Eder de Almeida Perez

Descritor de movimento baseado em tensor e histograma de gradientes

Dissertação apresentada ao Programa de Pós-graduação em Modelagem Computacional, da Universidade Federal de Juiz de Fora como requisito parcial à obtenção do grau de Mestre em Modelagem Computacional.

Orientador: Prof. D.Sc. Marcelo Bernardes Vieira

Juiz de Fora

2012

Perez, Eder de Almeida.
Descritor de movimento baseado em tensor e histograma de gradientes / Eder de Almeida Perez. – 2012.
61 f. : il.

Dissertação (Mestrado em Modelagem Computacional)–Universidade Federal de Juiz de Fora, Juiz de Fora, 2012.

1.Ciência da computação. 2. Inteligência artificial. 3. Tensores. 4. Visão computacional. 4. Aprendizagem. I. Título.

CDU 681.3

Eder de Almeida Perez

Descritor de movimento baseado em tensor e histograma de gradientes

Dissertação apresentada ao Programa de Pós-graduação em Modelagem Computacional, da Universidade Federal de Juiz de Fora como requisito parcial à obtenção do grau de Mestre em Modelagem Computacional.

Aprovada em 24 de Agosto de 2012.

BANCA EXAMINADORA

Prof. D.Sc. Marcelo Bernardes Vieira - Orientador
Universidade Federal de Juiz de Fora

Prof. D.Sc. Esteban Walter Gonzalez Clua
Universidade Federal Fluminense

Prof. D.Sc. Carlos Cristiano Hasenclever Borges
Universidade Federal de Juiz de Fora

Dedico este trabalho à minha esposa Natália, aos meus pais, irmã e amigos.

AGRADECIMENTOS

Agradeço primeiramente à minha esposa Natália pelo apoio incondicional durante todo mestrado e por ter sacrificado alguns finais de semana em prol da ciência. Aos meus pais e à minha irmã por estarem ao meu lado e permitirem que eu chegasse até aqui.

À minha sogra e meu sogro que sempre torceram pelo meu sucesso e sempre me incentivaram nos momentos difíceis.

Aos meus grandes amigos Peçanha, Tássio, Thales e Scoralick que são verdadeiros irmãos com quem eu sempre posso contar.

Ao meu orientador Marcelo Bernardes por todo ensinamento me dado durante esses longos anos de GCG.

À futura doutora Virgínia Mota pelo apoio nesse trabalho e nas publicações oriundas dele.

À ENSEA-UCP pelo ambiente RETIN SVM sem o qual esse trabalho não seria possível.

À UFJF e ao Grupo de Computação Gráfica onde eu tive a oportunidade de adquirir grande conhecimento e foi responsável pela minha formação profissional.

Agradeço também ao Luiz Maurílio pela enorme ajuda nos testes experimentais e a todos os membros do GCG pelos momentos de trabalho e diversão.

Aos membros da banca por terem aceitado o convite e por suas contribuições.

Aos professores do Mestrado em Modelagem Computacional e aos companheiros de turma.

À CAPES pelo suporte financeiro.

"Ser é ser percebido"

George Berkeley

RESUMO

O reconhecimento de padrões de movimentos tem se tornado um campo de pesquisa muito atrativo nos últimos anos devido, entre outros fatores, à grande massificação de dados em vídeos e a tendência na criação de interfaces homem-máquina que utilizam expressões faciais e corporais. Esse campo pode ser considerado um dos requisitos chave para análise e entendimento de vídeos.

Neste trabalho é proposto um descritor de movimentos baseado em tensores de 2ª ordem e histogramas de gradientes (HOG - *Histogram of Oriented Gradients*). O cálculo do descritor é rápido, simples e eficaz. Além disso, nenhum aprendizado prévio é necessário sendo que a adição de novas classes de movimentos ou novos vídeos não necessita de mudanças ou que se recalculem os descritores já existentes. Cada quadro do vídeo é particionado e em cada partição calcula-se o histograma de gradientes no espaço e no tempo. A partir daí calcula-se o tensor do quadro e o descritor final é formado por uma série de tensores de cada quadro.

O descritor criado é avaliado classificando-se as bases de vídeos KTH e Hollywood2, utilizadas na literatura atual, com um classificador Máquina Vetor Suporte (SVM). Os resultados obtidos na base KTH são próximos aos descritores do estado da arte que utilizam informação local do vídeo. Os resultados obtidos na base Hollywood2 não superam o estado da arte, mas são próximos o suficiente para concluirmos que o método proposto é eficaz. Apesar de a literatura apresentar descritores que possuem resultados superiores na classificação, suas abordagens são complexas e de alto custo computacional.

Palavras-chave: Descritor de movimento. Tensor de 2ª ordem. Série de tensores. SVM. Histograma de gradientes. Modelagem do movimento.

ABSTRACT

The motion pattern recognition has become a very attractive research field in recent years due to the large amount of video data and the creation of human-machine interfaces that use facial and body expressions. This field can be considered one of the key requirements for analysis and understanding in video.

This thesis proposes a motion descriptor based on second order tensor and histograms of oriented gradients. The calculation of the descriptor is fast, simple and effective. Furthermore, no prior knowledge of data basis is required and the addition of new classes of motion and videos do not need to recalculate the existing descriptors. The frame of a video is divided into a grid and the histogram of oriented gradients is computed in each cell. After that, the frame tensor is computed and the final descriptor is built by a series of frame tensors.

The descriptor is evaluated in both KTH and Hollywood2 data basis, used in the current literature, with a Support Vector Machine classifier (SVM). The results obtained on the basis KTH are very close to the descriptors of the *state-of-the-art* that use local information of the video. The results obtained on the basis Hollywood2 not outweigh the *state-of-the-art* but are close enough to conclude that the proposed method is effective. Although the literature presents descriptors that have superior results, their approaches are complex and with computational cost.

Keywords: Motion descriptor. Second order tensor. Series of tensors. SVM. Histogram of oriented gradients. Motion modeling.

SUMÁRIO

1 INTRODUÇÃO 11
1.1 Definição do problema 13
1.2 Objetivos 13
1.3 Contribuições e Publicações 14
1.4 Trabalhos relacionados 14
1.4.1 Descritores baseados em histogramas de gradientes 14
1.4.2 Descritores baseados em tensores 15
1.4.3 Descritores globais 17

2 FUNDAMENTOS 18
2.1 Máquina Vetor Suporte 18
2.2 SIFT 21
2.3 Histograma de gradientes 23
2.4 Tensor de 2ª ordem 24

3 DESCRITOR DE MOVIMENTO PROPOSTO 26
3.1 Gradiente espaço-temporal 28
3.2 Particionamento do quadro e histograma de gradientes 28
3.3 Criação dos tensores de 2ª ordem 29
3.3.1 Tensor de um quadro 29
3.3.2 Tensor final de um vídeo 30
3.3.3 Reflexão do tensor para captura de simetrias 31
3.4 Minimizando o efeito da variação de brilho 31

4 RESULTADOS E ANÁLISE COMPARATIVA 33
4.1 Base de vídeos 33
4.2 Resultados na base KTH 36
4.2.1 Reflexão do quadro para o cálculo do histograma 38
4.2.2 Usando limiarização da norma 39
4.2.3 Combinando limiarização e reflexão 40

4.2.4 Efeito do uso da função gaussiana na ponderação dos gradientes das partições 43
4.3 Resultados na base Hollywood2 45
4.3.1 Reflexão do quadro para o cálculo do histograma 47
4.3.2 Efeito do uso da função gaussiana na ponderação dos gradientes das partições 51
4.4 Comparação com descritores da literatura 53

5 CONCLUSÃO 55

REFERÊNCIAS 58

1 INTRODUÇÃO

Um dos primeiros estudos sobre a natureza do movimento foi feito pelo cientista francês Étienne-Jules Marey no século XIX. Sua ideia original foi registrar as várias etapas do movimento em uma única fotografia (Figura 1.1). Essas fotografias eram tiradas em um instrumento conhecido como fuzil cronofotográfico, capaz de produzir 12 quadros consecutivos em uma única imagem. Esses estudos revelaram aspectos interessantes na locomoção de animais e seres humanos [1].

Na década de 70, o cientista Gunnar Johansson realizou um experimento que consistia na colocação de pontos refletores de luz dispostos nas juntas de um modelo humano cujos movimentos eram capturados por uma câmera de vídeo [2]. Através desse experimento, conhecido como MLD (*Moving Light Display*), ele foi capaz de realizar estudos a respeito da percepção visual de padrões de movimentos. O trabalho de Johansson despertou grande interesse da neurociência no estudo e análise da percepção do movimento [1], abrindo caminho para a modelagem matemática de movimentos e reconhecimento automático que, naturalmente, envolve o campo da visão computacional e reconhecimento de padrões.

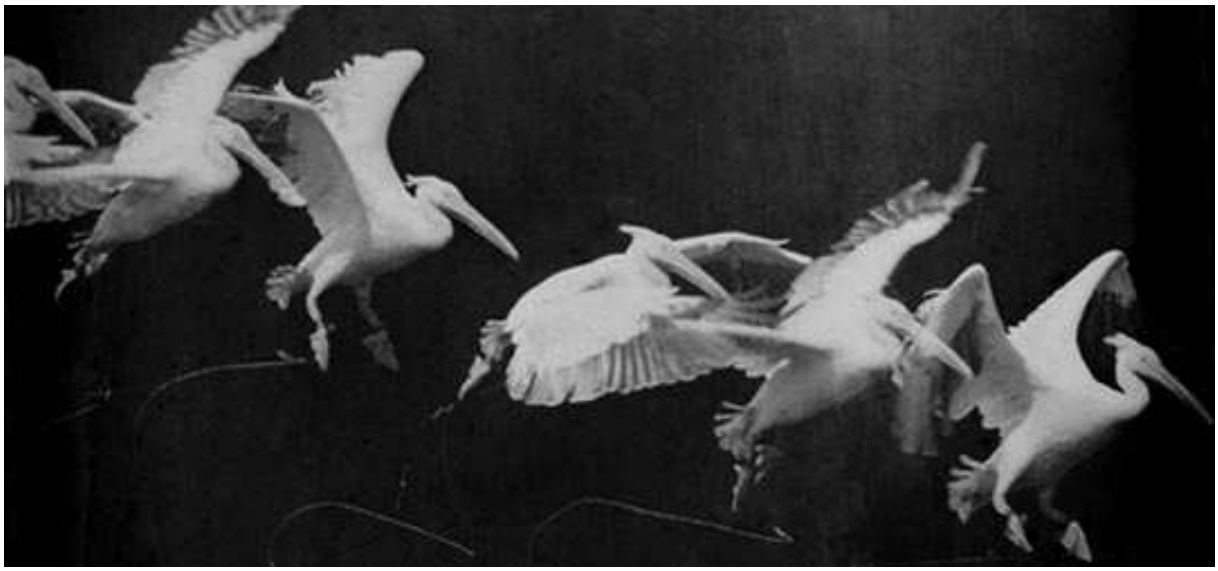


Figura 1.1: Voo de um pelicano. Foto tirada por Étienne-Jules Marey por volta de 1882 (*domínio público*).

O avanço tecnológico nos dispositivos de captura de imagem e vídeo e a popularização

de sites de compartilhamento deste tipo de mídia na internet, fez com que a pesquisa em reconhecimento de movimentos crescesse muito nos últimos anos. Algumas áreas de aplicação são [1]:

- Biometria Comportamental: A biometria envolve o reconhecimento de pessoas através de características fisiológicas como íris e impressões digitais. Mais recentemente, características comportamentais como o modo de agir e se movimentar tem atraído grande interesse nessa área. Diferentemente das características fisiológicas, é possível capturar informações que identificam um indivíduo sem a necessidade de interação com o mesmo ou interrompendo suas atividades. Com isso, o reconhecimento de movimentos em vídeos desempenha papel fundamental nessa tarefa. [3]

- Análise de vídeo baseada em conteúdo: Existem hoje inúmeros sites de compartilhamento de vídeos na internet. A classificação e armazenagem dessas mídias necessitam de métodos eficientes para que seja possível fazer buscas rápidas e aumentar a experiência do usuário. Tudo isso requer o aprendizado de padrões em vídeos classificando-os a partir de seu conteúdo. [4] [5]

- Segurança e Vigilância: Sistemas de segurança e vigilância geralmente contam com diversas câmeras espalhadas em locais estratégicos e um ou mais operadores monitorando cada uma delas em busca de ações suspeitas. Quanto mais câmeras, mais suscetível às falhas humanas torna-se o sistema. Tais falhas podem ser minimizadas através de sistemas de visão capazes de reconhecer ações suspeitas de maneira automática. [6] [7]

- Aplicações Interativas e Ambientes: A interação entre humanos e computadores através de comunicação visual é um grande desafio no projeto de interfaces homem-máquina. O reconhecimento eficiente de gestos e expressões faciais pode ajudar a criar computadores que interagem de forma fácil e rápida com pessoas. [8]

- Animação e síntese: A indústria de jogos e cinema faz uso intenso de sistemas de captura para síntese realística de movimentos em modelos tridimensionais. O avanço dos algoritmos e hardware torna a síntese de movimentos cada vez mais realista [9].

Antes de partirmos para definição do presente problema, faremos aqui algumas definições básicas:

Definição 1.0.1 (Imagem)**.** *Uma imagem I pode ser definida como uma função (Gomes e Velho [10]):*

$$I : U \subset \Re^2 \rightarrow \Re^n,$$

onde U é um conjunto suporte, ou seja, uma região onde a função toma valores e $\Re^n$ é o espaço de cores associado a cada ponto da imagem.

Definição 1.0.2 (Vídeo)**.** *Um vídeo s nada mais é do que uma sequência de imagens*[1]*:*

$$s : [U \subset \Re^2] \times \Re \rightarrow \Re^n,$$

que representa uma imagem I em um determinado tempo $t \in \Re$. Cada imagem em um vídeo é chamada de quadro.

1.1 Definição do problema

Dados vídeos s_1 e s_2 em um espaço de vídeos S, queremos encontrar uma função f

$$f : S \rightarrow \Re^m,$$

onde $\Re^m$ é um espaço euclidiano de descritores, tal que, se s_1 e s_2 contém movimentos similares, seus descritores são próximos segundo a norma euclidiana.

1.2 Objetivos

O objetivo deste trabalho é apresentar um descritor de movimentos em vídeos sem que nenhuma informação prévia ou aprendizado de uma base seja necessário. É primordial também que se utilizem poucos parâmetros e haja alto desempenho no tempo de cálculo dos descritores. A abordagem escolhida combina tensores de 2ª ordem e histogramas de gradientes na geração dos descritores utilizando informação de todo o quadro. Gradientes de imagens são bons estimadores de movimento. Eles representam a direção de máxima variação de brilho em um ponto da imagem, sendo usados, por exemplo, por

[1] Não estamos considerando aqui vídeos com áudio

diversos métodos para o cálculo do fluxo óptico [11]. Por outro lado, tensores são poderosas ferramentas matemáticas que vem sendo exploradas em diversas áreas da ciência. Tensores derivados dos gradientes na vizinhança de um ponto de uma imagem sintetizam suas direções predominantes, podendo-se explorar essa característica na descrição de movimentos.

Muitos trabalhos calculam pontos característicos, entre outras informações locais da imagem, para geração dos descritores (abordagem *local*). Isso torna o problema mais complexo de ser resolvido e aumenta o custo computacional. Neste trabalho os descritores são gerados utilizando toda informação do quadro (abordagem *global*), sendo mais simples e menos custoso computacionalmente. Além disso, a inserção de novos vídeos ou categorias não requer que se recalcule ou modifique os descritores gerados previamente.

1.3 Contribuições e Publicações

A principal contribuição deste trabalho está em combinar histogramas de gradientes com tensores de 2ª ordem para gerar descritores de movimentos simples, porém efetivos. O descritor é simples devido à baixa complexidade de tempo e espaço, necessitando de poucos parâmetros e gerando um descritor compacto que é calculado de maneira rápida se comparado à outros descritores. É efetivo porque consegue resultados competitivos em relação às abordagens locais da literatura.

Este trabalho gerou uma publicação no *International Conference on Pattern Recognition 2012* entitulada *Combining gradient histograms using orientation tensors for human action recognition* [12].

1.4 Trabalhos relacionados

São apresentados aqui alguns trabalhos sobre descritores de movimentos utilizando tensores e/ou histogramas de gradientes, além de alguns trabalhos sobre descritores globais.

1.4.1 Descritores baseados em histogramas de gradientes

Em [13], Lowe apresenta um novo método de reconhecimento de objetos em imagens usando características locais. Essas características são invariáveis à escala, translação,

rotação e, parcialmente invariáveis às mudanças de brilho e projeções afins [13]. Chamado de *Scale Invariante Feature Transform* ou SIFT, esse método transforma uma imagem em uma grande coleção de vetores de características locais. Um dos estágios na criação desses vetores é a geração de descritores a partir do gradiente local da imagem. Esses descritores são gerados por histogramas de gradientes e são altamente distintivos, permitindo que um vetor de características encontre, com alta probabilidade, seu correspondente em uma base de características. Apesar de não ser um descritor de movimentos, o trabalho de Lowe inspirou diversos trabalhos voltados para descrever movimentos em vídeos. Porém, seu desempenho em vídeos não é muito bom, pois é necessário a geração dos vetores de características em cada quadro, exigindo alto custo computacional.

Laptev [14] estende métodos conhecidos de reconhecimento em imagens para o domínio espaço-temporal a fim de classificar movimentos em vídeos. Para caracterizar o movimento, ele calcula histogramas em volumes espaço-temporais na vizinhança de pontos de interesse. Cada volume é subdividido em um conjunto de cuboides e para cada cuboide calculam-se histogramas de gradientes (HOG) e de fluxo óptico (HOF - *Histogram of Optical Flow*). Os histogramas são normalizados e concatenados em um descritor similar ao usado no SIFT [13]. Dado um conjunto desses descritores, é criado um *bag-of-features* (BoF) utilizado na posterior classificação. *Bag-of-features* podem ser utilizados na classificação de imagens. A ideia é representar uma imagem através de um conjunto de descritores locais que não possuem relação de ordem entre si. É análogo ao *bag-of-words* (BoW) em que um documento de texto é representado como um histograma das frequências de cada palavra (perdendo a relação de ordem entre as palavras - daí o termo *"bag"*). O uso de BoF requer a criação de um dicionário a partir de uma base de treino, tornando necessário um aprendizado prévio.

Kläser *et al.* [15] apresenta um descritor espaço-temporal baseado em HOG em três dimensões. Em seu trabalho, os histogramas de orientação são quantizados em poliedros regulares onde cada face do poliedro representa um intervalo de classe do histograma.

1.4.2 Descritores baseados em tensores

Kim *et al.* [16] introduzem um novo método chamado *Tensor Canonical Correlation Analysis* (TCCA) que é uma extensão do clássico *Canonical Correlation Analysis* (CCA[2])

[2]Uma ferramenta padrão para inspeção de relações lineares entre dois conjuntos de vetores [17, 18]

para tensores e o aplicam para a classificação de ações/gestos em vídeos. Nesse método, características de similaridade entre dois vídeos são produzidas através de relações lineares e combinadas com um seletor discriminativo de características e um classificador por "vizinho mais próximo" (*nearest neighbor*) para classificação de ações. Porém, o método exige alta demanda computacional caso movimentos similares entre dois vídeos não estejam alinhados no espaço e no tempo.

Krausz e Bauckhage [19] fazem o reconhecimento de ações baseado na ideia da fatorização de tensores não-negativos. Eles consideram uma sequência de vídeo como um tensor de terceira ordem e aplicam uma fatoração não negativa de tensores a essa sequência. Dessa fatoração são extraídas *imagens base* cuja combinação linear geram os quadros da sequência. Dado um conjunto de vídeos de teste, determina-se um conjunto de *imagens base* que representam diferentes partes da silhueta do objeto em movimento. Uma vez que diferentes combinações lineares dessas bases codificam diferentes poses, uma sequência particular de poses corresponde a uma sequência particular de coeficientes lineares. O reconhecimento é feito aplicando esse mecanismo a diferentes partes de um quadro. Como as *imagens base* são geradas previamente por uma base, é necessário gerar novas imagens a cada vez que um novo padrão de movimentos é inserido.

Jia *et al.* [20] apresentam um método de reconhecimento de ações usando análise tensorial e características em multiescala. Nesse método, uma série de silhuetas formam uma imagem chamada de *Serials-Frame* (SF). Assim, uma ação fica representada através de poses contínuas em uma imagem. A imagem SF é então associada a um auto-espaço de tensores chamado *SF-Tensor* (*Serials-Frame Tensor*). É através da análise desse espaço que são extraídas informações para o reconhecimento de diferentes tipos de ações. Assim como em [19], silhuetas representando um movimento são geradas previamente por uma base, resultando no mesmo problema quando necessário inserir novos padrões de movimento.

Khadem *et al.* [21], assim como em [20], utiliza tensores de terceira ordem a partir de silhuetas de um conjunto de testes. O tensor formado compreende três modos que são: pixels, ações e pessoas. São encontrados os coeficientes no espaço de ações bem como o operador de projeção. A sequência a ser consultada é projetada no espaço de ações e o vetor resultante é comparado aos vetores aprendidos para encontrar a classe correspondente à ação.

Kihl *et al.* [22] utiliza informação de movimento através do fluxo óptico. O campo vetorial gerado pelo cálculo do fluxo é projetado em uma base ortogonal de polinômios e uma medida de similaridade é criada usando o maior autovalor do tensor da projeção dos valores dos campos vetoriais. O custo computacional para a projeção do fluxo óptico na base de polinômios tende a aumentar consideravelmente na medida em que se aumenta o número de coeficientes da base.

Mota [23] propõe um descritor global de movimento baseado em um tensor de orientação. Esse tensor, assim como em [22], também é extraído da projeção do fluxo óptico em uma base ortogonal de polinômios.

1.4.3 Descritores globais

Zelnik-manor e Irani [24] desenvolvem um descritor global baseado em histogramas de gradientes. O descritor é obtido extraindo-se escalas multitemporais através da construção de uma pirâmide temporal. Para cada escala, o gradiente de cada pixel é calculado. Então, um HOG é criado para cada vídeo e comparado com outros histogramas para classificar a base de dados. Assim, dois movimentos serão considerados similares se seus histogramas, em uma mesma escala, são similares. Os testes foram realizados na base Weizmann.

Laptev *et al* [25] aplicam o descritor global de Zelnik-manor [24] na base KTH de duas maneiras: usando escalas multitemporais, como o original e usando escalas multitemporais e multiespaciais.

2 FUNDAMENTOS

Neste capítulo são apresentados os fundamentos os quais o descritor proposto se baseia. É feita uma introdução à Máquina Vetor Suporte, técnica utilizada na classificação dos descritores gerados em cada base de vídeos testada. Não nos aprofundaremos no estudo do SVM porque foge do escopo deste trabalho. A ideia é apenas usá-las para classificar os descritores gerados e testar a qualidade dos mesmos na discriminação de movimentos. Sendo assim, na seção 2.1 é feita uma introdução desta ferramenta.

Na seção 2.2 é introduzido o método SIFT, um algoritmo para detectar e descrever características locais em imagens. Uma das etapas desse método é gerar um descritor baseado em histograma de gradientes. Os histogramas de gradientes usados na criação do descritor proposto nesta dissertação são baseados especificamente nessa etapa.

Por fim, nas seções 2.3 e 2.4 são apresentados o histograma de gradientes (HOG) e tensores de 2ª ordem. É com base nessas duas ferramentas que o descritor proposto é criado.

2.1 Máquina Vetor Suporte

Uma máquina vetor suporte (SVM) é uma técnica de aprendizado supervisionado que utiliza algoritmos de aprendizado para analisar dados e reconhecer padrões. Basicamente, o SVM pega um conjunto de dados de entrada e prevê a qual de duas possíveis classes cada um deles pertence. A partir de um conjunto de treino, onde um dado é marcado como pertencente a uma de duas categorias distintas, a etapa de aprendizado do SVM constrói um modelo que associa cada dado a uma ou outra categoria. Um SVM pode classificar dados linearmente separáveis ou não linearmente separáveis. No caso linear, dado um conjunto de treino X de vetores de características x_i, com $i = 1, 2, 3, \ldots, N$, que pertencem a uma de duas classes ω_1 ou ω_2 linearmente separáveis [26], o objetivo é encontrar o hiperplano $g(x) = w^T x + w_0 = 0$ que classifica corretamente todos os vetores de X. A Figura 2.1 mostra um exemplo de uma solução para um dado conjunto de dados. Observe que o hiperplano $h(x)$ também consegue dividir as classes dos dados de treino de forma correta, porém, o hiperplano $g(x)$ consegue essa divisão com mais "folga"

permitindo que um conjunto submetido à classificação possa ter uma margem de variação maior sem que seja classificado de forma incorreta (Figura 2.2).

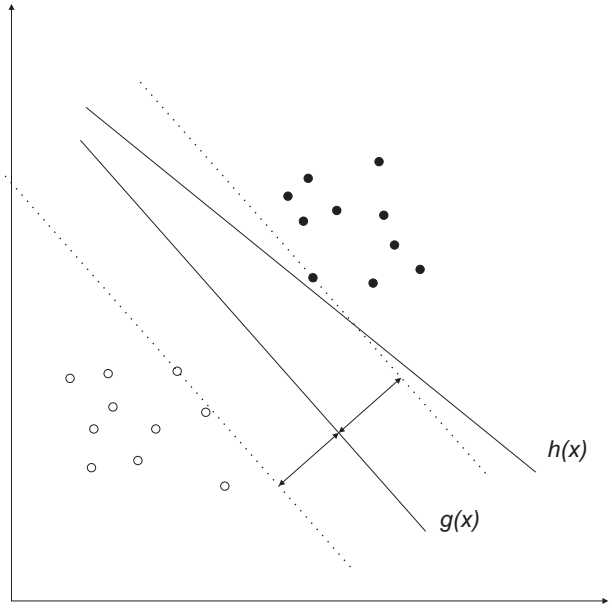


Figura 2.1: Exemplo de duas classes separáveis linearmente e os hiperplanos $g(x)$ e $h(x)$ que as separam.

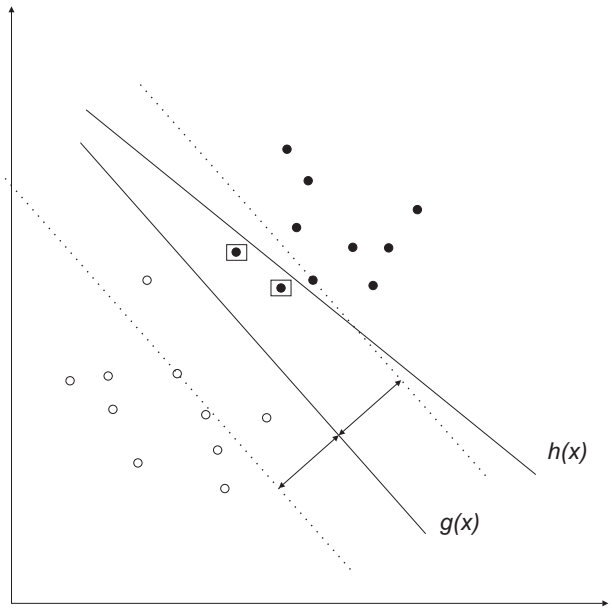


Figura 2.2: Dados classificados utilizando os hiperplanos da Figura 2.1. Observe que o hiperplano $h(x)$ permitiu que dois vetores fossem classificados incorretamente enquanto que $g(x)$ permitiu uma correta classificação.

Quando as classes não são separáveis linearmente (Figura 2.3), não é possível encontrar

um hiperplano que divida os vetores em duas classes distintas. Neste caso, uma função não linear f é usada para levar o conjunto de vetores a uma dimensão maior onde é possível separá-los por um hiperplano (Figura 2.4). Existem diversas funções que cumprem esse papel, chamadas núcleo ou *kernel*, e o resultado da classificação pode variar de acordo com a escolha da função, como visto em [23].

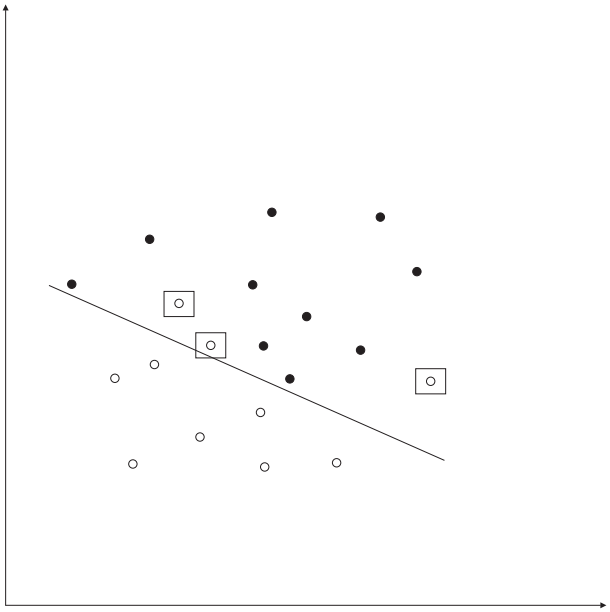


Figura 2.3: Não existe um hiperplano que divida os vetores em duas classes distintas.

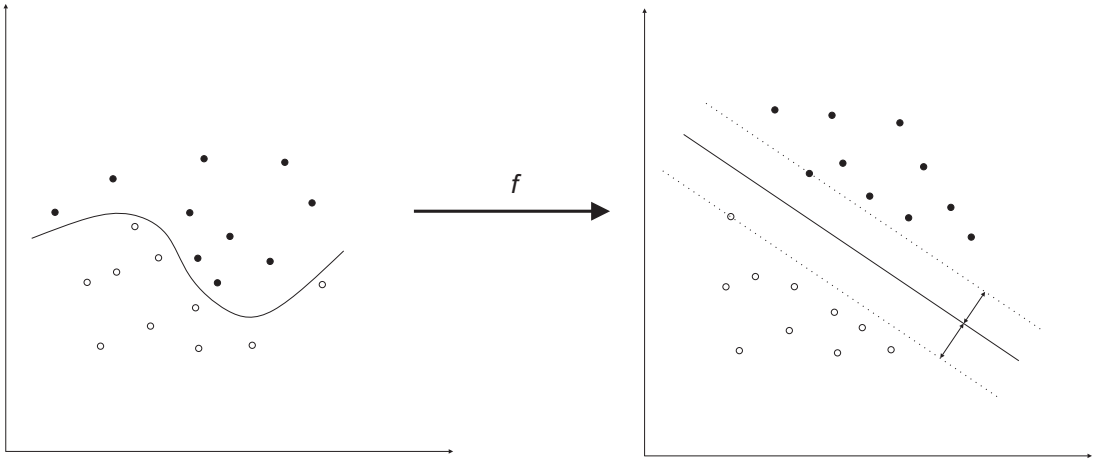


Figura 2.4: Os vetores são levados a uma dimensão maior por uma função f onde é possível separá-los linearmente.

O SVM classifica os dados em duas classes distintas, mas pode-se fazer uma classificação multiclasse considerando o problema, por exemplo, como um conjunto de M

classes dois a dois (abordagem conhecida como *um contra todos*) [26]. Nessa abordagem, para cada uma das classes, o objetivo é conseguir uma função $g_i(x), i = 1, 2, \ldots, M$ tal que $g_i(x) > g_j(x), \forall j \neq i$, se $x \in \omega_i$. Pode-se então projetar funções discriminantes tal que $g_i(x) = 0$ é o hiperplano otimal separando a classe ω_i de todas as outras. Assim, cada classificador é projetado para ter $g_i(x) > 0$ para $x \in \omega_i$ e $g_i(x) < 0$ caso contrário. A classificação é então alcançada de acordo com a regra:

$$i = \arg\max_k \{g_k(x)\} \Rightarrow x \in \omega_i \tag{2.1}$$

2.2 SIFT

SIFT (*Scale-Invariant Feature Transform*) é um método para extrair características distintas e invariantes em imagens, podendo ser usado para detecção de objetos ou cenas em diferentes imagens [27]. O vetor de características calculado é invariante à mudança de escala e rotação e parcialmente invariante à distorções afins, adição de ruído e mudanças de iluminação. Segundo Lowe, esse vetor possui certo número de propriedades em comum com as respostas dos neurônios do córtex inferior temporal dos primatas, responsável pelo reconhecimento de objetos no sistema de visão desses animais.

O cálculo dos vetores de características é feito em etapas. Primeiramente, deseja-se encontrar pontos no espaço de escalas que sejam invariantes à rotação, translação, escalamento e que sofram o mínimo de influência de ruídos e distorções. Isso é feito identificando pontos chave através de máximos e mínimos encontrados em funções geradas por diferenças de gaussianas, que nada mais são do que uma subtração entre duas imagens com um filtro gaussiano aplicado com valores diferentes de σ para cada uma delas.

Em seguida, é feita uma varredura de informações na vizinhança dos pontos localizados. Assim, pontos que tem baixo contraste (suscetíveis a ruídos) ou mal localizados em bordas são rejeitados e os pontos mantidos são chamados pontos chave (*keypoint*).

O próximo passo é associar uma orientação aos pontos chave baseado nas propriedades locais da imagem, tornando-o assim, invariante à rotação. Isso é feito calculando-se os vetores gradientes numa vizinhança do ponto chave e acumulando-os num histograma de gradientes. O pico desse histograma indica a tendência de orientação dos gradientes e será a orientação do ponto.

Os passos anteriores tratam da invariância quanto à localização, escala e rotação de um ponto chave. A última etapa calcula um descritor para cada ponto de modo que ele seja altamente distintivo e parcialmente invariante à iluminação, mudanças de câmera, etc. Primeiro um conjunto de histogramas de gradientes, com oito intervalos de classe cada, é criado em uma vizinhança de 4×4 pixels. Esses histogramas são calculados a partir dos valores da magnitude e orientação de amostras de 16×16 regiões ao redor do ponto chave, de forma que cada histograma contém amostras de uma sub-região de 4×4 pixels da vizinhança original da região. As magnitudes são ponderadas por uma função gaussiana com metade da largura da janela do descritor. O descritor então se torna um vetor com todos os valores dos histogramas. A Figura 2.5 exemplifica esse processo. O descritor é então normalizado a fim de aumentar a invariância de mudanças lineares de iluminação. Para reduzir os efeitos de mudanças não lineares um limiar de $0,2$ é aplicado ao vetor que é novamente normalizado. Esse valor de $0,2$ foi determinado experimentalmente e o autor ([27]) não dá informações detalhadas de como foi obtido.

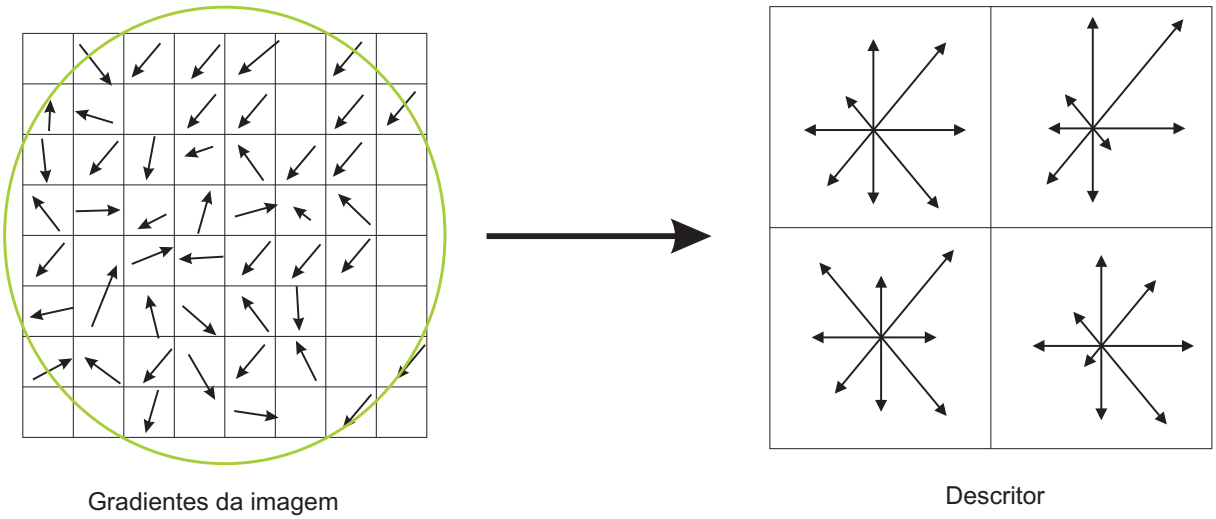


Figura 2.5: Exemplo de um descritor de um ponto no SIFT. Calcula-se o gradiente em cada ponto (imagem da esquerda) e pondera-se com uma janela gaussiana (indicada pelo círculo). Os gradientes são então acumulados em histogramas (imagem da direita) onde o comprimento de cada vetor corresponde à soma das magnitudes dos gradientes com orientação similar. O exemplo utiliza um descritor 2×2 calculado em uma janela 8×8 para melhor visualização.

2.3 Histograma de gradientes

O gradiente do j-ésimo quadro de um vídeo em um ponto p é dado por,

$$\nabla I_j(p) \equiv \left[\frac{\partial I_j(p)}{\partial x}, \frac{\partial I_j(p)}{\partial y}, \frac{\partial I_j(p)}{\partial t}\right], \tag{2.2}$$

ou, equivalentemente, em coordenadas esféricas,

$$\nabla I_j(p) \equiv [\rho_p, \theta_p, \varphi_p], \tag{2.3}$$

onde $\theta_p \in [0, \pi]$, $\varphi_p \in [0, 2\pi]$ e $\rho_p = \|\nabla I_j(p)\|$.

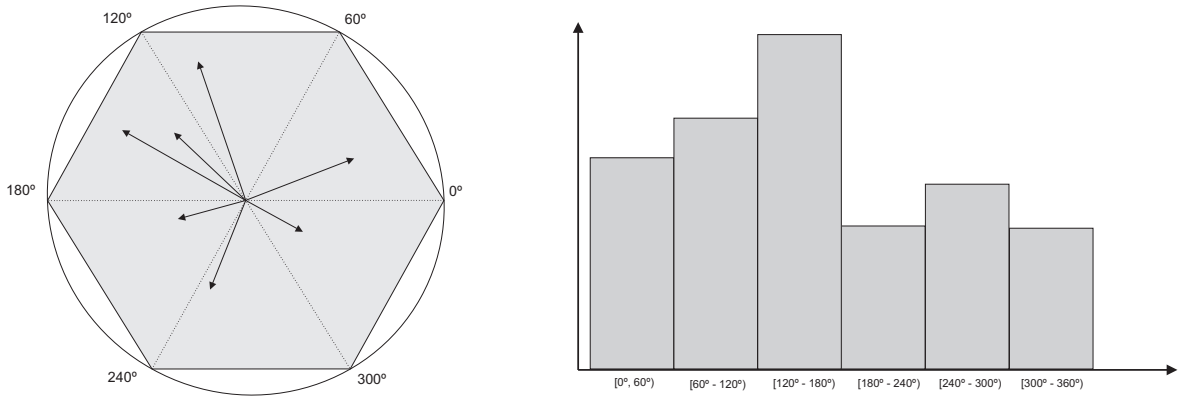
Este vetor aponta para a direção de maior variação de I no ponto p, o que pode indicar informação local de movimento.

Um histograma de gradientes (HOG) é uma distribuição das frequências de gradientes de um quadro ou imagem. Foi proposto por Dalal e Triggs [28], inicialmente utilizado para detecção de pessoas em imagens por ser um bom descritor de características.

A Figura 2.6 mostra um exemplo de um histograma de gradientes bidimensional subdivido em seis intervalos. Cada intervalo guarda a soma das magnitudes de todos os vetores pertencentes ao mesmo. Por exemplo, a frequência em $[120^o, 180^o)$ é a soma das magnitudes dos dois vetores desse intervalo. De fato, um histograma bidimensional pode ser visto como uma aproximação de um círculo por um polígono, onde cada lado do polígono corresponde a um intervalo de classe do histograma. Isso pode ser estendido para o caso tridimensional aproximando-se uma esfera por poliedros. Uma vez que estamos interessados em gradientes espaço-temporais, o histograma de gradientes tridimensionais $h_{k,l}$ com $k \in [1, b_\theta]$ e $l \in [1, b_\varphi]$, sendo b_θ e b_φ o número de intervalos de classe para θ e φ respectivamente, é calculado como:

$$h_{k,l} = \sum_p \rho_p, \tag{2.4}$$

onde $\{p \in I_j \mid k = 1 + \left\lfloor \frac{b_\theta \cdot \theta_p}{\pi} \right\rfloor, l = 1 + \left\lfloor \frac{b_\varphi \cdot \varphi_p}{2\pi} \right\rfloor\}$ são pontos cujos ângulos dos vetores gradientes são mapeados no intervalo de classe (k, l). O campo de gradientes fica então representado por um vetor $\vec{h}_j$ com $b_\theta \cdot b_\varphi$ elementos.



(a) (b)

Figura 2.6: (a) representação das orientações de vetores gradientes na aproximação do círculo por um hexágono. Cada lado do polígono corresponde a um intervalo de classe do histograma. (b) histograma gerado pelas magnitudes e orientações dos gradientes.

2.4 Tensor de 2ª ordem

Tensores são entidades matemáticas que generalizam o conceito de vetores e escalares. Ou seja, um vetor e um escalar são casos particulares de tensores sendo o vetor um tensor de primeira ordem e o escalar um tensor de ordem zero.

Um tensor de 2ª ordem é uma matriz $m \times m$ real e simétrica para sinais m-dimensionais. Podemos usá-los para representar as orientações predominantes em um campo de gradientes. Nesse contexto, são geralmente utilizados em processamento de imagens e visão computacional sendo aplicados, por exemplo, à detecção de pontos de interesse, análise de espaço de escalas [29] e no algoritmo para o cálculo do fluxo óptico de Lucas-Kanade [30].

Definimos o tensor de 2ª ordem T_f como:

$$T_f = \vec{v}\vec{v}^T, \tag{2.5}$$

onde $\vec{v}$ é um vetor com m elementos.

A fim de fornecer uma expressão do movimento médio de quadros consecutivos de um vídeo, podemos combinar os tensores em uma série dada por:

$$S_t = \sum_i T_i, \tag{2.6}$$

onde T_i é o tensor calculado no i-ésimo quadro de um vídeo.

3 DESCRITOR DE MOVIMENTO PROPOSTO

Neste capítulo é apresentado o descritor proposto nesta dissertação. Sua criação envolve o cálculo de um tensor em cada quadro do vídeo ou em um intervalo de quadros. O quadro é divido em partições (Figura 3.2) e em cada uma delas é calculado um histograma de gradientes. Um tensor intermediário é criado a partir dos histogramas e são somados gerando o tensor do quadro. Por fim, esses tensores são somados gerando o descritor final do vídeo. O diagrama da Figura 3.1 mostra as etapas do processo de obtenção do tensor de um quadro que será apresentado nas seções seguintes.

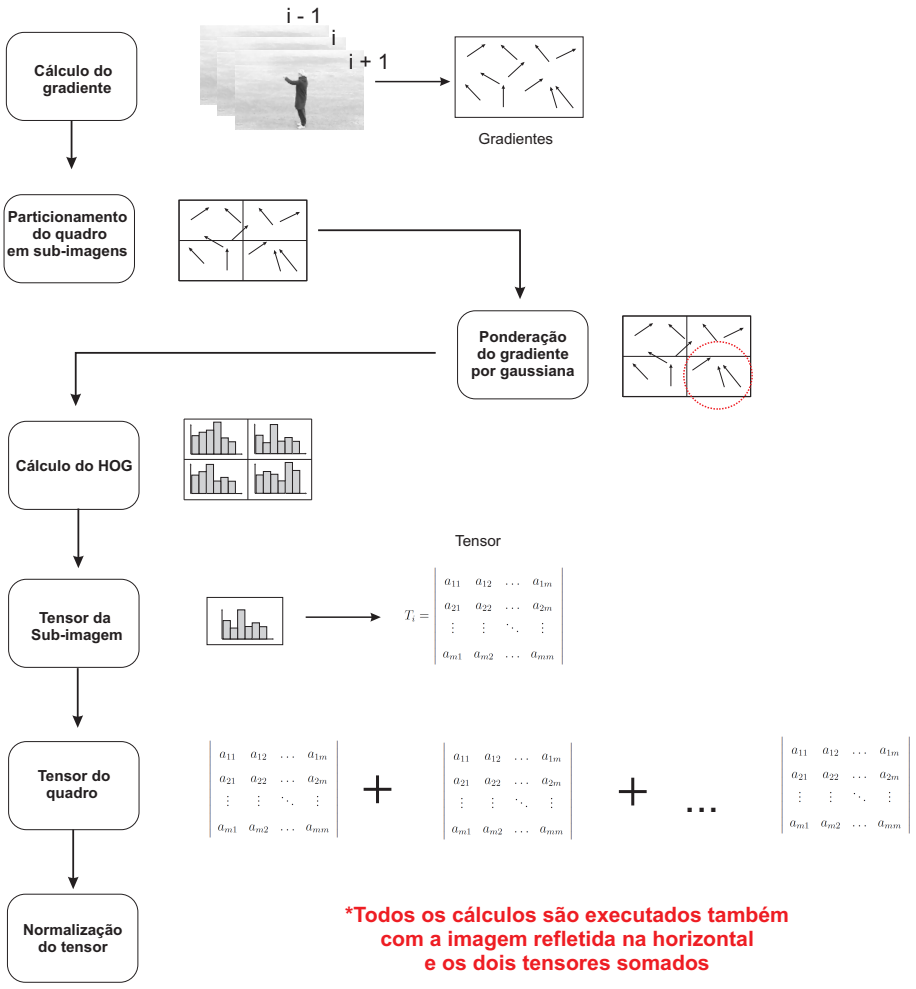


Figura 3.1: Diagrama do cálculo do tensor de um quadro. O descritor final é a soma dos tensores calculados num intervalo de quadros.

Abaixo é mostrado o pseudo-código do método, os passos para o cálculo do descritor serão apresentados nas seções seguintes.

Algorithm 1: Pseudocódigo do método proposto.

Entrada: Vídeo s; número de partições n_x e n_y

Saída: Descritor T_s dos movimentos do vídeo

início

  para *cada quadro* s_i *de* s **faça**

    $\hat{s_i} = s_i$ refletido horizontalmente;

    Calcula o gradiente de s_i e de $\hat{s_i}$;

    Particiona s_i e $\hat{s_i}$ em $n_x \times n_y$ partições;

    para *cada partição* p *de* s_i *e* $\hat{p}$ *de* $\hat{s_i}$ **faça**

      // Ponderação do gradiente

      $\nabla p = w \cdot \nabla p$;

      $\nabla \hat{p} = w \cdot \nabla \hat{p}$;

      // Cálculo do HOG

      Calcula o histograma $\vec{h}_{k,l}$ de p;

      Calcula o histograma $\hat{\vec{h}}_{k,l}$ de $\hat{p}$;

      // Calcula o tensor da sub-imagem

      $T_p = \vec{h}_{k,l} \cdot \vec{h}^T_{k,l}$;

      $\hat{T_p} = \hat{\vec{h}}_{k,l} \cdot \hat{\vec{h}}^T_{k,l}$;

    fim para

    $T_i = \sum_p T_p + \hat{T_p}$;

    Normaliza T_i;

  fim para

  $T_s = \sum_i T_i$;

  Normaliza T_s

fim

3.1 Gradiente espaço-temporal

A primeira etapa na criação do descritor é o cálculo dos vetores gradientes em cada pixel do quadro. Dado um vídeo s, o gradiente espaço-temporal de um quadro $s_i \in s$ é:

$$\nabla s_i \equiv \left[\frac{\partial s_i}{\partial x}, \frac{\partial s_i}{\partial y}, \frac{\partial s_i}{\partial t}\right], \tag{3.1}$$

onde $\left(\frac{\partial s_i}{\partial x}, \frac{\partial s_i}{\partial y}\right)$ é o gradiente espacial em s_i e $\left(\frac{\partial s_i}{\partial t}\right)$ é a taxa de variação entre s_i e o quadro consecutivo s_{i+1}. Esses vetores gradientes capturam variação tanto no espaço quanto no tempo permitindo obter informação de movimento.

3.2 Particionamento do quadro e histograma de gradientes

Quando o histograma de gradientes é calculado usando-se toda a imagem, perde-se qualquer correlação existente entre vetores gradientes que estejam em uma mesma vizinhança na imagem. Como observado em [13] e comprovado nos resultados apresentados no capítulo 4, o particionamento dos quadros do vídeo aumenta a taxa de reconhecimento. O número de partições não deve ser arbitrário e devemos encontrar o valor que proporciona a melhor taxa de reconhecimento. Além disso, essas partições devem se manter fixas em todos os quadros durante a geração dos descritores e um descritor deve ser comparado apenas com outro descritor gerado sob as mesmas configurações.

A segunda etapa na criação do descritor consiste então em dividir o quadro em partições e calcular o histograma de gradientes em cada uma delas. Seja s_i um quadro uniformemente dividido em $n_x \times n_y$ partições não sobrepostas (Figura 3.2). Cada uma das partições pode ser vista como o quadro de um vídeo distinto. Em cada um desses quadros é calculado um histograma de gradientes $\vec{h}^{a,b}_{k,l}$, onde $a \in [1, n_x]$ e $b \in [1, n_y]$. Essa subdivisão permite obtermos uma melhor correlação de posição entre os gradientes da imagem. No entanto, dados dois quadros consecutivos s_i e s_{i+1}, alguns vetores gradientes pertencentes à uma partição no primeiro quadro podem aparecer em uma partição vizinha no quadro seguinte. Isso pode acarretar em uma mudança brusca do histograma mesmo que o movimento seja suave. Para evitar isso, ponderamos cada vetor gradiente em uma partição com uma gaussiana cujo centro coincide com o centro da partição (Fi-

gura 3.3a). Isso faz com que vetores próximos à fronteira tenham um peso menor e com isso influenciem menos, caso eles transitem de uma partição para outra. Essa ponderação mostrou-se eficaz, como será visto no capítulo 4.

Figura 3.2: Exemplo de um quadro com nove partições. Cada partição gera um HOG.

3.3 Criação dos tensores de 2ª ordem

Após a criação dos histogramas de gradientes, o descritor final é criado a partir dos tensores formados em cada quadro do vídeo.

3.3.1 Tensor de um quadro

Primeiramente, cada histograma $\vec{h}_{k,l}^{a,b}$ de cada partição produz um tensor $T_{a,b}$ referente àquela partição (Figura 3.3). Esse tensor carrega a informação de movimento obtida dos gradientes daquela região e é dado por:

$$T_{a,b} = w_p \cdot \vec{h}_{k,l}^{a,b}\ \vec{h}_{k,l}^{a,b\,T}\,, \tag{3.2}$$

onde w_p é um fator de ponderação que é uniforme quando os quadros não são particionados e gaussiano quando são.

Individualmente, $T_{a,b}$ contém apenas informação referente à partição a qual ele pertence. Mas combinando os tensores de outras partições consegue-se obter covariância entre eles. Assim, criados todos os tensores das partições de um quadro s_i, calcula-se o

tensor final do quadro como:

$$T_i = \sum_{a,b} T_{a,b} \tag{3.3}$$

Esse tensor captura a incerteza da direção dos vetores m-dimensionais $\vec{h}_{k,l}^{a,b}$ de s_i. Além disso, a subdivisão da imagem não muda o tamanho do tensor, podendo-se então variar o número de partições sem interferir no tamanho de T_i e, consequentemente, do descritor final.

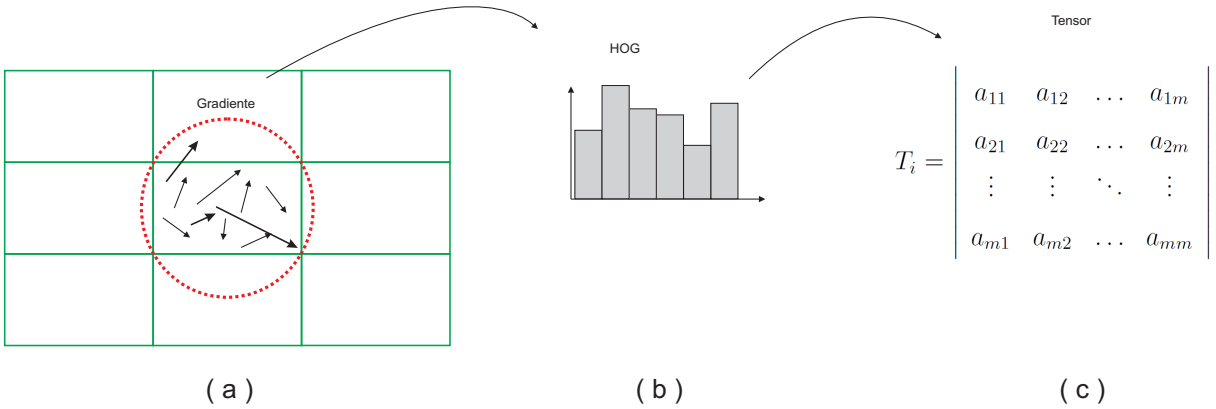


Figura 3.3: (a) gradientes em uma partição. O círculo tracejado representa a função gaussiana centrada no centro da partição. (b) histograma de gradientes. (c) tensor calculado a partir do histograma de gradientes gerado.

3.3.2 Tensor final de um vídeo

Uma vez calculado todos os tensores de todos os quadros, o descritor final T_s de um vídeo s é dado por:

$$T_s = \sum_{i} T_i \tag{3.4}$$

Esse descritor, representado por uma série de tensores, expressa a média de movimento dos quadros consecutivos de s. Podem-se usar todos os quadros do vídeo ou algum intervalo de interesse. O tamanho do tensor final depende exclusivamente da dimensão do histograma de gradientes e é dado por $(b_\theta \cdot b_\varphi)^2$. Porém, o tensor é uma matriz simétrica e pode ser armazenado com $\frac{m(m+1)}{2}$ elementos, onde m é o número de linhas ou colunas do tensor. Por exemplo, um histograma com $b_\theta = 8$ e $b_\varphi = 16$ geraria um tensor de tamanho

128×128 com um total de $(8 \cdot 16)^2 = 16384$ elementos. Porém, somente 8256 elementos precisariam ser armazenados.

Por fim, o descritor final é normalizado usando a norma L_2. Essa normalização é necessária para que descritores gerados por um número diferente de quadros ou por diferentes resoluções de imagem possam ser comparados.

3.3.3 Reflexão do tensor para captura de simetrias

É possível reforçar simetrias horizontais do gradiente que ocorrem no vídeo, mesmo aquelas entre múltiplos frames, através da reflexão horizontal do quadro. Com isso, calcula-se o tensor $\widehat{T}_i$ do quadro refletido e acumula-se com o tensor T_i gerando o tensor final:

$$\widehat{T}_s = \sum_i (T_i + \widehat{T}_i) \tag{3.5}$$

Essa mudança não interfere no processo de obtenção do tensor final que é o mesmo descrito em 3.3.2. Ou seja, somam-se os tensores gerados em cada quadro, com a diferença de que o tensor de cada quadro i passa a ser a soma de T_i com $\widehat{T}_i$.

No capítulo 4 mostra-se que a adição desse tensor aumenta consideravelmente a taxa de classificação dos dados.

3.4 Minimizando o efeito da variação de brilho

Variações na iluminação podem fazer com que dois descritores gerados para movimentos similares sejam bem diferentes já que a magnitude do vetor gradiente está diretamente ligada ao brilho da imagem. Para evitar os efeitos devido à mudança de iluminação nos quadros, é feita uma normalização usando a norma L_2 em cada $\vec{h}^{a,b}_{k,l}$. Como explicado em [27], uma mudança no contraste da imagem, no qual o valor de um pixel é multiplicado por uma constante, irá multiplicar o gradiente pela mesma constante, assim, a normalização irá cancelar o efeito dessa mudança. Porém, uma variação no brilho, na qual uma constante é somada ao valor de um pixel não afetará os valores do gradiente porque eles são calculados a partir de diferenças entre pixels.

Podem ocorrer também, mudanças não lineares de iluminação devido à saturação da câmera ou variações de iluminação em superfícies com diferentes orientações. Esses efeitos podem causar uma grande mudança nas magnitudes de alguns gradientes, mas são

menos prováveis de afetar sua orientação [27]. Isso é reduzido usando uma normalização igual a encontrada no SIFT, onde é feita uma limiarização dos valores do vetor unitário, normalizando novamente em seguida. Isso significa que gradientes com altas magnitudes não são mais importantes do que a própria distribuição de orientações. Na seção de resultados comprova-se que essa limiarização produz um aumento significativo na classificação dos movimentos. É importante dizer que essa normalização torna o tensor possivelmente indefinido, podendo ter autovalores negativos.

4 RESULTADOS E ANÁLISE COMPARATIVA

Neste capítulo é apresentada a avaliação do descritor de movimentos em diversas configurações e é feita uma comparação dos melhores resultados com o que há de mais recente na literatura. O descritor foi utilizado em um classificador Máquina de Vetor Suporte (SVM). Não é objetivo deste trabalho aprofundar-se no estudo do SVM e sim de como montar o descritor de forma simples e eficiente. As configurações adotadas para o SVM, incluindo sua função núcleo, são as mesmas utilizadas no trabalho de Mota [23]: função núcleo triangular e norma L^2.

O descritor foi avaliado através das bases KTH [31] e Hollywood2 [32], descritas na próxima seção. Ambas são amplamente utilizadas na literatura.

A geração dos descritores e sua classificação foram feitos no sistema RETIN (*REcherche et Traque INteractive d'images*) do laboratório ETIS (*Equipes Traitement de l'Information et Systèmes*) da ENSEA (*École Nationale Supérieure de l'Électronique et de ses Applications*) [33].

4.1 Base de vídeos

A base de vídeos KTH é composta por seis tipos de ações humanas:

- *Walking (Walk)*: movimento de pessoa caminhando;
- *Jogging (Jog)*: movimento entre uma corrida e uma caminhada;
- *Running (Run)*: movimento de pessoa correndo;
- *Boxing (Box)*: movimento de pessoa desferindo socos no ar;
- *Hand waving (HWav)*: movimento de pessoa agitando os braços;
- *Hand clapping (HClap)*: movimento de pessoa batendo palmas.

Estas ações são executadas diversas vezes por 25 pessoas e em quatro cenários diferentes (Figura 3.2):

- ambiente externo (s1);
- ambiente externo com variação de escala (s2);
- ambiente externo com variação de velocidade (s3);
- ambiente interno (s4).

No total são 2391 sequências realizadas com fundo homogêneo e uma câmera estática de 25 quadros por segundo. As sequências tem resolução de 160x120 pixels e duram, em média, quatro segundos.

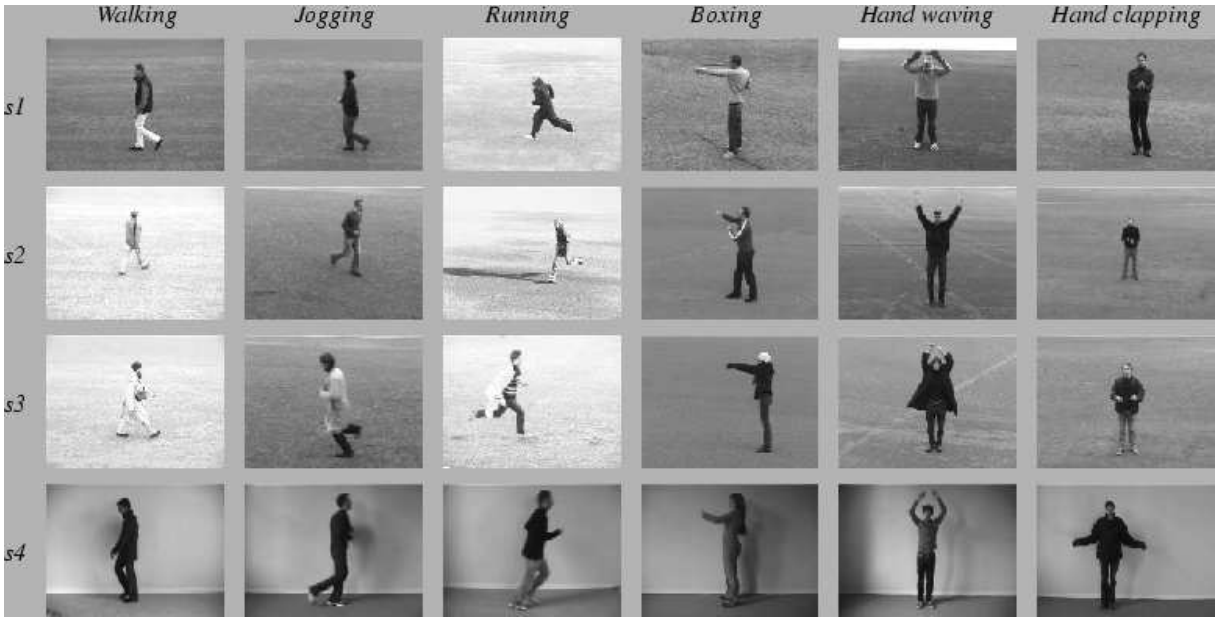


Figura 4.1: Seis tipos de ações em quatro diferentes cenários na base de vídeos KTH [31].

A base Hollywood2 é composta por 12 classes de ações humanas que são:

- *AnswerPhone*: pessoa atendendo o telefone;
- *DriveCar*: pessoa dirigindo;
- *Eat*: pessoa comendo;
- *FightPerson*: cena de luta;

- *GetOutCar*: pessoa saindo do carro;
- *HandShake*: aperto de mãos entre pessoas;
- *HugPerson*: pessoas se abraçando;
- *Kiss*: pessoas se beijando;
- *Run*: pessoa correndo;
- *SitDown*: pessoa sentando;
- *SitUp*: pessoa se levantando;
- *StandUp*: pessoa ficando em pé.

E por 10 classes de cenas tanto externas quanto internas: *EXT-House*, *EXT-Road*, *INT-Bedroom*, *INT-Car*, *INT-Hotel*, *INT-Kitchen*, *INT-LivingRoom*, *INT-Office*, *INT-Restaurant*, *INT-Shop*.

Tudo isso distribuídos em 2669 vídeos a partir de trechos de 69 filmes, totalizando aproximadamente 20.1 horas de gravação. O objetivo da Hollywood2 é fornecer uma base de cálculo para o reconhecimento de ações humanas em um ambiente realístico e desafiador [32].

(a) dirigindo (b) lutando

(c) aperto de mão (d) sentando

Figura 4.2: Exemplos de ações na base Hollywood2 [32].

4.2 Resultados na base KTH

Nesta seção são apresentados resultados classificando a base KTH com um classificador SVM. Para esta base, foi rodado um classificador multiclasse usando uma estratégia *um contra todos* e um critério de *Bayes* para seleção do modelo. A Figura 4.3 mostra a taxa de reconhecimento encontrada para diversos números de partições diferentes do quadro e um HOG de 16×8 intervalos de classe. Além disso, o histograma de cada partição é normalizado segundo a norma L^2. Na Tabela 4.1 são mostrados os valores exatos dessa classificação. Nota-se que o particionamento dos quadros aumenta consideravelmente a taxa de reconhecimento. Comparando-se o melhor resultado, obtido com o particionamento 10×10, com o resultado sem particionamento do quadro (1×1), obtém-se um ganho de $3,59\%$ na classificação. Além disso, como fica fácil observar no gráfico, o aumento do número de partições não garante um aumento na classificação.

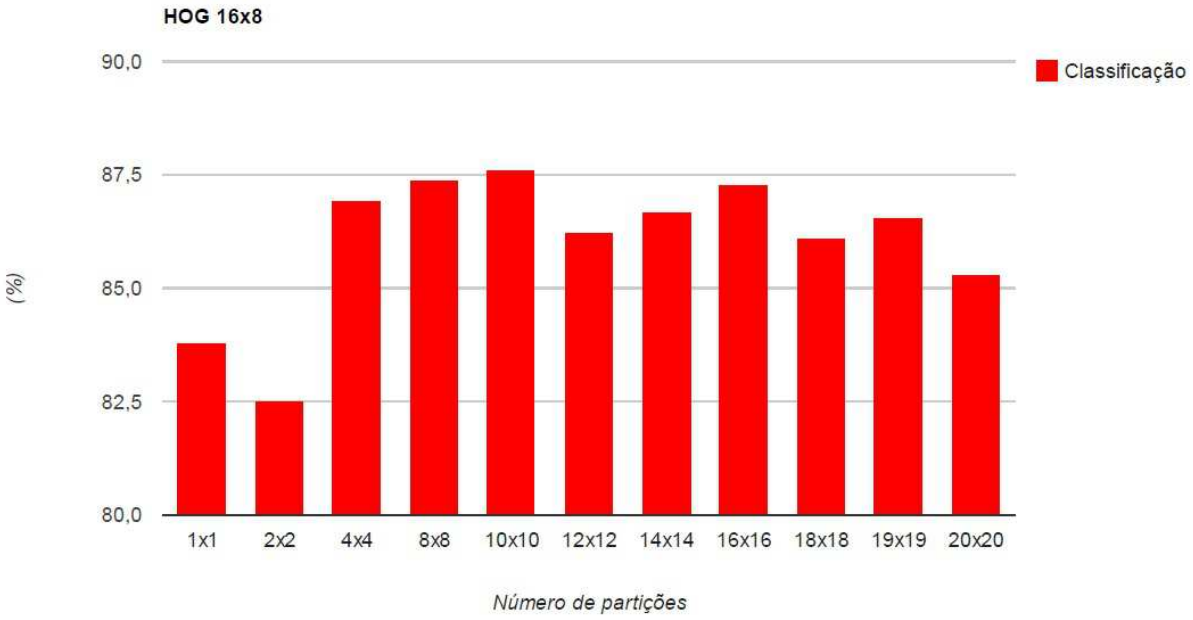


Figura 4.3: Resultados da classificação da base KTH usando HOG 16×8 com norma L^2.

Partição	Taxa de reconhecimento
1x1	83,7882%
2x2	82,5086%
4x4	86,9132%
8x8	87,3786%
10x10	**87,6101%**
12x12	86,2212%
14x14	86,6825%
16x16	87,2612%
18x18	86,1022%
19x19	86,5668%
20x20	85,2920%

Tabela 4.1: Valores da classificação do gráfico da Figura 4.3.

Na Tabela 4.2 são mostrados alguns resultados para outras dimensões do histograma de gradientes.

	Dimensão do HOG		
Particionamento	4x2	6x3	8x4
4x4	78,231%	77,877%	79,732%
8x8	79,163%	78,696%	80,897%
16x16	79,045%	77,656%	81,937%

Tabela 4.2: Resultados utilizando outras dimensões para o histograma de gradientes. Todas as outras configurações são as mesmas da tabela 4.1.

O melhor resultado para as configurações apresentadas até aqui foi com um histograma de dimensão 16×8, com 10×10 partições, normalizado com a norma L^2 em cada histograma gerado. A matriz de confusão para este resultado é apresentada na Tabela 4.3. Ela mostra o percentual de ações classificadas corretamente e também a porcentagem que foi classificada de forma errada.

	Box	HWav	HClap	Jog	Run	Walk
Box	**95,10%**	*7,64%*	*12,50%*	0,0%	0,0%	*0,0%*
HWav	0,0%	**89,58%**	*2,08%*	0,0%	0,0%	0,0%
HClap	*3,50%*	*0,69%*	**86,86%**	0,0%	0,0%	0,0%
Jog	0,0%	0,0%	0,0%	**84,03%**	*19,44%*	*9,03%*
Run	0,0%	0,0%	0,0%	*9,72%*	**80,55%**	0,0%
Walk	*1,40%*	2,08%	0,0%	*6,25%*	*0,0%*	**90,97%**

Tabela 4.3: Matriz de confusão para o melhor resultado.

Observa-se que no caso do *running* e do *jogging*, existe uma maior taxa de erros. Um vídeo que deveria ter sido classificado como *jogging* é classificado como *running* e vice-versa. Isso ocorre porque são movimentos muito similares, diferindo apenas por sua velocidade.

4.2.1 Reflexão do quadro para o cálculo do histograma

O uso de reflexão do quadro aumentou a taxa de reconhecimento como mostra a Tabela 4.4. Com a reflexão, é somado ao tensor gerado em uma partição o tensor gerado pela mesma partição, porém refletida na horizontal. A reflexão na vertical e a reflexão na vertical e horizontal também foram testadas, porém os resultados ficaram piores do que sem o uso de nenhuma reflexão. Essa piora nos resultados usando reflexão na vertical é possivelmente devido à falta de simetria vertical dos movimentos, como o movimento de caminhar ou correr, por exemplo. Os valores de classificação do melhor resultado da Tabela 4.1 comparado a um teste com a mesma configuração mas sem usar reflexão mostra um ganho de quase 2%. A Tabela 4.6 mostra a matriz de confusão da classificação sem reflexão mostrada na Tabela 4.4. Percebe-se uma melhora na classificação dos movimentos *jogging*, *running* e *walking* quando se usa reflexão.

HOG 16x8	
Partição	Taxa de reconhecimento
8x8 sem reflexão	87,609%
8x8 com reflexão	89,578%

Tabela 4.4: Ganho obtido com o uso de reflexão do quadro na geração do descritor.

	Box	HWav	HClap	Jog	Run	Walk
Box	**94,40%**	*2,78%*	*12,50%*	0,0%	0,0%	*0,69%*
HWav	0,0%	**96,53%**	0,0%	0,0%	0,0%	0,0%
HClap	*3,49%*	*0,69%*	**87,50%**	0,0%	0,0%	0,0%
Jog	0,0%	0,0%	0,0%	**77,78**	*18,75%*	*9,03%*
Run	0,0%	0,0%	0,0%	*16,67%*	**79,17%**	0,0%
Walk	*2,098%*	0,0%	0,0%	*5,56%*	*2,08%*	**90.28%**

Tabela 4.5: Matriz de confusão para o resultado sem reflexão da Tabela 4.4.

	Box	HWav	HClap	Jog	Run	Walk
Box	**95,80%**	*2,08%*	*12,50%*	0,0%	0,0%	*1,39%*
HWav	0,0%	**96,53%**	0,69%	0,0%	0,0%	0,0%
HClap	*0,70%*	*1,39%*	**86,81%**	0,0%	0,0%	0,0%
Jog	0,0%	0,0%	0,0%	**79,17**	*12,50%*	*4,86%*
Run	0,0%	0,0%	0,0%	*15,97%*	**85,42%**	0,0%
Walk	*3,50%*	0,0%	0,0%	*4,86%*	*2,08%*	**93.75%**

Tabela 4.6: Matriz de confusão para o resultado com reflexão da Tabela 4.4.

4.2.2 Usando limiarização da norma

O uso de limiarização da norma como mostrado na seção 3.4 também provou-se eficaz. Alguns valores para o limiar foram testados, mas o valor 0.2, que é o mesmo utilizado por [27], foi o que obteve melhor resultado. A Tabela 4.7 mostra um comparativo do resultado com e sem limiarização do melhor resultado obtido na Tabela 4.4.

HOG 16x8	
Partição	Taxa de reconhecimento
8x8 sem limiarização	89,578%
8x8 com limiarização	92,123%

Tabela 4.7: Ganho obtido com o uso de limiarização do tensor do quadro.

A matriz de confusão para o resultado de $92,123\%$ da Tabela 4.7 é mostrada na Tabela 4.9 e para o resultado de $89,578\%$ na Tabela 4.3. Comparando as duas tabelas, nota-se que o uso de limiarização melhorou significativamente os resultados para o movimento *jogging* (de $79,17\%$ para $86,11\%$) e para o movimento *hand clapping* (de $86,86\%$ para $94,44\%$). O alto ganho do movimento *hand clapping* se deu por conta da queda na taxa de classificação incorreta com o movimento *boxing* (de $12,50\%$ para $5,56\%$).

4.2.3 Combinando limiarização e reflexão

A Figura 4.4 compara os resultados obtidos com as diversas combinações de geração do descritor utilizando ou não reflexão e limiarização. Os valores dos resultados são mostrados na Tabela 4.8. A combinação de reflexão do quadro e limiarização do tensor proporcionou um aumento significativo na classificação.

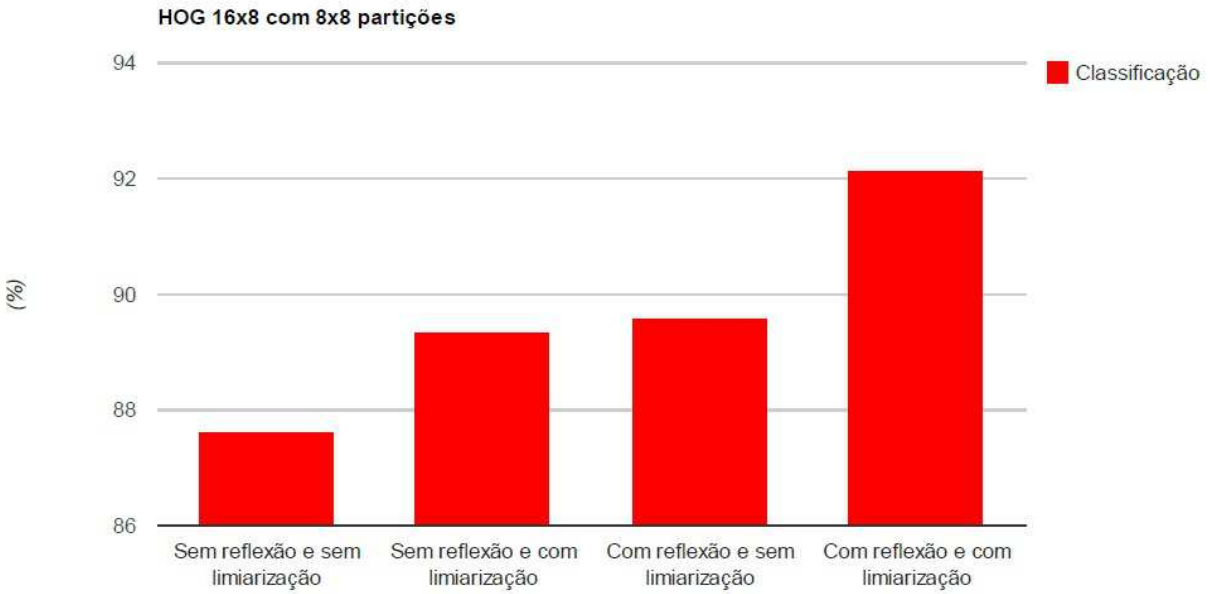


Figura 4.4: Resultados obtidos com diversas configurações no uso ou não de reflexão e limiarização.

Configuração	Taxa de reconhecimento
Sem reflexão e sem limiarização	87,61%
Sem reflexão e com limiarização	89,35%
Com reflexão e sem limiarização	89,58%
Com reflexão e com limiarização	92,12%

Tabela 4.8: Valores dos resultados do gráfico da Figura 4.4.

Através dos valores da tabela e da observação do gráfico verifica-se que o ganho na taxa de reconhecimento quando se utiliza apenas a reflexão do quadro ou apenas a limiarização do tensor são bem próximos, algo em torno de $1,85\%$. Porém, quando combinamos os dois, o ganho aumenta para $4,51\%$.

A Tabela 4.9 mostra a matriz de confusão para o resultado de $92,12\%$ que utiliza reflexão e limiarização. O grande problema continua sendo os movimentos *jogging* e *running*. A porcentagem de movimentos *running* classificados como *jogging* sofreu um aumento de $2,78\%$, em contrapartida, a taxa de movimentos *jogging* classificados como *running* caiu $6,94\%$. As tabelas 4.10 à 4.12 mostram as matrizes de confusão para as

demais configurações da Tabela 4.8.

	Box	HWav	HClap	Jog	Run	Walk
Box	**94,41%**	0,0%	*5,56%*	*0,69%*	0,0%	0,0%
HWav	*0.70%*	**98,61%**	0,0%	0,0%	0,0%	0,0%
HClap	*0.70%*	*1,39%*	**94,44%**	0,0%	0,0%	0,0%
Jog	0,0%	0,0%	0,0%	**86,11%**	*15,28%*	*5,56%*
Run	0,0%	0,0%	0,0%	*9,03%*	**84,72%**	0,0%
Walk	*4,20%*	0,0%	0,0%	*4,17%*	0,0%	**94,44%**

Tabela 4.9: Matriz de confusão para o melhor resultado na base KTH (com reflexão e com limiarização).

	Box	HWav	HClap	Jog	Run	Walk
Box	**94,41%**	*2,78%*	*12,50%*	0,0%	0,0%	*0,69%*
HWav	0,0%	**96,53%**	0,0%	0,0%	0,0%	0,0%
HClap	*3,50%*	*0,70%*	**87,50%**	0,0%	0,0%	0,0%
Jog	0,0%	0,0%	0,0%	**77,78%**	*18,75%*	*9,03%*
Run	0,0%	0,0%	0,0%	*16,67%*	**79,17%**	0,0%
Walk	*2,10%*	0,0%	0,0%	*5,56%*	*2,08%*	**90,28%**

Tabela 4.10: Matriz de confusão para o caso sem reflexão e sem limiarização.

	Box	HWav	HClap	Jog	Run	Walk
Box	**95,10%**	0,0%	*12,50%*	0,0%	0,0%	0,0%
HWav	0,0%	**100,53%**	0,0%	0,0%	0,0%	0,0%
HClap	*2,80%*	0,0%	**87,50%**	0,0%	0,0%	0,0%
Jog	0,0%	0,0%	0,0%	**81,25%**	*20,14%*	*6,94%*
Run	0,0%	0,0%	0,0%	*11,11%*	**77,78%**	0,0%
Walk	*2,10%*	0,0%	0,0%	*7,64%*	*2,08%*	**93,06%**

Tabela 4.11: Matriz de confusão para o caso sem reflexão e com limiarização.

	Box	HWav	HClap	Jog	Run	Walk
Box	**95,80%**	*2,083%*	*12,50%*	0,0%	0,0%	*1,39%*
HWav	0,0%	**96,53%**	*0,69%*	0,0%	0,0%	0,0%
HClap	*0,70%*	*1,39%*	**86,86%**	0,0%	0,0%	0,0%
Jog	0,0%	0,0%	0,0%	**79,17%**	*12,50%*	*4,86%*
Run	0,0%	0,0%	0,0%	*15,97%*	**85,42%**	0,0%
Walk	*3,50%*	0,0%	0,0%	*4,86%*	*2,08%*	**93,75%**

Tabela 4.12: Matriz de confusão para o caso com reflexão e sem limiarização.

4.2.4 Efeito do uso da função gaussiana na ponderação dos gradientes das partições

Todos os testes efetuados a partir da seção 4.2.1 utilizam a ponderação dos vetores gradientes por uma gaussiana como explicado na seção 3.3. O melhor valor para σ_x e σ_y, obtido através de testes, foi de 6.0 pixels. A Figura 4.5 mostra o resultado comparativo da classificação com e sem o uso dessa ponderação e seus valores são mostrados na Tabela 4.13. A matriz de confusão para o caso sem ponderação é visto na Tabela 4.14. Comparando-a com a matriz de confusão da Tabela 4.12, percebe-se que movimentos menos sutis, como o *running*, *hand clapping* e *hand waving* tiveram um aumento bastante significativo da taxa de reconhecimento quando usada a ponderação dos gradientes. No caso do movimento *hand waving* o salto na classificação foi de quase 8%. Esses movimentos têm uma taxa de variação maior e, consequentemente, a mudança de posição de um ponto entre dois quadros consecutivos é brusca. Com isso, o histograma de gradientes pode variar muito entre dois quadros. Dando-se menos peso à fronteira de uma partição, como acontece com o uso da ponderação, consegue-se uma transição mais suave entre tensores de quadros consecutivos.

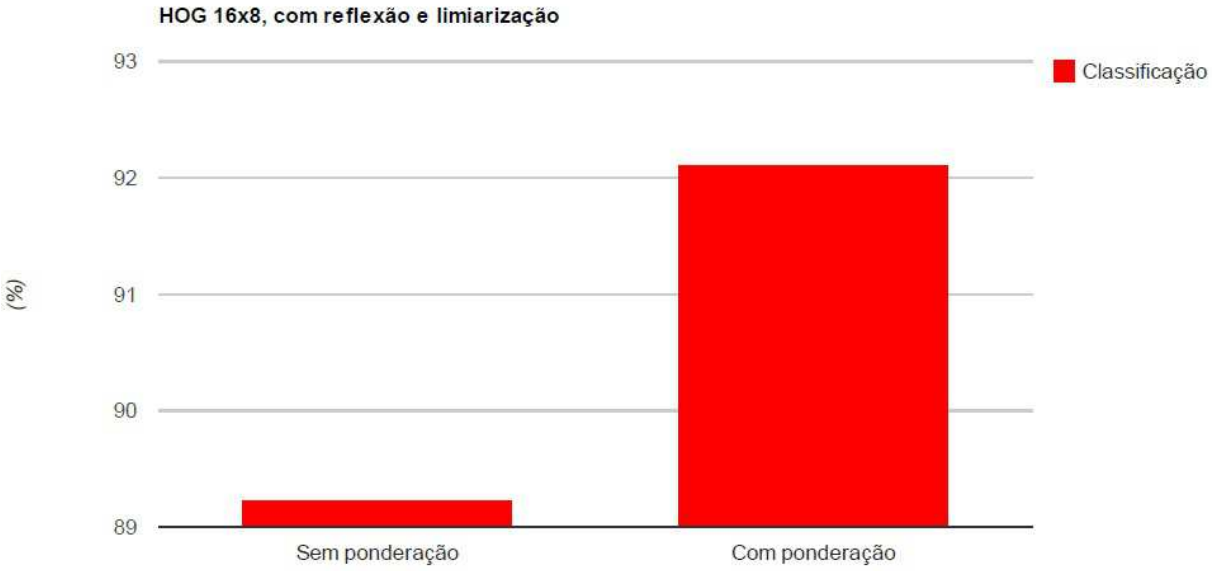


Figura 4.5: Resultados da melhor configuração com e sem ponderação dos vetores gradientes de uma partição.

Ponderação	Taxa de reconhecimento
Sem ponderação	89,229%
Com ponderação	92,123%

Tabela 4.13: Valores da classificação do gráfico da Figura 4.5.

	Box	HWav	HClap	Jog	Run	Walk
Box	**94,41%**	*6,25%*	*7,64%*	*0,69%*	0,0%	0,0%
HWav	*1,40%*	**90,97%**	*0,69%*	0,0%	0,0%	0,0%
HClap	0,0%	*1,39%*	**91,67%**	0,0%	0,0%	0,0%
Jog	0,0%	0,0%	0,0%	**84,03%**	*18,75%*	*6,25%*
Run	0,0%	0,0%	0,0%	*11,11%*	**80,56%**	0,0%
Walk	*4,20%*	*1,40%*	0,0%	*4,17%*	*0,69%*	**93,75%**

Tabela 4.14: Matriz de confusão para a configuração do melhor resultado sem ponderação dos vetores gradientes de uma partição.

4.3 Resultados na base Hollywood2

Nesta seção são apresentados resultados classificando a base Hollywood2 com um classificador SVM. Para esta base, foi rodado um classificador monoclasse, um critério de precisão média para seleção do modelo e validação cruzada.

A Figura 4.6 mostra a taxa de reconhecimento para diversas configurações de partição e histograma sem o uso de reflexão do quadro e com limiarização. Os valores exatos são mostrados na Tabela 4.15. Observa-se que o histograma com dimensões 16×8 é o que gera os melhores resultados, assim como ocorreu com a base KTH. Nas tabelas 4.16 a 4.19 são mostradas as precisões médias em cada classe de ação. As ações que tiveram os melhores resultados são, nessa ordem, *DriveCar*, *Run*, *FightPerson* e *Kiss*. Elas foram as únicas em que se obteve classificação maior que 50%. As ações que tiveram os piores resultados foram *SitUp*, *HandShake* e *AnswerPhone*, respectivamente.

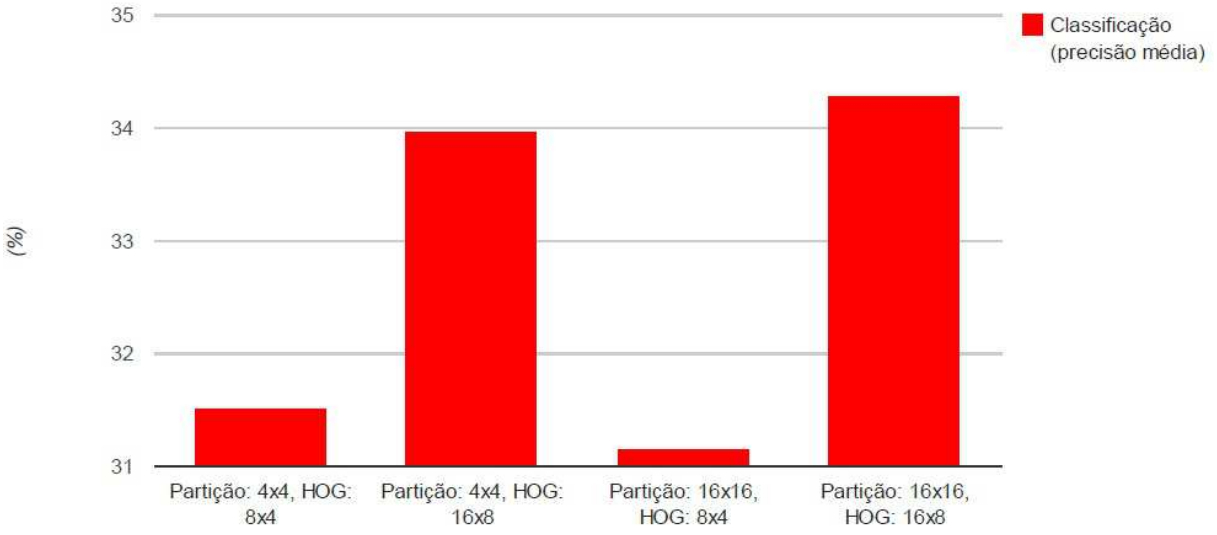


Figura 4.6: Taxas de reconhecimento da base Hollywood2.

Configuração	Taxa de reconhecimento
Partição: 4x4, HOG: 8x4	31.52%
Partição: 4x4, HOG: 16x8	33.98%
Partição: 16x16, HOG: 8x4	31.15%
Partição: 16x16, HOG: 16x8	34.28%

Tabela 4.15: Valores da classificação do gráfico da Figura 4.6.

Ação	**Precisão média**	**Ação**	**Precisão média**
AnswerPhone	13,80%	DriveCar	67,30%
Eat	20,69%	FightPerson	55,13%
GetOutCar	18,90%	HandShake	12,87%
HugPerson	19,33%	Kiss	40,37%
Run	59,09%	SitDown	34,70%
SitUp	7,82%	StandUp	39,42%
		Média	**31,52%**

Tabela 4.16: Precisão média para cada classe da base Hollywood2 usando partição 4×4 e HOG 8×4 sem reflexão.

Ação	**Precisão média**	**Ação**	**Precisão média**
AnswerPhone	13,73%	DriveCar	69,76%
Eat	23,78%	FightPerson	53,64%
GetOutCar	27,71%	HandShake	9,27%
HugPerson	22,46%	Kiss	49,80%
Run	56,68%	SitDown	43,67%
SitUp	9,12%	StandUp	39,05%
		Média	**33,98%**

Tabela 4.17: Precisão média para cada classe da base Hollywood2 usando partição 4×4 e HOG 16×8 sem reflexão.

Ação	Precisão média	Ação	Precisão média
AnswerPhone	12,95%	DriveCar	62,76%
Eat	26,78%	FightPerson	56,92%
GetOutCar	20,31%	HandShake	10,57%
HugPerson	19,24%	Kiss	40,76%
Run	61,16%	SitDown	30,65%
SitUp	7,42%	StandUp	35,11%
		Média	**31,15%**

Tabela 4.18: Precisão média para cada classe da base Hollywood2 usando partição 16×16 e HOG 8×4 sem reflexão.

Ação	Precisão média	Ação	Precisão média
AnswerPhone	14,68%	DriveCar	69,99%
Eat	27,69%	FightPerson	56,19%
GetOutCar	29,30%	HandShake	10,55%
HugPerson	18,66%	Kiss	50,16%
Run	57,94%	SitDown	42,32%
SitUp	11,30%	StandUp	37,80%
		Média	**34,28%**

Tabela 4.19: Precisão média para cada classe da base Hollywood2 usando partição 16×16 e HOG 16×8 sem reflexão.

4.3.1 Reflexão do quadro para o cálculo do histograma

A fim de melhorar o reconhecimento na base Hollywood2, inserimos a reflexão do quadro como feito para a base KTH. A Figura 4.7 mostra as taxas de reconhecimento usando as mesmas configurações dos testes sem reflexão (exceto pela própria reflexão). A Tabela 4.20 mostra os valores exatos da classificação. Comparando com a Tabela 4.15, houve um aumento de $2,24\%$ entre os resultados com 4×4 partições e HOG 16×8 e um aumento de $1,31\%$ entre os resultados com 16×16 e HOG 16×8. Ou seja, a configuração com um número maior de partições teve um aumento menor com a adição de reflexão do quadro no cálculo do histograma.

Como na base KTH, a Hollywood apresentou melhores resultados utilizando partições 4×4 e 8×8. O melhor resultado obtido com a base Hollywood foi com partição 8×8, HOG 16×8 e reflexão do quadro, alcançando $36,34\%$ de classificação. Apesar de esse resultado estar bem abaixo do que foi alcançado na KTH, deve-se levar em consideração o alto nível de complexidade da base Hollywood2. Todos os testes executados nessa subseção fizeram uso de normalização com limiarização e ponderação gaussiana nos gradientes.

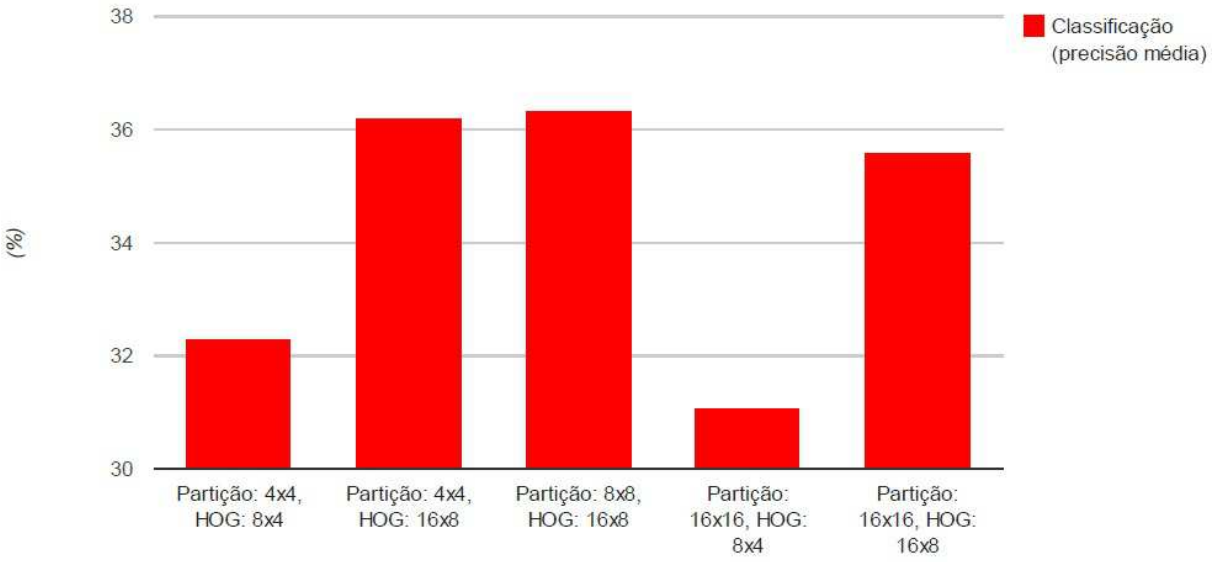


Figura 4.7: Taxas de reconhecimento da base Hollywood2.

Configuração	Taxa de reconhecimento
Partição: 4x4, HOG: 8x4	32,29%
Partição: 4x4, HOG: 16x8	36,22%
Partição: 8x8, HOG: 16x8	36,34%
Partição: 16x16, HOG: 8x4	31,07%
Partição: 16x16, HOG: 16x8	35,59%

Tabela 4.20: Valores da classificação do gráfico da Figura 4.7.

Ação	**Precisão média**	**Ação**	**Precisão média**
AnswerPhone	12,87%	DriveCar	69,68%
Eat	22,04%	FightPerson	42,20%
GetOutCar	26,55%	HandShake	19,32%
HugPerson	25,59%	Kiss	42,15%
Run	59,63%	SitDown	34,87%
SitUp	7,55%	StandUp	40,88%
		Média	**32,29%**

Tabela 4.21: Precisão média para cada classe da base Hollywood2 usando partição 4×4 e HOG 8×4 com reflexão.

Ação	**Precisão média**	**Ação**	**Precisão média**
AnswerPhone	19,30%	DriveCar	70,49%
Eat	22,23%	FightPerson	50,80%
GetOutCar	31,58%	HandShake	16,67%
HugPerson	27,53%	Kiss	50,93%
Run	58,57%	SitDown	48,51%
SitUp	10,61%	StandUp	41,89%
		Média	**36,22%**

Tabela 4.22: Precisão média para cada classe da base Hollywood2 usando partição 4×4 e HOG 16×8 com reflexão.

Ação	**Precisão média**	**Ação**	**Precisão média**
AnswerPhone	18,59%	DriveCar	70,51%
Eat	24,09%	FightPerson	52,80%
GetOutCar	34,82%	HandShake	14,72%
HugPerson	26,59%	Kiss	49,63%
Run	57,94%	SitDown	45,04%
SitUp	11,84%	StandUp	41,38%
		Média	**36,34%**

Tabela 4.23: Precisão média para cada classe da base Hollywood2 usando partição 8×8 e HOG 16×8 com reflexão.

Ação	**Precisão média**	**Ação**	**Precisão média**
AnswerPhone	12,17%	DriveCar	62,25%
Eat	17,87%	FightPerson	47,02%
GetOutCar	25,90%	HandShake	12,58%
HugPerson	22,96%	Kiss	41,48%
Run	61,21%	SitDown	32,67%
SitUp	10,55%	StandUp	39,20%
		Média	**31,07%**

Tabela 4.24: Precisão média para cada classe da base Hollywood2 usando partição 16×16 e HOG 8×4 com reflexão.

Ação	Precisão média	Ação	Precisão média
AnswerPhone	15,05%	DriveCar	70,48%
Eat	23,50%	FightPerson	51,22%
GetOutCar	36,48%	HandShake	12,46%
HugPerson	24,26%	Kiss	49,80%
Run	58,19%	SitDown	44,59%
SitUp	10,92%	StandUp	39,93%
		Média	**35,59%**

Tabela 4.25: Precisão média para cada classe da base Hollywood2 usando partição 16×16 e HOG 16×8 com reflexão.

4.3.2 Efeito do uso da função gaussiana na ponderação dos gradientes das partições

O uso de ponderação gaussiana nos gradientes também promoveu aumento na taxa de reconhecimento para a base Hollywood2. O gráfico da Figura 4.8 mostra esse ganho. Os valores exatos são mostrados na Tabela 4.26. Pela tabela verifica-se que usando ponderação gaussiana tem-se um ganho de $1,18\%$, que é um bom valor de aumento para essa base complexa. A precisão média para o caso sem ponderação é mostrada na Tabela 4.27. Comparando-a com a Tabela 4.23 do caso com ponderação, observa-se que o uso de ponderação acarreta em ganho para algumas classes de ações e perda para outras.

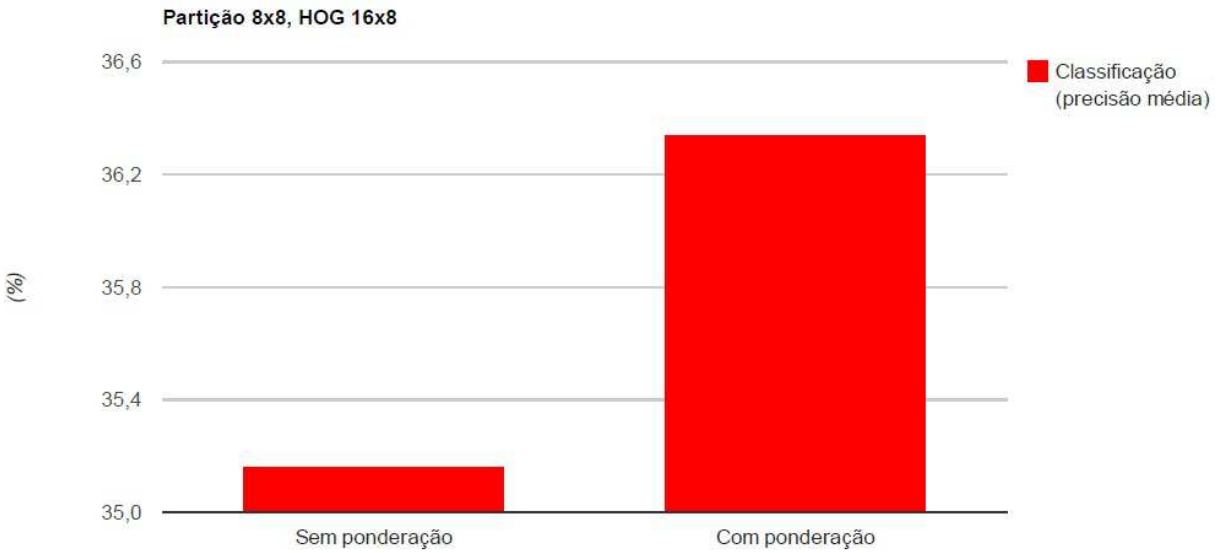


Figura 4.8: Comparação entre a melhor configuração para Hollywood usando ou não a ponderação gaussiana.

Configuração	Taxa de reconhecimento
Sem ponderação	35,16%
Com ponderação	36,34%

Tabela 4.26: Valores da classificação do gráfico da Figura 4.8.

Ação	**Precisão média**	**Ação**	**Precisão média**
AnswerPhone	15,43%	DriveCar	70,03%
Eat	19,81%	FightPerson	54,22%
GetOutCar	29,52%	HandShake	13,00%
HugPerson	26,27%	Kiss	51,42%
Run	55,94%	SitDown	45,41%
SitUp	13,57%	StandUp	40,35%
		Média	**35,16%**

Tabela 4.27: Precisão média para cada classe da base Hollywood2 usando partição 8×8 e HOG 16×8 com reflexão e sem ponderação gaussiana.

4.4 Comparação com descritores da literatura

Nesta seção são comparados os melhores resultados obtidos com outros descritores da literatura.

O desempenho do método proposto para a base KTH é mostrada na Tabela 4.28. A taxa de reconhecimento é comparada com os outros resultados na literatura que utilizam histogramas de gradientes e também com o trabalho de Mota [23] que utiliza tensores de orientação a partir de polinômios de Legendre. O método proposto consegue superar o reconhecimento alcançado por outros métodos.

Método	**Taxa de reconhecimento**
Pirâmides HOG [25]	72%
Polinômios de Legendre + Tensor [23]	86,8%
Harris3D + HOG3D [15]	91.4%
Harris3D + HOG/HOF [14]	91.8%
HOG3D + Tensor (este trabalho)	**92.12%**
ISA [34]	93.9%
TCCA [16]	95.33%

Tabela 4.28: Comparação das taxas de reconhecimento na base KTH.

Na base Hollywood2, este método não consegue superar os melhores resultados. Entretanto, ele consegue uma acurácia competitiva através de uma abordagem muito simples com poucos parâmetros. A Tabela 4.29 compara as taxas de reconhecimento do método proposto com descritores locais de outros trabalhos. Percebe-se que a informação local desempenha um papel fundamental nessa base e que métodos de aprendizado melhoram o reconhecimento de maneira geral.

Método	**Taxa de reconhecimento**
HOG3D + Tensor (estre trabalho)	**36.34%**
Harris3D + HOG3D [15, 35]	43.7%
Harris3D + HOG/HOF [14, 35]	45.2%
ISA [34]	53.3%

Tabela 4.29: Comparação das taxas de reconhecimento na base Hollywood2.

A taxa de reconhecimento do método proposto é menor do que as abordagens locais para a base Hollywood2, porém bastante competitiva. A abordagem apresentada neste texto é rápida e novos vídeos ou novas categorias de ações podem ser inseridas sem necessidade de recalcular os descritores já existentes. Quanto à complexidade de tempo,

os descritores foram calculados com uma média de 23*qps* (quadros por segundo) para todos os vídeos da base Hollywood2 em uma máquina Intel I7 2930MHz com 8Gb de memória. A Tabela 4.30 mostra o tempo gasto em cada etapa do processo na geração dos descritores na base KTH para a melhor configuração. Percebe-se que o cálculo das derivadas e a normalização com limiarização de cada descritor de um quadro dominam a complexidade de tempo. No caso da normalização com limiarização, isso ocorre porque ela é executada duas vezes em cada quadro. Para efeitos de comparação, somente a etapa de extração de características no trabalho de [16] é executada à $1,6qps$ para a base Hollywood2. Se comparado com [34], seu melhor resultado é executado com $10qps$, também para Hollywood2, usando uma GPU GTX270[1].

Etapa	**Tempo total**	**Tempo médio por vídeo**
Cálculo de todo o processo	9m e 50s	246ms
Normalização do tensor de um quadro	2m e 56s	73ms
Cálculo dos gradientes	2m e 35s	64ms
Histograma de Gradientes	2m e 29s	62ms
Normalização do descritor final	0m e 6s	2ms

Tabela 4.30: Tempo gasto nas etapas de geração dos descritores na base KTH com a melhor configuração. O tempo total refere-se ao tempo gasto para calcular os descritores em toda a base.

[1]O artigo não fornece detalhes sobre o que foi implementado em GPU.

5 CONCLUSÃO

Neste trabalho foi apresentado um método para descrever movimentos baseado na combinação de histogramas de gradientes com tensores de 2ª ordem. Para validação da qualidade do descritor proposto foram classificadas as bases KTH e Hollywood2 e seus resultados comparados com outros trabalhos na literatura.

A abordagem adotada é simples, mas efetiva para classificação de vídeos. Ela é simples pois possui baixa complexidade espacial e temporal. Somente poucos parâmetros são necessários, resultando em um descritor compacto. A complexidade de tempo é dominada pelo cálculo das derivadas, histograma e normalização dos tensores de um quadro (Tabela 4.30). Como esses cálculos dependem apenas dos quadros do vídeo, o processo pode ser escalável e capaz de receber melhorias através de paralelismo como instruções SIMD, processadores de múltiplos núcleos e GPUs.

É também uma abordagem efetiva porque alcança uma alta taxa de reconhecimento na base KTH ($92,12\%$), comparada com as melhores abordagens locais [34, 16] cujas complexidades são muito maiores. Para a base Hollywood, entretanto, foi percebido que a informação local possui papel importante e que métodos de aprendizado melhoram o reconhecimento de maneira geral. A taxa de reconhecimento alcançada por este método é menor que a de abordagens locais, mas ainda assim, bastante competitiva. Um alto índice de erros pode ser aceitável quando a base de dados é frequentemente atualizada ou o tempo de resposta é crítico. Este método não requer que sejam feitas mudanças ou que descritores sejam recalculados devido à adição de novos vídeos e/ou novas categorias de ações.

As melhorias propostas para o descritor se mostraram eficazes aumentando a taxa de reconhecimento tanto na base KTH quanto na Hollywood2. O uso de ponderação dos gradientes fez com que ocorresse um aumento significativo na classificação das bases (Tabela 5.1 e 5.2). Na base KTH, por exemplo, isso é mais visível principalmente em ações com movimentos mais rápidos como o *running*, *hand clapping* e *hand waving* que obteve um aumento de quase 8%.

Ponderação	Taxa de reconhecimento
Sem ponderação	89,229%
Com ponderação	92,123%

Tabela 5.1: Comparação da taxa de classificação com e sem o uso de ponderação na base KTH.

Configuração	Taxa de reconhecimento
Sem ponderação	35,16%
Com ponderação	36,34%

Tabela 5.2: Comparação da taxa de classificação com e sem o uso de ponderação na base Hollywood2.

Outra melhoria proposta foi o cálculo do tensor do quadro refletido horizontalmente. Isso permitiu reforçar simetrias horizontais do gradiente aumentando a taxa de classificação. O uso de reflexão na base KTH fez com que a classificação aumentasse de $89,35\%$ para $92,12\%$ para uma mesma configuração do descritor. No caso da Hollywood2 o aumento foi de $33,98\%$ para $36,22\%$ em uma das configurações testadas.

Uma terceira melhoria no descritor foi o uso da normalização usando um limiar. Essa limiarização, apresentada em [27] tem o objetivo de diminuir a influência que variações não lineares de iluminação. A Tabela 5.3 mostra o ganho obtido para a base KTH.

HOG 16x8	
Partição	Taxa de reconhecimento
8x8 sem limiarização	89,578%
8x8 com limiarização	92,123%

Tabela 5.3: Ganho obtido com o uso de limiarização.

Um interessante estudo futuro seria a exploração de informação local para melhorar o descritor e como agregá-la de maneira a aumentar as taxas de reconhecimento, principalmente na base Hollywood2 permitindo o uso do descritor em situações mais realistas.

Além disso, em algumas situações o cenário possui diversos movimentos de diversos objetos no fundo que não são de interesse. Isso acaba comprometendo a qualidade do descritor ou mesmo tornando-o não discriminante do movimento. É o que ocorre com frequência na base Hollywood2. Assim, a extração de objetos que não são de interesse também necessita de um estudo futuro. Mas ainda assim, ele pode ser de grande valia em um cenário onde nenhum método de classificação de ações humanas resolve todas as demandas de aplicação [35].

REFERÊNCIAS

[1] TURAGA, P., CHELLAPPA, R., SUBRAHMANIAN, V. S., UDREA, O., "Machine Recognition of Human Activities: A Survey", *Circuits and Systems for Video Technology, IEEE Transactions on*, v. 18, n. 11, pp. 1473–1488, Sept. 2008.

[2] JOHANSSON, G., "Visual perception of biological motion and a model for its analysis", *Attention Perception Psychophysics*, v. 14, n. 2, pp. 201–211, 1973.

[3] SARKAR, S., PHILLIPS, P. J., LIU, Z., VEGA, I. R., GROTHER, P., BOWYER, K. W., "The humanID gait challenge problem: Data sets, performance, and analysis", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 27, pp. 162–177, 2005.

[4] RUI, Y., HUANG, T. S., "Image retrieval: Current techniques, promising directions and open issues", *Journal of Visual Communication and Image Representation*, v. 10, pp. 39–62, 1999.

[5] CHANG, S.-F., "The holy grail of content-based media analysis", *IEEE Multimedia*, v. 9, pp. 6–10, 2002.

[6] ZHONG, H., SHI, J., VISONTAI, M., "Detecting Unusual Activity in Video", *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, v. 2, pp. 819–826, 2004.

[7] VASWANI, N., CHOWDHURY, A. R., CHELLAPPA, R., ""Shape Activity": A Continuous State HMM for Moving/Deforming Shapes with Application to Abnormal Activity Detection", *IEEE Trans. on Image Processing*, v. 14, pp. 1603–1616.

[8] PENTLAND, A., "Smart rooms, smart clothes". v. 2, pp. 949–953 vol.2, 1998.

[9] FORSYTH, D. A., ARIKAN, O., RAMANAN, D., "Computational Studies of Human Motion: Part 1, Tracking and Motion Synthesis". In: *Foundations and Trends in Computer Graphics and Vision*, p. 2006, Now Publishers Inc, 2006.

[10] GOMES, J., VELHO, L., *Fundamentos da Computação Gráfica*. 1st ed. Instituto Nacional de Matemática Pura e Aplicada: Rio de Janeiro, RJ, 2008.

[11] BEAUCHEMIN, S., BARRON, J., "The Computation of Optical Flow", 1995.

[12] PEREZ, E. A., MOTA, V. F., MACIEL, L. M., SAD, D., VIEIRA, M. B., "Combining gradient histograms using orientation tensors for human action recognition". In: *ICPR*, 2012.

[13] LOWE, D. G., "Object Recognition from Local Scale-Invariant Features". In: *Proceedings of the International Conference on Computer Vision-Volume 2 - Volume 2*, *ICCV '99*, pp. 1150–, IEEE Computer Society: Washington, DC, USA, 1999.

[14] LAPTEV, I., MARSZAŁEK, M., SCHMID, C., ROZENFELD, B., "Learning Realistic Human Actions from Movies". In: *Conference on Computer Vision & Pattern Recognition*, jun 2008.

[15] KLÄSER, A., MARSZAŁEK, M., SCHMID, C., "A Spatio-Temporal Descriptor Based on 3D-Gradients". In: *British Machine Vision Conference*, pp. 995–1004, sep 2008.

[16] KYUN KIM, T., FAI WONG, S., CIPOLLA, R., "R.: Tensor Canonical Correlation Analysis for Action Classification". In: *CVPR*, 2007.

[17] BACH, F. R., JORDAN, M. I., *A probabilistic interpretation of canonical correlation analysis*, Tech. rep., 2005.

[18] HARDOON, D. R., SZEDMAK, S., SZEDMAK, O., SHAWE-TAYLOR, J., *Canonical correlation analysis; An overview with application to learning methods*, Tech. rep., 2007.

[19] KRAUSZ, B., BAUCKHAGE, C., "Action Recognition in Videos Using Nonnegative Tensor Factorization". In: *ICPR*, pp. 1763–1766, 2010.

[20] JIA, C., WANG, S., XU, X., ZHOU, C., ZHANG, L., "Tensor analysis and multi-scale features based multi-view human action recognition". In: *International Conference on Computer Engineering and Technology*, 2010.

[21] KHADEM, B. S., RAJAN, D., "Appearance-based action recognition in the tensor framework". In: *Proceedings of the 8th IEEE international conference on*

Computational intelligence in robotics and automation, *CIRA'09*, pp. 398–403, IEEE Press: Piscataway, NJ, USA, 2009.

[22] KIHL, O., TREMBLAIS, B., AUGEREAU, B., KHOUDEIR, M., "Human activities discrimination with motion approximation in polynomial bases." In: *ICIP*, pp. 2469–2472, IEEE, 2010.

[23] MOTA, V. F., *Tensor baseado em fluxo óptico para descrição global de movimento em vídeos*, Mestrado, Universidade Federal de Juiz de Fora, Juiz de Fora, Brasil, 2011.

[24] ZELNIK-MANOR, L., IRANI, M., "Event-based analysis of video". In: *In Proc. CVPR*, pp. 123–130, 2001.

[25] LAPTEV, I., CAPUTO, B., SCHÜLDT, C., LINDEBERG, T., "Local velocity-adapted motion events for spatio-temporal recognition", *Comput. Vis. Image Underst.*, v. 108, n. 3, pp. 207–229, Dec. 2007.

[26] THEODORIDIS, S., KOUTROUMBAS, K., *Pattern Recognition, Fourth Edition*. 4th ed. Academic Press, 2008.

[27] LOWE, D. G., "Distinctive Image Features from Scale-Invariant Keypoints", *Int. J. Comput. Vision*, v. 60, n. 2, pp. 91–110, Nov. 2004.

[28] DALAL, N., TRIGGS, B., "Histograms of Oriented Gradients for Human Detection". In: *In CVPR*, pp. 886–893, 2005.

[29] LINDEBERG, T., "Scale-Space Theory in Computer Vision", 1994.

[30] LUCAS, B. D., KANADE, T., "An Iterative Image Registration Technique with an Application to Stereo Vision". pp. 674–679, 1981.

[31] SCHÜLDT, C., LAPTEV, I., CAPUTO, B., "Recognizing human actions: A local SVM approach". In: *In Proc. ICPR*, pp. 32–36, 2004.

[32] MARSZALEK, M., LAPTEV, I., SCHMID, C., "Actions in context", *IEEE Conf. Computer Vision and Pattern Recog*, 2009.

[33] FOURNIER, J., CORD, M., PHILIPP-FOLIGUET, S., PONTOISE CEDEX, F. C., "RETIN: A content-based image indexing and retrieval system", 2001.

[34] LE, Q. V., ZOU, W. Y., YEUNG, S. Y., NG, A. Y., "Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis". In: *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition*, *CVPR '11*, pp. 3361–3368, IEEE Computer Society: Washington, DC, USA, 2011.

[35] WANG, H., ULLAH, M. M., KLÄSER, A., LAPTEV, I., SCHMID, C., "Evaluation of local spatio-temporal features for action recognition". In: *University of Central Florida, U.S.A*, 2009.