Instituto de Computação
Universidade Estadual de Campinas

Detecção Rápida de Legendas em Vídeos Utilizando o Ritmo Visual

Felipe Braunger Valio

Este exemplar corresponde à redação final da Dissertação devidamente corrigida e defendida por Felipe Braunger Valio e aprovada pela Banca Examinadora.

Campinas, 27 de setembro de 2011.

Prof. Dr. Neucimar J. Leite (Orientador)

Prof. Dr. Hélio Pedrini (Co-orientador)

Dissertação apresentada ao Instituto de Computação, UNICAMP, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação.

FICHA CATALOGRÁFICA ELABORADA POR
MARIA FABIANA BEZERRA MÜLLER - CRB8/6162
BIBLIOTECA DO INSTITUTO DE MATEMÁTICA, ESTATÍSTICA E
COMPUTAÇÃO CIENTÍFICA - UNICAMP

V238d

Valio, Felipe Braunger, 1984-
Detecção rápida de legendas em vídeos utilizando o ritmo visual / Felipe Braunger Valio. - Campinas, SP : [s.n.], 2011.

Orientador: Neucimar Jerônimo Leite.
Coorientador: Hélio Pedrini.
Dissertação (mestrado) – Universidade Estadual de Campinas, Instituto de Computação.

1. Processamento de textos (Computação). 2. Indexação automática. 3. Rastreamento automático. I. Leite, Neucimar Jerônimo, 1961-. II. Pedrini, Hélio, 1963-. III. Universidade Estadual de Campinas. Instituto de Computação. IV.Título.

Informações para Biblioteca Digital

Título em inglês: Fast video caption detection based on visual rhythm
Palavras-chave em inglês:
Text processing (Computer science)
Automatic indexing
Automatic tracking
Área de concentração: Ciência da Computação
Titulação: Mestre em Ciência da Computação
Banca examinadora:
Hélio Pedrini [Coorientador]
Leyza Baldo Dorini
Ricardo da Silva Torres
Data da defesa: 29-07-2011
Programa de Pós Graduação: Ciência da Computação

TERMO DE APROVAÇÃO

Dissertação Defendida e Aprovada em 29 de julho de 2011, pela Banca examinadora composta pelos Professores Doutores:

Leyza Baldo Dorini

Profª. Drª. Leyza Elmeri Baldo Dorini
DAINF / UTFPR

Ricardo Torres

Prof. Dr. Ricardo da Silva Torres
IC / UNICAMP

Hélio Pedrini

Prof. Dr. Hélio Pedrini
IC / UNICAMP

Instituto de Computação
Universidade Estadual de Campinas

Detecção Rápida de Legendas em Vídeos Utilizando o Ritmo Visual

Felipe Braunger Valio

Julho de 2011

Banca Examinadora:

- Prof. Dr. Neucimar J. Leite (Orientador)
- Prof. Dr. Hélio Pedrini (Co-orientador)
- Profa. Dra. Leyza Baldo Dorini
  Departamento Acadêmico de Informática – UTFPR
- Prof. Dr. Ricardo da Silva Torres
  Instituto de Computação – UNICAMP

Resumo

Detecção de textos em imagens é um problema que vem sendo estudado a várias décadas. Existem muitos trabalhos que estendem os métodos existentes para uso em análise de vídeos, entretanto, poucos deles criam ou adaptam abordagens que consideram características inerentes dos vídeos, como as informações temporais. Um problema particular dos vídeos, que será o foco deste trabalho, é o de detecção de legendas. Uma abordagem rápida para localizar quadros de vídeos que contenham legendas é proposta baseada em uma estrutura de dados especial denominada ritmo visual. O método é robusto à detecção de legendas com respeito ao alfabeto utilizado, ao estilo de fontes, à intensidade de cores e à orientação das legendas. Vários conjuntos de testes foram utilizados em nosso experimentos para demonstrar a efetividade do método.

Abstract

Detection of text in images is a problem that has been studied for several decades. There are many works that extend the existing methods for use in video analysis, however, few of them create or adapt approaches that consider the inherent characteristics of video, such as temporal information. A particular problem of the videos, which will be the focus of this work, is the detection of subtitles. A fast method for locating video frames containing captions is proposed based on a special data structure called visual rhythm. The method is robust to the detection of legends with respect to the used alphabet, font style, color intensity and subtitle orientation. Several datasets were used in our experiments to demonstrate the effectiveness of the method.

Sumário

Resumo **v**

Abstract **vi**

1 Introdução **1**
1.1 Apresentação 1
1.2 Motivações 3
1.3 Contribuições 3
1.4 Organização do Texto 4

2 Trabalhos e Conceitos Relacionados **5**
2.1 Detecção de Textos em Imagens 5
2.1.1 Trabalhos Relacionados 6
2.2 Detecção de Legendas em Vídeos 7
2.2.1 Legendas em Vídeos 7
2.2.2 Trabalhos Relacionados 8
2.2.3 Notas sobre Legendas 10
2.3 Ritmo Visual 11
2.3.1 Definição 11
2.3.2 Trabalhos sobre Ritmo Visual 13
2.3.3 Características do Ritmo Visual 14
2.4 Curvas de Preenchimento do Espaço 15
2.4.1 Curva de Hilbert 16
2.4.2 Curva Zig-Zag 17
2.4.3 Linha Vertical 18
2.4.4 Outras Curvas 19

3 Ritmo Visual e Detecção de Legendas **21**
3.1 Obtenção do Ritmo Visual 21
3.2 Segmentação 26

3.3 Classificação 28
3.4 Análise de Complexidade 30

4 Experimentos e Resultados **31**
4.1 Requisitos 31
4.2 Análise de Desempenho 33
4.2.1 Comparação com outros métodos 33
4.2.2 Análise de desempenho do método 34
4.3 Análise da Acurácia 34
4.3.1 Comparação com outros métodos 38
4.4 Discussão 43

5 Conclusões e Trabalhos Futuros **48**
5.1 Conclusões 48
5.2 Trabalhos Futuros 48

Bibliografia **50**

Lista de Tabelas

2.1 Agrupamento de alguns trabalhos encontrados na literatura de acordo com certas características. 11

4.1 Descrição de algumas informações de cada vídeo existente na base de testes, como resolução e idioma. 32
4.2 Especificação dos casos de testes utilizados nos experimentos. 32
4.3 Comparação da taxa de quadros por segundo. 33
4.4 Taxa de quadros por segundo para o cálculo do ritmo visual utilizando a curva zig-zag. 34
4.5 Taxa de quadros por segundo para o cálculo do ritmo visual utilizando a curva de Hilbert. 37
4.6 Taxa total de quadros por segundo para a curva zig-zag, incluindo as etapas de segmentação e classificação. 37
4.7 Taxa de quadros por segundo para a curva de Hilbert, incluindo as etapas de segmentação e classificação. 38
4.8 Resultados de precisão (P), revocação (R) e *F-Measure* (FM), tanto para os três métodos utilizados para comparação quanto para o método proposto. Para este último, apenas o caso de teste 2 foi considerado por simplicidade e economia de espaço. Para cada vídeo, a melhor taxa de *F-Measure* obtida foi destacada em negrito. 39
4.9 Resultados para os três casos de testes que utilizam a curva zig-zag. 41
4.10 Resultados para os três casos de testes que utilizam a curva de Hilbert. . . 42
4.11 Erro médio da localização dos pontos extremos. Cada valor representa a diferença média entre pontos extremos da legenda e os detectados. Os casos em que não foi encontrada nenhuma correspondência com distância menor que 5 quadros foram marcados com $\times$. 43
4.12 Erro médio, em quadros, da localização dos pontos extremos para cada um dos métodos utilizados na comparação. 43

Lista de Figuras

1.1 Exemplos de legendas em vídeos. 2

2.1 Exemplo de como um vídeo pode ser cortado. Se realizado um corte como em (a), um quadro do vídeo será obtido. Entretanto, se realizado como em (b), o ritmo visual deste vídeo será obtido. 12
2.2 Aplicação, sobre um quadro, dos cortes horizontal, vertical e diagonal, definidos pelas Equações 2.1, 2.2 e 2.3, respectivamente. 13
2.3 Exemplo de um ritmo visual. Este ritmo foi construído utilizando como corte uma linha vertical central, aplicado em um vídeo de 872 quadros, cada um com dimensão de 448×336 pixels. 14
2.4 Algumas propriedades do ritmo visual em destaque. 15
2.5 Exemplo de duas orientações nas quais uma legenda pode ocorrer. Em (a), qualquer um dos três cortes realizados consegue intersectar a legenda. Em (b), apenas um dos cortes se mostra efetivo para este fim. Esses dois exemplos demonstram como pode não ser eficaz utilizar curvas com a mesma orientação das legendas. 16
2.6 A curva de Hilbert para as cinco primeiras aproximações. 17
2.7 Variações da curva em zig-zag. 17
2.8 A curva zig-zag pode ser construída seguindo dois sentidos diferentes. . . . 18
2.9 A curva zig-zag para as escalas 1, 3, 5, 7 e 9, respectivamente. 18
2.10 Exemplos de seis curvas que foram descartadas por não possuírem características desejáveis à deteção de legendas. 20

3.1 Fluxo básico do método proposto. 21
3.2 Uma imagem em (a) sendo particionada em (b) 3×3 e (c) 5×5 blocos. . . 23
3.3 Um quadro particionado em 4 blocos. Cada curva, que possui comprimento h, produz um ritmo visual. A união destes gera o ritmo visual final, com altura $4 * h$. 23

3.4 A figura (a) ilustra um fragmento de texto (“E!”) ampliado. Pode-se notar a suavização dos caracteres devido ao uso de *anti-aliasing*. Na figura (b), um exemplo de legenda no ritmo visual que foi intersectada justamente em uma borda suavizada, gerando um padrão incompleto. 24
3.5 Exemplos da melhora de qualidade do ritmo visual ao aplicar uma dilatação. Em (a) e (c) estão exemplos de ritmos visuais e, em (b) e (d), o resultado após dilatar a curva por um elemento estruturante quadrado de tamanho 3. Todos os exemplos foram contruídos com uma curva zig-zag com escala 1. 25
3.6 Um exemplo em (a) de ritmo visual no espaço de cor RGB. As demais figuras representam isoladamente as bandas do espaço de cor HSV. Pode-se notar que as legendas não ficaram visíveis na banda de matiz. Na banda de saturação, na parte inferior, surgiram artefatos de compressão (retângulos brancos), além do fato de que as legendas ficaram com a mesma cor do fundo (preto). A banda de intensidade, por outro lado, é a que melhor destaca as legendas e não acrescenta artefatos. 27
3.7 As figuras (a), (b) e (c) ilustram três regiões conexas. As figuras (d), (e) e (f) apresentam seus respectivos afinamentos. Como a região em (a) é retangular, o resultado do afinamento é uma reta horizontal, enquanto que o afinamento de (b) é uma linha irregular. Em (c), pode-se observado que o afinamento se deteriora conforme a região geradora se distancia de um retângulo. 29

4.1 Os gráficos mostram como o comprimento de cada curva se comporta de acordo com a escala da mesma. No eixo das ordenadas é exibida a razão entre o comprimento em pixels da curva e o total de pixels da imagem. . . 35
4.2 Comprimento das curvas de Hilbert e zig-zag, em pixels, quando são variados o grau de detalhamento das curvas e o número de blocos em que cada imagem é particionada. 36
4.3 Gráfico de precisão × revocação para variações da razão de aspecto (os números que acompanham a curva são os valores da razão). 40
4.4 Em (a), exemplo de um ritmo visual construído a partir de uma curva de complexidade alta. Em (b), o resultado após a classificação deste ritmo. As regiões cinzas representam as legendas encontradas no ritmo visual, entretanto, todas estão incorretas uma vez que não há legendas no vídeo em questão. 45

4.5 Em (a), um ritmo visual extraído de uma cena em que não há movimento de câmera nem de grande parte dos objetos. Em (b), o resultado da classificação, tal que uma faixa foi detectada como legenda, gerando um falso positivo. 46
4.6 Exemplo de legenda com possibilidade de se fundir com uma reta vertical de intensidade de cor semelhante. 47
4.7 Dois exemplos de legendas que não respeitam a exigência de intervalo de tempo. Em (a), houve uma sobreposição entre três legendas. Em (b), o problema foi evitado devido à diferença de altura das legendas no ritmo visual. 47

Capítulo 1

Introdução

1.1 Apresentação

Aplicações envolvendo vídeos estão se tornando cada vez mais comuns, impulsionadas por fatores como a popularização da televisão digital, aumento das taxas de transmissão de dados via Internet, evolução das pesquisas de realidade aumentada, desenvolvimento de dispositivos móveis voltados às aplicações multimídia, baixo custo das unidades de armazenamento de dados, entre outros.

Textos em vídeos podem prover importantes informações do conteúdo dos mesmos. Por exemplo, pequenas porções nos cantos da tela são comumente utilizados em noticiários para exibir informações como previsão do tempo, resultados de jogos, manchetes, nomes de pessoas, informações do mercado. Legendas são utilizadas em documentários para descrever uma localidade, pessoa, título ou evento. Quase todos os comerciais usam algum tipo de texto para prover mais informações sobre os produtos, uma vez que seu tempo de exibição é restrito a apenas alguns segundos. Em filmes, legendas são abundantemente utilizadas quando o idioma falado não é o nativo do seu local de exibição ou então para informar seu elenco. Há muitos sistemas para tradução automática de textos, navegação baseada em informação textual, indexação de bibliotecas multimídia, detecção de eventos como aparição de comerciais, blocos de noticiários, entre outros. A Figura 1.1 exibe alguns exemplos dos casos mencionados.

Para efetuar a tradução de uma legenda, primeiramente a sua localização deve ser obtida dentro de cada quadro. Este processo é chamado de detecção da legenda. A partir do momento em que o uso de vídeos se tornou difundido, muitas técnicas de detecção de legendas começaram a ser desenvolvidas.

Apesar do contínuo aperfeiçoamento dos recursos computacionais, analisar um vídeo ainda é uma tarefa custosa, devido ao grande volume de informações visuais a ser tratado. Desta forma, um método de detecção de legendas deve preferencialmente fazer uso de

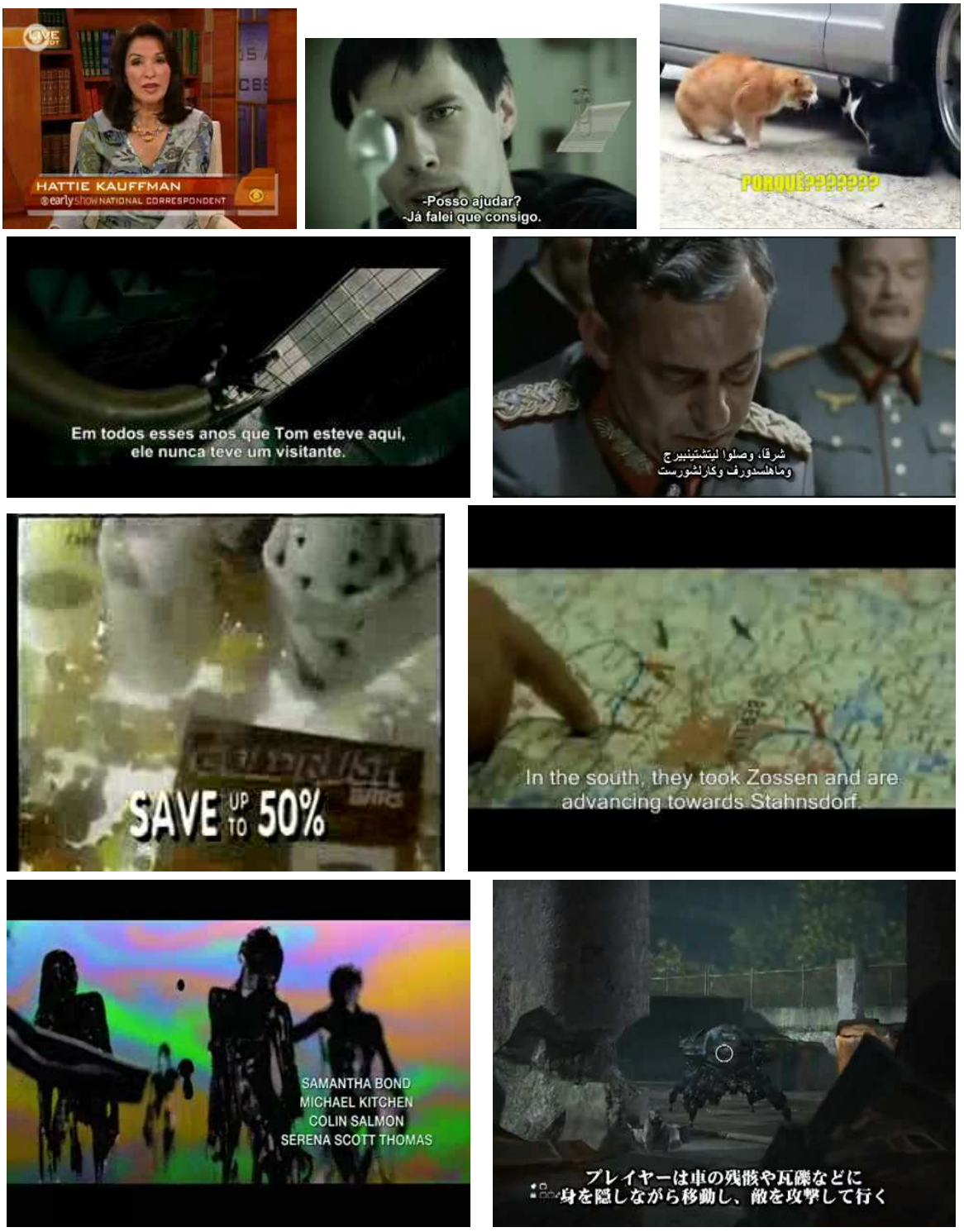


Figura 1.1: Exemplos de legendas em vídeos.

meios para agilizar o processo.

Uma forma de se reduzir o custo computacional é descartar trechos dos vídeos que certamente não possuem legendas, a fim de não desperdiçar processamento com os mesmos. Podemos fazer uso de uma variação do problema de detecção de legendas, que é o de detectar quadros que possuam legendas. Esta abordagem não requer descobrir a posição nos quadros onde a legenda ocorre, mas sim determinar apenas se um quadro possui ou não legenda.

O trabalho proposto nesta dissertação descreve e valida um método rápido para detecção de quadros que contenham legendas, utilizando para isso uma representação de dados denominada *ritmo visual*. O método é invariante a vários fatores, como o idioma ou alfabeto utilizado nas legendas, tipo e cor de fonte e orientação dos textos. O método se baseia no uso de curvas de preenchimento do espaço para construir ritmos visuais adequados, de forma que regiões de interesse possam ser extraídas e classificadas.

1.2 Motivações

A literatura possui muitas publicações sob o tema de detecção de legendas em vídeos, entretanto, grande parte desses trabalhos se limita a aplicar nos vídeos métodos existentes de detecção de textos em imagens. Apesar dessa abordagem resolver, em teoria, o problema, propriedades inerentes aos vídeos, como informação temporal, não são exploradas.

Além disso, a maioria dos trabalhos propostos atinge taxas de processamento bastante baixas, da ordem de apenas um ou dois quadros por segundo, contra a taxa de trinta quadros por segundo na qual os vídeos são exibidos. Neste sentido, conclui-se que há uma carência de métodos que consigam obter boas taxas de desempenho e que, ao mesmo tempo, não percam acurácia em detecção.

Ao mesmo tempo, conhecemos o ritmo visual, uma estrutura de dados que possibilita diversas análises de vídeos de forma ágil e eficaz, que pode ser explorado para o problema de detecção de legendas.

Esta dissertação propõe um método de detecção de quadros com legendas utilizando o ritmo visual para aumentar significantemente a velocidade de processamento, além de obter uma baixa taxa de falsos negativos.

1.3 Contribuições

As principais contribuições deste trabalho são listadas a seguir:

- realizar um estudo sobre o uso de curvas de preenchimento do espaço para a construção do ritmo visual. Nenhum outro trabalho, até então, propôs-se a utilizar outras curvas que não as retas horizontal, vertical ou diagonal.

- explorar o uso do ritmo visual para realizar a detecção de legendas em quadros de vídeos. O método proposto apresentou uma taxa de acurácia acima da média quando comparada com outros métodos da literatura e, ao mesmo tempo, mostrou-se significativamente mais rápido do que as abordagens quadro-a-quadro utilizadas na comparação.

- construir e disponibilizar uma base de vídeos voltada ao problema de detecção de legendas. As bases públicas existentes são focadas na detecção de textos genéricos. Muitos autores elaboram suas próprias bases, porém, não as disponibilizam.

1.4 Organização do Texto

O documento está organizado da seguinte forma: o Capítulo 2 caracteriza, em detalhes, o problema a ser abordado e realiza um resumo da literatura sobre o assunto, além de apresentar a definição do ritmo visual e outros conceitos relacionados. O Capítulo 3 descreve a metodologia proposta. O Capítulo 4 apresenta e discute os resultados experimentais obtidos a partir da metodologia proposta. O Capítulo 5 apresenta as conclusões e sugestões para trabalhos futuros.

Capítulo 2

Trabalhos e Conceitos Relacionados

Este capítulo inicialmente apresenta o problema de detecção de textos em imagens e vídeos. Trabalhos da literatura relacionados a esses problemas são descritos e discutidos. Em seguida, o ritmo visual é apresentado em termos de sua definição, peculiariedades e aplicações. Um estudo também é realizado sobre algumas curvas de preenchimento do espaço para uso na construção do ritmo visual.

2.1 Detecção de Textos em Imagens

O problema de detecção de textos em imagens é amplamente abordado na literatura e consiste em localizar as áreas da imagem que contenham textos. Isso significa, em geral, desenhar retângulos envoltórios (*bounding boxes*) na imagem com o menor tamanho possível ao redor dos textos. Algumas variações do problema procuram envolver cada letra de texto em um retângulo distinto, embora, dependendo do contexto, seja interessante isolar cada palavra, linha ou mesmo um parágrafo inteiro.

Uma característica desejável em um método de deteção é a invariância com respeito à rotação dos textos. Textos em cenas naturais podem conter distorções de perspectiva ou mesmo uma rotação pura, as quais dificultam a sua localização. Métodos robustos devem contornar esse problema.

Um subproblema bastante estudado é o de detecção de textos artificiais. Estes textos são inseridos nas imagens, após a sua obtenção, por meio de alguma ferramenta computacional, como um editor de imagens. Estes textos normalmente possuem características ideais, de forma a não sofrerem com distorções, rotações, oclusão ou iluminação natural. Tipicamente, eles possuem cor constante e nítida, alto contraste e tamanho adequado, uma vez que são inseridos nas imagens de forma a proporcionar uma fácil leitura pelo ser humano.

Em constraste com os textos artificiais, há os textos naturalmente presentes na cena.

As técnicas de localização de textos, em geral, focam neste segundo tipo por serem mais abrangentes. Um método que detecta textos naturais normalmente detecta textos artificiais.

2.1.1 Trabalhos Relacionados

Muitos métodos baseiam-se na sequência: 1) extrair informações da imagem, 2) fornecê-las a um classificador binário, 3) refinar ou formatar os resultados.

Em Chen et al. [5], as bordas verticais e horizontais são extraídas de uma imagem e então dilatações e erosões morfológicas são aplicadas para selecionar regiões candidatas à presença de textos. O mapa de bordas de cada região é então classificado utilizando-se uma máquina de vetores de suporte (SVM) [11]. Em [6], a extração de características é semelhante, no entanto, a classificação é realizada por um algoritmo EM (*Expectation-Maximization*).

Li et al. [28] utilizam wavelets para extrair um vetor de características contendo informações das bordas de uma imagem. Este vetor é classificado empregando-se uma rede neural.

Lienhart [30] faz uma transformação do espaço de cores *RGB* para *Lab*, então reduz a quantidade de cores da imagem supondo que o contraste dos caracteres com o fundo não se perderá, gerando regiões homogêneas que certamente não contêm textos. Um algoritmo de divisão-e-conquista é utilizado para encontrar as regiões que serão fornecidas a um OCR.

O algoritmo de Jui-Chen et al. [52] aplica operações morfológicas para unir os caracteres de cada palavra, extrair os contornos e binarizar, obtendo assim as regiões com caracteres. Um algoritmo simples é utilizado para mesclar regiões que correspondem a uma mesma linha de texto. Estas linhas são verificadas através da análise de projeção horizontal para eliminar falsos positivos. Segundo os autores, esta técnica é parcialmente invariante à rotação.

Em [1] é utilizada apenas a banda vermelha do espaço RGB, considerando-se que esta é suficiente para analisar as cores branca, amarela e preta, que são as mais comumente usadas na criação de textos. Uma detecção de arestas seguida de uma binarização com limiar global é aplicada para extrair as regiões de interesse. As regiões conexas dos pixels são construídas, filtradas e mescladas para gerar as caixas de textos.

O método de Wu et al. [53] é baseado em segmentação por textura. Características de textura são computadas para cada pixel em diferentes escalas. O algoritmo *K-means* é utilizado para separar essas características em três classes: texto, não-texto e incerto. O resultado obtido é muito sensível a ruídos.

Há vários trabalhos (*surveys*) que se dedicam a resumir e comparar as numerosas

técnicas existentes, podendo-se citar o trabalho de Jung et al. [24], em que é realizado um extenso estudo sobre os desafios da detecção de textos em imagens e vídeos, assim como a apresentação de muitas abordagens já publicadas sobre o problema. Outros trabalhos desse tipo também podem ser encontrados em [29] e [32].

Como um resumo geral das técnicas existentes na literatura, tem-se que:

- as características extraídas variam consideravelmente. Alguns trabalhos que operam sobre os gradientes e podem levar em conta a extração de bordas [6], contornos [1, 52] e derivadas [33]. Outros trabalhos operam no espaço de frequência a partir da transformada Fourier ou wavelet [28]. Alguns trabalhos consideram textos como uma textura [19] e, neste caso, utilizam as medidas de Haralick et al. [20] para construir um vetor de características. Informações de cromaticidade também são exploradas [55, 56]. Em [39], uma separação entre imagens de complexidade baixa, média e alta é realizada de acordo com a densidade das bordas. Cada classe é suavizada de forma distinta para a remoção de ruídos.

- os classificadores mais utilizados são as redes-neurais [28] e SVM [5]. Uma das técnicas mais conhecidas é a de Viola e Jones [50] para detecção de faces, que foi adaptada para a detecção de textos. Em [22], a extração de características é uma combinação entre as bases de Haar e LBP (*Local Binary Pattern*) [41] para obter invariância à rotação.

- o pós-processamento, em geral, consiste em desenhar uma janela ou um retângulo envoltório na região contendo texto. Os trabalhos existentes visam, nesta fase, refinar os resultados segundo alguns critérios, como remover janelas muito pequenas ou então janelas que não casam com uma certa proporção entre largura e altura previamente definidas [6].

2.2 Detecção de Legendas em Vídeos

Apesar dos vídeos serem representados de diversas formas, todo vídeo pode ser visto como uma sequência de quadros (imagens) de mesmas dimensões e que são exibidos linearmente a uma taxa fixa. Isso introduz o conceito de taxa de quadros, conhecido como *frame rate* ou *fps* (*frames per second*), que representa o número de quadros efetivamente exibidos a cada segundo.

2.2.1 Legendas em Vídeos

Imagens de vídeos possuem uma informação adicional que é a temporalidade. Esta informação pode ser usada para separar objetos com movimento dos objetos que perma-

necem estáticos durante um determinado intervalo de tempo. Isto é particularmente útil quando se trabalha com uma classe muito peculiar de textos, ou seja, as legendas.

As legendas são utilizadas em muitas aplicações, tais como filmes legendados e comerciais de televisão, onde aparecem telefones de contato e *slogans*, placares de jogos esportivos, noticários mostrando com frequência informações textuais como nomes de pessoas ou manchetes, entre outros. Todas as legendas possuem duas características que as tornam distintas de outros textos: 1) são geralmente isentas de movimento durante um certo intervalo de tempo e 2) foram incluídas artificialmente nas cenas. Essas características permitem uma fácil leitura pelo espectador, havendo um bom contraste das legendas com o fundo das imagens.

2.2.2 Trabalhos Relacionados

Há vários trabalhos publicados sobre detecção de legendas em vídeos e, conforme será visto, a temporalidade é uma informação que não é devidamente explorada por todos. As seções a seguir apresentam alguns trabalhos correlatos, subdividindo-os de acordo com o emprego ou não da informação temporal.

Uso da Temporalidade

O trabalho em [10] propõe um método para agilizar o processo de detecção de legendas ao usar o ritmo visual para pré-selecionar quadros candidatos. Um filtro de Prewitt [8] é aplicado sobre o ritmo visual para extrair gradientes verticais. Como resultado, cada região realçada é considerada uma possível legenda e um quadro é selecionado para sofrer uma análise mais rigorosa com relação à localização de legendas em imagens.

Em [37] é definido um vetor de informações temporais que descreve o comportamento de um único pixel ao longo de um determinado intervalo de tempo. Desse vetor é extraída a transição do início ou término de uma legenda utilizando-se uma limiarização. Apesar do vetor utilizar um conceito semelhante ao do ritmo visual, o método o aplica para todos os pixels de cada quadro, algo que demanda alto processamento. Vale salientar que a validação deste método considera apenas um vídeo de teste.

O trabalho em [3] é focado na detecção de legendas em vídeos esportivos utilizando uma extração de características conhecida como *salient points*. Para melhorar o desempenho do método, a taxa de quadros do vídeo é reduzida para 1/8 do original.

O trabalho em [7] considera e utiliza a base MPEG para agilizar o processo de detecção, criando um esquema de filtragem dos GOP (*Group of Images*) que certamente não possuem legendas. Da mesma forma, o trabalho em [58] utiliza os I-*frames* da base MPEG. Os coeficientes da transformada discreta do cosseno (DCT) são filtrados considerando-se métricas de textura.

Lienhart [34] analisa isoladamente alguns quadros do vídeo e tenta refinar o resultado para os quadros seguintes, supondo que os textos já detectados permanecerão estáticos durante algum período de tempo.

O trabalho em [23] realiza a detecção de legendas em vídeos de partidas de golfe a partir de um descritor de cores do MPEG-7, o *Dominant Color Descriptor* (DCD). Um processo de clusterização é utilizado para descobrir as cores que pertencem às legendas, refinando os *clusters* ao longo dos demais quadros do vídeo.

O trabalho em [47] utiliza uma rede neural para realizar a detecção em noticiários chineses. Uma métrica é proposta para localizar quadros nos quais ocorre uma transição de início ou término de uma legenda.

Demais Trabalhos

A seguir são mencionados alguns trabalhos que não fazem uso da informação temporal das legendas.

Em [52], um conjunto de operadores morfológicos é utilizado para extrair regiões candidatas de cada quadro. Para cada região é considerado seu momento estatístico na estimativa de sua orientação e realização da projeção horizontal da linha, que é classificada como texto a partir de uma análise da sua geometria.

Um filtro de convolução para salientar o contraste é utilizado em [1]. Uma pré-filtragem é aplicada considerando a densidade de bordas em cada sub-região de um quadro. A seguir, uma análise de componentes conexas é feita para unir eventuais regiões fragmentadas de texto.

Em [27], um classificador SVM é utilizado sobre características derivadas da transformada wavelet [13]. Para agilizar o processo, apenas 1 quadro é analisado a cada 30. Esta abordagem não visa obter precisamente os quadros de início e término das legendas.

Uma segmentação por redução de cores é descrita em [31], baseada no fato de que textos contrastam com o fundo e são monocromáticos. Uma filtragem é usada para eliminar regiões de contraste não condizentes com textos.

Em [49], legendas de vídeos esportivos são obtidas analisando apenas a região inferior dos quadros, porção na qual as legendas são mais frequentes. Apenas os quadros que passam por uma etapa inicial de filtragem por histograma de cores são analisados.

Um classificador SVM com uma função de base radial é descrito em [54]. Cada quadro é particionado em blocos de 8×8 pixels, servindo de entrada para o classificador. Vale salientar que os testes apresentados consideram apenas uma sequência de vídeo com 150 quadros.

Pontos Importantes

Com base nos trabalhos citados anteriormente, pode-se destacar alguns pontos importantes:

- é comum realizar uma redução da taxa de quadros dos vídeos a fim de agilizar o processo de detecção, o que pode ocasionar uma perda de informação do quadro exato no qual as legendas iniciam ou terminam.
- vários trabalhos realizam uma detecção de textos em um único quadro e então se dedicam em rastreá-los nos quadros seguintes, ao invés de realizar a detecção novamente. Esta estratégia exige que o intervalo de tempo entre cada detecção seja pequeno o bastante para que legendas não sejam perdidas.
- foram bastante exploradas categorias específicas de vídeos, como esportivas ou de noticiários. Estas categorias muitas vezes possuem características próprias que facilitam a detecção, no entanto, métodos desse tipo tendem a perder eficiência quando são portados para outras categorias.
- vários trabalhos assumem a possibilidade de se trabalhar com vídeos, mas na realidade abordam imagens isoladas e aplicam o método em todos os quadros de uma sequência de imagens.
- muitas abordagens focam em línguas específicas que, em geral, são a inglesa, japonesa ou chinesa. Não foram encontrados trabalhos referentes, por exemplo, ao idioma árabe.

A Tabela 2.1 relaciona vários trabalhos com algumas destas características.

2.2.3 Notas sobre Legendas

As legendas podem variar com relação a uma série de fatores, como o tamanho e a cor do texto, a posição em que aparecem no vídeo e o tempo de duração. Não há um padrão bem definido para a criação de legendas e, dada a massificação da capacidade de produção e edição de vídeos caseiros e amadores, qualquer padrão que venha a surgir não será garantidamente aceito. Isto não impede a utilização de padrões específicos como base.

Uma padronização para criação de legendas foi proposta em [25]. Apesar deste trabalho ser voltado para o padrão linguístico ocidental, suas definições podem ser adaptadas para outros alfabetos. Alguns pontos interessantes que podem ser destacados são:

Métodos	Observações
[1], [27], [31], [49], [52], [54]	Realizam a detecção quadro-a-quadro, sem considerar as características inerentes dos vídeos.
[3], [27]	Melhoram o desempenho ao reduzir a taxa de quadros dos vídeos.
[7], [23], [58]	Utilizam propriedades do MPEG para aumento de acurácia ou eficiência.
[3], [23], [27], [47], [49]	Baseiam-se em uma classe específica de vídeos, como noticiários ou esportivos.
[28],[34]	Realizam a detecção de legendas em alguns quadros e um rastreamento nos demais.

Tabela 2.1: Agrupamento de alguns trabalhos encontrados na literatura de acordo com certas características.

- uma linha de texto não deve possuir uma altura maior que $\frac{1}{12}$ da altura de cada quadro do vídeo, para não cobrir uma área muito grande do quadro.
- o ser humano necessita de pelo menos 1.5 segundos para ler uma única palavra isolada, enquanto que uma linha inteira de legenda exige 3.5 segundos de leitura; logo, uma legenda nunca deve durar menos que 1.5 segundos.
- entre duas legendas consecutivas, deve existir algum intervalo de tempo para que o telespectador note com mais facilidade que houve uma mudança de legendas.

2.3 Ritmo Visual

As próximas seções definem o ritmo visual, apresentam suas principais características e descrevem alguns trabalhos relevantes disponíveis na literatura.

2.3.1 Definição

O ritmo visual é uma representação de imagens utilizada para reduzir significativamente o volume de informações gráficas existentes em um vídeo. Mais do que isso, o ritmo visual permite rearranjar o seu conteúdo de forma a permitir uma análise do vídeo sobre a ótica da temporalidade e não de cada quadro isoladamente.

Tomando-se um vídeo como o paralelepípedo *largura* × *altura* × *tempo*, um *corte* pode ser definido em qualquer direção. No caso do ritmo visual, ele é realizado no sentido do eixo temporal (Figura 2.1).

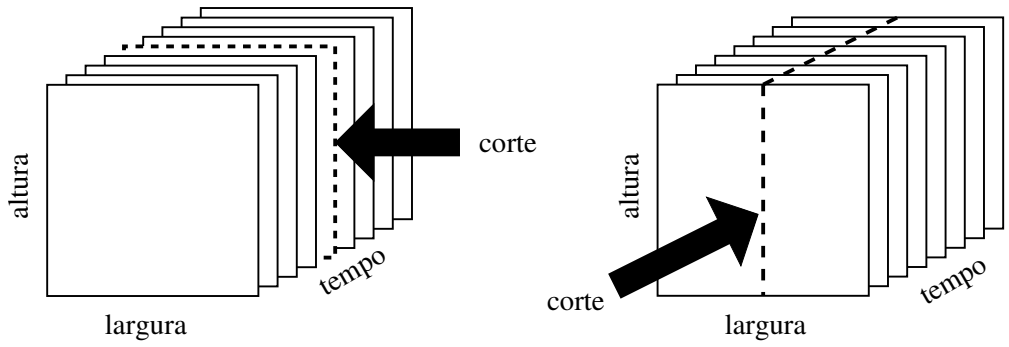


Figura 2.1: Exemplo de como um vídeo pode ser cortado. Se realizado um corte como em (a), um quadro do vídeo será obtido. Entretanto, se realizado como em (b), o ritmo visual deste vídeo será obtido.

Para uma definição mais formal de corte é preciso introduzir o conceito de *fatia* (*slice*). Uma fatia é uma curva utilizada para cortar um quadro, extraindo-se um conjunto de pixels para isso. As Equações 2.1, 2.2 e 2.3 exemplificam três funções, a de fatia horizontal, vertical e diagonal, respectivamente, para uma imagem F_t de dimensões $M \times N$ pixels.

$$fatia[i, F_t] = \sum_{p=k-j}^{k+j} F_t(p, i), k = M/2 \tag{2.1}$$

$$fatia[i, F_t] = \sum_{p=k-j}^{k+j} F_t(i, p), k = N/2 \tag{2.2}$$

$$fatia[i, F_t] = \sum_{p=i-j}^{i+j} F_t(p, i), \tag{2.3}$$

em que $0 \leq p < M$, $0 \leq p < N$. Quando $j = 0$, a coluna, linha ou diagonal central do quadro é definida como fatia. A Figura 2.2 ilustra os três cortes para um quadro do vídeo.

O ritmo visual é formado pela união das fatias de todos os quadros de uma sequência. Mais precisamente, o ritmo visual é a imagem construída por todas as fatias, da seguinte forma:

$$ritmo[t, i] = fatia[i, F_t],$$

em que F_t é o quadro no instante t e i é o i-ésimo pixel da fatia.

Outras curvas podem ser utilizadas para construir o ritmo visual e, de fato, isto será feito mais adiante neste capítulo.

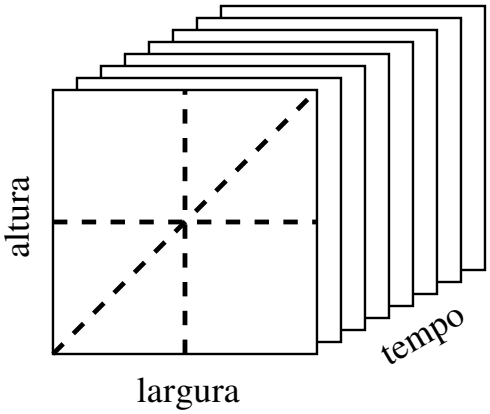


Figura 2.2: Aplicação, sobre um quadro, dos cortes horizontal, vertical e diagonal, definidos pelas Equações 2.1, 2.2 e 2.3, respectivamente.

2.3.2 Trabalhos sobre Ritmo Visual

O ritmo visual foi introduzido por Kong et al. [26] na detecção de cortes de cenas, em que são utilizadas as fatias definidas por uma linha vertical e outra horizontal. Duas alternativas para esta detecção são propostas, uma priorizando a eficiência e outra a acurácia. Na primeira, são consideradas apenas duas linhas horizontais de cada ritmo, nas quais uma análise de gradientes realiza a estimativa dos cortes de cenas. Na segunda, os dois ritmos são utilizados inteiros e, neste caso, é aplicado um detector de bordas para localizar os pontos de corte.

Ngo et al. [40] e [38] estendem o uso do ritmo visual para a detecção não apenas de cortes, mas também de *wipes*[1] e *dissolves*[2]. Os autores sugerem o uso da fatia diagonal para complementar as linhas horizontal e vertical.

O ritmo visual é utilizadoem [17] para a detecção de cortes de cenas utilizando uma linha diagonal. O ritmo visual é reduzido a um mapa de gradientes horizontais, no qual um processo de afinamento permite extrair os pontos de máximo, tomados como cortes de cenas. Uma variação do método é proposta para a detecção de *flashes*. Os mesmos autores dão continuidade ao trabalho em [18], em que é introduzido o conceito de ritmo visual por histograma para detecção de *fades*[3].

Em [36], a detecção de cortes pelo ritmo visual é utilizada na sumarização de vídeos. O trabalho em [4] utiliza o ritmo visual para sumarizar partidas de futebol. Dois descritores

[1]Transição de cena onde uma linha percorre gradualmente o quadro, sendo que, por onde ela passa, dá-se lugar à nova cena.

[2]Um *dissolve* ocorre quando há uma sobreposição de duas cenas, sendo que, enquanto uma perde força até desaparecer, a outra surge lentamente.

[3]O *fade-out* é uma transição de cenas onde a mesma perde força até desaparecer completamente, dando lugar a uma cena completamente preta (ou branca). O *fade-in* é o processo inverso.

são propostos, um de cor e outro de movimento, utilizados para obter informações sobre as transições de cenas peculiares dos jogos de futebol televisionados. Em [9], o ritmo visual é utilizado na localização de regiões de interesse do usuário, sendo definidas como áreas com movimento.

2.3.3 Características do Ritmo Visual

A Figura 2.3 ilustra um ritmo visual. Este ritmo foi contruído a partir de um vídeo com 872 quadros, o que implica uma largura de 872 pixels. A fatia foi obtida utilizando-se uma linha vertical, aplicada no centro do quadro (a altura do ritmo é a altura de cada quadro do vídeo).

Figura 2.3: Exemplo de um ritmo visual. Este ritmo foi construído utilizando como corte uma linha vertical central, aplicado em um vídeo de 872 quadros, cada um com dimensão de 448×336 pixels.

A Figura 2.4 destaca porções do ritmo visual que exemplificam algumas propriedades do mesmo (mais detalhes sobre as propriedades do ritmo visual são encontrados em [17, 18, 38]). O destaque em A exemplifica um corte entre cenas. Em B, tem-se um exemplo de transição de cena conhecido como *dissolve*. Em C, observa-se um padrão formado por um movimento de câmera de *zoom-in*. Em D, um trecho do vídeo é ilustrado em que não houve variação espacial dos objetos da cena. No destaque em E, há quatro regiões retangulares que aparecem na posição inferior do ritmo visual e que correspondem a legendas. Neste sentido, algumas observações importantes podem ser apontadas sobre as legendas:

- tratam-se de objetos que permanecem imóveis durante um certo intervalo de tempo.

- possuem um bom contraste com o restante da cena.
- possuem todas uma cor praticamente idêntica.
- não seguem nenhum tipo de transição de cenas observado neste vídeo, ou seja, a informação das legendas permanece independentemente do início ou fim de uma cena.
- neste caso específico, cada região de legendas aparece isoladamente uma das outras, ou seja, em nenhum quadro existe mais de uma legenda.
- não possuem uma altura significativa em relação à altura do ritmo.
- não aparecem por um período de tempo muito curto.
- existe um intervalo de tempo entre o término de uma legenda e o início da próxima.

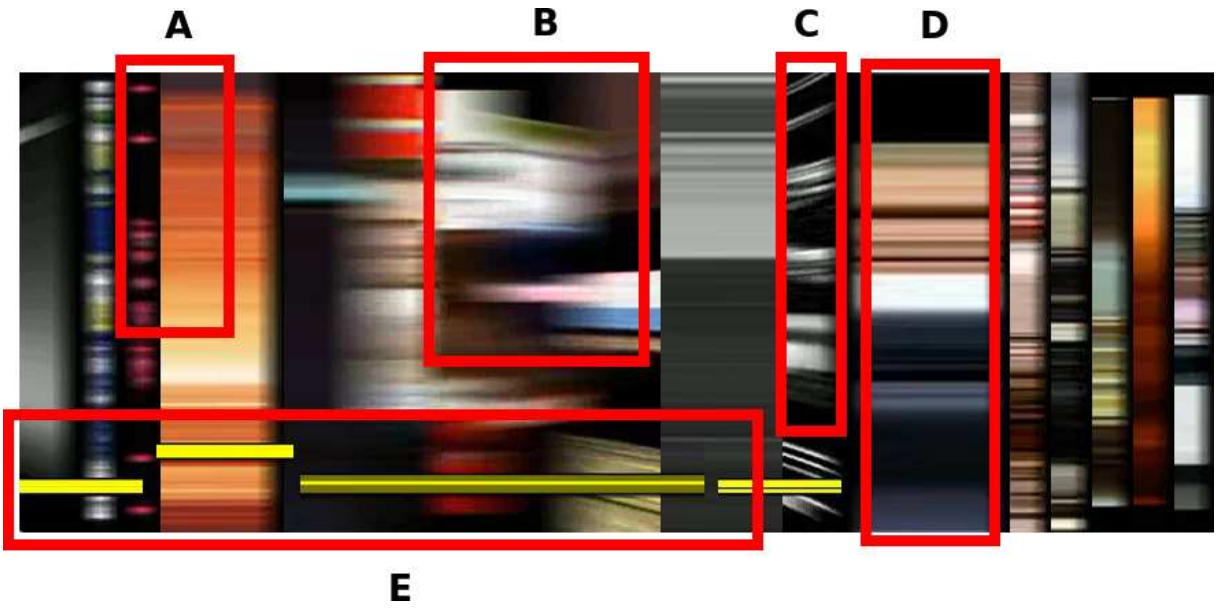


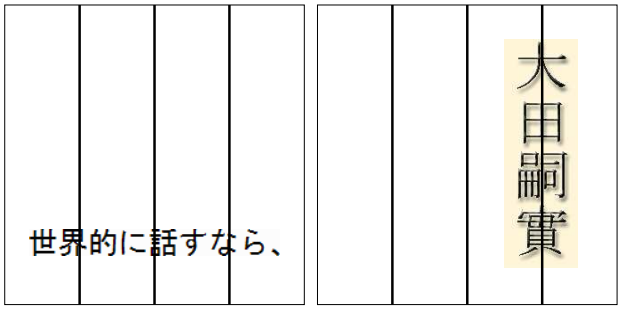
Figura 2.4: Algumas propriedades do ritmo visual em destaque.

Várias dessas características podem ser consideradas na detecção de legendas. O Capítulo 3 discute mais precisamente alguns desses aspectos.

2.4 Curvas de Preenchimento do Espaço

Um ponto importante sobre o uso do ritmo visual refere-se à escolha da curva utilizada para realizar o corte do vídeo.

Possuir informações *a priori* das posições das legendas pode ajudar na tarefa de escolher uma curva mais apropriada, entretanto, no caso geral, é necessário considerar curvas que sejam as mais abrangentes possíveis. Por exemplo, a linha vertical é eficaz na detecção de legendas que possuam um sentido horizontal, porém, é inapropriada para legendas que se encontram na vertical (Figura 2.5).



(a) (b)

Figura 2.5: Exemplo de duas orientações nas quais uma legenda pode ocorrer. Em (a), qualquer um dos três cortes realizados consegue intersectar a legenda. Em (b), apenas um dos cortes se mostra efetivo para este fim. Esses dois exemplos demonstram como pode não ser eficaz utilizar curvas com a mesma orientação das legendas.

Este trabalho propõe um método de detecção de legendas a partir da utilização de curvas mais detalhadas que cubram adequadamente o espaço definido pelos quadros em questão. Neste sentido, ele considera o conceito de curvas de preenchimento do espaço representado, por exemplo, pelas curvas de Peano, Hilbert, Lebesgue e zig-zag, entre outras [44]. As curvas de Hilbert e zig-zag são brevemente descritas a seguir.

2.4.1 Curva de Hilbert

A curva de Hilbert [21, 44] é uma representação fractal contínua construída para uma dada aproximação. Ela pode ser referenciada por H_n para a sua n-ésima aproximação. A Figura 2.6 exibe a curva de Hilbert para as cinco primeiras aproximações. Dentro do quadrado unitário, o comprimento de H_n é dado por $2^n - 1/2^n$, possuindo um crescimento exponencial em n.

Apesar desta curva estar confinada no quadrado unitário, na prática ela será inscrita em quadros de um vídeo. Isso significa não só ajustar sua escala, mas também sua proporção (uma vez que imagens são, em geral, retangulares).

Sendo também a curva desenhada sobre imagens, significa que ela será discreta, assim, seu comprimento será dado pelo número de pixels que a constrói.

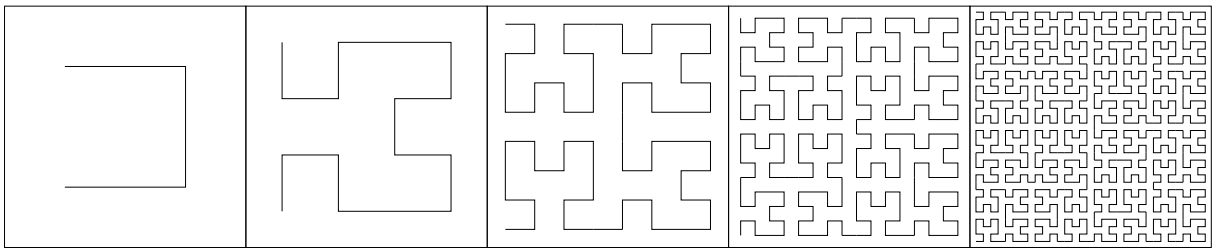


Figura 2.6: A curva de Hilbert para as cinco primeiras aproximações.

2.4.2 Curva Zig-Zag

A curva de preenchimento por zig-zag é composta por linhas quase ou totalmente paralelas, alinhadas lado-a-lado para preencher um espaço entre dois pontos, formando um padrão dentado e razoavelmente homogêneo (Figuras 2.7(a) e 2.7(b)). Esta curva se tornou mais conhecida após ser adotada pelo padrão JPEG para efetuar a compressão dos pixels por meio da transformada discreta do cosseno (DCT) [45, 51].

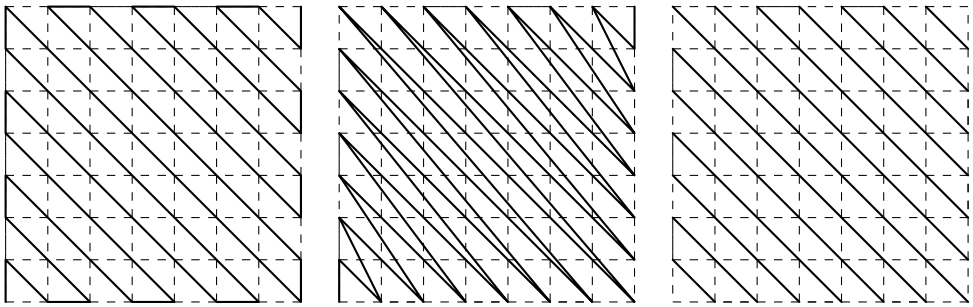


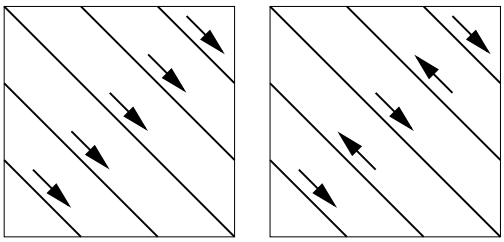
Figura 2.7: Variações da curva em zig-zag.

A curva em zig-zag tem a vantagem, para este trabalho, de poder ser constituída de segmentos de reta com ângulos próximos ou iguais a 45º. Sem possuir arestas predominantemente verticais ou horizontais e levando-se em conta que textos são em sua grande maioria orientados em uma dessas duas componentes, esta curva pode ser tida como bastante adequada à aplicação em questão. De fato, o ideal é que as arestas da curva se aproximem de um ângulo de 45º. Por esse motivo, neste trabalho será utilizada a variante da Figura 2.7(a), mostrada na Figura 2.7(c).

Além disso, não há necessidade de que a curva utilizada seja contínua, ou seja, a curva pode ser constituída por segmentos de reta desconexos. Desta forma, podem ser removidas as conexões das diagonais que tocam as bordas da imagem (Figura 2.7(c)).

As Figuras 2.8(a) e 2.8(b) ilustram duas possibilidades de orientação da curva. Uma

vez que a mesma não é contínua, não importa em qual sentido as diagonais estão conectadas. A primeira opção será utilizada neste trabalho.



(a) (b)

Figura 2.8: A curva zig-zag pode ser construída seguindo dois sentidos diferentes.

A escala da curva zig-zag será definida como o número de segmentos de reta diagonais que a constitui. A Figura 2.9 ilustra a curva para algumas escalas.

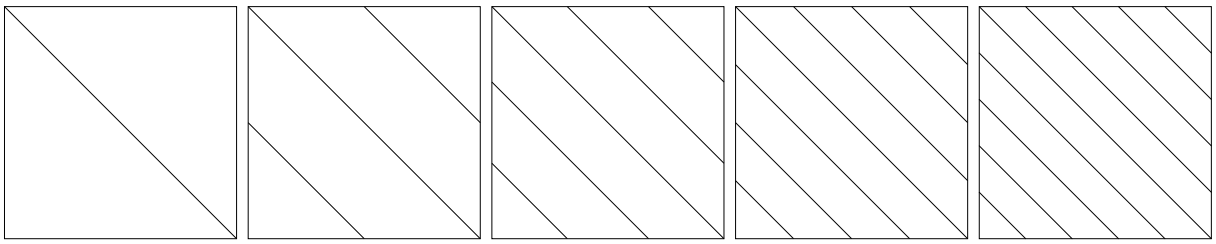


Figura 2.9: A curva zig-zag para as escalas 1, 3, 5, 7 e 9, respectivamente.

2.4.3 Linha Vertical

Levando-se em conta que as características topológicas das legendas permitem, quase sempre, encontrá-las dispostas horizontalmente, linhas verticais tornam-se um padrão para detecção de legendas.

Uma curva deste tipo pode ser formada por uma ou mais linhas verticais equidistantes e centralizadas. No entanto, considerando-se que legendas aparecem muito comumente na região horizontal central, uma única linha vertical pode ser suficiente para realizar a detecção. Desta forma, a curva vertical torna-se extremamente compacta, efetiva e simples de ser construída. No entanto, essas hipóteses nem sempre devem ser adotadas, pois são altamente dependentes do conhecimento *a priori* da aplicação, uma vez que os textos nem sempre possuem orientação horizontal.

2.4.4 Outras Curvas

Uma breve discussão sobre outras curvas é incluída a seguir:

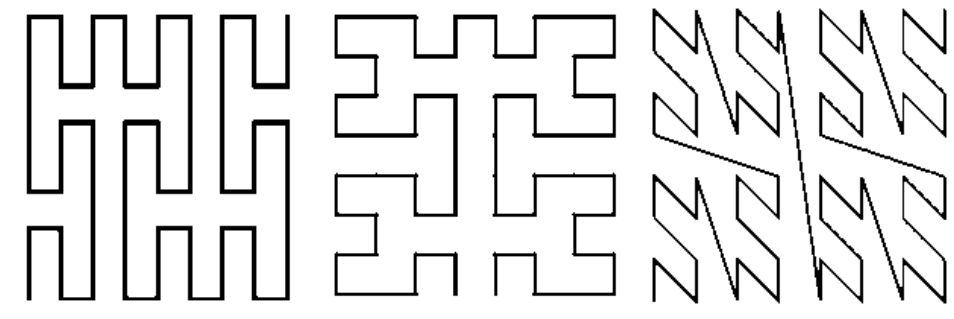
- *Peano:* muito semelhante à curva de Hilbert, a curva de Peano possui uma orientação predominante, seja na horizontal, seja na vertical. Desta forma, ela é pouco adequada a problemas de detecção de legendas em que não se conhece a orientação dos textos (Figura 2.10(a)).

- *Moore:* é basicamente a curva de Hilbert aplicada quatro vezes, de forma a tornar próximos os pontos inicial e final da curva, facilitando assim fechá-la e torná-la cíclica. Esta é uma propriedade irrelevante para este trabalho. Além disto, esta curva na sua escala menos detalhada é quatro vezes mais extensa do que a curva de Hilbert na mesma escala. Sendo assim, a curva de Moore não é tão simples quanto a de Hilbert (Figura 2.10(b)).

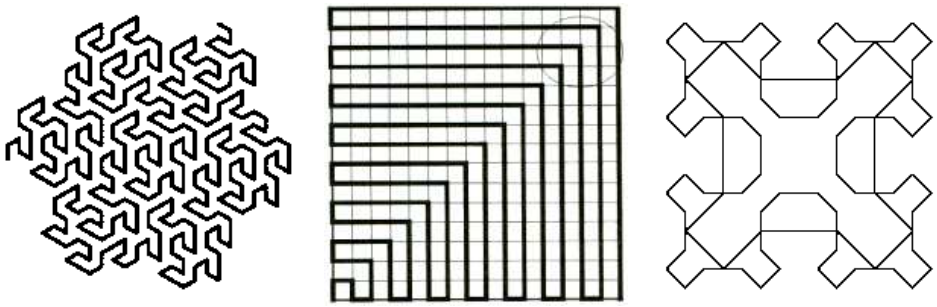
- *Lebesgue e Gosper:* quando se trabalha com curvas discretas em imagens, por questões de arredondamento de valores, pode ser desejado que estas possuam orientações apenas horizontal, vertical ou diagonal 45°. Essas duas curvas não possuem exclusivamente esta característica, logo, foram descartadas (Figuras 2.10(c) e 2.10(d)).

- *Regazzoni:* apesar da curva de Regazzoni possuir linhas apenas horizontais e verticais, estas variam de acordo com a região do plano, dificultando a previsibilidade da curva (Figura 2.10(e)).

- *Sierpinski:* Esta curva passa mais de uma vez por alguns pontos, o que ocasiona redundância de informações (Figura 2.10(f)).



(a) Peano (b) Moore (c) Lebesgue



(d) Gosper (e) Regazzoni (f) Sierpinski

Figura 2.10: Exemplos de seis curvas que foram descartadas por não possuírem características desejáveis à deteção de legendas.

Capítulo 3

Ritmo Visual e Detecção de Legendas

Este capítulo apresenta a metodologia proposta para realizar a deteção de legendas utilizando o ritmo visual. Resumidamente, o método está organizado de acordo com o diagrama de fluxo da Figura 3.1.

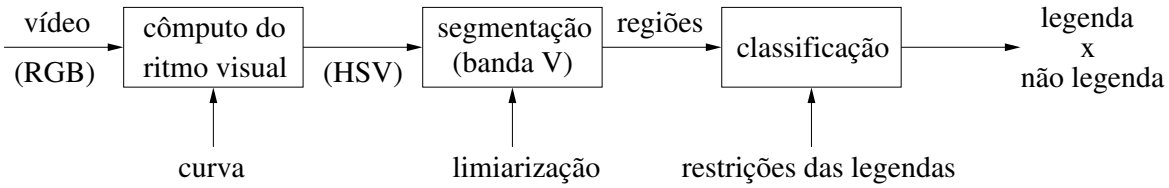


Figura 3.1: Fluxo básico do método proposto.

As seções deste capítulo estão organizadas da seguinte forma. A Seção 3.1 discute a obtenção do ritmo visual. A Seção 3.2 apresenta os algoritmos propostos para segmentar as regiões de interesse do ritmo visual. A Seção 3.3 apresenta várias medidas de filtragem das regiões obtidas na fase de segmentação. Finalmente, a Seção 3.4 analisa a complexidade computacional do método apresentado.

3.1 Obtenção do Ritmo Visual

Se uma dada curva de preenchimento deixar de intersectar uma legenda, o ritmo visual não conterá informações suficientes para detectá-la e, neste caso, é importante utilizar uma curva que seja a mais abrangente possível. Para atingir este propósito, os seguintes aspectos são considerados:

1. utilizar uma curva apropriada (conforme discutido anteriormente na Seção 2.4).
2. ajustar a escala da curva para obter um bom compromisso entre desempenho e acurácia.

3. particionar os quadros do vídeo, aplicando para cada partição uma diferente instância da curva.

4. dilatar a curva.

Utilização da Curva Apropriada

Opções de curvas apropriadas são indicadas na Seção 2.4, sobretudo a curva de Hilbert ou zig-zag. Parte-se do pressuposto que ambas são igualmente invariantes à orientação das legendas, embora, como será visto posteriormente nos experimentos, cada uma poderá obter melhores resultados específicos.

Ajuste da Curva

As curvas de Hilbert e zig-zag podem ser refinadas por uma escala. O incremento desta escala altera sensivelmente o comprimento da curva, ao mesmo tempo que aumenta o seu grau de detalhamento, permitindo que seja monitorada uma região maior do espaço ocupado.

Deve ser lembrado que a altura do ritmo visual é dado pelo comprimento da curva utilizada. Assim, quanto maior o detalhamento da curva, maior será o custo computacional para obter e processar a imagem do ritmo visual.

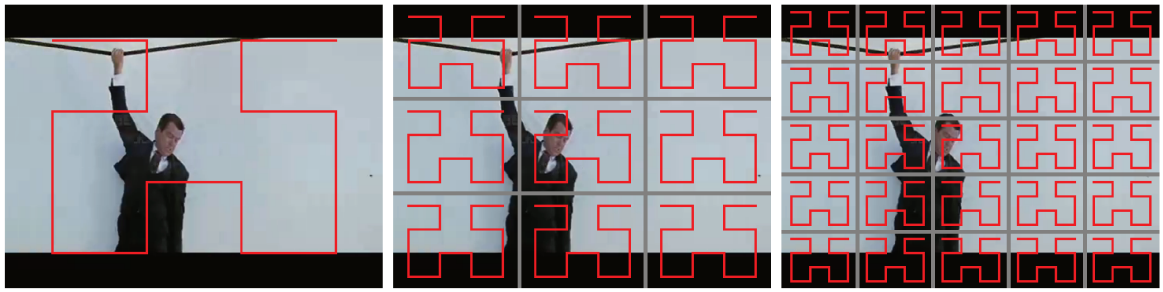
Fragmentação das Imagens

Dado que curvas muito detalhadas podem ser demasiadamente longas (principalmente a curva de Hilbert), um meio de manter a curva a mais simples possível é particionar a imagem original. Em cada parte, versões reduzidas da curva podem ser utilizadas, uma vez que a mesma é ajustada pela área do espaço que a contém.

Uma forma de fazer esta partição é dividir a imagem em blocos e, em cada bloco, aplicar a curva com escala reduzida. Um exemplo desse procedimento para a curva de Hilbert é apresentado na Figura 3.2, em que uma imagem foi subdividida em 3×3 e 5×5 blocos.

Dado que, ao efetuar esse particionamento, não resultará apenas uma curva e sim várias (9 e 25 nos exemplos das Figuras 3.1 e 3.1, respectivamente), o mesmo número de ritmos visuais será obtido. O procedimento a ser tomado neste caso é simplesmente empilhar os ritmos visuais, criando-se um novo ritmo cuja altura é a soma de todos os demais. Isto é ilustrado na Figura 3.3.

O particionamento da imagem em blocos efetivamente aumenta a região monitorada da imagem, mas, ao mesmo tempo, aumenta o número de pixels total usado para construir a curva final. A vantagem em utilizar esta técnica não está em reduzir o comprimento da



(a) (b) (c)

Figura 3.2: Uma imagem em (a) sendo particionada em (b) 3×3 e (c) 5×5 blocos.

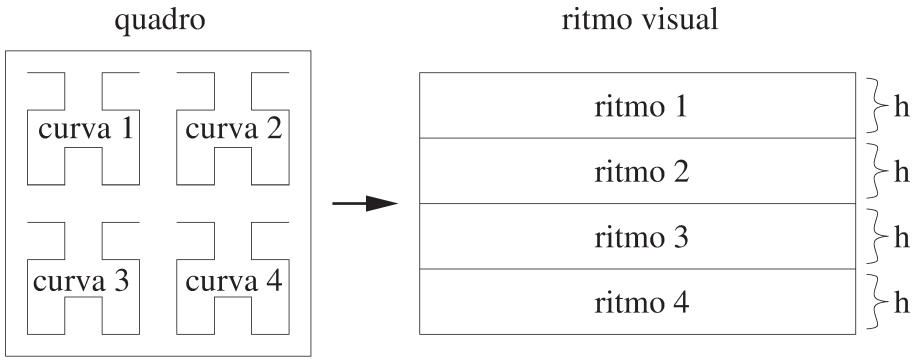


Figura 3.3: Um quadro particionado em 4 blocos. Cada curva, que possui comprimento h, produz um ritmo visual. A união destes gera o ritmo visual final, com altura $4 * h$.

curva final, mas sim, ao aumentar o número de partições da imagem, permitir que seja utilizada uma curva mais simples (menos detalhada).

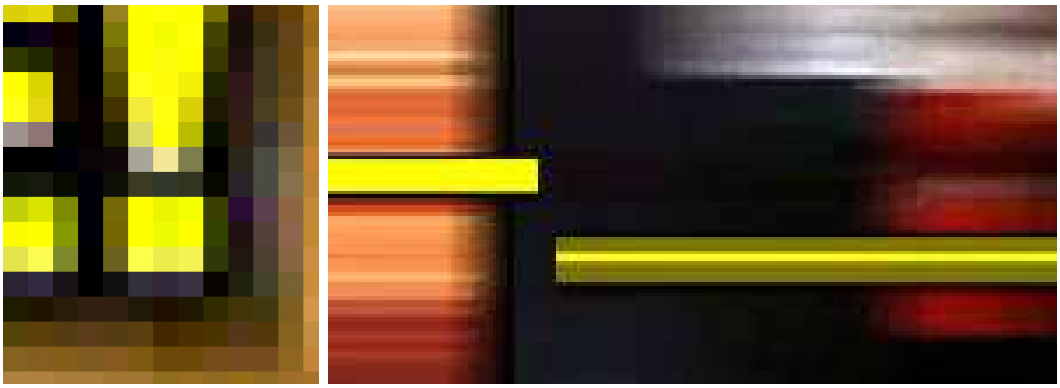
Um ponto importante é que, enquanto a curva de Hilbert possui crescimento exponencial, o número de partições de uma imagem cresce linearmente com o número de pixels. Desta forma, em algum momento, torna-se vantajoso elevar o número de partições e não a escala da curva.

Dilatação da Curva

Até este ponto, considera-se que a curva possui espessura de um pixel. No entanto, o espaçamento entre duas palavras de uma legenda pode ser igual ou maior que esse valor. Dependendo do caso, o espaço entre duas letras de uma mesma palavra pode ser suficiente para que a curva passe entre elas sem tocá-las. De fato, mesmo que a curva toque uma letra, se for tangenciando-a, já é suficiente para que a qualidade da detecção seja comprometida. Isso se deve ao fato de que as letras sempre estão sujeitas a suavizações

devido ao uso de técnicas de *anti-aliasing*[1]. O *anti-aliasing* é muito usado para renderizar caracteres [16], a fim de melhorar a qualidade visual do texto e, como as legendas sempre são inseridas por meio de edição, é importante considerar este fato.

A Figura 3.4 ilustra o que ocorre quando a curva passa levemente sobre uma letra que possui suavização. Em 3.1, é apresentado um fragmento de texto ("E!"), o qual foi bastante ampliado para destacar a suavização em seu contorno. Os efeitos deste tipo de suavização pode ser visto em 3.1, onde há duas legendas sendo capturadas (os dois retângulos). A legenda da esquerda foi corretamente intersectada pela curva, porém, na da direita, a curva passou tangenciando os caracteres, gerando assim o padrão que pode ser observado. O padrão é composto por basicamente três faixas horizontais de cores distintas, sendo que apenas a faixa central é que possui a real cor do texto.



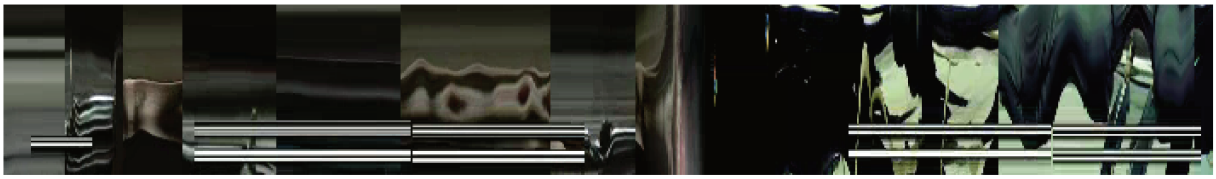
(a) (b)

Figura 3.4: A figura (a) ilustra um fragmento de texto ("E!") ampliado. Pode-se notar a suavização dos caracteres devido ao uso de *anti-aliasing*. Na figura (b), um exemplo de legenda no ritmo visual que foi intersectada justamente em uma borda suavizada, gerando um padrão incompleto.

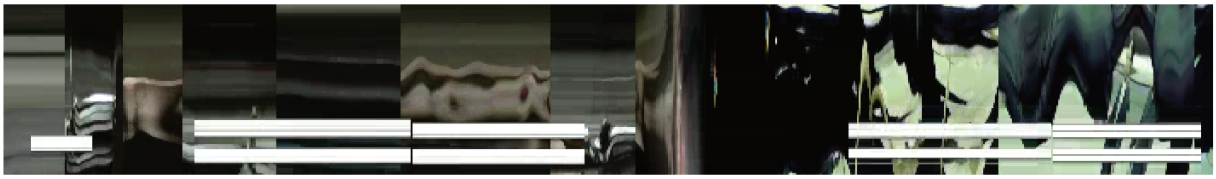
Este tipo de ocorrência é inconveniente por dois motivos: primeiro que esta variação de cor torna mais difícil a sua segmentação e, segundo, que isto não é intuitivo uma vez que textos normalmente possuem uma cor única e homogênea. Uma forma de evitar este tipo de problema é fazer com que a curva tenha mais do que um pixel de espessura. No entanto, isso viola a relação na qual a i-ésima coluna do ritmo visual representa o i-ésimo quadro do vídeo.

[1] O serrilhamento (*aliasing*) é o efeito em forma de serra que se cria ao desenhar uma reta em diagonal. Uma vez que a divisão mínima numa imagem é discreta (pixels) surge o aparecimento dos "dentes" de serra ao longo da reta desenhada. O antisserrilhamento (*anti-aliasing* [12, 14]) é um método para minimizar esse efeito, dando a ilusão de que a reta é mais perfeita. Ele baseia-se em borrar essas bordas, dando a impressão visual de um contorno mais suave.

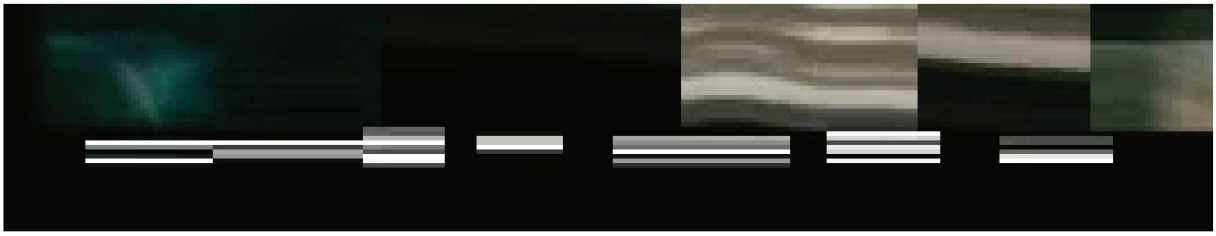
Uma solução encontrada é a de realizar uma dilatação morfológica [2] em cada quadro do vídeo, utilizando um elemento estruturante de tamanho igual ao número de pixels em que se deseja dilatar a curva. Elementos estruturantes circulares ou quadrados são boas opções. Como nem todos os pixels do quadro serão utilizados para construir o ritmo visual, a dilatação pode ser aplicada apenas sobre o caminho da curva, agilizando assim o processo. Alguns exemplos da melhora de qualidade podem ser vistos na Figura 3.5.



(a)



(b)



(c)

(d)

Figura 3.5: Exemplos da melhora de qualidade do ritmo visual ao aplicar uma dilatação. Em (a) e (c) estão exemplos de ritmos visuais e, em (b) e (d), o resultado após dilatar a curva por um elemento estruturante quadrado de tamanho 3. Todos os exemplos foram contruídos com uma curva zig-zag com escala 1.

3.2 Segmentação

A partir do ritmo visual, o próximo passo consiste na localização dos padrões que representam legendas. Como discutido na Seção 2.3.3, as legendas formam retângulos preenchidos por uma cor aproximadamente homogênea. Este padrão é normalmente destacado e bem visível.

Um fator desejável é que o algoritmo de segmentação seja linear em número de pixels, já que a largura do ritmo visual é proporcional ao número de quadros do vídeo, uma grandeza que não possui um limite definido.

Com base neste fato, um algoritmo de clusterização simples se mostrou adequado à tarefa básica de segmentação. O algoritmo em questão agrupa pixels semelhantes entre si, tendo como base um determinado valor, ou seja, dentro de cada grupo, a diferença entre quaisquer dois pixels será no máximo um valor T previamente definido.

Este método de segmentação é detalhado nos Algoritmos 1, 2 e 3, mostrados a seguir. O Algoritmo 1 corresponde a um método simples de crescimento de região [15], o Algoritmo 2 realiza o preenchimento de regiões conexas e o Algoritmo 3 é utilizado para decidir a pertinência ou não de um pixel a uma dada região.

Algoritmo 1 Segmentação de uma imagem

```
1: função Segmenta( imagem I ):
2:     cria uma lista G de grupos, inicialmente vazia
3:     para cada pixel p_{x,y} de I, faça:
4:         se p_{x,y} não pertence a nenhum grupo em G, então:
5:             insira em G um novo grupo g
6:             PopulaGrupo(g, p_{x,y})
```

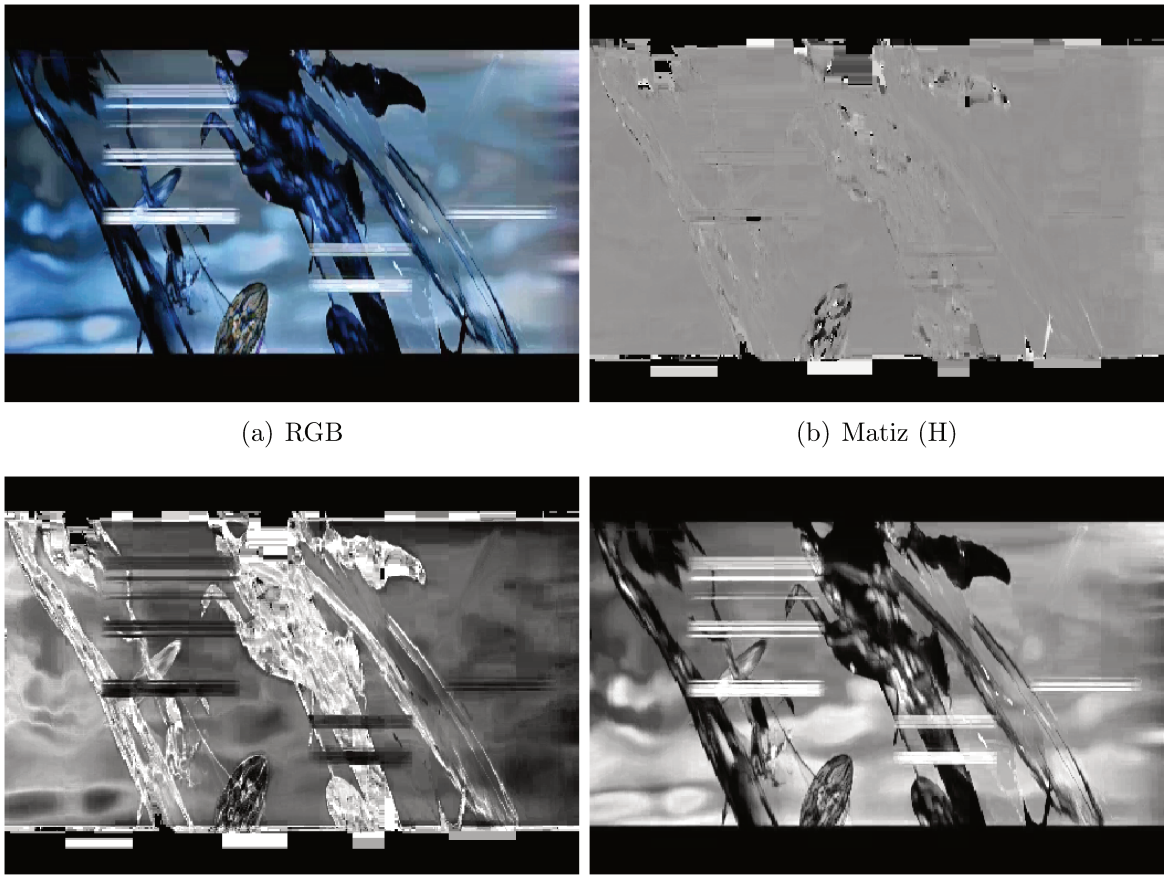
Algoritmo 2 Povoamento de uma região conexa

```
1: função PopulaGrupo( grupo g, pixel p_{x,y} ):
2:     se Verifica(g,p_{x,y}) = 1, então:
3:         insira p em g
4:         para todo p' ∈ {p_{x+1,y}, p_{x-1,y}, p_{x,y+1}, p_{x,y-1}}, faça:
5:         se p' não pertence a nenhum grupo em G, então:
6:             PopulaGrupo(g, p')
```

O ritmo visual, antes de ser segmentado, é convertido para o espaço de cor HSV [46]. Este espaço de cor, diferentemente do RGB, descorrelaciona as diferentes informações das cores. As informações de matiz e saturação do espaço HSV se mostraram experimentalmente ineficazes à segmentação das legendas, de tal forma que apenas a banda de intensidade foi considerada. Exemplos são ilustrados na Figura 3.6.

Algoritmo 3 Função de comparação entre dois pixels

```
1: função Verifica( grupo g, pixel p_{x,y} ):
2:     se existe algum pixel p ∈ g tal que |p − p_{x,y}| > T, então:
3:         retorna 0
4:     caso contrário:
5:         retorna 1
```



(a) RGB (b) Matiz (H)

(c) Saturação (S) (d) Intensidade (V)

Figura 3.6: Um exemplo em (a) de ritmo visual no espaço de cor RGB. As demais figuras representam isoladamente as bandas do espaço de cor HSV. Pode-se notar que as legendas não ficaram visíveis na banda de matiz. Na banda de saturação, na parte inferior, surgiram artefatos de compressão (retângulos brancos), além do fato de que as legendas ficaram com a mesma cor do fundo (preto). A banda de intensidade, por outro lado, é a que melhor destaca as legendas e não acrescenta artefatos.

3.3 Classificação

Como resultado da segmentação, tem-se uma imagem fragmentada em um conjunto de regiões (componentes conexas). Supõe-se que cada legenda forma uma região isolada de qualquer outro objeto presente nas cenas do vídeo e que estas regiões são retangulares. O próximo passo consiste em definir se cada uma dessas regiões representa uma legenda ou não. Para tomar esta decisão, foram considerados os três seguintes processos de filtragem:

1. aplicação das restrições impostas pela definição de legenda.
2. verificação da razão de aspecto de cada região.
3. análise do conjunto resultante do afinamento de cada região.

Cada região é testada segundo esses três procedimentos. Se for eliminada por qualquer um deles, a região é descartada. Assim, apenas as regiões que passarem em todos os testes serão classificadas como legendas.

Restrições das Legendas

A primeira filtragem diz respeito às características mencionadas na Seção 2.2.3 e referentes a algumas normas associadas à especificação das legendas. As duas regras consideradas mais importantes são:

1. uma legenda deve durar pelo menos $\frac{3}{2}$ segundos (45 quadros).
2. uma legenda não deve ter altura superior a $\frac{1}{12}$ da altura dos quadros do vídeo.

O item 1 indica que qualquer região com largura menor que 45 pixels pode ser desconsiderada como legenda. O item 2 filtra componentes segmentadas cuja altura ultrapassa o valor especificado.

Uma terceira restrição foi introduzida para eliminar regiões com altura menor que 3 pixels, uma vez que dificilmente representarão legendas. Esta restrição ajuda a reduzir a taxa de falsos positivos gerados por ruídos e artefatos provenientes da segmentação.

Verificação da Razão de Aspecto

A razão de aspecto é uma forma utilizada para estimar a retangularidade de um objeto [43] e é dada pela razão entre a área de um objeto e a área do seu retângulo envoltório. Naturalmente, quanto menor a diferença de área entre ambos, mais próximo de 1 será esta razão.

Para o método proposto, escolheu-se um limiar R ao qual pode ser descartada qualquer região com razão de aspecto menor que este valor. O valor deste limiar foi escolhido experimentalmente e será demonstrado mais adiante.

Análise do Afinamento

Uma forma retangular, quando idealmente afinada, gera uma linha reta. Esse conceito é de bom uso para classificar as legendas, uma vez que as mesmas sempre formam padrões retangulares. Neste caso, qualquer região que contenha buracos, bifurcações ou curvaturas não forma uma linha reta com seu afinamento (Figura 3.7). Uma análise geométrica e topológica dos respectivos elementos deve auxiliar na identificação das componentes de interesse.

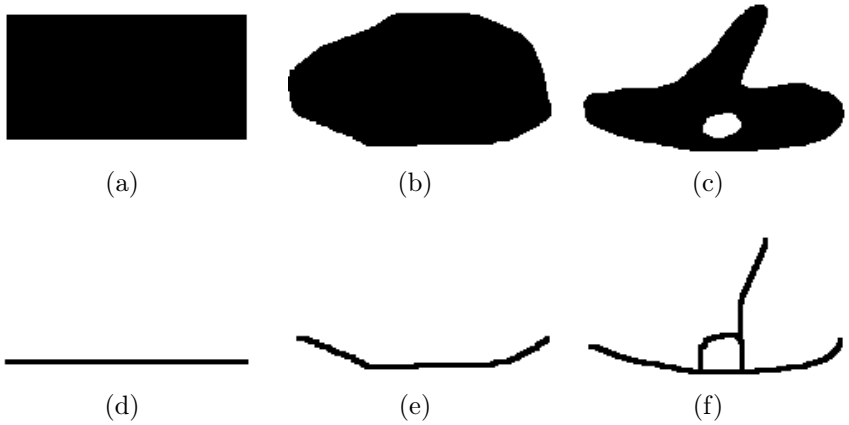


Figura 3.7: As figuras (a), (b) e (c) ilustram três regiões conexas. As figuras (d), (e) e (f) apresentam seus respectivos afinamentos. Como a região em (a) é retangular, o resultado do afinamento é uma reta horizontal, enquanto que o afinamento de (b) é uma linha irregular. Em (c), pode-se observado que o afinamento se deteriora conforme a região geradora se distancia de um retângulo.

O algoritmo de afinamento considerado neste trabalho é descrito em [57]. Para verificar se o resultado do afinamento se trata de uma linha horizontal, propôs-se um algoritmo que aproxima um conjunto de pixels para uma reta, obtendo o ângulo e o desvio padrão a partir dos quais os pixels são aproximados. A seguir, estes dois valores são analisados da seguinte forma:

1. se a reta aproximada não possuir ângulo próximo de zero, então a região não corresponderá a uma legenda. Deve-se lembrar que, no ritmo visual, qualquer legenda forma um padrão horizontal, independentemente da orientação dos textos no vídeo.

2. o desvio padrão deve ser menor do que um dado valor D, caso contrário, a reta não representa uma boa aproximação para esses pixels.

Esses dois itens oferecem um meio de decidir se o resultado do afinamento de uma região define ou não uma reta horizontal, e com isso, uma legenda. O método utilizado para obter a reta aproximada de uma região é descrito no Algoritmo 4.

Algoritmo 4 Aproximação de um conjunto de pontos para uma reta

```
1: função AproximaReta( pontos ):
2:     m ← Media(pontos)
3:     α1 ← 0
4:     α2 ← π/2;
5:     enquanto |α1 − α2| > L, faça:
6:         se σ1 > σ2, então:
7:             α1 = (α1+α2)/2
8:         caso contrário:
9:             α2 = (α1+α2)/2
```

Este algoritmo estima o ângulo que melhor define a reta aproximada. Considera-se que a mesma passa sobre o ponto médio do conjunto. O laço termina quando a diferença entre os dois ângulos de varredura, α_1 e α_2, é menor do que um dado limite L. O desvio padrão para cada ângulo é calculado pela distância entre todos os pontos e a reta. Ao término do algoritmo, tem-se como resultado o ângulo e o desvio padrão que serão utilizados para a classificação já mencionada.

3.4 Análise de Complexidade

Esta seção apresenta uma breve análise da complexidade assintótica do algoritmo proposto.

O comprimento do ritmo visual varia de acordo com dois parâmetros. A sua largura é dada pelo número t de quadros no vídeo. A altura é definida pela combinação entre a complexidade da curva e o número de blocos. No entanto, ambos são de tamanho fixo e determinado, logo, são constantes. Dessa forma, o número n de pixels no ritmo visual é $\Theta(t)$

A segmentação possui três funções aninhadas. A primeira (Algoritmo 1) é $\Theta(n)$, as outras duas (Algoritmos 2 e 3) são $O(1)$. Logo, a complexidade é $\Theta(n)$.

A classificação realiza três testes, sendo que cada um é $O(n)$. O terceiro teste (Algoritmo 4) possui um laço que itera enquanto o erro limite não for atingido, no entanto, experimentos mostraram que o resultado desejado sempre converge em um número pequeno de iterações (5 na prática); logo, o laço pode ser visto como $O(1)$.

Dessa forma, conclui-se que o método proposto é $\Theta(t)$, ou seja, é linear no número de quadros do vídeo.

Capítulo 4

Experimentos e Resultados

Este capítulo apresenta uma série de experimentos realizados com a finalidade de verificar o comportamento do método proposto no Capítulo 3. A base de dados construída para este intuito é descrita, bem como a plataforma utilizada nos experimentos. Comparações com outros métodos são realizadas a fim de analisar a eficácia obtida. Finalmente, uma discussão dos resultados obtidos é incluída no final do capítulo.

4.1 Requisitos

Os experimentos realizados para validar a metodologia utilizam uma base específica. Como não foi encontrada referência a nenhuma base que atenda aos requisitos desejados, foi elaborada uma própria. Ela foi constituída pela coletânea de diversos fragmentos de vídeos de diferentes categorias, incluindo noticiários, filmes, comerciais e animações computadorizadas. Há amostras de textos nos alfabetos ocidental (português e inglês), oriental (japonês) e árabico. Uma vez que é difícil encontrar vídeos com legendas orientadas em ângulos arbitrários, foi produzido um vídeo com legendas dispostas em ângulos de 10° e 20°.

A base contém 9 vídeos com um total de 41439 quadros. Ela está disponível para acesso público em [35]. Foi construído também o *ground truth* para cada vídeo, contendo as informações dos quadros de início e fim de cada legenda, que foi disponibilizado junto com a base. A Tabela 4.1 contém informações de cada vídeo.

O método proposto possui vários parâmetros que influenciam os resultados. Os principais deles são a curva utilizada, a sua escala e o número de blocos nos quais os quadros são particionados[1]. Três casos de testes foram idealizados, cada qual com configuração específica para esses parâmetros. Estes casos de testes estão detalhados na Tabela 4.2.

[1] A Seção 2.4 descreve cada curva e suas escalas, enquanto a Seção 3.1 discute a partição dos quadros em blocos.

Vídeo	**Nº de quadros**	**Resolução (pixels)**	**Idioma**	**Categoria**
1	2422	320 × 180	Português	Filme
2	2170	320 × 240	Português	Vídeo caseiro
3	5091	480 × 360	Inglês	Filme
4	6875	480 × 270	Árabe	Filme
5	5906	320 × 184	Português	Filme
6	6285	320 × 240	Inglês	Noticiário
7	11140	480 × 360	Japonês	Animação gráfica
8	900	294 × 240	Inglês	Comercial
9	650	426 × 240	Português	Legendas com rotação

Tabela 4.1: Descrição de algumas informações de cada vídeo existente na base de testes, como resolução e idioma.

Caso de teste	**Curva utilizada**	**Escala da curva**	**Nº de partições em cada quadro**
Teste 1	Zig-zag	1	1×1
Teste 2	Zig-zag	3	3×3
Teste 3	Zig-zag	5	5×5
Teste 4	Hilbert	1	1×1
Teste 5	Hilbert	2	3×3
Teste 6	Hilbert	3	5×5

Tabela 4.2: Especificação dos casos de testes utilizados nos experimentos.

Para o parâmetro T, definido no algoritmo de segmentação (Seção 3.2), utilizou-se o valor 40. O desvio padrão D, utilizado para a classificação do afinamento (Seção 3.3), foi escolhido como sendo 1. O limiar de erro L do Algoritmo 4 foi estabelecido como 0.1.

Todos os experimentos relatados neste capítulo foram executados em um computador com processdor Pentium IV Dual Core, 2.3 GHz e 1GB RAM. A implementação foi feita na linguagem de programação C++, utilizando a biblioteca gráfica Qt 3.3 [48]. O sistema operacional foi o Linux CentOS 2.6.18.

Para validar o método proposto, foram escolhidos outros três métodos para realizar a comparação. Deles, dois utilizam uma abordagem pixel-a-pixel [1, 52], enquanto que o terceiro [10] utiliza o ritmo visual para efetuar a detecção de legendas. Uma descrição mais detalhada desses trabalhos foi realizada na Seção 2.2.2.

Os autores desses trabalhos não disponibilizaram as bases de teste utilizadas na validação de seus resultados, o que permitiria facilitaria a comparação do nosso método com

os resultados divulgados. Também não foram disponibilizadas as implementações dos métodos, o que nos levou realizar a nossa própria implementação. A vantagem disto é que são evitadas divergências referentes ao custo de processamento, uma vez que os mesmos recursos computacionais foram utilizados nos experimentos.

Como esses três trabalhos serão citados ao longo desta seção, eles serão referidos como Método 1, Método 2 e Método 3, propostos em [52], [1] e [10], respectivamente.

4.2 Análise de Desempenho

Nesta seção serão realizados experimentos para validar o desempenho do método, considerando-se as taxas de quadros por segundo obtidas.

4.2.1 Comparação com outros métodos

A Tabela 4.3 exibe a taxa de quadros por segundo obtida tanto para o método proposto quanto para os outros três utilizados para comparação. Para o método proposto, foi tomado como base o caso de teste 2 (o motivo desta escolha será discutida mais adiante). Pode ser observado que os dois métodos baseados no ritmo visual foram significantemente mais rápidos, se comparados com as abordagens quadro-a-quadro.

Vídeo	**Proposto (fps)**	**Método 1 (fps)**	**Método 2 (fps)**	**Método 3 (fps)**
1	28.09	1.62	5.68	208.43
2	44.67	0.17	3.44	113.32
3	15.22	0.63	1.85	48.62
4	19.31	0.68	2.39	68.09
5	75.12	1.60	0.68	158.59
6	49.83	1.16	3.79	113.53
7	7.64	0.47	1.98	45.16
8	71.3	1.17	5.56	122.28
9	39.8	1.13	3.08	105.18
Média	**39.0**	**0.96**	**3.16**	**109.24**

Tabela 4.3: Comparação da taxa de quadros por segundo.

4.2.2 Análise de desempenho do método

A Figura 4.1 ilustra a variação do comprimento das curvas de Hilbert e zig-zag em função das suas escalas. O eixo das ordenadas exibe a proporção dos pixels da imagem compondo a curva. Pode-se notar que a curva de Hilbert possui o comportamento de uma curva exponencial, enquanto que a curva zig-zag possui crescimento linear.

A Figura 4.2 ilustra o comprimento das curvas de Hilbert e zig-zag em relação às suas escalas e o número de blocos. Pode-se observar que, para ambas as curvas, utilizar uma curva simples e elevar o número de blocos pode gerar uma curva mais curta do que a obtida elevando-se o detalhamento.

O próximo experimento visa analisar o custo de processamento para calcular apenas o ritmo visual. Os resultados para a curva zig-zag são mostrados na Tabela 4.4 e para a curva de Hilbert na Tabela 4.5. Pode ser observado que o custo da curva de Hilbert é menor do que a zig-zag no primeiro teste, porém, torna-se pior nos testes 2 e 3 devido ao fato de possuir crescimento exponencial.

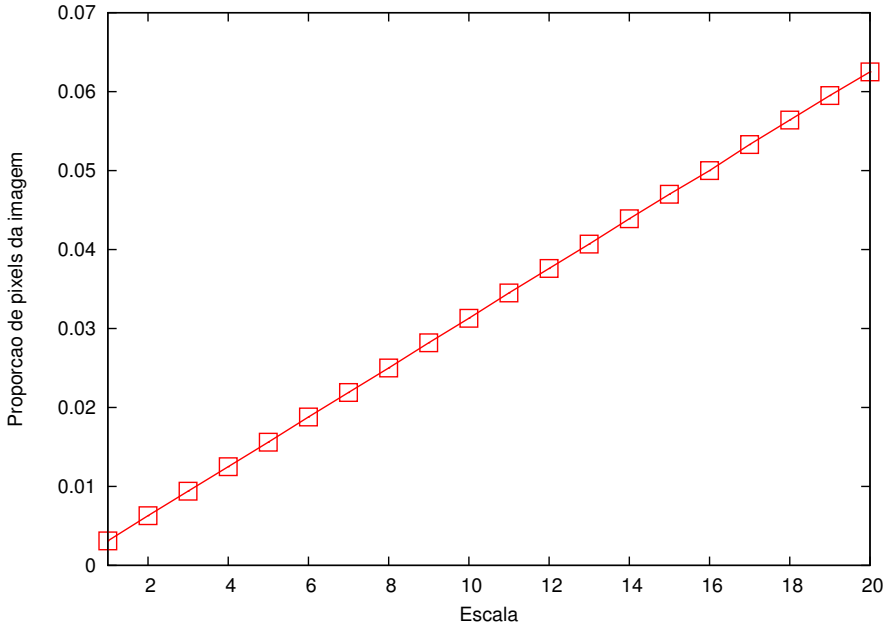
Vídeo	**Teste 1 (fps)**	**Teste 2 (fps)**	**Teste 3 (fps)**
1	326.9	184.3	128.7
2	146.7	90.4	70.4
3	49.8	43.1	34.4
4	72.4	54.3	45.2
5	199.3	141.3	103.0
6	128.3	102.4	73.6
7	53.7	38.5	29.6
8	147.1	93.0	69.1
9	120.6	90.6	63.0
Média	138.3	93.1	68.5

Tabela 4.4: Taxa de quadros por segundo para o cálculo do ritmo visual utilizando a curva zig-zag.

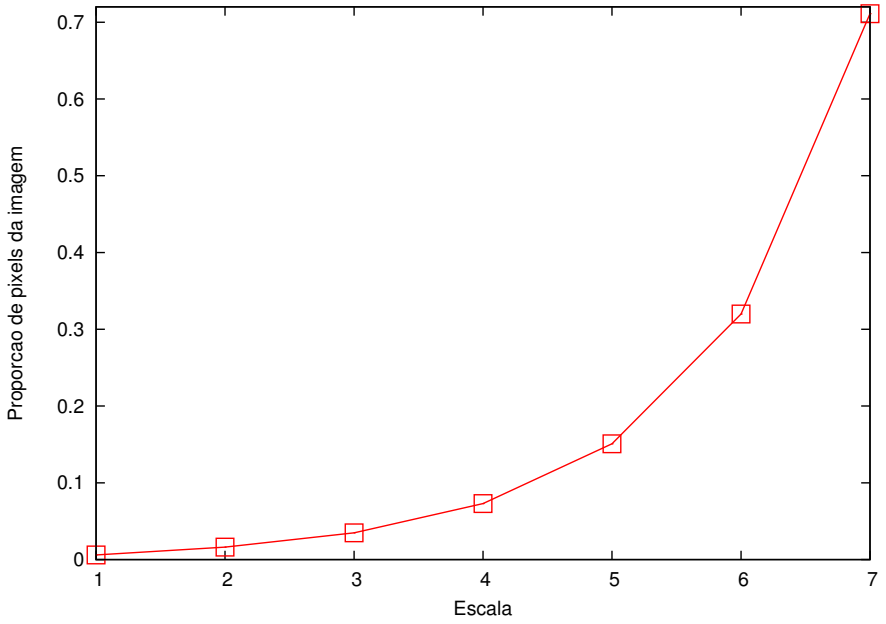
O custo total do algoritmo proposto pode ser visto nas Tabelas 4.6 e 4.7, sendo consideradas todas as etapas do algoritmo, incluindo as de segmentação e classificação.

4.3 Análise da Acurácia

Os experimentos apresentados nesta seção visam analisar os resultados obtidos pelo método proposto, comparados com o *ground truth* de cada vídeo da base. A comparação é feita quadro-a-quadro, realizando-se a seguinte análise:

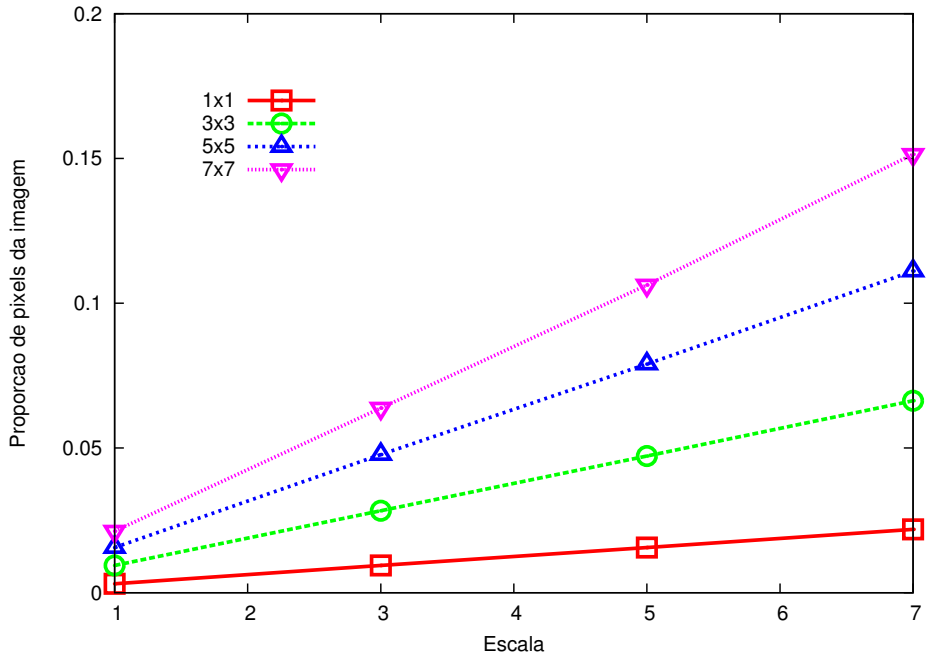


(a) Zig-zag

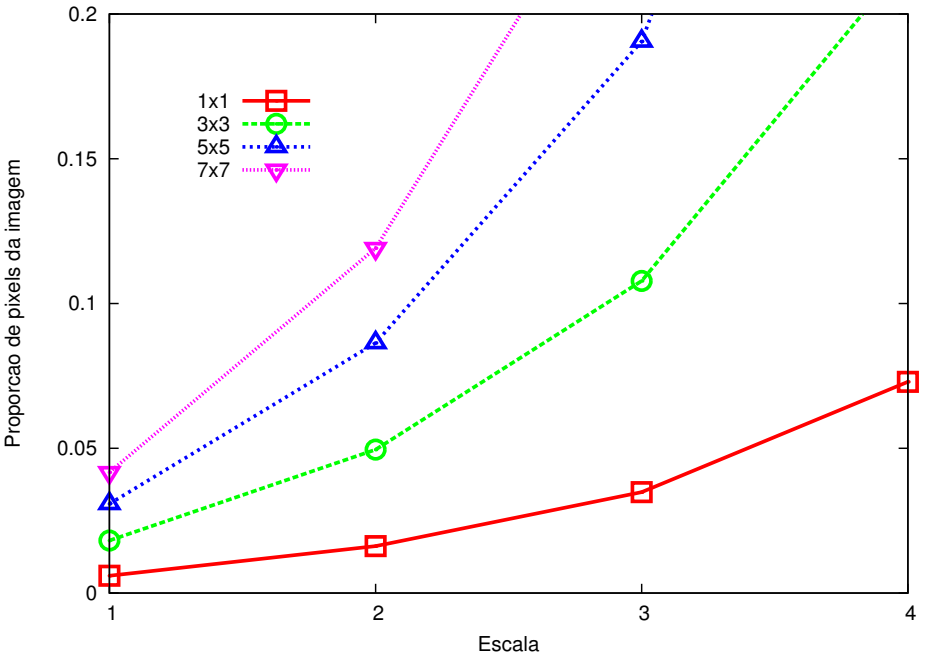


(b) Hilbert

Figura 4.1: Os gráficos mostram como o comprimento de cada curva se comporta de acordo com a escala da mesma. No eixo das ordenadas é exibida a razão entre o comprimento em pixels da curva e o total de pixels da imagem.



(a) Zig-zag



(b) Hilbert

Figura 4.2: Comprimento das curvas de Hilbert e zig-zag, em pixels, quando são variados o grau de detalhamento das curvas e o número de blocos em que cada imagem é particionada.

Vídeo	**Teste 4 (fps)**	**Teste 5 (fps)**	**Teste 6 (fps)**
1	294.65	176.15	74.55
2	149.35	98.06	42.47
3	59.05	45.81	23.72
4	80.63	60.40	28.57
5	214.06	139.72	56.46
6	152.29	107.09	49.10
7	56.92	43.36	23.84
8	181.82	117.34	54.98
9	146.40	97.89	47.86
Média	148.35	98.42	44.62

Tabela 4.5: Taxa de quadros por segundo para o cálculo do ritmo visual utilizando a curva de Hilbert.

Vídeo	**Teste 1 (fps)**	**Teste 2 (fps)**	**Teste 3 (fps)**
1	264.7	28.09	6.93
2	139.37	44.67	13.14
3	42.62	15.22	4.04
4	78.62	19.31	4.76
5	197.72	75.12	29.77
6	145.18	49.83	16.69
7	51.03	7.64	1.14
8	155.71	71.32	31.95
9	128.71	39.80	15.32
Média	131.32	34.27	12.55

Tabela 4.6: Taxa total de quadros por segundo para a curva zig-zag, incluindo as etapas de segmentação e classificação.

- se foi dito corretamente que um quadro possui legenda, então é um acerto positivo (*true positive* ou tp).
- se um quadro não possui legenda, mas foi dito que possui, então é um falso positivo (*false positive* ou fp).
- se um quadro possui legenda, mas foi dito que não possui, então é um falso negativo (*false negative* ou fn).

Vídeo	Teste 4 (fps)	Teste 5 (fps)	Teste 6 (fps)
1	210.24	12.91	2.56
2	118.39	29.36	5.27
3	51.12	10.26	2.06
4	68.08	8.42	3.52
5	176.46	49.8	12.93
6	131.73	40.09	6.69
7	47.59	9.56	1.92
8	159.86	58.37	16.9
9	105.86	25.5	6.79
Média	118.81	29.34	7.81

Tabela 4.7: Taxa de quadros por segundo para a curva de Hilbert, incluindo as etapas de segmentação e classificação.

Com base nessas três regras, a precisão, revocação e *F-Measure* [42] foram computadas de acordo com as Equações 4.1, 4.2 e 4.3.

$$\text{precisão} = \frac{\text{tp}}{\text{tp} + \text{fp}} \tag{4.1}$$

$$\text{revocação} = \frac{\text{tp}}{\text{tp} + \text{fn}} \tag{4.2}$$

$$\textit{F-Measure} = 2 * \frac{\text{precisão} * \text{revocação}}{\text{precisão} + \text{revocação}} \tag{4.3}$$

Estas três medidas são utilizadas para validar vários experimentos mostrados a seguir.

4.3.1 Comparação com outros métodos

A Tabela 4.8 exibe as taxas de precisão, revocação e *F-Measure* para os três métodos utilizados para comparação. Na tabela, estão também presentes os resultados para o método proposto utilizando o caso de teste 2.

Nota-se com estes resultados que o método proposto obteve a melhor taxa de *F-Measure*, 2% acima do segundo lugar. O método 1 possui a revocação mais alta (2.8%), embora a precisão seja inferior (5.4%). O método 2 possui um comportamento oposto, com revocação inferior (35.7%) e precisão superior (1.6%). Já o terceiro método obteve resultados inferiores nas duas medidas.

	Proposto			**Método 1**			**Método 2**			**Método 3**		
Vídeo	**P**	**R**	**FM**	**P**	**R**	**FM**	**P**	**R**	**FM**	**P**	**R**	**FM**
1	73.1	98.1	83.8	77.8	99.2	**87.2**	93.8	80.5	86.7	35.5	64.1	45.7
2	80.3	63.2	70.7	71.6	64.1	67.6	89.0	66.9	**76.3**	43.6	100.0	60.7
3	82.7	84.6	83.6	81.0	100.0	**89.5**	90.5	67.2	77.1	27.8	90.2	42.5
4	82.3	92.9	**87.3**	69.0	99.0	81.3	68.0	18.0	28.5	14.0	92.1	24.3
5	67.8	99.0	80.5	76.3	99.9	**86.5**	76.0	60.4	67.3	48.3	94.8	64.0
6	78.8	97.6	**87.2**	73.5	99.5	84.6	90.0	73.6	81.0	41.8	99.8	58.9
7	94.7	99.7	**97.1**	88.0	99.5	93.4	79.6	33.3	46.9	24.8	94.1	39.2
8	78.7	100.0	**88.1**	61.4	98.5	75.7	65.5	57.1	61.0	73.1	92.9	81.8
9	74.6	99.6	**85.3**	65.4	100.0	79.1	75.1	56.3	64.3	73.8	60.4	66.4
Média	79.2	92.7	**84.8**	73.8	95.5	82.8	80.8	57.0	65.5	42.5	87.6	53.7

Tabela 4.8: Resultados de precisão (P), revocação (R) e *F-Measure* (FM), tanto para os três métodos utilizados para comparação quanto para o método proposto. Para este último, apenas o caso de teste 2 foi considerado por simplicidade e economia de espaço. Para cada vídeo, a melhor taxa de *F-Measure* obtida foi destacada em negrito.

Acurácia do Método Proposto

A fim de estimar o melhor limiar a ser utilizado para a classificação por razão de aspecto, vários valores da razão foram analisados e então construído o gráfico precisão × revocação mostrado na Figura 4.3. Para este experimento, todos os vídeos foram testados utilizando o caso de teste 1, por ser este o mais rápido.

Como pode ser observado, a precisão e a revocação se comportam inversamente proporcionais à variação da razão de aspecto. Para a aplicação do método, propõe-se utilizar a taxa de 0.95, pois este é o valor mais próximo entre a precisão e a revocação, em torno de 0.9.

Escolhido um valor fixo para a razão de aspecto, pode-se executar os experimentos gerais, obtendo-se as medidas de acurácia para os seis casos de testes sugeridos. Esses resultados podem ser vistos nas Tabelas 4.9 e 4.10.

Sobre estes resultados podem ser feitas algumas observações.

- O vídeo 7 da tabela 4.10 obteve precisão e revocação quase zero no teste 1. Isto se deve ao fato que, por a curva ser muito simples, nenhuma legenda foi capturada.

- Na média, temos que a precisão é mais elevada no teste 1 do que no teste 3, e o inverso ocorre com a revocação. Isto é devido ao fato de que, quanto mais detalhada a curva, mais falsos positivos são obtidos, baixando a precisão, enquanto que uma

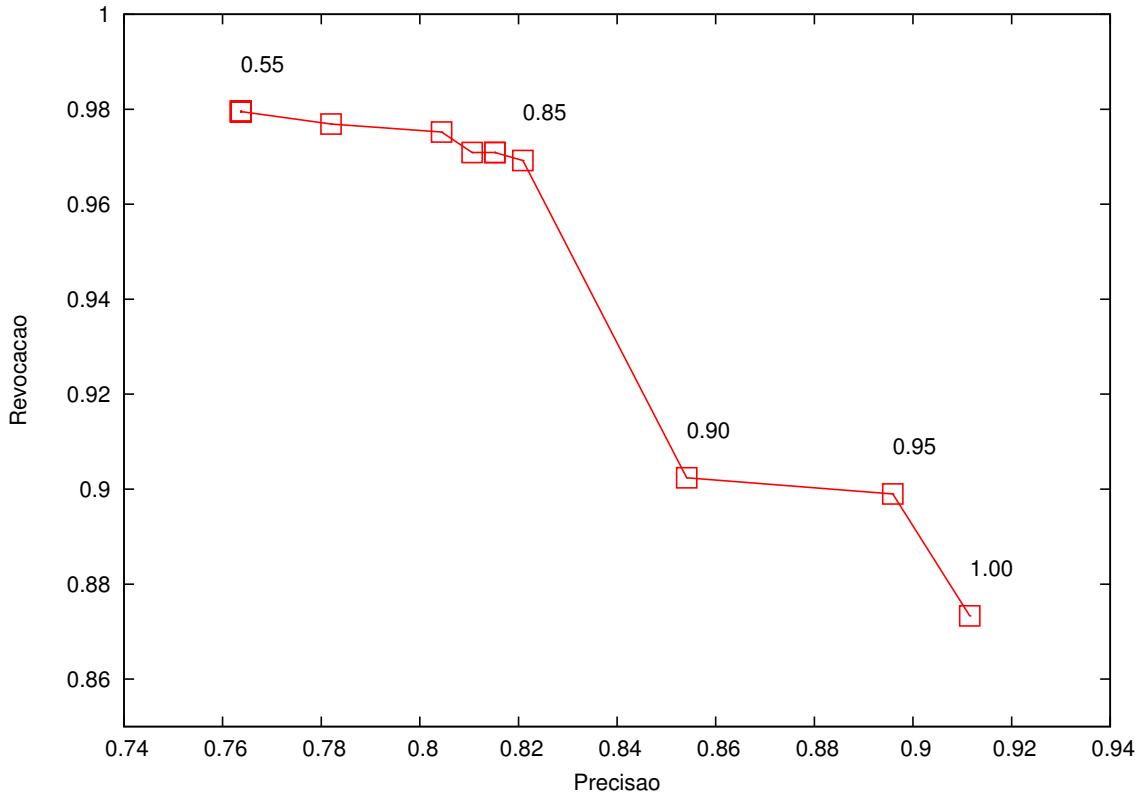


Figura 4.3: Gráfico de precisão × revocação para variações da razão de aspecto (os números que acompanham a curva são os valores da razão).

curva menos detalhada pode não ser capaz de intersectar alguma legenda, gerando falsos negativos e baixando a revocação.

- Apesar deste comportamento da precisão e revocação do item anterior, em alguns casos esta regra não foi seguida, como ocorre com a revocação do vídeo 2 na tabela 4.10. Nela, o valor do teste 5 é inferior ao dos testes 4 e 6. Isto se deve ao fato de que o aumento no detalhamento da curva pode gerar alguma combinação onde legendas sejam perdidas. Apesar disto, este problema ocorre em poucos casos.

Localização dos Pontos Extremos

Se, para uma legenda, não forem detectados apenas o primeiro e o último quadros, a taxa de acerto não será muito penalizada, no entanto, algumas aplicações podem requerer o conhecimento exato dos quadros inicial e final com legendas. Esses quadros de início e fim são denominados, neste trabalho, de pontos extremos, por representarem as extremidades de uma legenda. Cada legenda possui exatamente dois pontos extremos.

Vídeo	**Medidas**	**Teste 1 (%)**	**Teste 2 (%)**	**Teste 3 (%)**
1	Precisão	91.2	73.1	66.8
	Revocação	84.8	98.1	99.0
	F-Measure	87.9	83.8	79.8
2	Precisão	88.4	80.3	81.5
	Revocação	54.9	63.2	80.4
	F-Measure	67.7	70.7	80.9
3	Precisão	99.9	82.7	70.3
	Revocação	24.7	84.6	98.0
	F-Measure	39.6	83.6	81.9
4	Precisão	96.9	82.3	70.6
	Revocação	83.9	92.9	95.5
	F-Measure	89.9	87.3	81.2
5	Precisão	83.5	67.8	66.1
	Revocação	88.8	99.0	99.9
	F-Measure	86.1	80.5	79.6
6	Precisão	85.8	78.8	74.7
	Revocação	88.6	97.6	97.7
	F-Measure	87.2	87.2	84.7
7	Precisão	57.6	94.7	50.0
	Revocação	99.4	99.7	99.8
	F-Measure	72.9	97.1	66.6
8	Precisão	81.6	78.7	78.0
	Revocação	72.6	100.0	100.0
	F-Measure	76.8	88.1	87.6
9	Precisão	96.7	74.6	65.1
	Revocação	98.9	99.6	100.0
	F-Measure	97.8	85.3	78.9
Média	Precisão	86.8	74.8	69.2
	Revocação	77.4	92.7	96.7
	F-Measure	78.4	82.0	80.1

Tabela 4.9: Resultados para os três casos de testes que utilizam a curva zig-zag.

Na Tabela 4.11, a margem de erro na detecção dos pontos extremos das legendas é apresentada. Cada valor corresponde ao erro médio, em quadros, entre o ponto extremo real das legendas e o ponto extremo efetivamente obtido.

A Tabela 4.12 apresenta o erro médio da detecção dos pontos extremos das legendas para todos os métodos considerados. Pode-se observar que o método proposto possui resultados inferiores se comparado com os outros três métodos. Isto se deve provavelmente ao

Vídeo	**Medidas**	**Teste 4 (%)**	**Teste 5 (%)**	**Teste 6 (%)**
1	Precisão	87.3	78.6	72.3
	Revocação	29.5	98.5	99.0
	F-Measure	44.1	87.4	83.6
2	Precisão	89.2	83.8	74.4
	Revocação	60.6	51.1	90.6
	F-Measure	72.2	63.5	81.7
3	Precisão	99.2	70.3	66.2
	Revocação	45.3	90.9	99.7
	F-Measure	62.2	79.3	79.6
4	Precisão	83.3	86.1	77.1
	Revocação	5.5	93.1	95.3
	F-Measure	10.3	89.5	85.2
5	Precisão	63.0	74.7	65.2
	Revocação	26.9	99.5	99.9
	*F-Measur*e	37.7	85.3	78.9
6	Precisão	99.9	82.3	74.3
	Revocação	17.4	97.7	97.7
	F-Measure	29.6	89.3	84.4
7	Precisão	0.9	53.7	54.1
	Revocação	0	99.7	99.7
	F-Measure	0	69.8	70.1
8	Precisão	99.0	82.0	79.1
	Revocação	72.1	81.2	100.0
	F-Measure	83.4	81.6	88.3
9	Precisão	98.7	80.6	65.3
	Revocação	55.9	99.7	100.0
	F-Measure	71.4	89.1	79.0
Média	Precisão	63.9	72.5	66.9
	Revocação	20.3	94.2	98.2
	F-Measure	28.5	80.7	79.1

Tabela 4.10: Resultados para os três casos de testes que utilizam a curva de Hilbert.

método de segmentação adotado em nosso trabalho, o qual não gerou resultados satisfatórios. Neste caso, os métodos 1 e 2 se mostraram superiores, cuja análise quadro-a-quadro demonstra ser menos suscetível a erros em quadros de pontos extremos.

Vídeo	Teste 1	Teste 2	Teste 3	Teste 4	Teste 5	Teste 6
1	3.21	3.00	2.75	2.33	2.91	3.00
2	2.31	2.32	2.50	2.44	2.53	2.31
3	1.70	2.10	1.93	2.13	1.90	2.00
4	2.37	2.40	2.55	×	2.43	2.38
5	2.36	2.21	2.63	1.75	2.50	2.57
6	2.29	2.50	×	2.20	×	4.00
7	2.55	2.46	2.28	×	2.39	2.31
8	3.00	3.00	×	3.33	3.00	3.00
9	1.88	1.33	1.00	1.67	1.75	1.00
Média	**2.41**	**2.37**	**2.23**	**2.26**	**2.43**	**2.51**

Tabela 4.11: Erro médio da localização dos pontos extremos. Cada valor representa a diferença média entre pontos extremos da legenda e os detectados. Os casos em que não foi encontrada nenhuma correspondência com distância menor que 5 quadros foram marcados com ×.

Vídeo	Proposto	Método 1	Método 2	Método 3
1	3.00	1.75	1.74	1.67
2	2.32	1.30	1.25	×
3	2.10	1.87	1.28	1.25
4	2.40	1.39	1.63	0.75
5	2.21	1.22	1.79	1.60
6	2.50	2.00	1.22	4.00
7	2.46	1.00	1.55	1.09
8	3.00	×	1.00	×
9	1.33	1.50	1.6	2.00
Média	**2.37**	**1.50**	**1.45**	**1.77**

Tabela 4.12: Erro médio, em quadros, da localização dos pontos extremos para cada um dos métodos utilizados na comparação.

4.4 Discussão

Esta seção dedica-se a análises dos resultados experimentais obtidos a partir da aplicação da metodologia proposta.

As duas curvas investigadas (Hilbert e zig-zag), apesar de estruturalmente distintas, obtiveram acurácia próximas entre si. A curva de Hilbert não deve ser usada com um grau de detalhamento alto, pois o comprimento da curva tende a tornar o seu uso proibitivo, no entanto, os parâmetros analisados neste trabalho se mostraram suficientes para realizar a

detecção de legendas.

O ajuste dos parâmetros das curvas (escala e blocos) permite variar largamente entre desempenho e acurácia, sendo que o método proposto é, neste sentido, bastante flexível.

A combinação de parâmetros sugerida resultou numa acurácia superior aos outros três métodos comparados para a maioria dos vídeos da base de teste (superior em cinco dos nove vídeos testados). Quanto ao desempenho, o método proposto obteve uma taxa de quadros consideravelmente superior aos método baseados em análise quadro-a-quadro. O método 3 obteve uma taxa de quadros significativamente superior ao método proposto, no entanto, ele foi o que obteve piores resultados em termos de acurácia. Se o nosso método utilizar o conjunto de parâmetros com ênfase em desempenho, tanto acurácia quanto desempenho serão superiores ao método 3.

A seguir serão identificados alguns pontos negativos do método proposto, bem como apontados alguns problemas ainda sem solução satisfatória para o problema.

Super-segmentação

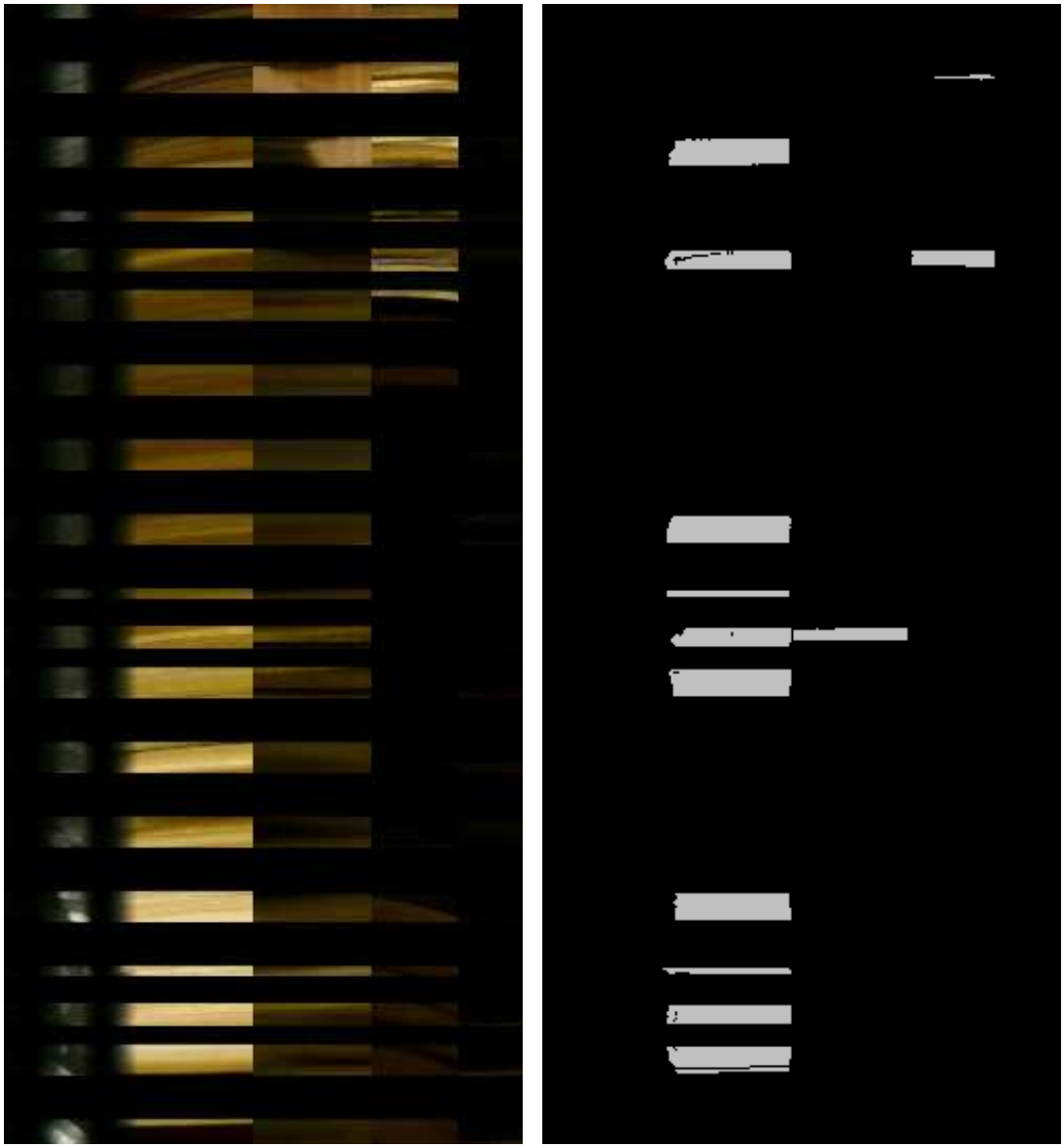
Os resultados apresentados nas Tabelas 4.9 e 4.10 mostram a tendência de queda da precisão à medida que o comprimento das curvas aumenta. Isso se deve pelo fato de que, quanto maior a curva, mais redundância das informações serão inseridas no ritmo visual. Se uma legenda for intersectada mais de uma vez pela curva, ela será representada por duas regiões distintas no ritmo visual e, com isso, há maiores chances de ela ser detectada. No entanto, a partir de um certo nível, a elevada complexidade da curva acaba inserindo padrões no ritmo visual que entram na definição de legenda.

Na Figura 4.4(a), pode-se observar um fragmento de ritmo visual construído com uma curva zig-zag, escala 5 e blocos 5×5, em um vídeo que não possui legendas. Vários padrões retangulares podem ser gerados a partir dessas cenas naturais, alguns dos quais com dimensões suficientes para caracterizar legendas. A Figura 4.4(b) mostra como muitas dessas regiões foram erroneamente classificadas como legendas, comprovando que uma curva muito complexa pode aumentar a taxa de falsos positivos do método.

Cenas com pouco movimento

Um problema foi identificado quando há cenas no vídeo com muitos objetos sem movimento. Esta situação é apresentada na Figura 4.4, em que grande parte dos objetos não se move na cena e não há movimento de câmera (de fato, o contexto da cena é que o tempo foi paralisado). A Figura 4.4 apresenta o resultado da classificação, em que foi detectada como legenda uma faixa na altura média do quadro, gerando um falso positivo.

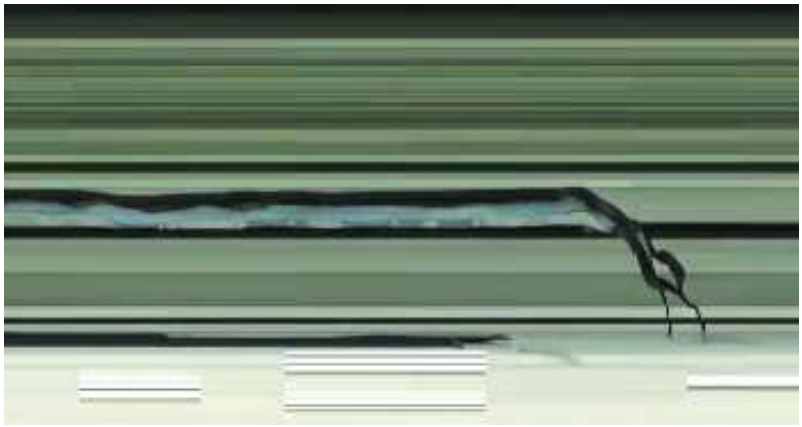
Apesar de existirem muitos padrões de objetos imóveis, apenas um foi detectado como legenda. Isso se deve ao fato de que todos os outros foram rejeitados devido às suas



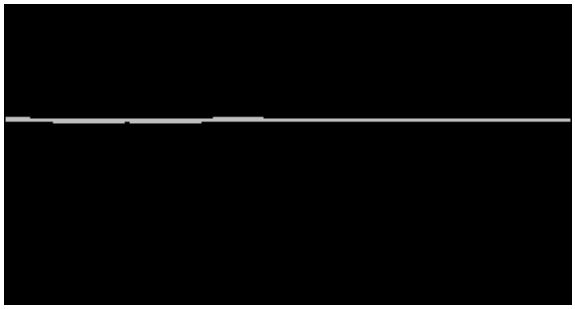
(a) Ritmo (b) Resultado

Figura 4.4: Em (a), exemplo de um ritmo visual construído a partir de uma curva de complexidade alta. Em (b), o resultado após a classificação deste ritmo. As regiões cinzas representam as legendas encontradas no ritmo visual, entretanto, todas estão incorretas uma vez que não há legendas no vídeo em questão.

outras propriedades que não a imobilidade como, por exemplo, as suas dimensões ou baixo contraste com o fundo.



(a) Ritmo



(b) Resultado

Figura 4.5: Em (a), um ritmo visual extraído de uma cena em que não há movimento de câmera nem de grande parte dos objetos. Em (b), o resultado da classificação, tal que uma faixa foi detectada como legenda, gerando um falso positivo.

Falhas na segmentação

Devido ao fato de o algoritmo de segmentação utilizar um limiar global, eventuais falhas em regiões de pouca variação de intensidade de cores podem ocorrer. A Figura 4.6 apresenta um trecho de ritmo visual onde há três legendas, representadas pelos retângulos na parte inferior. No entanto, uma das legendas está conectada a uma faixa fina vertical. Devido ao fato de todos esses padrões possuírem intensidade de cor muito próximas, a segmentação pode formar uma única região entre a terceira legenda e a faixa vertical. Se isto ocorrer, a legenda será eliminada por possuir uma razão de aspecto fora do padrão.

Intervalo entre legendas

Uma das restrições das legendas, discutida na Seção 2.2.3, é a obrigatoriedade de existir um intervalo de tempo entre duas legendas consecutivas, ou seja, deve haver pelo menos um quadro sem legenda após o término de uma legenda e antes do início da próxima.

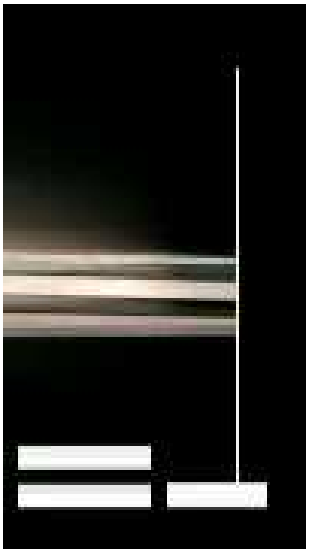
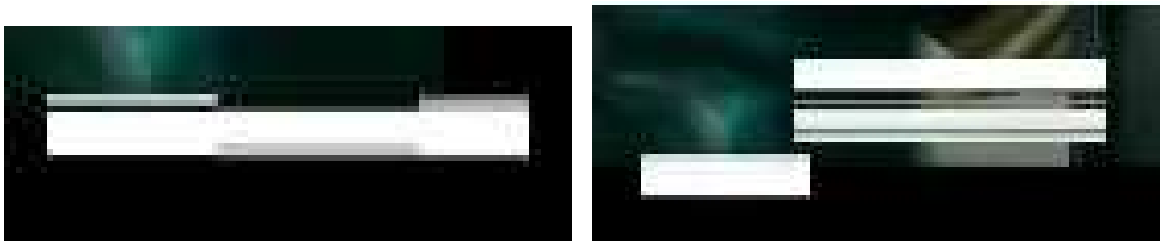


Figura 4.6: Exemplo de legenda com possibilidade de se fundir com uma reta vertical de intensidade de cor semelhante.

Esta restrição existe porque, se duas legendas consecutivas se sobrepuserem no ritmo visual, a dificuldade para detectá-las aumenta. Um exemplo é mostrado na Figura 4.7(a), em que três legendas se fundiram em apenas um retângulo. Esta configuração apresenta dois problemas: primeiro, o fato de dificultar a descoberta dos pontos extremos de cada legenda e, segundo, que a região resultante pode se distanciar de um retângulo, aumentando as chances das legendas não serem detectadas.

Um caso semelhante ocorre na Figura 4.7(b), em que não apenas não houve o espaçamento exigido entre as legendas como a segunda legenda se iniciou antes mesmo de que a primeira desaparecesse. Nesse caso em particular, não houve sobreposição entre as legendas devido ao fato de haver um desnível na altura delas.



(a) (b)

Figura 4.7: Dois exemplos de legendas que não respeitam a exigência de intervalo de tempo. Em (a), houve uma sobreposição entre três legendas. Em (b), o problema foi evitado devido à diferença de altura das legendas no ritmo visual.

Capítulo 5

Conclusões e Trabalhos Futuros

5.1 Conclusões

Este trabalho propôs um método rápido de detecção de legendas em quadros de vídeos, mantendo uma baixa taxa de falsos negativos. Uma representação de imagens, denominada ritmo visual, é criada pela da subamostragem de cada quadro de um vídeo.

Um conjunto de regras é utilizado para classificar como legendas as regiões extraídas do ritmo visual. O método é robusto a legendas localizadas em qualquer posição e orientação dos quadros, assim como à formatação dos textos ou línguas.

As duas curvas de preenchimento do espaço analisadas, Hilbert e zig-zag, mostraram-se equivalentes na terefa de construir o ritmo visual. A segmentação foi simples e capaz de processar eficientemente o ritmo visual, enquanto que as regras de classificação conseguiram separar, de forma eficaz, as regiões com ou sem legendas.

A comparação com outros métodos existentes na literatura revelou que o método proposto obteve uma acurácia superior (cerca de 2%), ao mesmo tempo em que a velocidade de processamento foi significativamente superior quando comparado com algumas abordagens (até 40 vezes mais rápido).

5.2 Trabalhos Futuros

Algumas sugestões de extensão deste trabalho são indicadas a seguir:

- investigar métodos de segmentação mais adaptados ao problema. A segmentação é uma etapa fundamental no processo proposto de detecção das legendas, enquanto que o método utilizado neste trabalho visa ser simples para obter uma boa velocidade de processamento. O estudo de outros métodos que possam substitui-lo

adequadamente é recomendado. Pode ser sugerido um método de segmentação baseado em cores, dado que as legendas costumam possuir a mesma cor dentro de um único vídeo.

- propor mecanismos para o processamento do vídeo por intervalos de tempo menores, a fim de evitar a contrução de um ritmo visual único para o vídeo inteiro, reduzindo assim o consumo de memória utilizada. Uma sugestão seria utilizar um ritmo visual circular que, nesse caso, manteria em memória somente a última porção analisada do vídeo.

- realizar uma análise incremental dos vídeos. Atualmente, é necessário processar o vídeo inteiro antes do início da etapa de detecção de legendas. Novamente, o ritmo visual circular seria uma sugestão. Atraso entre a geração de uma legenda e a sua detecção deveria ser evitado.

- aperfeiçoar o método proposto visando à detecção de legendas com movimentos, como por exemplo, translações verticais ou horizontais. Quando há uma translação com velocidade constante, as legendas formam retângulos com rotação no ritmo visual, entretanto, quando a velocidade varia, formas geométricas mais complexas são obtidas. O fluxo óptico aplicado sobre o ritmo visual pode ajudar a resolver este problema, permitindo analisar o comportamento da legenda ao longo do tempo.

- não foi levado em conta neste trabalho o fato de as legendas surgirem com mais frequência em regiões específicas dos quadros. Esta informação pode contribuir no processo de detecção, porém, uma análise mais detalhada é necessária para explorar tal fato.

- quando os quadros de um vídeo são fragmentados em vários blocos, os ritmos visuais são "empilhados" um sobre o outro. Neste trabalho, considerou-se que a ordem em que este empilhamento é feito não altera os resultados obtidos, no entanto, esta hipótese não foi devidamente validada. Formas mais promissoras de combinar vários ritmos visuais em um único poderiam também ser investigadas.

Bibliografia

[1] L. Agnihotri and N. Dimitrova. Text Detection for Video Analysis. In *IEEE Workshop on Content-Based Access of Image and Video Libraries*, pages 109–113, 1999.

[2] G. Banon and J. Barrera. *Bases da Morfologia Matemática para Análise de Imagens Binárias*. INPE, second edition, July 1998.

[3] M. Bertini, C. Colombo, and A. D. Bimbo. Automatic Caption Localization in Videos using Salient Points. In *IEEE Conference on Multimedia and Expo*, pages 68–71, Tokyo, Japan, Aug. 2001.

[4] F. N. Bezerra and E. Lima. Low Cost Soccer Video Summaries based on Visual Rhythm. In *8th ACM International Workshop on Multimedia Information Retrieval*, pages 71–78, Santa Barbara, CA, USA, 2006. ACM.

[5] D. Chen, H. Bourlard, and J.-P. Thiran. Text Identification in Complex Background Using SVM. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 621–626, 2001.

[6] D. Chen, J. M. Odobez, and H. Bourlard. Text Detection and Recognition in Images and Video Frames. *Pattern Recognition*, 37(3):595–608, Mar. 2004.

[7] D.-Y. Chen, M.-H. Hsiao, and S.-Y. Lee. Automatic Closed Caption Detection and Font Size Differentiation in MPEG Video. In *5th International Conference on Recent Advances in Visual Information Systems*, pages 276–287, Hsi Chu, Taiwan, Mar. 2002.

[8] A. Cherri and M. Karim. Optical Symbolic Substitution - Edge Detection using Prewitt, Sobel, and Roberts Operators. *Applied Optics*, 28:4644–4648, Nov. 1989.

[9] M.-C. Chi, C.-H. Yeh, and M.-J. Chen. Robust Region-of-Interest Determination Based on User Attention Model Through Visual Rhythm Analysis. *IEEE Transactions on Circuits and Systems for Video Technology*, 19(7):1025–1038, July 2009.

[10] S. S. Chun, K. Hyeokman, K. Jung-Rim, O. Sangwook, and S. Sanghoon. Fast Text Caption Localization on Video Using Visual Rhythm. In *5th International Conference on Recent Advances in Visual Information Systems*, pages 259–268, London, UK, UK, 2002.

[11] C. Cortes and V. Vapnik. Support-Vector Networks. *Machine Learning*, 20:273–297, 1995.

[12] F.-C. Crow. The Aliasing Problem in Computer-Generated Shaded Images. *Communications of the ACM*, 20:799–805, Nov. 1977.

[13] I. Daubechies. Orthonormal bases of compactly supported wavelets. *Communications on Pure and Applied Mathematics*, 41(7):909–996, 1988.

[14] H. Freeman. Computer Processing of Line-Drawing Images. *ACM Computing Surveys*, 6:57–97, Mar. 1974.

[15] R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2nd edition, 2001.

[16] J.-D. Gould, L. Alfaro, R. Finn, B. Haupt, and A. Minuto. Why Reading was Slower from CRT Displays than from Paper. In *Conference on Human Factors in Computing Systems and Graphics Interface*, volume 17, pages 7–11, New York, NY, USA, May 1987. ACM.

[17] S. J. F. Guimarães, M. Couprie, A. A. Araújo, and N. J. Leite. A Method for Cut Detection Based on Visual Rhythm. In *XIV Brazilian Symposium on Computer Graphics and Image Processing*, pages 297–304. IEEE Computer Society, 2001.

[18] S. J. F. Guimarães, M. Couprie, A. A. Araújo, and N. J. Leite. Video Segmentation Based on 2D Image Analysis. *Pattern Recognition Letters*, 24:947–957, Apr. 2003.

[19] S. M. Hanif and L. Prevost. Texture based Text Detection in Natural Scene Images - A Help to Blind and Visually Impaired Persons. In *Conference and Workshop on Assistive Technologies for People with Vision and Hearing Impairments: Assistive Technology for All Ages*, Granada, Spain, Aug. 2007.

[20] R. M. Haralick, K. Shanmugam, and I. Dinstein. Textural Features for Image Classification. *IEEE Transactions on Systems, Man and Cybernetics*, 3(6):610–621, 1973.

[21] H. V. Jagadish. Analysis of the Hilbert Curve for Representing Two-Dimensional Space. *Information Processing Letters*, 62(1):17–22, 1997.

[22] R. Ji, P. Xu, H. Yao, Z. Zhang, X. Sun, and T. Liu. Directional Correlation Analysis of Local Haar Binary Pattern for Text Detection. In *IEEE International Conference on Multimedia and Expo*, Apr. 2008.

[23] C. Jung and J. Kim. A Novel Approach for Key Caption Detection in Golf Videos Using Color Patterns. *ETRI Journal*, 30(5):750–752, 2008.

[24] K. Jung, K. I. Kim, and A. K. Jain. Text Information Extraction in Images and Video: A Survey. *Pattern Recognition*, 34(5):977–997, 2004.

[25] F. Karamitroglou. A Proposed Set of Subtitling Standards in Europe. *Translation Journal*, 2(2), Feb. 2007.

[26] W. Kong, R. Yao, and H. Lu. A New Scene Breakpoint Detection Algorithm Using Slice of Video Stream. In *IAPR International Workshop on Multimedia Information Analysis and Retrieval*, pages 175–180, London, UK, 1998. Springer-Verlag.

[27] C.-C. Lee, Y.-C. Chiang, H.-M. Huang, and C.-L. Tsai. A Fast Caption Localization and Detection for News Videos. In *International Conference on Innovative Computing, Information and Control*, pages 226–229, Los Alamitos, CA, USA, Sept. 2007.

[28] H. Li, D. Doermann, and O. Kia. Automatic Text Detection and Tracking in Digital Video. *IEEE Transactions on Image Processing*, 9(1):147–156, Jan. 2000.

[29] J. Liang, D. Doermann, and H. Li. Camera-Based Analysis of Text and Documents: A Survey. *International Journal of Document Analysis and Recognition*, 7(2–3), July 2005.

[30] R. Lienhart. Automatic Text Recognition for Video Indexing. In *Fourth ACM international Conference on Multimedia*, pages 11–20, Boston, MA, USA, 1996.

[31] R. Lienhart. Automatic Text Recognition for Video Indexing. In *Fourth ACM International Conference on Multimedia*, pages 11–20, Boston, MA, USA, Nov. 1996.

[32] R. Lienhart. Video OCR: A Survey and Practitioner's Guide. In *VideoMining*, page Chapter 6, 2003.

[33] R. Lienhart and A. Wernicke. Localizing and Segmenting Text in Images and Videos. *IEEE Transactions on Circuits and Systems for Video Technology*, 12(4):256–268, Apr. 2002.

[34] R. Lienhart and A. Wernicke. Localizing and Segmenting Text in Images and Videos. *IEEE Transactions on Circuits and Systems for Video Technology*, 12(4):256–268, Apr. 2002.

[35] LIV. Video Sequences, 2010. `http://www.liv.ic.unicamp.br/~felipebvalio/base.zip`.

[36] S. Lu, I. King, and M. Lyu. A Novel Video Summarization Framework for Document Preparation and Archival Applications. In *IEEE Aerospace Conference*, pages 1–10, Mar. 2005.

[37] B. Luo, X. Tang, J. Liu, and H. Zhang. Video Caption Detection and Extraction using Temporal Information. In *International Conference on Image Processing*, volume 1, pages I–297–300, Sept. 2003.

[38] C. Ngo, T. Pong, and R. Chin. Detection of Gradual Transitions through Temporal Slice Analysis. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 36–41, 1999.

[39] C.-W. Ngo and C.-K. Chan. Video Text Detection and Segmentation for Optical Character Recognition. *Multimedia Systems*, 22(9):261–272, 2005.

[40] C. W. Ngo, T. C. Pong, and R. T. Chin. Camera Break Detection by Partitioning of 2D Spatio-Temporal Images in MPEG Domain. In *IEEE International Conference on Multimedia Computing and Systems*, pages 750–755, Florence, Italy, 1999. IEEE Computer Society.

[41] T. Ojala, M. Pietikäinen, and D. Harwood. A Comparative Study of Texture Measures with Classification based on Feature Distributions. *Pattern Recognition*, 29:51–59, 1996.

[42] C. J. V. Rijsbergen. *Information Retrieval*. Butterworth-Heinemann, Newton, MA, USA, 2nd edition, 1979.

[43] P. L. Rosin. Measuring Shape: Ellipticity, Rectangularity, and Triangularity. In *15th International Conference on Pattern Recognition*, volume 1, pages 952–955, 2000.

[44] H. Sagan. *Space-Filling Curves*. Springer, 1994.

[45] R. Schafer and T. Sikora. Digital Video Coding Standards and their Role in Video Communications. In *Proceedings of IEEE*, pages 907–924, 1995.

[46] A. R. Smith. Color Gamut Transform Pairs. *SIGGRAPH Computer Graphics*, 12:12–19, Aug. 1978.

[47] X. Tang, X. Gao, J. Liu, and H. Zhang. A Spatial-Temporal Approach for Video Caption Detection and Recognition. *IEEE Transactions on Neural Networks*, 13(4):961–971, July 2002.

[48] Trolltech. Qt 3.3 Reference Site, 2005. `http://doc.qt.nokia.com/3.3/index.html`.

[49] V. Vijayakumar and R. Nedunchezhian. A Novel Method for Super Imposed Text Extraction in a Sports Video. *International Journal of Computer Applications*, 15(1):1–6, Feb. 2011. Published by Foundation of Computer Science.

[50] P. Viola and M. J. Jones. Robust Real-Time Face Detection. *International Journal of Computer Vision*, 57(2):137–154, 2004.

[51] G.-K. Wallace. The JPEG Still Picture Compression Standard. *Communications of the ACM*, 34(4):30–44, 1991.

[52] J.-C. Wu and J.-W. H. Y.-S. Chen. Morphology-based Text Line Extraction. *Machine Vision and Applications*, 19(3):195–207, 2008.

[53] V. Wu, , R. Manmatha, and E. M. Riseman. Finding Text in Images. In *Second ACM International Conference on Digital Libraries*, pages 3–12, Philadelphia, PA, USA, 1997.

[54] J. Xu and S. Li. Automatic Caption Detection in Video Frames Based on Support Vector Machine. In J. Wang, X.-F. Liao, and Z. Yi, editors, *Advances in Neural Networks - ISNN 2005*, volume 3497 of *Lecture Notes in Computer Science*, pages 813–813. Springer Berlin / Heidelberg, 2005.

[55] Q. Ye, J. Jiao, J. Huang, and H. Yu. Text Detection and Restoration in Natural Scene Images. *Journal of Visual Communication and Image Representation*, 18(6):504–513, 2007.

[56] H. Zhang, W. Gao, and X. C. D. Zhao. Object Detection using Spatial Histogram Features. *Image and Vision Computing*, 24(4):327–341, Apr. 2006.

[57] T. Y. Zhang and C. Y. Suen. A Fast Parallel Algorithm for Thinning Digital Patterns. *Communications of the ACM*, 27:236–239, Mar. 1984.

[58] Y. Zhong, H. Zhang, and A. Jain. Automatic Caption Localization in Compressed Video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(4):385–392, Apr. 2000.