

Um estudo sobre categorização de mídias através do método de Latent Dirichlet Allocation

Este exemplar corresponde à redação final da
Dissertação devidamente corrigida e defendida
por Glauber de Oliveira Costa e aprovada pela
Banca Examinadora.

Campinas, 12 de Julho de 2010.



Prof. Dr. Siome Klein Goldenstein
(Orientador)

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

**FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DO IMECC DA UNICAMP**

Bibliotecária: Maria Fabiana Bezerra Müller – CRB8 / 6162

Costa, Glauber de Oliveira

C823e Um estudo sobre categorização de mídias através do método de latent Dirichlet allocation/Glauber de Oliveira Costa -- Campinas, [S.P. : s.n.], 2010.

Orientador : Siome Klein Goldenstein.

Dissertação (mestrado) - Universidade Estadual de Campinas, Instituto de Computação.

1.Categorização de imagens. 2.Visão por computador.
3.Recuperação de informação . I. Goldenstein, Siome Klein. II.
Universidade Estadual de Campinas. Instituto de Computação. III.
Título.

Título em inglês: A study on media categorization using the latent Dirichlet allocation method

Palavras-chave em inglês (Keywords): 1. Image categorization. 2.Computer vision.
3. Information retrieval.

Área de concentração: Visão computacional

Titulação: Mestre em Ciência da Computação

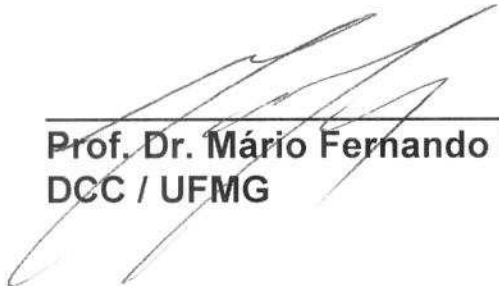
Banca examinadora: Prof. Dr. Siome Klein Goldenstein (IC - UNICAMP)
Prof. Dr. Neucimar Jerônimo Leite (IC – UNICAMP)
Prof. Dr. Márcio Fernando Montenegro Campos (DCC - UFMG)

Data da defesa: 12-07-2010

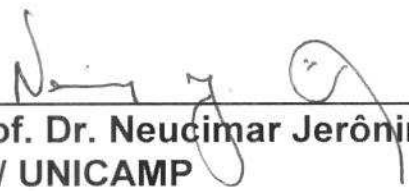
Programa de Pós-Graduação: Mestrado em Ciência da Computação

TERMO DE APROVAÇÃO

Dissertação Defendida e Aprovada em 12 de julho de 2010, pela Banca examinadora composta pelos Professores Doutores:



Prof. Dr. Mário Fernando Montenegro Campos
DCC / UFMG



Prof. Dr. Neucimar Jerônimo Leite
IC / UNICAMP



Prof. Dr. Siome Klein Goldenstein
IC / UNICAMP

Um estudo sobre categorização de mídias através do método de Latent Dirichlet Allocation

Glauber de Oliveira Costa

Maio de 2010

Banca Examinadora:

- Prof. Dr. Siome Klein Goldenstein (Orientador)
- Prof. Dr. Neucimar Jerônimo Leite
Instituto de Computação (IC)– Unicamp
- Prof. Dr. Mário Fernando Montenegro Campos
Departamento de Ciência da Computação (DCC) – UFMG
- Prof. Dr. Eduardo Alves do Valle Jr.
Instituto de Computação (IC) – Unicamp
- Prof. Dr. Léo Pini Magalhães
Faculdade de Engenharia Elétrica e Computação (FEEC) – Unicamp

Prefácio

Com o crescimento das bases de imagem digitais, motivado principalmente pela popularização da *World Wide Web*, bem como a massificação de dispositivos de captura digital de imagens, o processamento e extração de informações semânticas destas imagens cresce em importância.

A informação encerrada nestas imagens não tem significado semântico imediato, sendo necessário o uso de técnicas para capturá-la. A abordagem trivial, que envolve a anotação das imagens por humanos se torna falha à medida que o tamanho das bases cresce, sendo necessário voltar-se para métodos mais sofisticados.

Esta dissertação estudou a aplicação do método *Latent Dirichlet Allocation* em bases de imagens digitais, verificando a performance do algoritmo utilizando quatro métodos distintos de criação de dicionários visuais.

Este trabalho mostrou que a combinação de diferentes descritores capturando aspectos distintos das imagens, para a construção de classificadores pelo método *Latent Dirichlet Allocation* é capaz de obter taxas de acerto médias na faixa de 90%, ainda que cada classificador individualmente não tenha desempenho muito superior à chance.

Ainda, os experimentos realizados demonstraram que a influência do tamanho do dicionário e número de tópicos não é significativa, sendo possível construir classificadores com poucos tópicos latentes a partir de poucas palavras visuais, e portanto, eficientes.

Abstract

With the growth of digital image databases, mainly motivated by the spread of the *World Wide Web* and digital capture devices, processing and extraction of semantic information from these images gain importance.

The information contained within these images have no immediate semantic meaning and techniques must be used in order to acquire it. The trivial approach, which involves the manual annotation of the images by humans, becomes flawed as the size of the database grows. Using a more sophisticated method is needed.

This work studied applications of the *Latent Dirichlet Allocation* method for digital image databases, verifying the algorithm's performance using four different methods of codewords dictionary generation.

It shows that a combination of different descriptors capturing different aspects of the images, aiming at the construction of *Latent Dirichlet Allocation*-based classifiers, is capable of achieving hit ratios around 90%, even if each of the individual classifiers isn't largely superior to chance.

Yet, experiments performed during this work demonstrated that the influence of the codewords dictionary size and the number of topics in the model are not significant, making the construction of small-sized, and thus, efficient classifiers possible.

Agradecimentos

Escrever esta dissertação, concomitantemente à minha atividade profissional, não foi para mim tarefa fácil. Não foram poucas as vezes que o único caminho que me parecia possível era desistir. Dessa forma, não posso começar estes agradecimentos sem mencionar aquele que mesmo não estando mais presente, foi e continua sendo meu grande incentivador: Ayrton Senna do Brasil, meu herói, ídolo e ícone, que através de seus exemplos em vida, me ensinou o que é ser um vencedor.

Agradeço à minha esposa, Bruna, que durante todo esse tempo esteve ao meu lado, sendo grande incentivadora deste trabalho, ajudando inclusive na revisão das versões finais. Aos meus pais. Sem as atitudes de apoio à educação que deles recebi, eu hoje poderia estar processando o Boris Casoy. A bem da verdade, não fossem por atitudes deles no passado, eu nem mesmo existiria. Agradeço também ao meu irmão, essa pessoa hélpis, strogonoficamente sensível, que me serve todo dia de parâmetro para eu não me achar tão bobo assim.

Aos meus amigos Lucas e Moreno, por dentre tantas outras coisas, terem me cedido parte do seu poder computacional para que eu pudesse rodar uma parcela dos experimentos necessários para a conclusão dessa dissertação.

Agradeço também a Red Hat, minha empregadora durante todo esse tempo no qual eu trabalhei nessa dissertação, pela paciência raramente dispensada, em especial na fase final deste trabalho.

Muitos de meus colegas colaboraram com este trabalho, direta, ou indiretamente, na forma de conhecimento, apoio ou incentivo. Dentre eles, agradeço especialmente a Diego Aranha, Fernanda Andaló e Tiago Carvalho.

Para encerrar, sem prejuízo à sua importância, agradeço a Sylvester Stallone por ter feito os filmes do Rocky. Menos o quinto, que eu não gostei.

Sumário

Prefácio	v
Abstract	vi
Agradecimentos	vii
1 Introdução	1
1.1 Trabalhos Relacionados	3
1.2 Aplicações do LDA	4
2 Fundamentos Probabilísticos	9
2.1 Redes Bayesianas	11
2.2 Aprendizado de Parâmetros para Redes Bayesianas	14
2.3 Comparações entre Distribuições de Probabilidade	16
2.4 Expectation Maximization	17
2.5 K-Means	18
3 Classificadores Probabilísticos	20
3.1 Análise de Semântica Latente	21
3.2 Análise de Semântica Latente Probabilística	23
3.3 Alocação Latente de Dirichlet	25
3.4 LDA Suavizado	30
4 Descritores de Imagens	31
4.1 Descritores de Imagens Simples	31
4.2 BIC	32
4.3 SIFT	33
4.4 SURF	39

5 Experimentos	44
5.1 Bases de Imagens	44
5.2 Metodologia e Resultados	46
5.2.1 Experimento 1	51
5.2.2 Experimento 2	63
5.2.3 Experimento 3	71
6 Conclusões	75
Bibliografia	77

Lista de Tabelas

3.1	Resumo comparativo entre propriedades de modelos generativos e discriminativos.	22
5.1	Médias e desvios padrões encontrados para as categorias Normais e Anomalias.	53
5.2	Médias e desvios padrões encontrados para o método de votação com critério de maioria absoluta, para as categorias Normais e Anomalias.	53
5.3	Médias e desvios padrões encontrados para as cinco partições.	54
5.4	Médias e desvios padrões encontrados para as cinco partições, utilizando-se o método de votação com critério de maioria absoluta.	54
5.5	Taxas de acerto médias e desvios padrões encontrados para o descritor BIC, variando-se o tamanho do dicionário visual e o número de tópicos.	64
5.6	Taxas de acerto médias e desvios padrões encontrados para o descritor BIC, variando-se o tamanho do dicionário visual e o número de tópicos.	65
5.7	Taxas de acerto médias e desvios padrões encontrados para o descritor SURF, variando-se o tamanho do dicionário visual e o número de tópicos.	66
5.8	Taxas de acerto médias e desvios padrões encontrados para o descritor SURF, variando-se o tamanho do dicionário visual e o número de tópicos.	67
5.9	Taxas de acerto médias e desvios padrões encontrados para o descritor Patches, variando-se o tamanho do dicionário visual e o número de tópicos.	68
5.10	Taxas de acerto médias e desvios padrões encontrados para o descritor Patches, variando-se o tamanho do dicionário visual e o número de tópicos.	69
5.11	Médias e desvios padrões encontrados para três pontos na curva do número de imagens: 5, 14 e 54. Descritores individuais.	71
5.12	Médias e desvios padrões encontrados para três pontos na curva do número de imagens: 5, 14 e 54.	72

Lista de Figuras

2.1	Exemplo de uma rede Bayesiana geral. Dados valores das probabilidades de A , B , e C , é possível calcular a probabilidade de qualquer outro nó da rede através de um processo de inferência.	12
2.2	Classificador <i>Naïve Bayes</i>	13
3.1	Exemplo de mapeamento fornecido pela técnica LSA. Neste exemplo, as palavras 1, 2 e 4 são associadas ao conceito 1, as palavras 3 e 5 ao conceito 2, e a palavra 6 ao conceito 3.	22
3.2	Rede Bayesiana equivalente à formulação da Equação 3.4. Os nós d e w correspondem aos dados observáveis, enquanto o nó z corresponde às variáveis latentes.	24
3.3	Rede Bayesiana equivalente à formulação da Equação 3.5. Os nós d e w correspondem aos dados observáveis, enquanto o nó z corresponde às variáveis latentes.	24
3.4	Rede Bayesiana representando o modelo LDA. O nó w corresponde aos dados observáveis. α e β são os parâmetros do modelo. Os outros nós, θ e z , são variáveis latentes do modelo.	27
3.5	Rede Bayesiana representando o modelo LDA simplificado, de acordo com a Equação 3.14. Não há variáveis observáveis. γ e ϕ são os parâmetros do modelo, e z e θ variáveis latentes.	29
3.6	Rede Bayesiana representando o modelo LDA suavizado. Os nó w corresponde aos dados observáveis, enquanto os nós α e η representam os parâmetros do modelo.	30
4.1	Obtenção dos vetores de características nos pontos de interesse. Os gradientes são acumulados em um histograma de oito posições. Nesta imagem, 4 histogramas são criados, totalizando um vetor de $4 \times 8 = 32$ dimensões. Por simplicidade, apenas um dos histogramas é mostrado. Em sua execução usual, o SIFT cria 16 histogramas, totalizando 128 dimensões. . .	34

4.2	A imagem à esquerda é a imagem original. À direita, pontos de interesse encontrados pelo SIFT, com suas magnitudes e orientações dominantes. . .	35
4.3	Pontos encontrados pelo SIFT em uma imagem contendo apenas um objeto e seus casamentos em uma imagem contendo uma cena complexa da qual faz parte o objeto original. O objeto encontra-se na mesma pose em ambas as imagens.	36
4.4	Pontos encontrados pelo SIFT em uma imagem contendo apenas um objeto e seus casamentos em uma imagem contendo uma cena complexa da qual faz parte o objeto original. O objeto encontra-se em poses distintas em cada uma das imagens.	37
4.5	Pontos encontrados pelo SIFT e seus casamentos em duas imagens idênticas, exceto pelas condições de iluminação, artificialmente modificadas. . . .	38
4.6	Disposição dos pontos para o cálculo de uma Imagem Integral.	39
4.7	A imagem à esquerda é a imagem original. À direita, pontos de interesse encontrados pelo SURF.	40
4.8	Pontos encontrados pelo SURF em uma imagem contendo apenas um objeto e seus casamentos em uma imagem contendo uma cena complexa da qual faz parte o objeto original. O objeto encontra-se na mesma pose em ambas as imagens.	41
4.9	Pontos encontrados pelo SURF em uma imagem contendo apenas um objeto e seus casamentos em uma imagem contendo uma cena complexa da qual faz parte o objeto original. O objeto encontra-se em poses distintas em cada uma das imagens.	42
4.10	Pontos encontrados pelo SURF e seus casamentos em duas imagens idênticas, exceto pelas condições de iluminação, artificialmente modificadas. . .	43
5.1	Imagens de exemplo para a base de imagens de retina.	45
5.2	Imagens de exemplo para a base de imagens Free Foto Nature.	45
5.3	Exemplo de fragmentos da imagem, extraídos da base de dados de imagens Free Foto Nature. A Figura mostra 1200 fragmentos extraídos de todas as imagens do conjunto de treinamento para uma execução específica.	48
5.4	Exemplo de fragmentos da imagem, extraídos da base de dados de imagens de retina. A Figura mostra 1200 fragmentos extraídos de todas as imagens do conjunto de treinamento para uma execução específica.	49
5.5	Exemplo de fragmentos da imagem, extraídos da base de dados de imagens de retina. A Figura mostra quantidades variadas de fragmentos extraídos de todas as imagens do conjunto de treinamento para uma execução específica. .	50

5.6	Taxas de acerto para os diferentes métodos para a base de imagens de retina. Experimentos realizados com 50 imagens.	55
5.7	Taxas de acerto com método de votação para a base de imagens de retina. Experimentos realizados com 50 imagens.	56
5.8	Taxas de acerto para os diferentes métodos para a base de imagens de retina. Experimentos realizados com 50 imagens. As categorias <i>normais</i> e <i>anomalias</i> são mostradas separadamente.	57
5.9	Taxas de acerto com método de votação para a base de imagens de retina. Experimentos realizados com 50 imagens. As categorias <i>normais</i> e <i>anomalias</i> são mostradas separadamente	58
5.10	Proporção entre taxas de acerto por categorias e unicidade de palavras no conjunto de treinamento, para o descritor SURF.	59
5.11	Proporção entre taxas de acerto por categorias e unicidade de palavras no conjunto de treinamento, para o descritor SIFT.	60
5.12	Proporção entre taxas de acerto por categorias e unicidade de palavras no conjunto de treinamento, para o descritor patches.	61
5.13	Proporção entre taxas de acerto por categorias e unicidade de palavras no conjunto de treinamento, para o descritor BIC.	62
5.14	Tempos de treinamento para vários valores de tópicos para a base de imagens de retina, categoria Normais. Dicionário visual utilizado com 1200 palavras.	70
5.15	Taxas de acerto percentuais médias entre todas as categorias da base Free Foto Nature, em função do número de imagens de treinamento (por classe). As taxas são apresentadas separadamente para cada um dos descritores. . .	72
5.16	Taxas de acerto percentuais médias para as nove categorias da base Free Foto Nature, em função do número de imagens de treinamento (por classe), utilizando o método de votação. As taxas são apresentadas separadamente para cada conjunto de descritores presentes na votação.	73
5.17	Taxas de acerto percentuais médias para as nove categorias da base Free Foto Nature, em função do número de imagens de treinamento (por classe), utilizando o método de votação. As taxas são apresentadas separadamente para cada conjunto de descritores presentes na votação. Cada descritor é apresentado em escala própria, para melhor visualização	74

Capítulo 1

Introdução

Com o aumento das bases de imagem digitais, motivadas principalmente pela popularização da *World Wide Web*, bem como a massificação de dispositivos de captura digital de imagens, o processamento e extração de informações semânticas destas imagens cresce em importância.

Uma imagem, $I(x,y)$, é aqui representada por uma matriz bidimensional de *pixels*, ou *picture elements*. Dessa forma, a representação de uma imagem digital pertence ao domínio discreto. Cada pixel, no entanto, é obtido a partir de uma amostragem de uma função contínua e, como tal, é o menor elemento endereçável em uma dada imagem. Uma quantidade maior de pixels equivale a realizar esta amostragem em uma resolução maior, e consequentemente com menor perda de informação em relação ao domínio contínuo original.

A informação encerrada em $I(x,y)$, no entanto, não tem significado semântico imediato. Além disso, pouco pode-se inferir a respeito do significado da imagem a partir da análise dos *pixels* separadamente. Na tarefa de extração deste significado, deve-se recorrer a alguma técnica adicional.

A forma mais imediata para lidar com este problema consiste na anotação manual feita por humanos. Dessa forma, a cada imagem I_j de uma base de imagens, associa-se um conjunto de palavras w_j . Essa etapa mapeia o problema para o domínio da análise textual, para o qual há amplos resultados na literatura [52].

A grande quantidade de imagens presentes nestas bases inviabiliza esta técnica mais imediata. Além disso, há um lapso semântico entre a descrição textual e o conteúdo da imagem propriamente dito [22], o que traz impactos para quaisquer algoritmos que se apoiem no mapeamento entre texto e imagem.

Assim, deve-se investir em técnicas automatizadas que permitam realizar asserções a respeito das imagens, tendo como entrada nada mais do que as próprias imagens. Várias são as tarefas que podem ser realizadas para a extração de significado em imagens digitais.

Em muitas delas, é conveniente a utilização de etapas de pré-processamento, a fim de obter representações das imagens com maior carga semântica.

Algumas destas etapas envolvem a técnica de *segmentação*. O objetivo de tarefas de segmentação é separar objetos presentes em uma dada imagem. A partir da segmentação, é possível utilizar os objetos encontrados de modo a criar uma representação simplificada que torne a análise semântica mais viável [48]. A segmentação seguida de treinamento para classificação dos objetos encontrados é uma ferramenta útil na classificação semântica de imagens [55]. No entanto, abordagens baseadas na segmentação prévia do conjunto de trabalho esbarram no fato da segmentação ser um problema difícil em si mesmo [3].

Uma alternativa ao uso da segmentação é a representação das imagens através de *descritores de imagens*, sejam eles locais ou globais. Descritores globais descrevem o conjunto da imagem [14]. Um exemplo de descritor global é um histograma de cor. Descritores locais descrevem a vizinhança de pontos de interesse.

Entre as aplicações importantes para a descrição semântica de imagens está a Recuperação de Imagens por Conteúdo, ou CBIR (*Content-Based Image Retrieval*). De forma geral e não extensiva, três grandes classes de CBIR são descritas na literatura [49]: buscas associativas, onde para algum critério de similaridade, deseja-se encontrar imagens similares aos elementos da busca; busca específica, onde deseja-se encontrar uma imagem em particular; e busca por categoria, onde deseja-se encontrar uma imagem pertencente a uma dada categoria.

Uma abordagem possível para a descrição semântica das imagens, da qual trata este trabalho, é construção de *classificadores*. Dado um conjunto $\mathbf{I} = \{I_1, \dots, I_n\}$ de imagens e m grupos pré-definidos, deseja-se identificar a qual dos grupos cada imagem I_j pertence. *Classificadores* são divididos em dois grandes grupos: supervisionados e não-supervisionados.

São ditos *supervisionados*, os classificadores apresentados a dados anotados, ou seja, funções $y = f(x)$, que para um conjunto de dados de treinamento $\mathbf{x} = \{x_1, \dots, x_n\}$ associam cada x_j a uma entre as n classes do problema. A tarefa do classificador é estimar uma função que melhor aproxime a função real [23], a fim de utilizá-la para classificar dados sobre os quais o classificador não teve conhecimento prévio.

São ditos *não-supervisionados*, os classificadores para os quais a única informação disponível é o conjunto $\mathbf{x}$. Deseja-se capturar elementos estruturais dos dados, que permitam descrevê-los ou agrupá-los de forma satisfatória [23].

Também são encontrados classificadores que reúnem características de ambos os classificadores descritos. Classificadores *semi-supervisionados* são aqueles aos quais são apresentados um conjunto de dados $D = \{x_1, \dots, x_n, k_1, \dots, k_n\}$, com anotações $y = f(d = \{x_j\})$ disponíveis para um subconjunto $\{x_j\} \in D$ [9].

Uma abordagem sólida para o problema de classificação consiste no uso de estatística

Bayesiana. Classificadores Bayesianos são classificadores supervisionados empregados com bons resultados práticos na literatura [21, 12, 18]. Nestes, um grafo acíclico dirigido, ou *rede Bayesiana* é utilizado para modelar o grau de confiança para cada uma das classes do modelo através da especificação de probabilidades condicionais.

Redes Bayesianas realizam processos de inferência a partir de tabelas de probabilidades condicionais. Tanto estas probabilidades, como a própria estrutura das conexões de uma rede Bayesiana, podem ser previamente especificadas a partir de consulta a especialistas, ou estimadas em função dos dados de treinamento [41].

Apesar de classificadores Bayesianos serem classificadores supervisionados, a escolha de uma entre as várias redes Bayesianas para melhor descrever um classificador é um problema não supervisionado, uma vez que os exemplos fornecidos referem-se aos dados observados, e não à estrutura da rede em si [21].

Este trabalho é um estudo a respeito da aplicabilidade de classificadores baseados em redes Bayesianas para separar mídias discretizáveis. Atenção especial é dada ao método *Latent Dirichlet Allocation* no domínio específico das imagens digitais.

No restante do presente capítulo, será apresentado um breve resumo de aplicações de análise em imagens que se relacionam com métodos Bayesianos. Em seguida, são mostradas aplicações descritas na literatura para o método *Latent Dirichlet Allocation* tanto no domínio de textos, quanto na criação de classificadores para mídias complexas.

Uma revisão da teoria das probabilidades com foco no ferramental bayesiano é introduzida no Capítulo 2, juntamente com a base teórica de algoritmos da família *Expectation-Maximization*, utilizados para estimar parâmetros de distribuições. Segue-se a isso, no Capítulo 3, exemplos de classificadores probabilísticos, entre eles o *Latent Dirichlet Allocation*. Um resumo dos descritores de imagens utilizados nas práticas desta dissertação é apresentado no Capítulo 4.

Por fim, experimentos realizados no contexto deste trabalho mostraram que a combinação de diferentes descritores capturando aspectos distintos das imagens, para a construção de classificadores pelo método *Latent Dirichlet Allocation* é capaz de obter taxas de acerto médias satisfatórias, ainda que cada classificador individualmente não tenha desempenho muito superior à chance.

Ainda, os experimentos realizados demonstraram que a influência do tamanho do dicionário e número de tópicos não é significativa, sendo possível construir classificadores com poucos tópicos latentes a partir de poucas palavras visuais, e portanto, eficientes.

1.1 Trabalhos Relacionados

Fei-Fei et al. [20] utilizam uma Rede Bayesiana para reconhecimento de objetos utilizando um número reduzido de dados de treinamento. Os autores utilizam um processo

de aprendizado não-supervisionado onde informações genéricas sobre as classes já previamente conhecidas são acoplados à distribuição *a priori*. Esta abordagem reduz o número de imagens de treinamento necessárias para reconhecer uma nova classe de perto de 100, para técnicas convencionais, para um número entre 1 e 5.

Wrede et al. [55] analisam vídeos em busca de objetos pertencentes à classes pré-estabelecidas. Os autores utilizam uma Rede Bayesiana para representar as possíveis ações realizadas envolvendo esses objetos. A rede é utilizada em um processo de inferência, onde eles buscam determinar quais ações estão sendo realizadas a partir dos dados observados. Além disso, seu trabalho aumenta ou diminui as probabilidades da presença de um determinado objeto ou ação de acordo com os outros elementos presentes na cena. No entanto, a estrutura da rede é estática, e precisa ser definida caso a caso.

Gao e Fan [22] utilizam uma Rede Bayesiana para a classificação de imagens de acordo com seu conteúdo semântico em vários níveis. Ou seja: atribui-se uma mesma imagem aos grupos carro, veículo, etc.

1.2 Aplicações do LDA

Apesar de originalmente descrito para documentos de texto, o *Latent Dirichlet Allocation* (LDA) pode ser usado sobre qualquer corpus finito de elementos.

Wei e Croft [53] propõem a utilização do LDA para tarefas de recuperação de informação, em oposição ao uso clássico para classificação de documentos.

De forma geral, e independente do modelo utilizado, define-se uma expressão chamada *verossimilhança de consulta*, como

$$P(Q|D) = \prod_{q \in Q} P(q|D), \quad (1.1)$$

onde D corresponde aos parâmetros do modelo (α e β para o caso do LDA), e $Q = \{q_1, \dots, q_n\}$ os termos utilizados na consulta.

Nesse modelo, $P(Q|D)$ é a verossimilhança que associa a probabilidade de os termos de consulta serem gerados a partir dos parâmetros D . Note que a Equação 1.1 supõe que os termos são independentes entre si dado os parâmetros.

Além disso, nota-se que a Equação 1.1 não especifica um modelo generativo, dado que apenas fornece ferramental para calcular a distribuição *a posteriori*.

Cada termo de consulta q é uma palavra w pertencente a um vocabulário. Através de uma tabela de frequências aliada a algum método de suavização, como suavização de Dirichlet [56], pode-se especificar

$$P(w|D) = \frac{N_d}{N_d + \mu} P_{ML}(w|D) + \left(1 - \frac{N_d}{N_d + \mu}\right) P_{ML}(w|C), \quad (1.2)$$

como sendo a distribuição associada a cada palavra do modelo. Nesta expressão, $P_{ML}(w|D)$ é a estimativa que maximiza a verossimilhança da palavra w em um documento específico contendo N_d palavras, enquanto $P_{ML}(w|C)$ é a estimativa de máxima verossimilhança da palavra w em toda coleção de documentos, C . O termo μ é um parâmetro da distribuição.

O problema de modelos generativos como o LDA para o campo de recuperação de informação é o excesso de generalidade do modelo [11]. Possivelmente, há interesse em determinados termos específicos da consulta. Esta informação é perdida quando se trabalha no espaço de tópicos latentes produzido pelo LDA.

Como um exemplo deste problema, considera-se a seguinte sequência de termos de consulta:

gols brasil mundial Pelé

Espera-se de um modelo como o LDA, que associe os termos “**gols**, **brasil** e **mundial**” a tópicos latentes associados a futebol. Porém, neste caso, além dos termos discutidos, há interesse específico em documentos que mencionem explicitamente o termo **Pelé**. Esta relação não é capturável pelo modelo LDA, a menos que o termo específico apareça com frequência suficiente nos documentos pesquisados.

Uma solução para este problema é especificar as distribuições de probabilidades como combinações lineares entre a Equação 1.2 e o termo de verossimilhança do LDA, $P_{LDA}(w|D)$. Dessa forma, documentos que apresentem alta probabilidade relacionada aos termos específicos podem também contribuir para aumentar a verossimilhança. A Equação pode ser reescrita como

$$P(w|D) = \lambda \left(\frac{N_d}{N_d + \mu} P_{ML}(w|D) + \left(1 - \frac{N_d}{N_d + \mu} \right) P_{ML}(w|C) \right) + (1 - \lambda) P_{LDA}(w|D), \quad (1.3)$$

onde $0 \leq \lambda \leq 1$ é um fator real que determina a proporção entre as componentes. Porém, a melhor configuração entre as componentes gerais e específicas não é óbvia, tornando a solução não satisfatória em muitos casos [11].

Fei-Fei e Perona [1] utilizam os métodos LDA e *Probabilistic Latent Semantic Analysis* (PLSA) para construir um classificador de cenas naturais. A tarefa de categorização de cenas difere significativamente da tarefa de localização de objetos em imagens. Não há necessidade de especificar se e onde determinados objetos estão presentes na cena. Deseja-se apenas classificar a cena como pertencente a uma entre k classes pré-estabelecidas.

Os autores argumentam que seres humanos são capazes de detectar cenas complexas rapidamente, sem dispensar atenção especial. Essa informação sugere algum potencial de exploração de métodos generativos baseados em extração de informações brutas da imagem, sem prévia determinação dos objetos presentes.

A ideia central é representar a imagem como um dicionário de palavras visuais. Cada palavra é um pedaço da imagem, localizado aleatoriamente ou segundo algum padrão, como o uso de extratores de pontos característicos. Assim, cada localização é representada através de algum descritor local, como o *Scale-Invariant Feature Transform* (SIFT) [38]. Uma imagem do conjunto de treinamento é representada como um vetor de descritores. Para o caso do SIFT, por exemplo, a imagem é escrita como N vetores de 128 dimensões.

Uma vez realizado esse processo com todas as imagens do conjunto de treinamento, os N vetores são atribuídos a c “palavras”, através de algum algoritmo de agrupamento, como o *k-means*. Com essa informação é possível representar cada imagem como uma lista de palavras, apenas utilizando o índice do grupo a qual o descritor original pertence.

Um processo similar é empregado frente as imagens do conjunto de testes: elas são descritas como um conjunto de descritores locais que são atribuídos a um dos agrupamentos que compõem o dicionário.

Uma vez que cada imagem é representada por subconjuntos desse dicionário, é possível aprender os parâmetros α e β do modelo LDA para cada classe de imagens. O Algoritmo 1.1 ilustra este método.

Algoritmo 1.1 Algoritmo geral para o treinamento de classificadores baseados em LDA aplicados a imagens.

Entrada: k corpos de imagens $\mathbf{I}_k = \{I_1, \dots, I_n\}$, um conjunto de imagens representativas de cada uma das k classes.

Saída: Um conjunto $O = \{\alpha_1, \beta_1, \dots, \alpha_n, \beta_n\}$, os parâmetros relacionados a cada classe.

```

f ← ∅
for all i ∈ {1, ..., k} do
  for all I_j^i ∈ I_i do
    Extrair f_j^i ← F(I_j^i), conjunto de descritores locais da imagem.
    f ← f ∪ {f_j^i}
  end for
  Extrair C de f através de algum algoritmo de clusterização.
  for all i ∈ {1, ..., k} do
    for all I_j^i ∈ I_i do
      Representar I_j^i como um vetor de palavras de C.
    end for
    Estimar parâmetros α_i e β_i para a classe i.
    O ← O ∪ {α_i e β_i}.
  end for
end for

```

O estudo utilizou 13 classes distintas. As melhores performances foram encontradas

quando toda a imagem é utilizada como fonte de palavras visuais, em oposição ao uso de extratores de pontos característicos. Isso pode ser feito dividindo-se a imagem em grades de algum tamanho fixo ou variável e representando essa região através de algum descritor local como o SIFT. Os autores reportaram taxa de acerto média em torno de 64 % para um conjunto de 13 classes [1], utilizando 100 imagens de treinamento e 50 de testes. Também pode-se notar que classes similares apresentam alto grau de confusão. Ou seja, imagens da classe i sendo classificadas como pertencentes à classe j e vice-versa.

Dessa forma, classificar a imagem como pertencendo a uma entre as N classes com maior verossimilhança aumenta a taxa de acerto do algoritmo à medida que N cresce. É trivial notar que o algoritmo tem taxa de acerto de 100 % quando $N = 13$. Para a média entre todas as 13 classes, os autores reportaram uma taxa de acerto de ~ 82 % quando $N = 2$ [1].

A criação do vocabulário de palavras visuais pode também ocorrer de outras formas, que não o agrupamento de descritores locais. Elango e Jayaraman [19] segmentam as imagens utilizando como vocabulário informações a respeito de cada um dos segmentos, como histograma de cores, histograma de gradientes, contraste, etc.

Os vários segmentos encontrados são também agrupados através de algoritmos de clusterização, para reduzir a quantidade de termos possíveis. O objetivo do trabalho de Elango e Jayaraman, no entanto, não é a classificação de cenas. Os autores utilizaram a distribuição de tópicos de uma imagem em particular como uma representação concisa da mesma, a fim de realizar tarefas de clusterização sobre um espaço de menor dimensionalidade. Os autores reportaram uma taxa de acerto média de ~ 78 % utilizando a base de imagens CIRES [27].

Niebles et al. [42] utilizaram o LDA para encontrar trechos de vídeos que correspondam a ações humanas pertencentes a categorias pré-determinadas. As dificuldades desta tarefa incluem aquelas usualmente encontradas em visão computacional, como presença de oclusão, interferência do fundo, mas também as criadas pelo movimento da câmera. Um objeto que se move com a mesma velocidade que a câmera, por exemplo, aparentará estar parado.

Visando criar as condições para a aplicação da técnica, cada vídeo é representado por uma sequência finita de descritores espaço-temporais (*space-time interest points*). Para satisfazer as exigências matemáticas do modelo LDA, qualquer informação de estrutura geométrica ou temporal relativa aos descritores encontrados é desprezada. Dessa forma, não se espera que a técnica seja capaz de detectar padrões de movimento complexos, cujo significado dependa de uma evolução estrutural no tempo.

Vários descritores locais para vídeos são descritos na literatura [34]. Basicamente, dividem-se em três tipos: os descritores estáticos, baseados em cantos e pontos salientes; os dinâmicos, baseados na técnica do fluxo óptico; e os espaço-temporais, obtidos através

de análise locais das imagens presentes em pequenas janelas de tempo.

Segundo Niebles et al. [42], trabalhos anteriores com descritores espaço-temporais mostraram-se úteis na tarefa específica de determinação de ações humanas, tornando esse tipo de descritor uma escolha adequada para a tarefa proposta.

Para a detecção dos pontos de interesse, há técnicas como detectores de cantos generalizados, descritos por Laptev [42, 33]. Porém, este detector não produz um grande número de pontos em cenas complexas. Os autores utilizam em seu trabalho o método de filtro separador linear, descrito por [42, 17].

A implementação apresentada nessa dissertação é inspirada no trabalho de Fei-Fei e Perona [1]. Além da implementação do método descrito, o trabalho realizado estudou a influência da quantidade de tópicos latentes e do tamanho do dicionário visual na taxa de acerto do classificador. Foram também utilizados outros descritores, como SURF (Seção 4.4) e BIC (Seção 4.2), este último, um descritor global.

O comportamento do classificador quando exposto a informações distintas provenientes de vários descritores diferentes foi analisado, e mostrou alcançar resultados significativamente superiores ao descrito originalmente.

Capítulo 2

Fundamentos Probabilísticos

Define-se um *experimento aleatório* ou *experimento estocástico*, como um experimento cujos resultados são diferentes e imprevisíveis quando executados sob as mesmas condições [35]. De forma mais precisa, exige-se que o conhecimento do observador sobre o experimento seja o mesmo, ainda que as condições do experimento em si não sejam necessariamente as mesmas [41]. Em dois lançamentos de uma moeda, por exemplo, a força com que são realizados são provavelmente diferentes. Porém, estes dados são desconhecidos ao experimentador. O *espaço amostral*, Ω , associado a este experimento é o conjunto de todos os resultados possíveis.

Em um exemplo clássico, ao lançar uma moeda, tem-se o espaço amostral

$$\Omega = \{cara, coroa\}.$$

Qualquer subconjunto de Ω é chamado um *evento*. A cada um dos eventos $k \in \Omega$, pode-se definir um número real a ele associado, uma medida do quão provável é obter k como resultado deste experimento. Esta medida, denotada por $P(k)$ será chamada uma *probabilidade*, se obedecer aos seguintes axiomas [35]:

1. $0 \leq P(k)$,
2. $P(\Omega) = 1$, e
3. Se $A_i \cap A_j = \emptyset$, $i \neq j$, então $P(\cup A_k) = \sum P(A_k)$.

Há duas principais interpretações para o significado da probabilidade [41]:

- a interpretação *frequentista*, onde determina-se a probabilidade a partir de uma sequência infinita de experimentos aleatórios. Os frequentistas afirmam que no limite, a *frequência relativa* de um evento na sequência aleatória é a *probabilidade* deste evento. Ela é assim, uma propriedade física do estado do mundo e por isso é por vezes chamada de *probabilidade física*

- a interpretação *bayesiana*, ou *subjetivista*, na qual a probabilidade é uma característica interior ou subjetiva a um indivíduo. Dessa forma, a probabilidade representa um *grau de certeza* sobre um evento. Devido a seu conhecimento prévio, dois observadores podem atribuir probabilidades diferentes ao mesmo evento e estarem ambas certas. Em situações onde é possível obter frequências relativas sobre um evento, espera-se que ela corresponda à probabilidade que a maioria dos indivíduos atribui a um dado evento. Porém, a rigor, nenhum experimento é necessário para se atribuir probabilidades bayesianas a um evento [24].

O resultado do experimento aleatório não é necessariamente numérico. Pode-se atribuir alguma função a cada um dos resultados do experimento, de forma a obter algum número real. A função $X : \Omega \rightarrow \mathbb{R}$ responsável por esta associação recebe o nome de *variável aleatória*.

Dados dois eventos A e B , define-se a *probabilidade condicional*, $P(A|B)$, como a probabilidade do evento A , dada a ocorrência prévia de B . Isso é equivalente a computar a probabilidade do evento A num espaço amostral alternativo, definido pelo subconjunto $B \subset \Omega$.

Se

$$P(A|B) = P(A), \quad (2.1)$$

diz-se que os eventos A e B são *independentes*. Nesse caso, pode-se escrever

$$P(A \cap B) = P(A)P(B). \quad (2.2)$$

Dado um terceiro evento C , diz-se que A e B são *condicionalmente independentes* em relação a C , caso

$$P(A \cap B|C) = P(A|C)P(B|C), \quad (2.3)$$

de onde pode-se escrever que

$$P(A \cap B|C) = P(A|C)P(B|C). \quad (2.4)$$

Dadas três variáveis aleatórias X , Y e Z , denotamos $P(X, Y, Z) \rightarrow [0, 1]$ a probabilidade conjunta entre as variáveis. Esta distribuição de probabilidades encerra a informação sobre o quão provável é um evento na presença de todos os outros eventos possíveis.

A especificação de uma probabilidade conjunta é útil ao criar modelos que representem acontecimentos sujeitos a mais de uma variável. Porém, enfrenta um problema de especificação de valores [10]: mesmo tratando-se de variáveis aleatórias discretas, qualquer algoritmo que se baseie em uma distribuição de probabilidade conjunta deverá especificar uma quantidade exponencial de valores para definir completamente esta probabilidade.

Por exemplo, para representar a probabilidade conjunta de n variáveis binárias, são necessários $2^n - 1$ valores. A quantidade de valores especificados pode ser substancialmente reduzida se utilizarmos informações sobre independência entre as variáveis aleatórias.

2.1 Redes Bayesianas

Uma forma de reduzir a quantidade de valores a serem especificados por uma distribuição conjunta de probabilidade que modele um evento é o uso de redes Bayesianas. Estas, utilizam-se do Teorema de Bayes (Equação 2.6) para tratar a informação de dependência condicional entre as variáveis [41]. O uso de redes Bayesianas oferece uma série de vantagens, destacando-se entre elas [24]:

- oferecem um ferramental sólido para tratar instâncias de dados incompletas, ou seja, problemas onde apenas parte dos dados é conhecida,
- permitem que se faça afirmações entre relações causais entre as variáveis modeladas no problema,
- compõem um bom ferramental para a associação entre conhecimento prévio de um dado domínio, e conhecimento obtido a partir de dados observáveis, e
- oferecem uma abordagem eficiente, tanto no campo teórico quanto prático, para se evitar problemas de *overfitting* em relação aos dados observados.

Redes Bayesianas são grafos dirigidos acíclicos [10]. Os nós representam variáveis aleatórias, possivelmente com algum mapeamento para estados presentes no mundo real sendo modelado.

As arestas carregam a informação de relações de dependência entre os nós. A inexistência de um caminho entre dois nós X_i e X_j indica que estes nós são condicionalmente independentes. Redes Bayesianas obedecem ao critério local de Markov. Isso significa dizer que qualquer nó é condicionalmente independente de todos os outros, dado seus pais [46].

A distribuição de probabilidade conjunta para a rede é fatorada como

$$P(X_1, \dots, X_n) = \prod_{i=1}^n P(X_i | \{p_i\}), \quad (2.5)$$

onde $\{p_i\}$ representa o conjunto dos pais do nó X_i . Uma vez especificada a probabilidade de $\{p_i\}$, nenhum outro valor é necessário para o cálculo de X_i .

Dessa forma, apenas os valores $P(X_i | \{p_i\})$ necessitam ser especificados, quantidade proporcional à complexidade do modelo, mas em geral ainda menor que a quantidade exponencial de valores inicialmente exigida para se especificar a probabilidade conjunta [10]. Além disso, sob as mesmas suposições de dependência condicional, a fatorização produzida por uma rede Bayesiana é única [10].

Um exemplo de rede Bayesiana é mostrado na Figura 2.1. Nesta rede, G é condicionalmente independente de todos os outros nós dado F .

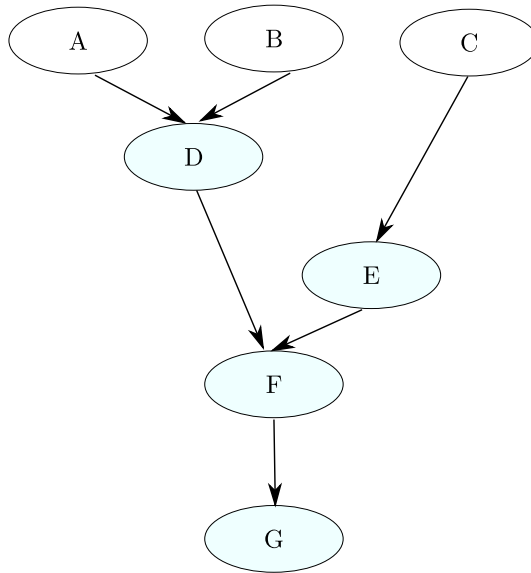


Figura 2.1: Exemplo de uma rede Bayesiana geral. Dados valores das probabilidades de A , B , e C , é possível calcular a probabilidade de qualquer outro nó da rede através de um processo de inferência.

De forma geral, deseja-se computar a probabilidade $P(h)$ de uma hipótese h ser verdadeira. A rede possui dois tipos de variáveis: as variáveis *observáveis*, D , cujo valor pode-se inferir a partir de um exame do estado do modelo, e as variáveis *latentes*, ou *não-observáveis*, cujo valor deseja-se calcular através de um processo de inferência.

Em um conjunto de dados de treinamento D , tem-se que $P(D)$ é a probabilidade *a priori*, representando a probabilidade de o subconjunto denotado por D se verificar no conjunto de dados de treinamento. A probabilidade *a priori* da hipótese h é denotada por $P(h)$. A probabilidade $P(D|h)$ é a probabilidade de se observar o subconjunto D de acordo com a hipótese a ser testada. Esta distribuição pode ser obtida através de resultados diretos de um modelo teórico, consulta a especialistas, aprendizado, etc. De qualquer forma, ela é tomada como elemento conhecido no processo de inferência. A probabilidade de maior interesse é $P(h|D)$, que representa a probabilidade *a posteriori*, ou seja: a probabilidade de h ser uma hipótese verdadeira dado que o subconjunto D foi de fato observado.

O Teorema de Bayes diz que

$$P(h|D) = \frac{P(D|h)P(h)}{P(D)}. \quad (2.6)$$

Dessa forma, é possível relacionar a distribuição *a posteriori* com as distribuições *a priori* e com a probabilidade condicional $P(D|h)$, todas conhecidas. O problema de inferência Bayesiana pode ser encarado como o problema de encontrar a distribuição que maximize a *verossimilhança*

$$h_v = \operatorname{argmax} P(D|h)P(h), \quad (2.7)$$

ou para o caso onde consideramos todas as hipóteses iniciais igualmente prováveis,

$$h_v = \operatorname{argmax} P(D|h), \quad (2.8)$$

onde o termo $P(D)$ é desprezado por se tratar de uma constante em relação à hipótese.

Um exemplo amplamente utilizado de rede Bayesiana é o classificador chamado *Naïve Bayes* [12], mostrado na Figura 2.2. Dado uma variável de decisão y , frequentemente binária, e um conjunto de variáveis $\mathbf{x} = \{x_1, x_2, \dots, x_n\}$, deseja-se estimar $P(y, \mathbf{x})$. É possível notar que para se especificar esta probabilidade conjunta, são necessários $2^{|\mathbf{x}|+1}$ valores. O modelo torna-se rapidamente intratável para um número pequeno de valores em $\mathbf{x}$.

O modelo *Naïve Bayes* assume que todas as variáveis são condicionalmente independentes entre si. Com isso, pode-se dizer que

$$P(y|\mathbf{x}) \sim p(y) \prod_{x_i \in \mathbf{x}} p(x_i|y). \quad (2.9)$$

Neste arranjo, apenas $|\mathbf{x}| + 1$ quantidades precisam ser especificadas. Apesar de simples, o classificador *Naïve Bayes* apresenta bons resultados para diversas aplicações práticas [12], mesmo em domínios onde não se espera que a hipótese de independência condicional seja verdadeira [18].

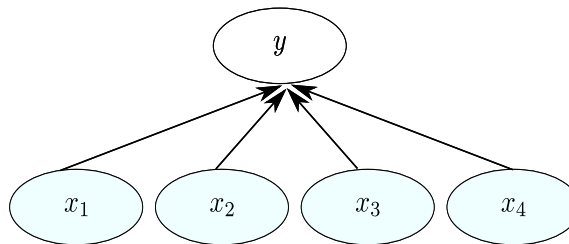


Figura 2.2: Classificador *Naïve Bayes*

A inferência em redes Bayesianas é conhecidamente um problema NP-difícil [40]. Técnicas aproximadas são normalmente utilizadas. Entre estas técnicas, destacam-se a MCMC (Monte-Carlo Markov-Chain), a aproximação de Laplace, e o método *Variational Bayes* [5, 20].

2.2 Aprendizado de Parâmetros para Redes Bayesianas

A forma mais trivial de se atribuir probabilidades a uma rede Bayesiana é manualmente, através de consulta a especialistas, ou outro método similar. Porém, em redes com muitos nós, esse processo se torna algo dispendioso [41]. Para o caso em que as probabilidades são frequências relativas, é possível aprender as probabilidades associadas às variáveis através da observação das frequências com que estas variáveis são observadas. As distribuições aprendidas são chamadas *parâmetros* [41].

De forma geral, o Teorema de Bayes fornece uma forma simples de se realizar aprendizado de máquina. O aprendizado acontece pela atualização dos parâmetros a partir das distribuições que representam as ocorrências dos dados de treinamento. Uma forma de se aprender os parâmetros da rede é o uso de um algoritmo EM (ver Seção 2.4), para estimativa de máxima verossimilhança [36].

Como se pode verificar da Equação 2.6, dois termos de grande importância para esta análise são a verossimilhança $P(D|h)$, e a distribuição *a priori* $P(h)$. A primeira representa a parcela do conhecimento referente aos dados a que a rede é exposta, ou seja, a evidência ou observação. A segunda, representa o conhecimento anterior, previamente adquirido. Através de ajustes nas distribuições, pode-se dar mais ou menos importância a cada uma delas [41].

Considerando as possíveis ocorrências dos dados de treinamento como independentes, podemos representar a densidade de probabilidade referente aos dados como uma distribuição binomial, para o caso de um experimento binário, ou multinomial, para o caso de haver mais de duas possibilidades para a instância dos dados sendo analisada. Assim, uma contagem simples das instâncias dos dados de treinamento já pode fornecer uma estrutura mínima para o aprendizado [41].

É possível notar que na Equação 2.6, o termo de verossimilhança $P(D|h)$ é multiplicado pela distribuição *a priori* $P(h)$. Deseja-se que este cálculo seja tão fácil quanto possível. É possível garantir isso com a utilização de uma distribuição *a priori* que pertença a uma classe conjugada à função de verossimilhança. Uma classe conjugada é uma classe de funções que garante que a distribuição *a posteriori* será da mesma classe que a distribuição *a priori* [43].

No caso dos dados serem unidimensionais e descritos por uma distribuição normal, deseja-se, por exemplo, uma distribuição *a priori* descrita por uma distribuição também normal. Distribuições normais sobre uma variável são ditas *auto-conjugadas*. Para o caso de uma distribuição normal sobre várias variáveis, tem-se a distribuição *Normal-Wishart* como sua conjugada.

Considerando os dados observáveis como modeláveis por uma distribuição multinomial, estamos interessados que a distribuição *a priori* seja uma distribuição de Dirichlet, sua

distribuição conjugada [41]. Na Seção 2.2, esta distribuição é apresentada em detalhes.

Distribuição de Dirichlet

A função de densidade de probabilidades de Dirichlet representa a probabilidade de que M eventos disjuntos x_i sejam observados $\alpha_i - 1$ vezes. Tem a forma [45]

$$D(h, \alpha_1, \alpha_2, \alpha_3, \dots, \alpha_M) = \beta \prod_{i=1}^{i=M} \theta^{(\alpha_i-1)}, \quad (2.10)$$

onde h é o conjunto das hipóteses, β é uma constante utilizada para se assegurar que a distribuição é de fato uma distribuição de probabilidades e α , os chamados parâmetros da distribuição, números reais e positivos. Uma característica da distribuição de Dirichlet é que a variável aleatória θ tem suas componentes pertencentes a um $M - 1$ -simplex, ou seja,

$$\sum_{i=1}^{i=M} \theta_i = 1, \quad 0 < \theta_i < 1. \quad (2.11)$$

Além do fato de ser uma função conjugada, a distribuição de Dirichlet tem outras propriedades interessantes que justificam seu uso para as tarefas de aprendizado. Entre elas, destaca-se o cálculo da esperança com solução analítica simples

$$E(h_j|\alpha) = \frac{\alpha_j}{\sum_{i=1}^{i=M} \alpha_i}. \quad (2.12)$$

A probabilidade de que dada uma observação que obedece uma distribuição multinomial (Ver Seção 2.2) representada pelo vetor $\mathbf{O}$, o valor esperado seja V_j , é dado por

$$P(V_j|\mathbf{O}) = \frac{O_j + \alpha_j}{\sum_{i=1}^{i=M} O_i + \sum_{i=1}^{i=M} O_i \alpha_i}. \quad (2.13)$$

Nota-se que nesta equação, há uma parcela de importância da distribuição *a priori*, representada pelos termos α_i , e uma parcela da observação propriamente dita, representada pelos valores de O_i .

Distribuição Multinomial

A distribuição multinomial é a generalização para várias variáveis da distribuição binomial, que representa a distribuição dos sucessos esperados em n experimentos de Bernoulli. Cada experimento de Bernoulli é um experimento aleatório com apenas dois resultados possíveis. Dessa forma, a distribuição binomial representa a probabilidade de obter p sucessos em n eventos distintos [35].

Para várias variáveis, não mais é mais admitido um resultado binário do tipo sucesso/falha, mas sim um entre k diferentes valores. Uma variável aleatória X_i associada a um destes experimentos representa a quantidade de vezes em que o resultado denotado por i foi observado em n tentativas. Assim, pode-se dizer que $X = (X_1, \dots, X_n)$ obedece uma distribuição multinomial.

Um exemplo de uma distribuição multinomial, é a contagem de termos em um documento, como será detalhado no Capítulo 3. Supondo um documento textual contendo palavras de um dicionário com 3 termos possíveis, e um total de N palavras,

$$p_i = \frac{c_i}{N} \quad (2.14)$$

é a probabilidade de que o termo i seja encontrado nesse documento, supondo que tenha sido observado c_i vezes.

De forma análoga à distribuição binomial, onde $p_{falha} = 1 - p_{sucesso}$, para a distribuição multinomial, pode-se dizer que

$$\sum_{i \in k} p_i = 1. \quad (2.15)$$

Trata-se de uma decorrência direta dos axiomas da probabilidade. Há de se notar que esta mesma restrição está presente em variáveis aleatórias que obedecem a uma distribuição de Dirichlet, como mostrado na Equação 2.11.

2.3 Comparações entre Distribuições de Probabilidade

Dadas duas distribuições de probabilidade, deseja-se frequentemente avaliar a similaridade entre elas. Essa necessidade aparece principalmente em tarefas onde deseja-se estimar uma distribuição de probabilidade Q , que aproxime uma outra distribuição, P , chamada de *alvo*, ou *objetivo* [29].

Uma forma usual de se expressar essa similaridade é através da divergência de Kullback-Leibler, também mencionada na literatura como distância de Kullback-Leibler [29].

Sejam duas distribuições de probabilidade P e Q . Supondo que ambas as distribuições sejam discretas, a divergência de Kullback-Leibler pode ser escrita na forma

$$D_{KL}(P||Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)}. \quad (2.16)$$

Para o caso de P e Q serem contínuas, a divergência tem a forma

$$D_{KL}(P||Q) = \int_{-\infty}^{\infty} p(x) \log \frac{p(x)}{q(x)} dx, \quad (2.17)$$

onde $p(x)$ e $q(x)$ são as funções densidade de probabilidade associadas [32].

É importante notar que apesar de fazer as vezes de uma distância, a divergência de Kullback-Leibler não é uma distância propriamente dita. Em particular, ela é não simétrica. Ou seja,

$$D_{KL}(P\|Q) \neq D_{KL}(Q\|P), \quad (2.18)$$

além de não obedecer à desigualdade triangular [13].

2.4 Expectation Maximization

O algoritmo *Expectation-Maximization*, ou simplesmente *EM*, em sua forma geral, é largamente utilizado em cálculos iterativos para estimativas de máxima-verossimilhança em parâmetros para modelos probabilísticos de tópicos latentes. O algoritmo garantidamente converge, chegando a um valor estacionário para os parâmetros estimados após um número finito de iterações [39].

A principal aplicação desta técnica é em problemas conhecidos como *problemas de dados incompletos*, onde a obtenção de estimativas exatas da verossimilhança são problemáticas [39]. Para tais problemas, os dados podem ser divididos em dados *observados*, cujos valores são conhecidos após algum experimento, e *não observados* ou *latentes*, cujos valores devem ser inferidos a partir de algum procedimento.

O algoritmo *Expectation-Maximization* associa aos problemas de dados incompletos, *problemas de dados completos* correspondentes, ou seja, problemas onde todos os dados são observados. Para estes, espera-se que o cálculo da máxima verossimilhança seja mais fácil.

O algoritmo é composto de duas etapas:

- etapa-E, onde realiza-se uma estimativa dos parâmetros não observados, ou latentes, e a partir disso, computa-se uma estimativa de máxima verossimilhança, e
- etapa-M, onde computam-se os parâmetros não observados, ou latentes, a partir da verossimilhança encontrada na etapa-E.

Como diversos outros métodos para o cálculo de estimativas de máxima verossimilhança, o algoritmo de *Expectation-Maximization* é um algoritmo que busca encontrar zeros de funções. Assim, ele pode ser comparado com técnicas tradicionais como *Newton-Raphson* ou pontuação de Fisher [39].

Algoritmos de EM são considerados *não-Bayesianos*, dado que seu resultado final é uma distribuição de probabilidade sobre os tópicos latentes, e uma estimativa pontual do parâmetro desejado. É possível obter uma versão totalmente Bayesiana do algoritmo, onde a saída seja uma distribuição de probabilidade sobre todas as variáveis, sejam latentes

ou observáveis, sem a presença de estimativas pontuais. Esta variação é conhecida por *Variational Bayes* [5].

2.5 K-Means

Um exemplo de algoritmo *Expectation-Maximization*, atraente por sua simplicidade, é o *k-means*. Seu objetivo é agrupar n pontos em k grupos distintos, de forma a minimizar o erro quadrático médio entre o posicionamento encontrado e o posicionamento ótimo [30, 7].

Este algoritmo é uma simplificação de um algoritmo EM para estimativa de parâmetros para uma mistura de gaussianas. Neste caso, deseja-se estimar as k médias dos grupos que tem mais probabilidade de gerar os dados observados.

A hipótese é que os dados são originários de distribuições gaussianas com matrizes de covariância proporcionais à identidade. O que equivale a dizer que os dados são isotrópicos, ou circularmente simétricos. Sob essa hipótese, o posicionamento ótimo do centro de um grupo corresponde ao seu centróide [30].

Inicialmente, o algoritmo seleciona k pontos, $\mu_1, \mu_2, \dots, \mu_k$, aleatoriamente, ou através de qualquer outra heurística. A partir disso, para cada $x_j \in \mathbb{R}^n$ pertencente ao conjunto de dados, deseja-se escolher μ_{i^*} tal que

$$\|x_j - \mu_{i^*}\| < \|x_j - \mu_i\|, \quad i = \{1, \dots, k\}, \quad (2.19)$$

o que é equivalente a dizer que

$$\sum_{m=1}^n (x_{jm} - \mu_{i_m^*})^2 < \sum_{m=1}^n (x_{jm} - \mu_{im})^2, \quad i = \{1, \dots, k\}. \quad (2.20)$$

Ou seja, deseja-se minimizar o erro quadrático entre o centro encontrado, e o centro verdadeiro da distribuição. Esta atribuição corresponde à etapa-E de um algoritmo *Expectation-Maximization*.

A partir desta atribuição, para cada um dos k grupos, calcula-se

$$\mu_i = \frac{1}{|\mathbf{x}_i|} \sum_{x_j \in \mathbf{x}_i} x_j, \quad (2.21)$$

onde $\mathbf{x}_i$ é o conjunto formado por todos os pontos atribuídos ao grupo i .

Dado um conjunto de inicialização, o algoritmo garantidamente converge para o erro mínimo. Porém, não há garantias de que este mínimo seja o mínimo global [30, 7]. Dessa forma, a escolha dos k centros tem grande importância na qualidade do resultado final obtido.

De fato, existem algoritmos como o *k-means++* [4], dedicados a encontrar um conjunto satisfatório de centros de forma que se possa impor garantias sobre a qualidade do resultado.

É importante fazer uma distinção entre o *algoritmo k-means* e o *problema k-means*. O algoritmo nada mais é do que a solução mais popular para o problema de se agrupar um conjunto de pontos em $\mathbb{R}^n$ em torno de k centros, de forma a minimizar uma métrica de erro. A utilização de um algoritmo do tipo *Expectation-Maximization* para obtenção da solução advém do fato de o *problema k-means* ser um problema NP-difícil [30].

Capítulo 3

Classificadores Probabilísticos

Dado um conjunto de dados observáveis D , condicionados a variáveis latentes x , há duas formas principais de categorizar classificadores: os modelos generativos e discriminativos [51].

Para os modelos discriminativos, a questão de interesse é determinar

$$P(x|D),$$

ou seja, qual distribuição sobre a variável aleatória x que melhor se adequa aos dados observados. Com essa distribuição, é possível fazer previsões a respeito dos valores de x , quando novos dados são observados. Técnicas como *Support Vector Machines* (SVM), Redes Neurais Artificiais (RNA), entre outras, são exemplos de modelos discriminativos [51].

Modelos generativos, ao contrário dos discriminativos, são capazes de gerar instâncias dos dados observáveis [44, 51]. Isso, porque um modelo generativo busca obter a distribuição de probabilidade conjunta

$$P(x, D),$$

sobre os dados observáveis e variáveis latentes. É fácil verificar que especificadas esta distribuição, e $P(D|x)$, obtido usualmente através de etapas de treinamento, pode-se facilmente obter a mesma distribuição *a posteriori* utilizada por modelos discriminativos. Isso pode ser feito através do Teorema de Bayes (Equação 2.6).

Modelos generativos possuem algumas vantagens em relação aos modelos discriminativos [51], como:

1. quando se realiza aprendizado supervisionado, modelos generativos são capazes de tratar conjuntos de dados com anotações incompletas, ou então estender conjuntos de dados parcialmente anotados com resultados superiores aos discriminativos, que precisam de quantidade abundante de dados [44],

2. não há necessidade de determinação do número de classes *a priori*, uma vez que uma nova classe $X + 1$ pode ser introduzida apenas determinando $p(D|X + 1)$, independentemente das outras classes já modeladas, e
3. são capazes de tratar combinações de elementos do espaço dos dados de treinamento separadamente. Por exemplo, é possível desenvolver classificadores para faces, bigodes, chapéus, e então reconhecê-los em conjunto. Modelos discriminativos, por outro lado, devem ser expostos a todas as combinações possíveis na etapa de treinamento.

Por sua vez, modelos discriminativos apresentam, também, vantagens sobre os generativos:

1. têm limites flexíveis entre as classes [28, 51], permitindo melhor classificação para regiões do espaço de dados onde a distribuição *a posteriori* tem valores distantes dos extremos do intervalo entre 0 e 1,
2. uma vez realizado o treinamento, modelos discriminativos, em geral, são capazes de efetuar predições mais eficientemente. Modelos generativos, em contraste, costumam requerer soluções iterativas, e
3. quando os dados de treinamento são abundantes, possuem um poder preditivo maior que os modelos generativos equivalentes.

As características apresentadas estão mostradas resumidamente na Tabela 3.1. Além dos classificadores puramente discriminativos ou generativos, é também possível construir classificadores híbridos, utilizando modelos que abrangem características generativas e discriminativas [44, 28].

3.1 Análise de Semântica Latente

A técnica da Análise Semântica Latente, do inglês *Latent Semantic Analysis* (LSA), originalmente concebida no contexto da análise de documentos textuais, baseia-se no mapeamento de vetores de alta dimensionalidade para espaços de dimensionalidade reduzida, chamados “*Latent Semantic Space*”, ou *Espaço Semântico Latente* [15]. Apesar da motivação inicial, a técnica é aplicável a quaisquer conjuntos de dados discretos originários de algum conjunto $\mathbf{w} = \{w_1, w_2, \dots, w_n\}$.

Dada uma observação definida como um vetor D com elementos pertencentes a $\mathbf{w}$, chama-se um *vetor de frequências* uma função $f(D)$, tal que $f(w_i) = \#(w_i)$, ou seja, a quantidade de ocorrências da palavra w_i no vetor D .

Modelo:	Generativo	Discriminativo
Poder Preditivo	Bom, na presença de poucos dados	Alto na presença de muitos dados
Adição de novas classes	Sem necessidade de retreinamento	Necessidade de retreinamento
Combinações dos dados de treinamento	Treinamento separado	Treinamento conjunto
Necessidade de Soluções Iterativas	Geralmente sim	Geralmente não

Tabela 3.1: Resumo comparativo entre propriedades de modelos generativos e discriminativos.

Por exemplo, em um texto contendo cinco palavras, denotadas $w_1, w_2, \dots, w_5$, o vetor $v = (2, 3, 1, 6, 9)$ é um vetor de frequências válido, indicando duas ocorrências da palavra 1, três ocorrências da palavra 2, etc.

O mapeamento produzido pelo LSA resulta em um vetor chamado *vetor de conceitos*. Ele tem como objetivo capturar e explicitar relações semânticas entre os elementos, como por exemplo, o uso de sinônimos. Espera-se que duas palavras que são usadas intercambiavelmente sejam mapeadas para o mesmo conceito. A Figura 3.1 mostra um exemplo deste mapeamento, onde seis palavras são mapeadas para três conceitos.

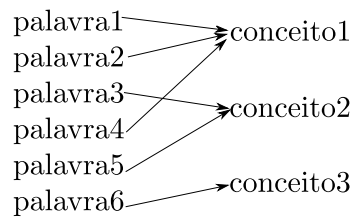


Figura 3.1: Exemplo de mapeamento fornecido pela técnica LSA. Neste exemplo, as palavras 1, 2 e 4 são associadas ao conceito 1, as palavras 3 e 5 ao conceito 2, e a palavra 6 ao conceito 3.

Para produzir o mapeamento, o LSA utiliza a técnica SVD, ou Singular Value Decomposition [15]. Um grupo de documentos é dado por $D = \{d_1, \dots, d_N\}$, composto por termos pertencentes a um vocabulário definido por $W = \{w_1, \dots, w_M\}$. Pode-se definir uma *tabela de co-ocorrências*, onde o posicionamento dos termos é desprezado. Essa tabela é uma matriz K de tamanho $N \times M$, onde o termo k_{ij} representa o número de vezes que o

termo w_j aparece no documento d_i .

De posse dessa tabela, pode-se exprimir a matrix K como

$$K = U\Sigma V^\top, \quad (3.1)$$

onde U é uma matriz unitária e $V^\top$ sua conjugada transposta.

Para um modelo de k tópicos latentes, escolhem-se os maiores k valores em Σ , enquanto todos os outros são forçados a zero. Assim, uma nova matriz $\tilde{\Sigma}$ é obtida. Utilizando-se as mesmas matrizes U e $V^\top$ pode-se aproximar a matriz original nesse novo espaço como

$$K = U\tilde{\Sigma}V^\top. \quad (3.2)$$

3.2 Análise de Semântica Latente Probabilística

Análise Semântica Latente Probabilística, do inglês *Probabilistic Latent Semantic Analysis* (PLSA), também conhecida por PLSI (*Probabilistic Latent Semantic Indexing*) é uma técnica inspirada no LSA, porém construída sobre um ferramental estatístico mais robusto. A necessidade da construção surge, segundo Hofmann [26], do fato de que o LSA sofre de falta de embasamento teórico adequado, além de não ser capaz de capturar construções no domínio textual como polissemia, ou seja, uma mesma palavra dotada de vários significados.

O ponto de partida desta técnica é um modelo estatístico conhecido por *modelo de aspecto*, ou *aspect model*. Este modelo associa a cada elemento observável uma *classe*, ou *conceito*, não observável. Cada conceito é denotado por $z \in Z = \{z_1, \dots, z_k\}$. Para um grupo de documentos $D = \{d_1, \dots, d_N\}$ com termos pertencentes a um vocabulário definido por $W = \{w_1, \dots, w_M\}$, pode-se definir a probabilidade conjunta de uma variável aleatória sobre $D \times W$. Supondo-se independência condicional entre w e d , essa probabilidade pode ser escrita como

$$P(d, w) = P(d)P(w|d), \quad (3.3)$$

sendo que

$$P(w|d) = \sum_{z \in Z} P(w|z)P(z|d), \quad (3.4)$$

ou de forma equivalente, como

$$P(d, w) = \sum_{z \in Z} P(z)P(d|z)P(w|z). \quad (3.5)$$

Ambas as formulações correspondem a redes Bayesianas válidas para representar $P(d, w)$. Na formulação da Equação 3.4, assume-se a independência condicional de d e w dado z ,

tendo-se uma distribuição *a priori* sobre d . A rede obtida neste cenário é mostrada na Figura 3.2.

Na formulação da Equação 3.5, a mesma hipótese de independência condicional é feita. Porém, a distribuição *a priori* utilizada distribui-se sobre z . A representação gráfica desta formulação é mostrada na Figura 3.3. A segunda formulação tem a vantagem de ser simétrica em relação ao termo latente z .

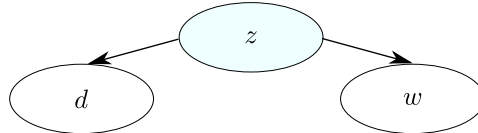


Figura 3.2: Rede Bayesiana equivalente à formulação da Equação 3.4. Os nós d e w correspondem aos dados observáveis, enquanto o nó z corresponde às variáveis latentes.



Figura 3.3: Rede Bayesiana equivalente à formulação da Equação 3.5. Os nós d e w correspondem aos dados observáveis, enquanto o nó z corresponde às variáveis latentes.

Como espera-se que $|Z| < |W|$, o PLSA também efetua, tal qual o LSA, uma redução de dimensionalidade. Em uma analogia com o LSA, pode-se dizer que as probabilidades $P(z_i)$, para cada conceito i , correspondem aos valores singulares da decomposição SVD utilizada no LSA. Denotando essa matriz por $\hat{\Sigma}$, e definindo as matrizes

$$\begin{aligned}\hat{\mathbf{U}} &= (P(d_i|z_k))_{i,k}, \\ \hat{\mathbf{V}} &= (P(w_i|z_k))_{i,k},\end{aligned}$$

é possível escrever o modelo de probabilidades $\mathbf{P}$, correspondente a $P(d,w)$ como

$$\mathbf{P} = \hat{\mathbf{U}}\hat{\Sigma}\hat{\mathbf{V}}. \quad (3.6)$$

Isso é equivalente ao descrito na Equação 3.5, porém, em relação ao LSA, a matriz $\mathbf{P}$ apresenta uma distribuição de probabilidades bem definida de onde se podem extrair significados, ao contrário dos valores singulares obtidos pelo SVD.

A etapa de treinamento tem como objetivo obter o valor de $P(z|d,w)$, ou seja, a distribuição de probabilidades sobre os termos e documentos presentes. O treinamento pode ser feito através da técnica *Expectation-Maximization* [26].

Do Teorema de Bayes, pode-se derivar a expressão [25]

$$P(z|d, w) = \frac{P(z)P(d|z)P(w|z)}{\sum_{z' \in Z} P(z')P(d|z')P(w|z')}, \quad (3.7)$$

utilizada para calcular a etapa *Expectation*, bem como as expressões

$$P(w|z) \propto \sum_{d \in D} n(d, w)P(z|d, w), \quad (3.8)$$

$$P(d|z) \propto \sum_{w \in W} n(d, w)P(z|d, w), \quad (3.9)$$

$$P(z) \propto \sum_{d \in D} \sum_{w \in W} n(d, w)P(z|d, w), \quad (3.10)$$

utilizadas na etapa *Maximization*.

Nestas expressões, $n(d, w)$ representa a probabilidade conjunta sobre D e W , e equivale à tabela de frequências obtida pela contagem dos termos nos documentos. A menos de uma constante, essa tabela corresponde a uma distribuição multinomial, detalhada na Seção 2.2.

3.3 Alocação Latente de Dirichlet

A técnica conhecida como LDA, Alocação Latente de Dirichlet, (do inglês *Latent Dirichlet Allocation*) é aplicável a quaisquer conjuntos de dados discretos. Foi proposta originalmente por Blei et al. [8], no contexto de classificação de conjuntos de documentos textuais.

O LDA inspira-se no PLSA, fornecendo uma rede Bayesiana similar, como mostrado na Figura 3.4. Porém, ao contrário do PLSA, o LDA fornece um modelo generativo bem definido [8, 53]. No PLSA, para cada documento, há uma distribuição $p(z|d)$ que define a proporção dos tópicos nos documentos. No entanto, d é uma variável aleatória multinomial, que tem tantos valores quanto documentos previamente analisados. De outra forma, a distribuição de tópicos em um documento depende diretamente do documento em si. Não há uma maneira óbvia de atribuir probabilidades a documentos não vistos [8]. Disso resultam problemas de *overfitting* [19].

Para superar essa limitação, o LDA define uma variável aleatória de Dirichlet $\theta \sim \text{Dir}(\alpha)$, onde α e β são os parâmetros da distribuição. Dessa forma, é possível gerar um novo documento, definindo-se a mistura de tópicos que o mesmo deve conter.

Supondo que um dado documento contém apenas um tópico latente, tem-se uma equivalência entre os modelos LDA e PLSA [8]. Isso pode ser facilmente visto comparando-se as Figuras 3.2 e 3.4.

A ideia central do método é que um dado documento é composto por um conjunto de tópicos latentes. A cada tópico é associada uma distribuição de probabilidade sobre as palavras do vocabulário. O processo de geração de um novo texto é descrito no Algoritmo 3.1.

Para a aplicação da técnica, deve-se definir as seguintes entidades:

1. Palavras, denotadas por w , são as unidades básicas, discretas, do corpo de dados que se pretende analisar. Cada palavra w pertence a um vocabulário contendo todas as palavras possíveis, e é usualmente denotada por um vetor de norma unitária, e dimensão igual à cardinalidade do vocabulário.
2. Um documento, denotado por $\mathbf{w} = (w_0, w_1, \dots, w_N)$, é uma sequência de N palavras.
3. Um *corpus*, é um conjunto de documentos, denotado por $D = (\mathbf{w}_0, \mathbf{w}_1, \dots, \mathbf{w}_m)$.
4. Um tópico latente z é uma variável não observável à qual se associa uma distribuição que determina uma dada frequência de palavras. A cada *corpus*, é associado um conjunto $\mathbf{z} = \{z_1, \dots, z_n\}$ cuja proporção é determinada por uma distribuição de Dirichlet.

Ou seja, ao se determinar os parâmetros de um modelo LDA define-se uma proporção entre todos os tópicos latentes e em outro nível, uma proporção entre as palavras em um dado tópico.

Algoritmo 3.1 Algoritmo utilizado para gerar um novo documento através do LDA.

Entrada: Os parâmetros do modelo, α e β . O número de palavras N , ou um parâmetro ξ .

Saída: Um vetor $\mathbf{w}$ contendo as palavras do documento

$\mathbf{x} \leftarrow \emptyset$

Escolher $N \sim \text{Poisson}(\xi)$ (ou fixo), o número de palavras no documento.

Amostrar $\theta \sim \text{Dir}(\alpha)$. Esta etapa pode ser entendida como a geração das proporções esperadas entre os tópicos z de um documento.

for all $n \in \{1, \dots, N\}$ **do**

Amostrar um tópico $z_n \sim \text{Multinomial}(\theta)$

Amostrar $w_n \sim p(w_n | z_n, \beta)$. Aqui, β é um parâmetro do modelo, e p uma distribuição multinomial, condicionada ao tópico

$\mathbf{W} \leftarrow \mathbf{W} \cup \{w_n\}$

end for

A etapa de treinamento do LDA consiste na determinação dos parâmetros α e β do modelo. O parâmetro β é uma variável aleatória multinomial, que pode ser representada

por uma matriz $k \times V$. Para um conjunto de k tópicos e V palavras, tem-se que

$$\beta_{ij} = p(w^j = 1 | z^i = 1). \quad (3.11)$$

Para um dado documento, β pode ser aproximado por uma tabela de frequências de cada termo, da mesma forma que o PLSA.

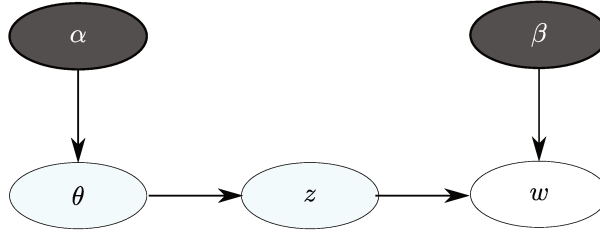


Figura 3.4: Rede Bayesiana representando o modelo LDA. O nó w corresponde aos dados observáveis. α e β são os parâmetros do modelo. Os outros nós, θ e z , são variáveis latentes do modelo.

O LDA assume que a ordem das palavras e tópicos em um dado documento é *intercambiável*. Um conjunto finito de variáveis aleatórias $z_1, \dots, z_N$ é dito intercambiável se sua probabilidade conjunta é invariante em relação à permutação [8]. Sendo π uma permutação de inteiros de 1 a N , isso equivale a dizer que

$$p(z_1, \dots, z_N) = p(z_{\pi(1)}, \dots, z_{\pi(N)}). \quad (3.12)$$

Esta suposição descarta informações estruturais a respeito do posicionamento relativo dos elementos do documento, mas é importante para simplificar o modelo matemático utilizado.

A informação sobre grupos de palavras que costumam ocorrer em adjacência pode ser incorporada adicionado ao vocabulário *digramas*, *trigramas*, ou de forma geral, *n-gramas*. Chama-se um *n-grama* uma sequência específica de n termos. De forma geral, a utilização de *n-gramas* em modelos estocásticos leva a resultados mais precisos [47].

Treinamento e Inferência

Para um dado *corpus*, a etapa de treinamento define os parâmetros α e β que melhor definem os documentos a ele pertencentes. Com estes parâmetros, é também possível realizar inferência Bayesiana, a fim de classificar novos documentos.

Para a classificação, deve-se obter a distribuição *a posteriori* dos tópicos latentes, dada por

$$p(\theta, z | w, \alpha, \beta) = \frac{p(\theta, z, w | \alpha, \beta)}{p(w | \alpha, \beta)}. \quad (3.13)$$

Esta equação supõe que são conhecidos os parâmetros α e β do modelo e as palavras $\mathbf{w}$ de um documento ao qual o modelo é exposto.

No entanto, a inferência exata sobre essa distribuição é computacionalmente um problema de complexidade exponencial [8, 16]. Assim sendo, deve-se utilizar algum algoritmo aproximado para sua obtenção. Uma alternativa, é representar a rede Bayesiana por um modelo alternativo sobre um conjunto de parâmetros diferentes, que aproxime a distribuição desejada ao mesmo tempo que produz uma distribuição *a posteriori* computável em tempo polinomial.

O modelo simplificado utilizado é mostrado na Figura 3.5. A ele correspondem uma família de distribuições q , indexadas pelos parâmetros auxiliares γ e ϕ , da forma

$$q(\theta, z|\gamma, \phi) = q(\theta|\gamma) \prod_{n=1}^N q(z_n|\phi_n). \quad (3.14)$$

A distribuição q utilizada deve aproximar a distribuição original, p , suficientemente bem. Para isso, a divergência de Kullback-Leibler (ver Seção 2.3), dada por

$$D_{KL}(q(\theta, z|\gamma, \phi) || p(\theta, z|w, \alpha, \beta)) = \int_{-\infty}^{\infty} q(\theta, z|\gamma, \phi) \log \frac{p(\theta, z|w, \alpha, \beta)}{q(\theta, z|\gamma, \phi)} dx, \quad (3.15)$$

deve ser minimizada. A tarefa da inferência passa a ser encontrar os parâmetros γ e ϕ que minimizem essa divergência. Pode-se demonstrar que estes parâmetros são [8]

$$\phi_{ni} \propto \beta_{iw_n} \exp\{E_q[\log(\theta_i)|\gamma]\}, e \quad (3.16)$$

$$\gamma_i = \alpha_i + \sum_{n=1}^N \theta_{ni}. \quad (3.17)$$

O termo $E_q[\log(\theta_i)|\gamma]$ não tem solução analítica, mas pode ser escrito como

$$E_q[\log(\theta_i)|\gamma] = \Psi(\gamma_i) - \Psi\left(\sum_{j=1}^k \gamma_j\right), \quad (3.18)$$

onde Ψ é a primeira derivada do logaritmo da função gamma [8]. Esta função, por sua vez, é computável através de uma série de Taylor [2]. Além disso estas equações são dependentes entre si, e podem ser avaliadas utilizando-se um método iterativo, como é detalhado no Algoritmo 3.2.

Um método similar ao utilizado para a inferência pode ser utilizado para o treinamento. Busca-se a obtenção dos parâmetros α e β do modelo através de um algoritmo do tipo *Expectation-Maximization*.

Os parâmetros alternativos γ e ϕ podem ser utilizados para determinar uma distribuição *a posteriori* em relação ao modelo. Em seguida, determinam-se α e β de forma a maximizar a verossimilhança com essa distribuição.

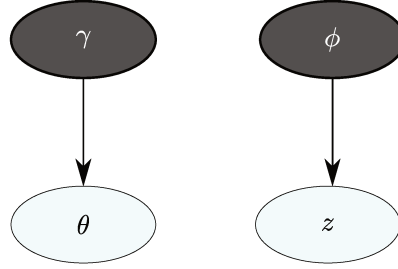


Figura 3.5: Rede Bayesiana representando o modelo LDA simplificado, de acordo com a Equação 3.14. Não há variáveis observáveis. γ e ϕ são os parâmetros do modelo, e z e θ variáveis latentes.

Algoritmo 3.2 Algoritmo de Inferência Variável para o LDA.

Entrada:

Saída: As variáveis ϕ e γ .

```

for all  $i \in \{1, \dots, k\}$  do
   $\gamma_i \leftarrow \alpha_i + \frac{1}{k}$ 
  for all  $n \in \{1, \dots, N\}$  do
     $\phi_{ni} \leftarrow \frac{1}{k}$ 
  end for
end for
while Não houver convergência do
  for all  $n \in \{1, \dots, N\}$  do
    for all  $i \in \{1, \dots, k\}$  do
       $\phi_{ni}^{t+1} \leftarrow \beta_{iw} \exp(E_q[\log(\theta_i) | \gamma])$ 
      Normalizar  $\phi_{ni}^{t+1}$ 
    end for
  end for
   $\gamma_{t+1} \leftarrow \alpha + \sum_{n=1}^N \phi_{ni}^{t+1}$ 
end while

```

O parâmetro β pode ser determinado analiticamente, e é dado por

$$\beta_{ij} \propto \sum_{d=1}^M \sum_{n=1}^{N_d} \phi^{dni} w_{dn}^j. \quad (3.19)$$

O parâmetro α pode ser determinado eficientemente através de métodos numéricos [8].

3.4 LDA Suavizado

Um problema encontrado ao se utilizar o LDA resulta diretamente do grande vocabulário de termos normalmente utilizado nos documentos. Considerando, por exemplo, documentos em linguagem natural, é provável que novos documentos contenham novos termos não encontrados previamente em nenhum documento do corpo de treinamento. Esses termos têm probabilidade nula. (É importante notar que probabilidade nula não implica evento impossível [35].) Como o modelo de inferência multiplica a probabilidade condicional de todos os termos (consequência direta da Equação 3.13 e da hipótese de intercambialidade dos termos em um documento), documentos contendo esses termos terão probabilidade invariavelmente nula.

Para tratar esse problema, pode-se modelar o parâmetro multinomial β como tendo se originado de uma distribuição de Dirichlet η , como mostrado na Figura 3.6. O modelo obtido possui formulação matemática similar. Como diferença principal, porém, tem-se que este modelo não tem mais componentes frequentistas, dado que β agora não é mais modelado como uma tabela de frequências, e sim como uma distribuição multinomial que melhor se adapte aos dados e à distribuição *a priori* em η .

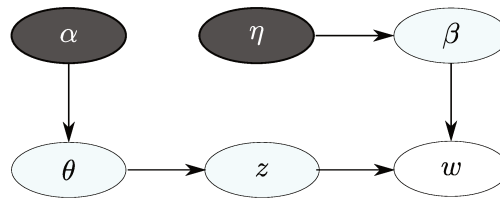


Figura 3.6: Rede Bayesiana representando o modelo LDA suavizado. Os nó w corresponde aos dados observáveis, enquanto os nós α e η representam os parâmetros do modelo.

Capítulo 4

Descritores de Imagens

As técnicas apresentadas no Capítulo 3 aplicam-se a documentos textuais. Para o domínio das imagens é necessário lançar mão de alguma técnica que permita a expressão de uma imagem através de elementos discretos. Neste capítulo serão apresentadas representações concisas de imagens conhecidas como *descritores de imagens*. A partir desta discretização, é possível mapear uma imagem para um documento textual.

4.1 Descritores de Imagens Simples

Dada uma imagem I , representada como uma matriz de *pixels*, deseja-se construir uma representação alternativa que melhor descreva suas propriedades.

Para esta tarefa pode-se utilizar um vetor de características: um ponto $\mathbf{v}_i = (v_1, \dots, v_n) \in \mathbb{R}^n$, capaz de expressar alguma propriedade de uma imagem, como cor, forma, textura, etc. Ao conjunto formado por um vetor de características e uma função $\delta : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$ é dado o nome de *descriptor de imagem* [14]. A função δ mede o grau de similaridade entre dois descritores distintos. Por se tratar de uma função análoga a uma distância, quanto maior a similaridade entre dois descritores menor será o valor de δ .

Descritores de imagem podem ser agrupados em três categorias principais [3]:

- **Métodos Globais**, que descrevem a distribuição global de características numa imagem, sem considerar características localmente relevantes.
- **Métodos Regionais**, que decompõem a imagem, através de processos de segmentação de acordo com as características da imagem.
- **Métodos Locais**, que localizam pontos importantes e constroem descritores em torno deles.

De todos os descritores de imagem possíveis, há interesse particularmente nos chamados invariantes em relação a algum tipo de propriedade. Encontrar descritores invariantes a uma propriedade significa que será possível associar esta imagem com outra na qual as mesmas regiões são encontradas, porém com variações nesta propriedade [38]. Entre as propriedades invariantes usuais, destacam-se o ponto de vista, escala e rotação.

Um descritor simples descrito na literatura [1] e utilizado neste trabalho consiste em separar a imagem em janelas de tamanho fixo. A localização dos pontos em torno dos quais se deseja estabelecer as janelas pode ser obtida através de uma varredura em grade na imagem. Nesse processo, percorre-se a imagem em incrementos fixos em x e y . Alternativamente, pode-se utilizar qualquer algoritmo de detecção de pontos característicos, como os próprios SIFT ou SURF, descritos nas Seções 4.3 e 4.4.

Para cada um destes pontos, seleciona-se aleatoriamente uma escala e cria-se uma janela quadrada centrada no ponto de interesse, com lados iguais a duas vezes a escala utilizada. Esta janela é reduzida para uma de tamanho fixo, por exemplo, 11×11 . A imagem I é representada como

$$I = \{w_1, \dots, w_n\}, \quad (4.1)$$

onde cada w_i é um vetor contendo o valor da janela correspondente em cada um de seus pontos. Supondo uma imagem em tons de cinza, dividida em janelas 11×11 , cada w_i é um vetor de 121 dimensões, onde cada elemento contém o valor da intensidade em cada um dos pixels da janela. Neste trabalho, esse descritor simples será referido como *Janelas* ou *Patches*.

4.2 BIC

O BIC (Border/Interior Pixel Classification) é um descritor de imagens baseado em cores [50]. A imagem sofre uma transformação inicial na qual é quantizada, resultando em uma versão com 64 classes de cor.

Em seguida, o algoritmo classifica todos os pontos da imagem em duas classes: borda e interior. Esta classificação é feita através de uma comparação com quatro de seus vizinhos (superior, inferior, esquerdo, e direito). A comparação com oito vizinhos é desnecessária, pois adiciona complexidade sem trazer ganhos adicionais [50].

Um ponto é classificado como interior, se seus quatro vizinhos tem a mesma cor quantizada. Caso pelo menos um de seus quatro vizinhos tenha uma cor quantizada diferente, o pixel é considerado como borda.

Para cada cor presente na imagem, constroem-se dois histogramas, sendo um para pontos de interior e outro para pontos de borda. Cada um dos histogramas contém uma classe para cada cor quantizada, totalizando 128 classes: 64 para pontos de borda e 64 para pontos de interior.

Para comparar dois descritores pode-se utilizar a métrica $dLog$. Dados dois descritores q e d , pode-se escrever

$$dLog(q, d) = |f(q[i]) - f(d[i])|, \quad (4.2)$$

onde

$$f(x) = \begin{cases} 0, & \text{se } x = 0, \\ 1, & \text{se } 0 < x \leq 1, \\ \lceil \log_2 x \rceil + 1, & \text{se } x > 1. \end{cases} \quad (4.3)$$

A utilização de comparações logarítmicas, através da função $dLog$, em contraste com métricas como L_1 e L_2 tem a vantagem de minimizar distorções, onde uma dada cor quantizada domina o histograma, mesmo sem participar de regiões semanticamente relevantes [50].

Os histogramas podem ser armazenados diretamente em escala logarítmica, o que simplifica a comparação posterior das distâncias. Além disso, a utilização de escalas logarítmicas utiliza apenas metade do espaço necessário para o armazenamento do histograma comum.

4.3 SIFT

O SIFT [38], do inglês *Scale-Invariant Feature Transform* é um algoritmo de detecção e descrição de pontos característicos em uma imagem. Os descritores encontrados pelo SIFT são vetores, normalmente 128-dimensionais, que representam características locais da imagem, invariantes à escala, rotação e iluminação, além de pequenas alterações de ponto de vista e rotação 3D. No algoritmo, as características são encontradas através de uso de filtros em vários estágios. No primeiro estágio, buscam-se regiões da imagem que sejam consideradas estáveis em relação às características invariantes.

Isso pode ser obtido através da busca por máximos e mínimos locais em uma função contínua das escalas, conhecida como espaço-escala [54]. Em geral, o espaço-escala utilizado corresponde a um filtro gaussiano [37]. Dessa forma, para uma imagem $I(x, y)$, pode-se escrever o espaço escala L , como

$$L(x, y, \sigma) = \left(\frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}} \right) * I(x, y), \quad (4.4)$$

onde $*$ é a operação de convolução.

Pontos estáveis podem ser encontrados através dos mínimos e máximos locais da função *diferença de gaussianas*, que corresponde à diferença entre dois pontos próximos no espaço escala [37, 38].

A cada ponto encontrado, atribui-se uma escala e orientação. A escala é aquela na qual o ponto estável foi encontrado. Por simplicidade, chama-se o espaço-escala L nesta escala em particular L_σ . A orientação é um vetor gradiente com componentes $m(x,y)$ e $\theta(x,y)$, correspondentes à magnitude e direção, dadas por

$$m(x,y) = \sqrt{(L_\sigma(x+1,y) - L_\sigma(x-1,y))^2 + (L_\sigma(x,y+1) - L_\sigma(x,y-1))^2}, \quad (4.5)$$

$$\theta(x,y) = \tan^{-1} \left(\frac{L_\sigma(x,y+1) - L_\sigma(x,y-1)}{L_\sigma(x+1,y) - L_\sigma(x-1,y)} \right). \quad (4.6)$$

É formado um histograma de orientações a partir de pontos na vizinhança do ponto localizado. Neste histograma, cada componente de orientação é ponderada por uma função de sua magnitude.

A partir disso, as informações de cada ponto são utilizadas para gerar um vetor de características. Este vetor representa o histograma de orientações na vizinhança do ponto de interesse. A vizinhança do ponto é dividida em n regiões de igual tamanho. Em cada uma delas, projetam-se os gradientes em um histograma de oito posições. A Figura 4.1 detalha esse processo, onde os gradientes obtidos a partir de uma janela 8×8 são acumulados em oito direções de suporte.

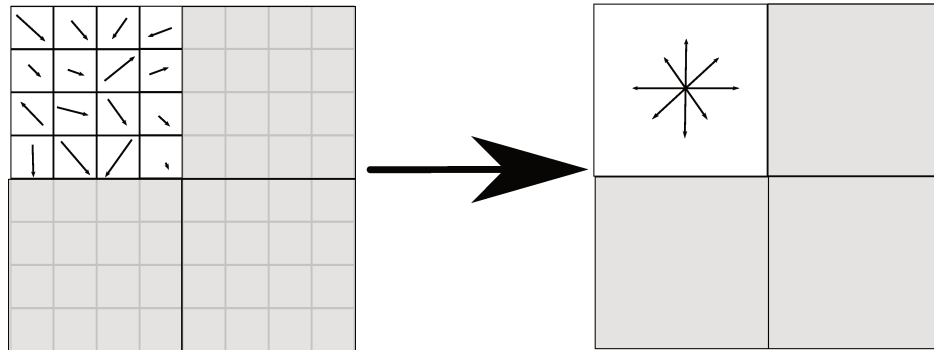


Figura 4.1: Obtenção dos vetores de características nos pontos de interesse. Os gradientes são acumulados em um histograma de oito posições. Nesta imagem, 4 histogramas são criados, totalizando um vetor de $4 \times 8 = 32$ dimensões. Por simplicidade, apenas um dos histogramas é mostrado. Em sua execução usual, o SIFT cria 16 histogramas, totalizando 128 dimensões.

Em uma última etapa, transformações são aplicadas para uma maior robustez das características. Para aumentar a invariância à iluminação, são selecionados apenas os gradientes que forem maior que um determinado limiar definido em relação ao maior

gradiente encontrado. Para a obtenção de invariância à rotação, a direção do gradiente é ajustada em relação à uma orientação comum.

Após encontradas, as características podem ser localizadas em outras imagens nas quais supostamente estejam presentes, através de uma busca por vizinhos mais próximos. Para um dado objeto na imagem original, cerca de três características encontradas resultam em grandes chances de se tratar de uma ocorrência do objeto em uma cena [38]. A Figura 4.2 demonstra um exemplo de aplicação do algoritmo. Os pontos de interesse são mostrados na imagem à direita. Para cada ponto, são representadas a magnitude e orientação principal do vetor de características. As Figuras 4.4, 4.3 e 4.5 mostram o algoritmo sendo utilizado para o casamento de pontos em imagens distintas. Note que é possível localizar objetos em pose e iluminação distintas.



Figura 4.2: A imagem à esquerda é a imagem original. À direita, pontos de interesse encontrados pelo SIFT, com suas magnitudes e orientações dominantes.

Este algoritmo tem sido utilizado para fornecer pontos característicos locais em diversos trabalhos [3, 22]. Variações do SIFT são encontradas na literatura, como o PCA-SIFT [31]. A técnica difere do SIFT tradicional na computação final dos histogramas a partir dos gradientes. Em vez do processo previamente descrito, o PCA (Principal Component Analysis) é aplicado aos gradientes a fim de selecionar apenas as direções que mais contêm informações.



Figura 4.3: Pontos encontrados pelo SIFT em uma imagem contendo apenas um objeto e seus casamentos em uma imagem contendo uma cena complexa da qual faz parte o objeto original. O objeto encontra-se na mesma pose em ambas as imagens.

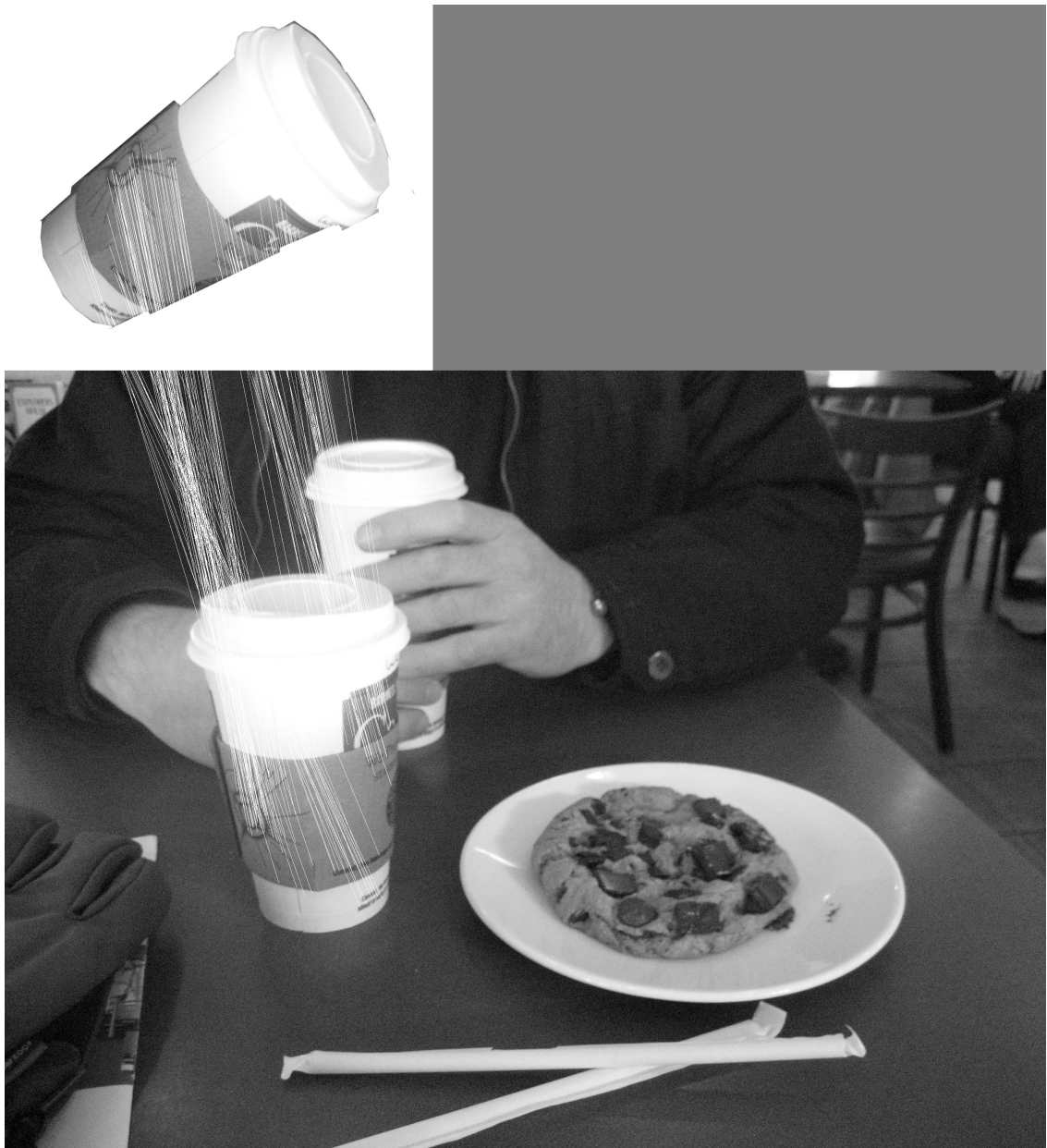


Figura 4.4: Pontos encontrados pelo SIFT em uma imagem contendo apenas um objeto e seus casamentos em uma imagem contendo uma cena complexa da qual faz parte o objeto original. O objeto encontra-se em poses distintas em cada uma das imagens.

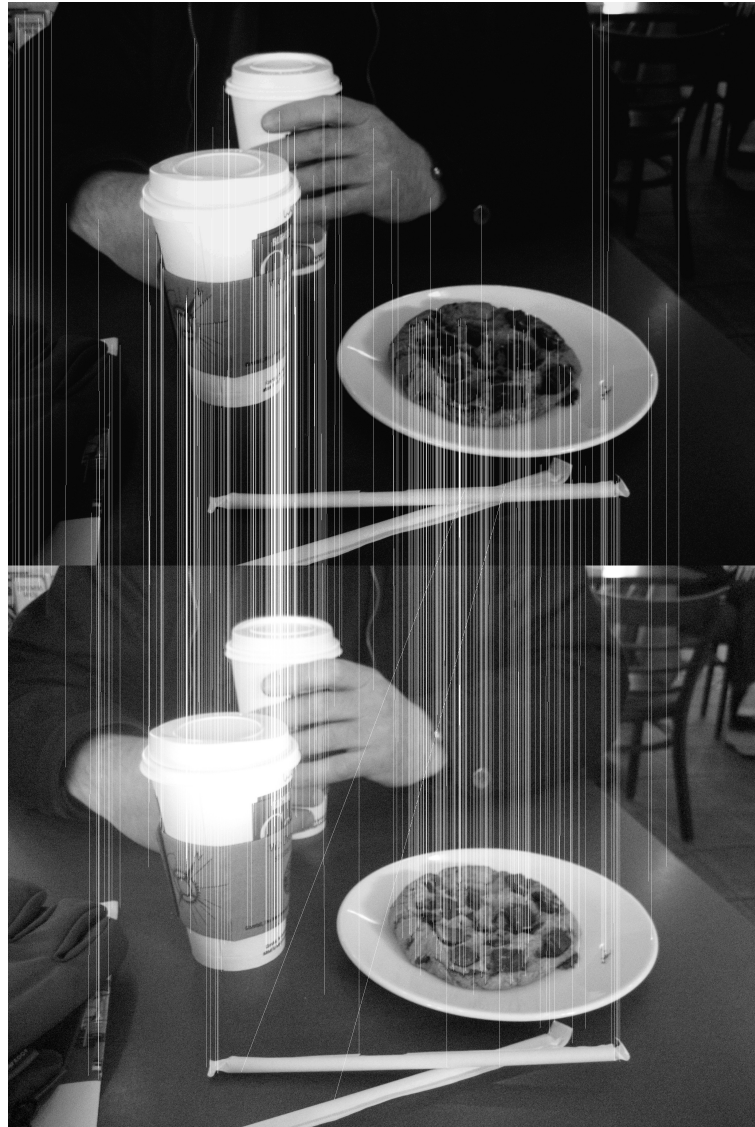


Figura 4.5: Pontos encontrados pelo SIFT e seus casamentos em duas imagens idênticas, exceto pelas condições de iluminação, artificialmente modificadas.

4.4 SURF

SURF (*Speeded Up Robust Features*) é um descritor de imagens invariante a mudanças de escala e rotação [6], que faz uso intensivo de imagens integrais.

Para um dado ponto $\mathbf{x} = (x, y)$, define-se a imagem integral em relação a $\mathbf{x}$ como a soma de todos os pontos interiores à região retangular formada entre a origem e o ponto $\mathbf{x}$:

$$S_x = \sum_{i=0}^{i \leq x} \sum_{j=0}^{j \leq y} I(i, j). \quad (4.7)$$

O uso de imagens integrais permite que se compute o somatório de todas as intensidades no interior de uma região retangular de forma eficiente. Apenas 3 operações de adição e quatro acessos à memória são necessários. Dado um sub-retângulo qualquer de uma imagem, caracterizado pelos pontos A , B , C e D dispostos como na Figura 4.6, o somatório das intensidades nesta região é dado por

$$\sum_{I_{A,B,C,D}} = S_A - S_B - S_C + S_D. \quad (4.8)$$

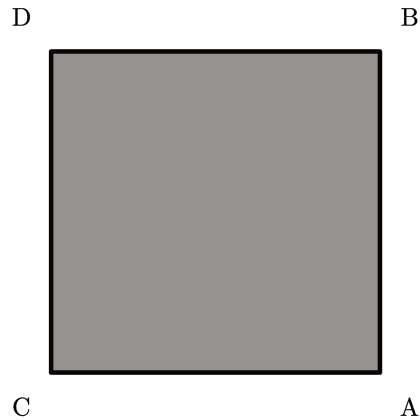


Figura 4.6: Disposição dos pontos para o cálculo de uma Imagem Integral.

Esta propriedade torna possível a computação de convoluções no espaço-escala sem que se compute uma pirâmide gaussiana explicitamente. Todas as operações podem ser feitas na imagem original, aumentando-se o tamanho do filtro, em vez de diminuir o tamanho da imagem. Como um exemplo, um filtro de tamanho 9×9 corresponde ao nível $\sigma = 1.2$ do espaço-escala, enquanto um filtro 15×15 , a $\sigma = 2$. Além desta propriedade, como não há a necessidade de subamostragem, efeitos de aliasing não são observados [6].

Para encontrar pontos de interesse estáveis, o algoritmo utiliza-se de uma matriz Hessiana, uma matriz contendo as derivadas parciais de segunda ordem de uma dada função. [6].

Para o caso do SURF, utiliza-se a matriz hessiana composta por filtros gaussianos, $\mathcal{H}$, dada por

$$\mathcal{H}(x, \sigma) = \begin{bmatrix} L_{xx}(x, \sigma) & L_{xy}(x, \sigma) \\ L_{xy}(x, \sigma) & L_{yy}(x, \sigma) \end{bmatrix}, \quad (4.9)$$

onde os termos $L(x, \sigma)$ representam a segunda derivada da função de convolução da imagem I no ponto x , com a função gaussiana com $G(\sigma)$. O método aproxima essas funções através de filtros de média, aproximações do filtro gaussiano com baixo custo computacional, e acurácia similar ao original [6].

Os pontos de interesse são encontrados através da localização de máximos locais em uma vizinhança de $3 \times 3 \times 3$, em relação a cada ponto. O descritor local produzido pelo algoritmo captura a informação das intensidades na vizinhança do ponto de interesse. Porém, ao contrário do uso de gradientes, como em outros métodos como o SIFT (Ver Seção 4.3), o SURF utiliza wavelets Haar nas direções x e y . Além disso, o descritor possui apenas 64 dimensões, o que acelera o processo de casamento, além de mitigar efeitos relacionados a alta dimensionalidade [23].

Uma versão do SURF chamada U-SURF pode ser utilizada para casos onde não se deseja um descritor necessariamente invariante à rotação. O U-SURF pode ser extraído e casado de forma mais eficiente que o SURF convencional [6].

A Figura 4.7 demonstra um exemplo de aplicação do algoritmo. Os pontos de interesse são mostrados na imagem à direita. As Figuras 4.8, 4.9 e 4.10 mostram o algoritmo sendo utilizado para o casamento de pontos em imagens distintas. Note que é possível localizar objetos em pose e iluminação distintas.



Figura 4.7: A imagem à esquerda é a imagem original. À direita, pontos de interesse encontrados pelo SURF.



Figura 4.8: Pontos encontrados pelo SURF em uma imagem contendo apenas um objeto e seus casamentos em uma imagem contendo uma cena complexa da qual faz parte o objeto original. O objeto encontra-se na mesma pose em ambas as imagens.

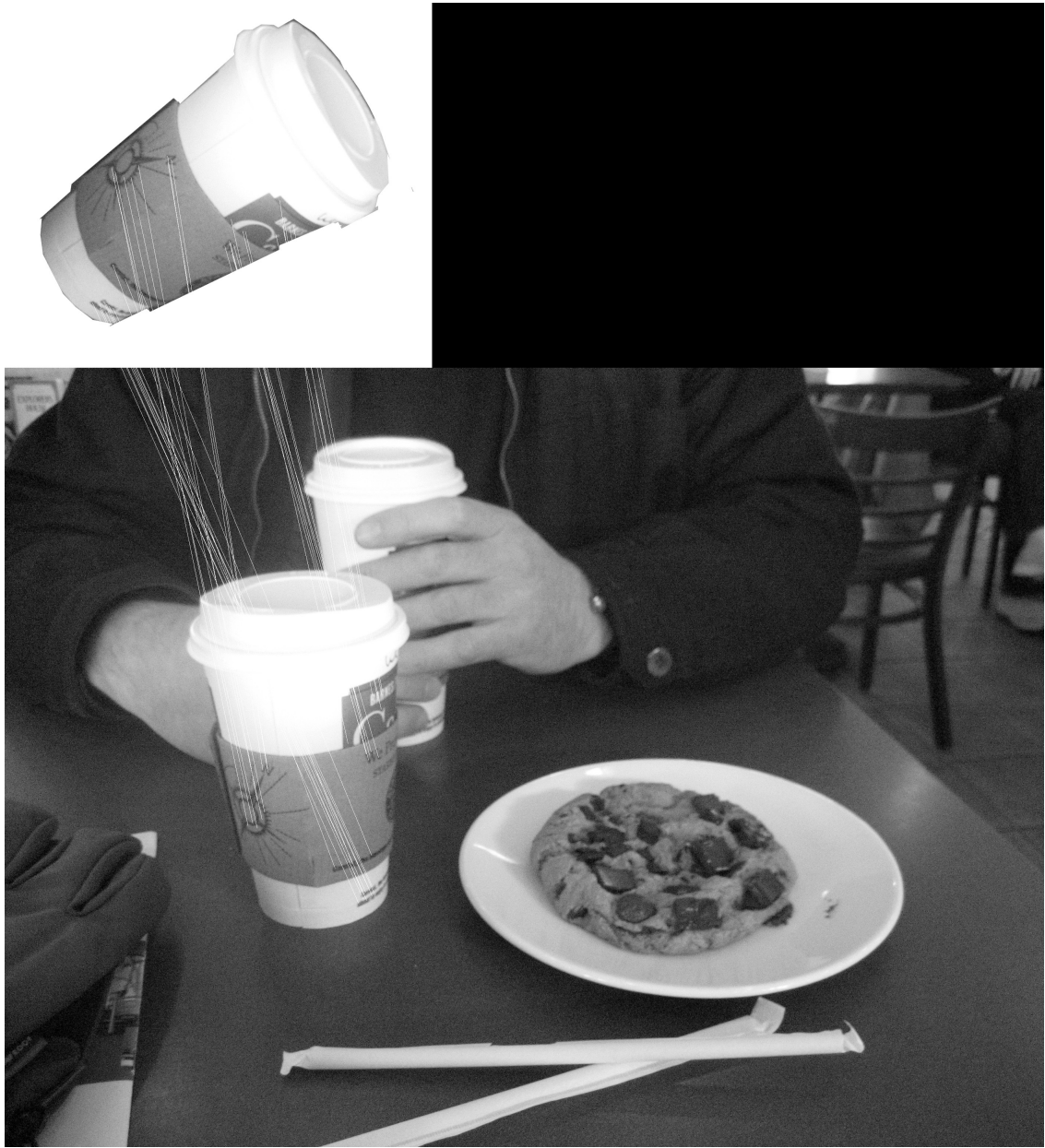


Figura 4.9: Pontos encontrados pelo SURF em uma imagem contendo apenas um objeto e seus casamentos em uma imagem contendo uma cena complexa da qual faz parte o objeto original. O objeto encontra-se em poses distintas em cada uma das imagens.



Figura 4.10: Pontos encontrados pelo SURF e seus casamentos em duas imagens idênticas, exceto pelas condições de iluminação, artificialmente modificadas.

Capítulo 5

Experimentos

5.1 Bases de Imagens

Este trabalho avaliou a aplicação do método *Latent Dirichlet Allocation*, a imagens. Nos experimentos, as bases são divididas entre um conjunto de *treinamento* e um conjunto de *testes*. Em cada um deles, há n categorias distintas.

Treinam-se n modelos LDA, um para cada categoria do conjunto de treinamento. A cada imagem I_t do conjunto de testes, deseja-se associar uma categoria do conjunto de treinamento.

O método foi executado em duas bases de imagens distintas:

- uma base de imagens médicas, contendo imagens de fundo do olho humano, utilizadas em diagnósticos de patologias. A base é composta de duas categorias: imagens de fundos de olhos normais, e imagens de fundos de olhos com anomalias. A categoria *normais* tem um total de 687 imagens, enquanto a categoria *anomalias*, 693. As imagens foram obtidas junto à Universidade Federal de São Paulo (UNIFESP). Médicos especialistas da própria UNIFESP anotaram as imagens, fazendo a separação entre as classes.

Exemplos de imagens para esta base são mostrados na Figura 5.1

- Free Foto Nature, contendo fotografias de cenas naturais. A base possui nove categorias, como folhas, árvores, praias, etc, nomeadas 001, 002, ..., 009. As categorias possuem um total de 496, 854, 483, 204, 70, 616, 83, 162 e 494 imagens respectivamente, e encontra-se disponível na Internet no endereço www.freefoto.com.

Exemplos de imagens para esta base são mostrados na Figura 5.2

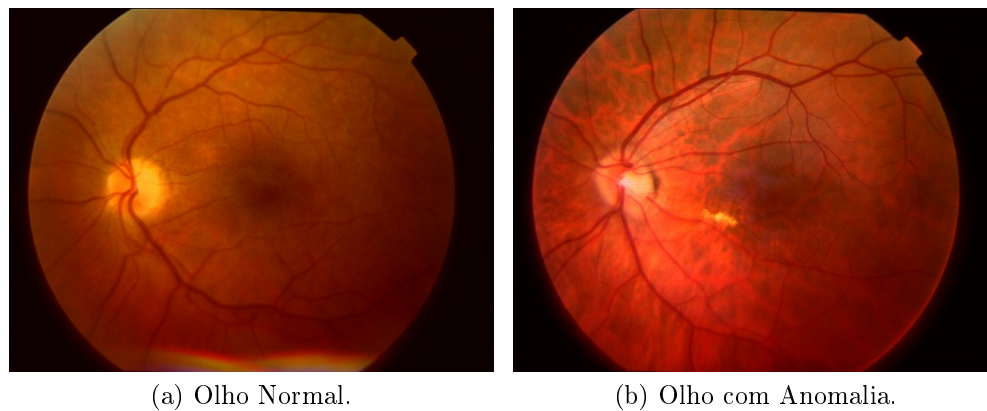


Figura 5.1: Imagens de exemplo para a base de imagens de retina.



Figura 5.2: Imagens de exemplo para a base de imagens Free Foto Nature.

5.2 Metodologia e Resultados

Quatro métodos foram utilizados para a criação do vocabulário: SIFT, BIC, SURF, e *patches*. Para o método *patches*, os descritores foram criados com tamanho 11×11 , totalizando 121 dimensões. Para o método SURF, foi utilizado a sua variante U-SURF, com 64 dimensões. O SIFT foi utilizado em sua forma padrão com 128 dimensões. O BIC é um descritor global, gerando uma palavra para cada imagem. Estes métodos são descritos no Capítulo 4.

Os descritores encontrados agrupados de acordo com sua similaridade através de algum algoritmo de clusterização. No presente estudo, utilizou-se o método *k-means*, um algoritmo do tipo *Expectation-Maximization*, que busca agrupar estes descritores em k grupos distintos, representados pelo centro de cada grupo. Cada um dos k centros é chamado uma *palavra visual*. Ao conjunto de todas as palavras visuais dá-se o nome de *dicionário visual*.

É importante ressaltar que a utilização do LDA com palavras visuais da forma apresentada neste trabalho dispensa a etapa de suavização descrita na Seção 3.4. Como cada descritor é atribuído a uma palavra pré-existente no dicionário visual, não há ocorrência de palavras não presentes na etapa de treinamento.

Na Figura 5.3, algumas das palavras visuais encontradas para a base Free Foto Nature a partir do agrupamento dos descritores encontrados pelo método *patches* são apresentadas. Analogamente, a Figura 5.4 apresenta palavras encontradas para a base de imagens de retina. Para esta base em particular, apresentam-se também, na Figura 5.5 as palavras encontradas para dicionários de tamanho $k = 120, 80$ e 40 .

Dada uma imagem de treinamento, o primeiro passo é descrevê-la através dos descritores utilizados. Cada imagem é representada por um vetor

$$I_d = (d_1, \dots, d_n),$$

onde d_i são os descritores.

Em seguida, cada um dos descritores encontrados é comparado com as palavras pertencentes ao dicionário visual da base de imagens. O descritor é associado à palavra visual w_j cuja distância

$$d_{ij} = \text{dist}(d_i, w_j)$$

entre d_i e w_j for mínima, para algum critério de distância. Neste trabalho, é utilizada a distância euclidiana, também chamada L_2 . Em seguida cria-se uma nova representação da imagem, correspondente à tabela de frequências para cada uma das palavras.

Para cada descritor constroi-se um classificador baseado no método LDA, que corresponde a uma função $h(I) : \{\mathbf{I}\} \rightarrow \{C_1, \dots, C_n\}$, que associa uma dada imagem entre todas

as imagens possíveis, a uma entre as n categorias existentes. É trivial perceber que um classificador binário g pode ser construído a partir de h utilizando-se a função

$$g(I, C_j) = \begin{cases} 1, & h_i(I) = C_j \\ 0, & \text{Caso Contrário.} \end{cases} \quad (5.1)$$

Além dos classificadores alimentados por apenas um descritor, este trabalho analisou o comportamento do classificador em um método de votação. Para este método, obtém-se a resposta do classificador para os quatro descritores utilizados ou, alternativamente, para algum subconjunto de três descritores entre os quatro disponíveis.

Considerando uma votação com n participantes onde cada um deles oferece uma resposta binária $g(I, C_j)$ de acordo com a Equação 5.1, define-se a função

$$f(I, C_j) = \begin{cases} 1, & \sum_{i=1}^n g_i(I, C_j) > \lfloor \frac{n}{2} \rfloor \\ 0, & \text{Caso Contrário.} \end{cases} \quad (5.2)$$

Dessa forma, o resultado do classificador para o método de votação é também uma resposta binária sobre a pertinência ou não a uma dada classe. Esta abordagem difere de um classificador que deve ter uma classe única como resposta. É importante notar que este método é aplicável apenas a classificadores que devem decidir entre mais de duas classes, visto que uma imagem pode ser enquadrada em mais de uma classe (devido ao uso da função $\lfloor \rfloor$). Este tipo de resposta é conveniente para bases de imagens com alto grau de similaridade interna entre subconjuntos das classes, onde mesmo humanos têm dificuldade em determinar precisamente a classe a qual uma imagem pertence.

Por exemplo, é possível que três métodos distintos a , b e c tenham como resposta as classes C_i , C_j e C_k , com $i \neq j \neq k$. Um classificador não binário que deseje detectar a qual classe pertence a imagem em questão deve decidir entre as três classes utilizando-se de algum critério.

Para decisões entre apenas duas classes, a votação deve exigir maioria absoluta, de forma que uma imagem seja sempre atribuída a apenas uma classe. Pode-se alcançar isso através da substituição da função $\lfloor \rfloor$ por $\lceil \rceil$.

Neste trabalho, foram realizados três experimentos:

1. No primeiro experimento estudou-se a performance do classificador *LDA* na base de imagens de retina, com metodologia de validação cruzada. A base foi dividida em cinco partições disjuntas, contendo aproximadamente um quinto das imagens em cada uma delas. De cada uma das categorias para a partição de teste, 138 imagens são selecionadas para os testes e 50 imagens por categoria são escolhidas aleatoriamente dentre as outras quatro partições para treinamento, totalizando 100 imagens de treinamento.

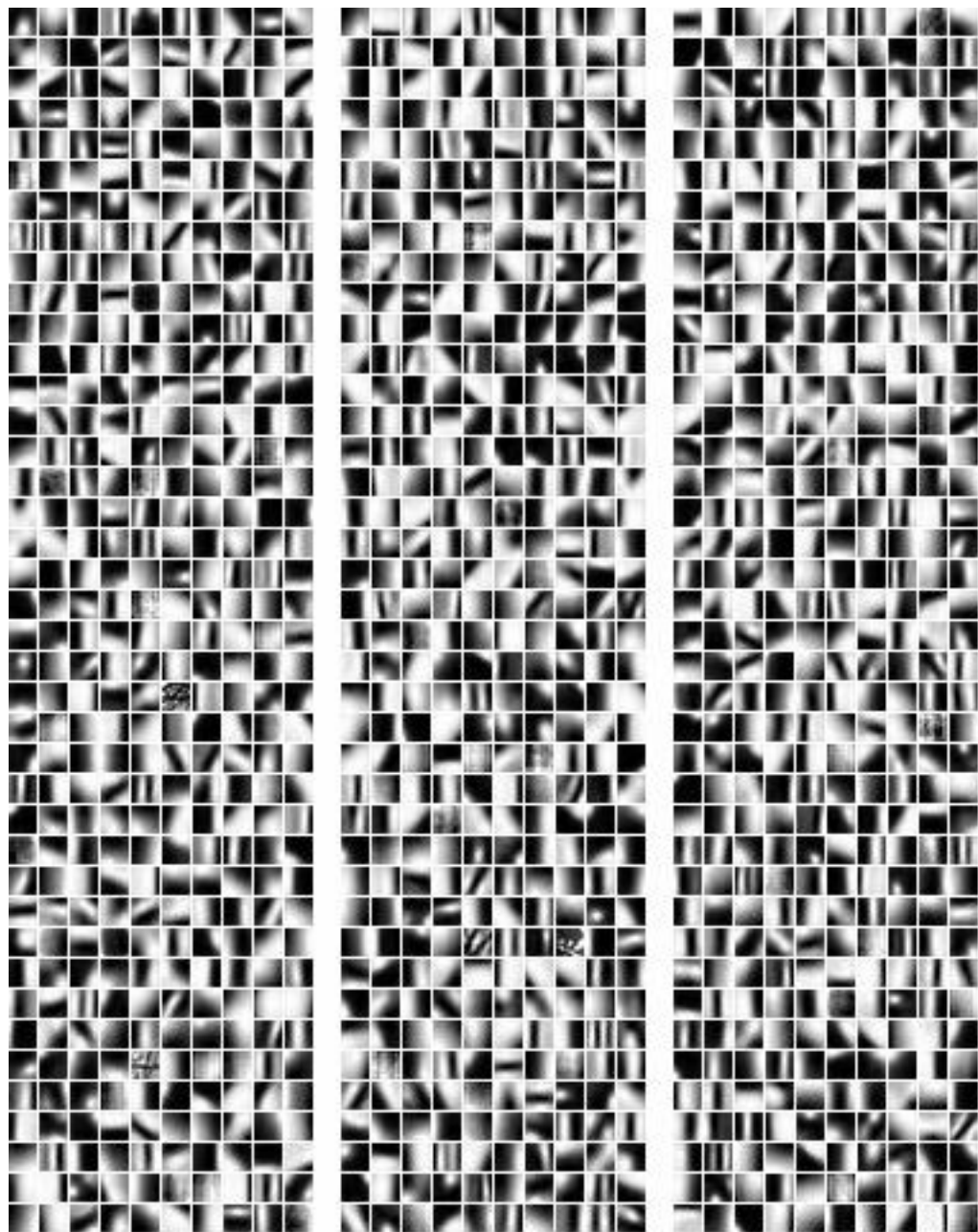


Figura 5.3: Exemplo de fragmentos da imagem, extraídos da base de dados de imagens Free Foto Nature. A Figura mostra 1200 fragmentos extraídos de todas as imagens do conjunto de treinamento para uma execução específica.

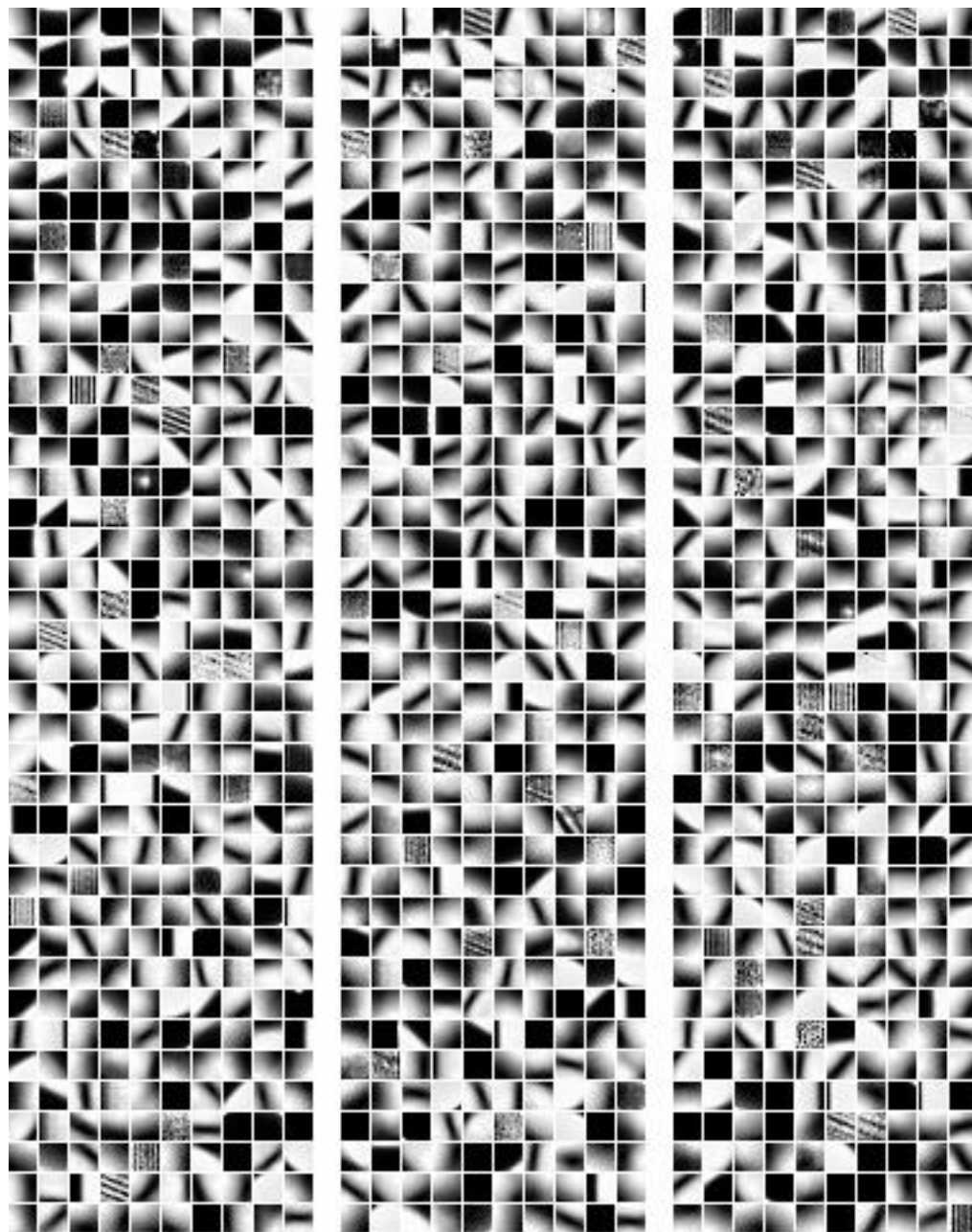
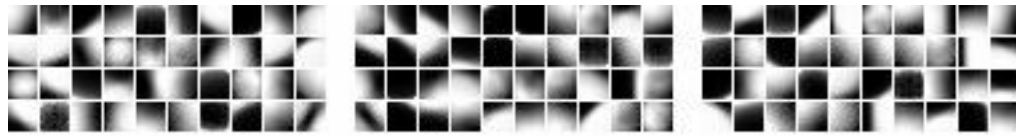
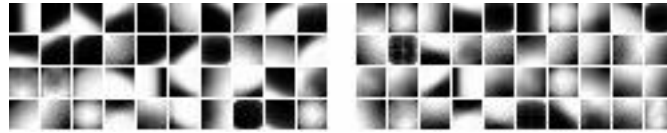


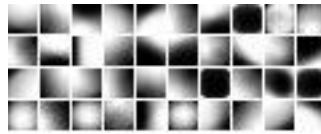
Figura 5.4: Exemplo de fragmentos da imagem, extraídos da base de dados de imagens de retina. A Figura mostra 1200 fragmentos extraídos de todas as imagens do conjunto de treinamento para uma execução específica.



(a) 120 palavras visuais



(b) 80 palavras visuais



(c) 40 palavras visuais.

Figura 5.5: Exemplo de fragmentos da imagem, extraídos da base de dados de imagens de retina. A Figura mostra quantidades variadas de fragmentos extraídos de todas as imagens do conjunto de treinamento para uma execução específica.

Como parâmetros, foram utilizados 120 tópicos latentes para o modelo *LDA* e um dicionário visual contendo 1200 palavras. Este experimento é repetido 30 vezes para cada partição, a fim de obter significância estatística. A performance da combinação dos classificadores pelo método de votação é analisada. Por se tratar de um classificador que deve separar as imagens em apenas duas categorias distintas, avaliam-se os resultados da votação com exigência de maioria absoluta.

2. No segundo experimento foi estudado o efeito do tamanho do dicionário visual, K , e quantidade dos tópicos latentes, T , na performance do classificador. A base utilizada foi a de imagens de retina, com 50 imagens de treinamento e 138 de testes por categoria. Para os quatro descritores utilizados, a performance do classificador foi medida variando-se o tamanho do dicionário visual para valores no conjunto $K = \{200, 400, 600, \dots, 2000\}$, e a quantidade de tópicos latentes para o modelo *LDA* para valores no conjunto $T = \{10, 20, \dots, 200\}$. Também foi observado o comportamento do classificador com $K = \{20, 40, \dots, 100\}$ palavras visuais. Porém, neste caso, a quantidade de tópicos latentes foi no máximo tão grande quanto o número de palavras visuais. Para cada ponto, 15 iterações foram realizadas.
3. No terceiro experimento foi estudada a influência da quantidade de imagens no conjunto de treinamento. A base Free Foto Nature foi dividida em cinco partições

distintas. A fim de manter as comparações entre as categorias equalizadas, utilizou-se como limite o tamanho da menor categoria. Dessa forma, foram utilizadas $n = \{5, 6, \dots, 54\}$ imagens de treinamento e 14 imagens de testes por categoria. A escolha do número 5 como limite inferior do intervalo é arbitrária.

Diferentemente da base de imagens de retina, o conjunto de testes foi mantido fixo, sendo uma partição permanentemente separada para as imagens de treinamento, sem o uso de validação cruzada. Foram realizados 15 experimentos para cada número de imagem.

5.2.1 Experimento 1

Para o primeiro experimento, com a base de imagens de retina, as médias para as taxas de acerto para cada uma das categorias com intervalo de dois desvios padrões são apresentadas nas Tabelas 5.3, para os descritores, e 5.4 para a votação entre eles. Por tratar-se de uma decisão entre duas classes, utiliza-se um critério de votação por maioria absoluta.

As Tabelas 5.1 e 5.2 mostram as médias detalhadas para cada uma das categorias separadamente, também com intervalo de dois desvios padrões. Estes mesmos dados são apresentados nas Figuras 5.6 e 5.7 para a média de descritores e votação, e nas Figuras 5.8 e 5.9 para cada uma das categorias separadamente.

Sob a ótica da validação cruzada, as médias obtidas para cada uma das partições se mantêm dentro do intervalo de erro obtido para cada partição separadamente. No entanto, as médias encontram-se apenas ligeiramente superiores à chance. Para as imagens desta base, SIFT e SURF tem resultados ligeiramente melhores, porém, ainda dentro do intervalo de erro. A votação entre os métodos não melhora significativamente o desempenho do classificador. A análise das Tabelas 5.4 e 5.3 revela que o melhor desempenho médio do classificador é encontrado para o caso *Todos - Patches*. Neste caso, a grande similaridade entre os descritores SIFT e SURF fazem com que as respostas emitidas por ambos os classificadores tendam a ter um alto índice de coincidência. Vale lembrar, que a quantidade de imagens utilizada, 50 imagens, é uma quantidade considerada pequena para modelos discriminativos normalmente utilizados nas tarefas de classificação de imagens deste tipo.

Analisando as Tabelas 5.1 e 5.2, bem como os gráficos correspondentes apresentados nas Figuras 5.8 e 5.9, percebe-se que à exceção do descritor BIC, que apresenta taxas similares para ambas as categorias, os outros descritores privilegiam a categoria anomalias.

Este resultado é coerente com o que intuitivamente se espera para essa base. Espera-se que descritores de cor, como o BIC, sejam menos capazes de se decidir entre as classes, dada a alta similaridade de cores entre as categorias (ver Figura 5.1).

Para os outros descritores, também é razoável esperar similaridade entre as imagens,

pois todas são imagens do olho humano. Espera-se encontrar na categoria anomalias, características não presentes nas imagens normais, por serem representativas das próprias anomalias. Além disso, não se espera que o classificador seja capaz de encarar a falta de evidência como informação relevante do modelo, de forma que a ausência dos descritores relacionados com anomalias não indica uma maior probabilidade de tratar-se de uma imagem normal.

No entanto, deve-se lembrar que as imagens apresentam informações imprecisas. Devido a condições de iluminação, foco, perda de dados na amostragem, além da alguma imprecisão imposta pelos próprios descritores, não se espera que as imagens do conjunto normais sejam um subconjunto de anomalias. Em situações reais, deverão haver palavras presentes exclusivamente no conjunto das imagens normais, mas em menor quantidade. De fato, uma análise cuidadosa revela que esse é o caso.

Sendo $\mathbf{A} = \{w_a^1, \dots, w_a^n\}$ o conjunto de palavras associadas às imagens anômalas do conjunto de treinamento e $\mathbf{N} = \{w_n^1, \dots, w_n^n\}$ o análogo para as imagens normais, pode-se definir, fazendo uso dos operadores de subtração de conjuntos, $\setminus$, e intersecção de conjuntos, $\cap$, as relações

$$p_a = |\mathbf{A} \setminus \mathbf{A} \cap \mathbf{N}|, \quad (5.3)$$

$$p_n = |\mathbf{N} \setminus \mathbf{N} \cap \mathbf{A}|. \quad (5.4)$$

Cada um desses termos representa a quantidade de palavras presentes em uma categoria, independente de sua frequência, mas totalmente ausentes na outra. A métrica

$$r_p = \frac{p_a}{p_n} - 1 \quad (5.5)$$

é a razão entre a quantidade de palavras únicas entre as categorias normais e anomalias. O termo -1 faz com que os pontos estejam centrados no zero, com conjuntos onde há mais palavras únicas no conjunto das imagens normais obtendo um valor negativo.

Da mesma forma define-se

$$r_h = \frac{h_a}{h_n} - 1, \quad (5.6)$$

onde h_a é a taxa de acerto do classificador para as imagens anômalas, e h_n , para as imagens normais. Para melhor compreender a relação entre estas duas métricas, acompanhou-se a performance dos descritores e as palavras geradas pelo conjunto de treinamento ao longo de 15 execuções.

A Figura 5.10 mostra o resultado da comparação entre r_h e r_p para o descritores SURF. Em todas elas, ambos os termos são sempre positivos, indicando que uma taxa de acerto maior da categoria anomalias vem acompanhada de um maior número de palavras únicas. Além do fato da unicidade em si, r_p cresce a medida que r_h cresce, apesar de não se observar uma relação de proporcionalidade direta.

Método	BIC	SIFT	SURF	Patches
Normais	$0,5 \pm 0,2$	$0,3 \pm 0,2$	$0,4 \pm 0,2$	$0,4 \pm 0,2$
Anomalias	$0,5 \pm 0,2$	$0,8 \pm 0,2$	$0,7 \pm 0,2$	$0,7 \pm 0,2$

Tabela 5.1: Médias e desvios padrões encontrados para as categorias Normais e Anomalias.

Método	Todos	Todos - BIC	Todos - SIFT	Todos - SURF	Todos - Patches
Normais	$0,2 \pm 0,2$	$0,3 \pm 0,2$	$0,4 \pm 0,2$	$0,4 \pm 0,2$	$0,4 \pm 0,2$
Anomalias	$0,7 \pm 0,2$	$0,8 \pm 0,2$	$0,7 \pm 0,2$	$0,7 \pm 0,2$	$0,8 \pm 0,2$

Tabela 5.2: Médias e desvios padrões encontrados para o método de votação com critério de maioria absoluta, para as categorias Normais e Anomalias.

Para os descritores SIFT e patches, mostrado nas Figuras 5.11 e 5.12, algumas execuções têm a categoria normais com taxa de acerto superior à anomalias. Porém nestas, r_p é também negativo. A única exceção acontece na execução 4 para o descritor patches. Contudo, ambas as métricas são próximas a zero, sem nenhuma categoria tendo um número significativo de palavras únicas.

Uma situação totalmente diferente é verificada para o descritor BIC. Para todas as execuções, r_p permanece próximo a zero e r_h se alterna. Este resultado corrobora a afirmação de que a baixa variabilidade entre as classes em relação a este descritor afeta a capacidade de distinção do classificador.

Estes resultados são compatíveis com o cenário encontrado nas Figuras 5.8 e 5.9 e previamente descrito. Disso, pode-se concluir que em situações de alta similaridade inter-classes onde uma das classes aproxima-se de um subconjunto da outra, a maior classe será privilegiada pelo classificador.

Método	BIC	SIFT	SURF	Patches
Partição 1	$0,55 \pm 0,06$	$0,59 \pm 0,06$	$0,59 \pm 0,06$	$0,53 \pm 0,06$
Partição 2	$0,56 \pm 0,06$	$0,60 \pm 0,06$	$0,59 \pm 0,08$	$0,53 \pm 0,04$
Partição 3	$0,52 \pm 0,06$	$0,56 \pm 0,08$	$0,57 \pm 0,06$	$0,51 \pm 0,06$
Partição 4	$0,53 \pm 0,06$	$0,60 \pm 0,06$	$0,61 \pm 0,08$	$0,53 \pm 0,04$
Partição 5	$0,52 \pm 0,06$	$0,57 \pm 0,06$	$0,57 \pm 0,06$	$0,53 \pm 0,04$
Média	$0,54 \pm 0,06$	$0,58 \pm 0,08$	$0,59 \pm 0,08$	$0,53 \pm 0,06$

Tabela 5.3: Médias e desvios padrões encontrados para as cinco partições.

Método	Todos	Todos - BIC	Todos - SIFT	Todos - SURF	Todos - Patches
Partição 1	$0,44 \pm 0,06$	$0,58 \pm 0,06$	$0,58 \pm 0,07$	$0,58 \pm 0,06$	$0,61 \pm 0,07$
Partição 2	$0,45 \pm 0,05$	$0,59 \pm 0,07$	$0,59 \pm 0,06$	$0,59 \pm 0,06$	$0,62 \pm 0,07$
Partição 3	$0,41 \pm 0,05$	$0,55 \pm 0,05$	$0,55 \pm 0,05$	$0,54 \pm 0,05$	$0,57 \pm 0,05$
Partição 4	$0,5 \pm 0,1$	$0,55 \pm 0,2$	$0,57 \pm 0,07$	$0,53 \pm 0,23$	$0,5 \pm 0,2$
Partição 5	$0,42 \pm 0,09$	$0,56 \pm 0,1$	$0,56 \pm 0,05$	$0,55 \pm 0,11$	$0,5 \pm 0,1$
Média	$0,44 \pm 0,09$	$0,6 \pm 0,1$	$0,57 \pm 0,07$	$0,56 \pm 0,1$	$0,58 \pm 0,1$

Tabela 5.4: Médias e desvios padrões encontrados para as cinco partições, utilizando-se o método de votação com critério de maioria absoluta.

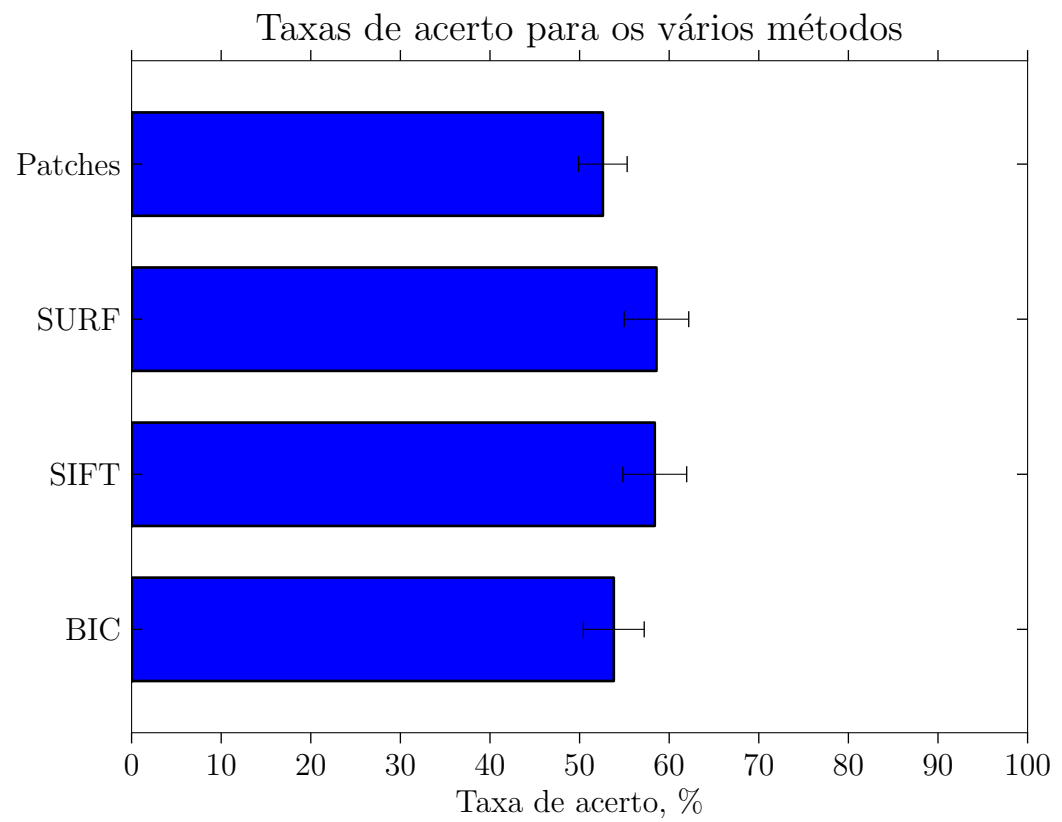


Figura 5.6: Taxas de acerto para os diferentes métodos para a base de imagens de retina. Experimentos realizados com 50 imagens.

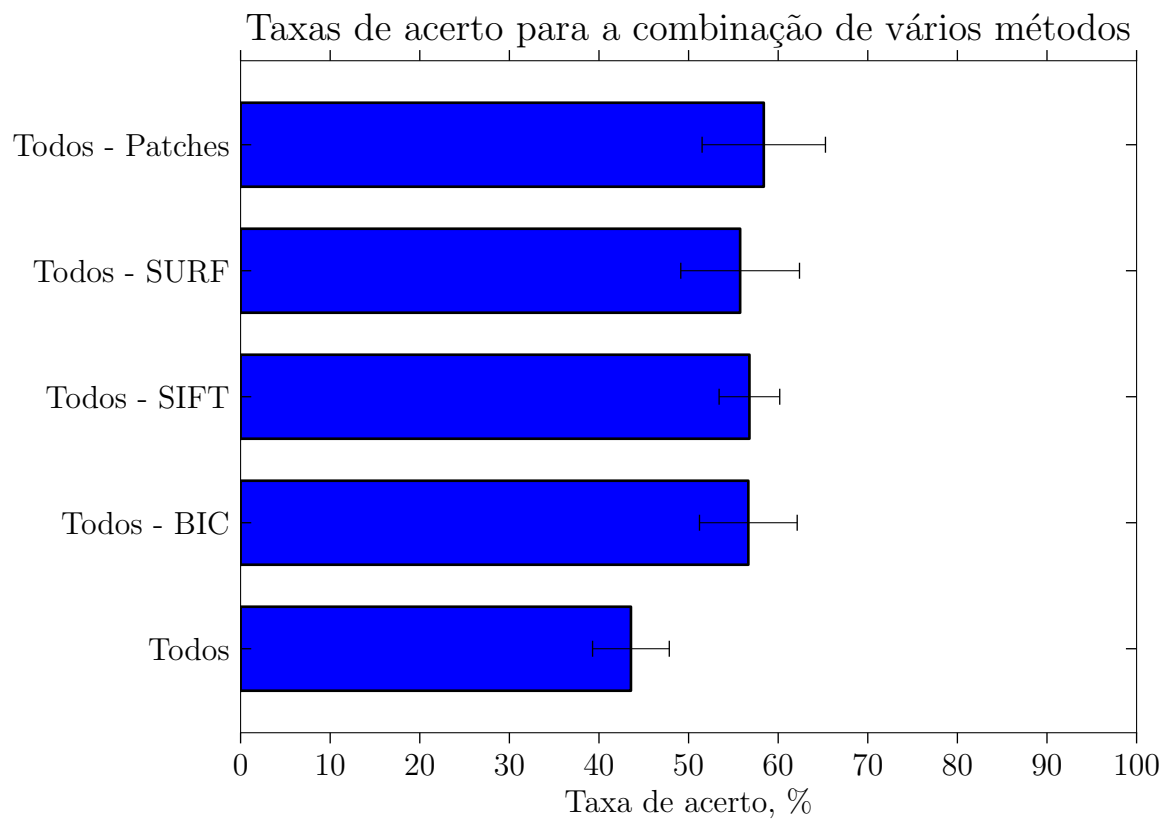


Figura 5.7: Taxas de acerto com método de votação para a base de imagens de retina. Experimentos realizados com 50 imagens.

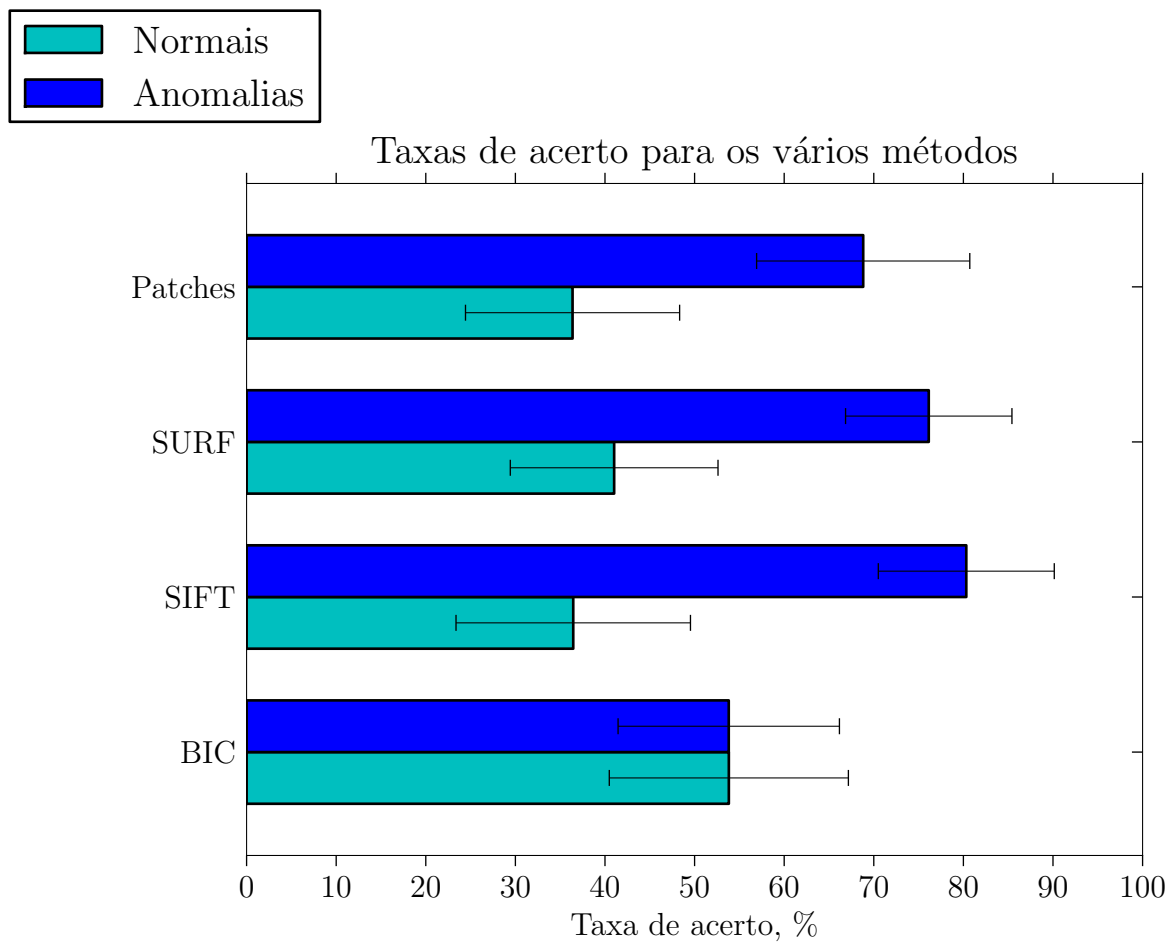


Figura 5.8: Taxas de acerto para os diferentes métodos para a base de imagens de retina. Experimentos realizados com 50 imagens. As categorias *normais* e *anomalias* são mostradas separadamente.

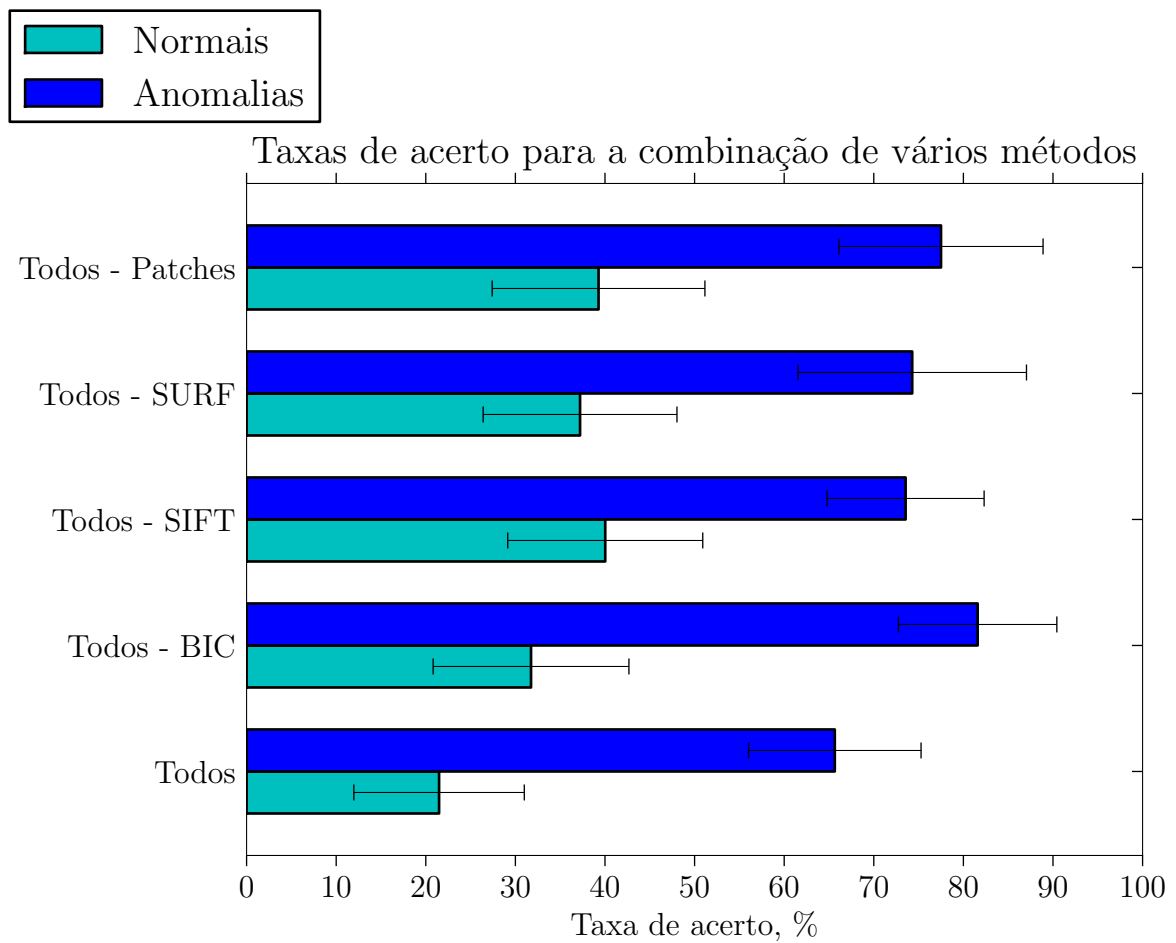


Figura 5.9: Taxas de acerto com método de votação para a base de imagens de retina. Experimentos realizados com 50 imagens. As categorias *normais* e *anomalias* são mostradas separadamente

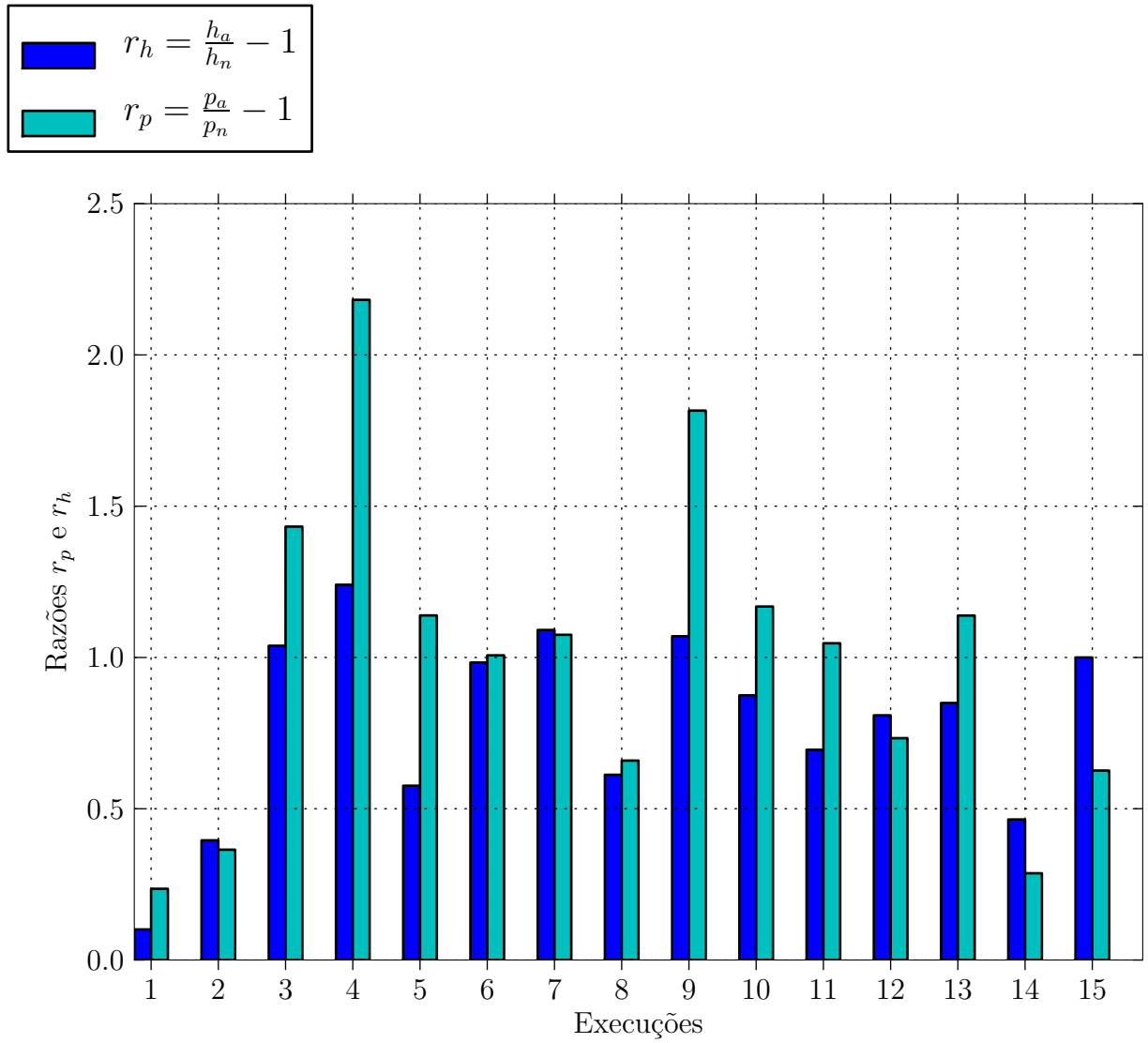


Figura 5.10: Proporção entre taxas de acerto por categorias e unicidade de palavras no conjunto de treinamento, para o descritor SURF.

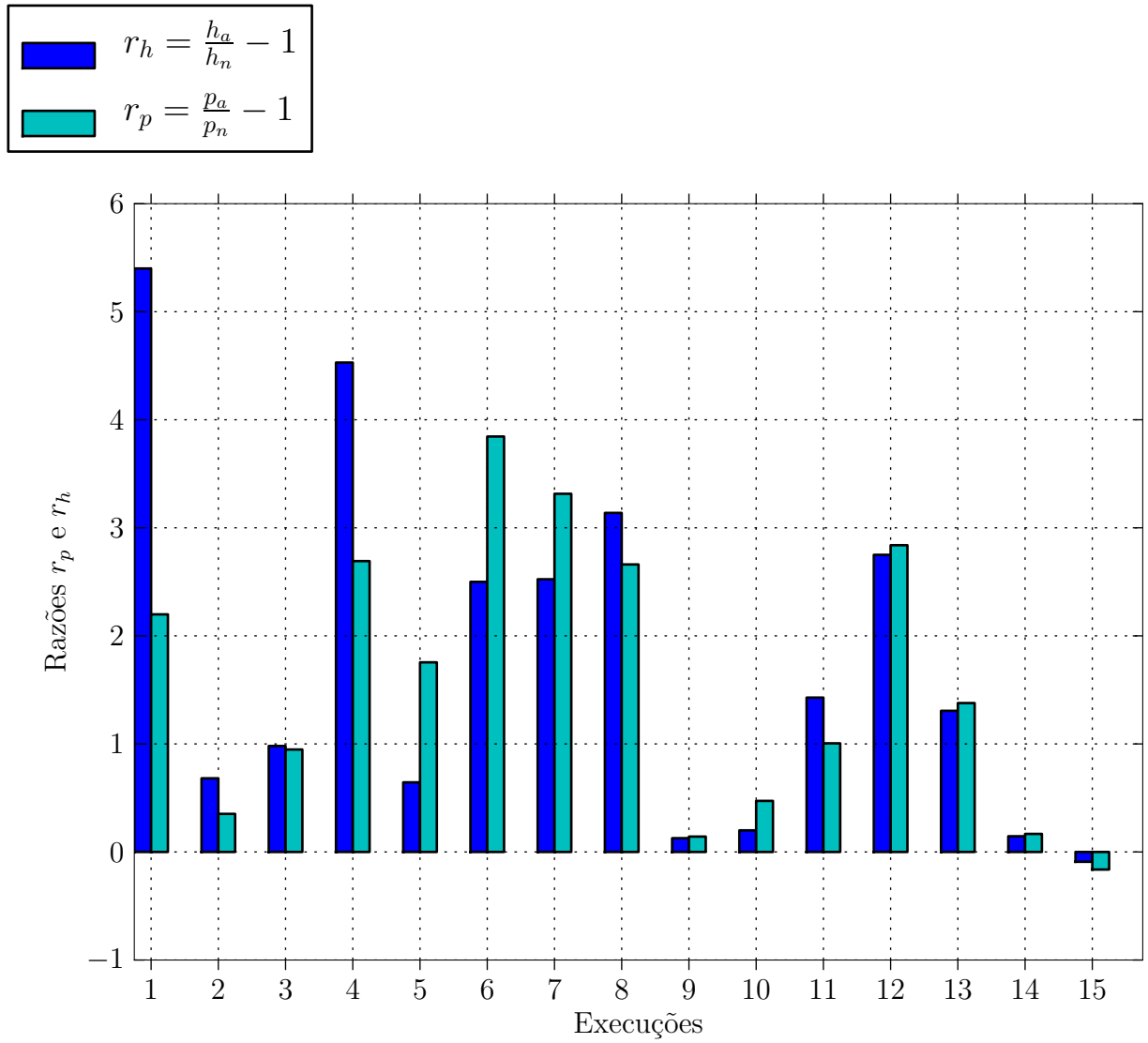


Figura 5.11: Proporção entre taxas de acerto por categorias e unicidade de palavras no conjunto de treinamento, para o descritor SIFT.

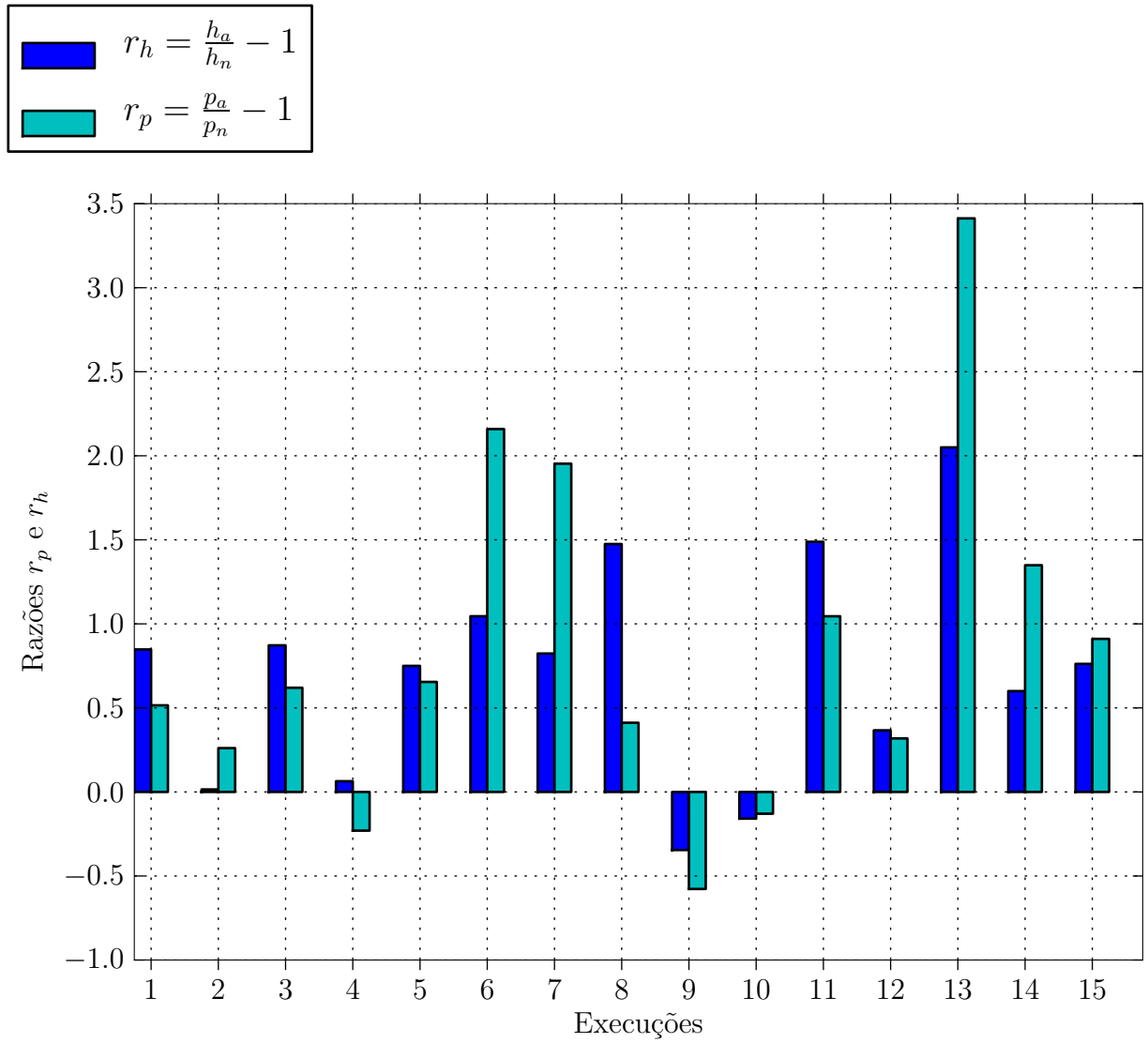


Figura 5.12: Proporção entre taxas de acerto por categorias e unicidade de palavras no conjunto de treinamento, para o descritor patches.

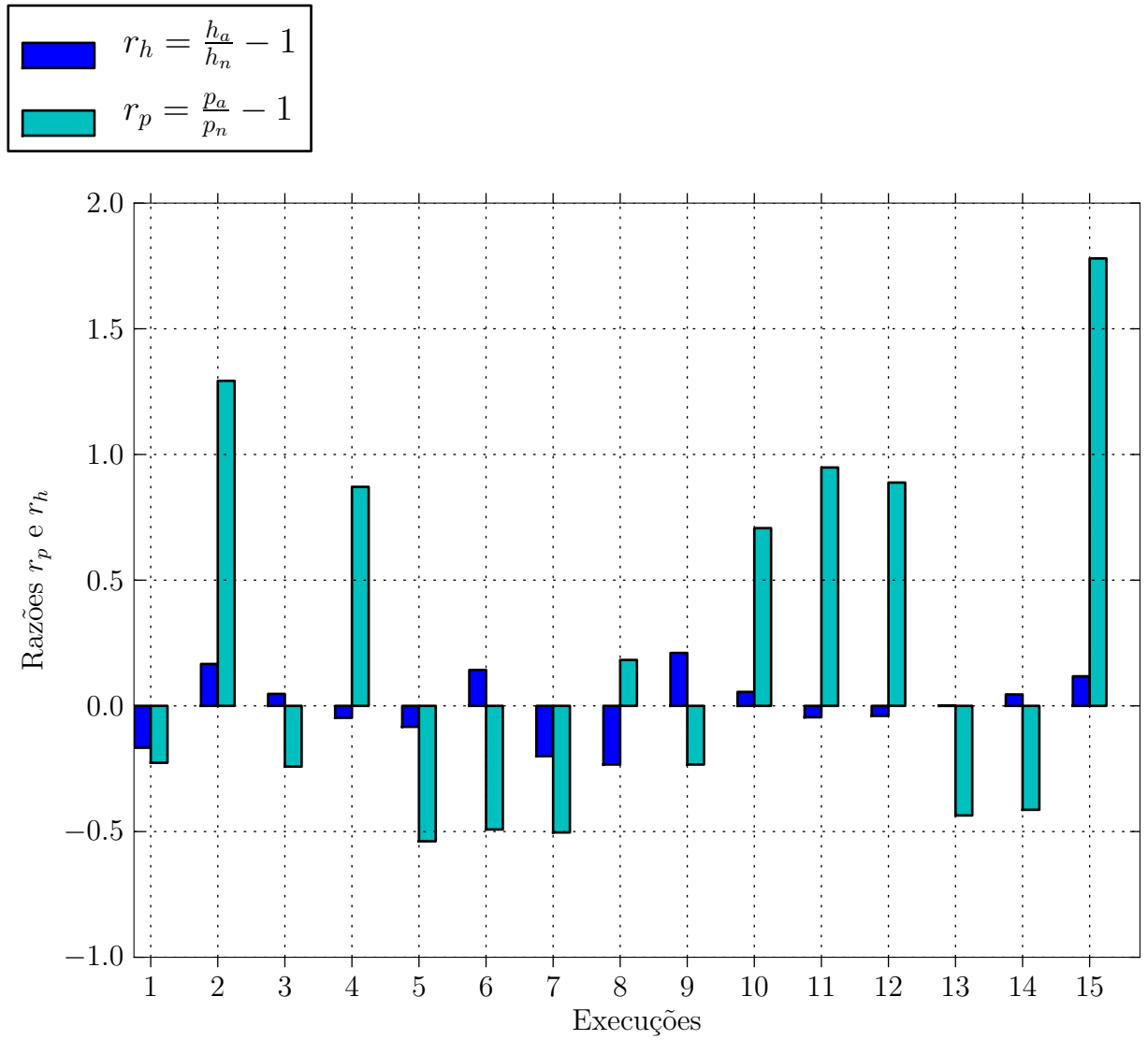


Figura 5.13: Proporção entre taxas de acerto por categorias e unicidade de palavras no conjunto de treinamento, para o descritor BIC.

5.2.2 Experimento 2

As Tabelas 5.6, 5.8 e 5.10 mostram o resultado do segundo experimento, onde estudou-se a influência do número de imagens no dicionário visual e quantidade de tópicos latentes do modelo para a base de imagens de retina.

No intervalo $K = \{400, 600, \dots, 2000\}$, não foi constatada nenhuma diferença fora do intervalo de erro para nenhum dos três descritores testados. O mesmo pode ser verificado para dicionários menores, no intervalo $K = \{20, \dots, 200\}$, como pode ser verificado nas Tabelas 5.5, 5.7 e 5.9. A comparação dos extremos do intervalo, no entanto, para os descritores SURF e patches, mostram que alguma diferença começa a poder ser notada.

O resultado indica que modelos podem ser treinados com números reduzido de palavras visuais, bem como de tópicos latentes. Para o algoritmo de clusterização utilizado, tanto a etapa-E quanto a etapa-M devem comparar todos os pontos candidatos a cada um dos centróides¹. É trivial perceber que a utilização de menos centros resulta em menor tempo de execução.

O resultado não surpreende, após uma análise cuidadosa da Figura 5.4. Pode-se perceber que ao menos para o descritor *patches*, para o qual é possível fazer essa análise visual, muitas das palavras visuais encontradas são bastante similares. Palavras similares ainda são encontradas mesmo para dicionários menores, como se pode perceber pela Figura 5.5.

Muitos dos padrões distintos na Figura 5.4, no entanto, não se repetem na Figura 5.5, principalmente os padrões de texturas complexos. Isso sugere que texturas complexas têm poder distintivo para as imagens dessa base. Padrões simples, como os da Figura 5.5 demonstram-se suficientes para definir a performance do classificador.

Quanto ao número de tópicos, a Figura 5.14 apresenta as médias em 20 execuções com um intervalo de dois desvios padrões para os tempos de execução de um aprendizado de parâmetros para 50 imagens aleatórias da base de retina, categoria Normais. O dicionário utilizado continha 1200 palavras visuais.

Como se pode perceber, o tempo de execução também sobe a medida que se aumentam o número de tópicos. Para valores pequenos, o resultado do experimento é errático, com tempos de execução por vezes superiores aos obtidos para mais tópicos, e altos desvios padrões. Tal comportamento advém do fato do algoritmo utilizado para a determinação dos parâmetros ser um algoritmo iterativo. Para poucos tópicos, o algoritmo necessita de um número maior de iterações para atingir a convergência. Porém, para valores próximos a 90, até o ponto final da aferição, 300, pode-se realizar a medida com precisão.

¹Rigorosamente falando, a comparação com todos os centróides não é necessária. O uso de estruturas de dados como *kd-trees* [30] resulta em menor tempo de execução do que uma comparação exaustiva.

Tópicos	20	40	60	80	100	200
10	$0,57 \pm 0,07$	$0,55 \pm 0,07$	$0,57 \pm 0,07$	$0,57 \pm 0,05$	$0,55 \pm 0,05$	$0,55 \pm 0,07$
20	$0,54 \pm 0,05$	$0,55 \pm 0,06$	$0,56 \pm 0,06$	$0,56 \pm 0,06$	$0,55 \pm 0,05$	$0,56 \pm 0,05$
30		$0,55 \pm 0,06$	$0,55 \pm 0,05$	$0,55 \pm 0,07$	$0,56 \pm 0,03$	$0,56 \pm 0,06$
40		$0,55 \pm 0,07$	$0,57 \pm 0,05$	$0,55 \pm 0,05$	$0,56 \pm 0,06$	$0,55 \pm 0,06$
50			$0,57 \pm 0,05$	$0,56 \pm 0,06$	$0,56 \pm 0,07$	$0,56 \pm 0,05$
60			$0,56 \pm 0,06$	$0,57 \pm 0,07$	$0,55 \pm 0,06$	$0,57 \pm 0,05$
70				$0,55 \pm 0,03$	$0,55 \pm 0,05$	$0,55 \pm 0,05$
80				$0,56 \pm 0,07$	$0,55 \pm 0,05$	$0,56 \pm 0,06$
90					$0,56 \pm 0,05$	$0,56 \pm 0,06$
100					$0,57 \pm 0,05$	$0,55 \pm 0,07$
110						$0,57 \pm 0,06$
120						$0,57 \pm 0,05$
130						$0,56 \pm 0,05$
140						$0,55 \pm 0,04$
150						$0,56 \pm 0,07$
160						$0,55 \pm 0,08$
170						$0,56 \pm 0,04$
180						$0,56 \pm 0,06$
190						$0,56 \pm 0,05$
200						$0,57 \pm 0,06$

Tabela 5.5: Taxas de acerto médias e desvios padrões encontrados para o descritor BIC, variando-se o tamanho do dicionário visual e o número de tópicos.

Tópicos	400	600	800	1000	1200	1400	1600	1800	2000
10	0,56 ± 0,06	0,57 ± 0,06	0,55 ± 0,06	0,55 ± 0,06	0,56 ± 0,06	0,56 ± 0,07	0,55 ± 0,06	0,56 ± 0,05	0,56 ± 0,06
20	0,56 ± 0,06	0,55 ± 0,05	0,56 ± 0,05	0,55 ± 0,06	0,56 ± 0,05	0,56 ± 0,06	0,56 ± 0,06	0,55 ± 0,06	0,55 ± 0,05
30	0,55 ± 0,05	0,56 ± 0,07	0,56 ± 0,06	0,56 ± 0,05	0,55 ± 0,07	0,55 ± 0,06	0,56 ± 0,06	0,55 ± 0,05	0,56 ± 0,05
40	0,55 ± 0,07	0,56 ± 0,06	0,56 ± 0,06	0,55 ± 0,06	0,56 ± 0,06	0,55 ± 0,06	0,56 ± 0,07	0,56 ± 0,06	0,55 ± 0,06
50	0,55 ± 0,06	0,56 ± 0,06	0,55 ± 0,07	0,56 ± 0,06	0,55 ± 0,06	0,56 ± 0,05	0,56 ± 0,06	0,56 ± 0,06	0,55 ± 0,05
60	0,55 ± 0,06	0,56 ± 0,06	0,56 ± 0,07	0,56 ± 0,06	0,56 ± 0,06	0,56 ± 0,06	0,56 ± 0,05	0,56 ± 0,06	0,56 ± 0,07
70	0,55 ± 0,06	0,56 ± 0,05	0,56 ± 0,06	0,55 ± 0,06	0,56 ± 0,06	0,55 ± 0,06	0,55 ± 0,06	0,56 ± 0,06	0,56 ± 0,06
80	0,55 ± 0,06	0,56 ± 0,05	0,56 ± 0,07	0,56 ± 0,06	0,56 ± 0,06	0,55 ± 0,06	0,55 ± 0,06	0,56 ± 0,06	0,55 ± 0,05
90	0,56 ± 0,07	0,56 ± 0,07	0,56 ± 0,06	0,56 ± 0,05	0,56 ± 0,06	0,55 ± 0,07	0,56 ± 0,06	0,56 ± 0,06	0,56 ± 0,05
100	0,57 ± 0,06	0,56 ± 0,07	0,55 ± 0,07	0,55 ± 0,08	0,56 ± 0,05	0,56 ± 0,05	0,56 ± 0,05	0,56 ± 0,06	0,56 ± 0,05
110	0,56 ± 0,06	0,56 ± 0,06	0,56 ± 0,04	0,57 ± 0,06	0,56 ± 0,06	0,56 ± 0,06	0,55 ± 0,06	0,56 ± 0,05	0,55 ± 0,05
120	0,56 ± 0,05	0,56 ± 0,04	0,56 ± 0,07	0,54 ± 0,07	0,56 ± 0,05	0,56 ± 0,05	0,56 ± 0,05	0,55 ± 0,07	0,56 ± 0,05
130	0,56 ± 0,06	0,57 ± 0,06	0,56 ± 0,05	0,55 ± 0,07	0,55 ± 0,06	0,56 ± 0,05	0,56 ± 0,05	0,55 ± 0,05	0,56 ± 0,05
140	0,56 ± 0,07	0,55 ± 0,05	0,55 ± 0,07	0,57 ± 0,06	0,55 ± 0,05	0,56 ± 0,05	0,56 ± 0,06	0,56 ± 0,06	0,57 ± 0,06
150	0,56 ± 0,05	0,55 ± 0,08	0,57 ± 0,06	0,56 ± 0,05	0,55 ± 0,06	0,56 ± 0,06	0,56 ± 0,06	0,56 ± 0,06	0,56 ± 0,05
160	0,56 ± 0,05	0,56 ± 0,06	0,56 ± 0,06	0,55 ± 0,06	0,56 ± 0,05	0,56 ± 0,07	0,55 ± 0,05	0,56 ± 0,06	0,57 ± 0,05
170	0,56 ± 0,06	0,56 ± 0,05	0,56 ± 0,07	0,55 ± 0,06	0,55 ± 0,05	0,56 ± 0,07	0,57 ± 0,06	0,55 ± 0,06	0,56 ± 0,05
180	0,56 ± 0,06	0,55 ± 0,05	0,55 ± 0,06	0,55 ± 0,05	0,56 ± 0,05	0,55 ± 0,06	0,56 ± 0,06	0,57 ± 0,07	0,57 ± 0,05
190	0,56 ± 0,06	0,56 ± 0,06	0,55 ± 0,07	0,56 ± 0,06	0,56 ± 0,06	0,55 ± 0,05	0,55 ± 0,06	0,56 ± 0,05	0,56 ± 0,06
200	0,55 ± 0,05	0,55 ± 0,05	0,56 ± 0,07	0,56 ± 0,06	0,56 ± 0,05	0,56 ± 0,06	0,56 ± 0,06	0,56 ± 0,06	0,56 ± 0,06

Tabela 5.6: Taxas de acerto médias e desvios padrões encontrados para o descritor BIC, variando-se o tamanho do dicionário visual e o número de tópicos.

Tópicos	20	40	60	80	100	200
10	$0,52 \pm 0,05$	$0,55 \pm 0,06$	$0,56 \pm 0,08$	$0,58 \pm 0,06$	$0,56 \pm 0,07$	$0,58 \pm 0,05$
20	$0,53 \pm 0,07$	$0,56 \pm 0,08$	$0,55 \pm 0,06$	$0,56 \pm 0,06$	$0,57 \pm 0,07$	$0,58 \pm 0,07$
30		$0,55 \pm 0,10$	$0,55 \pm 0,08$	$0,55 \pm 0,06$	$0,56 \pm 0,08$	$0,58 \pm 0,07$
40		$0,53 \pm 0,05$	$0,54 \pm 0,06$	$0,55 \pm 0,06$	$0,56 \pm 0,07$	$0,58 \pm 0,05$
50			$0,54 \pm 0,08$	$0,54 \pm 0,06$	$0,55 \pm 0,06$	$0,59 \pm 0,06$
60			$0,55 \pm 0,09$	$0,54 \pm 0,07$	$0,56 \pm 0,07$	$0,58 \pm 0,08$
70				$0,57 \pm 0,10$	$0,55 \pm 0,09$	$0,58 \pm 0,08$
80				$0,56 \pm 0,09$	$0,57 \pm 0,08$	$0,57 \pm 0,06$
90					$0,56 \pm 0,09$	$0,58 \pm 0,06$
100					$0,57 \pm 0,09$	$0,59 \pm 0,07$
110						$0,58 \pm 0,08$
120						$0,59 \pm 0,06$
130						$0,58 \pm 0,10$
140						$0,56 \pm 0,09$
150						$0,56 \pm 0,09$
160						$0,58 \pm 0,06$
170						$0,59 \pm 0,10$
180						$0,58 \pm 0,09$
190						$0,58 \pm 0,04$
200						$0,60 \pm 0,07$

Tabela 5.7: Taxas de acerto médias e desvios padrões encontrados para o descritor SURF, variando-se o tamanho do dicionário visual e o número de tópicos.

Tópicos	400	600	800	1000	1200	1400	1600	1800	2000
10	0,57 ± 0,07	0,57 ± 0,06	0,58 ± 0,05	0,57 ± 0,07	0,59 ± 0,05	0,59 ± 0,08	0,60 ± 0,06	0,60 ± 0,07	0,61 ± 0,06
20	0,57 ± 0,05	0,58 ± 0,05	0,57 ± 0,07	0,59 ± 0,06	0,58 ± 0,05	0,59 ± 0,08	0,60 ± 0,07	0,60 ± 0,07	0,60 ± 0,06
30	0,56 ± 0,05	0,57 ± 0,05	0,60 ± 0,06	0,58 ± 0,04	0,59 ± 0,07	0,59 ± 0,06	0,61 ± 0,07	0,61 ± 0,06	0,62 ± 0,08
40	0,57 ± 0,06	0,57 ± 0,05	0,59 ± 0,07	0,59 ± 0,04	0,60 ± 0,06	0,60 ± 0,06	0,60 ± 0,05	0,62 ± 0,05	0,60 ± 0,07
50	0,56 ± 0,07	0,57 ± 0,06	0,59 ± 0,07	0,59 ± 0,04	0,60 ± 0,06	0,61 ± 0,06	0,59 ± 0,07	0,61 ± 0,07	0,60 ± 0,07
60	0,56 ± 0,06	0,57 ± 0,08	0,58 ± 0,08	0,57 ± 0,07	0,59 ± 0,06	0,60 ± 0,04	0,60 ± 0,06	0,61 ± 0,06	0,60 ± 0,06
70	0,57 ± 0,06	0,58 ± 0,05	0,58 ± 0,06	0,58 ± 0,07	0,61 ± 0,06	0,59 ± 0,07	0,60 ± 0,06	0,61 ± 0,06	0,61 ± 0,05
80	0,57 ± 0,05	0,57 ± 0,08	0,57 ± 0,06	0,58 ± 0,08	0,58 ± 0,04	0,59 ± 0,07	0,60 ± 0,07	0,60 ± 0,05	0,62 ± 0,06
90	0,59 ± 0,06	0,57 ± 0,05	0,57 ± 0,06	0,58 ± 0,07	0,59 ± 0,06	0,60 ± 0,06	0,60 ± 0,06	0,60 ± 0,08	0,62 ± 0,07
100	0,58 ± 0,07	0,57 ± 0,07	0,58 ± 0,04	0,58 ± 0,06	0,59 ± 0,09	0,61 ± 0,06	0,61 ± 0,04	0,60 ± 0,07	0,61 ± 0,06
110	0,57 ± 0,06	0,57 ± 0,05	0,58 ± 0,06	0,59 ± 0,06	0,59 ± 0,06	0,60 ± 0,07	0,61 ± 0,06	0,60 ± 0,07	0,61 ± 0,06
120	0,57 ± 0,06	0,58 ± 0,05	0,56 ± 0,06	0,59 ± 0,06	0,59 ± 0,06	0,59 ± 0,06	0,60 ± 0,06	0,60 ± 0,06	0,62 ± 0,08
130	0,59 ± 0,06	0,57 ± 0,07	0,56 ± 0,07	0,57 ± 0,03	0,58 ± 0,05	0,60 ± 0,08	0,59 ± 0,07	0,60 ± 0,06	0,60 ± 0,07
140	0,58 ± 0,04	0,58 ± 0,07	0,58 ± 0,06	0,58 ± 0,08	0,59 ± 0,08	0,60 ± 0,07	0,59 ± 0,06	0,60 ± 0,06	0,61 ± 0,06
150	0,57 ± 0,06	0,58 ± 0,05	0,60 ± 0,04	0,59 ± 0,05	0,58 ± 0,07	0,61 ± 0,06	0,60 ± 0,06	0,59 ± 0,08	0,62 ± 0,05
160	0,58 ± 0,07	0,58 ± 0,06	0,57 ± 0,08	0,59 ± 0,06	0,59 ± 0,05	0,59 ± 0,07	0,60 ± 0,05	0,60 ± 0,08	0,61 ± 0,06
170	0,59 ± 0,07	0,58 ± 0,06	0,58 ± 0,06	0,59 ± 0,05	0,59 ± 0,07	0,60 ± 0,07	0,61 ± 0,08	0,60 ± 0,06	0,62 ± 0,07
180	0,57 ± 0,07	0,58 ± 0,07	0,58 ± 0,05	0,58 ± 0,07	0,57 ± 0,04	0,60 ± 0,09	0,60 ± 0,05	0,61 ± 0,07	0,61 ± 0,07
190	0,58 ± 0,07	0,58 ± 0,06	0,58 ± 0,05	0,58 ± 0,07	0,59 ± 0,06	0,62 ± 0,08	0,60 ± 0,06	0,59 ± 0,06	0,61 ± 0,06
200	0,59 ± 0,07	0,58 ± 0,06	0,58 ± 0,08	0,58 ± 0,08	0,60 ± 0,05	0,60 ± 0,08	0,60 ± 0,06	0,61 ± 0,06	0,60 ± 0,06

Tabela 5.8: Taxas de acerto médias e desvios padrões encontrados para o descritor SURF, variando-se o tamanho do dicionário visual e o número de tópicos.

Tópicos	20	40	60	80	100	200
10	0,54 $\pm$ 0,07	0,55 $\pm$ 0,06	0,53 $\pm$ 0,06	0,53 $\pm$ 0,06	0,51 $\pm$ 0,06	0,53 $\pm$ 0,04
20	0,53 $\pm$ 0,06	0,54 $\pm$ 0,07	0,54 $\pm$ 0,05	0,54 $\pm$ 0,05	0,52 $\pm$ 0,06	0,54 $\pm$ 0,05
30		0,54 $\pm$ 0,05	0,53 $\pm$ 0,05	0,52 $\pm$ 0,06	0,54 $\pm$ 0,04	0,54 $\pm$ 0,06
40		0,52 $\pm$ 0,05	0,53 $\pm$ 0,06	0,52 $\pm$ 0,06	0,54 $\pm$ 0,06	0,53 $\pm$ 0,05
50			0,52 $\pm$ 0,06	0,53 $\pm$ 0,08	0,53 $\pm$ 0,05	0,53 $\pm$ 0,05
60			0,53 $\pm$ 0,05	0,53 $\pm$ 0,05	0,53 $\pm$ 0,06	0,53 $\pm$ 0,05
70				0,53 $\pm$ 0,06	0,53 $\pm$ 0,07	0,55 $\pm$ 0,04
80				0,52 $\pm$ 0,06	0,53 $\pm$ 0,05	0,55 $\pm$ 0,05
90					0,52 $\pm$ 0,05	0,54 $\pm$ 0,07
100					0,53 $\pm$ 0,07	0,54 $\pm$ 0,06
110						0,55 $\pm$ 0,03
120						0,56 $\pm$ 0,07
130						0,55 $\pm$ 0,07
140						0,56 $\pm$ 0,05
150						0,55 $\pm$ 0,07
160						0,55 $\pm$ 0,04
170						0,55 $\pm$ 0,06
180						0,56 $\pm$ 0,05
190						0,56 $\pm$ 0,07
200						0,56 $\pm$ 0,06

Tabela 5.9: Taxas de acerto médias e desvios padrões encontrados para o descritor Patches, variando-se o tamanho do dicionário visual e o número de tópicos.

Tópicos	400	600	800	1000	1200	1400	1600	1800	2000
10	0,53 ± 0,05	0,55 ± 0,05	0,54 ± 0,05	0,53 ± 0,07	0,54 ± 0,07	0,53 ± 0,07	0,53 ± 0,05	0,54 ± 0,05	0,54 ± 0,06
20	0,54 ± 0,06	0,53 ± 0,04	0,54 ± 0,06	0,54 ± 0,06	0,54 ± 0,07	0,53 ± 0,05	0,54 ± 0,06	0,54 ± 0,05	0,52 ± 0,06
30	0,53 ± 0,06	0,54 ± 0,05	0,53 ± 0,07	0,54 ± 0,06	0,53 ± 0,06	0,52 ± 0,04	0,52 ± 0,05	0,53 ± 0,06	0,54 ± 0,06
40	0,54 ± 0,05	0,53 ± 0,04	0,53 ± 0,05	0,53 ± 0,07	0,55 ± 0,06	0,53 ± 0,05	0,53 ± 0,05	0,52 ± 0,07	0,53 ± 0,05
50	0,54 ± 0,05	0,53 ± 0,06	0,53 ± 0,06	0,54 ± 0,07	0,51 ± 0,05	0,54 ± 0,04	0,54 ± 0,05	0,53 ± 0,05	0,53 ± 0,05
60	0,54 ± 0,05	0,54 ± 0,05	0,53 ± 0,08	0,53 ± 0,06	0,54 ± 0,06	0,54 ± 0,05	0,53 ± 0,08	0,52 ± 0,04	0,53 ± 0,05
70	0,54 ± 0,04	0,54 ± 0,07	0,54 ± 0,07	0,54 ± 0,06	0,53 ± 0,05	0,53 ± 0,05	0,53 ± 0,06	0,52 ± 0,04	0,53 ± 0,06
80	0,53 ± 0,07	0,53 ± 0,06	0,53 ± 0,04	0,52 ± 0,04	0,54 ± 0,04	0,52 ± 0,06	0,53 ± 0,07	0,54 ± 0,05	0,53 ± 0,05
90	0,54 ± 0,07	0,55 ± 0,06	0,54 ± 0,06	0,54 ± 0,03	0,53 ± 0,05	0,53 ± 0,05	0,53 ± 0,06	0,53 ± 0,04	0,53 ± 0,06
100	0,54 ± 0,05	0,54 ± 0,06	0,53 ± 0,05	0,54 ± 0,06	0,54 ± 0,04	0,54 ± 0,06	0,53 ± 0,05	0,53 ± 0,06	0,53 ± 0,06
110	0,53 ± 0,05	0,55 ± 0,06	0,53 ± 0,06	0,54 ± 0,05	0,52 ± 0,06	0,54 ± 0,05	0,53 ± 0,05	0,52 ± 0,05	0,54 ± 0,04
120	0,53 ± 0,05	0,53 ± 0,04	0,54 ± 0,07	0,55 ± 0,07	0,52 ± 0,05	0,54 ± 0,05	0,54 ± 0,05	0,53 ± 0,07	0,53 ± 0,05
130	0,54 ± 0,08	0,54 ± 0,05	0,55 ± 0,08	0,54 ± 0,04	0,53 ± 0,07	0,53 ± 0,06	0,54 ± 0,05	0,53 ± 0,05	0,53 ± 0,05
140	0,54 ± 0,07	0,54 ± 0,06	0,55 ± 0,05	0,55 ± 0,06	0,55 ± 0,05	0,53 ± 0,06	0,53 ± 0,06	0,53 ± 0,04	0,53 ± 0,04
150	0,55 ± 0,05	0,53 ± 0,08	0,55 ± 0,07	0,54 ± 0,06	0,54 ± 0,07	0,53 ± 0,06	0,53 ± 0,03	0,53 ± 0,06	0,52 ± 0,05
160	0,55 ± 0,06	0,53 ± 0,04	0,55 ± 0,05	0,54 ± 0,05	0,55 ± 0,04	0,54 ± 0,05	0,54 ± 0,06	0,53 ± 0,05	0,54 ± 0,06
170	0,55 ± 0,06	0,55 ± 0,05	0,54 ± 0,05	0,53 ± 0,05	0,53 ± 0,06	0,54 ± 0,07	0,54 ± 0,05	0,53 ± 0,07	0,55 ± 0,06
180	0,56 ± 0,06	0,56 ± 0,05	0,53 ± 0,06	0,55 ± 0,05	0,55 ± 0,05	0,54 ± 0,06	0,55 ± 0,06	0,53 ± 0,06	0,53 ± 0,05
190	0,56 ± 0,04	0,54 ± 0,06	0,55 ± 0,05	0,55 ± 0,03	0,53 ± 0,06	0,53 ± 0,05	0,53 ± 0,05	0,53 ± 0,07	0,52 ± 0,05
200	0,56 ± 0,06	0,52 ± 0,04	0,54 ± 0,07	0,53 ± 0,07	0,55 ± 0,07	0,54 ± 0,05	0,53 ± 0,08	0,53 ± 0,06	0,54 ± 0,05

Tabela 5.10: Taxas de acerto médias e desvios padrões encontrados para o descritor Patches, variando-se o tamanho do dicionário visual e o número de tópicos.

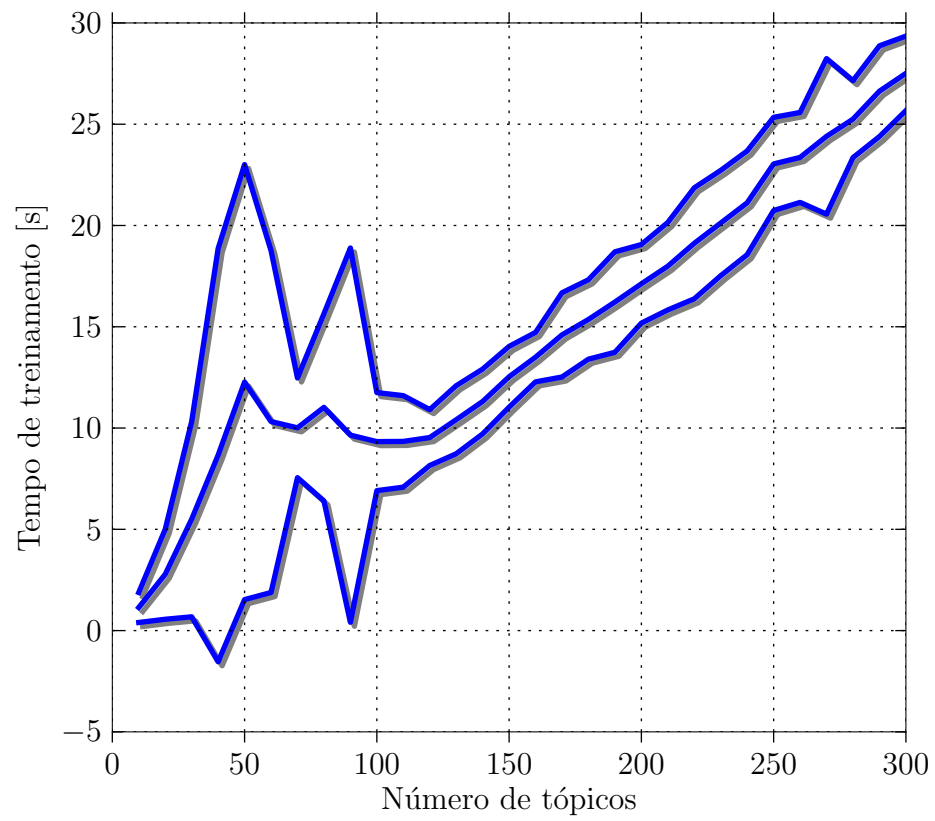


Figura 5.14: Tempos de treinamento para vários valores de tópicos para a base de imagens de retina, categoria Normais. Dicionário visual utilizado com 1200 palavras.

Método	BIC	SIFT	SURF	Patches
Mínimo	$0,43 \pm 0,09$	$0,18 \pm 0,09$	$0,22 \pm 0,06$	$0,3 \pm 0,1$
Inflexão	$0,61 \pm 0,05$	$0,40 \pm 0,06$	$0,25 \pm 0,07$	$0,41 \pm 0,07$
Máximo	$0,76 \pm 0,05$	$0,51 \pm 0,03$	$0,50 \pm 0,07$	$0,51 \pm 0,06$

Tabela 5.11: Médias e desvios padrões encontrados para três pontos na curva do número de imagens: 5, 14 e 54. Descritores individuais.

5.2.3 Experimento 3

Para o terceiro experimento, utilizando a base Free Foto Nature, apresentam-se os resultados para os quatro descritores na Figura 5.15. Na Figura 5.16 são mostrados os resultados para o método de votações na mesma escala dos resultados anteriores, a fim de facilitar a comparação entre eles. Porém, dado que a maior parte dos pontos se encontram concentrados no intervalo $> 90\%$, a Figura 5.17 mostra os mesmos dados, em escalas independentes.

Diferentemente do resultado obtido na Seção 5.2.1, o descritor BIC é o descritor que obtém maior taxa de sucesso individual. Essa constatação acompanha a intuição sobre a base, dado que há alta variabilidade de cor entre as imagens das diferentes categorias, como evidenciado pela Figura 5.2.

De fato, a total retirada da informação de cor, representada pela curva de votação com três classificadores, exceto o BIC, penaliza a performance do classificador final.

As Tabelas 5.11 e 5.12 mostram para os descritores e para a votação respectivamente, a performance do classificador em três pontos distintos da curva: o ponto *mínimo*, é o ponto correspondente ao número mínimo de imagens por categoria (5); o ponto *máximo*, é o ponto corresponde ao número máximo de imagens por categoria (54); enquanto o ponto de *inflexão*, é definido como o ponto no qual o número de imagens no conjunto de treinamento é igual ao número de imagens no conjunto de testes por categoria (14).

Em relação ao ponto mínimo, verifica-se que o ponto máximo tem performance consistentemente e significativamente maior, mesmo considerando o erro estatístico.

O ponto de inflexão, no entanto tem sua performance dentro do intervalo de erro para os melhores classificadores. Assim, pode-se concluir que é possível construir classificadores LDA com performance satisfatória utilizando-se de um número reduzido de imagens.

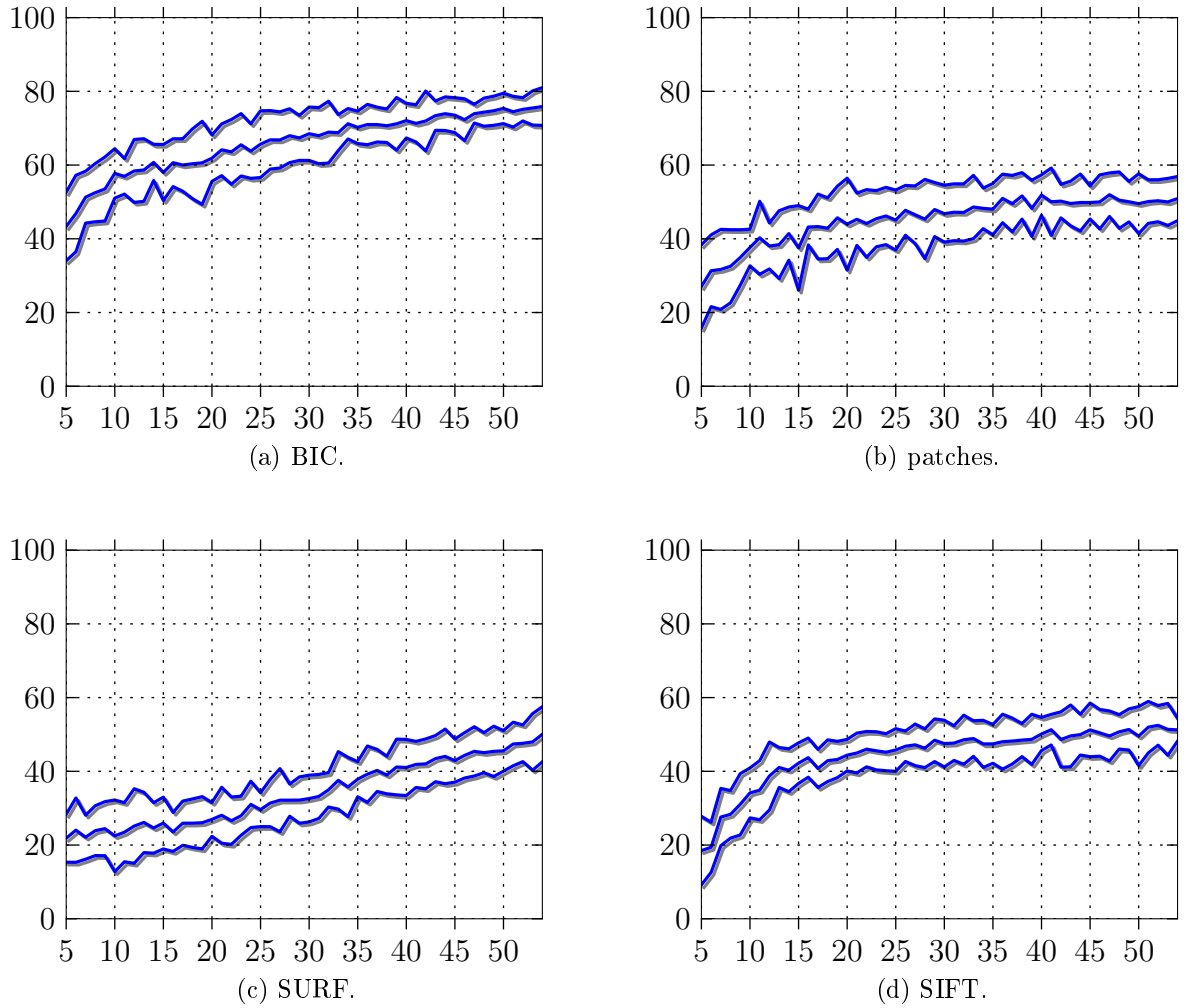


Figura 5.15: Taxas de acerto percentuais médias entre todas as categorias da base Free Foto Nature, em função do número de imagens de treinamento (por classe). As taxas são apresentadas separadamente para cada um dos descritores.

Método	Todos	Todos - BIC	Todos - SIFT	Todos - SURF	Todos - Patches
Mínimo	$0,89 \pm 0,07$	$0,35 \pm 0,08$	$0,88 \pm 0,06$	$0,86 \pm 0,07$	$0,89 \pm 0,07$
Inflexão	$0,96 \pm 0,02$	$0,50 \pm 0,07$	$0,94 \pm 0,03$	$0,95 \pm 0,02$	$0,96 \pm 0,02$
Máximo	$0,98 \pm 0,02$	$0,69 \pm 0,07$	$0,98 \pm 0,02$	$0,97 \pm 0,02$	$0,98 \pm 0,02$

Tabela 5.12: Médias e desvios padrões encontrados para três pontos na curva do número de imagens: 5, 14 e 54.

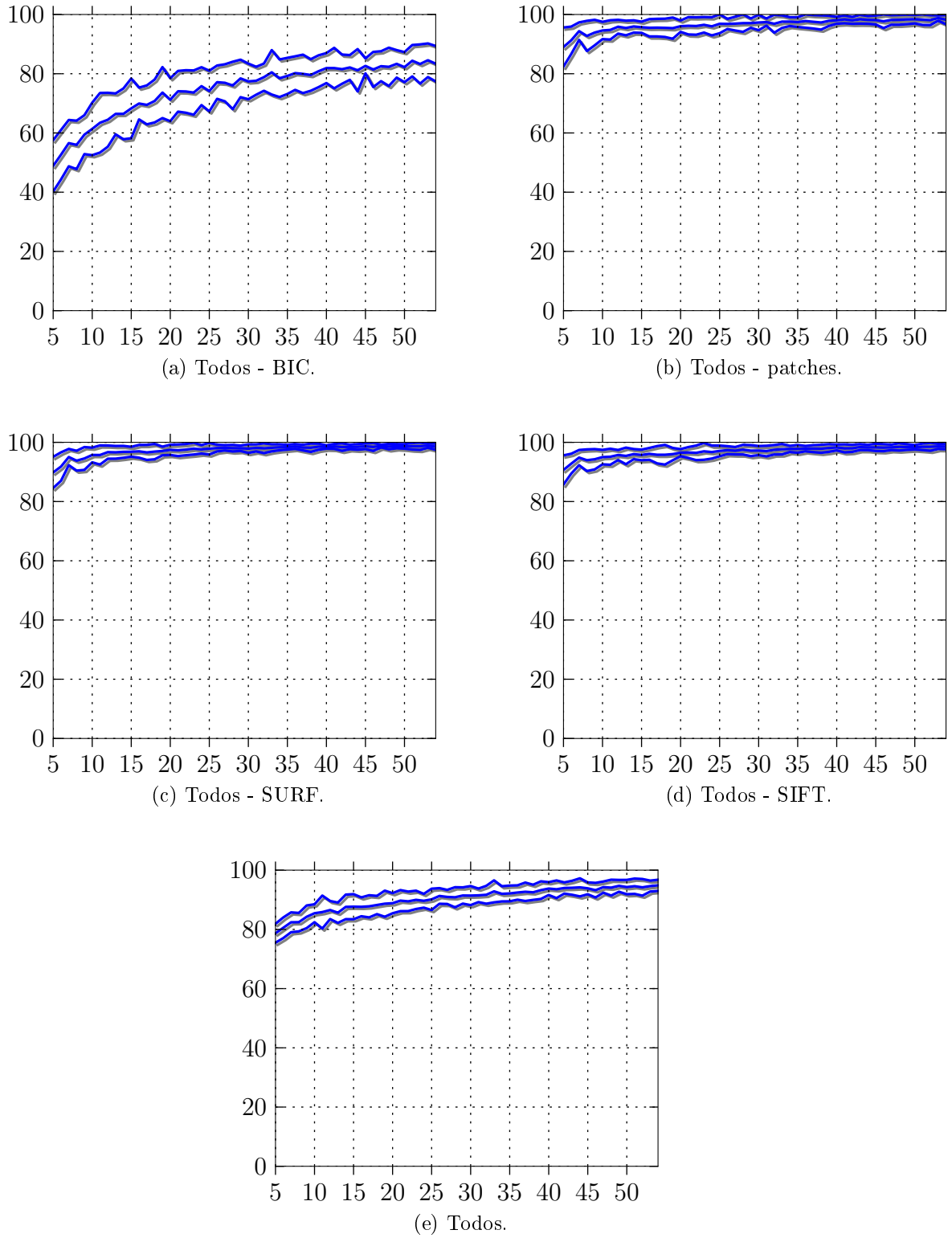


Figura 5.16: Taxas de acerto percentuais médias para as nove categorias da base Free Foto Nature, em função do número de imagens de treinamento (por classe), utilizando o método de votação. As taxas são apresentadas separadamente para cada conjunto de descritores presentes na votação.

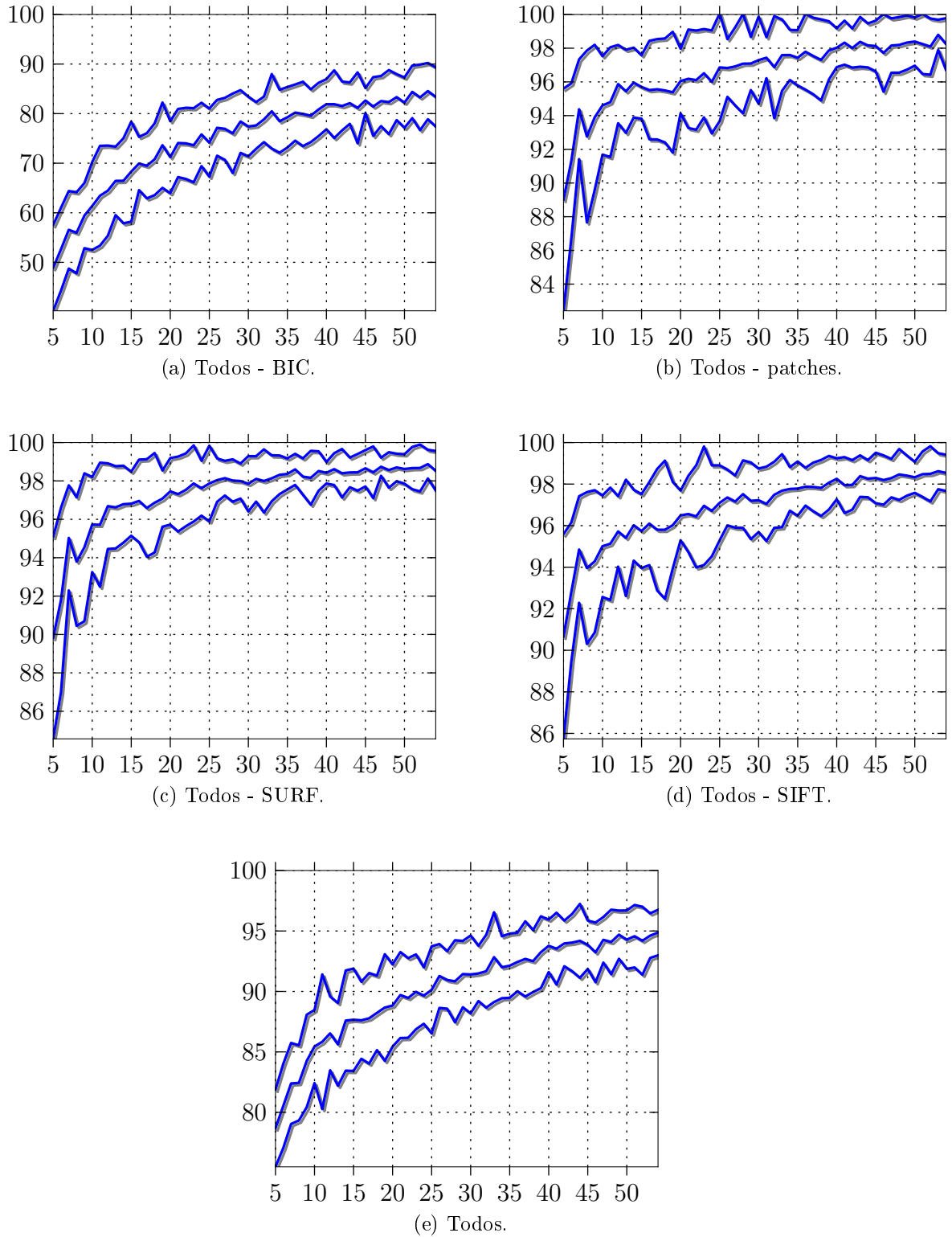


Figura 5.17: Taxas de acerto percentuais médias para as nove categorias da base Free Foto Nature, em função do número de imagens de treinamento (por classe), utilizando o método de votação. As taxas são apresentadas separadamente para cada conjunto de descritores presentes na votação. Cada descritor é apresentado em escala própria, para melhor visualização

Capítulo 6

Conclusões

Através dos experimentos realizados, concluímos que a técnica tem boa performance na tarefa de categorização de imagens, dado que algumas condições sejam satisfeitas. As classes nas quais deseja-se separar as imagens devem ser suficientemente distintas. Esta distinção entre as classes é uma função dos descritores utilizados na formação do dicionário de palavras visuais, de forma que diferentes descritores levam a resultados diversos nas taxas de acertos dos classificadores.

Para classes de imagens, associadas a um descritor particular, onde o conjunto de palavras visuais gerados para uma das classes, dita classe maior, seja um subconjunto de uma outra classe, dita classe menor, verificou-se que os classificadores baseados em LDA tendem a privilegiar a classe maior. Os classificadores construídos não são capazes de interpretar a falta de informação como evidência positiva da pertinência da imagem na classe menor.

Diferentes descritores capturam diferentes características das imagens utilizadas. Como resultado desse fato, a combinação de diferentes descritores em métodos de votação pode resultar em melhores taxas de acerto do que cada um dos descritores individualmente. Votações estritas, com exigência de maioria absoluta não apresentam resultados significativamente superiores aos descritores individuais. Ao relaxar a condição de maioria absoluta, permitindo que uma imagem seja considerada como pertencente a uma classe quando essa for a resposta da metade dos classificadores, obtém-se taxas de acerto altas, perto de 90 %. O relaxamento dessa condição, no entanto, só é aplicável para bases de imagens com grande número de classes e alta similaridade inter-classes, onde seja aceitável que uma imagem seja classificada como pertencente a mais de uma classe (ex: montanha, praia, etc).

A escolha dos melhores subconjuntos de descritores para compor os votantes afeta significativamente a performance final. O melhor subconjunto de descritores depende da base de dados utilizada, sendo também função direta da variabilidade intra-classes de

suas características. Por exemplo, bases de imagens onde a cor é fator determinante para a diferenciação entre as classes, tiveram melhores resultados quando descritores de cor foram incluídos no conjunto de votantes.

Verificou-se também que o modelo se comporta bem quando são utilizados dicionários visuais pequenos e poucos tópicos latentes na etapa de treinamento: não foi notada diferença significativa fora da margem estatística entre a precisão dos classificadores para os vários valores experimentados destes dois parâmetros. Uma vez que dicionários menores, bem como modelos com poucos tópicos latentes são mais treinados e avaliados em menor tempo, essa observação permite a construção de classificadores mais eficientes.

Referências Bibliográficas

- [1] *A Bayesian hierarchical model for learning natural scene categories*, volume 2, Washington, DC, USA, June 2005. IEEE Computer Society.
- [2] Milton Abramowitz and Irene A. Stegun. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. Dover, New York, ninth dover printing, tenth gpo printing edition, 1964.
- [3] Jurandy Almeida, Anderson Rocha, Ricardo Torres, and Siome Goldenstein. Making colors worth more than a thousand words. In *SAC '08: Proceedings of the 2008 ACM symposium on Applied computing*, pages 1180–1186, New York, NY, USA, 2008. ACM.
- [4] David Arthur and Sergei Vassilvitskii. k-means++: the advantages of careful seeding. In *SODA '07: Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1027–1035, Philadelphia, PA, USA, 2007. Society for Industrial and Applied Mathematics.
- [5] Hagai Attias and London Wcn Ar. Inferring parameters and structure of latent variable models by variational bayes. pages 21–30. Morgan Kaufmann Publishers, 1999.
- [6] Herbert Bay, Tinne Tuytelaars, Van Gool, and L. Surf: Speeded up robust features. In *9th European Conference on Computer Vision*, Graz Austria, May 2006.
- [7] Yoshua Bengio. Convergence properties of the k-means algorithms. In *Advances in Neural Information Processing Systems 7*, pages 585–592. MIT Press, 1995.
- [8] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, January 2003.
- [9] Avrim Blum and Tom Mitchell. Combining labeled and unlabeled data with co-training. In *COLT' 98: Proceedings of the eleventh annual conference on Computational learning theory*, pages 92–100, New York, NY, USA, 1998. ACM.

- [10] Eugene Charniak. Bayesian networks without tears. *AI MAGAZINE*, 12(4):50–63, 1991.
- [11] Chaitanya Chemudugunta, Padhraic Smyth, and Mark Steyvers. Modeling general and specific aspects of documents with a probabilistic topic model. In *NIPS*, pages 241–248, 2006.
- [12] Jie Cheng and Russ Greiner. Comparing bayesian network classifiers, 1999.
- [13] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley-Interscience, 99th edition, August 1991.
- [14] Ricardo da Silva Torres and Alexandre Xavier Falcão. Content-based image retrieval: Theory and applications. *Revista de Informática Teórica e Aplicada*, 13:161–185.
- [15] Scott C. Deerwester, Susan T. Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6):391–407, 1990.
- [16] James M. Dickey. Multiple hypergeometric functions: Probabilistic interpretations and statistical uses. *Journal of the American Statistical Association*, 78(383):628–637, 1983.
- [17] Piotr Dollár, Vincent Rabaud, Garrison Cottrell, and Serge Belongie. Behavior recognition via sparse spatio-temporal features. In *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS)*, Beijing, China, 2005.
- [18] Pedro Domingos and Michael Pazzani. Beyond independence: Conditions for the optimality of the simple bayesian classifier. In *Machine Learning*, pages 105–112. Morgan Kaufmann, 1996.
- [19] Pradheep K Elango and Karthik Jayaraman. Clustering images using the latent dirichlet allocation model, 2005.
- [20] L. Fei-Fei, R. Fergus, and P. Perona. One-shot learning of object categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(4):594–611, 2006.
- [21] Nir Friedman and Moises Goldszmidt. Building classifiers using bayesian networks. In *In Proceedings of the thirteenth national conference on artificial intelligence*, pages 1277–1284. AAAI Press, 1996.

- [22] Yul Gao and Jianping Fan. Incorporating concept ontology to enable probabilistic concept reasoning for multi-level image annotation. In *MIR '06: Proceedings of the 8th ACM international workshop on Multimedia information retrieval*, pages 79–88, New York, NY, USA, 2006. ACM.
- [23] T. Hastie, R. Tibshirani, and J. H. Friedman. *The Elements of Statistical Learning*. Springer, corrected edition, July 2003.
- [24] David Heckerman. A tutorial on learning with bayesian networks. Technical report, Learning in Graphical Models, 1996.
- [25] Thomas Hofmann. Unsupervised learning from dyadic data. pages 466–472. MIT Press, 1998.
- [26] Thomas Hofmann. Probabilistic latent semantic indexing. pages 50–57, 1999.
- [27] Qasim Iqbal and J. K. Aggarwal. Cires: A system for content-based retrieval in digital image libraries. In *in Invited Session on Content-based Image Retrieval: Techniques and Applications, 7 th International Conference on Control Automation, Robotics and Vision (ICARCV)*, pages 205–210, 2002.
- [28] Tommi Jaakkola and David Haussler. Exploiting generative models in discriminative classifiers. In *In Advances in Neural Information Processing Systems 11*, pages 487–493. MIT Press, 1998.
- [29] Finn V. Jensen. *Statistics for Engineering and Information Science*. Springer, 1st edition, 2001.
- [30] Tapas Kanungo, David M. Mount, Nathan S. Netanyahu, Christine D. Piatko, Ruth Silverman, and Angela Y. Wu. An efficient k-means clustering algorithm: Analysis and implementation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(7):881–892, July 2002.
- [31] Yan Ke and Rahul Sukthankar. Pca-sift: A more distinctive representation for local image descriptors. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 2:506–513, 2004.
- [32] S. Kullback and R. A. Leibler. On information and sufficiency. *Annals of Mathematical Statistics*, 22:49–86, 1951.
- [33] Ivan Laptev. On space-time interest points. *Int. J. Comput. Vision*, 64(2-3):107–123, 2005.

- [34] Ivan Laptev and Tony Lindeberg. Local descriptors for spatio-temporal recognition. In *In First International Workshop on Spatial Coherence for Visual Motion Analysis*, 2004.
- [35] Alberto Leon-Garcia. *Probability and Random Processes for Electrical Engineers*. Addison-Wesley Publishing Company, 1st edition, 1989.
- [36] Rose F. Liu and Rusmin Soetjipto. Learning on bayesian networks, 2004.
- [37] David G. Lowe. Object recognition from local scale-invariant features. *Computer Vision, IEEE International Conference on*, 2:1150–1157 vol.2, August 1999.
- [38] David G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, 2004.
- [39] Geoffrey J. McLachlan and Thriyambakam Krishnan. *The EM Algorithm and Extensions (Wiley Series in Probability and Statistics)*. Wiley-Interscience, 2 edition, March 2008.
- [40] T. Mitchell. *Machine Learning (Mcgraw-Hill International Edit)*. McGraw-Hill Education (ISE Editions), 1st edition, October 1997.
- [41] Richard E. Neapolitan. *Learning Bayesian Networks*. Prentice Hall, April 2003.
- [42] Juan Carlos Niebles, Hongcheng Wang, and Li Fei-fei. Unsupervised learning of human action categories using spatial-temporal words. In *In Proc. BMVC*, 2006.
- [43] S. James Press. *Bayesian Statistics: Principles, Models, and Applications*. John Wiley & Sons, 1st edition, 1989.
- [44] Rajat Raina, Yirong Shen, Andrew Y. Ng, and Andrew McCallum. Classification with hybrid generative / discriminative models. In *In Advances in Neural Information Processing Systems 16*. MIT Press, 2003.
- [45] Christian P. Robert. *The Bayesian choice : a decision-theoretic motivation / Christian P. Robert*. Springer-Verlag, New York :, 1994.
- [46] Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Prentice-Hall, Englewood Cliffs, NJ, 2nd edition edition, 2003.
- [47] Claude E. Shannon. *A Mathematical Theory of Communication*. CSLI Publications, 1948.

- [48] Linda G. Shapiro, George C. Stockman, Linda G. Shapiro, and George Stockman. *Computer Vision*. Prentice Hall, January 2001.
- [49] Arnold W. M. Smeulders, Marcel Worring, Simone Santini, Amarnath Gupta, and Ramesh Jain. Content-based image retrieval at the end of the early years. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(12):1349–1380, 2000.
- [50] Renato O. Stehling, Mario A. Nascimento, and Alexandre X. Falcão. A compact and efficient image retrieval approach based on border/interior pixel classification. In *CIKM '02: Proceedings of the eleventh international conference on Information and knowledge management*, pages 102–109, New York, NY, USA, 2002. ACM.
- [51] Ilkay Ulusoy and Christopher M. Bishop. Generative versus discriminative methods for object recognition. In *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2*, pages 258–265, Washington, DC, USA, 2005. IEEE Computer Society.
- [52] Remco C. Veltkamp and Mirela Tanase. Content-based image retrieval systems: A survey. Technical report, Department of Computing Science, Utrecht University, 2002.
- [53] Xing Wei and W. Bruce Croft. LDA-based document models for ad-hoc retrieval. In *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 178–185, New York, NY, USA, 2006. ACM.
- [54] Andrew P. Witkin. Scale-space filtering. In *8th Int. Joint Conf. Artificial Intelligence*, volume 2, pages 1019–1022, Karlsruhe, August 1983.
- [55] Sebastian Wrede, Marc Hanheide, Christian Bauckhage, and Gerhard Sagerer. An active memory as a model for information fusion. In *In Int. Conf. on Information Fusion, number 1*, pages 198–205, 2004.
- [56] Chengxiang Zhai and John Lafferty. A study of smoothing methods for language models applied to information retrieval. *ACM Trans. Inf. Syst.*, 22(2):179–214, 2004.