 1080	32323	0.83425	0.95804	0.79925	2972.2875
FOX Life HD	1920 x 1080	16396	0.87868	0.98968	0.86961	1458.0151
Fox Movies	720 x 576	13976	0.97893	0.99213	0.97123	344.9696
Gazeta TV	768 x 432	8035	0.79694	0.94737	0.755	329.3171
La Sexta 1	1280 x 720	8673	0.98214	0.96016	0.94301	525.836
National Geographic	1280 x 720	10195	0.96613	1	0.96613	400.8327
National Geographic Wild	1280 x 720	6821	0.88159	0.99843	0.88021	326.6453
Odisseia	720 x 576	7143	0.99607	0.98088	0.97703	166.8414
RTP 1	720 x 576	6641	0.85996	0.9945	0.85523	211.3076
RTP 3	720 x 576	8631	0.80693	0.97353	0.78557	180.9636
Sic	720 x 576	10778	0.99986	0.98201	0.98188	263.1122
Sic Caras	720 x 576	5352	0.91929	1	0.91929	142.9872
Sic Mulher	720 x 576	23458	0.94128	0.97484	0.9176	533.0909
Sic Noticias	720 x 576	7079	0.98333	0.76206	0.74936	188.5595
Sic Radical	960 x 540	7623	0.93095	1	0.93095	185.5187
SportTV HD	1440 x 1080	11944	0.99357	0.95528	0.94914	890.7477
SportTV SD1	720 x 576	7932	0.99264	0.81019	0.80423	182.4486
Syfy	720 x 576	7138	0.96482	0.92056	0.88818	140.9859
TVCine 3	1280 x 720	6287	0.97561	1	0.97561	216.253
TVI	720 x 576	15537	0.66814	0.95148	0.63572	334.2795
TVI 24	720 x 576	7156	0.83664	0.93391	0.78134	159.4554

Tabela 5.6: Resultados para o Teste 2

A redução dos valores dos parâmetros de entrada do método, permitiu equilibrar os valores de sensibilidade e especificidade garantindo um resultado com melhor desempenho. Esse resultado pode ser visualizado na tabela 5.6.

Em comparação com o Teste 1, conseguiu-se aumentar substancialmente a especificidade para valores acima de 75.5 %. No entanto, em alguns casos, verificou-se uma ligeira redução dos valores de sensibilidade.

Por outro lado, como seria de esperar, os tempos de reconhecimento também melhoraram, isso deve-se sobretudo ao facto de termos reduzido o raio de procura r em duas unidades.

Os resultados obtidos no Teste 2, nos valores de sensibilidade e especificidade, demonstram que este algoritmo pode ser aplicado aos logos do tipo opaco e semi-transparente. Esse facto, torna este algoritmo mais útil de ser aplicado num sistema de reconhecimento

automático.

5.2 Reconhecimento de Separadores

Nesta secção, serão apresentados as condições de teste para os métodos de reconhecimento de separadores: CLD, CH, DCD.

O primeiro passo, após a extração das frames de cada canal, é identificar quais as sequências de vídeo que são separadores. A cada sequência de vídeo está associada uma divisão por classes. Todas as classes estão representadas a 8 *frames* por segundo. Este valor, definido empiricamente, como sendo um *trade-off* que privilegia uma solução computacionalmente mais eficiente não abdicando das frames necessárias para representar um separador.

Após o processo de identificação dos separadores, foram importados para a base de dados as sequências de vídeo de cada canal, organizadas por classes.

Neste contexto, a sensibilidade mede a qualidade de separadores correctamente identificados.

As sub-secções (5.2.1, 5.2.2, 5.2.3) mostram os resultados para cada método proposto para o reconhecimento de separadores.

5.2.1 Descritor de Layout de Cor

As tabelas 5.7 e 5.8 representam os resultados obtidos para o método proposto baseado no descritor de Layout de Cor em configuração de Threshold = 10, 20, respectivamente.

Destaca-se em ambas as tabelas de resultados valores de especificidade elevados, mais precisamente em 5.7 o que, apesar deste método se focar no reconhecimento de separadores, ou seja, conteúdos de publicidade, garante que outro conteúdo do tipo programa não é incorrectamente identificado.

A sensibilidade obtida na tabela 5.8, apresentou melhores resultados em praticamente todos os casos de teste, reduzindo o número de falsos positivos.

Em ambos os casos de teste, os tempos de reconhecimento foram semelhantes. Repara-se que, em média, é possível analisar 26 frames por segundo em conteúdos SD e 8 frames por segundo em conteúdos HD do stream.

Informação do Vídeo			Separador		Classificação		Resultados	
Canal	Resolução	Frames	Frames	Classes	Sens.	Especif.	R_{SE}	Tempo
Disney Channel	720 x 576	15274	137	6	0.75	0.99928	0.74946	573.8724
Fox Crime HD	1920 x 1080	32382	170	5	0.91892	0.98629	0.90632	5088.8854
Fox Life HD	1920 x 1080	16595	290	9	0.95238	0.99891	0.95135	2589.5935
RTP 1	720 x 576	6700	132	3	0.83333	1	0.83333	253.8076
RTP 3	720 x 576	8690	108	3	0.94872	0.99965	0.94839	323.2648
Sic	720 x 576	10837	114	4	0.60377	0.99926	0.60333	387.0962
Sic Mulher	720 x 576	23517	157	3	0.65079	0.99936	0.65038	836.8913
Sic Radical	960 x 540	7862	116	2	0.72222	1	0.72222	323.8079
Sport TV 1 SD	720 x 576	7991	130	4	0.96296	0.99874	0.96175	273.6357
TVI	720 x 576	15596	84	3	0.69091	1	0.69091	522.5193

Tabela 5.7: Resultados para Threshold = 10

Informação do Vídeo			Separador		Classificação		Resultados	
Canal	Resolução	Frames	Frames	Classes	Sens.	Especif.	R_{SE}	Tempo
Disney Channel	720 x 576	15274	137	6	0.74194	0.99915	0.7413	619.1169
Fox Crime HD	1920 x 1080	32382	170	5	0.96622	0.85137	0.82261	5284.1205
Fox Life HD	1920 x 1080	16595	290	9	0.95238	0.99867	0.95112	1419.0838
RTP 1	720 x 576	6700	132	3	0.97222	0.99761	0.9699	206.465
RTP 3	720 x 576	8690	108	3	1	0.99936	0.99936	255.7651
Sic	720 x 576	10837	114	4	0.66038	0.9992	0.65985	348.7163
Sic Mulher	720 x 576	23517	157	3	0.66667	0.99928	0.66618	786.2818
Sic Radical	960 x 540	7862	116	2	0.88889	1	0.88889	341.2929
Sport TV 1 SD	720 x 576	7991	130	4	1	0.95082	0.95082	287.3566
TVI	720 x 576	15596	84	3	0.87273	0.99395	0.86745	554.1647

Tabela 5.8: Resultados para Threshold = 20

5.2.2 Histograma de cor

Na tabela de configuração 5.9 sugere-se dois casos de teste a serem treinados. Um deles usando a totalidade dos 8 bits para representar uma cor com um limiar de 95% e, o outro com apenas 4 bits/cor e com um limiar de semelhança ligeiramente maior, de 97%.

TesteID	nbins	T_semelhanca
Teste 1	256	0.95
Teste 2	128	0.97

Tabela 5.9: Valores de configuração do método Histograma de cor

Para cada um dos TestesID referidos em 5.9 foram obtidos os resultados presentes nas tabelas 5.10 e 5.11. Começando por analisar o primeiro, denota-se valores de especificidade em geral acima de 97%, à semelhança do que foi demonstrado no método anterior em 5.2.1.

Do mesmo modo, a sensibilidade apresentou valores idênticos, registrando igualmente o pior caso para os canais da Sic e Sic Mulher.

Para o Teste 2, em que se reduziu o número de bins para metade e aumentou em 2% o grau de semelhança, os resultados na maioria dos canais não demonstram ser satisfatórios a reconhecer a presença de separadores, em comparação com o Teste 1. No entanto, apresenta melhores valores de especificidade.

Repare-se que este método, em comparação com o apresentado na secção 5.2.1, teve um tempo de execução mais eficiente, permitindo que os canais SD fossem analisados em média a 47 frames/segundo e os canais HD a 12 frames/segundo.

Informação do Vídeo			Separador		Classificação		Resultados	
Canal	Resolução	Frames	Frames	Classes	Sens.	Especif.	R_{SE}	Tempo
Disney Channel	720 x 576	15274	137	6	0.77419	0.97376	0.75388	353.9454
Fox Crime HD	1920 x 1080	32382	170	5	0.93919	0.99407	0.93362	2735.5909
Fox Life HD	1920 x 1080	16595	290	9	1	0.99988	0.99988	1337.4231
RTP 1	720 x 576	6700	132	3	0.91667	1	0.91667	139.7996
RTP 3	720 x 576	8690	108	3	0.66667	0.99988	0.66659	145.6055
Sic	720 x 576	10837	114	4	0.67925	0.97162	0.65997	230.53
Sic Mulher	720 x 576	23517	157	3	0.66667	0.99906	0.66604	572.058
Sic Radical	960 x 540	7862	116	2	0.72222	1	0.72222	170.8976
Sport TV 1 SD	720 x 576	7991	130	4	0.92593	0.99229	0.91879	165.1944
TVI	720 x 576	15596	84	3	0.98182	1	0.98182	285.825

Tabela 5.10: Resultados para o Teste 1

Informação do Vídeo			Separador		Classificação		Resultados	
Canal	Resolução	Frames	Frames	Classes	Sens.	Especif.	R_{SE}	Tempo
Disney Channel	720 x 576	15274	137	6	0.77419	0.99475	0.77013	363.3447
Fox Crime HD	1920 x 1080	32382	170	5	0.93243	0.99528	0.92804	3127.1342
Fox Life HD	1920 x 1080	16595	290	9	1	0.99988	0.99988	1419.0838
RTP 1	720 x 576	6700	132	3	1	1	1	154.4876
RTP 3	720 x 576	8690	108	3	0.66667	1	0.66667	165.7254
Sic	720 x 576	10837	114	4	0.30189	0.99954	0.30175	264.2439
Sic Mulher	720 x 576	23517	157	3	0.66667	0.99928	0.66618	543.6084
Sic Radical	960 x 540	7862	116	2	0.72222	1	0.72222	212.9573
Sport TV 1 SD	720 x 576	7991	130	4	0.92593	0.99406	0.92042	186.3307
TVI	720 x 576	15596	84	3	0.89091	1	0.89091	324.8451

Tabela 5.11: Resultados para o Teste 2

5.2.3 Descritor de Cor Dominante

No Descritor de Cor Dominante, os parâmetros de entrada como o número de cores dominantes N , o número de bins n_{bins} e o número de blocos n_{blocks} foram testados conforme

indicado na tabela 5.12. Este descritor consiste em encontrar as N cores que predominam na imagem, sabendo que quanto maior for o número de bins, melhor qualidade terá a imagem contudo, menos eficiente será a procura.

TesteID	N	nbins	nblocks
Teste 1	8	64	16
Teste 2	6	16	16

Tabela 5.12: Paramêtros de configuração do método Descritor de Cor Dominante

Por observação das tabelas de resultados 5.13, 5.14 para este método, destaca-se negativamente, o tempo de execução do algoritmo. Para além disso, os valores de sensibilidade para o Teste 1 encontram-se, em alguns canais, abaixo dos 50%. Esta situação, justifica-se com o facto do número de bins considerado permitir milhares de cores diferentes o que, em separadores cuja representação cromática é muito diversa, pequenas variações de cor podem trazer grandes alterações ao nível das cores dominantes. Os casos em que, ainda assim, se registaram bons reconhecimentos são aqueles em que os separadores são, na sua grande maioria, compostos por pouca diversidade de cores.

Reduzindo o número de bins e consequentemente o leque de cores disponíveis, bem como diminuindo o número de cores dominantes, no caso do Teste 2 trouxe melhores resultados de sensibilidade. No entanto, foi verificada uma ligeira redução dos valores da especificidade.

Em suma, este método demonstrou ser menos eficiente face aos outros dois apresentados na secções (5.2.2, 5.2.1). Para além disso, os seus valores de sensibilidade e especificidade também foram inferiores.

Informação do Vídeo			Separador		Classificação		Resultados	
Canal	Resolução	Frames	Frames	Classes	Sens.	Especif.	R_{SE}	Tempo
Disney Channel	720 x 576	15274	137	6	0.35714	0.99403	0.35501	1361.9405
Fox Crime HD	1920 x 1080	32382	170	5	0.83108	0.9235	0.7675	9458.418
Fox Life HD	1920 x 1080	16595	290	9	1	0.99867	0.99867	4712.5797
RTP 1	720 x 576	6700	132	3	1	0.9952	0.9952	610.9013
RTP 3	720 x 576	8690	108	3	0.74359	0.96729	0.71926	255.7651
Sic	720 x 576	10837	114	4	0.37736	0.99063	0.37382	931.7082
Sic Mulher	720 x 576	23517	157	3	0.61905	0.97804	0.60545	786.2818
Sic Radical	960 x 540	7862	116	2	0.5	1	0.5	707.3918
Sport TV 1 SD	720 x 576	7991	130	4	0.79012	0.99305	0.78463	755.6678
TVI	720 x 576	15596	84	3	0.23636	0.99826	0.23595	1363.85

Tabela 5.13: Resultados do Descritor de Cor Dominante para o Teste 1

Informação do Vídeo			Separador		Classificação		Resultados	
Canal	Resolução	Frames	Frames	Classes	Sens.	Especif.	R_{SE}	Tempo
Disney Channel	720 x 576	15274	137	6	0.75	0.91716	0.74946	1438.9242
Fox Crime HD	1920 x 1080	32382	170	5	0.87838	0.57204	0.50246	9638.4721
Fox Life HD	1920 x 1080	16595	290	9	1	0.99222	0.99222	5060.3696
RTP 1	720 x 576	6700	132	3	1	0.99025	0.99025	603.727
RTP 3	720 x 576	8690	108	3	1	0.95989	0.95989	768.1083
Sic	720 x 576	10837	114	4	0.88679	0.87519	0.77611	988.3342
Sic Mulher	720 x 576	23517	157	3	0.84127	0.94918	0.79851	2134.0984
Sic Radical	960 x 540	7862	116	2	0.77778	0.95512	0.74287	773.4826
Sport TV 1 SD	720 x 576	7991	130	4	0.96296	0.70809	0.68187	704.2953
TVI	720 x 576	15596	84	3	0.92727	0.86314	0.80036	1405.5823

Tabela 5.14: Resultados do Descritor de Cor Dominante para o Teste 2

5.2.4 Análise Crítica

Nas secções (5.2.1, 5.2.2, 5.2.3) foram apresentados os resultados para o reconhecimento de separadores de vídeo. Cada um dos métodos propostos nessas secções mostrou-se apto para reconhecer separadores.

Considerando os melhores resultados de cada método proposto, referidos nas respectivas secções, o histograma de cor mostrou-se ser a solução mais eficiente, apresentando valores de sensibilidade e especificidade equivalentes ao descritor de layout de cor. Contrariamente ao descritor de cor dominante, que registou os tempos de reconhecimento mais elevados.

6 Conclusão e Trabalho Futuro

6.1 Conclusão

Esta dissertação apresentou vários métodos de identificação de conteúdos de vídeo baseados em técnicas de visão por computador e processamento de imagem. Estes algoritmos foram desenvolvidos com intenção de serem aplicados numa box da WeTek, aumentando as suas funcionalidades. Apesar da implementação ter sido concebida em MATLAB, considera-se que numa implantação em C poderá trazer ganhos de desempenho, sensivelmente, dez vezes superior.

Os testes realizados evidenciam que o sistema tem um desempenho de reconhecimento de programas e separadores de vídeo superior a 90%, numa amostragem significativa de 29 canais. No entanto, reconhece-se a complexidade de adoptar um sistema que se adapte a qualquer canal.

6.2 Trabalho Futuro

Numa perspectiva de continuação do trabalho, sugere-se que seja aplicado em paralelo ao reconhecimento de separadores uma abordagem baseada em áudio. Sabe-se que, na grande maioria dos casos, o som de entrada dos programas e dos separadores é igual. Essa característica possibilita que estes possam ser treinados, guardados e, posteriormente comparados, aumentando a confiança do reconhecimento.

Do mesmo modo, seria interessante explorar técnicas de aprendizagem automática no reconhecimento de programas e publicidade, com base em características previamente treinadas.

Conforme foi realçado anteriormente, os métodos desenvolvidos não só permitem reconhecer a presença de logos como de outros objectos que permanecem estacionariamente ao longo do tempo. Para o efeito, podem ser testados em simultâneo o reconhecimento deste

tipo de objectos. Para além disso, certo tipo de conteúdo pode apresentar legendas, um atributo encontrado em programas.

Por outro lado, seria relevante classificar o conteúdo de vídeo quanto à categoria a que pertence, nomeadamente, se é desporto, notícias, filmes ou séries, infantil, etc. Essa avaliação poderá traçar o perfil de utilizador, clarificando os seus conteúdos de interesse.

Por fim, implementar os métodos que foram propostos nesta dissertação na box da WeTek.

7 Bibliografia

- [1] B. Satterwhite and O. Marques. Automatic detection of tv commercials. *IEEE Potentials*, 23(2):9–12, April 2004.
- [2] Ying Li and C.-C. Jay Kuo. Detecting commercial breaks in real tv programs based on audiovisual information, 2000.
- [3] X. Wang and Z. Guo. A novel real-time commercial detection scheme. In *2008 3rd International Conference on Innovative Computing Information and Control*, pages 536–536, June 2008.
- [4] R. Lienhart, C. Kuhmunch, and W. Effelsberg. On the detection and recognition of television commercials. In *Proceedings of IEEE International Conference on Multimedia Computing and Systems*, pages 509–516, Jun 1997.
- [5] S. S. Cheung and A. Zakhor. Efficient video similarity measurement with video signature. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(1):59–74, Jan 2003.
- [6] Y. Chen, W. Hu, X. Zeng, and W. Li. Indexing and matching of video shots based on motion and color analysis. In *2006 9th International Conference on Control, Automation, Robotics and Vision*, pages 1–6, Dec 2006.
- [7] Kaashoek. Comskip. <http://www.kaashoek.com/comskip>. [Online; accessed 24-November-2016].
- [8] Moving Picture Experts Group. Mpeg | the moving picture experts group. <http://mpeg.chiariglione.org>. [Online; accessed 02-January-2017].
- [9] B. S Manjunath, Phillipe Salembier, and Thomas Sikora. *Introduction to MPEG-7*. Wiley, 1 edition, 2002.

-
- [10] H. Shao, Y. Wu, W. Cui, and J. Zhang. Image retrieval based on mpeg-7 dominant color descriptor. In *2008 The 9th International Conference for Young Computer Scientists*, pages 753–757, Nov 2008.
- [11] Ahmed Talib, Massudi Mahmuddin, Husniza Husni, and Loay E. George. A weighted dominant color descriptor for content-based image retrieval. *Journal of Visual Communication and Image Representation*, 24(3):345 – 360, 2013.
- [12] Nai-Chung Yang, Wei-Han Chang, Chung-Ming Kuo, and Tsia-Hsing Li. A fast mpeg-7 dominant color extraction with new similarity measure for image retrieval. *Journal of Visual Communication and Image Representation*, 19(2):92 – 105, 2008.
- [13] R. Chakravarti and X. Meng. A study of color histogram based image retrieval. In *2009 Sixth International Conference on Information Technology: New Generations*, pages 1323–1328, April 2009.
- [14] Sung-Hyuk Cha. Comprehensive survey on distance/similarity measures between probability density functions, 2007.
- [15] J. Wang, L. Duan, Z. Li, J. Liu, H. Lu, and J. S. Jin. A robust method for tv logo tracking in video streams. In *2006 IEEE International Conference on Multimedia and Expo*, pages 1041–1044, July 2006.
- [16] Alex Reis Dos Santos and Hae Yong Kim. Real-time opaque and semi-transparent tv logos detection. In *In WACV*, 2007.
- [17] Nir Friedman and Stuart Russell. Image segmentation in video sequences: A probabilistic approach. In *Proceedings of the Thirteenth Conference on Uncertainty in Artificial Intelligence*, UAI’97, pages 175–181, San Francisco, CA, USA, 1997. Morgan Kaufmann Publishers Inc.
- [18] Stefan Duffner and Christophe Garcia. A neural scheme for robust detection of transparent logos in tv programs. In *Proceedings of the 16th International Conference on Artificial Neural Networks - Volume Part II*, ICANN’06, pages 14–23, Berlin, Heidelberg, 2006. Springer-Verlag.
- [19] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Comput. Vis. Image Underst.*, 110(3):346–359, June 2008.

- [20] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. *SURF: Speeded Up Robust Features*, pages 404–417. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.
- [21] Jerry D. Gibson and Al Bovik, editors. *Handbook of Image and Video Processing*. Academic Press, Inc., Orlando, FL, USA, 1st edition, 2000.
- [22] Yiannis Kompatsiaris, Bernard Merialdo, and Shiguo Lian. *TV content analysis: Techniques and applications*. Auerbach Publications, October 2011, ISBN: 9781439855607, 10 2011.