Histograma de Palavras Visuais para Caracterização de Texturas e Cenas Dinâmicas

Wesley Eiji Sanches Kanashiro
Orientador: Wesley Nunes Gonçalves

FACOM - Universidade Federal de Mato Grosso do Sul
Agosto/2016

Agradecimentos

A Deus, por proporcionar tudo em minha vida.

Ao meu orientador, professor Wesley Nunes Gonçalves, pela atenção, dedicação, paciência e amizade ao longo destes dois anos. Ele mostrou que a distância não foi um empecilho para a orientação do mestrado.

Aos professores e membros da banca, Hemerson Pistori e Jonathan de Andrade Silva, pelos direcionamentos e contribuições.

Aos meus pais Gerson e Fernanda, que mesmo de longe me incentivaram nesta jornada.

À minha avó Yolanda, que não está aqui para presenciar a concretização deste sonho, mas com certeza me ajudou e torceu por mim a todo momento.

À Camila Higa, pela compreensão e incentivo, e também pelo apoio nos momentos mais difíceis.

Aos meus amigos queridos que sempre estiveram comigo e torceram por mim desde o processo seletivo para ingresso no mestrado até agora.

A todos os professores da FACOM que contribuíram de forma direta ou indireta na minha formação.

À CAPES pelo apoio financeiro.

Resumo

A caracterização de vídeos vem sendo pesquisada cada vez mais na área de visão computacional por ser um tema desafiador. Caracterizar vídeos não é uma tarefa trivial, pois é preciso levar em consideração tanto a informação espacial (aparência), quanto a informação temporal (movimento). As texturas dinâmicas são um caso particular de vídeos, que podem ser definidas como movimentos de padrões que apresentam propriedades estacionárias ao longo do espaço e tempo. Exemplos de textura dinâmica podem ser encontrados em situações do dia-a-dia como por exemplo, em sequências de imagens de ondas do mar, fumaça, fogo, escada rolante, entre outras. Outro caso particular de vídeos são as cenas dinâmicas, que são composições de uma ou mais texturas dinâmicas, mas com um local ou cenário caracterizando-as. Este trabalho tem por objetivo estender o Histograma de Palavras Visuais (BoVW) para caracterização de texturas e cenas dinâmicas. O BoVW é aplicado em três planos ortogonais do vídeo para que sejam obtidas informações espaciais e de movimento, melhorando assim, a caracterização de vídeos. Para avaliar a proposta deste trabalho, experimentos foram realizados em duas bases de vídeos: tráfego de carros e cenas dinâmicas. Os resultados foram comparados com os obtidos por métodos da literatura e em ambas as bases de vídeos, o método proposto apresentou resultados promissores. Na base de cenas dinâmicas, pode-se concluir que a inclusão da informação de movimento para caracterização dos vídeos aumentou consideravelmente a taxa de classificação correta. Enquanto que na base de tráfego de carros, a informação temporal não influenciou de forma tão considerável a taxa de classificação correta, apesar de contribuir de certa forma na caracterização dos vídeos.

Palavras-chave: Texturas dinâmicas, Histograma de palavras visuais, Cenas dinâmicas, Planos ortogonais

Abstract

The video characterization has been widely studied in computer vision because it is a fundamental challenge. Video characterization is not a basic task, because it is necessary to take both the spatial (appearence), and temporal information (movement). Dynamic textures are a particular video case, that can be defined as movement patterns which have stationary properties in spacetime. Examples of dynamic texture can be found in real situation such as, sequence of images of sea waves, smoke, fire, escalator, among others. Another particular video case are dynamic scenes that are compositions of one or more dynamic textures, with a location or scene characterizing them. This work aims to extend the Bag of Visual Words (BoVW) to characterize dynamic textures and dynamic scenes. The BoVW is applied on three orthogonal planes to obtain spatial and motion information, for improving the video characterization. To evaluate the proposal of this work, experiments were conducted on two video databases: car traffic and dynamic scenes. The results were compared with those obtained by literature methods and, in both video databases, the proposed method showed promising results. On the dynamic scenes database, it can be concluded that the inclusion of motion information to video characterization greatly increased the correct classification rate. Nevertheless, on the car traffic database, the motion information did not significantly influence the correct classification rate, despite contributing to the video characterization.

Keywords: Dynamic textures, Bag of visual words, Dynamic scenes, Orthogonal planes

Sumário

Lista de Figuras

Lista de Tabelas

CAPÍTULO

1

Introdução

1.1 Motivação e Justificativa

Em visão computacional, a extração de características é uma etapa de extrema importância para a classificação de imagens, que por sua vez, possui diversas áreas de aplicações, tais como médica (diagnóstico de tumores e câncer) [41], reconhecimento de caracteres (placas automotivas) [21], agricultura de precisão (doenças em folhas de soja) [29], entre outras. Dada uma imagem, é possível extrair suas características utilizando um dos métodos mais importantes atualmente da visão computacional, o Histograma de Palavras Visuais - BoVW [42] (do inglês *Bag-of-Visual-Words*, para que posteriormente ela possa ser classificada com base em um conjunto de dados.

O BoVW consiste em construir um vocabulário de palavras visuais a partir de pontos de interesse detectados e descritos previamente com descritores locais. Após a construção do vocabulário, dada uma nova imagem, seus pontos de interesse são rotulados em palavras visuais e um histograma com a frequência de cada palavra é obtido para caracterizar a imagem. A detecção e descrição de pontos de interesse em uma imagem pode ser desempenhada eficientemente por um dos seguintes descritores locais conhecidos na literatura: SIFT (*Scale Invariant Feature Transform*) [25], SIFT Denso [24], SURF (*Speeded Up Robust Features*) [2], PHOW (*Pyramid Histogram Of Visual Words*) [3], HOG (*Histograms of Oriented Gradients*) [10] e o LBP (*Local Binary Patterns*) [27].

Devido a sua importância e desempenho, o BoVW foi estendido para caracterização de vídeos que, por sua vez, também possuem várias aplicações, tais como monitoramento de tráfego de carros [4], ferrugem [45], reconhecimento de expressões faciais [43]. Porém, caracterizar vídeos não é uma tarefa trivial, pois não basta apenas segmentar um determinado vídeo ou aplicar um descritor para cada quadro. Mais do que isso, é necessário também caracterizar a informação de tempo, ou seja, como um determinado ponto de interesse varia ao longo do tempo. Caso a informação temporal seja descartada, uma caracterização incorreta pode ocorrer pelo fato de que, dois vídeos distintos, podem ter as mesmas variações de intensidade no espaço (pontos de interesse), mas em movimentos distintos.

Um caso particular de vídeos são as texturas dinâmicas, que vêm sendo cada vez mais pesquisadas na área de visão computacional devido aos avanços em equipamentos, recursos

de captura e o processamento computacional. Apesar da dificuldade na definição do termo textura, suas características estão diretamente relacionadas às propriedades físicas dos objetos, tornando-a importante para as aplicações na área de visão computacional [18]. Texturas dinâmicas são movimentos de padrões que apresentam propriedades estacionárias ao longo do espaço e tempo, ou seja, as características de imagens de textura são estendidas para o domínio espaço-temporal, em que o domínio espacial refere-se à aparência, enquanto que o domínio temporal se refere ao movimento da textura [19]. Exemplos de textura dinâmica podem ser encontrados em situações reais que estão em sequências de imagens de ondas do mar, fumaça, fogo, bandeira balançando ao vento, escada rolante e grupo de pessoas andando [7].

Outro caso particular de vídeos são as cenas dinâmicas, que vem sendo pesquisadas recentemente. Por conta disso, há também certa dificuldade na definição deste termo. De acordo com Qi et. al [31] e Derpanis et. al [11], cenas dinâmicas estão diretamente ligadas às texturas dinâmicas com um cenário ou lugar por trás que as caracterizem. Além disso, uma cena dinâmica pode também ser a composição de várias texturas dinâmicas em um mesmo cenário ou lugar, como por exemplo, uma bandeira balançando ao vento e a água de uma fonte em uma determinada praça.

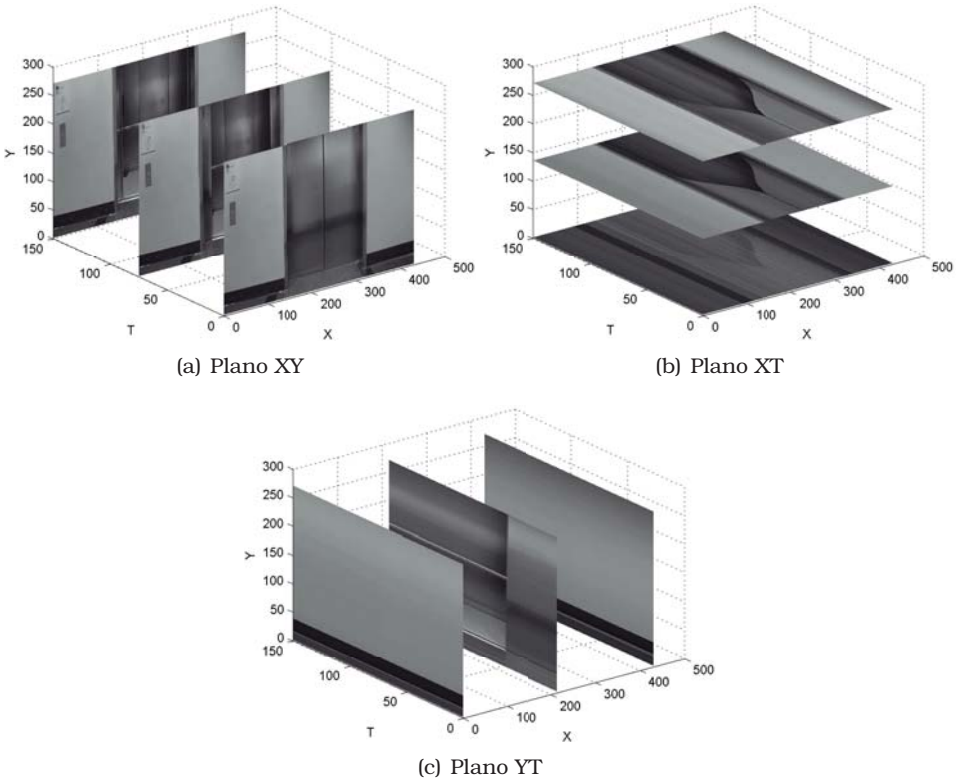
(a) Somente textura dinâmica (b) Composição de texturas dinâmicas (c) Textura dinâmica em um determinado local

Figura 1.1: Exemplos de texturas e cenas dinâmicas através de quadros extraídos de vídeos da base de dados de cenas dinâmicas [11]. Em (a) é mostrada apenas a água do mar em movimento. Em seguida (b) é mostrada uma cena composta por duas texturas dinâmicas: bandeira e água da fonte em movimento. Por fim é mostrado um cenário ou local onde há água de uma fonte em movimento (c).

A Figura 1.1 ilustra três situações de texturas dinâmicas. A primeira (Figura 1.1a) exemplifica uma textura dinâmica que corresponde a água do mar em movimento; as Figuras 1.1b e 1.1c exemplificam cenas dinâmicas em duas situações distintas: uma representando a composição de texturas dinâmicas e outra representando a textura dinâmica em um determinado local ou cenário, com um fundo praticamente estático. Dessa forma, texturas dinâmicas podem ser definidas como repetições de padrões tanto espaço quanto no tempo e, considerando que não há uma definição formal na literatura, cenas dinâmicas podem ser definidas como composições de uma ou mais texturas dinâmicas com um lugar ou cenário caracterizando-as.

Uma maneira simples de usar o BoVW na caracterização de vídeos é obter um descritor local para cada quadro do vídeo, e ao final, uni-los em um único descritor correspondente ao vídeo. Porém, esta estratégia fornece apenas características de aparência em função do tempo. Uma outra alternativa, é utilizar o SIFT 3D, proposto por Scovanner et al. [33], para gerar os descritores para o BoVW. O SIFT 3D consiste em extrair características espaciais para cada ponto, incluindo também a informação temporal no seu descritor. Portanto, este trabalho tem como objetivo propor uma extensão do BoVW aplicando o descritor local nos três planos ortogonais do vídeo, denominados planos XY, XT e YT. O plano XY fornece informação relacionada à aparência enquanto que os planos XT e YT remetem informações de movimento. Dessa forma, a caracterização dos vídeos usando os planos ortogonais é melhorada, conside-

rando a informação temporal em conjunto com a informação espacial. Para cada plano, um vocabulário é aprendido e um histograma é obtido. Os histogramas dos três planos são concatenados para formar o vetor de características. A Figura 1.2 ilustra os três planos ortogonais de um vídeo de elevador.



(a) Plano XY (b) Plano XT

(c) Plano YT

Figura 1.2: Os planos ortogonais XY, XT e YT. Os quadros foram extraídos de um vídeo de elevador da base de cenas dinâmicas [11]. O vídeo é representado por um cubo sobre um plano 3D e possui dimensões $X = 452$, $Y = 270$ e $T = 150$.

Neste trabalho, o método proposto foi comparado com o BoVW tradicional utilizando o descritor local SIFT 3D, e com duas técnicas de caracterização de textura dinâmica baseadas nos padrões locais binários (RI-VLBP[1] e LBP-TOP[2]). Para isso, experimentos foram realizados em duas bases de vídeos amplamente utilizadas na literatura, para avaliar o método proposto: uma contendo vídeos de tráfego de carros [4] e outra contendo vídeos de cenas dinâmicas [11].

Em ambas as bases de vídeos, o método proposto obteve resultados superiores aos obtidos com o BoVW tradicional utilizando o SIFT 3D como descritor local, RI-VLBP e LBP-TOP. Além disso, especificamente na base de cenas dinâmicas, o método proposto também foi comparado com outros métodos da literatura e os resultados mostraram que a proposta apresentada neste trabalho é promissora para a caracterização de texturas dinâmicas.

1.2 Objetivos

O objetivo deste trabalho é a caracterização de texturas e cenas dinâmicas em vídeos utilizando Histograma de Palavras Visuais. Para isso, foi proposto um método em que o BoVW é estendido para descrever vídeos. Na etapa de detecção e descrição de pontos de interesse, um descritor local é aplicado nos três planos ortogonais do vídeo, gerando um vetor de características para cada plano. Por fim, os descritores correspondentes aos planos XY,

[1] *Rotation Invariant Volumetric Local Binary Patterns*
[2] *Local Binary Patterns on Three Orthogonal Planes*

XT e YT, são utilizados pelo BoVW para gerar um vetor de características final, usado para a classificação de vídeos.

Os objetivos do trabalho podem ser resumidos em:

- Investigar e analisar os métodos que envolvem o BoVW para vídeos e métodos para caracterização de texturas dinâmicas;
- Propor um método em que o BoVW é estendido para que seja aplicado sobre os três planos ortogonais do vídeo, XY, XT e YT;
- Aplicar o método proposto em bases de dados de vídeos de textura dinâmica;
- Avaliar o impacto na caracterização de vídeos, combinando diferentes planos ortogonais;
- Comparar os resultados do método proposto com os resultados utilizando o BoVW com o SIFT 3D e duas variações do LBP para vídeos.

1.3 Revisão de Literatura

Esta seção descreve os principais métodos que estenderam o BoVW para vídeos e métodos para caracterização de texturas dinâmicas.

1.3.1 Histograma de Palavras Visuais em Vídeos

Scovanner et al. [33] propuseram um descritor SIFT[3] tridimensional (3D) para vídeos ou imagens 3D. Neste trabalho é mostrado como esse novo descritor é capaz de representar melhor a natureza de dados em vídeos, por meio de aplicações em reconhecimento de ação. O SIFT 3D se assemelha ao 2D, diferenciando-se basicamente na adição do atributo temporal. Para cada ponto de interesse é calculado o ângulo de orientação e magnitude no espaço, ou seja, em qual direção está ocorrendo a variação de intensidade. Similarmente, o descritor proposto calcula também o ângulo de orientação no tempo, garantindo assim que, cada ponto no SIFT 3D, possua sua respectiva informação de orientação no tempo e no espaço. Com a informação adicional, o trabalho mostra que houve 82.6% de acerto com o descritor proposto, contra 30.4% de acerto do SIFT 2D, em uma base de 92 vídeos de pessoas realizando 10 ações diferentes.

Uma extensão do BoVW para classificação de ações humanas (pessoas atendendo o telefone, correndo, andando, aperto de mão, comendo, dirigindo) em vídeos reais é apresentada por Ullah et al. [39]. Neste trabalho, a ideia principal é segmentar um vídeo em regiões significativas e calcular um histograma de palavras visuais para cada região. Em seguida, todos os "histogramas locais" são concatenados em um único histograma, denominado canal. Os experimentos mostraram que essa segmentação ajuda a obter uma melhora significativa, eliminando a ambiguidade dos histogramas de palavras visuais locais. Para isso, foram analisados quatro métodos de segmentação de vídeo e verificou-se que cada um fornece uma determinada informação para as características locais. Com base nisso, é mostrado como essa informação pode ser integrada com o BoVW em um arcabouço (*framework*).

Ballan et al. [1] apresentaram um método para introduzir informação temporal no BoVW para lidar com o problema de classificação de eventos em vídeos. Eventos podem ser definidos como uma sequência composta por histogramas de palavras visuais, calculada a partir de cada quadro do vídeo, utilizando o BoVW tradicional. As sequências são tratadas como frases (*strings*) compostas por uma sequência temporal de histogramas, que são considerados

[3] *Scale Invariant Feature Transform*

caracteres. A classificação desses eventos é realizada através do SVM[4], com um núcleo de *strings* que utiliza a distância de *Needlemann-Wunsch*. Foram realizados experimentos utilizando vídeos de futebol e do laboratório TRECVID 2005 para avaliar o método proposto. Os resultados dos experimentos mostraram que o método proposto neste estudo supera o BoVW tradicional em 7% na taxa de precisão média (do inglês *Mean Average Precision*).

Considerando o contexto espaço-temporal, uma extensão do BoVW foi proposta através da introdução de estratégias de pré-processamento de vídeo com a ajuda de um modelo de retina [37]. Esse pré-processamento ocorre antes da extração dos descritores do BoVW tradicional, aumentando a robustez das características locais com relação a variações de ruído e iluminação. Além disso, a razão pela qual se utiliza o modelo de retina está relacionada com a detecção de áreas potencialmente relevantes e construção de descritores espaço-temporais. Experimentos foram realizados em três métodos de extração de características (SIFT, SURF[5] e FREAK[6]) para avaliar a proposta do trabalho e verificou-se que o modelo de retina também permite que um conjunto de vários descritores distintos e complementares seja projetado, de modo que se obtenha resultados ainda melhores do que os métodos tradicionais.

Um novo modelo para representação de vídeos que incorpora ordem temporal no BoVW foi proposto por Glaser et al. [17]. A Sequência Contextual de Palavras (CSoW) é similar ao *Bag-of-Visual-Words*, diferindo em que além de utilizar um histograma sobre as palavras visuais, também é considerado a ordem em que elas aparecem no decorrer do vídeo. Além disso, o CSoW mantém o contexto temporal de uma sequência de vídeo em três escalas, denominadas fina, média e global. Cada uma dessas escalas lida com possíveis problemas em erros de classificação no BoVW tradicional. Os experimentos mostraram que esta abordagem leva a uma melhoria significativa nas taxas de reconhecimento de ações. Os autores também comentam que o CSoW pode ser aplicado a qualquer sinal cuja ordem temporal e contextual é significativa, como por exemplo, sinais de áudio.

1.3.2 Métodos de Caracterização de Texturas Dinâmicas

Em texturas dinâmicas, a ideia de auto similaridade identificada em imagens de texturas é estendida para o domínio espaço-temporal, ou seja, textura dinâmica, refere-se a uma contínua mudança tanto no movimento quanto na aparência, tornando a sua aplicação um tanto quanto desafiadora.

Poucas abordagens foram propostas na literatura devido aos estudos nesta área de pesquisa ainda serem recentes. Os métodos existentes para caracterização de texturas dinâmicas podem ser agrupados em quatro categorias [18]: (a) baseado em movimento; (b) baseado em filtragem; (c) baseado em modelos e; (d) baseado em propriedades geométricas.

(a) Métodos baseados em movimento: Lu et al. [26] apresentam um método que utiliza histogramas de multiresolução[7] espaço-temporal baseado em campos de velocidade e aceleração. O método proposto é capaz de capturar de forma confiável e representar as características de movimento de sequências de imagens. Chen et al. [6] propuseram um método baseado nas informações de aparência e movimento para segmentação de texturas dinâmicas. Para caracterizar estes padrões, são utilizados descritores de textura espaço-temporal locais e o histograma de fluxo óptico orientado. E por fim, um estudo para investigar a eficiência das características de fluxo óptico na classificação de texturas dinâmicas, foi realizado por Fazekas e Chetverika [13].

[4] *Máquina de Vetores de Suporte* (*Support Vector Machine*)
[5] *Speeded Up Robust Features*
[6] *Fast Retina Keypoint*
[7] Extensão do histograma tradicional combinando informação espacial e de intensidade [26].

(b) Métodos baseados em filtragem: Um método que estende os filtros de Gabor para texturas dinâmicas foi proposto por Gonçalves et al. [20]. Para modelar uma textura dinâmica, uma sequência de imagens é convoluída com um banco de filtros de Gabor espaço-temporal. Para cada resposta da convolução, um vetor de características é construído através da energia. Os resultados de experimentos realizados indicaram que o método proposto é uma abordagem robusta para reconhecimento de texturas dinâmicas. Quatro métodos para caracterização de texturas dinâmicas utilizando a transformada *wavelet* foram apresentados por Dubois et al. [12]. Neste trabalho, o principal objetivo é avaliar a influência das informações de tempo e espaço utilizando os métodos de decomposição *wavelet* e as características extraídas de uma base de vídeos de textura dinâmica.

(c) Métodos baseados em modelos: Ravichandran et al. [32] propuseram um método para categorizar texturas dinâmicas em diferentes escalas e posições utilizando o Histograma de Sistemas - BoS (do inglês *Bag-of-Systems*). Este modelo é análogo ao BoVW para reconhecimento de objetos, diferindo apenas que no método proposto são utilizados os sistemas dinâmicos lineares (do inglês *Linear Dynamical System*) como descritores de características. Um trabalho realizado por Chan e Vasconcelos [5] estuda a mistura de texturas dinâmicas, que é um modelo probabilístico que estende o modelo de textura dinâmica. Enquanto uma textura dinâmica modela uma simples sequência de vídeo de um sistema dinâmico linear, uma mistura de texturas dinâmicas modela uma coleção de sequências de um conjunto de sistemas dinâmicos lineares. Para a aprendizagem dos parâmetros, o modelo foi derivado do algoritmo *expectation-maximization* e um comparativo entre tal método é realizado com os demais do estado da arte. Dois métodos baseados nos padrões locais binários foram propostos por Zhao e Pietikáinen: os padrões locais binários volumétricos invariantes à rotação (do inglês *Rotation Invariant Volumetric Local Binary Patterns - RI-VLBP*) [44] e os padrões locais binários nos três planos ortogonais (do inglês *Local Binary Patterns on Three Orthogonal Planes - LBP-TOP*) [43]. As principais diferenças entre os dois é que o RI-VLBP utiliza três quadros paralelos, de tal forma que apenas o quadro do meio contém o pixel central. Enquanto que o LBP-TOP leva em consideração os três cortes nos planos ortogonais que interceptam o pixel central. A outra diferença está relacionada ao tamanho do vetor de características: o RI-VLBP gera um código para cada pixel utilizando três quadros paralelos, acarretando num vetor grande demais. Já o LBP-TOP gera os descritores separadamente para cada plano ortogonal e depois concatena-os em um único vetor de características.

(d) Métodos baseados em propriedades geométricas: Um método para extrair características baseado no modelo de plano tangente foi proposto por Otsuka et al. [30]. A partir da distribuição de planos tangentes, é possível identificar características temporais e espaciais obtidas através da velocidade transacional dominante e contornos com movimentos dominantes.

1.4 Organização do Texto

Este texto está organizado da seguinte maneira:

Capítulo 2: neste capítulo são descritos os principais conceitos necessários para o entendimento deste trabalho. É abordada a etapa de extração e descrição de pontos de interesse, aprofundando especificamente sobre o SIFT, sua versão densa (SIFT Denso), tridimensional para vídeos (SIFT 3D), SIFT Denso invariante a escala (PHOW), histograma de palavras visuais

(BoVW), padrões locais binários (LBP), sua versão volumétrica (VLBP) e a aplicada em planos ortogonais (LBP-TOP).

Capítulo 3: uma metodologia para caracterizar vídeos utilizando o BoVW com base em três planos ortogonais, é detalhada neste capítulo.

Capítulo 4: apresenta os experimentos e resultados obtidos com o método proposto utilizando os descritores locais SIFT, SIFT Denso e PHOW; o BoVW com o descritor local SIFT 3D; e os resultados obtidos com dois métodos de textura dinâmica.

Capítulo 5: são discutidas as conclusões deste trabalho e apresentados alguns trabalhos futuros.

CAPÍTULO

2

Referencial Teórico

Atualmente, um dos maiores desafios para a área de visão computacional é fazer com que o computador execute tarefas com o desempenho mais próximo possível ao do ser humano. A extração de características é uma etapa realizada em muitas aplicações dessa área, no intuito de encontrar características relevantes em uma determinada imagem. Dessa forma, os descritores locais têm sido bastante utilizados atualmente para detectar características de imagens. Esses métodos possuem duas etapas principais:

- Detecção de pontos de interesse: Envolve a identificação de pontos na imagem onde ocorre variação relevante.
- Descrição dos pontos de interesse: Para cada ponto de interesse, é calculado um vetor de características que descreve a região ao redor daquele ponto.

2.1 Scale-Invariant Feature Transform (SIFT)

O SIFT, do inglês *Scale-Invariant Feature Transform* [25], é um dos métodos mais utilizados atualmente para identificar e descrever pontos de interesse (*keypoints*) em uma imagem. A sua vantagem está relacionada à invariância às transformações de escala e rotação, e parcialmente invariantes às mudanças de iluminação e perspectiva.

Além disso, as características são selecionadas levando em consideração tanto o domínio de frequência quanto o domínio espacial, sendo portanto, robusto à oclusão e ao ruído. Dessa forma, a classificação de uma imagem não é influenciada caso as imagens variem dentre os fatores citados acima. Assim como os demais descritores locais, o SIFT é dividido em dois passos principais, detecção e descrição de pontos de interesse, que são subdivididos em quatro etapas:

1. Construir espaço de escalas.
2. Localizar pontos de interesse.
3. Atribuir orientação para cada ponto de interesse.
4. Descrever os pontos de interesse.

Estas etapas são descritas em detalhes nas subseções abaixo.

2.1.1 Espaço de Escalas

O SIFT considera que os pontos de interesse estão localizados em regiões com variação em intensidade. Para ser invariante à escala, é necessário que estes pontos sejam identificados em várias escalas da mesma imagem. Assim, o espaço de escalas de uma imagem é definido como um conjunto de imagens $L(x,y,\sigma)$, que são obtidas através da convolução de uma Gaussiana $G(x,y,\sigma)$ na escala σ, com a imagem original de entrada, $I(x,y)$:

$$L(x,y,\sigma) = G(x,y,\sigma) \otimes I(x,y) \tag{2.1}$$

onde $G(x,y,\sigma)$ é uma Gaussiana 2D com desvio padrão σ:

$$G(x,y,\sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}} \tag{2.2}$$

Para detectar pontos com mudança de intensidade de modo eficiente no espaço de escalas, o SIFT usa a função DoG (Diferença das Gaussianas) [22], que pode ser calculada pela subtração de duas imagens em escalas separadas por uma constante k:

$$\begin{aligned} D(x,y,\sigma) &= (G(x,y,k\sigma) \otimes I(x,y)) - (G(x,y,\sigma) \otimes I(x,y)) \\ &= L(x,y,k\sigma) - L(x,y,\sigma) \end{aligned} \tag{2.3}$$

De acordo com Lowe [25], a razão para se utilizar esta função se deve ao fato de ser uma função eficiente para calcular as imagens suavizadas (L) e portanto, DoG pode ser calculado através de uma simples subtração de imagens suavizadas. Além disso, de acordo com Crowley et al. [8], o DoG fornece uma aproximação do Laplaciano da Gaussiana (LoG). A Figura 2.1 mostra uma imagem original que foi convoluída em três escalas σ_1, σ_2 e σ_3. Em seguida, duas imagens adjacentes são subtraídas das convoluções obtidas anteriormente. A imagem resultante destaca pontos onde há variação considerada relevante.

A Figura 2.2 mostra uma aproximação eficiente para a construção do espaço de escalas. Dessa forma, a imagem inicial é convoluída com Gaussianas para produzir imagens separadas por uma constante k no espaço de escalas, como mostrado na coluna da esquerda na figura. As imagens Gaussianas adjacentes são subtraídas para gerar as imagens da Diferença da Gaussiana (coluna da direita na figura). A partir desse filtro, é possível detectar variações de intensidade na imagem, como por exemplo, bordas. Esse processo gera as oitavas (*octaves*), que representa um conjunto de imagens L e D em diferentes tamanhos. Todo esse procedimento repete-se para um determinado número de oitavas. Assim, quando uma oitava é processada, a imagem será redimensionada para a metade do seu tamanho, de tal forma que ela seja a entrada para o processamento da próxima oitava. Vale ressaltar que as oitavas possuem um número de intervalos que deve ser previamente definido. Ao final desta etapa, é gerada uma pirâmide com as Diferenças de Gaussianas calculadas com base nas oitavas.

2.1.2 Localizar Pontos de Interesse

Dada a pirâmide DoG, o próximo passo é localizar os pontos de máximo e de mínimo de cada oitava. Assim, cada pixel $D(x,y,\sigma)$ é comparado com seus pixels vizinhos em sua escala, na escala acima e na escala abaixo, sendo portanto, 26 vizinhos no total. Um pixel é determinado como ponto de interesse se ele for maior ou menor do que todos os seus vizinhos. A Figura 2.3 mostra um ponto de interesse candidato em uma determinada oitava. Neste

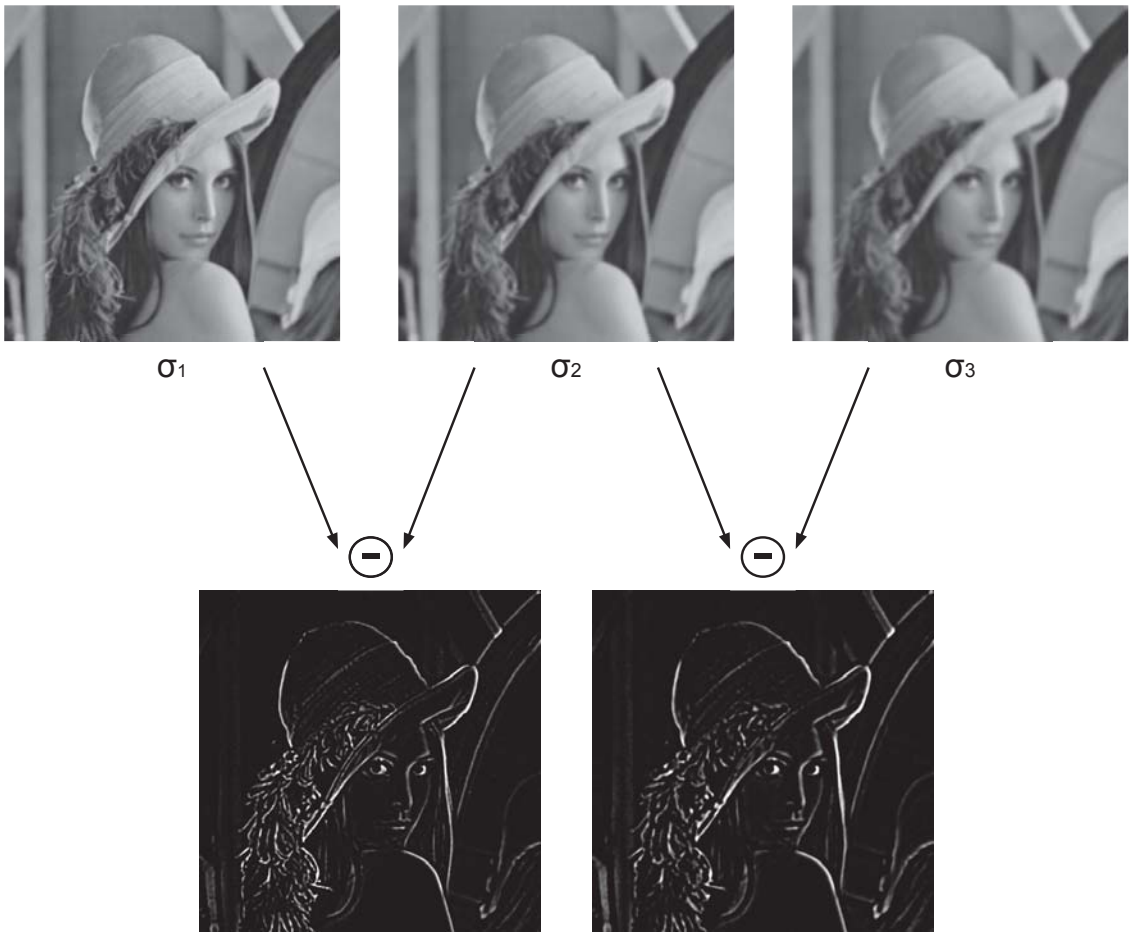


Figura 2.1: Aplicação do filtro Gaussiano seguido da Diferença de Gaussianas entre as imagens convoluídas anteriormente.

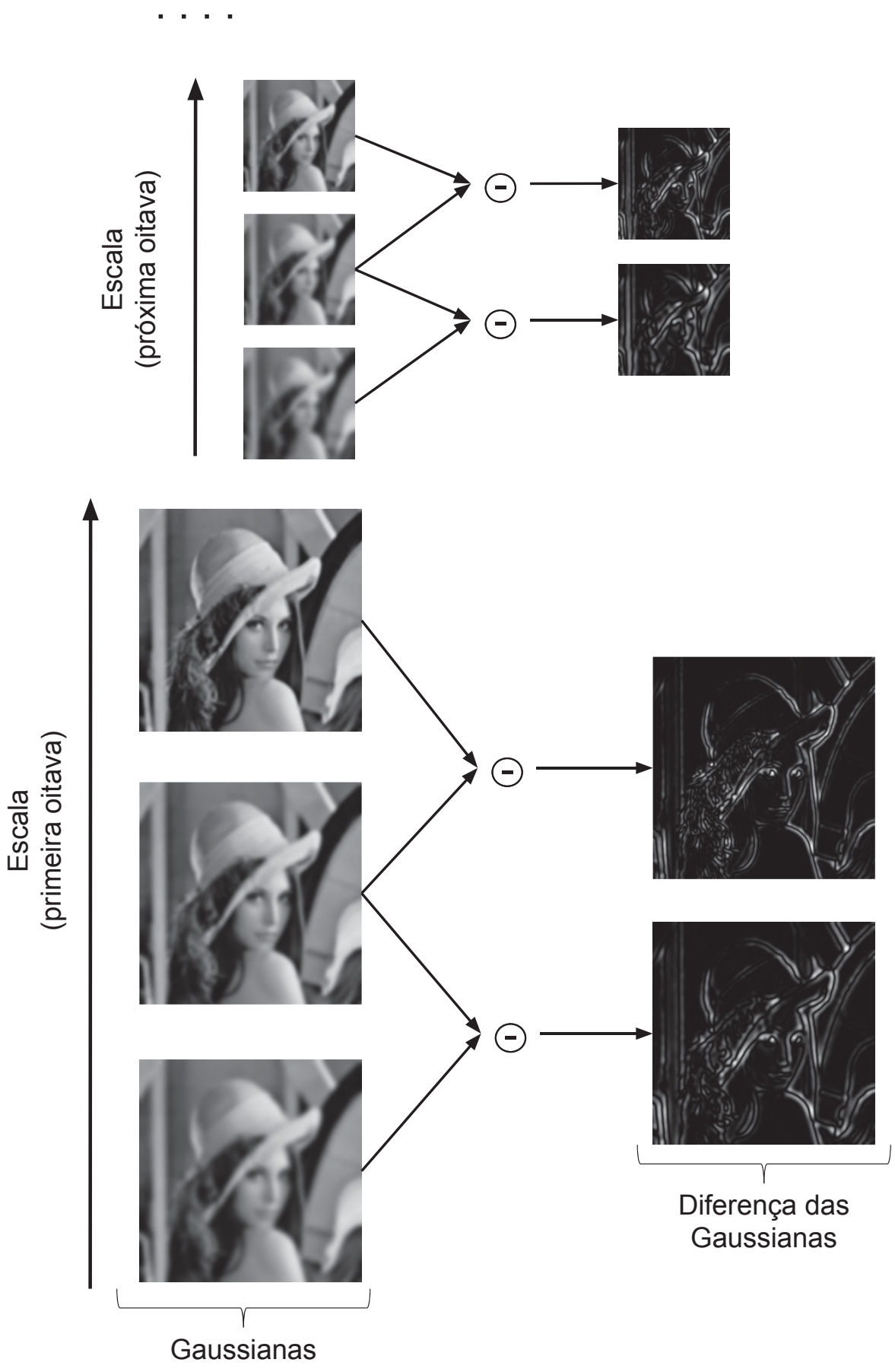


Figura 2.2: Agrupamento das Diferenças das Gaussianas por escala. A cada escala a imagem é redimensionada para a metade do seu tamanho.

caso, cada quadrado corresponde a um pixel e o marcado por um X corresponde ao ponto de interesse candidato e os círculos, correspondem aos seus vizinhos. Se o ponto candidato for maior ou menor que todos os vizinhos, ele é considerado um ponto de interesse, pois a sua variação na intensidade é maior.

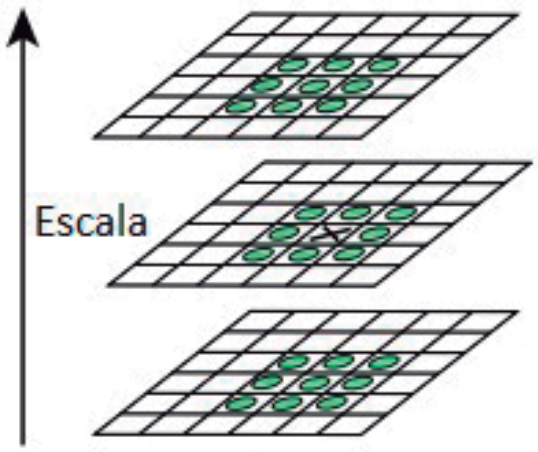


Figura 2.3: Comparação pixel a pixel entre os vizinhos na escala acima e abaixo para a detecção de pontos de interesse. Adaptado de [25].

Em contrapartida, esse procedimento gera muitos pontos de interesse, que por sua vez, muitos deles não são relevantes para um determinado objetivo de classificação. Por exemplo, dada uma imagem de uma paisagem onde metade da imagem seja de um céu azul: serão detectados muitos pontos de interesse (devido às pequenas variações de intensidade) nesta parte da imagem que podem ser irrelevantes. Então, após a detecção dos pontos de interesse, é realizado um ajuste detalhado nos dados da imagem local, determinando a posição, a escala e a proporção de curvatura. Isso permite que pontos detectados sejam rejeitados, por serem sensíveis ao ruído ou por estarem localizados ao longo das bordas. Para identificar pontos com baixo contraste, realiza-se um cálculo baseado na expansão de Taylor da função $D(x,y,\sigma)$ para cada ponto. Lowe [25] recomenda que se rejeite valores cujo resultado deste cálculo seja inferior a um determinado limiar e, aconselha que se trabalhe com o valor 0.03 para este limiar (assumindo que os valores estejam entre 0 e 1).

Além disso, Lowe [25] também afirma que a função da Diferença das Gaussianas gera pontos de resposta forte em uma aresta em única direção, fazendo com que esses pontos sejam instáveis para ruído. Assim, é proposto um procedimento para rejeição de pontos, baseado na matriz Hessiana, formada pelas derivadas parciais:

$$H = \begin{pmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{pmatrix}. \tag{2.4}$$

Com base nessa matriz, calcula-se a soma dos autovalores por meio do traço de H e o produto pelo determinante. O determinante dessa matriz corresponderá ao quanto a função está variando naquela região. Com estes valores calculados, para conferir a razão entre as curvaturas basta checar se:

$$\frac{Tr(H)^2}{Det(H)} < \frac{(r+1)^2}{r}. \tag{2.5}$$

Se a condição é atendida, os pontos de interesse em questão são rejeitados. Após diversos experimentos, Lowe [25] propõe o uso de $r = 10$. Ao final desta etapa, é obtido um conjunto de

Figura 2.4: Detecção de pontos onde há variação de intensidade (em amarelo). Cada ponto possui uma orientação e escala em que foi identificada variação de intensidade.

pontos de interesse (x,y) e suas escalas correspondentes σ. A Figura 2.4 ilustra uma imagem de tal forma que seus respectivos pontos de interesse (em amarelo) foram detectados pelo SIFT.

2.1.3 Atribuir Orientação aos Pontos de Interesse

O objetivo desta etapa é tornar um ponto invariante à rotação. Para cada ponto de interesse detectado na etapa anterior, calcula-se a magnitude $m(x,y,\sigma)$ e a orientação $\theta(x,y,\sigma)$ de seus pixels vizinhos na imagem em escala σ que o ponto foi detectado. As equações seguem abaixo:

$$m(x,y,\sigma) = \sqrt{(L(x+1,y,\sigma) - L(x-1,y,\sigma))^2 + (L(x,y+1,\sigma) - L(x,y-1,\sigma))^2}. \tag{2.6}$$

$$\theta(x,y,\sigma) = \tan^{-1}\left(\frac{L(x,y+1,\sigma) - L(x,y-1,\sigma)}{L(x+1,y,\sigma) - L(x-1,y,\sigma)}\right). \tag{2.7}$$

A Equação (2.6) calcula a força da variação no ponto (x,y) e a Equação (2.7) indica a orientação que essa variação ocorre. O próximo passo é construir um histograma de orientação a partir das magnitudes dos pixels ao redor do ponto de interesse. Esse histograma tem o intuito de identificar qual a orientação dominante ao redor do ponto de interesse. O histograma é uma função com um determinado número de valores discretos de θ, que Lowe [25] sugere que sejam 36, para que os 360^o de orientações possam ser alcançados. Os picos no histograma correspondem à orientação dominante para aquela região. O maior pico no histograma e aqueles com o valor acima de 80% do maior pico, são utilizados para definir a orientação do ponto de interesse.

2.1.4 Descrição dos Pontos de Interesse

As três etapas anteriores detectaram os pontos de interesse e as suas respectivas orientações. Assim, cada ponto de interesse é representado pela sua posição espacial (x,y), a escala σ e a orientação dominante θ. Nesta etapa, cada ponto de interesse é descrito com base na sua região. A descrição de cada ponto de interesse é dividida nos seguintes passos:

1. Definir uma região de 16×16 ao redor do ponto de interesse considerando a sua orientação e escala conforme a Figura 2.5a.

2. Dividir a região formada em grades 4×4 (Figura 2.5b).

3. Criar um histograma com 8 direções conforme a Figura 2.5c para cada grade. O histograma é construído com base nas magnitudes dos gradientes dos pixels de cada região 4×4, dividida no passo anterior e semelhante à etapa de atribuição de orientação dos pontos de interesse.

4. O descritor é representado pela concatenação dos histogramas de cada grade que resultará em um vetor de características, onde cada posição do vetor refere-se a cada uma das direções do histograma de cada grade. Neste caso, são 16 histogramas, onde cada um possui 8 direções, ou seja, o vetor terá tamanho 128. Portanto, para cada ponto de interesse, são extraídas 128 características.

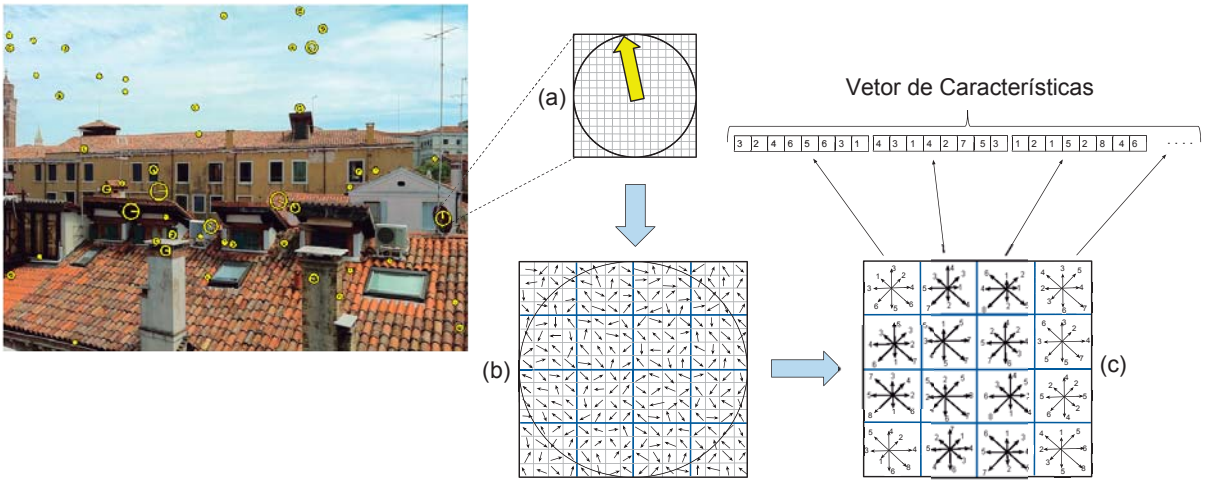


Figura 2.5: Visão geral do processo de criação do vetor de características de um ponto de interesse da imagem. Para cada ponto de interesse identificado, é definida uma região 16×16 ao seu redor, considerando sua orientação e escala (a); Em seguida (b), essa região é dividida em grades 4×4, considerando as orientações da vizinhança na escala que o ponto foi detectado; Em (c) será calculado para cada grade 4×4 um histograma de 8 direções levando em consideração as magnitudes dos gradientes da vizinhança; E por fim, os histogramas obtidos em cada grade serão concatenados, formando o vetor de características daquele ponto de interesse.

2.2 Scale-Invariant Feature Transform 3D (SIFT 3D)

Esta subseção descreve as principais diferenças entre o SIFT 2D aplicado em imagens e o SIFT 3D [33] aplicado em vídeos. O SIFT 3D calcula a orientação e magnitude em uma vizinhança 3D e, por fim, a codifica em sub-histogramas que compõem o descritor 3D. A orientação e magnitude do gradiente em um determinado pixel (x,y) em imagens 2D pode ser calculado conforme as Equações 2.6 e 2.7. De forma similar, em 3D (x, y e t), o gradiente espaço-temporal é calculado por meio da diferença entre quadros do vídeo: $L(x,y,t+1,\sigma) - L(x,y,t-1,\sigma)$. Portanto, a orientação e magnitude em 3D são dadas por:

$$m_{3D}(x,y,t,\sigma) = \begin{array}{c} \sqrt{(L(x+1,y,t,\sigma) - L(x-1,y,t,\sigma))^2 + } \\ \overline{(L(x,y+1,t,\sigma) - L(x,y-1,t,\sigma))^2 + } \\ \overline{(L(x,y,t+1,\sigma) - L(x,y,t-1,\sigma))^2} \end{array} \tag{2.8}$$

$$\theta(x,y,t,\sigma) = \tan^{-1}\left(\frac{L(x,y+1,t,\sigma) - L(x,y-1,t,\sigma)}{L(x+1,y,t,\sigma) - L(x-1,y,t,\sigma)}\right) \tag{2.9}$$

$$\phi(x,y,t,\sigma) = \tan^{-1}\left(\frac{L(x,y,t+1,\sigma) - L(x,y,t-1,\sigma)}{\sqrt{(L(x+1,y,t,\sigma) - L(x-1,y,t,\sigma))^2 + (L(x,y+1,t,\sigma) - L(x,y-1,t,\sigma))^2}}\right). \quad (2.10)$$

É possível observar que ϕ codifica o ângulo espaço-temporal e, devido ao fato do denominador da Equação 2.10 ser sempre positivo, ϕ varia entre $(-\frac{\pi}{2}, \frac{\pi}{2})$. Dessa forma, a orientação para vídeos é representada por um par (θ, ϕ). O próximo passo é construir um histograma similar ao do SIFT 2D através da divisão de θ e ϕ em faixas (meridianos e paralelos), formando um histograma 2D. Os picos deste histograma representam as orientações dominantes que são utilizadas para rotacionar a vizinhança 3D. Deve-se notar que a rotação neste caso torna-se uma rotação 3D devido ao par de orientações dominantes.

A Figura 2.6 ilustra o processo para obtenção do descritor do SIFT 3D de um determinado ponto de interesse. O descritor, é calculado com base na vizinhança 3D ao redor do ponto de interesse. Essa vizinhança é dividida em $2 \times 2 \times 2$ ou $4 \times 4 \times 4$ sub-regiões, em que cada pixel possui uma magnitude e duas orientações (uma no tempo e outra no espaço). Na Figura 2.6a é apresentada a vizinhança 3D (sub-regiões), citada anteriormente e em amarelo a representação de uma sub-região composta por vários pixels. As setas em vermelho representam uma das 4 orientações no tempo e as setas na cor preta, representam uma das 8 orientações no espaço. Para cada sub-região, é calculado um histograma 2D acumulando as magnitudes dos gradientes (Figura 2.6b). Por fim, os histogramas das sub-regiões são concatenados em um único vetor que descreve a vizinhança ao redor do ponto de interesse (Figura 2.6c).

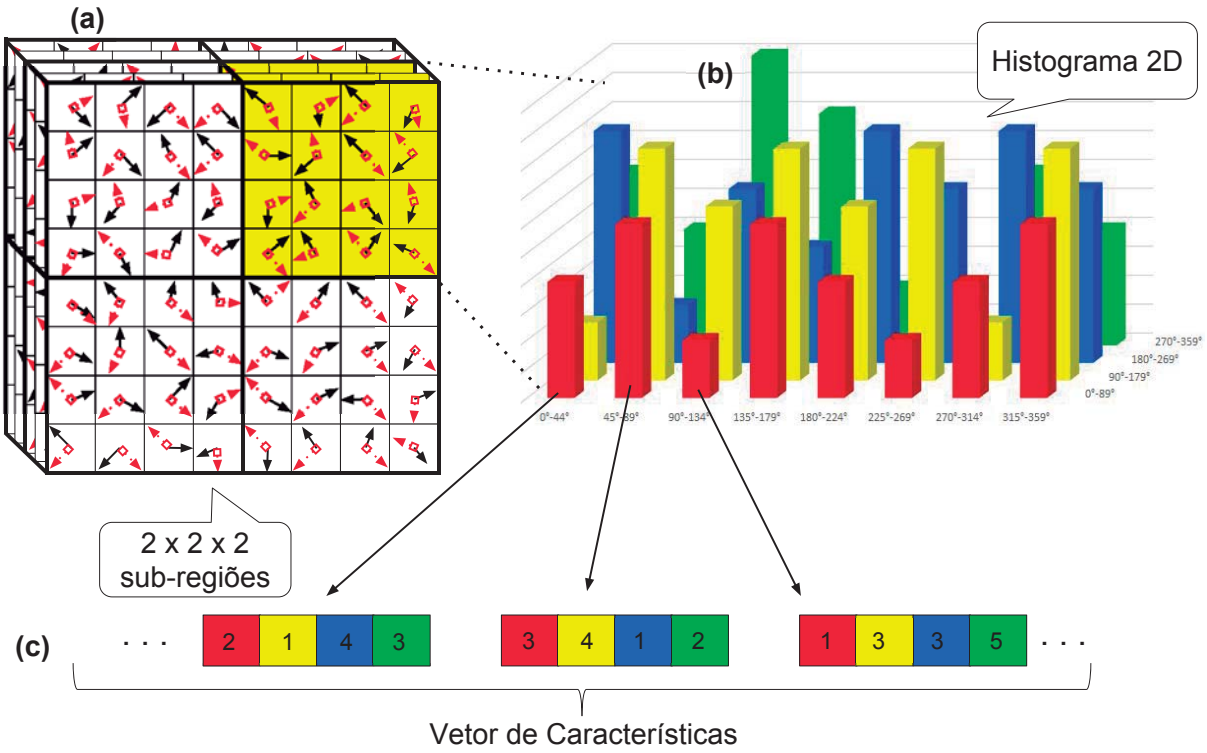


Figura 2.6: Visão geral do processo de criação do descritor de um determinado ponto do vídeo. Inicialmente é definida a vizinhança 3D (sub-regiões) ao redor do ponto de interesse (a). Cada sub-região é composta por uma grade de pixels, onde cada um possui uma magnitude e duas orientações dominantes: uma no tempo e outra no espaço. Em seguida, de cada sub-região é construído um histograma bidimensional, para calcular a frequência de cada orientação no espaço para cada uma das orientações no tempo (b). E por fim, o vetor de características (c) é formado a partir da concatenação dos histogramas 2D de cada sub-região ao redor do ponto de interesse. Neste caso, para cada ponto de interesse, um vetor de 256 características será obtido.

2.3 Dense Scale-Invariant Feature Transform (SIFT Denso)

Nesta subseção são apresentadas as principais diferenças ente o SIFT e a sua versão densa (SIFT Denso), ambos aplicados em imagens 2D. O SIFT Denso [24] implementa a descrição de pontos de forma densa, ou seja, calcula a orientação e magnitude de cada pixel (x,y) na imagem e codifica em sub-histogramas de 128 valores que compõem o seu descritor. O SIFT Denso obtém descritores para todos os pixels da imagem e por isso, a etapa de detecção de pontos de interesse (como é executada pelo SIFT tradicional) não é realizada.

No SIFT tradicional, há duas etapas principais: detecção de pontos interesse e descrição de pontos de interesse. A etapa de detecção é subdividida em 3 fases: a primeira envolve construir um espaço com a imagem em diferentes escalas. Depois, com as imagens Gaussianas adjacentes, são calculadas as diferenças das Gaussianas para localizar os pontos de interesse onde há variação de intensidade (segunda fase). E por fim, a terceira fase consiste em calcular a magnitude e orientação dos pixels vizinhos para que se obtenha a orientação dominante ao redor de cada ponto de interesse.

A etapa de detecção de pontos de interesse é responsável por tornar o SIFT tradicional invariante à escala e rotação, justificando o fato do SIFT Denso não ser invariante a estes dois atributos. Portanto, como o SIFT Denso não lida com escalas diferentes de uma imagem I, as magnitudes e orientações de cada pixel $I(x,y)$ podem ser obtidas como mostram as Equações 2.11 e 2.12:

$$m(x,y) = \sqrt{(I(x+1,y) - I(x-1,y))^2 + (I(x,y+1) - I(x,y-1))^2}. \tag{2.11}$$

$$\theta(x,y) = \tan^{-1}\left(\frac{I(x,y+1) - I(x,y-1)}{I(x+1,y) - I(x-1,y)}\right). \tag{2.12}$$

A descrição dos pontos no SIFT Denso é similar ao tradicional, diferindo que o descritor é calculado para todos os pixels da imagem. Dessa forma, cada ponto é representado apenas pela sua posição espacial (x,y) e descrito com base na sua vizinhança. Esta etapa é subdividida em 4 passos: definir uma região de 16×16 ao redor do ponto considerando as orientações e magnitudes de seus pixels vizinhos; dividir a região formada em uma grade 4×4; criar um histograma de 8 direções baseado nos pixels de cada grade e; concatenar os histogramas de cada grade.

Assim como o SIFT tradicional, na sua versão densa, o descritor de cada ponto é representado por um vetor de características, onde cada posição do vetor refere-se a cada uma das direções do histograma de cada grade. De cada ponto são calculados 16 histogramas, com 8 direções cada um, totalizando 128 posições no vetor de características.

A principal diferença entre o SIFT e a versão densa é que o SIFT Denso não é invariante à escala nem rotação, mas em contrapartida, gera mais descritores do que o SIFT tradicional. Dessa forma, essa não-invariância à escala e rotação, pode não ser tão relevante em determinados problemas de caracterização de imagens, tais como aplicações em que as imagens possuem poucas variações de escala. Além disso, o fato do SIFT Denso gerar descritores para todos os pontos da imagem implica no ganho de informação para caracterização, em comparação ao SIFT tradicional que seleciona as regiões da imagem a serem descritas.

2.4 Histograma Piramidal de Palavras Visuais (PHOW)

O Histograma Piramidal de Palavras Visuais, do inglês *Pyramidal Histogram of Visual Words* (PHOW) [3], é uma abordagem para descrever imagens que incrementa o SIFT Denso [24]

descrito na Seção 2.3. Apesar do SIFT Denso ser considerado uma das melhores técnicas para descrever imagens [14], ele possui algumas desvantagens como a não-invariância à rotação e escala. Para isso, o PHOW foi proposto para lidar com parte desta desvantagem: tornar o SIFT Denso invariante à escala.

O PHOW consiste na aplicação do SIFT Denso em 4 escalas diferentes da imagem. Para isso, as magnitudes e orientações dos pixels de uma imagem I também são calculadas em escalas diferentes, como mostram as Equações 2.13 e 2.14:

$$m(x,y,\sigma) = \sqrt{(I(x+1,y,\sigma) - I(x-1,y,\sigma))^2 + (I(x,y+1,\sigma) - I(x,y-1,\sigma))^2}. \tag{2.13}$$

$$\theta(x,y,\sigma) = \tan^{-1}\left(\frac{I(x,y+1,\sigma) - I(x,y-1,\sigma)}{I(x+1,y,\sigma) - I(x-1,y,\sigma)}\right). \tag{2.14}$$

Em seguida, são calculados os descritores dos pixels de cada imagem na escala σ. O descritor de cada ponto, representado pela sua posição espacial (x,y) e sua escala σ, também é calculado como no SIFT: com base em sua vizinhança. Dessa forma, para cada ponto da imagem é definida uma região 16×16 que levará em consideração as orientações e magnitudes da vizinhança na escala σ, calculadas nas Equações 2.13 e 2.14. Em seguida, a partir dessa região, são formadas grades 4×4 e, para cada região, é criado um histograma de 8 direções, baseados nos pixels de cada grade. Por fim, esses histogramas são concatenados, formando um vetor de 128 características para o ponto (x,y) na escala σ.

Ao final, o descritor PHOW será formado através dos 4 vetores de características dos pontos da imagem obtidos em escalas distintas, cujo seu tamanho será o quádruplo do tamanho do descritor gerado pelo SIFT Denso.

2.5 Histograma de Palavras Visuais

Histograma de Palavras Visuais, do inglês *Bag-of-Visual-Words*, é uma técnica que tem por finalidade criar um vocabulário de palavras visuais que caracterizem um conjunto de imagens. O Histograma de Palavras Visuais é baseado no Histograma de Palavras (do inglês *Bag-of-Words* - BoW), onde neste último, a principal tarefa é determinar características relevantes em um texto [40, 9]. Dado um determinado texto, ele é representado por um histograma com a frequência das palavras-chave que o caracterizam. Similarmente aos termos ou palavras-chave de um texto, uma imagem possui os pontos de interesse locais ou pontos de interesse, que formam pequenas regiões contendo alguma informação rica capaz de caracterizá-la [42].

Para a imagem, a principal tarefa é definir o que é relevante para caracterizá-la. Neste âmbito que surgem as denominadas Palavras Visuais, que significam à grosso modo, regiões de uma imagem que a caracterizam. Por exemplo, cor, variação, cantos, bordas são atributos que servem para caracterizar uma imagem. Em geral, uma imagem possui um conjunto de pontos de interesse que são considerados relevantes para sua caracterização. Para definir um vocabulário de palavras visuais, é necessário identificar pontos de interesse em várias imagens e determinar a partir destes pontos, regiões que sejam capazes de caracterizá-las. Esse procedimento é semelhante para gerar o vocabulário de palavras de texto, diferindo em que neste caso, trata-se de imagem e não texto. O processo de criação do histograma de palavras visuais é dividido nas seguintes etapas:

1. Detecção dos pontos de interesse nas imagens.
2. Criação do vocabulário de palavras visuais.
3. Rotulação das palavras visuais.

4. Construção do histograma das palavras.

Estas etapas são descritas nas subseções seguintes deste capítulo.

2.5.1 Detecção de Pontos de Interesse

Esta etapa consiste na identificação dos pontos de interesse (que são considerados relevantes para caracterizar uma imagem) seguida pela definição de um vetor de características (descritores) para cada ponto. Os descritores preferencialmente devem ser invariantes às transformações de escala, rotação, iluminação, perspectiva e oclusão, e ricos o suficiente para gerar informação discriminatória para a classificação de uma imagem [9]. O SIFT [25] é um dos métodos mais utilizados na literatura para identificar e descrever pontos de interesse de uma imagem, pois sua vantagem está relacionada às invariações de imagem já mencionadas anteriormente.

Ao final desta etapa, obtém-se um vetor de características (descritor) para cada ponto de interesse detectado. Geralmente, esses vetores possuem um tamanho fixo e não precisam necessariamente representar partes da imagem. Portanto, dada uma imagem I, um conjunto de descritores é obtido: $D_I = [d_0^I, d_1^I, d_i^I, ..., d_n^I]$, onde n é o número de pontos de interesse detectados.

2.5.2 Vocabulário de Palavras Visuais

Com base nos pontos de interesse já identificados e descritos, nesta etapa é criado o vocabulário de palavras visuais. Em geral, nos trabalhos relacionados da área, é aplicado o algoritmo de agrupamento (*clustering*) K-Médias para criar esse vocabulário [36]. O algoritmo K-Médias, do inglês *K-Means*, consiste em gerar K grupos, onde cada grupo contém um representante, denominado centroide. Partindo de um chute inicial de centroides, cada dado é atribuído a um grupo, de acordo com a sua distância ao centroide mais próximo. Esse processo é realizado iterativamente, ou seja, a cada iteração os centroides são recalculados e, consequentemente, é gerado um novo agrupamento. Isso se repete até que os valores dos centroides não se alterem mais ou um número máximo de iterações seja alcançado [34]. No BoVW, cada centroide representa uma palavra visual no vocabulário de palavras visuais. Assim, o tamanho do vocabulário de palavras visuais fica restrito ao valor de K, parâmetro do K-Médias, que representa a quantidade de grupos a serem gerados ao final da execução do algoritmo.

A Figura 2.7 ilustra o processo de criação de um vocabulário de palavras visuais contendo três palavras visuais. Inicialmente, os descritores são extraídos de cada imagem do conjunto de treinamento. Dessa forma, os conjuntos de descritores $D_{I_1}, D_{I_2}, ..., D_{I_n}$ de cada imagem são reunidos em um único conjunto $D = [D_{I_1}, D_{I_2}, ..., D_{I_n}]$ para que em seguida seja aplicado o algoritmo K-Médias com $K = 3$:

$$C = \text{K-Médias}(D) \tag{2.15}$$

O algoritmo resulta na criação de três grupos distintos, cada um com seu respectivo centroide. Os três centroides gerados compõem o vocabulário de palavras visuais.

Uma das desvantagens do K-Médias está justamente relacionada ao parâmetro K: um valor para K grande ou pequeno demais pode gerar um agrupamento de baixa qualidade. Além disso, um determinado ponto com valores altos demais pode causar uma grande alteração no centro de gravidade dos centroides, podendo ocasionar também um agrupamento de baixa qualidade.

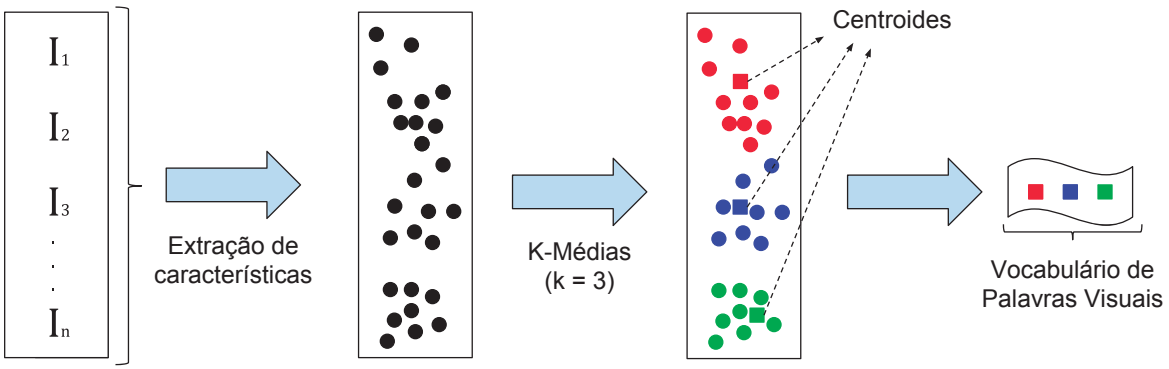


Figura 2.7: Processo de criação das palavras visuais.

2.5.3 Rotulação

Dada uma imagem I, é realizado o procedimento para detecção de seus pontos de interesse e, cada ponto de interesse é associado a uma palavra visual, com base no vocabulário das palavras visuais C, criado na etapa anterior. O processo de atribuição dos descritores dos pontos de interesse da imagem a um grupo consiste em realizar o cálculo da distância Euclidiana entre cada palavra do vocabulário de palavras visuais e cada descritor da imagem. Assim, cada ponto de interesse é rotulado a uma palavra visual cuja distância Euclidiana entre ela e o descritor obtiver o menor valor:

$$w_i^I = \arg\min_{j=1}^{K} \|d_i^I, C_j\|, \tag{2.16}$$

onde w_i^I é o índice da palavra visual do descritor i da imagem I.

2.5.4 Histograma

Por fim, a última etapa consiste em gerar um histograma h_I das palavras visuais da imagem cujos pontos de interesse foram rotulados na etapa anterior. O histograma possui o tamanho do vocabulário de palavras visuais e cada posição representa a frequência em que cada palavra visual ocorre na imagem, de acordo com a Equação 2.17. O cálculo da frequência consiste em somar a quantidade de descritores que foram associados a cada palavra visual. Ao final deste processo, tem-se um histograma com K frequências de cada palavra visual do vocabulário na imagem.

Assim, o histograma gerado possui atributos relevantes que são utilizados na etapa de classificação das imagens, ou seja, com base no histograma, é possível classificar uma determinada imagem, de acordo com seus respectivos pontos de interesse.

$$h_I(j) = \sum_{i=1}^{n} \begin{cases} 1, & \text{se } w_i^I = C_j \\ 0, & \text{caso contrário} \end{cases}, 1 \leq j \leq K \tag{2.17}$$

2.6 Padrões Locais Binários

Nesta subseção são descritos dois métodos baseados nos padrões locais binários, apontando as principais diferenças entre eles, para a caracterização de texturas dinâmicas. Os padrões locais binários, do inglês *Local Binary Patterns* (LBP) proposto por Ojala et. al [27], consiste em calcular um código para cada pixel na imagem em tons de cinza, obtendo ao final um histograma com as frequências dos códigos. O código de cada pixel é calculado com

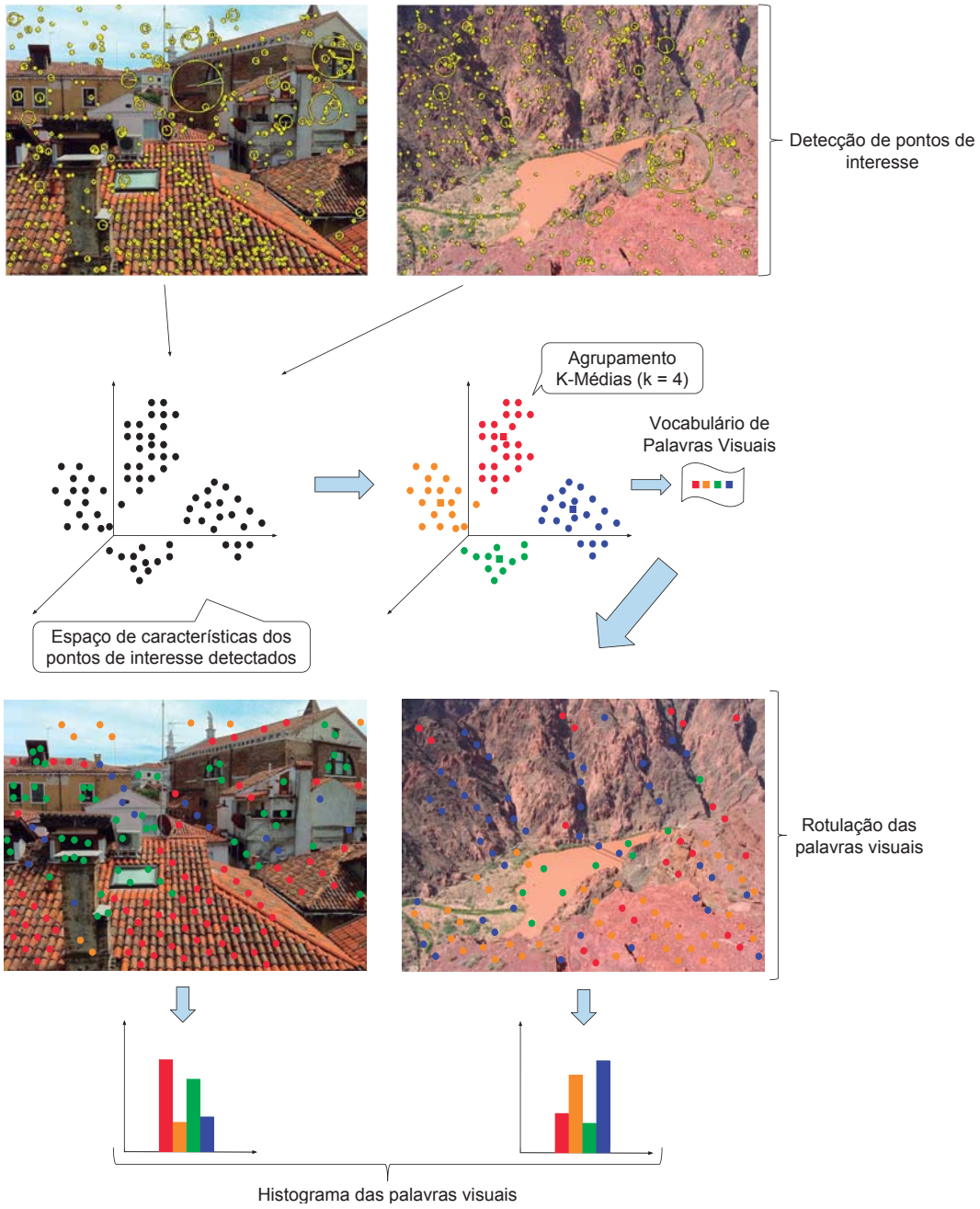


Figura 2.8: Visão geral das etapas do Histograma de Palavras Visuais (*Bag-of-Visual-Words*).

base na sua respectiva vizinhança (pixels ao redor), onde o número de vizinhos pode ser ajustado para cada aplicação ou problema. Cada pixel na imagem é representado pelo conjunto $V = v(g_c, g_0, g_1, \ldots, g_{P-1})$, onde g_c representa o valor do pixel central em níveis de cinza e g_p, $0 \leq p \leq P-1$, representa os valores em níveis de cinza da vizinhança que estão em um raio de distância R de g_c.

Para tornar invariante à iluminação, o valor de cada pixel g_p da vizinhança é subtraído por g_c. A invariância é atingida considerando apenas sinais s ao invés dos valores exatos das diferenças e o código em binário é formado a partir da concatenação dos sinais da vizinhança do pixel central. A conversão deste binário para o seu respectivo valor em decimal consiste no somatório da multiplicação de s de cada vizinho p pela sua respectiva potência de 2, como

é mostrado pela Equação 2.18:

$$LBP_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_c) \times 2^p, \text{ onde } s(x) = \begin{cases} 1, & \text{se } x \geq 0 \\ 0, & \text{se } x < 0 \end{cases}. \tag{2.18}$$

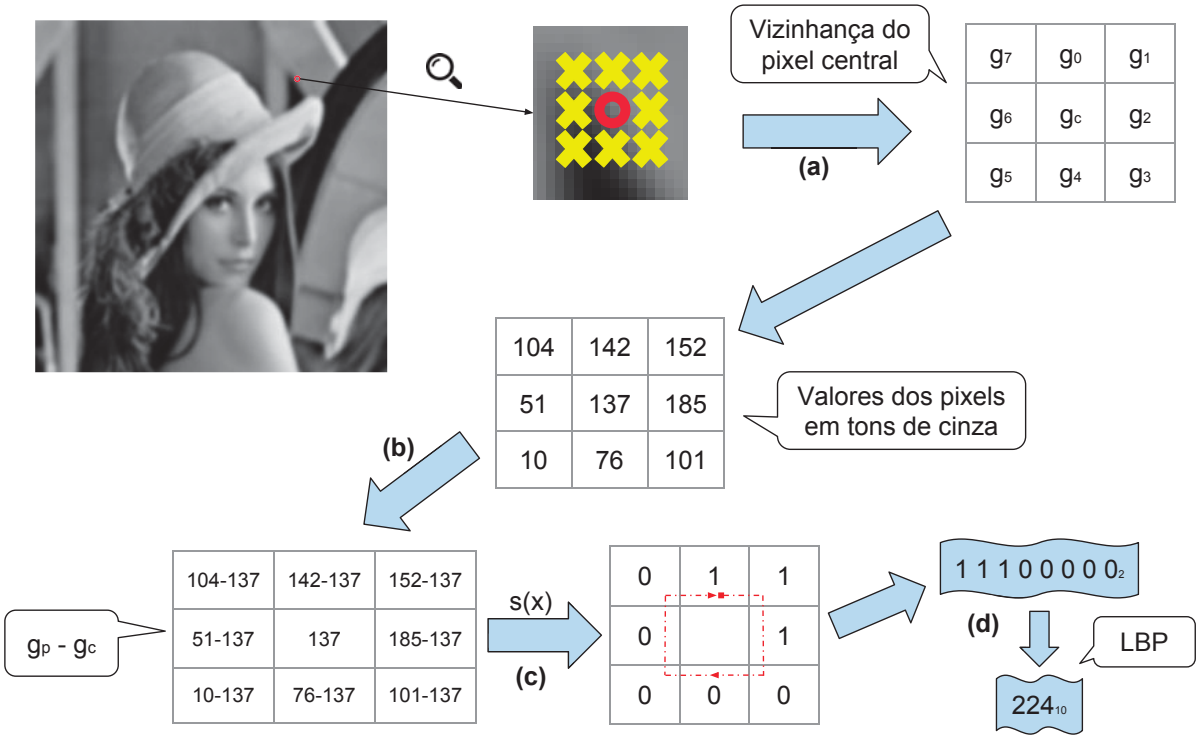


Figura 2.9: Visão geral do LBP. Em cada pixel da imagem são identificados os pixels ao redor que corresponde à vizinhança (a). Em seguida, os valores em tons de cinza de cada pixel vizinho são subtraídos do valor em tons de cinza do pixel central (b). O próximo passo (c) é calcular um sinal para cada pixel vizinho, com base no valor da subtração anterior: o sinal será 0 para aquele vizinho se o valor da subtração resultou em um número negativo; ou 1, caso contrário. Por fim, através da concatenação dos sinais, é obtido um código binário que é convertido para a base decimal (d). O valor final obtido é utilizado para calcular um histograma de frequências dos valores em decimais de cada pixel da imagem.

A Figura 2.9 ilustra o processo para obtenção do valor em decimal de um determinado pixel da imagem, com $P = 8$ e $R = 1.0$, que será utilizado na próxima etapa, para a construção de um histograma. O histograma é construído com as frequências de todos os valores em decimal, obtidos de cada pixel da imagem. Cada pixel possui P vizinhos e, portanto, o histograma possui 2^P posições.

2.6.1 Padrões Locais Binários Volumétricos Invariantes à Rotação

Os Padrões Locais Binários Volumétricos Invariantes à Rotação, do inglês *Rotation Invariant Volumetric Local Binary Patterns* (RI-VLBP) [44], são uma complementação dos tradicionais Padrões Locais Binários Volumétricos (VLBP). Ambas as técnicas possuem sua aplicação em texturas dinâmicas, ou seja, em vídeos onde há um padrão na aparência e no movimento. Um vídeo pode ser representado através um cubo tridimensional formado pelo seu conjunto de imagens sequenciadas em níveis de cinza. Os eixos X e Y do cubo representam a informação espacial e o eixo T representa os quadros ao longo do vídeo. Portanto, cada pixel neste cubo é representado através da combinação das coordenadas de cada eixo $p = \{x, y, t \mid x \in X, y \in Y, t \in T\}$.

A aplicação do VLBP para caracterização de um vídeo é similar ao tradicional LBP diferindo em que neste, a técnica lida com um volume em representação ao vídeo, ao invés de uma simples imagem. Portanto, para cada pixel do volume, o VLBP extrai um código que o represente. A aplicação do VLBP requer além do número de vizinhos P e o raio de distância R, mais um parâmetro L. Como estamos lidando com um conjunto de imagens sequenciadas, o parâmetro L é responsável por indicar o intervalo do quadro anterior (Figura 2.10a) e posterior (Figura 2.10c) em relação ao quadro do pixel central (Figura 2.10b). O parâmetro P está relacionado com a quantidade de pixels vizinhos em relação ao pixel central. O parâmetro R indica o raio de distância de cada vizinho em relação ao pixel central.

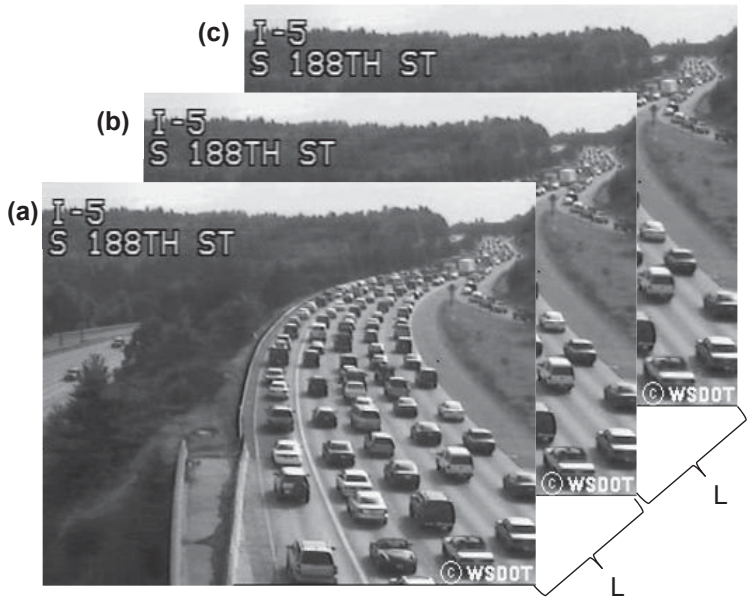


Figura 2.10: Intervalo de L quadros anteriores e posteriores em relação ao quadro que contém o pixel central (b).

O quadro posterior e anterior cuja distância é L em relação ao quadro do pixel central, também possuem P pixels vizinhos, além de um pixel central em cada um, com distância R de seus vizinhos. Portanto para cada pixel do vídeo será calculado um código em binário de tamanho $3P+2$, que será convertido para o seu respectivo valor na base decimal. Similarmente ao LBP tradicional, com base nos valores decimais obtidos de cada pixel do vídeo, será gerado um histograma com as frequências de cada valor em decimal, de tamanho 2^{3P+2}.

Cada pixel p do volume que terá seu código binário calculado, será o pixel central de sua região local. O seu valor em níveis de cinza é representado por $g_{t_c,c}$ e os valores de sua vizinhança no mesmo quadro são representados por $g_{t_c,0}$, $g_{t_c,1}$, $\ldots$, $g_{t_c,P-1}$. Considerando o intervalo de L quadros, $g_{t_c-L,c}$ e $g_{t_c+L,c}$ correspondem aos valores do pixel central do quadro anterior e posterior em relação ao quadro que contém o valor $g_{t_c,c}$. Os valores $g_{t,p}$, com $t \in \{t_c-L,\ t_c,\ t_c+L\}$ e $p \in \{0,\ 1\ ,\ \ldots\ ,\ P-1\}$, são subtraídos do valor $g_{t_c,c}$ do pixel central, como é apresentado pela Equação 2.19:

$$\begin{aligned} V_1 = v\big(&g_{t_c-L,c}-g_{t_c,c},\ g_{t_c-L,0}-g_{t_c,c},\ g_{t_c-L,1}-g_{t_c,c},\ \cdots\ ,\ g_{t_c-L,P-1}-g_{t_c,c},\\ &g_{t_c,0}-g_{t_c,c},\ g_{t_c,1}-g_{t_c,c},\ \cdots\ ,\ g_{t_c,P-1}-g_{t_c,c},\\ &g_{t_c+L,0}-g_{t_c,c},\ g_{t_c+L,1}-g_{t_c,c},\ \cdots,\ g_{t_c+L,P-1}-g_{t_c,c},\ g_{t_c+L,c}-g_{t_c,c}\big). \end{aligned} \tag{2.19}$$

Em seguida, para que de forma similar ao LBP original o descritor se torne invariante à escala de cinza, os valores $g_{t,p}$ são ajustados desconsiderando o valor exato da subtração por $g_{t_c,c}$. O ajuste consiste em redefinir o valor da vizinhança local do pixel central para 0 ou 1,

como é mostrado na Equação 2.20:

$$V_2 = \begin{array}{c} v([s(g_{t_c-L,c}-g_{t_c,c})],\ [s(g_{t_c-L,0}-g_{t_c,c})],\ [s(g_{t_c-L,1}-g_{t_c,c})],\ \dots\ ,\ [s(g_{t_c-L,P-1}-g_{t_c,c})], \\ [s(g_{t_c,0}-g_{t_c,c})],\ [s(g_{t_c,1}-g_{t_c,c})],\ \dots\ ,\ [s(g_{t_c,P-1}-g_{t_c,c})], \\ [s(g_{t_c+L,0}-g_{t_c,c})],\ [s(g_{t_c+L,1}-g_{t_c,c})],\ \dots,\ [s(g_{t_c+L,P-1}-g_{t_c,c})],\ [s(g_{t_c+L,c}-g_{t_c,c})]), \\ \text{onde } s(x) = \begin{cases} 1, & \text{se } x \geq 0 \\ 0, & \text{se } x < 0 \end{cases}. \end{array} \tag{2.20}$$

A partir disso, similarmente ao LBP original, V_2 também é tomado como um vetor de tamanho $3P+2$, $v(v_0,\ v_1,\ \dots\ ,\ v_q,\ \dots\ ,v_{3P+1})$. Esses valores concatenados, formam o código binário que representa um determinado pixel. A transformação do código binário no seu respectivo valor em decimal consiste no somatório das multiplicações de cada binário pela sua respectiva potência de 2, como mostra a Equação 2.21:

$$VLBP_{L,P,R} = \sum_{q=0}^{3P+1} v_q \times 2^q. \tag{2.21}$$

E por fim, tendo calculado os valores de todos os pixels do volume, a próxima etapa é gerar um histograma com as frequências dos valores em decimal e com base neste histograma, um determinado vídeo poderá ser classificado. O histograma será representado um vetor de tamanho 2^{3P+2}. A Figura 2.11 ilustra o processo de obtenção do valor decimal do código VLBP de um determinado pixel do volume de um vídeo.

Caso a aplicação esteja lidando com a classificação de vídeos com orientações aleatórias, esta abordagem possui uma desvantagem: não é invariante à rotação. Suponha que A e B são dois vídeos idênticos, sendo B rotacionado em 180^o. Os vídeos A e B podem ser classificados como dois vídeos distintos pelo fato de que o VLBP leva em consideração a rotação. Para tornar invariante à rotação, o processo para geração dos códigos binários no VLBP foi incrementado.

No RI-VLBP [44], o procedimento para a definição dos sinais 0 ou 1 do código binário de um determinado pixel p é similar ao do VLBP tradicional, como foi apresentado nas Equações 2.20 e 2.21. Porém neste caso, para que a técnica seja invariante à rotação, é necessário obter o menor código binário de acordo com todas as combinações possíveis. Para isso, o binário V_2 da Equação 2.20 é dividido em 5 partes na seguinte ordem: V_{antC}, V_{antN}, V_{atualN}, V_{posN}, V_{posC}; e V_3 representa a concatenação dessas partes ordenadas:

$$V_{antC} = [s(g_{t_c-L,c}-g_{t_c,c})] \tag{2.22}$$

$$V_{antN} = [s(g_{t_c-L,0}-g_{t_c,c}),\ s(g_{t_c-L,1}-g_{t_c,c}),\ \dots\ ,\ s(g_{t_c-L,P-1}-g_{t_c,c})] \tag{2.23}$$

$$V_{atualN} = [s(g_{t_c,0}-g_{t_c,c}),\ s(g_{t_c,1}-g_{t_c,c}),\ \dots\ ,\ s(g_{t_c,P-1}-g_{t_c,c})] \tag{2.24}$$

$$V_{posN} = [s(g_{t_c+L,0}-g_{t_c,c}),\ s(g_{t_c+L,1}-g_{t_c,c}),\ \dots,\ s(g_{t_c+L,P-1}-g_{t_c,c})] \tag{2.25}$$

$$V_{posC} = [s(g_{t_c+L,c}-g_{t_c,c})] \tag{2.26}$$

$$V_3 = [V_{antC}\ V_{antN}\ V_{atualN}\ V_{posN}\ V_{posC}] \tag{2.27}$$

V_{antC} e V_{posC} referem-se ao valor binário do pixel central do quadro anterior e posterior. As partes V_{antN}, V_{atualN} e V_{posN} referem-se ao código LBP calculado nos quadros anterior, atual e posterior, respectivamente. Para tornar invariante à rotação, esses valores na base binária são padronizadamente submetidos a rotações circulares de i bits para a direita ($0 \leq i \leq P-1$) e depois cada parte é concatenada para formar o binário original de tamanho $3P+2$. A Figura 2.12 apresenta um exemplo desse procedimento com os parâmetros $L=1$, $P=4$ e $R=1$. Essas rotações têm por objetivo gerar P códigos binários de tamanho $3P+2$ em que o menor deles será o escolhido, pois será o invariante à rotação. Este processo é realizado para todo pixel

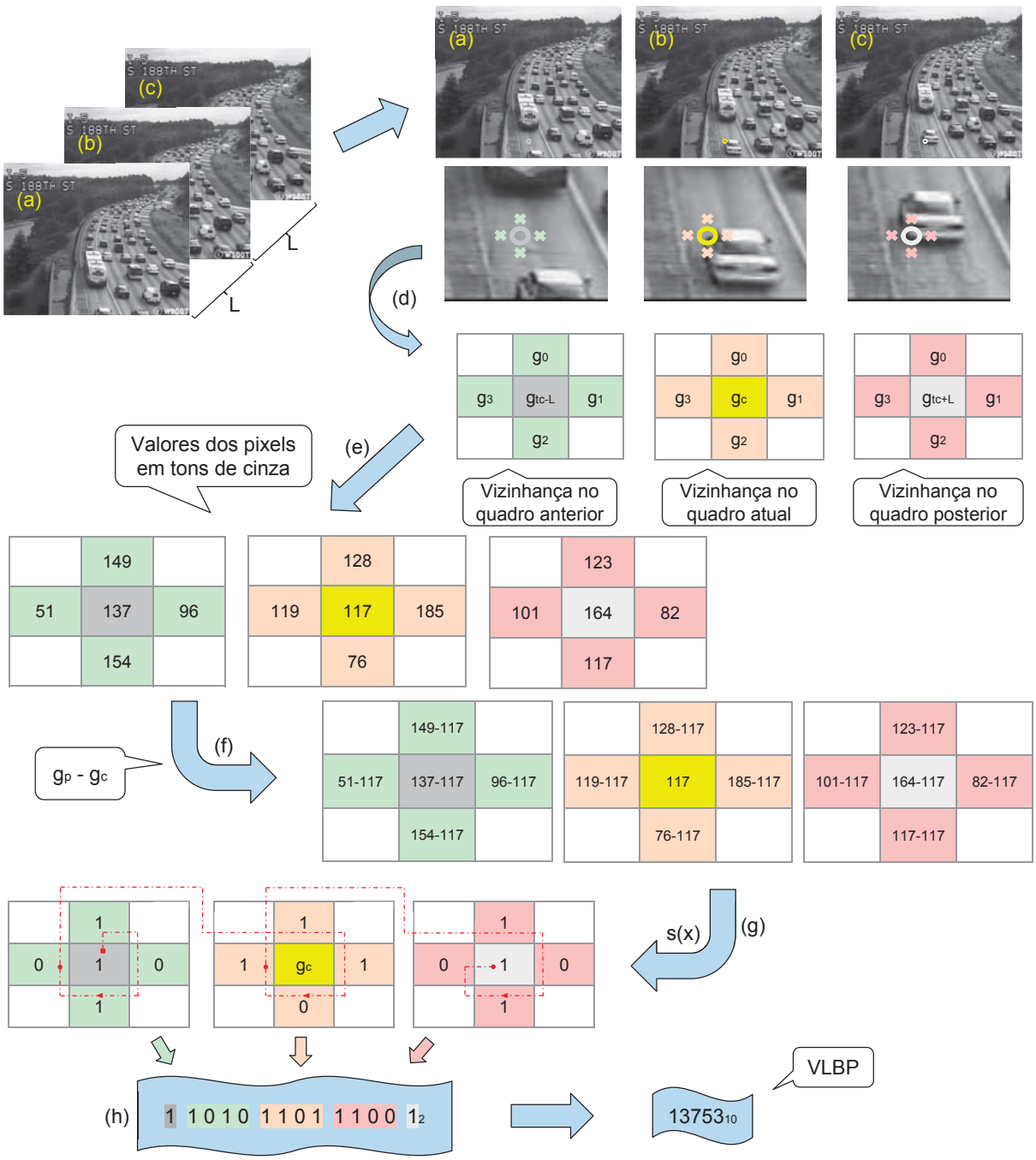


Figura 2.11: Visão geral do VLBP com $P = 4$, $R = 1$ e $L = 5$. As letras (a) e (c) representam respectivamente os quadros anterior e posterior com distância L em relação ao quadro do pixel central (b); A vizinhança nos 3 quadros é representada em (d), cada um contendo seu respectivo valor em níveis de cinza (e), que serão subtraídos pelo valor do pixel central (f) para o cálculo dos sinais (g). O código binário (h) é obtido através da concatenação dos sinais dos 3 quadros.

do volume, resultando ao final um histograma de tamanho 2^{3P+2}, contendo as frequências de cada binário convertido para a base decimal obtidos de cada pixel do volume.

2.6.2 Padrões Locais Binários nos Três Planos Ortogonais

Um vídeo é formado por um conjunto de imagens sequenciadas e pode ser representado por um cubo sobre um plano tridimensional com valores variando em três eixos: X, Y e T. A partir deste cubo, imagens podem ser obtidas através de cortes nos planos ortogonais XY,

XT e YT (Figura 1.2). Os Padrões Locais Binários nos Três Planos Ortogonais, do inglês *Local Binary Patterns on Three Orthogonal Planes* (LBP-TOP) [43], consiste em aplicar o LBP nos pixels das imagens obtidas através de cortes nos três planos ortogonais do vídeo, XY, XT e YT. O código binário resultante da aplicação do LBP para um pixel em um determinado quadro de cada plano XY, XT e YT é denotado por LBP^{XY}, LBP^{XT} e LBP^{YT}, conforme mostra a Equação 2.28). Além disso, a quantidade de vizinhos P e o raio de distância R entre a vizinhança e o pixel central são os mesmos para o cálculo dos códigos dos pixels de cada quadro dos planos ortogonais.

$$\begin{aligned} &LBP^{XY}_{i,t}, \text{ código binário de um pixel i no quadro } 1 \leq t \leq T \text{ do plano XY.} \\ &LBP^{XT}_{i,y}, \text{ código binário de um pixel i no quadro } 1 \leq y \leq Y \text{ do plano XT.} \\ &LBP^{YT}_{i,x}, \text{ código binário de um pixel i no quadro } 1 \leq x \leq X \text{ do plano YT.} \end{aligned} \tag{2.28}$$

Assim, cada plano ortogonal também terá o seu respectivo histograma de tamanho 2^P com a frequência dos códigos binários obtidos em cada quadro do seu respectivo plano, convertidos para a base decimal:

$$\begin{aligned} h^{XY}(k) &= \sum_{t=1}^{T}\sum_{i=1}^{n_t} \begin{cases} 1, & \text{se } LBP^{XY}_{i,t} = k \\ 0, & \text{caso contrário} \end{cases}, 0 \leq k < 2^P \\ h^{XT}(k) &= \sum_{y=1}^{Y}\sum_{i=1}^{n_y} \begin{cases} 1, & \text{se } LBP^{XT}_{i,y} = k \\ 0, & \text{caso contrário} \end{cases}, 0 \leq k < 2^P \\ h^{YT}(k) &= \sum_{x=1}^{X}\sum_{i=1}^{n_x} \begin{cases} 1, & \text{se } LBP^{YT}_{i,x} = k \\ 0, & \text{caso contrário} \end{cases}, 0 \leq k < 2^P \end{aligned} \tag{2.29}$$

Após o cálculo de todas as frequências, esses histogramas são concatenados, formando um único vetor de características de tamanho 3×2^P:

$$LBP\text{-}TOP = [h^{XY}\ h^{XT}\ h^{YT}] \tag{2.30}$$

A principal diferença entre o VLBP e o LBP-TOP está no tamanho do histograma gerado após o cálculo dos binários. Para P vizinhos, é gerado um histograma com 2^P posições. Como nesta técnica o LBP é aplicado nos três planos ortogonais do vídeo, são gerados 3 histogramas de tamanho 2^P, que são concatenados em um único vetor com 3×2^P características. Essa diferença sutil entre as duas técnicas possibilita por exemplo, um aumento do parâmetro P sem interferir potencialmente no tamanho do vetor de características.

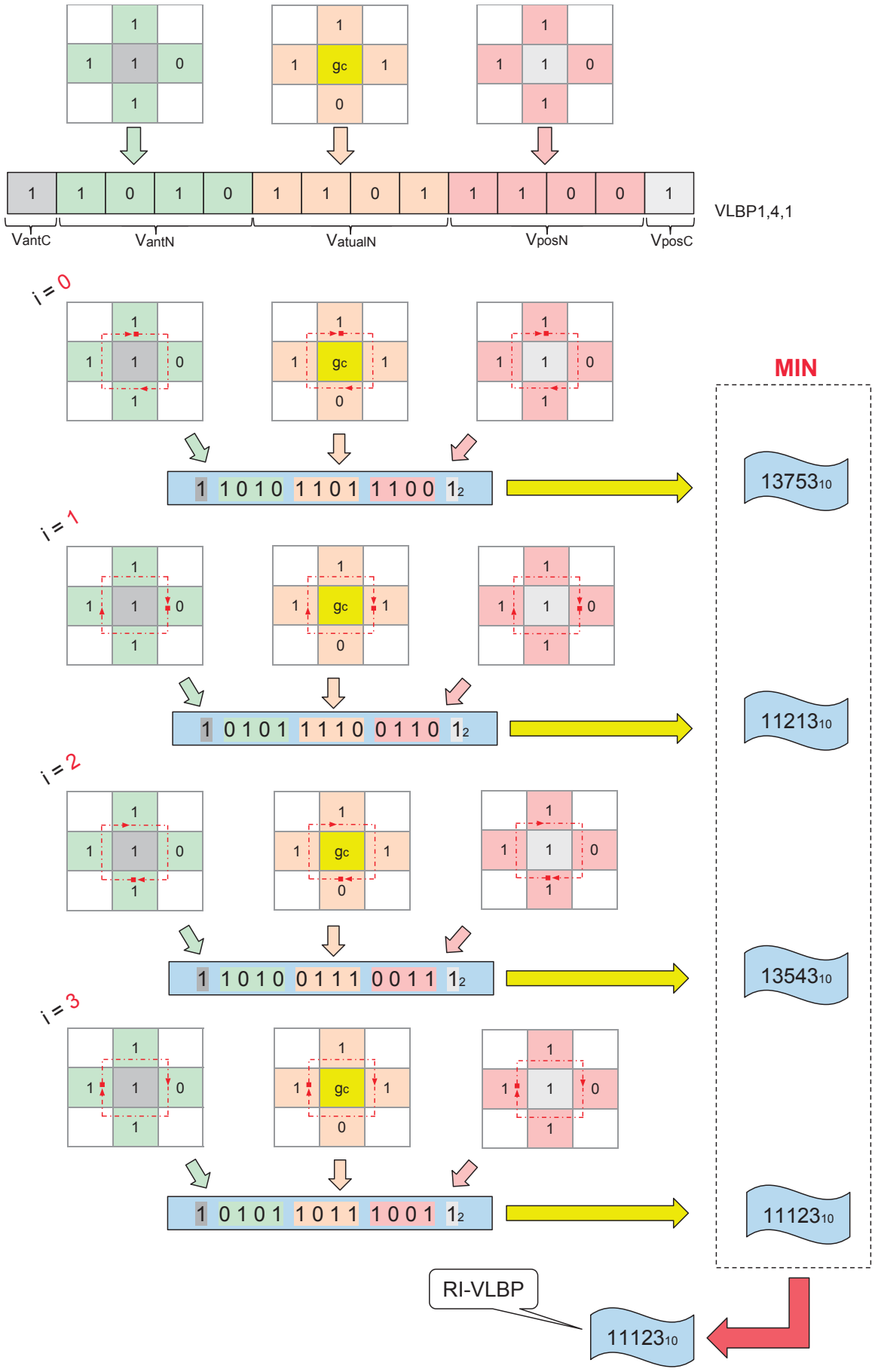


Figura 2.12: Processo para obtenção do menor código invariante a rotação. Os bits dos códigos binários dos quadros anterior, atual e posterior em relação aos seus pixels centrais são rotacionados circularmente, i vezes para a direita ($0 \leq i \leq P-1$) e a cada rotação é gerado um novo código VLBP, onde o invariante à rotação será o menor.

CAPÍTULO

3

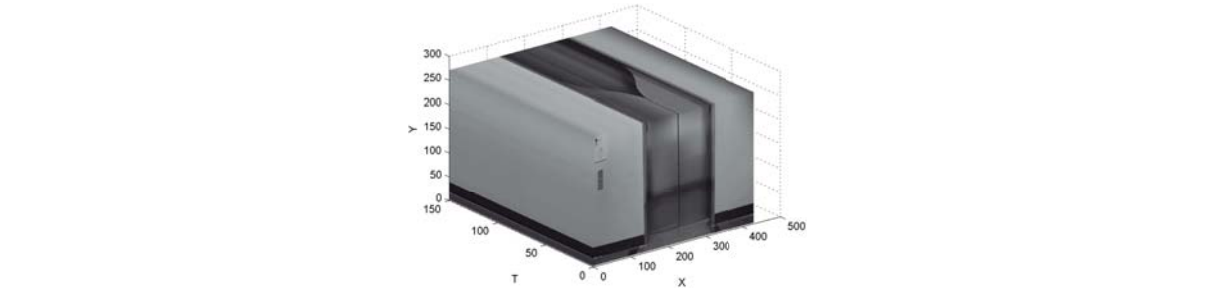
Histograma de Palavras Visuais em Planos Ortogonais

Este capítulo apresenta de forma detalhada cada etapa da metodologia utilizada para estender o BoVW para a caracterização de vídeos. Vale ressaltar que neste trabalho, são utilizados o SIFT e suas variações (SIFT Denso e PHOW) como descritores na etapa de extração de características, mas isso não restringe o método proposto a apenas estes descritores.

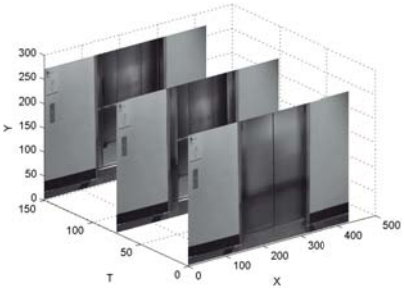
Um vídeo dividido em um conjunto de imagens sequenciadas, denominados quadros, pode ser representado por um cubo tridimensional sobre um plano com valores variando em três eixos, X, Y e T, como mostra a Figura 3.1a. Dessa forma, um determinado valor t refere-se a um quadro correspondente no vídeo. O processo dos planos ortogonais é semelhante ao do LBP-TOP [43], ou seja, a partir do supracitado cubo tridimensional é possível extrair informações referentes a três planos ortogonais denominados: plano XY, XT e YT. Nas Figuras 3.1b, 3.1c e 3.1d são ilustrados cortes em cada um dos três planos.

Similarmente ao BoVW tradicional, em sua primeira etapa, é aplicado um descritor local (por exemplo, o SIFT) para obter descritores, diferindo em que no método proposto, o descritor local é utilizado para obter descritores nos três planos ortogonais, e consequentemente são gerados três histogramas: um para cada plano ortogonal do vídeo. A Figura 3.2 ilustra o processo para gerar os três histogramas e mostra também como é gerado o descritor de um plano ortogonal do vídeo. O método proposto resume-se nas seguintes etapas aplicadas nos três planos ortogonais:

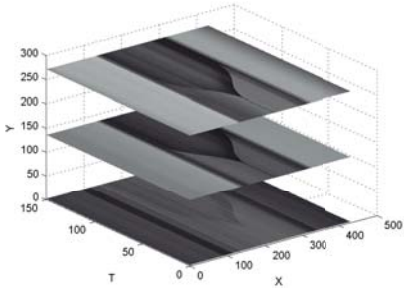
1. Detecção e descrição dos pontos de interesse nos planos XY, XT e YT.
2. Criação do vocabulário de palavras visuais para cada plano.
3. Rotulação das palavras visuais de cada plano.
4. Construção do histograma das palavras visuais.



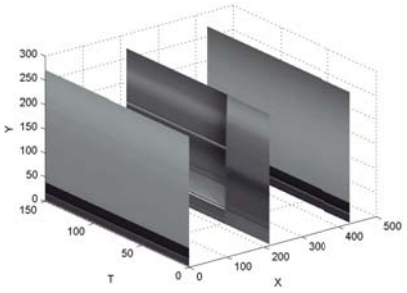
(a) Cubo 3D



(b) Plano XY



(c) Plano XT



(d) Plano YT

Figura 3.1: Representação de um vídeo de elevador da base de vídeos de cenas dinâmicas [11], no plano de dimensões X, Y, T e seus respectivos planos ortogonais. O cubo no plano 3D possui dimensões $X = 452$, $Y = 270$ e $T = 150$.

3.1 Detecção e Descrição de Pontos de Interesse nos Planos Ortogonais

Esta etapa consiste em realizar a detecção dos pontos de interesse em cada plano ortogonal do vídeo, seguida pela definição de um vetor de características (descritores) para cada ponto. Para essa detecção e descrição, diversos descritores locais podem ser aplicados. Dessa forma, ao final desta etapa, um conjunto de descritores é obtido para cada plano.

Conforme descrito anteriormente, um vídeo pode ser representado por um cubo tridimensional. Para obter-se imagens em que detectores e descritores locais possam ser aplicados, cortes em três planos ortogonais podem ser realizados. Por exemplo, para o plano XY, os cortes correspondem aos T quadros do vídeo. Para o plano XT, podem ser realizados Y cortes, onde Y corresponde à altura das imagens do vídeo.

Portanto, cada plano ortogonal é composto por vários quadros (imagens), tal que em cada quadro, pontos de interesse podem ser detectados. Assim, cada ponto de interesse de cada quadro de um determinado plano ortogonal é representado da seguinte forma:

$$d^{XY}_{t,i}\text{, descritor do ponto de interesse } i \text{ obtido no quadro } 1 \leq t \leq T \text{ do plano XY.}$$

$$d^{XT}_{y,i}\text{, descritor do ponto de interesse } i \text{ obtido no quadro } 1 \leq y \leq Y \text{ do plano XT.} \tag{3.1}$$

$$d^{YT}_{x,i}\text{, descritor do ponto de interesse } i \text{ obtido no quadro } 1 \leq x \leq X \text{ do plano YT.}$$

Ao final desta etapa, obtém-se descritores para cada plano ortogonal do vídeo, com base nos pontos de interesse detectados em seus respectivos quadros. Portanto, para um dado

vídeo V, três conjuntos de descritores são obtidos:

$$D_V^{XY} = [d_{t_1,0}^{XY}, d_{t_1,1}^{XY}, \ldots, d_{t_1,n_{t_1}}^{XY}, d_{t_2,0}^{XY}, d_{t_2,1}^{XY}, \ldots, d_{t_T,n_{t_T}}^{XY}],$$

onde $d_{t_i,n}^{XY}$ corresponde ao descritor do n-ésimo ponto de interesse do quadro i do plano XY;

$$D_V^{XT} = [d_{y_1,0}^{XT}, d_{y_1,1}^{XT}, \ldots, d_{y_1,n_{y_1}}^{XT}, d_{y_2,0}^{XT}, d_{y_2,1}^{XT}, \ldots, d_{y_Y,n_{y_Y}}^{XT}], \tag{3.2}$$

onde $d_{y_i,n}^{XT}$ corresponde ao descritor do n-ésimo ponto de interesse do quadro i do plano XT;

$$D_V^{YT} = [d_{x_1,0}^{YT}, d_{x_1,1}^{YT}, \ldots, d_{x_1,n_{x_1}}^{YT}, d_{x_2,0}^{YT}, d_{x_2,1}^{YT}, \ldots, d_{x_X,n_{x_X}}^{YT}],$$

onde $d_{x_i,n}^{YT}$ corresponde ao descritor do n-ésimo ponto de interesse do quadro i do plano YT.

Além disso, cada descritor $d_{t_i,n}^{XY}$ caracteriza a aparência de um ponto de interesse do plano XY. Enquanto que cada descritor $d_{y_i,n}^{XT}$ e $d_{x_i,n}^{YT}$ caracteriza o movimento de um ponto de interesse dos planos temporais XT e YT.

3.2 Vocabulários de Palavras Visuais

Similarmente ao BoVW tradicional, os conjuntos de descritores de vários vídeos são concatenados para obter-se as palavras visuais. Portanto, D^{XY} corresponde a concatenação dos descritores obtidos do plano XY de vários vídeos. Assim, formam-se três conjuntos de descritores, um para cada plano, conforme a Equação 3.3.

$$\begin{aligned} D^{XY} &= [D_{V_1}^{XY}, D_{V_2}^{XY}, ..., D_{V_n}^{XY}] \\ D^{XT} &= [D_{V_1}^{XT}, D_{V_2}^{XT}, ..., D_{V_n}^{XT}] \\ D^{YT} &= [D_{V_1}^{YT}, D_{V_2}^{YT}, ..., D_{V_n}^{YT}] \end{aligned} \tag{3.3}$$

Para cada plano, um vocabulário de palavras visuais é gerado por meio da aplicação do algoritmo K-Médias, como mostra a Equação 3.4. Conforme visto na Subseção 2.5.2, o K-Médias consiste em gerar K centroides, em que a cada iteração seus valores são recalculados, gerando um novo agrupamento. Assim, são gerados três vocabulários de palavras visuais, C^{XY}, C^{XT} e C^{YT} que contém as palavras visuais dos planos XY, XT e YT. Dessa forma, cada vocabulário possui informações relevantes de características dos vídeos, no seu respectivo plano. O vocabulário C^{XY} descreve as características de aparência enquanto os vocabulários C^{XT} e C^{YT} sumarizam as características de movimento dos vídeos.

$$\begin{aligned} C^{XY} &= \text{K-Médias}(D^{XY}) \\ C^{XT} &= \text{K-Médias}(D^{XT}) \\ C^{YT} &= \text{K-Médias}(D^{YT}) \end{aligned} \tag{3.4}$$

3.3 Rotulação

Dado um vídeo V, é realizado o procedimento para detecção dos pontos de interesse dos quadros em cada plano ortogonal. Em seguida, cada ponto de interesse é associado a uma palavra visual, com base no seu respectivo vocabulário de palavras visuais C, criado na etapa anterior. Por exemplo, os pontos de interesse detectados nos quadros do plano XY, são associados às palavras visuais do vocabulário C^{XY}.

A atribuição de uma palavra visual aos pontos de interesse do vídeo também se baseia no cálculo da distância Euclidiana entre cada palavra do vocabulário com um determinado descritor. Portanto, cada ponto de interesse do plano é rotulado a uma palavra visual cuja distância Euclidiana entre ela e o descritor obtiver menor valor:

$$\begin{aligned} w_{t,i}^{V^{XY}} &= \arg\min_{j=1}^{K} \| d_{t,i}^{XY}, C_j^{XY} \|,\ 1 \leq t \leq T \\ w_{y,i}^{V^{XT}} &= \arg\min_{j=1}^{K} \| d_{y,i}^{XT}, C_j^{XT} \|,\ 1 \leq y \leq Y \\ w_{x,i}^{V^{YT}} &= \arg\min_{j=1}^{K} \| d_{x,i}^{YT}, C_j^{YT} \|,\ 1 \leq x \leq X \end{aligned} \tag{3.5}$$

3.4 Histograma dos Três Planos Ortogonais

Por fim, esta etapa consiste em gerar três histogramas h^{XY}, h^{XT} e h^{YT} das palavras visuais de cada plano ortogonal do vídeo cujos seus respectivos pontos de interesse foram rotulados na etapa anterior. Cada histograma possui o tamanho do vocabulário de palavras visuais do seu respectivo plano ortogonal e cada posição representa a frequência em que cada palavra visual ocorre naquele plano ortogonal do vídeo, como apresentado na Equação 3.6. Assim como no BoVW tradicional, o cálculo da frequência consiste em somar a quantidade de descritores que foram associados a cada palavra visual, em cada plano ortogonal.

$$\begin{aligned} h^{XY}(j) &= \sum_{t=1}^{T} \sum_{i=1}^{n_t} \begin{cases} 1, & \text{se } w_{t,i}^{V^{XY}} = C_j^{XY} \\ 0, & \text{caso contrário} \end{cases}, 1 \leq j \leq K \\ h^{XT}(j) &= \sum_{y=1}^{Y} \sum_{i=1}^{n_y} \begin{cases} 1, & \text{se } w_{y,i}^{V^{XT}} = C_j^{XT} \\ 0, & \text{caso contrário} \end{cases}, 1 \leq j \leq K \\ h^{YT}(j) &= \sum_{x=1}^{X} \sum_{i=1}^{n_x} \begin{cases} 1, & \text{se } w_{x,i}^{V^{YT}} = C_j^{YT} \\ 0, & \text{caso contrário} \end{cases}, 1 \leq j \leq K \end{aligned} \tag{3.6}$$

A Figura 3.2 resume os passos do método proposto. Inicialmente é aplicado um descritor local em cada plano ortogonal dos vídeos para detectar e descrever os pontos de interesse (Figura 3.2a), gerando 3 vetores de características: um para cada plano XY, XT e YT. Com base nos descritores dos vídeos obtidos anteriormente, é aplicado o algoritmo de agrupamento K-Médias, gerando também um vocabulário de palavras visuais para cada plano ortogonal (Figura 3.2b). Em seguida, cada ponto de interesse é rotulado a uma palavra visual do vocabulário do seu respectivo plano ortogonal (Figura 3.2c). E finalmente para cada vídeo, são calculados 3 histogramas: cada um contendo a frequência das palavras visuais do seu respectivo plano ortogonal (Figura 3.2d).

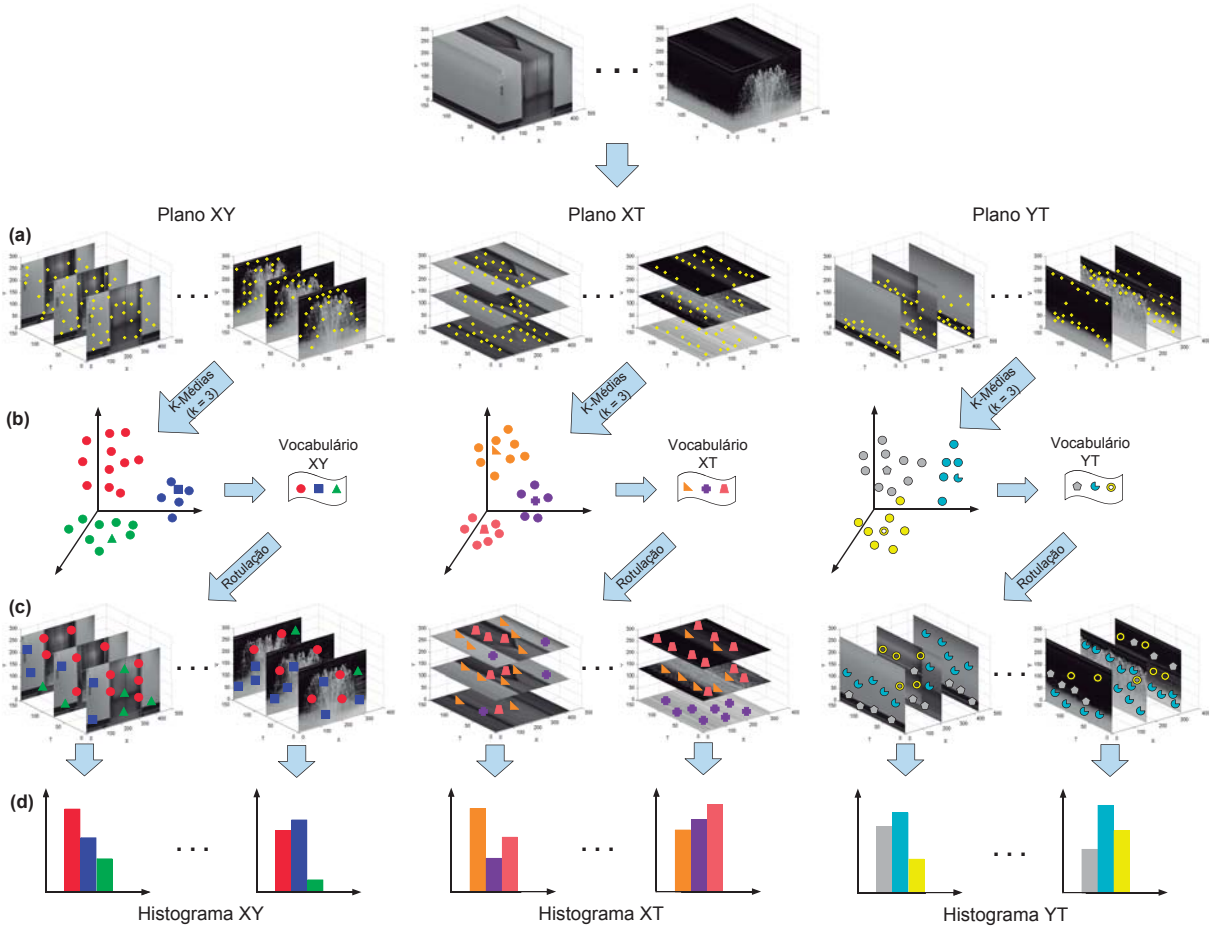


Figura 3.2: Visão geral da metodologia para obtenção dos três histogramas de cada vídeo. Primeiro é aplicado o descritor local em cada plano ortogonal, para detectar e descrever os pontos de interesse, resultando em 3 vetores de características para cada vídeo (a). Com base nos descritores obtidos em cada plano dos vídeos, é aplicado o K-Médias, gerando 3 vocabulários de palavras visuais (b). Em seguida é realizado o procedimento de rotulação (c) dos pontos de interesse detectados em (a), com base nas palavras obtidas anteriormente. E por fim são calculados 3 histogramas, cada um contendo a frequência de palavras visuais do seu respectivo plano ortogonal (d).

CAPÍTULO

4

Experimentos e Resultados

Este capítulo descreve os experimentos e respectivos resultados obtidos pelo método proposto, pelo BoVW utilizando o SIFT 3D e pelos métodos de texturas dinâmicas. Para isso, foram utilizadas duas bases de vídeos, descritas na Seção 4.1. Nos experimentos com o método proposto foram utilizados três descritores locais aplicados nos planos ortogonais: SIFT [25], SIFT Denso [24] e PHOW [3]. Os resultados obtidos pelo método proposto foram comparados aos obtidos pelo BoVW com o SIFT 3D e com dois métodos de texturas dinâmicas que estendem os Padrões Locais Binários (LBP) para caracterização de vídeos: Padrões Locais Binários Volumétricos Invariantes à Rotação [44] (RI-VLBP); e Padrões Locais Binários nos Três Planos Ortogonais [43] (LBP-TOP).

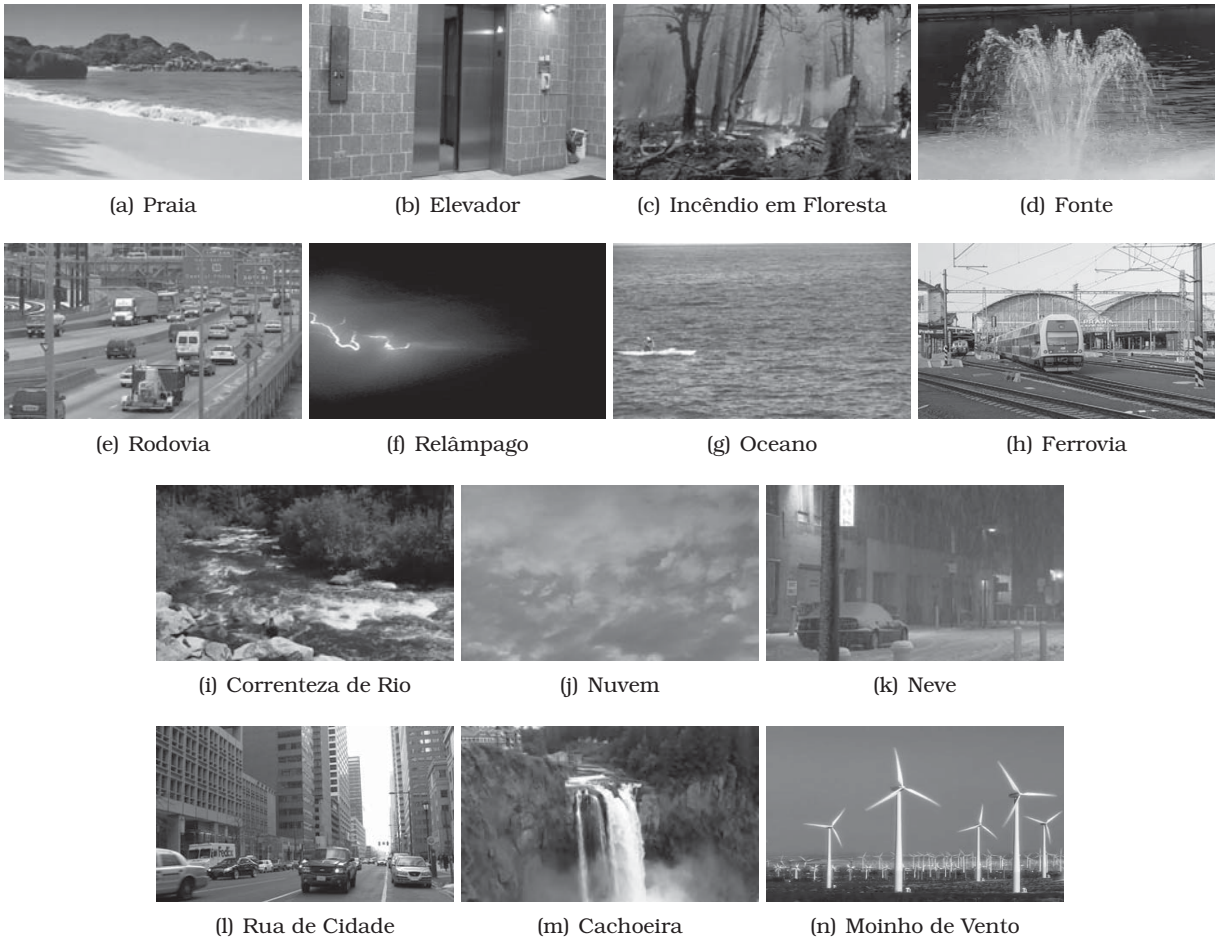
4.1 Experimentos

Duas bases de dados foram utilizadas para os experimentos: uma contendo vídeos de tráfego de carros [4]; e a segunda, contendo vídeos de cenas dinâmicas [11]. A primeira base de vídeos está separada em três classes de tráfego: leve, médio e pesado. Cada uma possui, respectivamente, 165, 45 e 44 vídeos, totalizando 254 exemplos, com cada vídeo sendo composto, em média, por 51 quadros com dimensões de aproximadamente 352×288 pixels. Além disso, a posição da câmera que capturou todos os vídeos dessa base é praticamente a mesma.

(a) Leve (b) Médio (c) Pesado

Figura 4.1: Exemplos de quadros extraídos dos vídeos de cada classe da base de dados de tráfego de carros [4].

A segunda base de vídeos totaliza 420 vídeos, divididos igualmente entre 14 classes de cenas dinâmicas: praia, elevador, incêndio em floresta, fonte, ferrovia, correnteza de rio, nuvem, neve, rodovia, relâmpago, oceano, rua de cidade, cachoeira e moinho de vento em fazenda. Cada vídeo é composto por, em média, 145 quadros com dimensões de aproximadamente 370×250 pixels. As Figuras 4.1 e 4.2 apresentam, alguns exemplos de quadros extraídos dos vídeos de cada classe das bases de tráfego de carros e cenas dinâmicas.



(a) Praia (b) Elevador (c) Incêndio em Floresta (d) Fonte

(e) Rodovia (f) Relâmpago (g) Oceano (h) Ferrovia

(i) Correnteza de Rio (j) Nuvem (k) Neve

(l) Rua de Cidade (m) Cachoeira (n) Moinho de Vento

Figura 4.2: Exemplos de quadros extraídos dos vídeos de cada categoria da base de dados de cenas dinâmicas [11].

No método proposto, os vídeos tiveram seus descritores extraídos a partir do SIFT, SIFT Denso e PHOW aplicados nos planos ortogonais gerando três descritores, um para cada plano: XY, XT e YT. Para cada plano, foi aplicado o algoritmo K-Médias para construir um vocabulário com K palavras visuais. Nos experimentos, os valores dos centroides eram recalculados enquanto a alteração dos valores dos centroides fosse maior ou igual a 0.00001 ou o número de iterações fosse menor ou igual a 100. Além disso, o número de palavras visuais foi variado de 100 a 3500. Por fim, um histograma para cada plano foi obtido. Para avaliar a influência de cada plano no reconhecimento, diferentes combinações de planos foram testadas: XY, XT, XY (planos individuais); XY-XT, XY-YT, XT-YT (combinação de dois planos); e a combinação dos três planos XY-XT-YT.

Na etapa de classificação foi utilizado o classificador Máquina de Vetores de Suporte (do inglês - *Support Vector Machines* - SVM) [23]. O SVM foi avaliado por meio do *leave-one-video-out*, a fim de manter coerência com os resultados de avaliações realizadas em trabalhos correlatos [35, 11, 38, 16, 15]. E a taxa de classificação correta foi utilizada como métrica para a comparação dos resultados obtidos com o método proposto com métodos da literatura.

4.2 Resultados e Discussão

Esta seção está dividida em quatro subseções a fim de organizar melhor as discussões, de acordo com cada possível fator de influência nos resultados da abordagem proposta: tamanho do vocabulário (Subseção 4.2.1) e planos ortogonais (Subseção 4.2.2). Na Subseção 4.2.3 são discutidos alguns vídeos classificados incorretamente pelo método proposto em seu melhor resultado. E por fim, na Subseção 4.2.4, os resultados da abordagem proposta neste trabalho foram comparados com os obtidos por métodos da literatura.

4.2.1 Tamanho do Vocabulário

Nestes experimentos, em ambas as bases e por questões de tempo e recurso computacional, o SIFT Denso e PHOW foram modificados para que fosse escolhido um ponto a cada 16 pixels dos quadros nos planos ortogonais dos vídeos, na primeira etapa do método proposto (extração de características). Dessa forma seus resultados puderam ser comparados de forma mais justa com os obtidos pelo SIFT 3D (Subseção 4.2.4) que também foi modificado para extrair um descritor a cada 16 pixels do cubo 3D dos vídeos, conforme sugerido pelos autores.

Os gráficos da Figura 4.3 apresentam, para a base de tráfego de carros, as taxas de classificações corretas obtidas pelo método proposto utilizando separadamente os descritores SIFT, SIFT Denso e PHOW, em cada combinação distinta dos planos ortogonais. Cada curva representa uma combinação: as curvas em vermelho e tracejado representam os resultados obtidos com os planos individualmente; as curvas em azul e pontilhado representam os resultados obtidos através da combinação de dois planos ortogonais e; a curva em linha contínua na cor verde, representa os resultados obtidos com a combinação dos três planos XY-XT-YT. De acordo com a Figura 4.3, o método proposto obteve os seguintes resultados usando cada descritor:

- SIFT: o método proposto utilizando o SIFT como descritor local precisou da combinação dos planos ortogonais XY-YT e tamanho do vocabulário de palavras visuais $K = 300$ para obter 98.03% de taxa de classificação correta. Considerando a combinação XY-YT que obteve melhor resultado, a taxa de classificação correta não aumentou para tamanhos de vocabulários K maiores até 3500, oscilando entre 95.66% e 97.63%. Na combinação dos três planos ortogonais XY-XT-YT, o método proposto alcançou um pico de 97.24% na taxa de classificação correta com tamanho do vocabulário $K = 600$, não superando o resultado anterior. Além disso, o aumento da quantidade de palavras do vocabulário também não melhorou a taxa de classificação correta obtida com a combinação XY-XT-YT, indicando que vocabulários maiores não auxiliaram ou melhoraram a modelagem das palavras visuais.

- SIFT Denso: o método proposto utilizando o SIFT Denso como descritor local obteve a mesma taxa de classificação correta que a obtida utilizando o SIFT. Porém, precisou de 700 palavras a mais no vocabulário e precisou combinar os três planos ortogonais XY-XT-YT para obter esse resultado. As demais combinações não alcançaram ou superaram a taxa de classificação correta obtida com a combinação XY-XT-YT. Considerando esta combinação, para tamanhos de vocabulários K maiores do que 1000 até 3500, a taxa de classificação correta oscilou na faixa dos 97%, não superando a obtida com $K = 1000$.

- PHOW: o método proposto com o PHOW precisou de 2000 palavras no vocabulário para alcançar os mesmos 98.03% de classificação correta. No entanto, obteve essa taxa em três combinações de planos: XY, XY-YT e XY-XT-YT. Além disso, o método proposto também alcançou a mesma taxa de classificação correta com tamanhos de vocabulários

$K = 2500$ e $K = 3000$ na combinação de planos XY-XT-YT. Isso mostra que os tamanhos de vocabulário de 2000 a 3000 geraram bons agrupamentos, suficientes para manter a melhor taxa de classificação correta.

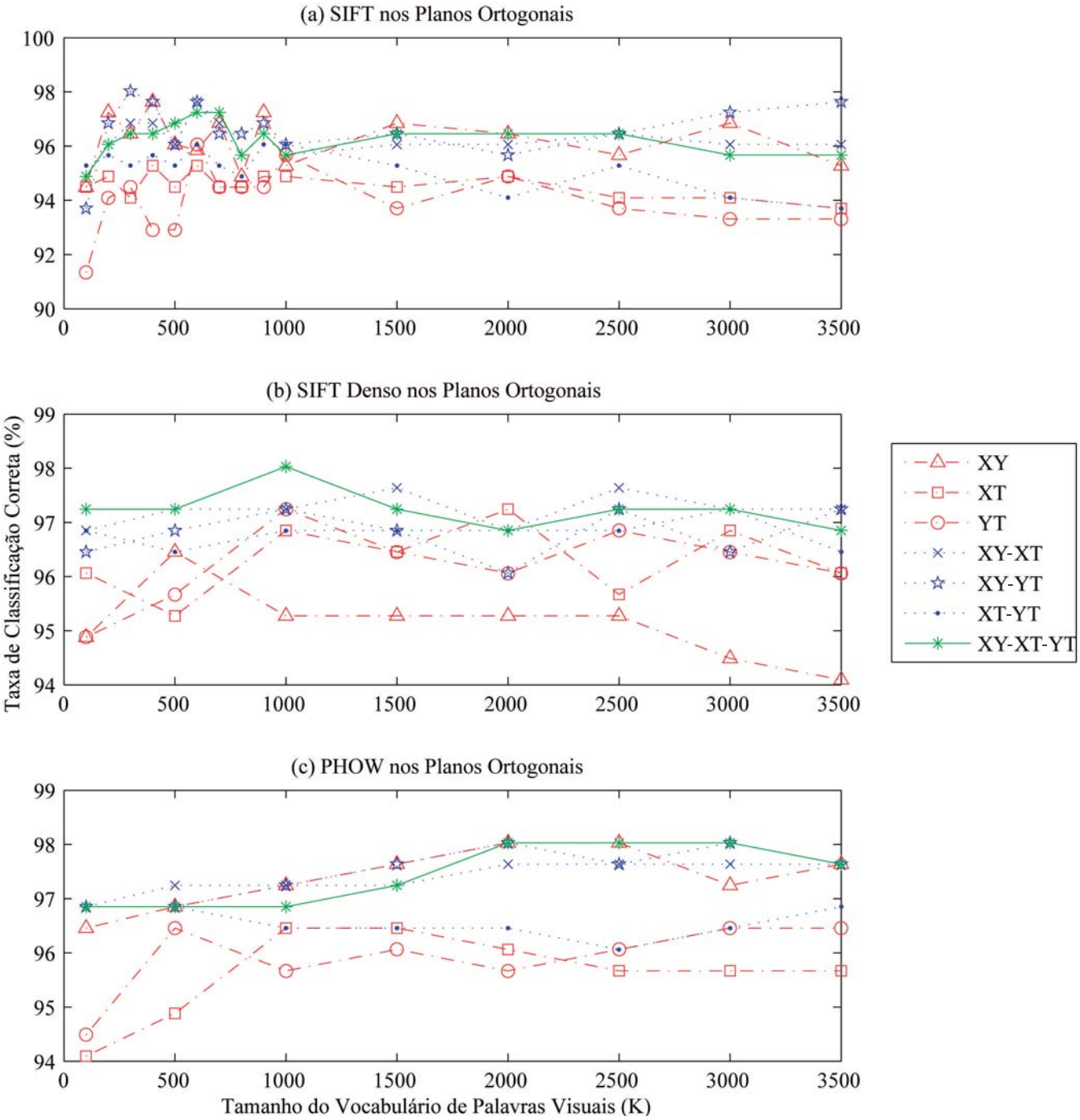


Figura 4.3: Taxa de classificação correta na base de vídeos de tráfego de carros, obtida pelo método proposto utilizando os descritores SIFT (a), SIFT Denso (b) e PHOW (c). Os descritores foram aplicados nos planos ortogonais dos vídeos e o método proposto foi avaliado pelo classificador *SVM*, com o vocabulário de palavras K variando de 100 a 3500.

Apesar do método proposto ter obtido a mesma taxa de classificação correta com os três descritores, o SIFT Denso e PHOW utilizaram mais pontos de interesse para construção do vocabulário, já que descrevem de forma densa os pontos da imagem e, neste caso, de cada quadro nos planos ortogonais.

Para a base de cenas dinâmicas, as taxas de classificações corretas obtidas pelo método

proposto com cada descritor nas 7 combinações de planos ortogonais e K variando de 100 a 3500, são apresentadas nos gráficos da Figura 4.4. De forma similar à Figura 4.3, os resultados apresentados pelas curvas estão destacados pela quantidade de planos ortogonais envolvidos nas combinações: XY, XT, YT em vermelho tracejado; XY-XT, XY-YT, XT-YT em azul pontilhado; e XY-XT-YT em linha contínua verde. De acordo com a Figura 4.4 o método proposto obteve os seguintes resultados:

- SIFT: o método proposto utilizando o SIFT como descritor local precisou de 900 palavras no vocabulário e combinação dos três planos ortogonais XY-XT-YT para alcançar sua melhor taxa de classificação correta, 93.80%. A partir de $K = 1000$, a taxa de classificação correta diminuiu, oscilando na faixa dos 92% até $K = 3500$.

- SIFT Denso: o método proposto utilizando o SIFT Denso obteve sua melhor taxa de classificação correta em 93.33%, combinando apenas os planos temporais XT-YT e com tamanho do vocabulário de palavras $K = 500$. O aumento da quantidade de palavras do vocabulário pode não ter gerado bons agrupamentos, visto que para tamanhos de vocabulários K maiores do que 500, a taxa de classificação correta não aumentou com a combinação XT-YT. Considerando a combinação dos planos XY-XT-YT o método proposto alcançou um pico de 93.09% na taxa de classificação correta com tamanho de vocabulário $K = 3000$. Para tamanhos de vocabulários maiores do que 3000, a taxa de classificação correta diminuiu.

- PHOW: o método proposto utilizando o PHOW como descritor local obteve a melhor taxa de classificação correta em comparação às obtidas com os descritores SIFT e SIFT Denso: 97.14%. Para alcançar esse resultado, foi necessária a combinação dos planos XY-XT-YT e 2000 palavras no vocabulário. Com tamanhos de vocabulários K maiores que 2000, a taxa de classificação correta diminuiu e estabilizou em 96.19% até $K = 3500$.

4.2.2 Planos Ortogonais

O gráfico da Figura 4.5 apresenta para a base de vídeos de tráfego de carros, um comparativo entre planos ortogonais baseado na melhor taxa de classificação correta obtida pelo método proposto utilizando os descritores locais SIFT, SIFT Denso e PHOW, e o tamanho do vocabulário de palavras visuais K. De acordo com a Figura 4.5 foram obtidos os seguintes resultados:

- SIFT: considerando os planos individualmente, o plano XY acarretou em melhor taxa de classificação correta do que os planos temporais XT ou YT. Consequentemente, a combinação do plano espacial com um dos planos temporais (XY-XT ou XY-YT), levou o método proposto a alcançar taxas de classificações corretas superiores à combinação dos dois planos temporais XT-YT. O melhor resultado do método proposto utilizando este descritor foi com a combinação XY-YT. Em contrapartida, quando os três planos ortogonais foram combinados, a taxa de classificação correta do método proposto foi superior apenas à combinação dos dois planos temporais e aos planos XT e YT individualmente.

- SIFT Denso: para este descritor, o plano XY acarretou em taxa de classificação correta inferior aos planos temporais XT ou YT. Apesar disso, o plano espacial contribuiu de tal forma que as combinações XY-XT e XY-YT foram superiores à combinação XT-YT. Mas com a combinação dos três planos ortogonais XY-XT-YT, a taxa de classificação correta obtida pelo método proposto foi superior às demais combinações dos planos.

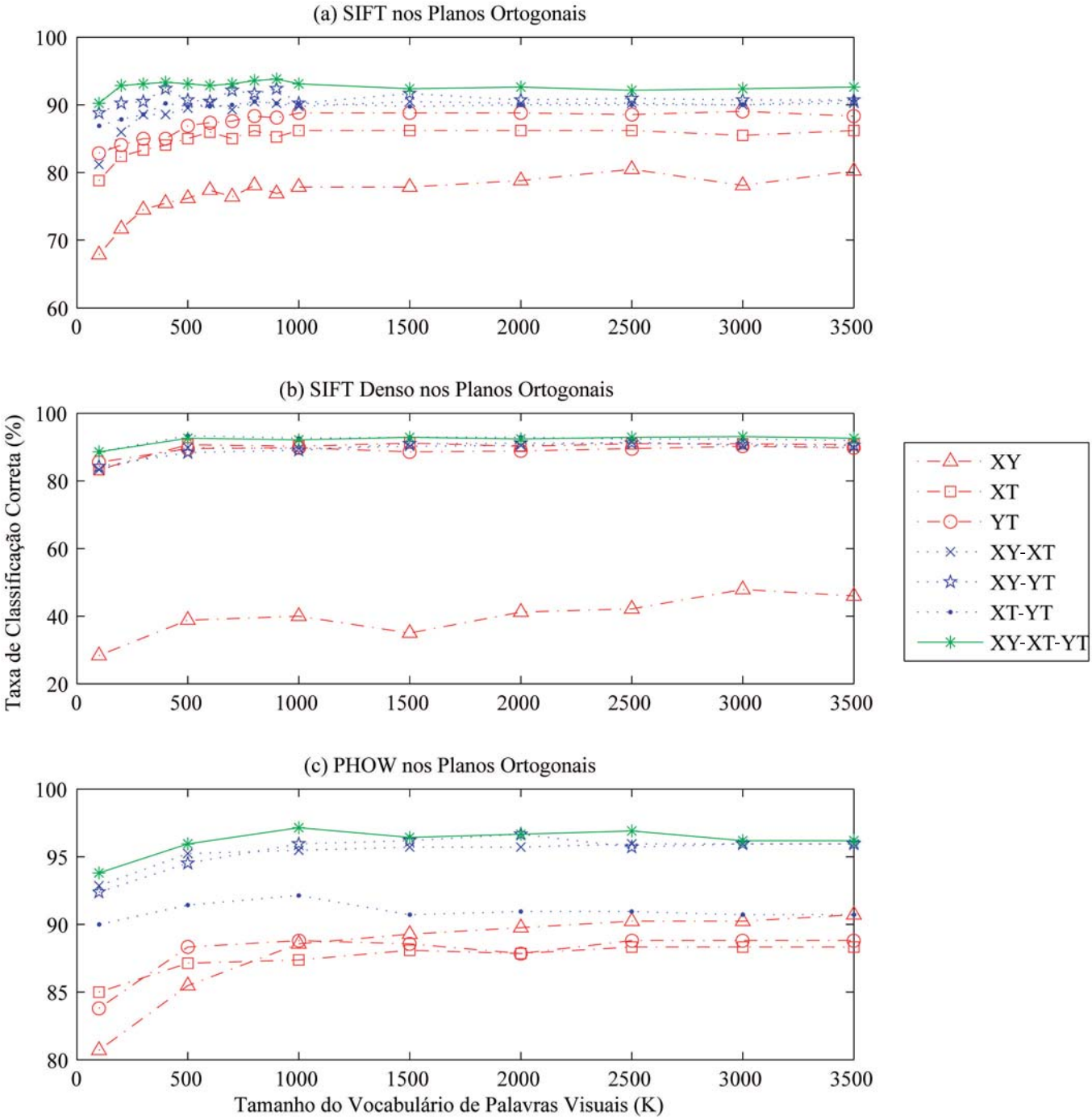


Figura 4.4: Taxa de classificação correta na base de vídeos de cenas dinâmicas, obtida pelo método proposto utilizando os descritores SIFT (a), SIFT Denso (b) e PHOW (c). Os descritores foram aplicados nos planos ortogonais dos vídeos e o método proposto foi avaliado pelo classificador *SVM*, com o vocabulário de palavras K variando de 100 a 3500.

- PHOW: a taxa de classificação correta obtida pelo método proposto utilizando este descritor foi superior com o plano XY em relação aos planos temporais XT e YT. De forma similar ao SIFT, os planos XT e YT contribuíram de tal forma para que as combinações XY-XT e XY-YT acarretassem em taxas de classificações corretas maiores à obtida pela combinação dos planos temporais XT-YT. Embora a combinação dos três planos XY-XT-YT tenha gerado a melhor taxa de classificação correta, o método proposto utilizando este descritor local também alcançou um resultado equivalente ao obtido com as combinações XY e XY-YT, superiores às demais combinações de planos ortogonais.

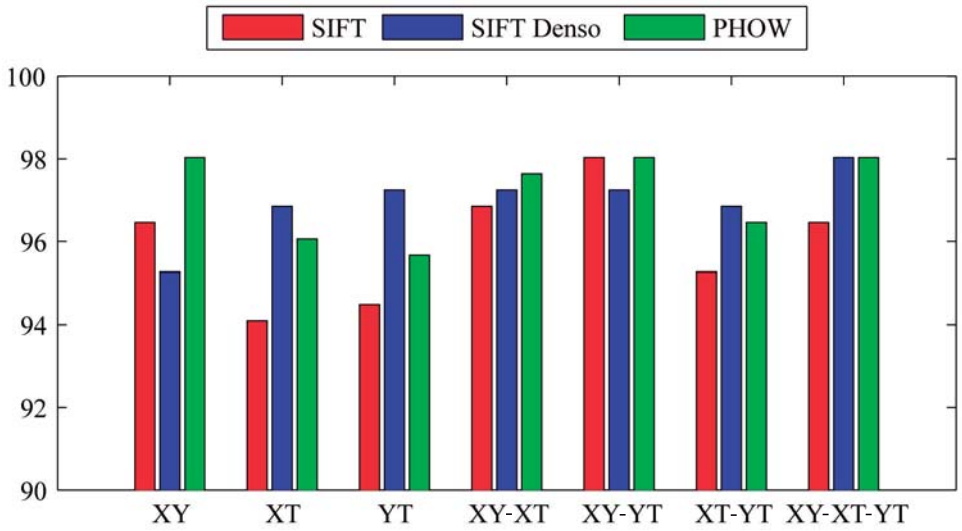


Figura 4.5: Gráfico em coluna com os resultados das combinações de planos ortogonais que obtiveram as melhores taxas de classificação correta na base de tráfego de carros, de acordo com o seu respectivo tamanho de vocabulário de palavras visuais K obtido por cada descritor.

Especificamente nesta base, a posição da câmera que capturou os tráfegos é praticamente a mesma em todos os vídeos. Todas as amostras são da mesma rodovia e os carros sempre trafegam ao longo do eixo Y. Por conta disso o plano YT pode ter conseguido caracterizar os vídeos melhor do que o plano XT, com exceção do PHOW. Além disso, o plano XY contribui de tal forma que é possível discriminar parcialmente a categoria de tráfego apenas pela quantidade de carros na rodovia, embora a inclusão de informação temporal aumente a taxa de classificação correta.

O gráfico da Figura 4.6 apresenta um comparativo entre planos ortogonais com os descritores locais SIFT, SIFT Denso e PHOW, na base de vídeos de cenas dinâmicas. O método proposto obteve os seguintes resultados:

- SIFT: o método proposto utilizando este descritor com o plano XY obteve taxa de classificação correta inferior às obtidas pelos planos temporais XT e YT. Apesar disso, o plano XY contribuiu de certa forma para que as combinações XY-XT e XY-YT fossem superiores à combinação XT-YT. No entanto, método proposto utilizando o SIFT como descritor local obteve sua melhor taxa de classificação correta com a combinação dos três planos ortogonais (XY-XT-YT).

- SIFT Denso: a taxa de classificação correta obtida pelo método proposto com o plano XY foi bastante inferior às obtidas pelos planos temporais XT e YT. Por conta disso, o método proposto utilizando o SIFT Denso obteve maior taxa de classificação correta na combinação dos dois planos temporais (XT-YT) em relação às combinações que envolvem o plano espacial e um plano temporal (XY-XT e XY-YT). Além disso, o melhor resultado do método proposto foi obtido com essa combinação XT-YT, embora a combinação dos três planos ortogonais tenha sido melhor do que às obtidas pelas outras combinações.

- PHOW: dentre os três planos ortogonais, o YT foi o que acarretou na maior taxa de classificação correta obtida pelo método proposto. Embora, a combinação XY-YT tenha sido melhor do que as demais combinações de dois planos, a combinação dos dois planos temporais XT-YT foi inferior às combinações de um plano espacial com um plano temporal. No entanto, a melhor taxa de classificação correta alcançada foi obtida com a combinação dos três planos XY-XT-YT.

Diferentemente da base de tráfego de carros, os vídeos de cada categoria de cenas dinâmicas foram capturados em lugares e posições da câmera distintos. Essa pode ser a razão pela qual a informação espacial obtida pelo plano XY tenha acarretado em taxas de classificações corretas inferiores às obtidas pelos planos temporais XT ou YT, ou seja, nesta base os vídeos são mais discriminados pelo movimento (informação temporal) do que pela aparência (informação espacial). Entretanto, a combinação das duas informações melhora a caracterização de vídeos.

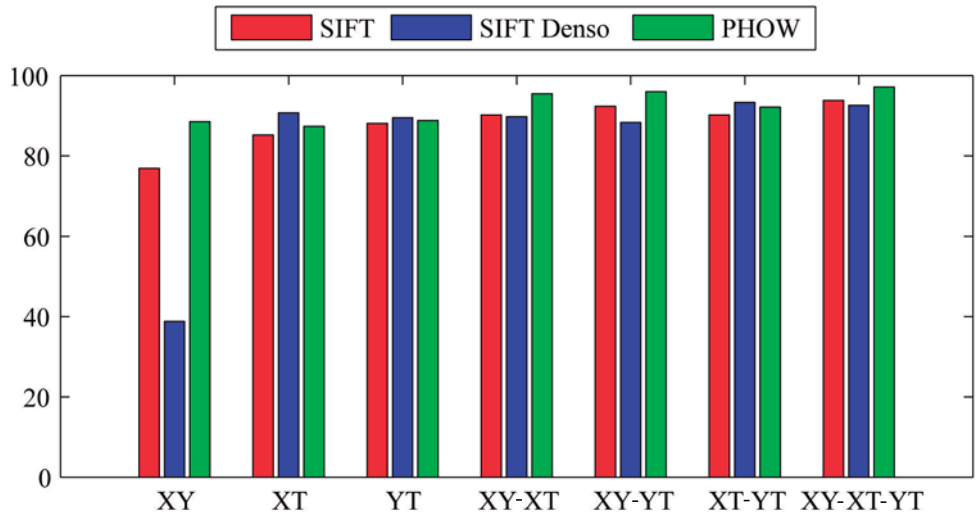


Figura 4.6: Gráfico em coluna com os resultados das combinações de planos ortogonais que obtiveram as melhores taxas de classificação correta na base de cenas dinâmicas, de acordo com o seu respectivo tamanho de vocabulário de palavras visuais K obtido por cada descritor.

4.2.3 Vídeos Classificados Incorretamente

Nesta subseção são discutidos alguns vídeos classificados incorretamente pelo método proposto utilizando o PHOW como descritor local, pois este descritor se destacou nos resultados obtidos pelo método proposto em ambas as bases de vídeos. Portanto, para a base de tráfego de carros além do descritor supracitado, foi considerado o vocabulário de palavras visuais de tamanho $K = 2000$ e combinação dos planos ortogonais XY-XT-YT. Enquanto que para os vídeos de cenas dinâmicas, foram considerados: o PHOW como descritor local, vocabulário de palavras visuais de tamanho $K = 1000$ e combinação dos planos ortogonais XY-XT-YT.

A Figura 4.7 apresenta os quadros de vídeos da base de tráfego de carros que o método proposto classificou incorretamente. O vídeo 1 foi classificado incorretamente como tráfego médio pois dos 165 vídeos de tráfego leve, este é o que mais se difere dos demais vídeos da categoria em relação à quantidade de carros e velocidade de tráfego. Como pode ser visto na Figura 4.7a, tal vídeo se parece bastante com alguns vídeos de tráfego médio. Além disso, nos 164 vídeos de tráfego leve, os carros trafegam em velocidade média de aproximadamente 96 km/h (quilômetros por hora) enquanto que no vídeo 1 a velocidade média dos carros é de aproximadamente 41 km/h, mostrando que a informação temporal pode ter contribuído para a classificação incorreta.

O método proposto classificou incorretamente os vídeos 44 e 45 de tráfego médio (Figuras 4.7b e 4.7c) como tráfego pesado. A condição climática pode ter influenciado na classificação, pois geralmente sob chuva os carros tendem a se movimentar mais devagar. E por consequência disso, a aparência (quantidade) e o movimento dos carros fazem com que o vídeo tenha características parecidas com alguns vídeos de tráfego pesado. Além disso, no vídeo 44, tanto a velocidade média (aproximadamente 27 km/h) quanto a quantidade de carros na rodovia

se assemelha aos vídeos de tráfego pesado, mostrando que ambas as informações espaço-temporais podem ter contribuído na classificação incorreta. No vídeo 45, apesar da velocidade média dos carros (aproximadamente 38 km/h) estar próxima da velocidade média de tráfego médio, a quantidade de carros deste vídeo também se assemelha aos vídeos de tráfego pesado, ou seja, a informação espacial também pode ter contribuído na classificação incorreta.

O vídeo 6 (Figura 4.7d) foi classificado incorretamente como tráfego médio. Apesar da velocidade média dos carros deste vídeo se assemelhar à velocidade média dos carros nos vídeos de tráfego pesado (aproximadamente $25km/h$), há certa similaridade com diversos vídeos de tráfego médio, como por exemplo, os vídeos 44 e 45 mostrado nas Figuras 4.7b e 4.7c.

(a) Leve - Vídeo 1 (b) Médio - Vídeo 44 (c) Médio - Vídeo 45 (d) Pesado - Vídeo 6

Figura 4.7: Exemplos de quadros extraídos de vídeos de tráfego de carros que o melhor resultado obtido pelo método proposto utilizando o descritor local PHOW, errou na classificação. Além do PHOW, o método proposto utilizou um vocabulário de palavras de tamanho $K = 2000$ e combinação de planos ortogonais XY-XT-YT. Em cada exemplo está descrito a categoria de tráfego a qual ele pertence e o seu número em relação a quantidade de vídeos de sua categoria.

Para a base de cenas dinâmicas, a Figura 4.8 apresenta quadros de alguns vídeos que foram classificados incorretamente pelo método proposto em seu melhor resultado. Os vídeos 6 e 17 de fonte (Figuras 4.8a e 4.8b) foram classificados incorretamente como cachoeira. Na maioria dos vídeos de cachoeira há uma paisagem estática ao fundo composta por árvores e plantas, em conjunto com a queda d'água que geralmente está presente ao longo do eixo Y do cubo tridimensional que representa os vídeos. Além disso, os cortes nos planos temporais XT e YT também caracterizam de forma parcial a aparência, ao longo dos eixos X e Y, ou seja, a informação espacial pode ter contribuído para a classificação incorreta dos vídeos 6 e 17. As informações de movimento dos vídeos de fonte podem ser parecidas com as dos vídeos de cachoeira e, portanto, também terem contribuído para a classificação incorreta desses vídeos.

O vídeo de relâmpago 14 foi classificado incorretamente como incêndio em floresta. De acordo com a Figura 4.8c, é possível notar que o vídeo ilustra a cena de uma floresta onde em um determinado momento ocorre o relâmpago. Por conta disso, pode ser que a informação espacial extraída deste vídeo seja parecida com a dos vídeos de incêndio em floresta, já que em tal categoria, o elemento dominante dos vídeos é a floresta. Além disso, o que pode ser observado nos vídeos de relâmpago é a sua frequência constante nos quadros e de forma geral, o elemento dominante (relâmpago) aparece em poucos quadros dos vídeos, em relação a sua quantidade total.

O método proposto classificou incorretamente os vídeos 11 e 28 de cachoeira (Figuras 4.8d e 4.8e) como correnteza de rio. Apesar do elemento dominante (água em movimento) estar presente nos vídeos, o comprimento da queda d'água faz com que se pareçam com os vídeos de correnteza de rio. Como em ambas as categorias há água em movimento sob uma paisagem estática composta por árvores e plantas, as informações temporais e espaciais dos vídeos de ambas as categorias podem ser parecidas.

(a) Fonte - Vídeo 6 (b) Fonte - Vídeo 17 (c) Relâmpago - Vídeo 14

(d) Cachoeira - Vídeo 11 (e) Cachoeira - Vídeo 28

Figura 4.8: Exemplos de quadros extraídos de vídeos de cenas dinâmicas que o melhor resultado obtido pelo método proposto utilizando o descritor local PHOW, errou na classificação. Além do PHOW, o método proposto utilizou um vocabulário de palavras de tamanho $K = 1000$ e combinação de planos ortogonais XY-XT-YT. Em cada exemplo está descrito a categoria de cena a qual ele pertence e o seu número em relação a quantidade de vídeos de sua categoria.

4.2.4 Comparação com Literatura

A Tabela 4.1 apresenta um comparativo da taxa de classificação correta por classe dos vídeos de tráfego de carros, entre o método proposto utilizando o SIFT, SIFT Denso e PHOW como descritores locais; o BoVW utilizando o SIFT 3D como descritor local e; os métodos de textura dinâmica.

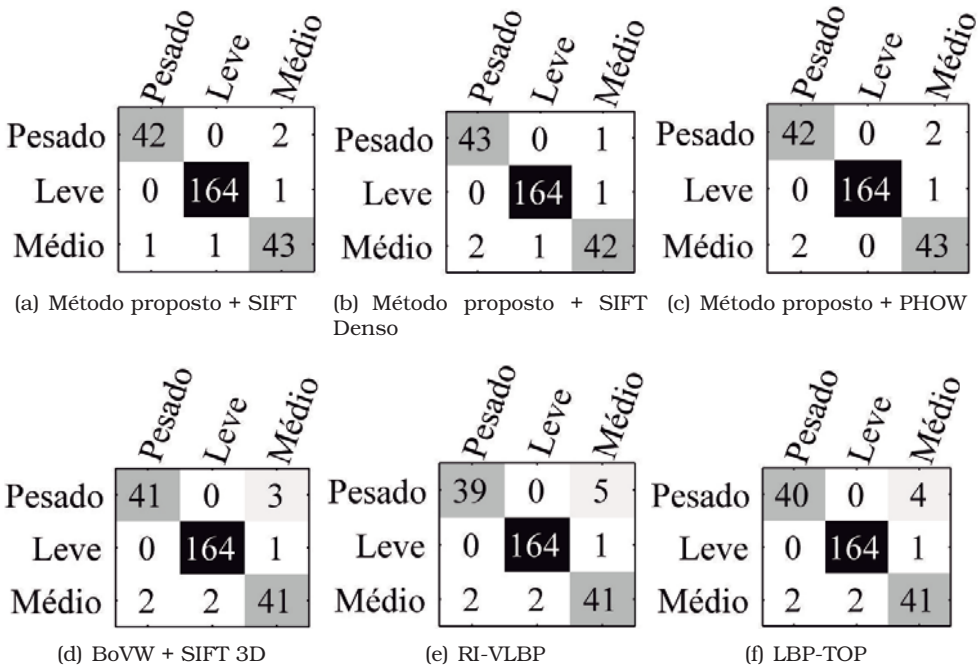
Tabela 4.1: Classificação por classe - Vídeos de tráfego de carros

Classe	Método proposto + SIFT	Método proposto + SIFT Denso	Método proposto + PHOW	BoVW + SIFT 3D	RI-VLBP	LBP-TOP
Pesado	95.45	**97.72**	95.45	93.18	88.63	90.90
Leve	**99.39**	**99.39**	**99.39**	**99.39**	**99.39**	**99.39**
Médio	**95.55**	93.33	**95.55**	91.11	91.11	91.11
Média	96.79	**96.81**	96.79	94.56	93.04	93.80

De acordo com a Tabela, dentre os métodos de textura dinâmica, o LBP-TOP foi o que se destacou. Embora todas as abordagens tenham obtido a mesma média de classificação correta para a categoria de tráfego leve, o método proposto utilizando o PHOW como descritor local obteve média de classificação correta superior ao BoVW utilizando SIFT 3D e ao LBP-TOP. Em contrapartida, o método proposto utilizando o SIFT Denso obteve melhor média de classificação correta, por ter classificado incorretamente apenas um vídeo de tráfego pesado. Além disso, o método proposto classificou incorretamente o mesmo vídeo de tráfego leve utilizando os três descritores locais, acarretando em mesma média de classificação correta dos vídeos desta categoria para as três abordagens.

A Figura 4.9 apresenta as matrizes de confusão obtidas através dos melhores resultados de cada descritor utilizado pelo método proposto e os métodos da literatura, na base de tráfego de carros. Nestas Figuras, a técnica com bom desempenho é representada por uma matriz com a maioria dos seus valores na diagonal principal e poucos valores fora dela. Com base nessas matrizes, é possível concluir que existe uma maior confusão entre os vídeos de tráfego pesado com tráfego médio e vice-versa. Vale a pena ressaltar que o método proposto usando o PHOW não gerou confusão entre os vídeos de tráfego médio com tráfego leve.

A Tabela 4.2 apresenta as taxas de classificações corretas por classe para os vídeos de



(a) Método proposto + SIFT (b) Método proposto + SIFT Denso (c) Método proposto + PHOW

(d) BoVW + SIFT 3D (e) RI-VLBP (f) LBP-TOP

Figura 4.9: Matrizes de confusão do SIFT, SIFT Denso, PHOW, SIFT 3D, RI-VLBP e LBP-TOP, obtidas pelo classificador *SVM*, na base de vídeos de tráfego de carros.

cenas dinâmicas, obtidas com: o método proposto utilizando os descritores aplicados nos planos ortogonais; BoVW utilizando SIFT 3D; RI-VLBP e; LBP-TOP. De forma similar à base de tráfego de carros, dentre as técnicas de textura dinâmica, o LBP-TOP foi o que se destacou, pois seus descritores acarretaram em maior média de classificação correta que os obtidos com o RI-VLBP.

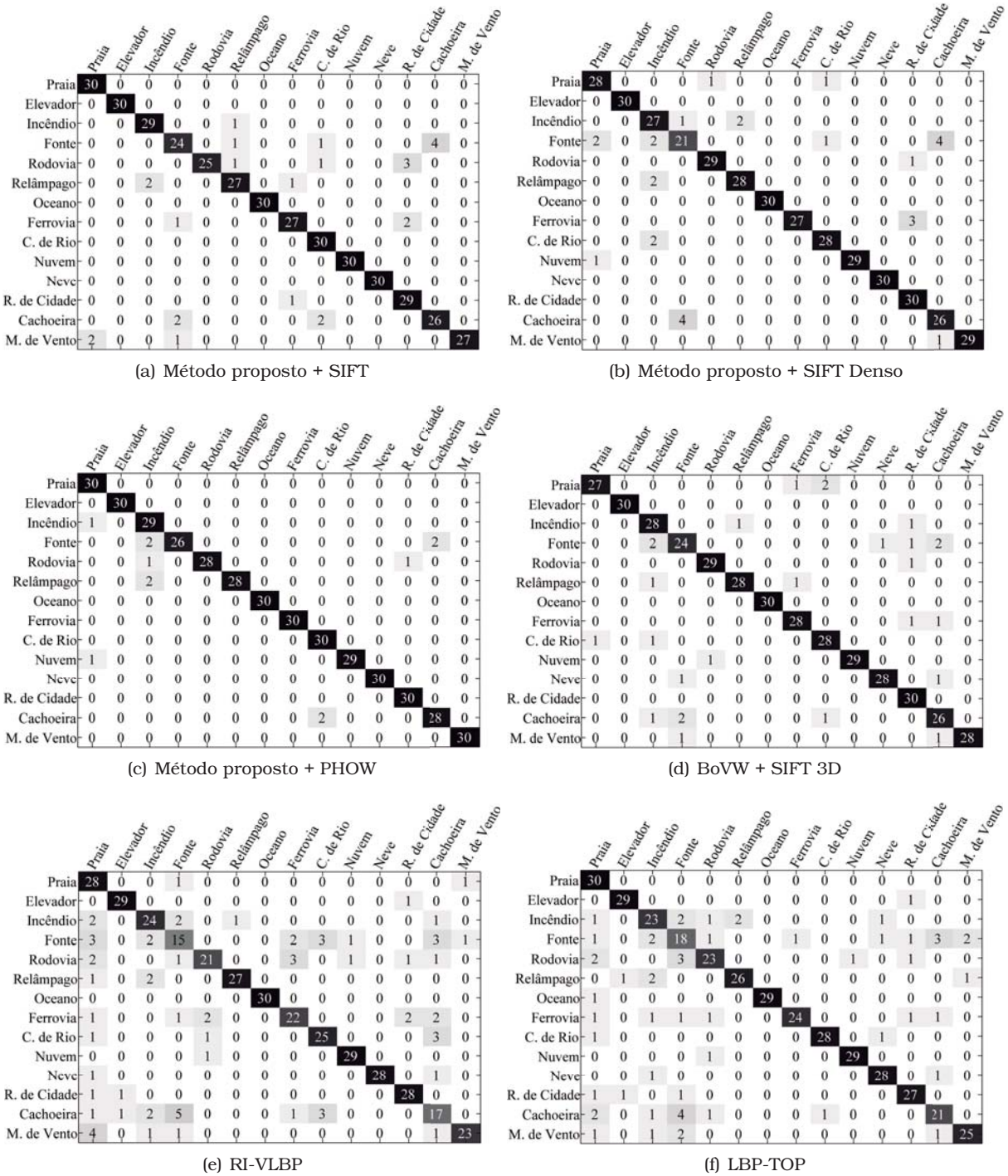
Tabela 4.2: Classificação por classe - Vídeos de cenas dinâmicas

Classe	Método proposto + SIFT	Método proposto + SIFT Denso	Método proposto + PHOW	BoVW + SIFT 3D	RI-VLBP	LBP-TOP
Praia	**100.00**	93.33	**100.00**	90.00	93.33	**100.00**
Elevador	**100.00**	**100.00**	**100.00**	**100.00**	96.66	96.66
Incêndio	**96.66**	90.00	**96.66**	93.33	80.00	76.66
Fonte	80.00	70.00	**86.66**	80.00	50.00	60.00
Rodovia	83.33	**96.66**	93.33	**96.66**	70.00	76.66
Relâmpago	90.00	**93.33**	**93.33**	**93.33**	90.00	86.66
Oceano	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	96.66
Ferrovia	90.00	90.00	**100.00**	93.33	73.33	80.00
C. de Rio	**100.00**	93.33	**100.00**	93.33	83.33	93.33
Nuvem	**100.00**	96.66	96.66	96.66	96.66	96.66
Neve	**100.00**	**100.00**	**100.00**	93.33	93.33	93.33
R. de Cidade	96.66	**100.00**	**100.00**	**100.00**	93.33	90.00
Cachoeira	86.66	86.66	**93.33**	86.66	56.66	70.00
M. de Vento	90.00	96.66	**100.00**	93.33	76.66	83.33
Média	93.80	93.33	**97.14**	93.57	82.38	85.71

De acordo com a Tabela, pode-se observar que as médias de classificação são semelhantes às taxas de classificações corretas que foram discutidas na Subseção 4.2.1 e apresentadas pela Figura 4.4. Isso se deve ao fato de que, ao contrário da base de vídeos de tráfego de carros, todas as classes dessa base possuem a mesma quantidade de vídeos.

Apesar do BoVW com o SIFT 3D ter acarretado maior média de classificação correta (93.57%) do que a obtida com o LBP-TOP (85.71%), o método proposto utilizando o descritor local PHOW

foi superior com 97.14%, pois dentre os três, se destacou na classificação das categorias: incêndio em floresta, fonte, ferrovia, correnteza de rio, neve, cachoeira e moinho de vento; enquanto que o BoVW com o SIFT 3D se destacou apenas na classificação dos vídeos de rodovia. De acordo com as matrizes de confusão da Figura 4.10 é possível observar que de forma geral, houve maior confusão entre os vídeos de cachoeira, correnteza de rio e fonte.



(a) Método proposto + SIFT

(b) Método proposto + SIFT Denso

(c) Método proposto + PHOW

(d) BoVW + SIFT 3D

(e) RI-VLBP

(f) LBP-TOP

Figura 4.10: Matrizes de confusão do SIFT, SIFT Denso, PHOW, SIFT 3D, RI-VLBP e LBP-TOP, obtidas pelo classificador *SVM*, na base de vídeos de cenas dinâmicas.

Estado da Arte

Especificamente na base de vídeos de cenas dinâmicas, o melhor resultado obtido com a proposta deste trabalho foi comparado com os resultados obtidos por outros métodos da

literatura: histograma de fluxo (HOF) [10] + GIST [28], características dinâmicas caóticas (Chaos) [35] + GIST, energias espaço-temporais (SOE) [11], análise de características lentas (SFA) [38], orientação espaço-temporal complementar (CSO) [16] e histogramas de energias espaço-temporais (BoSE) [15].

Tabela 4.3: Classificação por classe - Métodos da literatura x método proposto - Vídeos de cenas dinâmicas

Classe	Chaos + GIST	HOF + GIST	SOE	SFA	CSO	BoSE	Método proposto
Praia	30.00	87.00	93.00	93.00	**100.00**	**100.00**	**100.00**
Elevador	47.00	87.00	**100.00**	97.00	**100.00**	97.00	**100.00**
Incêndio	17.00	63.00	67.00	70.00	83.00	93.00	**96.66**
Fonte	3.00	43.00	43.00	57.00	47.00	**87.00**	86.66
Rodovia	23.00	47.00	70.00	93.00	73.00	**100.00**	93.33
Relâmpago	37.00	63.00	77.00	87.00	93.00	**97.00**	93.33
Oceano	43.00	97.00	**100.00**	**100.00**	90.00	**100.00**	**100.00**
Ferrovia	7.00	83.00	80.00	93.00	93.00	**100.00**	**100.00**
C. de Rio	10.00	77.00	93.00	87.00	97.00	97.00	**100.00**
Nuvem	47.00	87.00	83.00	93.00	**100.00**	97.00	96.66
Neve	10.00	47.00	87.00	70.00	57.00	97.00	**100.00**
R. de Cidade	17.00	77.00	90.00	97.00	97.00	**100.00**	**100.00**
Cachoeira	10.00	47.00	63.00	73.00	77.00	83.00	**93.33**
M. de Vento	17.00	53.00	83.00	87.00	93.00	**100.00**	**100.00**
Média	22.86	68.33	80.71	85.48	85.95	96.19	**97.14**

A Tabela 4.3 apresenta a taxa de classificação correta por classe de cada método da literatura supracitado, seguido de sua respectiva média de taxa de classificação correta. Embora a comparação entre os resultados obtidos pelas técnicas não possa ser de forma direta, pode-se observar que o método proposto neste trabalho conseguiu superar os resultados obtidos por outros métodos da literatura, na base vídeos de cenas dinâmicas. Das 14 categorias da base de cenas, o método proposto foi inferior aos resultados da literatura apenas nas categorias fonte, rodovia, relâmpago e nuvem, alcançando uma taxa de classificação correta de 97.14% contra 96.19% do BoSE.

CAPÍTULO

5

Conclusões e Trabalhos Futuros

Este trabalho apresentou um novo método para caracterização de vídeos de textura dinâmica. Os descritores locais SIFT, SIFT Denso e PHOW foram utilizados para detectar e descrever pontos de interesse em três planos ortogonais dos vídeos: XY, XT e YT. Em seguida, o BoVW foi utilizado para combinar os pontos de interesse de cada plano em um histograma de palavras visuais, devido a sua popularidade na comunidade de visão computacional. Um histograma para cada plano ortogonal foi obtido e as características extraídas foram avaliadas pelo classificador SVM.

Para isso, o método proposto foi aplicado em duas bases de vídeos: a primeira contendo vídeos de tráfego de carros e a segunda, vídeos de cenas dinâmicas. Os resultados do método proposto foram comparados aos obtidos por métodos da literatura: BoVW utilizando o descritor local SIFT 3D, RI-VLBP e LBP-TOP. O método proposto utilizando o descritor local PHOW se destacou em ambas as bases de vídeos. Na base de tráfego de carros é possível concluir que a informação de aparência fornecida através do plano XY permite discriminar de forma parcial a categoria de tráfego dos vídeos. Porém a informação de movimento fornecida pelos planos temporais XT e YT também contribui na classificação. Nesta base de vídeos, o método proposto com o descritor local PHOW alcançou 98.03% de taxa de classificação correta, contra 94.56% do BoVW com o SIFT 3D e 93.80% do LBP-TOP.

Na base de cenas dinâmicas foi possível concluir que apesar da informação espacial contribuir de certa forma, a informação temporal remetida através dos planos XT e YT contribuiu consideravelmente para a caracterização dos vídeos desta base. O método proposto com o PHOW alcançou 97.14% contra 93.57% do BoVW com SIFT 3D e 85.71% do LBP-TOP. Além disso, o resultado do método proposto foi comparado com resultados obtidos por métodos da literatura, mostrando que o método proposto conseguiu superar os 96.19% alcançados com o histograma de energias espaço-temporais (BoSE).

Como proposta de trabalhos futuros, podemos elencar:

- Aplicação do método proposto em outra base de texturas dinâmicas, onde a câmera está em movimento;
- Identificar se as características dos descritores influenciaram na classificação dos vídeos (por exemplo: invariância à escala, iluminação, rotação);

- Verificar o desempenho do método proposto utilizando as versões do descritor local PHOW para imagens coloridas (PHOW-RSV, PHOW-RGB, PHOW-OPPONENT);
- Identificar se tamanhos diferentes de vocabulários para cada plano ortogonal utilizado em determinada combinação influencia na classificação dos vídeos;
- Identificar pontos de interesse que são realmente relevantes para a discriminar as categorias de vídeo, para a base de cenas dinâmicas, por exemplo, diminuindo assim o tamanho dos descritores obtidos;
- Aplicar o método proposto em problemas reais, como por exemplo, identificação do tipo de corrosão de um material.

Referências Bibliográficas

[1] Lamberto Ballan, Marco Bertini, Alberto Bimbo, and Giuseppe Serra. Video event classification using bag of words and string kernels. In *Proceedings of the 15th International Conference on Image Analysis and Processing*, pages 170–178. Springer-Verlag, 2009. Citado na página 4.

[2] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *Computer vision–ECCV 2006*, pages 404–417. Springer, 2006. Citado na página 1.

[3] Anna Bosch, Andrew Zisserman, and Xavier Muoz. Image classification using random forests and ferns. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE, 2007. Citado nas páginas 1, 16, e 32.

[4] A.B. Chan and N. Vasconcelos. Classification and retrieval of traffic video using auto-regressive stochastic processes. In *Intelligent Vehicles Symposium, 2005. Proceedings. IEEE*, pages 771–776, June 2005. Citado nas páginas xiii, 1, 3, e 32.

[5] A.B. Chan and N. Vasconcelos. Modeling, clustering, and segmenting video with mixtures of dynamic textures. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(5):909–926, May 2008. Citado na página 6.

[6] Jie Chen, Guoying Zhao, M. Salo, E. Rahtu, and M. Pietikainen. Automatic dynamic texture segmentation using local descriptors and optical flow. *Image Processing, IEEE Transactions on*, 22(1):326–339, Jan 2013. Citado na página 5.

[7] Dmitry Chetverikov and Renaud Péteri. A brief survey of dynamic texture description and recognition. In Marek Kurzyński, Edward Puchała, Michał Woźniak, and Andrzej żołnierek, editors, *Computer Recognition Systems*, volume 30 of *Advances in Soft Computing*, pages 17–26. Springer Berlin Heidelberg, 2005. Citado na página 2.

[8] JamesL. Crowley and Olivier Riff. Fast computation of scale normalised gaussian receptive fields. In LewisD. Griffin and Martin Lillholm, editors, *Scale Space Methods in Computer Vision*, volume 2695 of *Lecture Notes in Computer Science*, pages 584–598. Springer Berlin Heidelberg, 2003. Citado na página 9.

[9] Gabriella Csurka, Christopher Dance, Lixin Fan, Jutta Willamowski, and Cédric Bray. Visual categorization with bags of keypoints. In *Workshop on statistical learning in computer vision, ECCV*, volume 1, pages 1–2. Prague, 2004. Citado nas páginas 17 e 18.

[10] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893. IEEE, 2005. Citado nas páginas 1 e 44.

[11] K.G. Derpanis, M. Lecce, K. Daniilidis, and R.P. Wildes. Dynamic scene understanding: The role of orientation features in space and time in scene classification. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1306–1313, June 2012. Citado nas páginas xi, xii, xiii, 2, 3, 28, 32, 33, e 44.

[12] Sloven Dubois, Renaud Péteri, and Michel Ménard. A comparison of wavelet based spatio-temporal decomposition methods for dynamic texture recognition. In Helder Araujo, Ana-Maria Mendonça, ArmandoJ. Pinho, and MaríaInés Torres, editors, *Pattern Recognition and Image Analysis*, volume 5524 of *Lecture Notes in Computer Science*, pages 314–321. Springer Berlin Heidelberg, 2009. Citado na página 6.

[13] Sándor Fazekas and Dmitry Chetverikov. Analysis and performance evaluation of optical flow features for dynamic texture recognition. *Signal Processing: Image Communication*, 22(7à8):680–691, 2007. Citado na página 5.

[14] L. Fei-Fei and P. Perona. A bayesian hierarchical model for learning natural scene categories. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 524–531 vol. 2, June 2005. Citado na página 17.

[15] C. Feichtenhofer, A. Pinz, and R. P. Wildes. Bags of spacetime energies for dynamic scene recognition. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2681–2688, June 2014. Citado nas páginas 33 e 44.

[16] Christoph Feichtenhofer, Axel Pinz, and Richard P Wildes. Spacetime forests with complementary features for dynamic scene recognition. In *BMVC*, 2013. Citado nas páginas 33 e 44.

[17] Tamar Glaser and Lihi Zelnik-Manor. Incorporating temporal context in bag-of-words models. In *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, pages 1562–1569. IEEE, 2011. Citado na página 5.

[18] Wesley Nunes Gonçalves. Análise de texturas estáticas e dinâmicas e suas aplicações em biologia e nanotecnologia. tese (doutorado em física aplicada), 2013. Disponível em: <http://www.teses.usp.br/teses/disponiveis/76/76132/tde-25092013-093513/>, Acesso em: 2015-08-20. Citado nas páginas 2 e 5.

[19] Wesley Nunes Gonçalves and Odemir Martinez Bruno. Dynamic texture analysis and segmentation using deterministic partially self-avoiding walks. *Expert Systems with Applications*, 40(11):4283 – 4300, 2013. Citado na página 2.

[20] Wesley Nunes Gonçalves, Bruno Brandoli Machado, and Odemir Martinez Bruno. Spatiotemporal gabor filters: a new method for dynamic texture recognition. *arXiv preprint arXiv:1201.3612*, 2012. Citado na página 6.

[21] Adilson Gonzaga and Leonardo Augusto Oliveira. Localização, segmentação e reconhecimento de caracteres em placas de automóveis. In *Avanços em Visão Computacional*, pages 283–302, Curitiba, PR, 2012. Omnipax. 406 p. Citado na página 1.

[22] Giancarlo Luis Gómez Gonzáles. *Aplicação da Técnica SIFT para Determinação de Campos de Deformações de Materiais usando Visão Computacional*. PhD thesis, PUC-Rio, 2010. Citado na página 9.

[23] Marti A. Hearst, ST Dumais, E Osman, John Platt, and Bernhard Scholkopf. Support vector machines. *Intelligent Systems and their Applications, IEEE*, 13(4):18–28, 1998. Citado na página 33.

[24] Ce Liu, Jenny Yuen, and Antonio Torralba. Sift flow: Dense correspondence across scenes and its applications. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(5):978–994, 2011. Citado nas páginas 1, 16, e 32.

[25] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004. Citado nas páginas xi, 1, 8, 9, 12, 13, 18, e 32.

[26] Zongqing Lu, Weixin Xie, Jihong Pei, and JianJun Huang. Dynamic texture recognition by spatio-temporal multiresolution histograms. In *Application of Computer Vision, 2005. WACV/MOTIONS '05 Volume 1. Seventh IEEE Workshops on*, volume 2, pages 241–246, Jan 2005. Citado na página 5.

[27] Timo Ojala, Matti Pietikainen, and Topi Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(7):971–987, 2002. Citado nas páginas 1 e 19.

[28] Aude Oliva and Antonio Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42(3):145–175, 2001. Citado na página 44.

[29] Jonatan Patrick Margarido Oruê, Rillian Diello Lucas Pires, Wesley Eiji Sanches Kanashiro, Wesley Nunes Gonçalves, Bruno Brandoli Machado, and Mauro dos Santos Arruda. Identification of foliar soybean diseases using local descriptors. In *XI Workshop de Visão Computacional*, pages 242–247, 2015. Citado na página 1.

[30] K. Otsuka, T. Horikoshi, S. Suzuki, and M. Fujii. Feature extraction of temporal texture based on spatiotemporal motion trajectory. In *Pattern Recognition, 1998. Proceedings. Fourteenth International Conference on*, volume 2, pages 1047–1051 vol.2, Aug 1998. Citado na página 6.

[31] Xianbiao Qi, Chun-Guang Li, Guoying Zhao, Xiaopeng Hong, and Matti Pietikäinen. Dynamic texture and scene classification by transferring deep image features. *Neurocomputing*, 171:1230–1241, 2016. Citado na página 2.

[32] A. Ravichandran, R. Chaudhry, and R. Vidal. Categorizing dynamic textures using a bag of dynamical systems. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(2):342–353, Feb 2013. Citado na página 6.

[33] Paul Scovanner, Saad Ali, and Mubarak Shah. A 3-dimensional sift descriptor and its application to action recognition. In *Proceedings of the 15th International Conference on Multimedia*, MULTIMEDIA '07, pages 357–360, New York, NY, USA, 2007. ACM. Citado nas páginas 2, 4, e 14.

[34] S. Z. Selim and M. A. Ismail. K-means-type algorithms: A generalized convergence theorem and characterization of local optimality. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6(1):81–87, Jan 1984. Citado na página 18.

[35] N. Shroff, P. Turaga, and R. Chellappa. Moving vistas: Exploiting motion for describing scenes. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1911–1918, June 2010. Citado nas páginas 33 e 44.

[36] Robson de Carvalho Soares. Extensão do bag-of-visual-features para incorporar informação espacial na descrição de características de acordo com a percepção visual humana. 2012. Citado na página 18.

[37] Sabin Tiberius Strat, Alexandre Benoit, and Patrick Lambert. Retina enhanced bag of words descriptors for video classification. In *Signal Processing Conference (EUSIPCO), 2013 Proceedings of the 22nd European*, pages 1307–1311. IEEE, 2014. Citado na página 5.

[38] C. Thériault, N. Thome, and M. Cord. Dynamic scene classification: Learning motion descriptors with slow features analysis. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 2603–2610, June 2013. Citado nas páginas 33 e 44.

[39] Muhammad Muneeb Ullah, Sobhan Naderi Parizi, and Ivan Laptev. Improving bag-of-features action recognition with non-local cues. In *BMVC*, volume 10, pages 95–1. Citeseer, 2010. Citado na página 4.

[40] Hanna M. Wallach. Topic modeling: Beyond bag-of-words. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, pages 977–984, New York, NY, USA, 2006. ACM. Citado na página 17.

[41] Albert da Costa Xavier, João Ricardo Sato, Gilson Antônio Giraldi, Paulo Sérgio Rodrigues, and Carlos Eduardo Thomaz. Classificação e extração de características discriminantes de imagens 2d de ultrasonografia mamária. In *Avanços em Visão Computacional*, pages 65–84, Curitiba, PR, 2012. Omnipax. 406 p. Citado na página 1.

[42] Jun Yang, Yu-Gang Jiang, Alexander G Hauptmann, and Chong-Wah Ngo. Evaluating bag-of-visual-words representations in scene classification. In *Proceedings of the international workshop on Workshop on multimedia information retrieval*, pages 197–206. ACM, 2007. Citado nas páginas 1 e 17.

[43] Guoying Zhao and M. Pietikainen. Dynamic texture recognition using local binary patterns with an application to facial expressions. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(6):915–928, June 2007. Citado nas páginas 1, 6, 25, 27, e 32.

[44] Guoying Zhao and Matti Pietikäinen. Improving rotation invariance of the volume local binary pattern operator. In *MVA*, pages 327–330, 2007. Citado nas páginas 6, 21, 23, e 32.

[45] Alexsandro Mendes Zimer, Emerson Costa Rios, Paulo de Carvalho Dias Mendes, Wesley Nunes Gonçalves, Odemir Martinez Bruno, Ernesto Chaves Pereira, and Lucia Helena Mascaro. Investigation of {AISI} 1040 steel corrosion in {H2S} solution containing chloride ions by digital image processing coupled with electrochemical techniques. *Corrosion Science*, 53(10):3193 – 3201, 2011. Citado na página 1.